

# ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

## НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інженерії програмного забезпечення

**Пояснювальна записка**  
до бакалаврської роботи  
на ступінь вищої освіти бакалавр

на тему: «РОЗРОБКА ІНТЕРНЕТ-СЕРВІСУ ДЛЯ СПІЛКУВАННЯ МІЖ  
КОРИСТУВАЧАМИ МОВОЮ PYTHON»

Виконав: студент 4 курсу, групи ПД–42  
спеціальності

121 Інженерія програмного забезпечення

(шифр і назва спеціальності/спеціалізації)

Кряж І.М.

(прізвище та ініціали)

Керівник Штimmerман А.М.

(прізвище та ініціали)

---

Рецензент

---

(прізвище та ініціали)

# ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

## НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти -«Бакалавр»

Спеціальність підготовки – 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри  
Інженерії програмного  
забезпечення

Негоденко

О.В.

“ \_\_\_\_\_ ” \_\_\_\_\_ 2022 року

### ЗАВДАННЯ НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТА КРЯЖ ІЛЛЯ МАКСИМОВИЧ (прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка інтернет-сервісу для спілкування між користувачами»

Керівник роботи: Штиммерман А.М., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом вищого навчального закладу від «16» лютого 2022 року №22.

2. Строк подання студентом роботи «03» червня 2022 року

3. Вхідні дані до роботи

Інтернет-сервіс для спілкування між користувачами;

Науково-технічна література з питань, пов'язаних з програмним  
забезпеченням щодо інтернет-сервісів між користувачами;

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

4.1 Інтернет-сервіс для спілкування між користувачами

4.2 Система рекомендацій інтернет-сервісу.

4.3 Вимоги та оцінка якості інтернет-сервісу.

4.4 Опис проектування інтернет-сервісу.

4.5 Опис використаних технологій.

5. Перелік демонстраційного матеріалу (назва основних слайдів)

1. Актуальність проблеми

2. Огляд сучасних інтернет-сервісів для спілкування між  
користувачами

3. Принцип роботи інтернет-сервісу
4. Огляд існуючих засобів розробки інтернет сервісів для спілкування між користувачами
5. Огляд існуючих методів рекомендаційних систем
6. Діаграма прецедентів
7. Проектування бази даних

6. Дата видачі завдання «11» квітня 2022

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	11.04-14.04	
2	Аналіз вимоги до інтернет-сервісу	15.04-17.04	
3	Проектування інтернет-сервісу	18.04-21.04	
4	Створення бази даних	22.04-25.04	
5	Створення рекомендаційної системи	26.04-05.05	
6	Вступ, висновки, реферат	05.05-07.05	
7	Розробка обов'язкових демонстраційних матеріалів	07.05-10.05	
8	Попередній захист роботи	16.05-01.06	
9	Здача роботи	03.06	

Студент \_\_\_\_\_  
( підпис ) (прізвище та ініціали)

Керівник роботи \_\_\_\_\_  
( підпис ) (прізвище та ініціали)





## РЕФЕРАТ

Текстова частина магістерської роботи 75 с., 24 рис., 10 таблиці., 15 джерел.

### РОЗРОБКА ІНТЕРНЕТ-СЕРВІСУ ДЛЯ СПІЛКУВАННЯ МІЖ КОРИСТУВАЧАМИ МОВОЮ PYTHON

*Об'єкт дослідження* – процес рекомендацій тем у веб форумах на основі сучасних рекомендаційних систем.

*Предмет дослідження* – інтернет-сервіс для спілкування між користувачами.

*Мета роботи* – спрощення процесу спілкування між користувачами за рахунок користування веб-сервісу розробленого на мові Python.

*Методи дослідження* – методи проектування та розробки сучасних рекомендаційних систем, методи аналізу та збору інформації.

Для реалізації поставленої мети потрібно вирішити наступні завдання:

1. Провести аналіз веб-сервісів для спілкування між користувачами.
2. Розглянути програмне забезпечення, яке може бути використано для спілкування між користувачами.
3. Провести аналіз засобів розробки програмного забезпечення.
4. Спроекувати інтернет-сервіс для спілкування між користувачами та схему бази даних.
5. Розробити інтернет-сервіс для спілкування між користувачами
6. Розробити систему рекомендацій тем користувачу.

Загальною проблемою цих інтернет-сервісів є застаріла система пошуку інформації, що може зменшити популярність та притік нових користувачів до інтернет-сервісу.

Особливості методу є можливість рекомендацій користувачу розділов та тем інтернет-сервісу на основі алгоритмів сучасних рекомендаційних систем. За основу було взято мову Python та фреймворк Django з серверної сторони та HTML/CSS з клієнтської сторони. Сервіс працює з базами даних. В якості серверу бази даних, було взято Mysql.

Даний інтернет-сервіс може бути використано у всіх сферах, які потребують спілкування між користувачами в рамках мережі інтернет.

*Галузь використання – спілкування між користувачами.*

## ЗМІСТ

<b>ВСТУП.....</b>	<b>9</b>
<b>РОЗДІЛ 1 ІНТЕРНЕТ-СЕРВІСИ ДЛЯ СПІЛКУВАННЯ МІЖ КОРИСТУВАЧАМИ.....</b>	<b>12</b>
1.1 Принципи створення інтернет сервісу для спілкування між користувачами	8
1.2 Огляд сучасних інтернет-сервісів для спілкування між користувачами	20
<b>РОЗДІЛ 2. ПРОЕКТУВАННЯ ІНТЕРНЕТ-СЕРВІСУ ДЛЯ СПІЛКУВАННЯ МІЖ КОРИСТУВАЧАМИ.....</b>	<b>26</b>
2.1 Постановка задачі	26
2.2 Огляд існуючих засобів розробки інтернет сервісів для спілкування між користувачами	26
2.3 Огляд існуючих методів рекомендаційних систем	33
2.4 Проектування бази даних	46
<b>РОЗДІЛ 3 РОЗРОБКА ІНТЕРНЕТ-СЕРВІСУ.....</b>	<b>52</b>
3.1 Діаграма прецедентів	52
3.2 Розробка рекомендаційної системи	5
3.3 Функціонал інтернет-сервісу	66
3.4 Сторінка рекомендацій	71
<b>ВИСНОВКИ.....</b>	<b>73</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>74</b>



## ВСТУП

Актуальність роботи полягає у тому що веб-форуми залишаються досить популярними сервісами для отримання інформації. З зростанням популярності, на форумах з'являються багато розділів та тем, серед яких користувачам складно орієнтуватись. Впровадження рекомендаційних систем може зменшити час на пошук тем цікавих користувачу.

В епоху глобалізації та інформатизації населення планети та його освіти, виникла нагальна потреба. Щодня мільйони людей отримують і здобувають нові знання даних із глобальної мережі Інтернет, оскільки це значно спростило їхні механізми отримання інформації. Дешевість і мобільність нових технологій призвели до великої кількості користувачів Інтернету по всьому світу.

С розвитком технології мережі інтернет, у її користувачів з'явилася потреба у спілкуванні між собою та отримувати відповіді на запитання по самих різних темах. Унікальність форумів у тому, що на них можливо отримати відповіді на дуже специфічні теми, які можуть бути недостатньо розкриті на інших веб-ресурсах.

*Об'єкт дослідження* – процес рекомендацій тем у веб форумах на основі сучасних рекомендаційних систем.

*Предмет дослідження* – інтернет-сервіс для спілкування між користувачами.

*Мета роботи* – спрощення процесу спілкування між користувачами за рахунок користування веб-сервісу розробленого на мові Python.

*Методи дослідження* – методи проектування та розробки сучасних рекомендаційних систем, методи аналізу та збору інформації.

Для реалізації поставленої мети потрібно вирішити наступні завдання:

Майже всі сучасні популярні веб-форуми мають функціонал створення користувачами загальних та вложенних розділів форуму для того, щоб іншим користувачам було зрозуміло, який зміст цього розділу та про що в ньому йдеться

розмова. Цей функціонал створений для того, щоб структурувати веб-форум та користувачам було зручно шукати інтересують їх теми.

При збільшенні користувачів на форумі можуть бути кілька десятків тисяч різних або схожих між собою тем, та користувачу буде складно знайти саме той розділ, в якому він зацікавлений навіть при наявності на форумі фільтрів пошуку.

Рішенням цієї проблеми є можливість рекомендацій користувачу розділів та тем інтернет-сервісу на основі алгоритмів сучасних рекомендаційних систем.

*Галузь використання* - інтернет-сервіс може бути використано у всіх сферах, які потребують спілкування між користувачами в рамках мережі інтернет.

## 1 ІНТЕРНЕТ-СЕРВІСИ ДЛЯ СПІЛКУВАННЯ МІЖ КОРИСТУВАЧАМИ

На сучасному рівні розвитку інформаційних технологій використання комп'ютерів для зберігання будь-якого типу інформації стає єдиним інструментом, що надає численні можливості для управління інформацією. Інтернет відіграє важливу роль у процесі отримання інформації.

Сьогодні в Україні близько 9 мільйонів людей користуються Інтернет послугами різної частоти. На сьогоднішній день Інтернет став найдосконалішою інформаційною системою у світі, яка використовується для спілкування мільйонів користувачів.

Інтернет-сервіси для спілкування між користувачами є частиною всесвітньої павутини і часто використовуються для того, щоб задати запитання та отримати інформацію відповідей. Це популярні платформи для обміну інформацією та порівняння думок. Багато людей запитувати та отримувати інформацію з Інтернет-форумів на різні теми, багато з яких пов'язані з наукою і технології. Сюди входять здоров'я, косметика, технічні пристрої та екологічні проблеми.[1]

Задача яку потрібно вирішити в рамках цієї роботи це розробка системи навігації та пошуку нових та інтересуючих користувачів тем у великих веб-форумах.

Цільовою аудиторію в даному випадку будуть безпосередньо користувачі форуму на якому буде введена ця система навігації.

Розглянемо проблему більш детально.

Веб-форуми не повинні відразу мати розгалужену структуру. Користувачі форуму мають можливість самотужки створювати розділи та теми для форумів що робе кожен форум унікальним. Але потрібно пам'ятати що при збільшенні користувачів на форумі розділи та теми можуть зростати в геометричній прогресії. Та новим користувачам буде дуже важко шукати популярні або набираючі популярність розділу форуму щоб стати повноцінним членом

суспільства. Це може призвести к зниженню нових активних користувачів на форумі та в цілому згіршить зручність використання веб-форуму.

Гарним рішенням може стати системи рекомендації контенту були створені щоб підказувати користувачу інтересуючі його теми або товари на основі обробки інформації про його дії. А саме алгоритм колабораційний фільтрації яка на основі збігів однакових дій різних користувачів може може рекомендувати їм однакові теми.

Колаборативна фільтрація – це метод рекомендації, у якому аналізується лише реакція користувачів на об'єкти: оцінки, які виставляють користувачі об'єктам. Оцінки можуть бути як очевидними (користувач виставляє оцінки об'єктам), так і неявними (наприклад, кількість переглядів одного ролика). Чим більше оцінок збирається, тим точніше вийдуть рекомендації.

Методи Колаборативної фільтрації дають конкретні рекомендації користувача предмети, засновані на моделях оцінок або використання (наприклад, покупки) без потреби в екзогенній інформації ні про елементи, ні про користувачів. Зараз добре відпрацьовані методи працюють адекватно для багатьох цілей.[2]

Таким чином, користувачі допомагають один одному у фільтрації об'єктів, тому такий метод називається також спільною фільтрацією.

## 1.1 Принципи створення інтернет сервісу для спілкування між користувачами

Інтернет сервіс – сукупність веб сторінок та залежного вмісту, доступних у мережі Інтернет, які об'єднані як за змістом, так і за навігацією під єдиним доменним ім'ям.[3]

Для того щоб сайт був успішним фахівці виділяють кілька ключових моментів під час його створення.

Інтернет-сервіс має містити інформацію (контент). Чим більше конкретної інформації, тим легше зібрати велику аудиторію його відвідувачів. Хороший приклад такого підходу – це тематичні інформаційні ресурси, що містять велику кількість матеріалів певної тематики, які мають власне доменне ім'я (відповідне одержимою у ньому інформацією), створені фірмами окремо від своїх корпоративних сайтів та доменних імен. І вже рекламою та конкретними посиланнями з нього що залучають користувачів на свій "діловий" сайт. Така політика не лише приваблює та утримує аудиторію сайту вона допомагає його власникам просувати інші сайти.

Перше, з чого починають оцінку інтернет-сервісу, це його зовнішній вигляд, а вже Потім роблять висновки, вивчаючи вміст. Зовнішній вигляд – це перший контакт користувача з сервісом та якість (рівень) оформлення інтернет-сервісу прямо показує рівень поваги до користувача. Окрім акуратного виконання, існує ще безліч правил, які потрібно враховувати. Є технічні тонкощі, наприклад - вага (розмір) графічних файлів, виправданість застосування деяких технологій; режими завантаження сторінок. А є більш приховані: естетична цінність, правильність та сумісність колірного балансу, грамотність пропорцій та перспектив за обсягом та висвітленням. Плюс, трохи психології – правильне асоціативне сприйняття графічного оформлення, відповідно до тематики інтернет-сервісу та його завдань.

Хороший інтернет-сервіс завжди ретельно продуманий. Користувач повинен легко розуміти структуру сайту та специфіку пропозиції сервісу щодо навігації, а

інструменти навігації інтернет-сервісу мають бути зрозумілими. Добре, коли навігація враховує не лише розділи, але й зручність роботи з кожною сторінкою інтернет-сервісу. Хороший інтернет-сервісу не повинен мати "порожніх" сторінок з повідомленнями про роботи з наповнення.

Щоб користувачі постійно поверталися, інтернет-сервіс має бути "живим". Оновлення застарілої інформації та поповнення нової повинно проводитися регулярно. Часто це призводить до регулярних витрат - оплата кваліфікованого спеціаліста (web майстра) або регулярні платежі за підтримку сайту.

Існують різні браузерери і безліч їх версій, користувачів різні розміри моніторів і різна роздільна здатність екранів, є різні кодування тексту та різні шрифти. Інтернет-сервіс повинен вміти працювати в будь-яких умовах користувачів з будь-якими конфігураціями комп'ютерів. Хороший інтернет-сервіс не повинен утримувати зайвий код, а програмні рішення повинні працювати без помилок і "допомагати" звичайному користувачу помилятися.

Користувач завжди повинен мати можливість звернутися з проханням або питанням до адміністратора інтернет-сервісу. Для цього має бути легко доступні адреси електронної пошти, ICQ, номери телефонів та інше. Звичайно, сприятливе враження про інтернет-сервісу складається тільки в тому у випадку, якщо на задане питання було отримано швидко та кваліфікована відповідь. Різні програмні рішення на сайті можуть ще більше сприятиме підвищенню інтерактивності.

Більшість користувачів знаходять необхідну інформацію в пошукових системах та каталогах посилань. Хороший інтернет-сервісу має містити необхідні ключові слова та опис для пошукових систем, утримувати якісний контент, з урахуванням ефективного відгуку на запит користувача. А рекламна компанія, що проводиться, не повинна використовувати "спам".

У рекламній системі можуть крутитися різні банери, як за зовнішньому вигляду, і за змістом. Може виникнути безглузда ситуація, якщо поруч із вашим логотипом опиниться реклама конкурента чи сайту "вільного змісту". Крім того велика кількість рекламних банерів може викликати негативне ставлення до інтернет-сервісу.

Розміщення інтернет-сервісу – дуже важливе питання. Інформаційний ресурс, що не має свого доменного імені, не може викликати пошану. Розміщення сайту на безкоштовних серверах (хостингу) - привід задуматися про серйозність компанії. Ще одна тонкість – це пропускна здатність каналу у хост-провайдера та його віддаленість від потенційної аудиторії (наприклад, якщо сервер фізично знаходиться в іншій країні). Все це виливається у швидкість завантаження сайту. Ще один важливе питання - зручність та способи спілкування зі службою технічної підтримки. І, звичайно, список послуг та можливостей (бази даних, активні сторінки тощо), кількість місця та ціна, що надається.

Форум є досить серйозним інтерактивним програмним модулем. Його мета – забезпечити обмін думками користувачів та адміністрацію сайту. Насамперед потрібно створити спільноту, яка була б на цьому форумі та вело активний діалог. Спільнота – це об'єднання людей, мають спільні інтереси, цілі. Загалом, можна навести такі плюси створення форуму:

- 1) Регулярний зворотний зв'язок із споживачами.
- 2) Можливість розсилки важливих повідомлень.
- 3) Збір адрес електронної пошти потреб маркетингу.
- 4) Збір думок про нові продукти та послуги.
- 5) Сприяння визнанню вашого імені та підвищення довіри.
- 6) Можливість розміщення реклами.

Плюсів безперечно багато, проте є і суттєві мінуси:

- 1) Форум, особливо на початкових етапах свого розвитку, вимагає собі дуже пильної уваги та постійної роботи з ним.
- 2) Форум, так само як і сайт, необхідно постійно розвивати та просувати.
- 3) На формі можливі висловлення негативного характеру
- 4) Форум обов'язково повинен бути моделюваний, інакше він перетвориться на величезний "смітник" з не тематичних висловлювань і реклама.

До успішного форуму пред'являються досить жорсткі вимоги:

- 1) Наявність керівництва використання форуму доступне для читання з будь-якої сторінки;
- 2) Заповнення реєстраційної форми як обов'язкової умови для участі на форумі
- 3) Визначення модераторів, спираючись на їх особисті переваги та інтереси членів спільноти.

Також онлайнві спільноти можуть суттєво покращити Інтернет Представництво компанії, якщо форум використовується як частка бізнесу. Співтовариства можуть підвищити впізнаваність торгової марки та сприяти формуванню довіри до компанії, затримати відвідувачів на сайті, суттєво збільшити відвідуваність, більшити індекс цитованості, створити дух спільноти та товариства серед своїх членів.

При ретельному плануванні та реалізації, онлайнва спільнота може працювати практично самостійно без зовнішнього втручання та здатне значно підвищити відвідуваність більшості сайтів. Одним із різновидів форумів є форуми підтримки. Вони створюються компаніями, що постачають якісь складні товари або послуги для відповідей на численні запитання клієнтів щодо цих товарів та послуг. Найбільш поширені форуми підтримки серед розробників програмного забезпечення.

Очевидний плюс форуму підтримки над іншими технологіями (ICQ, телефон, email ...) в тому, що клієнт може знайти вже готову відповідь на свій питання і не турбувати підтримку. Таким чином економиться час і часто гроші компанії-постачальника.

Однак, крім функції безпосередньо підтримуй форум впливає ще і на маркетинг компанії, і навіть на розвиток бренду та бізнесу. створення форуму підтримки може стати великим кроком до створення спільноти клієнтів.

На відміну від форумів підтримки, форум спільноти не повинен відразу мати розгалужену структуру. Користувачі форуму мають можливість самотужки створювати розділи та теми для форумів що робе кожен форум унікальним.

Важливо стежити за дотриманням форуму. Необхідно щоб розділи та теми форуму були легко зрозумілими та відвідувачі могли легко знайти відповіді на



запитання. Якщо відвідувач відкриває потрібний розділ, а там питання упереміш з різних тем, він природно воліє поставити своє запитання, а чи не шукати інформацію на інших ресурсах.

Якщо користувач залишив своє питання у розділі форуму, що не відповідає за тематикою, перенесіть його у потрібний розділ, залишивши посилання на старому місці. А через деякий час видаліть це посилання.

Якщо на форумі багато нових тем і питань, логічно припустити, що форум користуються популярністю. На форумах суспільства розділи і теми форуму формують безпосередньо самі користувачі щоб відокремити свої питання або спільні теми від інших. Так користувачам більш зручно спостерігати за ходом бесіди та не відволікатися на інші розмови на формі. Як би на форумах всі теми та питання були в одному місці. Це призвело би до того що користувачі швидко заплутались або не змогли би знайти відповідь на своє питання, наприклад як у сучасних месенджерах де на велику групу людей виділяється всього один чат. Перевага форумів у тому що спілкування людей відбувається більш розділено та структуровано.

Потрібно пам'ятати що при збільшенні користувачів на форумі розділи та теми можуть зростати в геометричній прогресії. Та новим користувачам буде дуже важко шукати популярні або набираючі популярність розділу форуму щоб стати повноцінним членом суспільства. Це може призвести к зниженню нових активних користувачів на форумі.

Системи рекомендації контенту були створені щоб підказувати користувачу інтересуючі його теми або товари на основі обробки інформації про його дії.

Гарним прикладом може стати алгоритм колабораційний фільтрації яка на основі збігів однакових дій різних користувачів може рекомендувати їм однакові теми.

## 1.2 Огляд сучасних інтернет-сервісів для спілкування між

Програмне забезпечення для веб-додатків для спілкування між користувачами можуть бути описані наступним переліком[4]

- Табличні
- Деревоподібні
- Універсальні

Табличні форуми мають суворий лад у постах відвідувачів форуму. Повідомлення у темах створюють стрічки, тому табличні форуми ще називають стрічковими. Для визначення адресата та адресанта відповіді, для діалогу з ними доступна панель інструментів, а саме звернення до автора, цитування та багато іншого. Завдяки суворій ієрархії у форумі з такою структурою користувачам спілкування не завдає труднощів. Натиснувши одну з тем форуму, можна побачити всі повідомлення даного топіка. Найпопулярніші теми знаходяться у топі, на перших місцях. Рейтинг створюється за рахунок підрахунку повідомлень та перегляду.

Одним з прикладів такого форуму є «Харків Форум». На основній сторінки має стилізовану таблицю з списком розділів. Таблиця поділяється на п'ять стовпів в яких вказано статус розділу, назва розділу, останній допис, кількість тем, кількість дописів. Форум має просту систему текстового пошуку по темам, або дописам. При виборі розділу та теми, відкривається сторінка з стрічкою повідомлень де є можливість залишати свої повідомлення та цитувати повідомлення інших учасників теми.

Розділ	Останній допис	Тем	Дописів
<b>Харків</b>			
Головний (Присутніх - 481) Головний Харківський форум	Новий виток войны России против Украины... від KRlpa Сьогодні 00:07	79524	8568501
Жіночий форум (Присутніх - 26) Тихий куточок	сиде, мнго, за долларом и тонкие шутки... від Keelem Вчора 23:18	16927	3251260
Чоловічий форум (Присутніх - 26) Яхти і шлюпки	Мужики, как оно там, когда за 40? від Стасичка Вчора 18:43	8238	2211475
Новини (Присутніх - 80) Новини Харкова, України, світу	Росіяни одночасно відкрили вогонь по... від 057.ua Вчора 22:31	259745	3346083
Наше місто (Присутніх - 3) Обговорюємо петиції і проблеми нашого міста. Екологія нашого регіону	Парк Горького - переименовать! від jasmijn_852 Вчора 20:43	1161	59364
Політика (Присутніх - 20) Всі політичні дебати тут	Зеленский как президент. Отзыви від ashim1963 Вчора 22:20	22680	2717992
Студентський форум (Присутніх - 1) Про надходження, про навчання і про студентське життя в Харкові	Посоветуйте курсы по Английскому від Негоциант Вчора 08:19	2802	79639
Анонси (Присутніх - 2) Скоро в Харкові! Концерти, фестивали, виставки, змагання в Харкові	Вместе на выставку, в галерею, в музей... від beg 28.04.2022 22:36	1203	6886
Бюро знахідок Харкова. Пошук людей (Присутніх - 5) Бюро знахідок Харкова. Пошук людей	Україн паспорт (Новоожаново) від Tlanka 28.04.2022 19:11	6308	41891

Рис. 1.1 – Приклад табличного форуму

Переваги використання табличного форуму:

- Висока популярність завдяки простоті спілкування.
- Мають строгий порядок у постах відвідувачів форуму.

До недоліків використання можна віднести наступне:

- Складно контролювати повідомлення із за великого трафіку.
- Для контролю за повідомленнями можуть бути залучені люди та самі користувачі форуму що може негативно вплинути на якість роботи контролю за повідомленнями.

Веб-форуми з деревоподібною структурою мають строгую ієрархію повідомлень, але іншого плану, ніж з табличною. У таких форумах все чітко і конкретно, у пості завжди вказується, кому надсилається повідомлення. Легко можна простежити, хто з ким спілкується та до якої відповіді ставиться той чи інший коментар. Повідомлення мають вигляд дерева, кожна відповідь та наступні коментарі можуть створити нову гілку обговорення.

Гарним прикладом деревоподібною форуму є веб-форум «Reddit».

«Reddit» — це розважальний, новинний онлайн-сервіс, також інтернет-ЗМІ, де зареєстровані користувачі можуть додавати свій контент, такий як текстові пости або прямі посилання й обговорювати їх. Як і багато інших подібних сайтів, Reddit підтримує систему голосування за вподобані повідомлення — найбільш популярні з них з'являються на головній сторінці сайту. Також найпопулярніші пости показуються угорі відповідних категорій. Контент організовано за допомогою осередків за інтересами, які зветься «subreddits». Тематика сабрєдїтів досить широка: новини, ігри, кіно, музика, книги, спорт, їжа, фотографії тощо.[5]

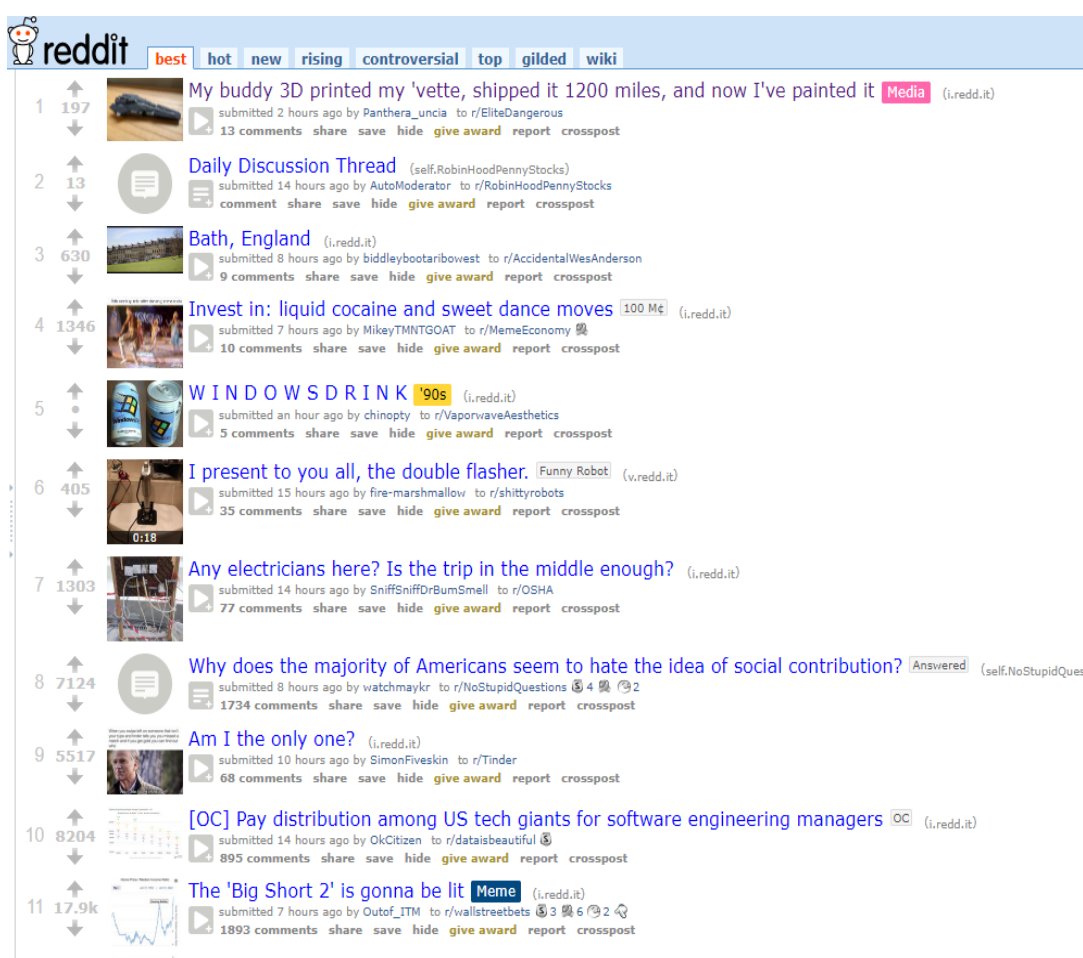


Рис. 1.2 – Приклад деревоподібного форуму

Переваги використання табличного форуму:

- Можливість стежити за спілкуванням між учасниками та відповідями, які вони залишають один одному.

- Учасники мають можливість ігнорувати певні відгалуження, що не заважає загальному спілкуванню.

До недоліків використання можна віднести наступне:

- Складна структура
- Незручності перегляду повідомлень, для чого потрібно натискати по кожному заголовку.

Універсальні форуми мають позитивні та негативні сторони кожного з перерахованих вище типів. На них дуже зручно спілкуватися та переглядати теми, за бажання можна перемикатися між типами. Але всі знають, що все універсальне не може бути надто хорошим у чомусь, як і надто поганим. З позитивних сторін виділяється можливість вибирати найбільш зручний режим роботи, а з мінусів досить висока складність.

Веб-форум «4chan» гарний приклад універсального веб-форуму. Який уособлює в собі якості табличних форумів, таких як чітка структура розділів та тем, і включає риси деревоподібних форумів, таких як строгу ієрархію повідомлень. На веб-форумі є таблиця з розділами і темами. При виборі однієї з тем відкривається стрічка повідомлень, але на відміну від табличного форуму в повідомленнях вказуються відправник та одержувач.



Рис. 1.3 – Приклад універсального форуму

Зведені результати аналізу характеристик розглянутих додатків наведено у таблиці 1.1

Таблиця 1.1 – Зведені результати аналізу характеристик веб-додатків для спілкування між користувачами.

<b>Назва</b>	<b>Харків Форум</b>	<b>Reddit</b>	<b>4chan</b>
<b>Платформи</b>	web	web, мобільний додаток	web
<b>Ієрархія розділів</b>	+	+	+
<b>Система пошуку</b>	+	+	+
<b>Модерація</b>	+	+	-
<b>Реклама</b>	+	-	-
<b>Цитування повідомлень</b>	+	-	-
<b>Можливість додавати медіа контент у повідомлення</b>	-	+	+
<b>Системи рекомендацій</b>	-	-	-
<b>Інші особливості</b>	Складний пошук інформації через неможливість встановити одержувача в стрічки повідомлень	Наявність системи заохочення користувачів через активність на форумі	Відсутність будь якої модерації та цензури

## **2 ПРОЕКТУВАННЯ ІНТЕРНЕТ-СЕРВІСУ ДЛЯ СПІЛКУВАННЯ МІЖ КОРИСТУВАЧАМИ**

### **2.1 Постановка задачі**

Ціллю дипломної роботи є розробка інтернет сервісу для спілкування між користувачами.

Незалежно від тематики інтернет сервісу відвідувачам сайту повинна представлятися можливість реєстрації та авторизації, знаходити цікаві їм розділи форуму на базі сучасної системи рекомендацій, продивлятися їм, та доповнювати своїми повідомленнями. Таким чином розроблений інтернет сервіс дозволяє користувачам знаходити або отримувати відповіді на свої питання, спілкуватися між собою та об'єднуватися в групи.

Аби добитися поставленої цілі необхідно вирішити наступні задачі

- 1) зробити огляд та проаналізувати існуючі засоби розробки інтернет сервісів для спілкування між користувачами.
- 2) спроектувати клієнт-серверну архітектуру інтернет-сервісу
- 3) розробити базу даних
- 4) розробити алгоритм рекомендаційної системи
- 5) реалізувати інтернет сервіс

### **2.2 Огляд існуючих засобів розробки інтернет сервісів для спілкування між користувачами.**

Для реалізації інтернет сервісів можуть використовуватися такі технології:

- мова програмування Python
- мова гіпертекстової розмітки (HTML), застосовується для розробки веб сторінок
- сценарії (скрипти), що готуються на різних скриптових мовах (JavaScript, TypeScript) та вставляються в Web-документ;



- Бази даних які використовуються для доступу та збереження інформації на формі(MySQL, PostgreSQL);
- Фреймворк Django, розроблений на мові програмування Python
- Бібліотеки штучного інтелекту розроблені на мові Python

Охарактеризуємо перелічені технології.

Мова програмування Python була розроблена у 1980-х співробітником голландського інституту CWI Гвидо ван Россумом. Мова почала вільно поширюватися через Інтернет і сподобалась іншим програмістам. З 1991 року Python є цілком об'єктно-орієнтованим. Python також запозичив багато рис таких мов, як C, C++, Modula-3 та Icon, і окремі риси функціонального програмування з Ліспу. На сьогоднішній час Python входить до 5 найпопулярніших мов програмування.

Однією з переваг Python є портованість, Python працює на всіх відомих платформах від КПК до мейнфреймів. Також мова портується на Microsoft Windows та на всі дистрибутиви Unix систем.

Python це інтерпретована та об'єктно-орієнтована мова програмування високого рівня з строгою динамічною типізацією. Динамічне зв'язування, семантика та простий синтаксис роблять цю мову дуже зручною для швидкої та результативної розробки.

На відміну від PHP бібліотеки Python доступні як у скомпільованій так і у вихідній формі. Інтерпритатор може бути расширений типами даних на C та C++ що може значно прискорює швидкість роботи програм.

Python був створений для використання вченими, має простий та елегантний синтаксис. Та потужні функції для роботи з комплексними числами та розробки штучного інтелекту.

Мова гіпертекстової розмітки (HTML) мова розмітки гіпертекстових документів. Текстова мова розмітки, призначена для розмітки документів, що містять текст, зображення, гіперпосилання, тощо.

Розмітка документів – це опис різних фрагментів документа та їх взаємного розташування. Розмітка здійснюється за допомогою тегів мови HTML.

Переглянути розмітку HTML-документа можна, якщо відкрити його у будь-якому текстовому редакторі (наприклад Блокнот). Щоб переглянути HTML-код у веб-браузері, клацнути правою кнопкою миші на документі та обрати команду «Переглянути джерело сторінки». Створювати та працювати з HTML-документами (файлами) можна і без доступу до мережі Інтернет. Під час роботи HTML-документ може бути відкритий у текстовому редакторі та у браузері одночасно. Після збереження змін у коді, потрібно оновити сторінку браузера, на якій відкрито цей же документ, щоб переглянути HTML-документ.

Сценарії (скрипти), написані мовами JavaScript, TypeScript використовують для динамічного керування компонентами Web-документа. Ці мови є мовами типу, що інтерпретується, тобто, виконуваний файл не створюється, а код виконується безпосередньо в процесі інтерпретації. Інтерпретація та виконання сценаріїв здійснюється браузером і Web-сервером. Частина сценарію може виконуватися під час завантаження документа, а частина сценарію, реалізована, як правило, у вигляді функцій виконуватися у відповідь на дії користувача.

База даних — це певний набір даних, пов'язаних між собою спільною ознакою і властивістю, та впорядковані, наприклад, за алфавітом.

Об'єднання великої кількості даних в єдину базу дає можливість для формування безлічі варіації групування інформації — особисті дані клієнта, історія замовлень, каталог товарів та інше.

Головним перевагою БД є швидкість внесення та використання потрібної інформації. Завдяки спеціальним алгоритмом, які використовуються для баз даних, можна легко знаходити необхідні дані всього за кілька секунд. Також у базі даних існує певна взаємозв'язок інформації: зміна в одній рядку може спричинити зміни в інших рядках — це допомагає працювати з інформацією простіше і швидше.

Щоб створити запит до бази даних, часто використовують Structured Query Language. SQL дає змогу додавати, редагувати та видаляти інформацію, що міститься в таблицях. Під час програмування сайтів використовують різні системи управління БД. До основних СУБД відносять:

- об'єктно-реляційна система управління базами даних Oracle Database;
- вільна система управління базами даних PostgreSQL;
- система управління базами даних Microsoft SQL Server;
- вільна система управління базами даних MySQL;

Такі системи управління відрізняються централізованою обробкою запитів, що забезпечують надійність, доступність та безпеку БД. Найбільш популярною системою управління є MySQL, вона дає зручний доступ для управління БД та підтримує велику кількість таблиць різних типів.

Django — це високорівневий веб-фреймворк Python, який заохочує швидкий розвиток і чистий, прагматичний дизайн. Створений досвідченими розробниками, він вирішує велику частину клопоту веб-розробки, тому ви можете зосередитися на написанні свого додатка без необхідності винаходити велосипед. Це безкоштовно та з відкритим вихідним кодом. Django є надзвичайно популярним та повнофункціональним серверним веб-фреймворком, написаним на Python. Фреймворк був розроблений, щоб пришвидшити розробку та якнайшвидше переходити додатки від концепції до завершення. Фреймворк Django включає десятки додаткових функцій, які можна використовувати для виконання звичайних завдань веб-розробки. Django займається аутентифікацією користувачів, адмініструванням вмісту, картами сайту, RSS-каналами та багатьма іншими завданнями функціоналом який вже включен у фреймворк. Фреймворк Django серйозно ставиться до безпеки і допомагає розробникам уникнути багатьох поширених помилок безпеки, таких як ін'єкція SQL, міжсайтові сценарії, підробка міжсайтових запитів і клікджекінг. Його система аутентифікації користувачів забезпечує безпечний спосіб керування обліковими записами та паролями користувачів. Деякі з найзавантаженіших сайтів на планеті використовують здатність Django швидко й гнучко масштабуватися, щоб задовольнити найвищі вимоги трафіку.

Слід зазначити, що важливі можливості мов програмування приблизно однакові. Усі мови підтримують розробку об'єктно-орієнтованих програм. Мова програмування C++ має значно більш поглиблену підтримку низькорівневих

програмних модулів проти інших. Мова C# має обмежену підтримку розробку об'єктно-орієнтованих програм коду на Unix системах, проте це компенсується Jit компіляцією та динамічним розвитком мови з боку розробника, компанії Microsoft. В цілому ж за функціональністю ці мови досить схожі і відрізняються в основному синтаксичними конструкціями уявлення подібних операцій;

Значно важливішим фактором, що впливає на загальну вартість створення Web сайту, є набір готових бібліотек для даної мови програмування.

Тут виділяється система програмування Perl своїм глобальним структурованим архівом відкритих текстів CPAN (Comprehensive Perl Archive Network). Для інших систем програмування бібліотеки поширюються або у вигляді набору сторінок, що настроюються, або у вигляді прикладів вихідних текстів, які згодом повинні бути піддані процедурі переробки та інтеграції у проект.

Слід також звернути увагу на важливу ідею використовувати безкоштовні вихідні тексти модулів у комерційних проектах, оскільки їх надійність ніким не гарантується, і, можливо, вартість тестування та переробки "під себе" такого модуля виявиться вище за вартість розробки з нуля. Слід зазначити помітну обмеженість функціональності стандартних бібліотек, що поставляються Microsoft у комплекті із засобами розробки. Помітно позначається відсутність вбудованих можливостей щодо формуванню графічних зображень, обробці XML та багато чого іншого. Проте це з лишком компенсується можливістю роботи з COM і COM+ об'єктами (що необхідно, наприклад, для роботи документами MS Office на стороні сервера) - в Python та Perl на операційних системах Linux, FreeBSD така функціональність принципово неможлива.

При розробці системи розмежування прав доступу, обмеженість її концептуальної основи визначається більшою мірою операційною системою та програмним забезпеченням Web-сервера, ніж використовуваними системами програмування. Пояснюється це тим, що зв'язка Internet Information Server і Microsoft Windows крім стандартної авторизації надають можливість

використання стандартних технологій реєстрації в домені Windows, що є актуальним для простих схем розмежування прав доступу (наприклад, загальна - адміністративна частина сайту тощо).

Як результат вищепереліченим особливостям мов програмування далі наведено зведену таблицю, в якій зведено разом всі переваги та недоліки мов програмування в області створення веб-додатків.

З урахуванням всього вищевикладеного для реалізації форуму виберемо технологію динамічного форматування сторінок, заснованої на використанні мови Python та бази MySQL. Як сервер більшість джерел [6-8] рекомендують використовувати Nginx.

Таблиця 2.1 – Коротке порівняння вибору платформи для web

Характеристика	C++	Java	Python	Perl	C#
швидкість роботи програмного забезпечення	дуже висока	висока	висока	середня	висока
Кросплатформенність	+	+	+	+	+
Надійність	Низька	Висока	Висока	Висока	Висока
Безпека	Низька	Висока	Середня	Середня	Висока
Масштабованість	Низька	Висока	Висока	Низька	Висока
Готові бібліотеки	Багато	Дуже багато	Багато	Багато	Багато
Вартість	Безкоштовна	Висока	Безкоштовна	Безкоштовна	Висока
Складність розробки	високий	високий	середній	середній	високий
Супровід та розширення функціональності іншими людьми	складно (висока ймовірність помилок)	складно (Потрібен високий рівень)	легко	висока ймовірність написання складно підтримуваного коду	складно (Потрібен високий рівень)
Додатково	Краще використати тільки для "внутрішніх", критичних до швидкості частин.	При виборі Java краще всю систему спочатку писати з її використанням	Найбільш популярна та легка в використанні система	Perl використовує вузьке коло профі. Відмінні можливості по роботі із текстами.	Краще працює на інфраструктурі компанії Microsoft.

## 2.3 Огляд існуючих методів рекомендаційних систем

Рекомендаційна система – система, яка дає рекомендації користувачеві. Рекомендаційна система дозволяє користувачеві помічати свої переваги і повертає результати, які корисні для нього, ґрунтуючись на оцінках інших користувачів та припущеннях самої системи. На відміну від пошукових систем, щоб отримати відповідь, рекомендаційна система не потребує чіткого запиту. Користувачеві пропонується оцінити деякі об'єкти з колекції та на підставі його оцінок та порівняння їх з оцінками інших користувачів будуються припущення та повертаються найближчі до них результати.

У кожній рекомендаційній системі ми маємо справу з користувачем, якому надається багато альтернатив, серед яких йому необхідно здійснити свій вибір. Користувачеві може не вистачати досвіду та знань для того, щоб самостійно відкинути альтернативи, які не відповідають його потреб. Користувач у певній формі, явно чи неявно, надає системі інформацію про свої переваги, при цьому про деякі альтернативи він може навіть і не знати. Таким чином, рекомендаційна система представляється як система, яка використовує певний тип фільтрації та існуючі відомості про потреби користувача, для рекомендації йому набору альтернатив, які вважає найбільш корисними йому [10].

Для розробки рекомендаційної системи використовують 4 типи фільтрації:

- 1) заснована на контенті (Content base);
- 2) колаборативна (Collaboration);
- 3) заснована на знаннях (Knowledge base);
- 4) гібридна (Hybrid) [11].

Контентна фільтрація формує рекомендацію на основі поведінки користувача. Наприклад, цей підхід може використовувати ретроспективну інформацію про перегляди (які блоги читає користувач та характеристики цих блогів). Якщо будь-який користувач зазвичай читає статті про Linux або регулярно залишає коментарі в блогах з проектування програмного забезпечення, то контентна фільтрація може використовувати цю ретроспективну інформацію для

виявлення подібного контенту та пропозиції такого контенту як рекомендований для цього користувача (статті в блогах по Linux або в інших блогах із проектування програмного забезпечення). Цей контент може бути визначений у ручному режимі або витягнутий автоматично на основі інших методів подібності [12].

Колаборативна фільтрація виробляє рекомендації на основі моделі попереднього поведінки користувача. Ця модель може бути побудована виключно на основі поведінки даного користувача або що більш ефективно - з урахуванням поведінки інших користувачів зі подібними характеристиками. У тих випадках, коли колаборативна фільтрація приймає до уваги поведінка інших користувачів, вона використовує знання про групу (group knowledge) для вироблення рекомендацій на основі подібності користувачів.

По суті, рекомендації базуються на автоматичному співробітництві безлічі користувачів і на виділенні (методом фільтрації) тих користувачів, які демонструють схожі переваги чи шаблони поведінки.

Як приклад припустимо, що створюється веб-сайт, що пропонує його відвідувачам рекомендації щодо блогів. На основі інформації від багатьох користувачів, які підписуються на блоги та читають їх, можна згрупувати цих користувачів за їх перевагами. Наприклад, можна об'єднати в одну групу користувачів, які читають кілька одних і тих ж блогів. За цією інформацією ідентифікуються найпопулярніші блоги серед тих, що читають учасники цієї групи. Потім - конкретному користувачу цієї групи - можна порекомендувати найпопулярніший блог з тих, куди він ще не підписаний і які не читає [12].

В основному системи, в яких для отримання рекомендацій використовуються отримані якимось чином знання, найчастіше ці знання додаються вручну. Наприклад, користувачі онлайн-магазину можуть вказувати товару інші схожі товари. На основі цих даних і створюються рекомендації [20].

Гібридні підходи поєднують колаборативну та контентну фільтрацію. Такі підходи підвищують ефективність (і складність) рекомендацій. систем. Об'єднання результатів колаборативної та контентної фільтрації потенційно дозволяє



підвищити точність рекомендації. Крім того, гібридний підхід може бути корисний, якщо застосування колаборативної фільтрації починається при значній розрідженості даних.

Гібридний підхід дозволяє спочатку зважувати результати згідно контентної фільтрації, а потім зміщувати ці ваги у напрямку до колаборативної фільтрації (у міру "дозрівання" доступного набору даних за конкретним користувачем) [12].

У рекомендаційних механізмах можуть бути використані різні алгоритми. Отримані результати можуть відрізнятися залежно від проблеми, для вирішення якої спроектований конкретний алгоритм, та відносин, які присутні у даних. Багато з цих алгоритмів прийшли з галузі машинного навчання (яка, у свою чергу, є підобласть штучного інтелекту), яка займається алгоритмами для навчання, прогнозування та прийняття рішень [12].

Метод, що обчислює евклідову відстань використовується в колаборативних рекомендаційних системах і дозволяє визначити, наскільки схожі уподобання користувачів шляхом обчислення коефіцієнта подібності (або оцінки подібності). Один із найпростіших способів обчислення оцінки подоби – це евклідова відстань. І тут предмети, які оцінювали, представляються як координатних осей. Тепер у цій системі координат можна розташувати точки, які відповідають людям, і подивитися, наскільки вони виявилися близькими (рис. 1.1).

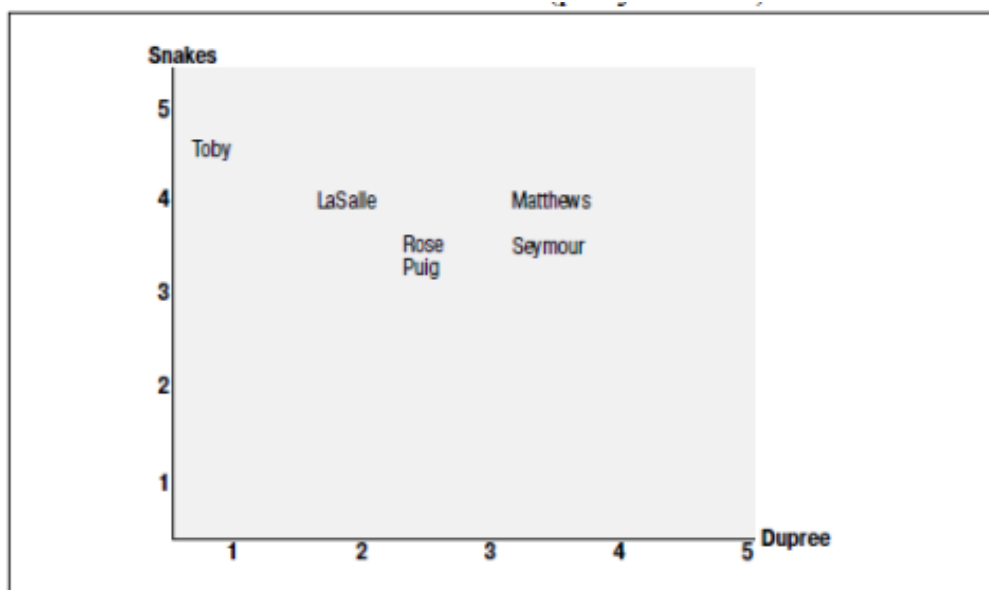


Рис. 2.1 – Люди у просторі переваг

На цьому малюнку показані люди, представлені точками у просторі переваг. Toby має координату 4,5 по осі Snakes та координату 1,0 по осі Dupree. Чим ближче дві людини у просторі переваг, тим більше схожі їх переваги. Оскільки ця діаграма двовимірною, то одночасно можна дивитися тільки на два показники, але принцип залишається тим самим і для більшої кількості показників.

Щоб обчислити відстань між Toby та LaSalle на цій діаграмі, візьмемо різниці координат по кожній осі, зведемо їх у квадрат, складемо, а потім витягнемо квадратний корінь із суми(2.1):

(2.1)

$$\sqrt{(4.5-4)^2+(1-2)^2}=1.11803399$$

Відстань, обчислена за цією формулою, буде тим меншою, чим більше подібності для людей. Проте нам потрібна функція, значення якої більше, ніж люди більш схожі один на одного. Цього можна досягти, додавши до обчислень відстані 1 (щоб ніколи не ділити на 0) і взявши зворотну величину (2.2):

(2.2)

$$\frac{1}{1+\sqrt{(4.5-4)^2+(1-2)^2}}=0.47213595$$

Нова функція завжди повертає значення від 0 до 1, причому 1 виходить, коли переваги двох людей точно збігаються [13]

Трохи складніший спосіб визначити ступінь схожості інтересів людей дає коефіцієнт кореляції Пірсона. Коефіцієнт кореляції – це міра того, як добре два набори даних лягають на пряму.

Формула складніше, ніж для обчислення евклідової відстані, але вона дає кращі результати, коли дані погано нормалізовані, наприклад, якщо деякі критик стійко виставляє фільмам нижчі оцінки, ніж у середньому.

Для візуалізації цього методу можна нанести на діаграму оцінки, виставлені двома критиками. Mick LaSalle оцінив фільм "Superman" на 3, а Gene Seymour – на

5, тому ми наносимо крапку (3,5) (малюнок 1.2). На діаграмі також зображено пряму лінію. Вона називається лінією найкращого наближення, оскільки проходить настільки близько до всіх точок на діаграмі, наскільки це можливо. Якби обидва критики виставили всім фільмам однакові оцінки, то ця лінія виявилася б діагональною та пройшла б через усі точки. У цьому випадку вийшла б ідеальна кореляція з коефіцієнтом 1. Але в нашому випадку критики розійшлися в оцінках, тому коефіцієнт кореляції дорівнює 0,4.

Також можна навести приклад із набагато вищим коефіцієнтом кореляції 0,75 (Рис. 2.2).

Коефіцієнт кореляції Пірсона має одне цікава властивість, яку можна спостерігати на малюнку – він коригує знецінення оцінок.

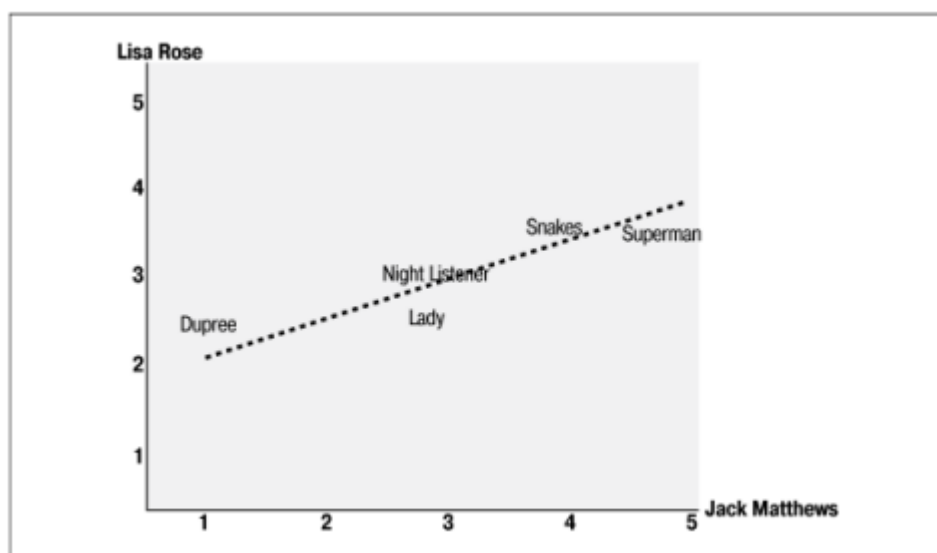


Рис. 2.2 – Два критики з високим коефіцієнтом кореляції

Програма для обчислення коефіцієнта кореляції Пірсона спочатку знаходить фільми, оцінені обома критиками, і обчислює суму та суму квадратів виставлених ними оцінок, і навіть суму творів оцінок. На останньому етапі знайдені значення використовуються для обчислення коефіцієнта кореляції. На відміну від евклідової метрики, ця формула не така очевидна (2.3):

(2.3)

$$w_{i,j} = \frac{\sum_a (r_{i,a} - \bar{r}_i)(r_{j,a} - \bar{r}_j)}{\sqrt{\sum_a (r_{i,a} - \bar{r}_i)^2 \sum_a (r_{j,a} - \bar{r}_j)^2}},$$

де  $w_{i,j}$  - Коефіцієнт подібності користувача  $i$  з користувачем  $j$ ;

$\bar{r}_i$  – середня арифметична оцінка користувача  $i$ ;

$\bar{r}_j$  – середня арифметична оцінка користувача  $j$ ;

$r_{i,a}$  – оцінка користувача  $i$  об'єкту  $a$ ;

$r_{j,a}$  – оцінка користувача  $j$  об'єкту  $a$ .

Ця функція повертає значення від -1 до 1. Значення 1 означає, що два людини виставили кожному предмету точно однакові оцінки. На відміну від евклідової метрики, масштабувати повернене значення для приведення до потрібного діапазону не потрібно [13].

GroupLens алгоритм був розроблений в ході проекту GroupLens. Цей проект був розпочато в 1992 році Полом Резніком і Джоном Рідлом і був націлений на дослідження в галузі колаборативної фільтрації.

Алгоритм GroupLens складається з двох етапів:

- 1). Знайти у базі даних користувачів зі схожими інтересами.
- 2). Передбачити за оцінками інших користувачів оцінки поточного користувача ще не оціненим їм продуктам, враховуючи з великою вагою тих користувачів, які більше схожі на поточного.

Щоб визначити, наскільки схожі уподобання користувачів, необхідно кожну людину порівняти з усіма іншими та обчислити коефіцієнт подібності. Зробити це можна декількома способами, наприклад, за допомогою евклідового відстані чи коефіцієнта кореляції Пірсона, розглянутих раніше.

Для прогнозування оцінки будь-якому контенту необхідно скласти середній рейтинг поточного користувача з відхиленнями від середніх рейтингів інших користувачів, зважених отриманими раніше коефіцієнтами подібності (2.4):

(2.4)

$$\frac{(\hat{r}_{i,a} - r'_j) w_{i,j}}{\sum_j |w_{i,j}|},$$

$$\hat{r}_{i,a} = r'_i + \hat{\epsilon}_i$$

де  $\hat{r}_i$ ,  $a$  – прогнозована оцінка об'єкту  $a$ , яка буде виставлена користувачем  $i$ ;

$r'_i$  – середня арифметична оцінка користувача  $i$ ;

$r_{j,a}$  - оцінка користувача  $j$  об'єкту  $a$ ;

$r'_j$  – середня арифметична оцінка користувача  $j$ ;

$w_{i,j}$  - Коефіцієнт подібності користувача  $i$  з користувачем  $j$  [14].

Сингулярне розкладання - потужний метод, який використовується для скорочення розмірності матриць. Застосовуючи його у рекомендаційній системі, можна прогнозувати оцінки, які можуть виставити користувачі деяким продуктів.

У теоремі про сингулярне розкладання йдеться, що будь-яку матрицю  $A$  розміром  $n \times m$  можна розкласти у добуток трьох матриць  $U$ ,  $\Sigma$  та  $V^T$  (2.5):

(2.5)

$$A_{n \times m} = U_{n \times n} \times \Sigma_{n \times m} \times V_{m \times m}^T,$$

де  $U$  та  $V$  – ортогональні матриці;

$\Sigma$  – діагональна матриця, елементи якої розташовані за спаданням.

Крім звичайного розкладання буває ще зрізане, коли з ненульових елементів матриці залишаються лише перші  $d$ , інші ж прирівнюємо нулю. Це рівнозначно тому, що у матриць  $U$  та  $V$  ми залишаємо тільки перші  $d$  стовпців, а матрицю  $\Sigma$  обрізаємо до квадратної  $d \times d$  (2.6):

(2.6)

$$A'_{n \times m} = U'_{n \times d} \times \Sigma'_{d \times d} \times V'^T_{d \times m}.$$

Отримана матриця  $A'$  добре наближає вихідну матрицю  $A$  і більше того, є найкращим низькоранговим наближенням з погляду середньоквадратичного відхилення. Застосовуючи це розкладання в рекомендаційних системах, матриця  $A$  є рейтинги, виставлені користувачами деяким фільмам, добуток матриць  $U'$  і  $\Sigma'$  представляються наборами параметрів, що характеризують користувачів, матриця  $V'$  – параметрами фільмів (Рис. 2.3).

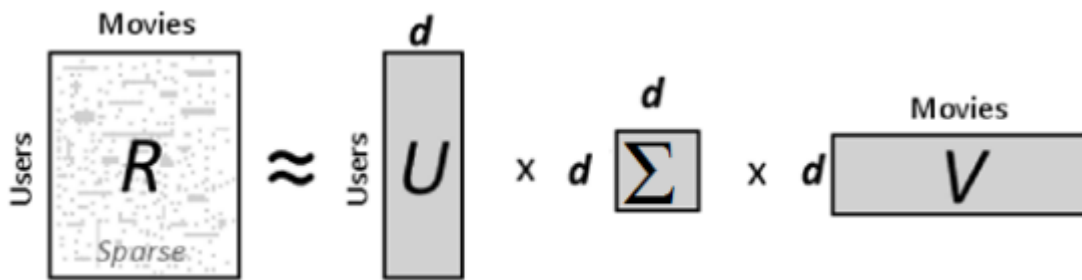


Рис. 2.3 – Подання сингулярного розкладання у рекомендаційних системах

Щоб передбачити оцінку користувача  $U$  для об'єкта  $I$ , ми беремо деякий вектор  $p_u$  (набір параметрів) для даного користувача та вектор для даного об'єкта  $q_i$ . Їхній скалярний твір і буде потрібним нам прогнозом [13] (2.6):

$$\hat{r}_{ui} = \langle p_u, q_i \rangle. \quad (2.7)$$

Алгоритм kNN, отримуючи новий зразок, для якого необхідно спрогнозувати числове значення, порівнює його із зразками, значення яких вже відомі. Він шукає зразки, максимально схожі на знову пред'явлений, і усереднює їх значення. Розглянемо список цифрових камер, для кожної з яких вказано дозвіл у мегапікселях, максимальний збільшення фокусної відстані (зум) та ціна (Таблиця 2.2).

Таблиця 2.2 – Ціни на цифрові камери

Камера	Мегапікселі	Зум	Ціна
C1	7,1	3,8	399
C2	5,0	2,4	299
C3	6,0	4,0	349
C4	6,0	12,0	399
C5	10	3	449

Допустимо, потрібно дізнатися ціну нової камери з шістьма мегапікселями та об'єктивом, що має зум  $6\times$ . Насамперед потрібно знайти спосіб виміряти ступінь подібності двох зразків. Зробити це можна використовуючи евклідово відстань або кореляція Пірсона, які були розглянуті раніше. В термінів евклідова відстані найближчою камерою буде C3. Для візуалізації результату можна нанести зразки на діаграму, де по осі x відкладено мегапікселі, а по осі y – зум, а самі зразки увявити на діаграмі своїми цінами (Рис. 2.5).

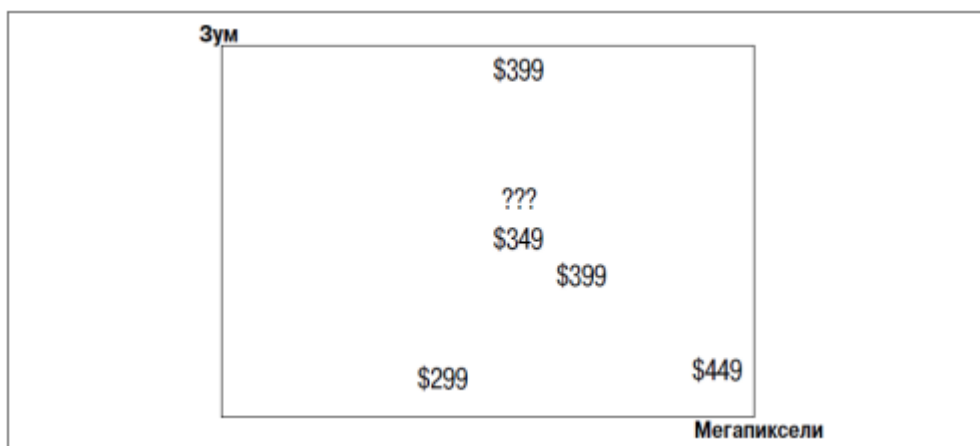


Рис. 2.5 – Ціни камер у просторі зум-мегапікселі

Замість прямолінійного усереднення можна обчислити середньозважене враховуючи те, наскільки далеко відстоять зразки. Чим більша відстань, тим менше вага. У цьому прикладі ціною \$349 можна приписати найбільшу вагу, а двом цінам \$399 – ваги менші. Наприклад:

$$\text{Ціна} = 0,5 \times 349 + 0,25 \times 399 + 0,25 \times 399 = 374.$$

Так само, як і метод сингулярного розкладання, у рекомендаційних системах метод найближчих сусідів дозволяє спрогнозувати, які оцінки користувач може виставити контенту, що дозволяє сформуванню йому рекомендації [13].

Класифікатор Байєса як і всі алгоритми навчання з учителем, навчається на прикладах. Приклад – це список ознак зразка та його класифікація. Припустимо, що потрібно навчити класифікатор розпізнавати, чи належить документ, що містить слово `python` до мови програмування або до зміїв. Для початку створимо навчальний набір для цієї Цілі (Таблиця 1.2).

Класифікатор запам'ятовує всі ознаки, що зустрілися, а також ймовірності того, що ознака асоційована з конкретною класифікацією. Приклади пред'являються класифікатору по одному. Після кожного прикладу класифікатор оновлює свої дані, обчислюючи ймовірність того, що документ із зазначеної категорії містить те чи інше слово.

Наприклад, після навчання на наведеному вище прикладі може вийти наступний набір ймовірностей (Таблиця 2.3).



Таблиця 2.3 – Набір ймовірностей

Ознаки	Класифікація
Пітони – констриктори, які харчуються птахами та ссавцями	Змія
Python з самого початку розроблявся як мова сценаріїв	Мова
В Індонезії знайшли пітона 49 футів	Змія
У Python реалізована динамічна система типів	Мова
Пітон із яскравими лусочками	Змія
Проект із відкритими вихідними текстами	Мова

Таблиця 2.4 – Можливості власності слів до категорій

Ознака	Мова	Змія
Динамічний	0,6	0,1
Констриктор	0,0	0,6
Довжина	0,1	0,2
Початковий	0,3	0,1
	0,95	0,95

З цієї таблиці видно, що після навчання асоціації ознак з різними категоріями стають сильнішими або слабшими. Слово «констриктор» має велику ймовірність для змії, а слово «динамічна» – для мови програмування. Неоднозначні ознаки, наприклад, союз «і» мають близькі ймовірності для обох

категорій (слово «і» зустрічається майже в будь-якому документі незалежно від його тематики).

Навчений класифікатор – це лише список ознак разом із асоційованими ймовірностями. На відміну від деяких інших методів класифікації, тут не потрібно зберігати вихідні дані, у яких проводилося навчання. Після навчання байєсовський класифікатор можна застосовувати для автоматична класифікація нових зразків. Допустимо, є документ, містить слова «довжина», «динамічний» та вихідний. Також є ймовірності кожного слова, але вони обчислені лише окремих слів. Якби всі слова з високою ймовірністю потрапляли в ту саму категорію, то відповідь була б зрозуміла. Однак для слова «динамічний» вища ймовірність потрапляння до категорії «Мова», а для слова «довжина» – до категорії «Змія». Щоб класифікувати новий документ, необхідно за ймовірністю ознак обчислити ймовірність всього зразка.

Один із методів вирішення цього завдання – наївний байєсівський класифікатор. Він обчислює сукупну ймовірність за такою формулою:

$$\Pr(\text{Категорія}|\text{Документ}) = \Pr(\text{Документ}|\text{Категорія}) \times \Pr(\text{Категорія}),$$

де  $\Pr(\text{Документ}|\text{Категорія}) = \Pr(\text{Слово1}|\text{Категорія}) \times \Pr(\text{Слово2}|\text{Категорія}) \times \dots$

Величини  $\Pr(\text{Слово}|\text{Категорія})$  – значення, взяті з таблиці, наприклад,  $\Pr(\text{Динамічний} | \text{Мова}) = 0,6$ . Величина  $\Pr(\text{Категорія})$  дорівнює повній частоті народження даної категорії. Оскільки категорія «Мова» зустрічалась у половині випадків, то  $\Pr(\text{Мова}) = 0,5$ . Результатом вважається та категорія, для якою  $\Pr(\text{Категорія}|\text{Документ})$  набуває максимального значення.

У рекомендаційних системах наївний байєвський класифікатор дозволяє визначити, чи сподобається користувачеві якийсь контент чи ні. Наприклад, на сукупності веб-сторінок, оцінених клієнтом як «корисні» та «непотрібні», класифікатор використовується, щоб класифікувати неоцінені веб-сторінки. Інакше кажучи, класифікатор має передбачити ймовірність того, що належить до класу важливих або неважливих сторінок виходячи з ключових слів на цій сторінці [13].

## 2.4 Проектування бази даних

Розробка бази даних починається з концептуального проектування.

Для цього необхідно виділити усі сутності та зв'язки в системі.

Користувачі інтернет-сервісу можуть продивлятися теми, шукати їх, та залишати повідомлення у них.

На основі цих даних було виведенно 6 сутностей та 8 зв'язків один до багатьох (Рис. 2.6).

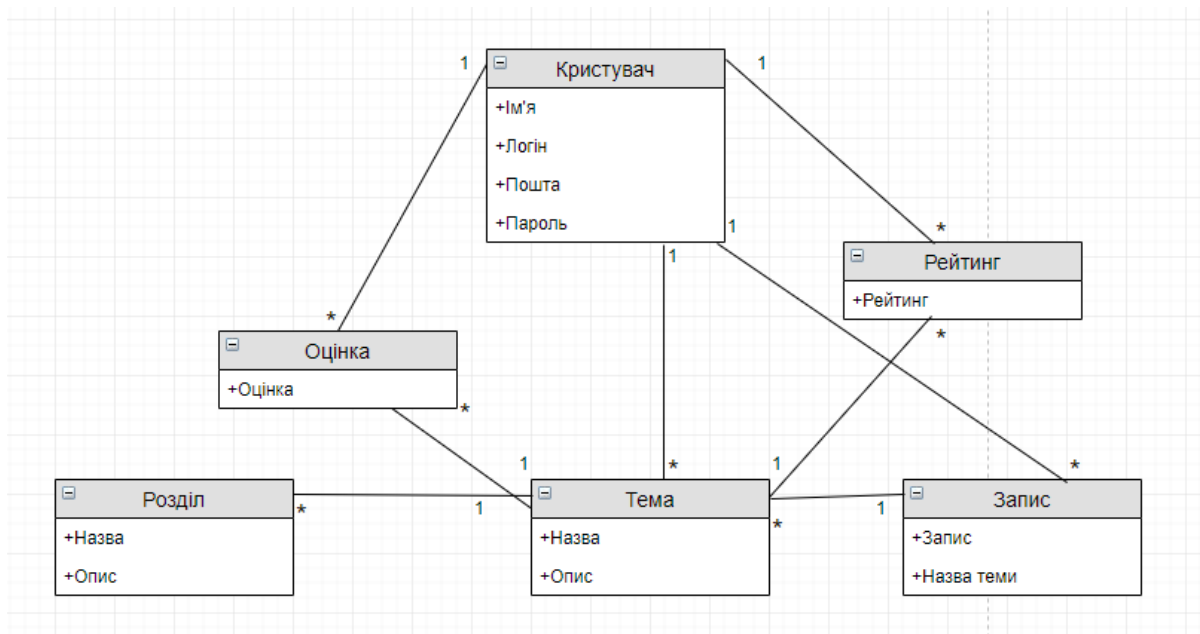


Рис. 2.6 – ER-діаграма бази даних

На рис. 2.7 наведено схему реляційної бази даних для системи, що розробляється.

База даних призначена для зберігання всієї довготривалої інформації системи: зареєстровані користувачі, їх оцінки тем які формуються в залежності від того скільки повідомлень залишив користувач в темі, інформація про повідомлення, теми та розділи. Також є тимчасова таблиця з рекомендаціями, куди заносяться прогнозовані рейтинги для неоцінених тем поточним користувачем.

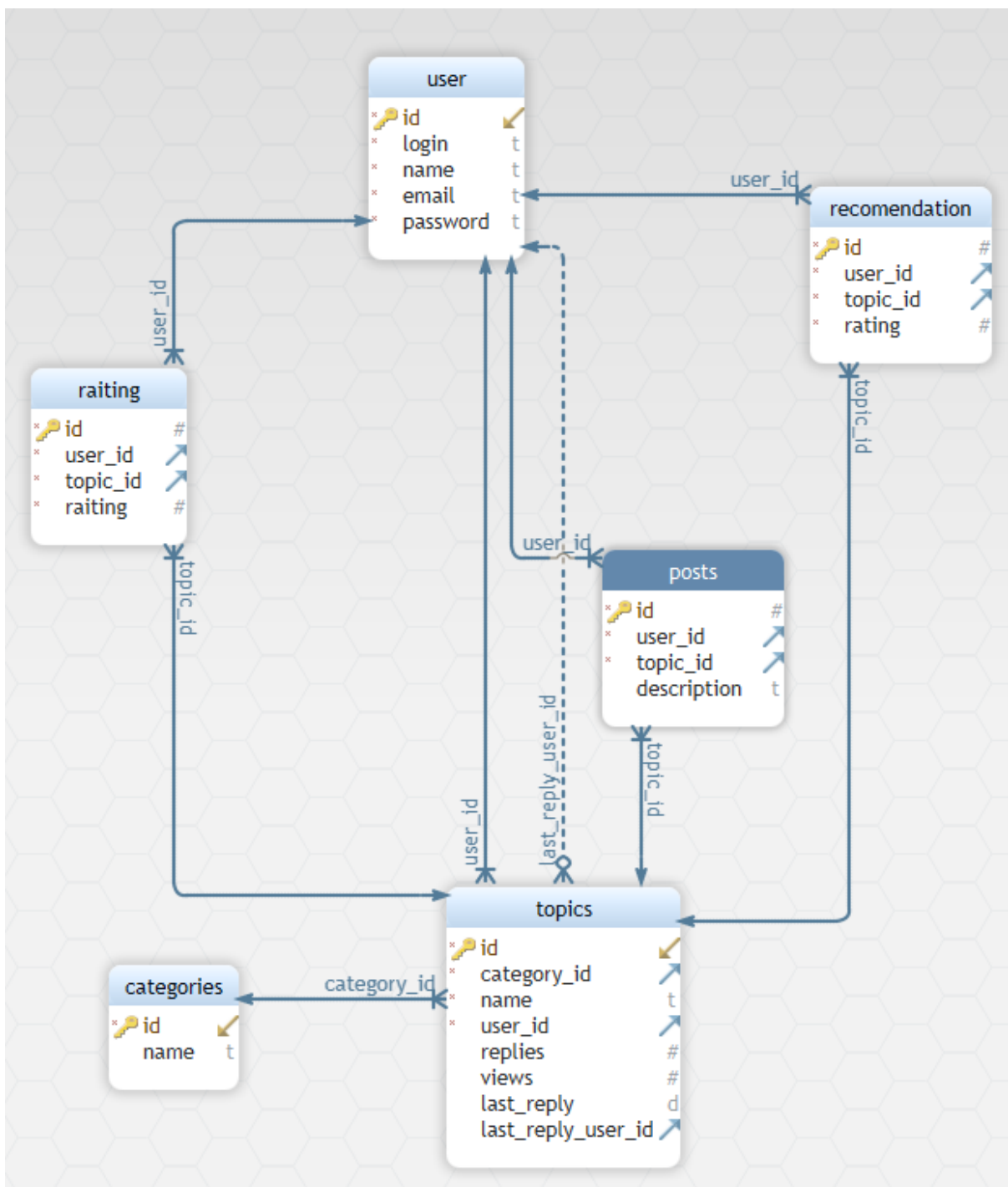


Рис. 2.7 – Схема реляційної бази даних

Опис призначення та властивостей полів бази даних наведено у таблицях 2.5-2.10.

Таблиця 2.5 – «user» користувачі системи

	Тип даних	Значення за замовчуванням	Обов'язковість	Первинний ключ	Зовнішній ключ	Обмеження
id	int	-	+	+	-	-
login	varchar	-	+	-	-	унікальний
name	varchar	-	+	-	-	-
email	varchar	-	+	-	-	шаблон
password	varchar	-	+	-	-	не менше 6 символів

У цій таблиці зберігаються дані користувача – ім'я (name), логін (username), поштова адреса (email) та пароль (password). Логін має бути унікальним. З метою підвищення безпеки пароль не повинен бути коротшим 6 символів.

Таблиця 2.6 – «rating» список оцінок

	Тип даних	Значення за замовчуванням	Обов'язковість	Первинний ключ	Зовнішній ключ	Обмеження
id	int	-	+	+	-	-
user_id	int	-	+	-	+	-
topic_id	int	-	+	-	+	-
rating	int	-	+	-	-	-

У цій таблиці зберігаються всі дані про оцінки – ідентифікатор користувача (user\_id), ідентифікатор теми, якій була виставлена оцінка (item\_id) та сама оцінка (rating), яка може мати значення від 1 до 10.

Таблиця 2.7 – «posts» список повідомлень

	Тип даних	Значення за замовчуванням	Обов'язковість	Первинний ключ	Зовнішній ключ	Обмеження
id	int	-	+	+	-	-
user_id	int	-	+	-	+	-
topic_id	int	-	+	-	+	-
description	text	-	-	-	-	-

У цій таблиці зберігаються дані про повідомлення користувачів у темах.

ідентифікатор користувача (user\_id), ідентифікатор теми (topic\_id) та саме повідомлення (description).

Таблиця 2.8 – «categories» список розділів

	Тип даних	Значення за замовчуванням	Обов'язковість	Первинний ключ	Зовнішній ключ	Обмеження
id	int	-	+	+	-	-
name	varchar	-	+	-	-	-

Таблиця 2.9 – «topics» список розділів

	Тип даних	Значення за замовчуванням	Обов'язковість	Первинний ключ	Зовнішній ключ	Обмеження
id	int	-	+	+	-	-
category_id	int	-	+	-	+	-
name	varchar	-	+	-	-	-
user_id	int	-	+	-	+	-
replies	int	-	-	-	-	-
views	int	-	-	-	-	-
last_reply	datetime	-	-	-	-	-
last_reply_user_id	int	-	-	-	-	-

У цій таблиці зберігаються дані про теми. Ідентифікатор розділу до якого належить тема (category\_id), назва теми (name), ідентифікатор користувача (автора теми) (user\_id), кількість повідомлень у темі (replies), кількість переглядів у темі (views), дата останнього повідомлення (last\_reply), ідентифікатор автора останнього повідомлення (last\_reply\_user\_id).

Таблиця 2.10 – «raiting» рекомендації користувачів

	Тип даних	Значення за замовчуванням	Обов'язковість	Первинний ключ	Зовнішній ключ	Обмеження
id	int	-	-	+	-	-
user_id	int	-	-	-	+	-
topic_id	int	-	-	-	+	-
raiting	int	-	-	-	-	-

У цій таблиці тимчасово зберігаються прогнозовані оцінки неоціненим користувачем теми з моменту запиту рекомендацій до моменту їх отримання. Таблиця містить таку інформацію, як ідентифікатор користувача, який зробив запит (userID), ідентифікатор теми, який не був оцінений користувачем (itemID) та прогнозовану оцінку цієї теми (rating). Оцінка знаходиться в інтервалі від 1 до 5 і може бути нецілим числом.



## 3 РОЗРОБКА ІНТЕРНЕТ-СЕРВІСУ

### 3.1 Діаграма прецедентів

Діаграма прецедентів, також звана діаграмою варіантів використання (use-case diagram), відображає відносини між акторами та прецедентами і дає уявлення про функціональну поведінку системи [15]. На малюнку 3.1 представлена діаграма, що описує типи користувачів та їх взаємодія із системою.

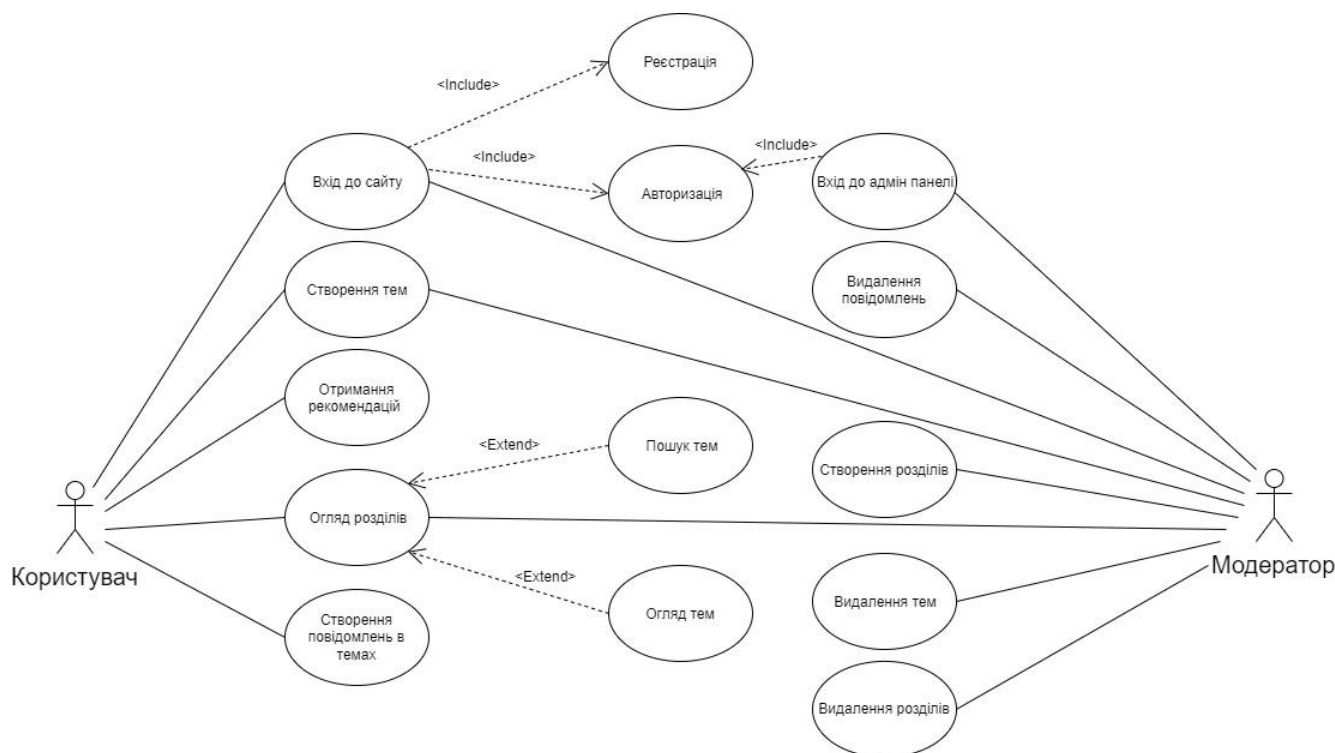


Рис. 3.1 - діаграма прецедентів

Далі наводиться словесний опис (сценарій) для основних прецедентів

Use-case: «Реєстрація»

Для отримання доступу до розширеного функціоналу сайту користувач може зареєструватися у ньому, вказавши необхідні відомості себе.

Головна послідовність:

Користувач вводить ім'я, логін, адресу пошти та пароль для перевірки коректності адреси пошти та пароля повторює їх введення у формі відповідній сторінці та підтверджує введення. Система перекладає користувача на сторінку з інформацією про профіль.

Альтернативна послідовність 1:

Введення вже існуючих логін або пошти. Система видає повідомлення про те, що логін або пошта зайняті, та пропонує повторити введення.

Альтернативна послідовність 2:

Введіть неправильну адресу пошти. Система видає повідомлення про те, що адреса пошти введено помилково і пропонує повторити введення.

Альтернативна послідовність 3:

Введіть короткий пароль. Система видає повідомлення про те, що пароль менший чотири символи, і пропонує повторити введення.

Альтернативна послідовність 4:

Неправильне підтвердження адреси пошти чи пароля. Система видає повідомлення про те, що пошта або пароль невірно підтвержені, та пропонує повторити введення.

Use-case: "Авторизація"

Якщо відвідувач вже має профіль на сайті, то він може авторизуватися на ньому, вказавши свої логін та пароль. Користувач вводить логін та пароль у формі на сторінці авторизації та підтверджує введення. Система переводить користувача на сторінку з інформацією про профіль.

Неправильне введення логіна або пароля. Система видає повідомлення про те, що логін або пароль введено неправильно, і пропонує повторити введення.

Use-case: «Огляд розділів»

На головній сторінці сайту представлений каталог назв усіх розділів, що існують у базі даних. Разом із розділом представленні його короткий опис, кількість тем у цьому розділі та власник і дата останньої теми доданої у розділ. Користувач може переглядати їх, прокручуючи коліщатко мишки або використовуючи смугу прокручування.

Use-case: «Огляд тем»

При виборі одного з розділів користувач переходить на іншу сторінку де представлений каталог назв усіх тем, що існують у базі даних. Разом із темою представленні її власник, кількість повідомлень та просмотрів у темі, автор та час останнього повідомлення.

Use-case: «Пошук тем»

Якщо відвідувач сайту не бажає переглядати всі існуючі теми, він може скористатися системою пошуку та виставити необхідні фільтри та переглянути тільки ті теми, які задовольняють ім.

На головній сторінці сайту у спеціальній формі користувач за бажанням вибирає пошук по назві або опису теми, вводить назву теми або частину назви та підтверджує введення. Після цього виводяться теми, відповідні виставленим фільтрам.

Use-case: «Створення тем»

У авторизованого користувача є можливість створення окремої теми. Для цього потрібно у спеціальній формі вказати назву тему, опис теми та вибрати до якого розділу стосується ця тема. Після цього нова тема буде збережена у бази даних та стане доступною для всіх інших користувачів форуму.

Use-case: «Видалення тем»

Якщо авторизуватися як модератор інтернет-сервісу то при виборі теми з'явиться можливість видалення повідомлень. Модератор може вибрати повідомлення та натиснути на кнопку видалення біля нього. Після цього небажане повідомлення видаляється з теми та бази даних.

Use-case: «Повідомлення в темах»

Якщо користувач бажає залишити повідомлення у темі він повинен заповнити спеціальну форму у кінці сторінки з списком повідомлень та підтвердити її. Після цього повідомлення збережеться у базі даних та з'явиться у списку повідомлень відповідної теми. Повідомлення буде доступне для перегляду усім іншим користувачам форуму.

Use-case: «Отримання рекомендацій»

Авторизований на сайті користувач може отримати каталог тем, які, за припущенням рекомендаційної системи сайту, можуть бути йому цікаві. Користувач за бажанням натискає кнопку "Рекомендації" на головній сторінці сайту та переходить на сторінку рекомендацій. Сторінка перезавантажується та виводяться рекомендовані йому теми.

Якщо користувач не авторизований. Система виводить повідомлення про неможливість отримання рекомендацій та пропонує пройти авторизацію.

Якщо користувач ще не залишив повідомлення в жодній темі. Система виводить повідомлення про неможливість отримання рекомендацій та пропонує збільшити активність на інтернет-сервіси. Якщо немає користувачів, з якими можна порівняти поточного (тобто поточний користувач та інші користувачі залишили повідомлення у різних темах або решта користувачів не порявили

активності взагалі). Система виводить повідомлення про неможливість отримання рекомендацій та пропонує повторити спробу пізніше.

### 3.2 Розробка рекомендаційної системи

Як метод побудови рекомендаційної системи було обрано GroupLens алгоритм. Цей алгоритм складається з двох етапів. На першому етапі відбувається визначення схожості користувача, для якого необхідно передбачити оцінки, які може виставити, з іншими користувачами. Для цього використовується коефіцієнт кореляції Пірсона, що обчислюється формулою (1.1). Оцінки виставляються пропорційно тому скільки повідомлень залишив користувач у темі чим більше повідомлень тим вища оцінка. Під час другого етапу відбувається обчислення передбачуваного рейтингу тем, в яких користувач ще не залишав повідомлення. Для цього необхідно використати формулу (1.2). Схема алгоритму рекомендаційної системи наведено рис. 3.2 - рис.3.8.

Алгоритм рекомендаційної системи починається з перевірки авторизації користувача. Якщо він не авторизований, то йому виводиться повідомлення про неможливість отримання рекомендацій та пропонується пройти її. Далі формуються запити до бази даних отримання інформації про користувачів, темах та оцінках. Отримана інформація зберігається у вигляді одномірних або двовимірних масивів. У масиві з оцінками поточного користувача індекс є ідентифікатор теми, а значення – оцінку, виставлену темі. Аналогічно влаштований масив для оцінок інших користувачів, але в цьому випадку він буде двовимірним. У ньому кожен рядок відповідає будь-якому користувачеві.

Після отримання з бази необхідних даних здійснюється перевірка оцінок поточного користувача. Якщо їм не було залишено повідомлень, то система виводить повідомлення про неможливість отримання рекомендацій та пропонує збільшити активність на форумі. На наступному етапі визначається коефіцієнт схожості поточного користувача з усіма іншими. Для кожного

користувача знаходяться теми, які були оцінені обома, після чого формула (1.1) розбивається на три частини і обчислюється кожна частина. Якщо хоча б одна частина, розташована в знаменнику, що дорівнює нулю, то весь коефіцієнт кореляції між цими користувачами, що дорівнює нулю. Інакше всі частини підставляються на дану формулу та відбувається її обчислення.

Після того, як були визначені коефіцієнти подібності користувачів, виконується їхня перевірка. Якщо всі коефіцієнти виявилися нульовими, то система виводить повідомлення про неможливість отримання рекомендацій та пропонує повторити спробу пізніше. Далі для кожної теми, який не було оцінено поточним користувачем, обчислюється прогнозований рейтинг. Якщо є користувачі, які виставили оцінку темі, то формула (1.2) розбивається на 2 частини та обчислюється кожна їх. Якщо якась із частин дорівнює 0, то прогнозований рейтинг дорівнює середньому арифметичному оцінкам поточного користувача. Інакше частини підставляються у формулу та обчислюються рейтинг.

Після всіх обчислень отримані прогнози заносяться до спеціальної таблицю, надсилається запит на отримання тем, відсортованих у порядку спадання за рейтингом, відбувається отримання його результатів та оновлення сторінки. В результаті користувачеві надаються рекомендовані йому теми.



Рис. 3.2 – Схема частини алгоритма рекомендаційної системи

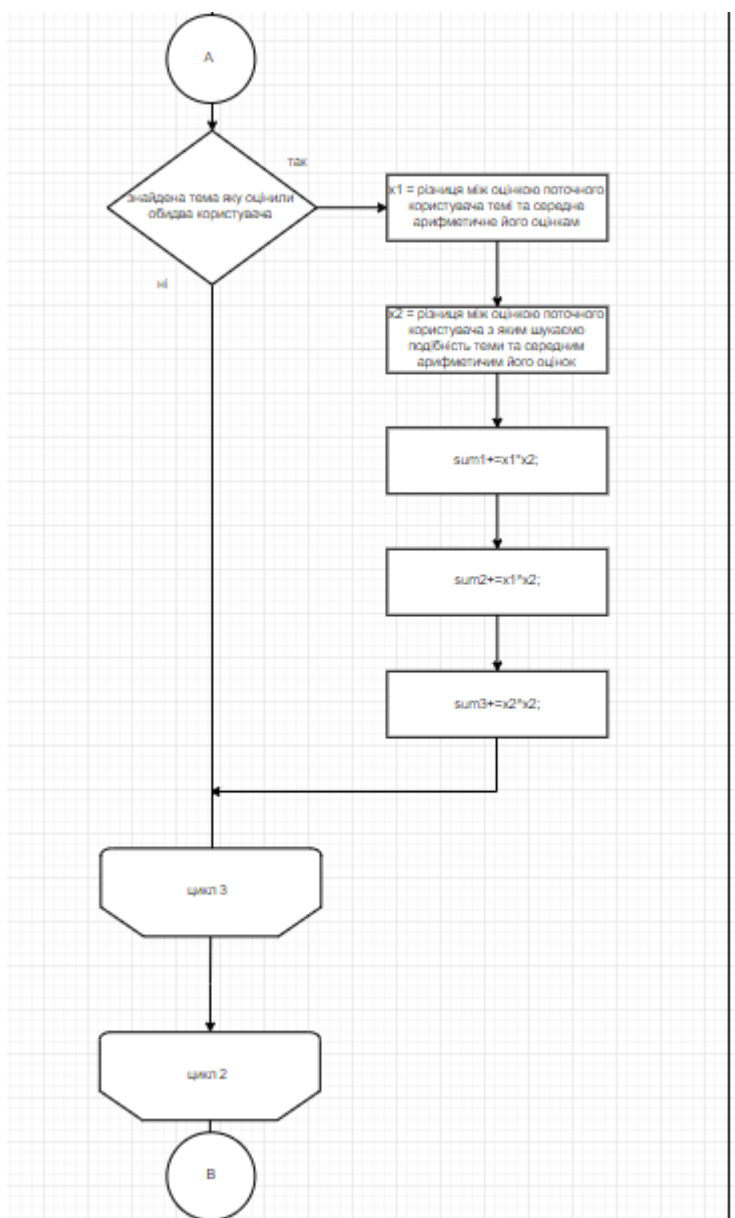


Рис. 3.3 – Схема частини алгоритма рекомендаційної системи



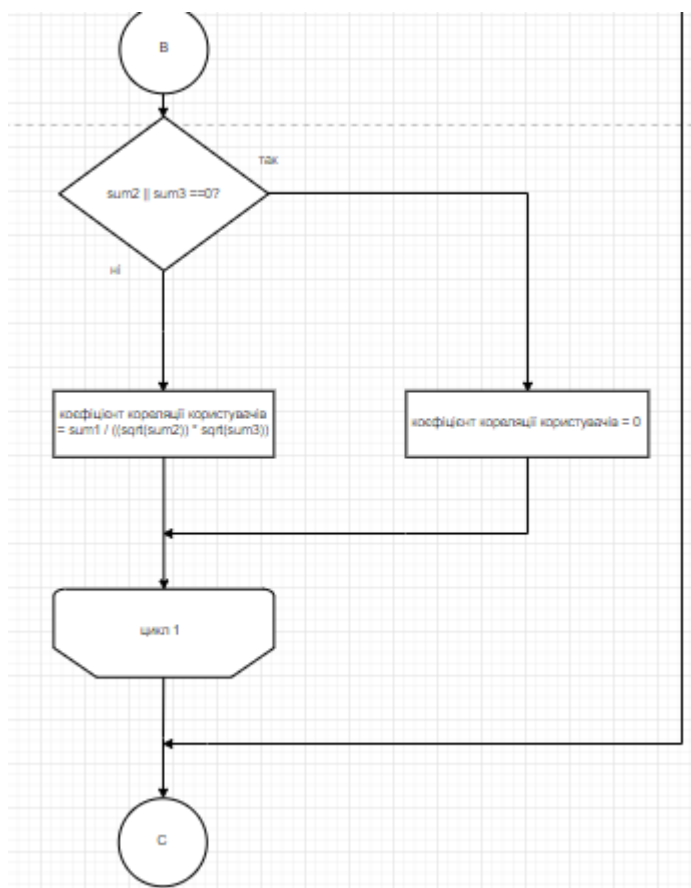


Рис. 3.4 – Схема частини алгоритма рекомендаційної системи

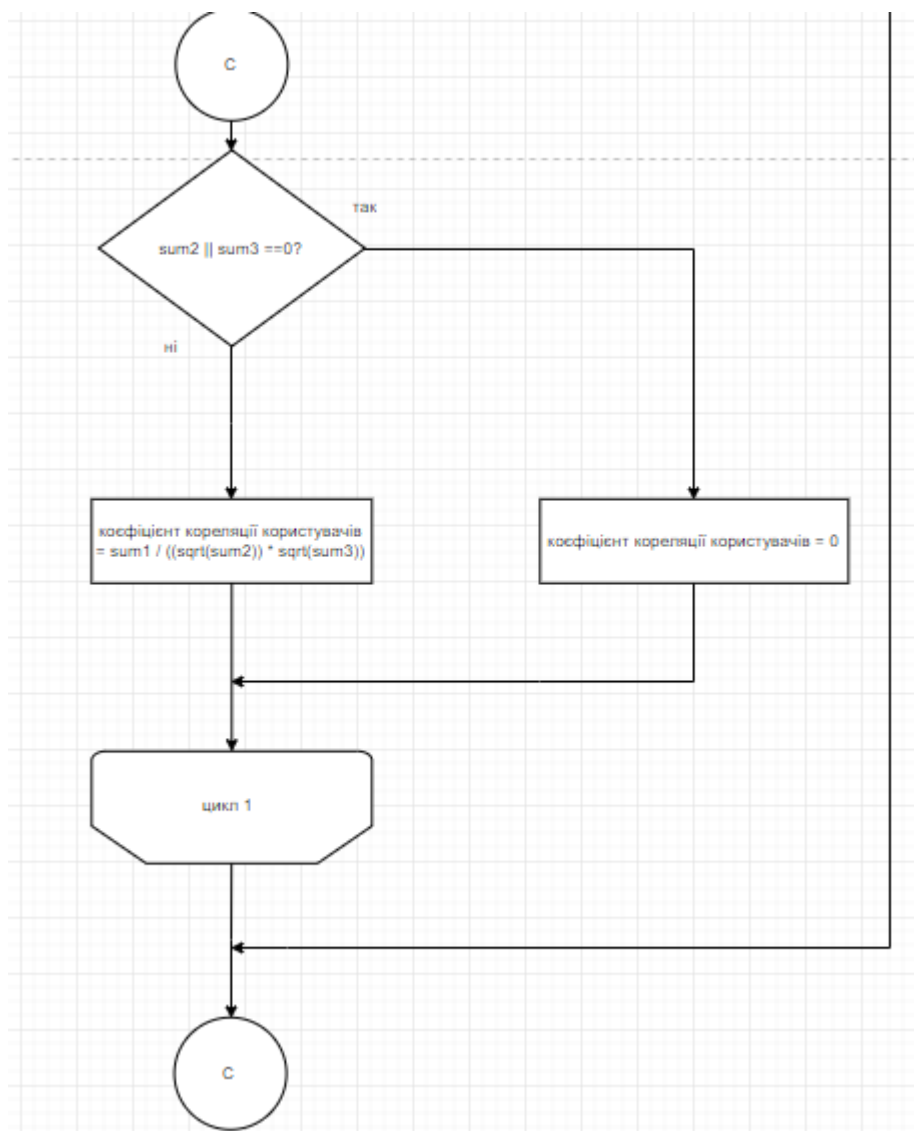


Рис. 3.5 – Схема частини алгоритма рекомендаційної системи

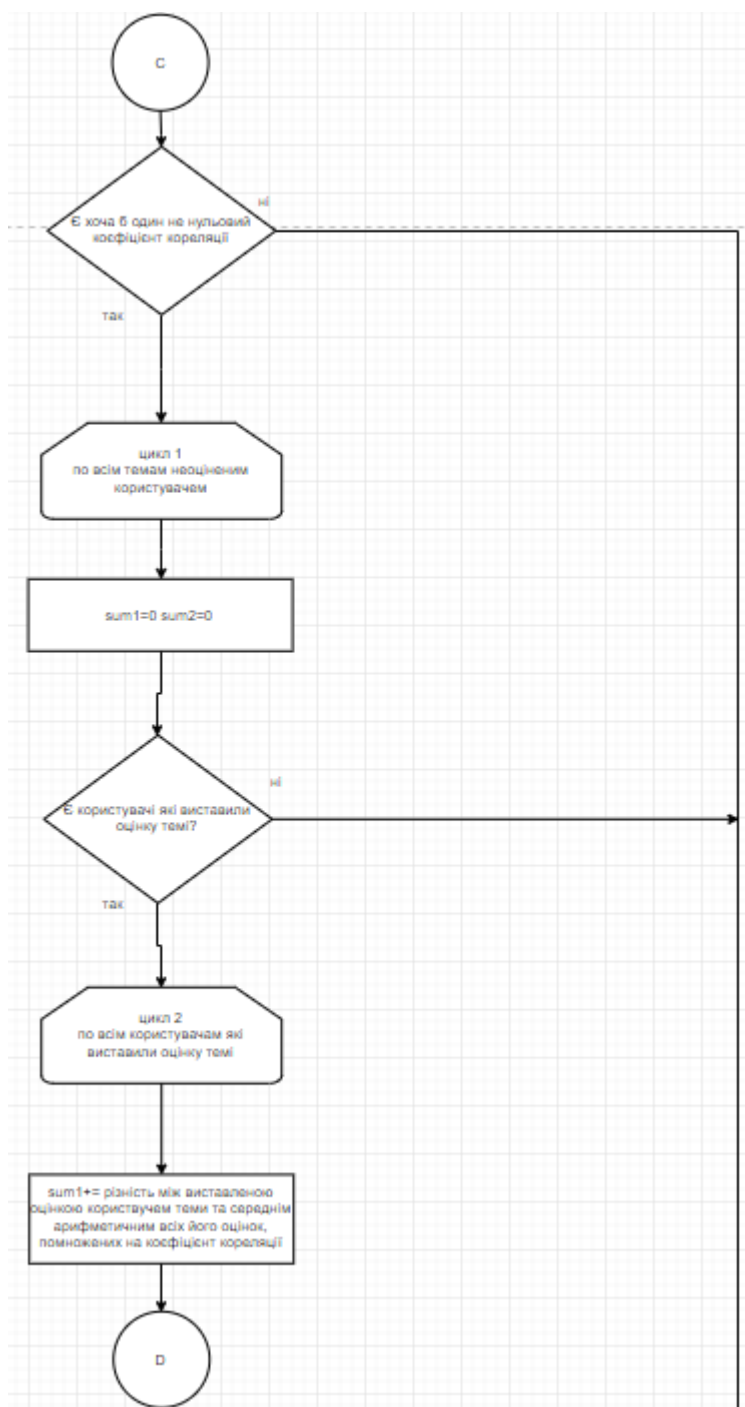


Рис. 3.6 – Схема частини алгоритма рекомендаційної системи

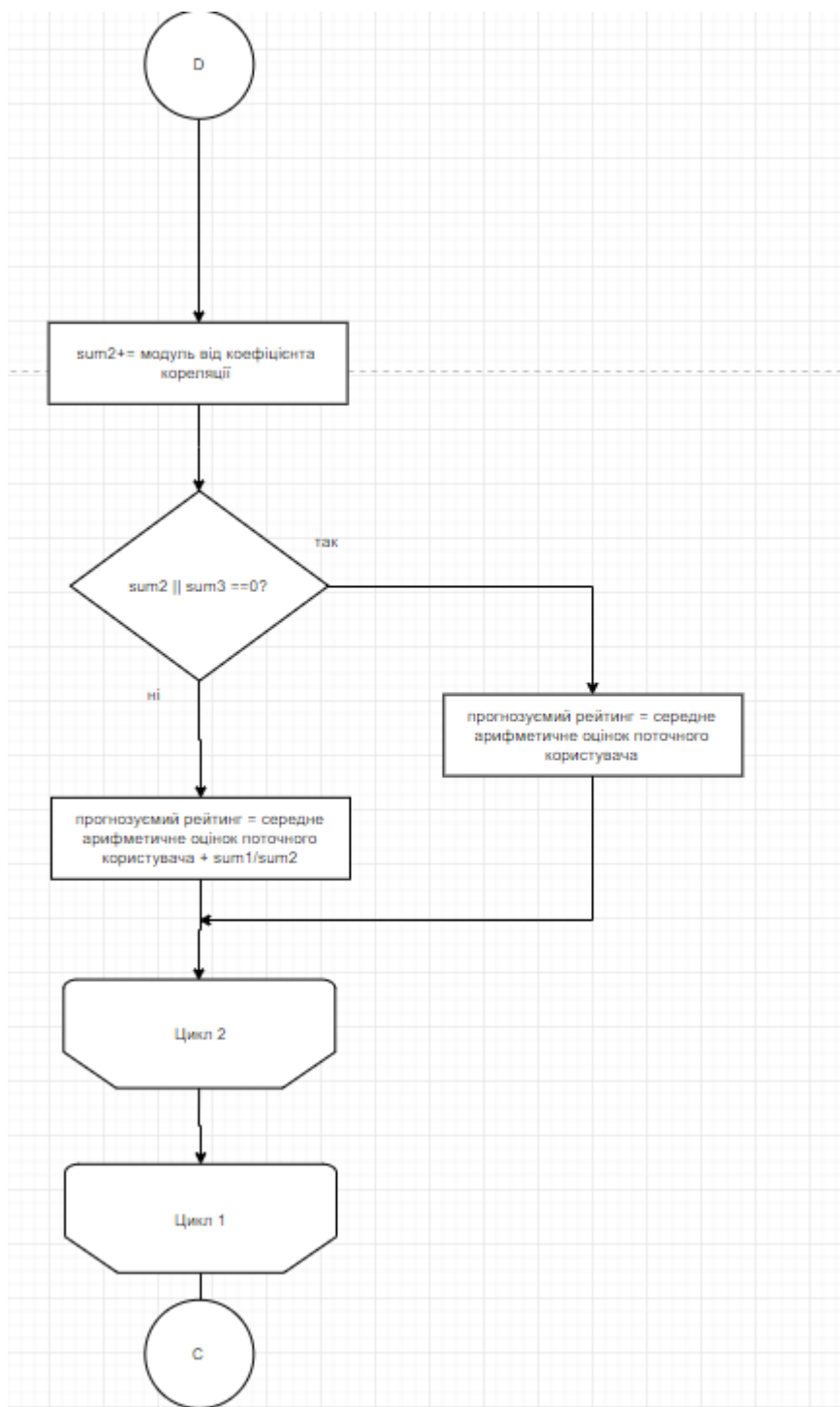


Рис. 3.7 – Схема частини алгоритма рекомендаційної системи

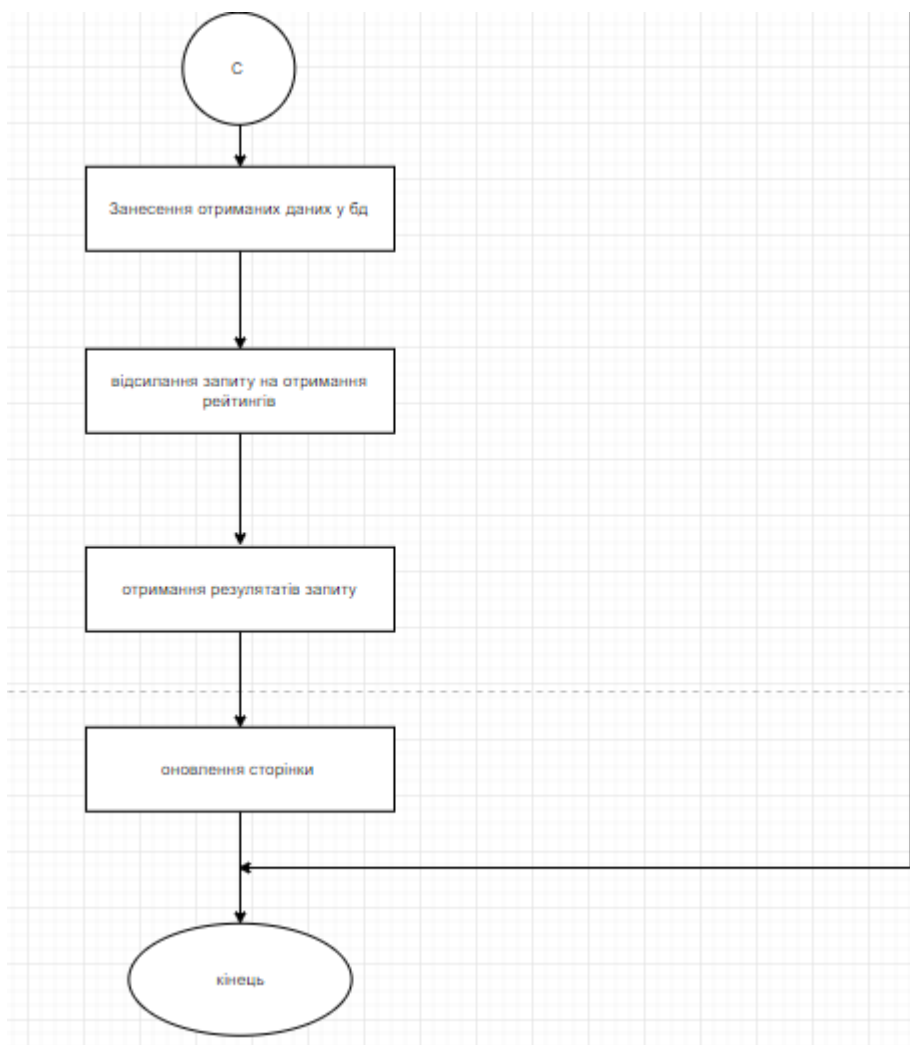


Рис. 3.8 – Схема частини алгоритма рекомендаційної системи

### 3.3 Функціонал інтернет-сервісу

Головна сторінка містить шапку сайту, панель пошуку, каталог розділів та статистику інтернет-сервісу.

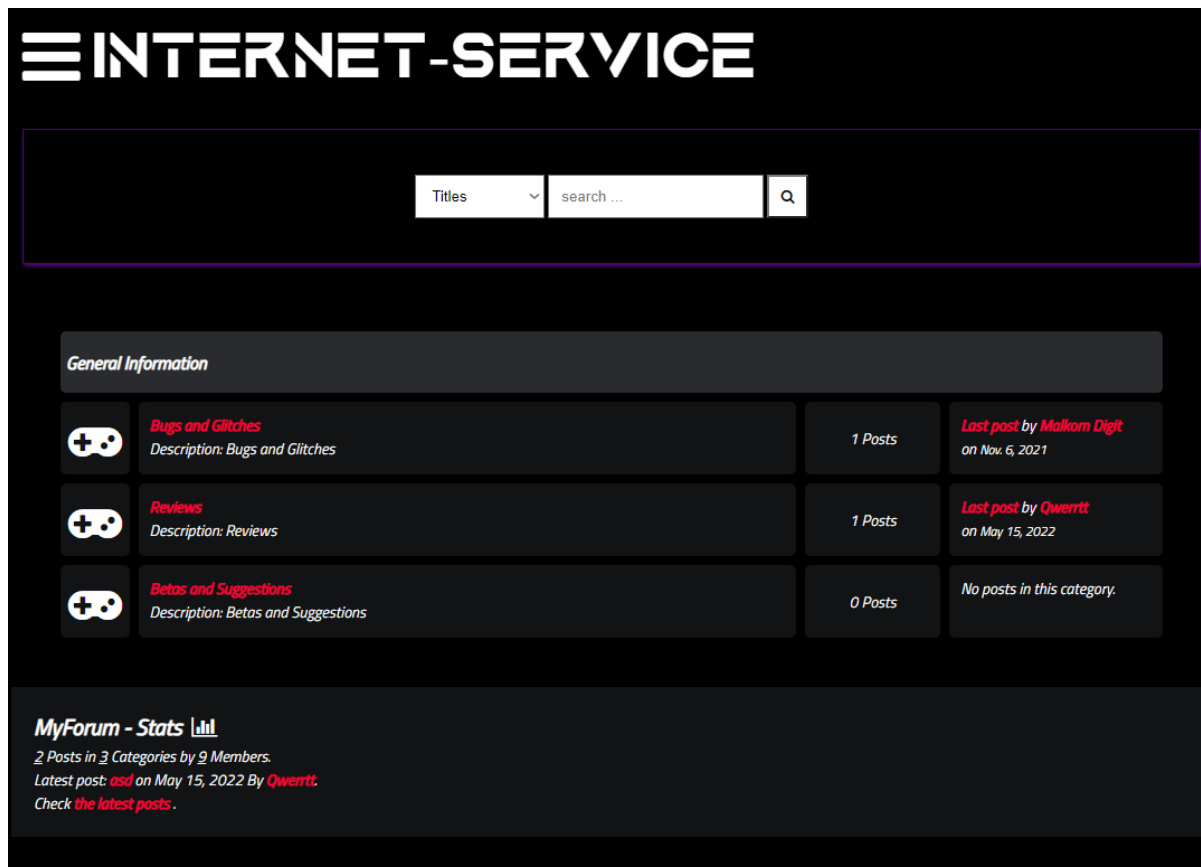


Рис. 4.1 – Головна сторінка інтернет-сервісу

Шапка сайту, яка також міститься і на інших сторінках сайту, складається з логотипу та кнопок «Увійти» та «Реєстрація». Якщо користувач авторизован з'являється кнопка «Рекомендації» При натисканні на логотип можна перейти на головну сторінку, перебуваючи при цьому на будь-якій іншій. Натиснувши на іконку "Увійти" або "Реєстрація", користувач переходить на сторінку авторизації чи реєстрації відповідно.

Панель пошуку за фільтрами дозволяє користувачеві знайти тему по назві або опису, необхідно лише виставити необхідні параметри та натиснути на іконку«Пошук».

Кнопка «Рекомендації» впорядковує ще не оцінені користувачем альбоми ступеня інтересу в порядку зменшення

Каталог розділів складається з усіх розділів, що містяться на сайті. Розділи розташовані у вигляді рядків. Кожен рядок має назву розділу, опис та назву автора і дату остальньої теми доданої у розділ.

Кількість рядків може бути обмежена і залежить від кількості розділів, що є в базі даних. При натисканні на розділ користувач переходить на присвячену йому сторінку.

На сторінці розділу користувач може отримати інформацію про теми які присутні в розділі та перейти до сторінці теми натиснувши на неї.

В колонці теми також можна побачити власника теми, кількість переглядів та повідомлень та дату і назву автора останнього повідомлення.

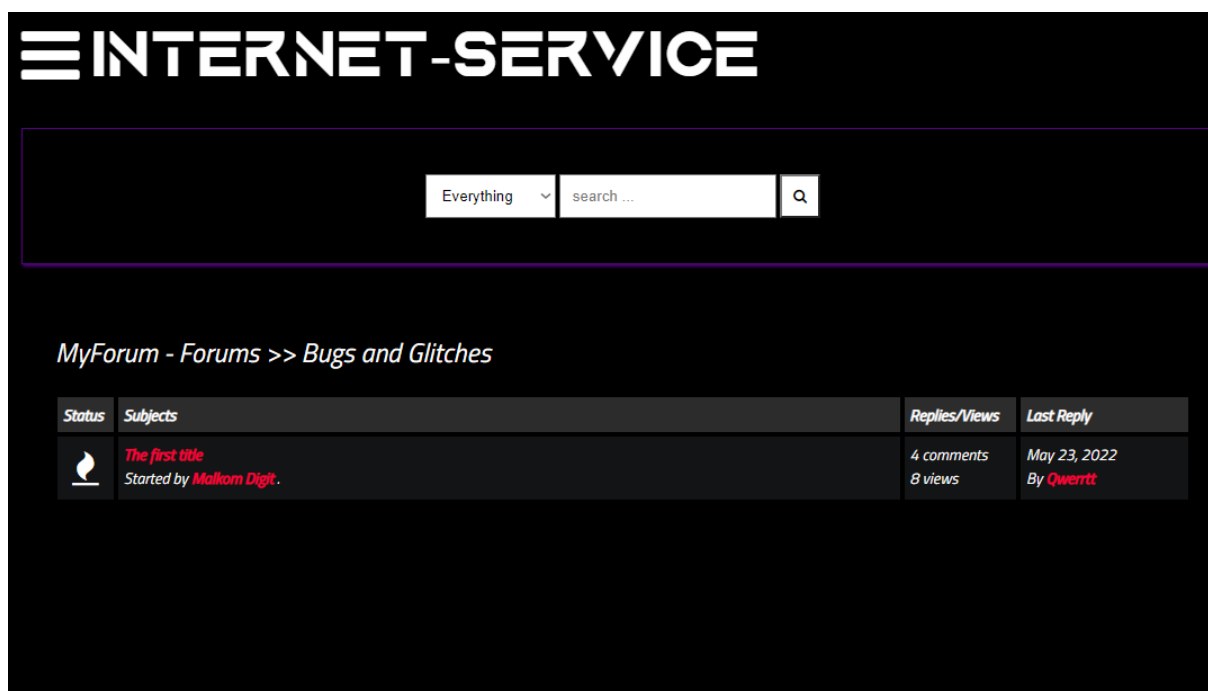


Рис. 4.2 – Сторінка розділу

На сторінці теми користувач може отримати список повідомлень цієї теми. Якщо він авторизований він може натиснути на кнопку «Comments» після чого відкривається спеціальна форма де можна вписати повідомлення та натиснути

кнопку «Submit». Повідомлення збережеться у базі даних та з'явиться на сторінці теми.

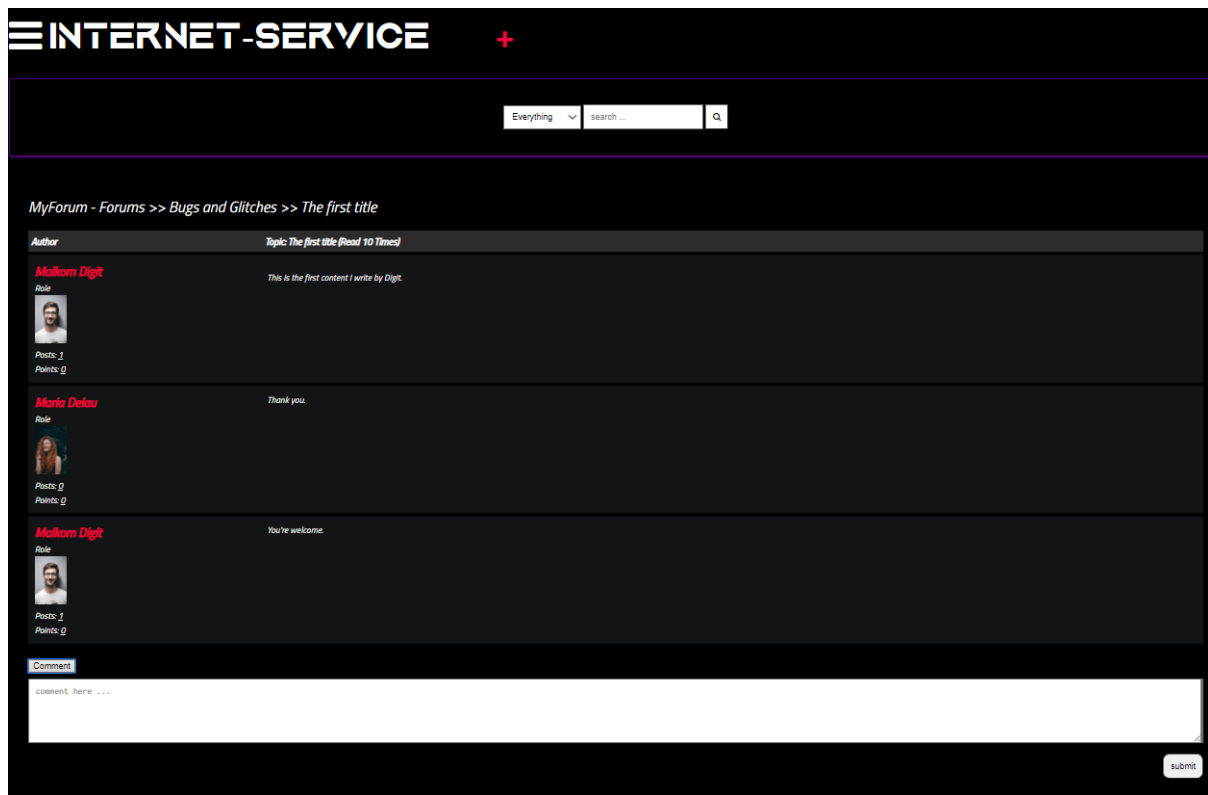
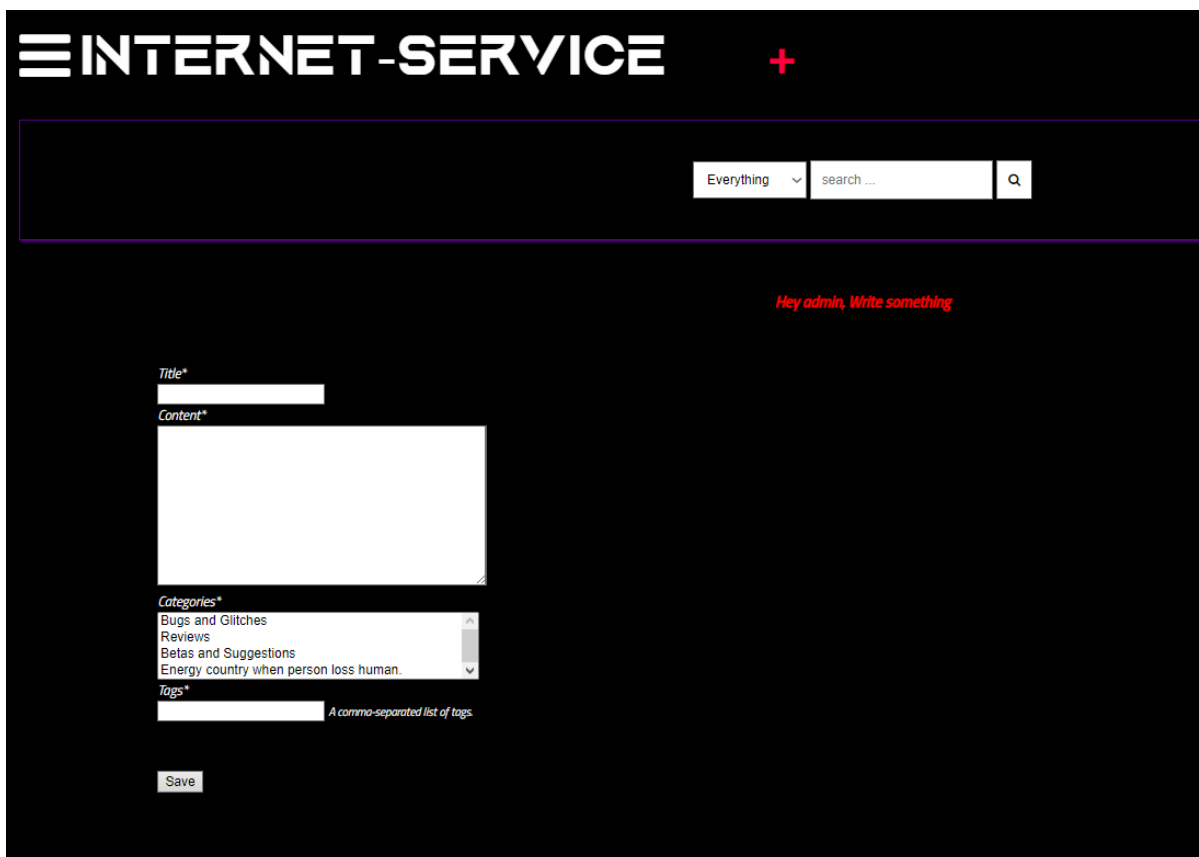


Рис. 4.3 – Сторінка теми

Для авторизованих користувачів є можливість створення тем, для цього потрібно натиснути на іконку «+» у шапці сайту, сторінка оновиться та з'явиться спеціальна форма. У формі потрібно вписати назву теми, її опис, вибрати розділ в якому буде розташована тема, після цього натиснути на кнопку «submit», сторінка знов оновиться. Створена тема збережиться в базі даних та буде доступна для перегляду іншим користувачам.



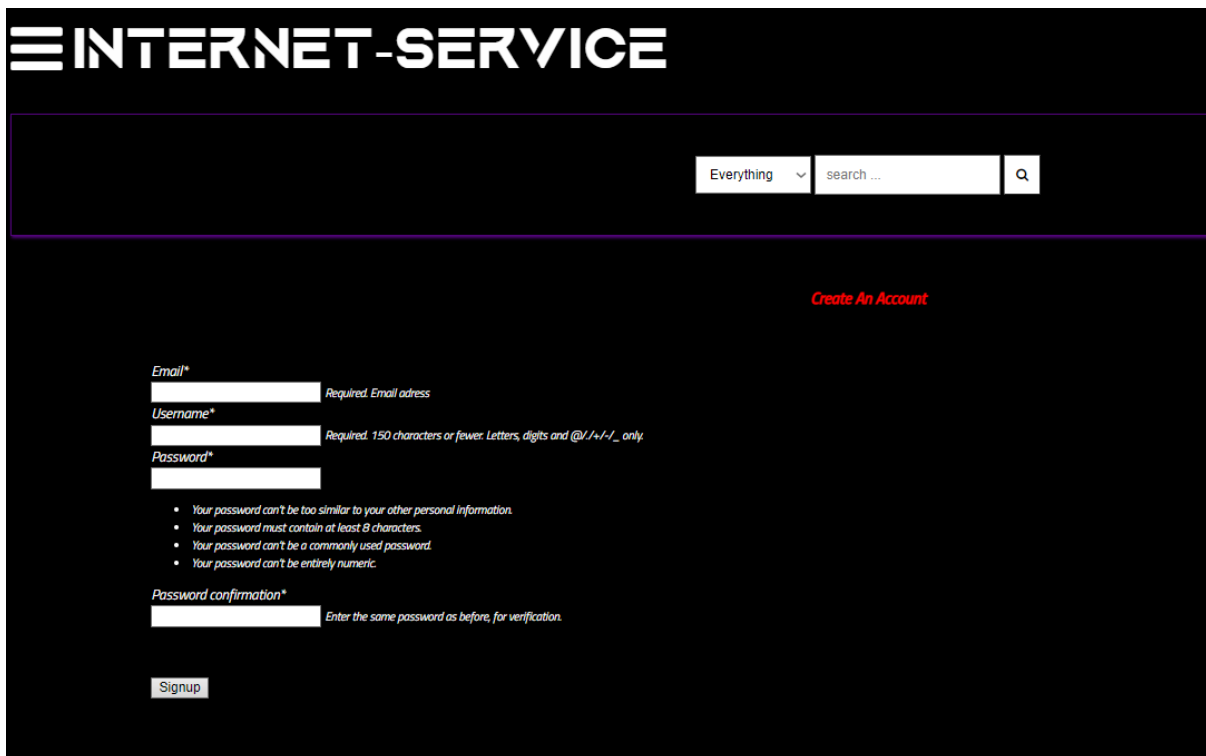


The screenshot displays the 'INTERNET-SERVICE' website interface. At the top, there is a navigation bar with the site name and a red plus sign. Below this is a search bar with a dropdown menu set to 'Everything' and a search icon. A red message reads 'Hey admin, Write something'. The main form for creating a topic contains the following elements:

- Title\***: A text input field.
- Content\***: A large text area for the main body of the post.
- Categories\***: A dropdown menu with options: 'Bugs and Glitches', 'Reviews', 'Betas and Suggestions', and 'Energy country when person loss human'.
- Tags\***: A text input field with a placeholder 'A comma-separated list of tags'.
- Save**: A button at the bottom of the form.

Рис. 4.4 – Створення теми

Якщо відвідувач сайту ще не має свого облікового запису, він може клацнути на кнопку «Реєстрація». Після цього йому відкриється сторінка із формою реєстрації, де йому потрібно ввести свої ім'я, логін, пошту та пароль, а також підтвердити пароль. Для завершення реєстрації необхідно натиснути кнопку «submit».



The image shows a registration form for 'INTERNET-SERVICE'. At the top, there is a search bar with a dropdown menu set to 'Everything' and a search icon. Below the search bar, there is a red link that says 'Create An Account'. The registration form consists of several input fields: 'Email\*' with a required email address note, 'Username\*' with a note about character limits and allowed characters, 'Password\*' with a list of password requirements, and 'Password confirmation\*' with a note to enter the same password for verification. A 'Signup' button is located at the bottom of the form.

**INTERNET-SERVICE**

Everything search ... Q

*Create An Account*

*Email\**  
Required. Email adress

*Username\**  
Required. 150 characters or fewer. Letters, digits and @/./+/-/\_ only.

*Password\**

- Your password can't be too similar to your other personal information.
- Your password must contain at least 8 characters.
- Your password can't be a commonly used password.
- Your password can't be entirely numeric.

*Password confirmation\**  
Enter the same password as before, for verification.

Signup

Рис. 4.5 – Регістрація

Щоб мати можливість виставляти оцінки та отримувати рекомендації, відвідувачу сайту необхідно авторизуватись. Для цього потрібно клікнути на кнопку «Увійти». Після цього користувачеві відкриється сторінка авторизації на якою йому пропонується ввести свій логін та пароль. Для завершення процесу входу на сайт потрібно натиснути кнопку «Увійти». Далі користувачу відкривається головна сторінка і з'являється кнопка «Вийти», натиснувши яку можна вийти з профілю. Авторизований користувач може створювати теми, залишати повідомлення у темах та отримувати рекомендації.

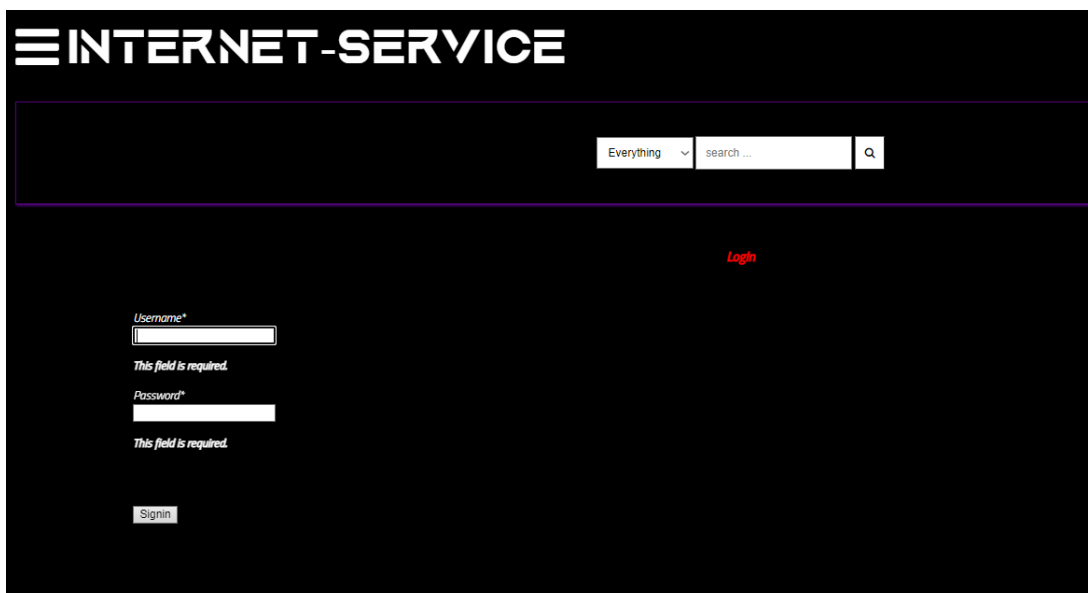


Рис. 4.6 – Авторизація

### 3.4 Сторінка рекомендацій

Якщо користувач авторизований, він може скористатися рекомендаційною системою, яку надає сайт. Для цього необхідно натиснути кнопку «Рекомендації», розташовану на головній сторінці, після чого в області відображення всіх альбомів з'являться ті альбоми, які користувач ще не оцінив, крім того, вони будуть розташовані в порядку зменшення: від більш цікавих до менш цікавих.

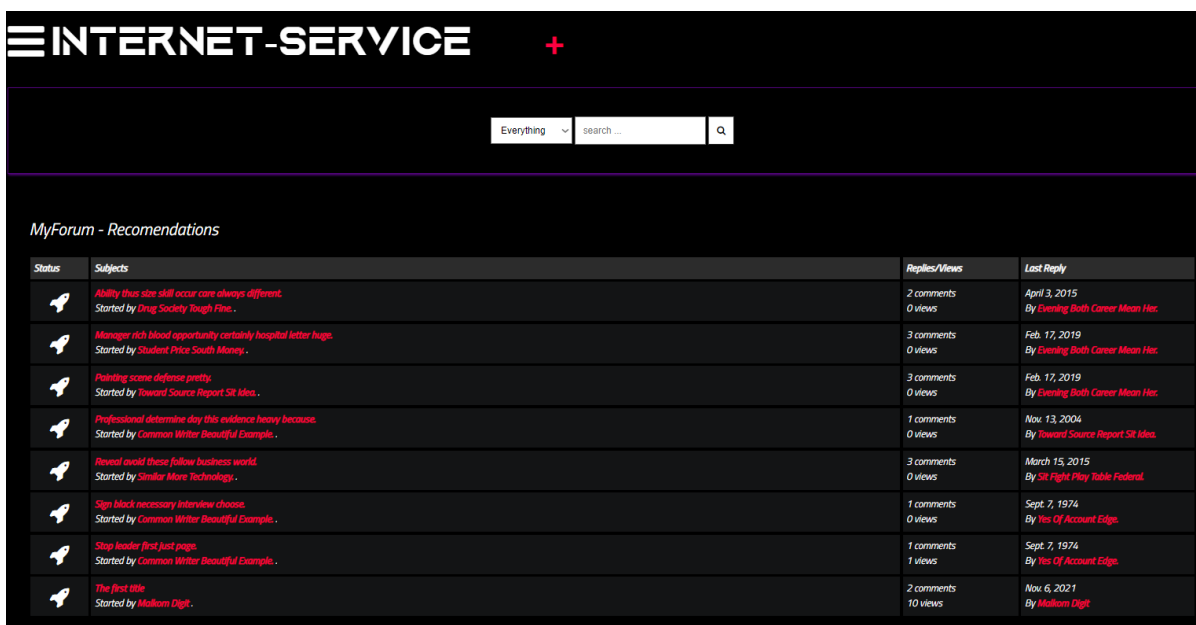


Рис. 4.7 – Сторінка рекомендацій

## ВИСНОВКИ

У результаті виконання даної дипломної роботи було розроблено інтернет сервіс для спілкування між користувачами мовою Python.

Проведено аналіз веб-сервісів для спілкування між користувачами. Ключові проблеми, які можна виділити, пов'язані з системою пошуку. Це призводить до зменшення нових користувачів та активності на веб-сервісі.

1. Проведено аналіз веб-сервісів для спілкування між користувачами. Ключові проблеми, які можна виділити, пов'язані з системою пошуку. Це призводить до зменшення нових користувачів та активності на інтернет-сервісі.
2. Розглянуто програмне забезпечення, яке може бути використано для спілкування між користувачами. Ключовими недоліками існуючих систем є відсутність рекомендаційних систем.
3. Проведено аналіз засобів розробки програмного забезпечення. Обрано мову програмування Python, що забезпечує швидкість, безпеку, зручність.
4. Розроблено структуру інтернет-сервісу для спілкування між користувачами та структуру бази даних.
5. Розроблений інтернет-сервіс для спілкування між користувачами.
6. Розроблена система рекомендацій тем користувачу.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

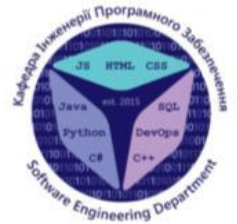
1. Internet Forums in Class: Innovation for Science Education by Focusing on Social Media. Johanna Dittmar, Ingo Eilks. University of Bremen. [www.arisejournal.com](http://www.arisejournal.com). ARISE Action Research and Innovation in Science Education, 2020.
2. Advances in Collaborative Filtering, Yehuda Koren, Robert Bell, 2022.
3. Веб-сайт: визначення й застосування. [Електронний ресурс]. – Режим доступу: <http://www.webtec.com.ua/uk/articles/index/view/2011-05-05/web-site> (дата звернення: 01.11.2021). – Назва з екрана.
4. Коггзолл Д. PHP 5. Полное руководство.: Пер. с англ. – М.: Издательский дом Вильямс, 2006. – 772 с.
5. Веб-ресурс “<https://linkbuilder.su/glossary/veb-forum>” дата звернення 11.04.2022
6. Оганесян А. Економіка підприємства (курс лекцій). М.: Видавничий дім "Вільямс", 1999. - 608 с.
7. Пейн К. Освой самостійно ASP.NET за 21 день: Пер. з англ. - М.: Видавничий будинок Вільям, 2002. - 832 с.
8. Санітарні правила та норми. Гігієнічні вимоги до відеодисплейних терміналів, персональних електронно-обчислювальних машин та організації роботи. СанПін 2.2.2/2.4.1340–03
9. Веб-ресурс <https://uk.wikipedia.org/wiki/Reddit> дата звернення 20.04.2022
10. Глібовець, Н.М. Створення рекомендаційної системи навчального типу з використанням фреймворку/Н.Н. Глібовець, М.О. Сидоренко // Проблеми інтелектуалізації комп'ютера: зб. ст. /Інститут кібернетики ім. В.М. Глушкова НАН України. - Київ, 2012. С. 176 - 181.
11. Recommender systems: an introduction / D. Jannach, M. Zanker, A. Felferning [et al.]. – New-York : Cambridge University Press, 2011. – 352 p
12. Джонс, М. Рекомендаційні системи: Частина 1. Введення у підходи та алгоритми. – Дата звернення : 29.04.2014. URL : <http://www.ibm.com/developerworks/ru/library/os-recommender1.html> (дата звернення: 22.04.2022).
13. Сегеран, Т. Програмуємо колективний розум / Т. Сегеран; пров. з англ. А. Слінкіна. - Спб. : Символ-Плюс, 2008. - 368 с.

14. Джонс, М. Рекомендаційні системи: Частина 2. Механізми з відкритим кодом. – Дата звернення : 29.04.2014. URL : <http://www.ibm.com/developerworks/gu/library/os-recommender2.html> (дата звернення: 27.04.2022).
15. Буч, М. Мова UML. Посібник користувача. [Електронний ресурс]/Г. Буч, Д. Рамбо, І. Якобсон. - Електрон. дано. - М.: ДМК Прес, 2008. - 496 с. – Режим доступу: <http://e.lanbook.com/book/1246> – Загл. з екрану.

## ДОДАТОК А



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ  
ТЕХНОЛОГІЙ  
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



### Розробка інтернет-сервісу для спілкування між користувачами мовою Python

Виконав студент 4 курсу  
групи ПД-42

Кряж Ілля Максимович

Керівник роботи

Доцент. Штіммерман Аксенія Миколаївна

Київ – 2022

### АНАЛОГИ

Назва	Харків Форум	Reddit	4chan
Платформи	web	web, мобільний додаток	web
Ієрархія розділів	+	+	+
Система пошуку	+	+	+
Модерація	+	+	-
Реклама	+	-	-
Цитування повідомлень	+	-	-
Можливість додавати медіа контент у повідомлення	-	+	+
Системи рекомендацій	-	-	-

# МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

**Мета роботи** – спрощення процесу спілкування між користувачами за рахунок користування інтернет-сервісу розробленого на мові Python.

**Об'єкт дослідження** – процес рекомендацій тем у веб форумах на основі сучасних рекомендаційних систем

**Предмет дослідження** – Інтернет-сервіс для спілкування між користувачами.

3

## Технічне завдання

1. Розробити базу даних.
2. Розробити систему авторизації та реєстрації.
3. Розробити функціонал перегляду розділів та тем.
4. Розробити можливість додавання повідомлень у теми.
5. Розробити адмін-панель для менеджера.
6. Розробити рекомендаційну систему.

4

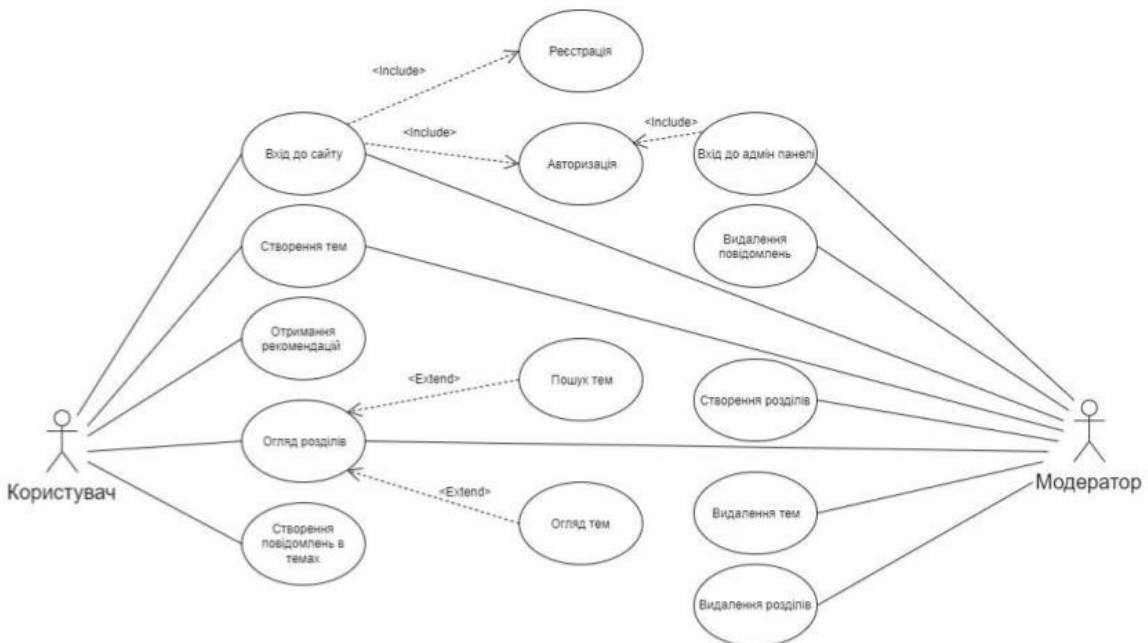


# ЗАСОБИ РОЗРОБКИ



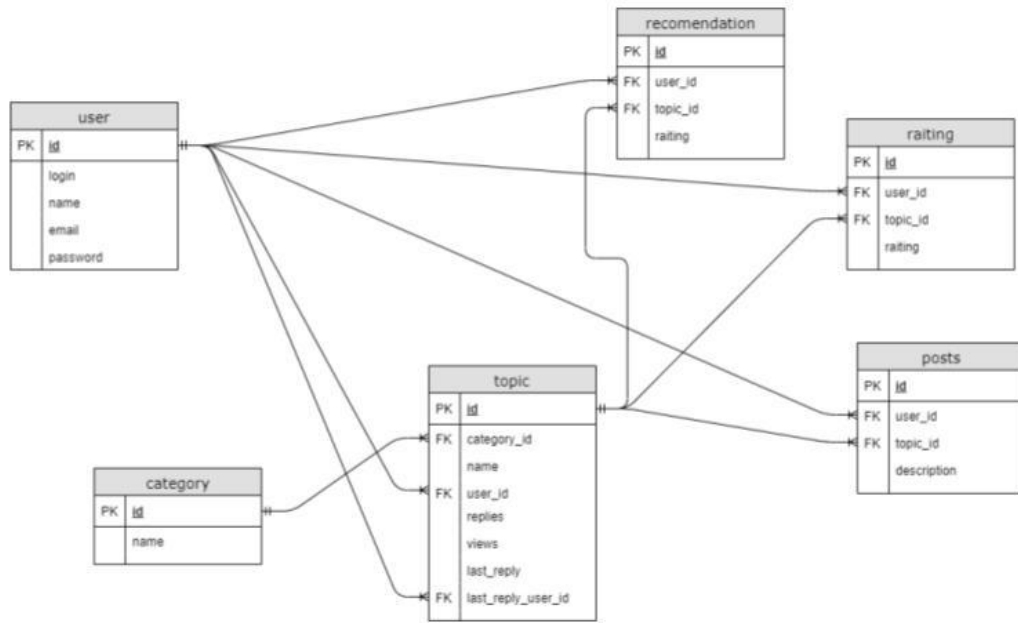
5

## Діаграма прецедентів



6

## Схема бази даних



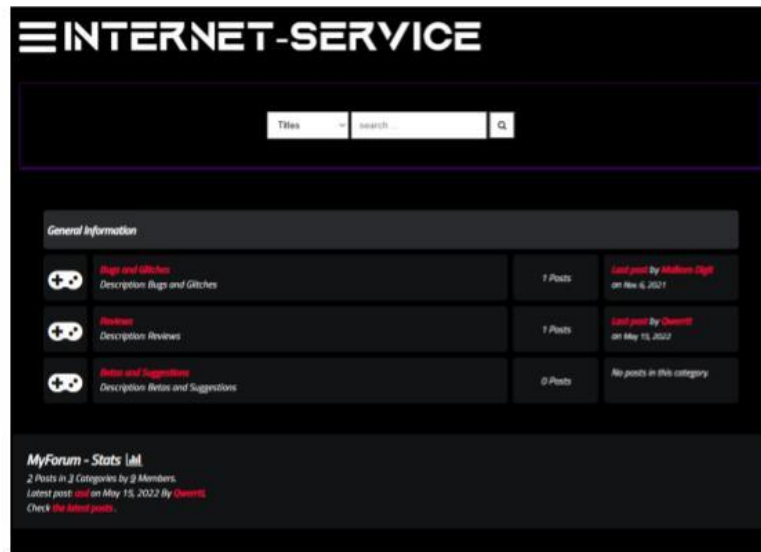
7

## Діаграма діяльності рекомендаційної системи



8

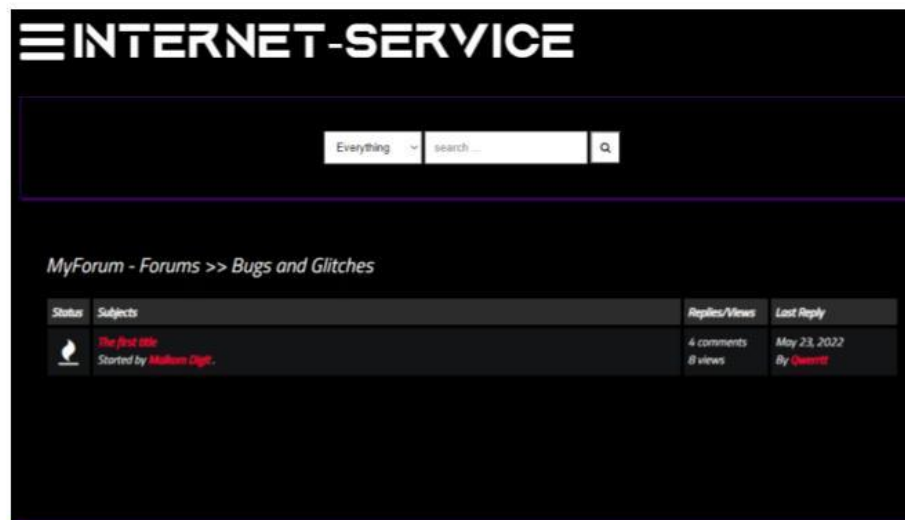
## Екранні форми



Сторінка розділів форуму

9

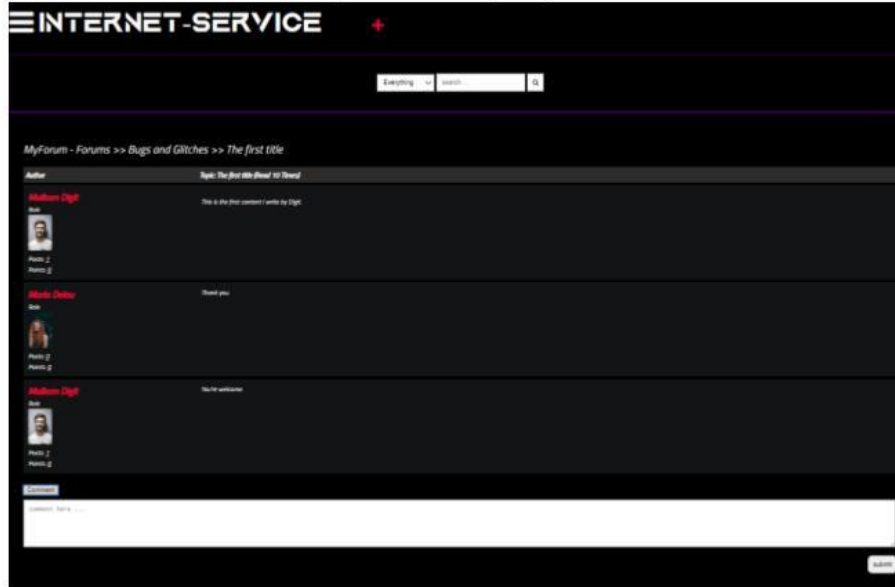
## Екранні форми



Сторінка тем форуму

10

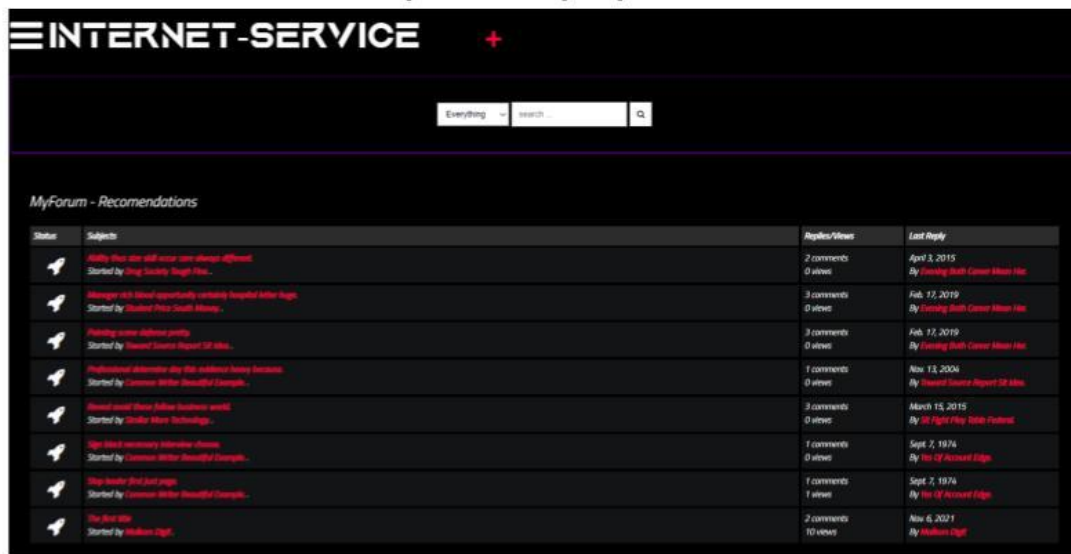
# Екранні форми



Сторінка повідомлень в темі

11

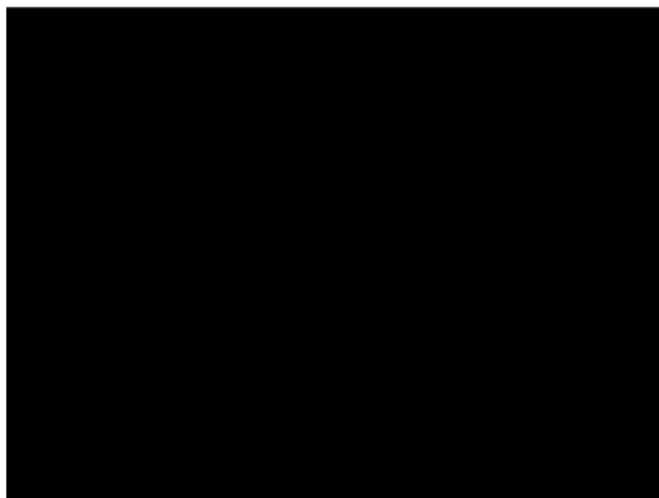
# Екранні форми



Сторінка рекомендацій

12

Відео роботи застосунку



13

## АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

Кряж І.М. Розробка веб-ресурсу для спілкування між користувачами мовою Python: Матеріали науково-технічної конференції застосування програмного забезпечення в інфокомунікаційних технологіях. Збірник матеріалів. 20.04.2022, ДУТ, м. Київ — К.:ДУТ, 2022, — С. 29.

14

## ВИСНОВКИ

1. Проведено аналіз веб-сервісів для спілкування між користувачами. Ключові проблеми, які можна виділити, пов'язані з системою пошуку. Це призводить до зменшення нових користувачів та активності на інтернет-сервісі.
2. Розглянуто програмне забезпечення, яке може бути використано для спілкування між користувачами. Ключовими недоліками існуючих систем є відсутність рекомендаційних систем.
3. Проведено аналіз засобів розробки програмного забезпечення. Обрано мову програмування Python, що забезпечує швидкість, безпеку, зручність.
4. Розроблено структуру інтернет-сервісу для спілкування між користувачами та структуру бази даних.
5. Розроблений інтернет-сервіс для спілкування між користувачами.
6. Розроблена система рекомендацій тем користувачу.

15

**ДЯКУЮ ЗА УВАГУ!**