

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіоелектроніки,
Факультет комп'ютерних наук і технологій

(повне найменування інституту, назва факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ
РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ ДЛЯ ПОРТАЛУ НОВИН
RESEARCH AND SOFTWARE IMPLEMENTATION OF THE
RECOMMENDATION SYSTEM FOR A NEWS PORTAL

Виконав: студент 2 курсу, групи КНТ-119м
Спеціальності 121 Інженерія програмного
забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

Макаренко Р.І.

(прізвище та ініціали)

Керівник Колпакова Т.О.

(прізвище та ініціали)

Рецензент Казурова А.Є.

(прізвище та ініціали)

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

Інститут, факультет ІРЕ, ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти магістр
 Спеціальність 121 Інженерія програмного забезпечення
 (КОД І НАЙМЕНУВАННЯ)
 Освітня програма (спеціалізація) Інженерія програмного забезпечення
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
С.О. Субботін
 “ ” 20__ року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

Макаренка Родіона Івановича

(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна реалізація рекомендаційної системи для порталу новин Research and software implementation of the recommendation system for a news portal

керівник проєкту (роботи) Колпакова Тетяна Олексіївна, к.т.н., доцент,
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “3” листопада 2020 року № 301

2. Строк подання студентом проєкту (роботи) 5 грудня 2020 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз проблеми та постановка завдань дослідження. 2. Матеріали і методи. 3. Розробка архітектури програми. 4. Розробка програми. 5. Експлуатація, тестування та експериментальне дослідження програми. 6. Організаційно-економічна частина. 7. Охорона праці та цивільна безпека. 8. Додатки

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

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-5 Основна частина	Колпакова Т.О., доцент		
Організаційно-економічна частина	Пожуєва Т.О., професор		
Охорона праці та цивільна безпека	Коробко О.В., ст. викладач		
Нормоконтролер	Дейнега Л.Ю., ст. викладач		

7. Дата видачі завдання “ 4 ” вересня 2020 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	2–3 тижні	Розділ 1
3	Розробка та удосконалення методів, моделей й алгоритмів вирішення задачі.	4–5 тижні	Розділ 2
4	Розробка архітектури програми.	6 тиждень	Розділ 3
5	Розробка програми.	7–8 тижні	Розділ 4
6	Тестування та експериментальне дослідження програмного забезпечення.	9 тиждень	Розділ 5
7	Розробка організаційно-економічної частини	10 тиждень	Розділ 6
8	Розробка заходів з охорони праці та цивільної безпеки.	11 тиждень	Розділ 7
9	Оформлення пояснювальної записки та документів до неї.	12-13 тиждень	Додатки
10	Нормоконтроль та рецензування.	14–15 тижні	
11	Захист роботи.	16 тиждень	

Студент(ка)

(підпис) Макаренко Р.І.
(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис) Колпакова Т.О.
(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
107 с., 7 табл., 19 рис., 3 дод., 31 джерел.

НОВИНИ, ІНТЕРНЕТ ПОРТАЛ, РЕКОМЕНДАЦІЇ, МАШИННЕ
НАВЧАННЯ, КЛІЄНТ, СЕРВЕР, ЗАПИТ

Об'єкт дослідження: дослідження та програмна реалізація
рекомендаційної системи для порталу новин.

Предмет дослідження: програмний застосунок для рекомендацій новин
на основі вподобань користувача.

Мета роботи: створення програмної рекомендаційної системи підбору
новин для кожного читача. Удосконалення методу к найближчих сусідів.

Матеріали та методи: Мова програмування PHP, бібліотека PHP
Machine learning.

Технічні засоби: для розробки використано персональний комп'ютер з
процесором Intel Core 2 Duo під управлінням операційної системи Microsoft
Windows 10.

Результати: Створено веб сайт новин, який підбирає кожному читачеві
новини, за темою якою він цікавиться.

Висновки. Розроблено програмну систему для рекомендацій новин за
інформацією про попередні перегляди користувача.

Галузь використання: інтернет портали новин.

ABSTRACT

Explanatory note to the master's thesis: 107 pp., 7 tables, 19 figures, 3 appendices, 31 sources.

NEWS, INTERNET PORTAL, RECOMMENDATIONS, MACHINE LEARNING, CLIENT, SERVER, REQUEST

Object of study is a research and software implementation of the recommendation system for the news portal.

Subject of study is a software application for news recommendations based on user preferences.

The purpose of the work is to create a software recommendation system for selecting news for each reader. Improving the method to the nearest neighbors

Materials and methods: PHP programming language, PHP Machine learning library

Hardware: Developed using a personal computer with an Intel Core 2 Duo processor running Microsoft Windows 10.

Results: A news website has been created, which selects news for each reader on the topic he is interested in.

Conclusions. A software system has been developed for news recommendations based on information about user previews.

Field of use is a Internet news portals.

ЗМІСТ

	С.
Перелік умовних позначень і скорочень.....	10
Вступ.....	11
1 Аналіз проблем та постановка завдань дослідження	12
1.1 Огляд існуючих методів вирішення	12
1.2 Огляд існуючих програмних продуктів, що вирішують аналогічні завдання.....	14
1.2.1 Програмний продукт Google новини.....	14
1.2.2 Програмний продукт News360.....	15
1.2.3 Програмний продукт Flipboard	16
1.3 Огляд методів рекомендаційних систем	18
1.4 Постановка завдання роботи	19
1.5 Висновки за розділом 1	19
2 Матеріали і методи.....	21
2.1 Вибір підходу розробки	21
2.1.1 Метод,орієнтований на зміст(Content-Based).....	21
2.1.2 Метод, орієнтований на користувача(User-Based).....	22
2.1.3 Метод, орієнтований на елемент(Item-Based)	24
2.2 Обґрунтування вибору	25
2.3 Вибір алгоритму для реалізації системи	25
2.4 Алгоритм пошуку k найближчих сусідів	25
2.5 Бібліотека PHP machine learning	26
2.6 Висновки за розділом 2	27
3 Розробка архітектури програми.....	28
3.1 Нормалізація даних.....	28
3.1.1 Коефіцієнт кореляції Пірсона.	28
3.1.2 Коефіцієнт кореляції Спірмана.....	29
3.2 Оцінювання якості роботи системи	29

3.2.1	Статистичні метрики.....	29
3.2.2	Метрики MAE і RMSE.....	30
3.2.3	Метрики підтримки прийнятих рішень.....	30
3.3	Розроблений метод	Error! Bookmark not defined.
3.4	Можливі проблеми системи	32
3.5	Висновки за розділом 3	33
4	Розробка програми	34
4.1	Опис основних етапів розробки	34
4.2	Процес навчання системи	34
4.2.1	Опис класу «Learning»	34
4.2.2	Захист від дублювання інформації	36
4.2.3	Захист від запису неавторизованих користувачів.....	36
4.3	Процес використання отриманих знань	36
4.3.1	Отримання даних.....	37
4.3.2	Перетворення даних	37
4.3.3	Використання методу найближчих сусідів.....	37
4.4	Порівняння оригінального алгоритму і створеного	37
4.5	Висновки за розділом 4	40
5	Експлуатація та тестування програми	41
5.1	Керівництво програміста	41
5.1.1	Призначення програми	41
5.1.2	Функції, виконувані програмою	41
5.1.3	Умови, необхідні для застосування програми.....	41
5.1.4	Опис основних характеристик програми.....	42
5.1.5	Опис основних особливостей програми	42
5.1.6	Вхідні дані	42
5.1.7	Вихідні дані.....	42
5.1.8	Повідомлення.....	42
5.2	Керівництво користувача.....	43

5.2.1	Призначення та функції програми.....	43
5.2.2	Умови виконання програми	43
5.2.3	Виконання програми	44
5.3	Керівництво системного програміста.....	44
5.3.1	Призначення та функції програми.....	44
5.3.2	Структура програми	44
5.3.3	Налаштування програми.....	47
5.3.4	Перевірка	48
5.3.5	Додаткові можливості.....	48
5.4	Тестування програми.....	48
5.4.1	Тестовий випадок №1	49
5.4.2	Тестовий випадок №2	49
5.4.3	Тестовий випадок №3	50
5.4.4	Тестовий випадок №4	51
5.4.5	Тестовий випадок №5	52
5.5	Висновки за розділом 5	53
6	Організаційно-Економічна частина	54
6.1	Планування розробки системи	54
6.2	Планування розробки	54
6.3	Визначення витрат на розробку	55
6.3.1	Розрахунок основної заробітної плати	56
6.3.2	Розрахунок додаткової заробітної платні	56
6.3.3	Відрахування на єдиний соціальний внесок.....	57
6.3.4	Визначення витрат на матеріали.....	57
6.3.5	Витрати на ПК	58
6.3.6	Витрати на розробку	61
6.3.7	Витрати на сервер.....	62
6.3.8	Накладні витрати	63
6.3.8	Розрахунок експлуатаційних витрат	64

6.3.9 Розрахунок витрат на адміністратора	65
6.4 Розрахунок капітальних вкладень.....	65
6.5 Оцінка економічної ефективності	66
6.6 Висновки за розділом 6	67
7 Охорона праці та безпека у надзвичайних ситуаціях.....	69
7.1 Аналіз потенційних небезпек	69
7.2 Заходи із забезпечення безпеки.....	71
7.3 Заходи з забезпечення виробничої санітарії та гігієни праці.....	74
7.4 Заходи безпеки у надзвичайних ситуаціях.....	80
7.4.1 Заходи з пожежної безпеки	80
7.4.2 Заходи з цивільного захисту. Вплив електромагнітного імпульсу на елементи виробництва	83
7.5 Висновки за розділом 7	85
Висновки	86
Перелік джерел і посилання.....	87
Додаток А Технічне завдання	90
Додаток Б Текст розробленої програми.....	94
Додаток В Слайди презентації.....	101

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ І СКОРОЧЕНЬ

БД	— база даних;
КТЗ	— комплекс технічних засобів;
ООП	— об’єктно-орієнтоване програмування;
ПП	— програмний продукт;
СКБД	— система керування базами даних;
ТЗ	— технічне завдання;
VHLL	— мови програмування надвисокого рівня;
MAE	— середня абсолютна помилка;
RMSE	— середньоквадратична помилка;
TF-IDF	— Term frequency Inverse document frequency.

ВСТУП

Згідно з щорічним звітом «Digital 2020» у 2020 році інтернетом користувалось 4.5 мільярди людей у світі. Кожен з цих користувачів має певні вподобання та теми, що їх цікавлять [1]. Потік інформації, що регулярно споживає користувач через засоби масової інформації такі як газети, телебачення, радіо, інтернет, постійно зростає, і вже немає фізичної можливості регулярно переглядати всі новини по всім темам, що обговорюються в світі.

Люди все більше жадають персоналізованої добірки новин, яка б враховувала тільки ті теми, що їх цікавлять, приховуючи зайві або небажані теми.

Тож актуальність створення рекомендаційної системи пов'язана з потребою в фільтрації інформації і полягає в демонстрації користувачеві лише того, що він з більшою вірогідністю побажає прочитати. Для вирішення даної проблеми необхідно створити модель, яка б запам'ятовувала інтереси кожного користувача, зберігала їх в базі даних та використовувала тоді, коли користувач знову завітав на сайт за новою добіркою новин.

Програмний продукт матиме клієнт-серверну архітектуру, що дозволить користуватися ним одночасно декількома користувачами, а також має бути кросбраузерним.

1 АНАЛІЗ ПРОБЛЕМ ТА ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ

1.1 Огляд існуючих методів вирішення

Новина – оперативне інформаційне повідомлення, яке містить суспільно важливу та актуальну інформацію, що стосується певної сфери життя суспільства загалом чи окремих його груп. В журналістиці — окремий інформаційний жанр, який характеризується стислим викладом ключової інформації щодо певної події, яка сталася нещодавно.

Написання новин полягає у пошуку відповідей всього на шість основних запитань: «Хто?», «Що?», «Де?», «Коли?», «Чому?», «Яким чином?». Іноді цей список доповнюють питаннями «Що це означає?» «Кому вигідно?» і «Що буде?», але головними залишаються перші шість. Також у новинному повідомленні обов'язково вказується джерело інформації – дається відповідь на запитання «Звідки ми це знаємо?».

Новинні замітки бувають короткі та розширені. Коротка новинна замітка складається з 10-20 рядків і містить відповіді на ці головні 6 запитань.

Залежно від інформаційного приводу розрізняють такі типи новин:

- новина-факт – повідомлення про ситуацію, наприклад про прийняття нового закону, рішення суду чи результати переговорів. Замітка починається з найважливішого аспекту цієї ситуації;

- новина-подія – повідомлення про дію, наприклад покорення гірської вершини або про пограбування банку. При цьому дотримуватися хронології події не потрібно;

- новина-цитата – повідомлення про виступ якоїсь дуже важливої людини. Зазвичай новини-цитати містяться у розділах політики та економіки і присвячені тому, що політичні лідери та бізнесмени щось обіцяють або до чогось закликають.

В журналістиці виділяють декілька основних категорій новин:

- надзвичайні ситуації;
- злочини;
- місцеві і центральні органи влади;
- плани і проєкти;
- конфлікти і суперечки;
- господарство;
- здоров'я;
- визначні особи;
- спорт;
- погода;
- ситуації на шляхах [2].

Через велику кількість новин щоденно виникає потреба в їх фільтрації для кожної окремо взятої людини. Для вирішення даної задачі використовують рекомендаційні системи.

Рекомендаційна система – підклас системи фільтрації інформації, яка будує рейтинговий перелік об'єктів, яким користувач може надати перевагу.

На сьогоднішній день подібні системи поширені не тільки в сфері журналістики а й у інших сферах життя, наприклад інтернет магазинах

В процесі роботи рекомендаційні системи збирають дані про користувачів, використовуючи поєднання явних і неявних методів. Приклади явного збору даних:

- запит у користувача оцінки об'єкта за диференційованою шкалою;
- запит у користувача ранжування групи об'єктів від найкращого до найгіршого;
- пред'явлення користувачеві двох об'єктів з питанням про те, який з них краще;
- пропозиція створити список об'єктів, улюблених користувачем.

Приклади неявного збору даних:

- спостереження за тим, що оглядає користувач в інтернет-магазинах або базах даних іншого типу;
- ведення записів про поведінку користувача онлайн;
- відстеження вмісту комп'ютера користувача [3].

Рекомендаційні системи порівнюють однотипні дані від різних людей і обчислюють список рекомендацій для конкретного користувача. Для обчислення рекомендацій використовується граф інтересів. Рекомендаційні системи – зручна альтернатива пошуковим алгоритмам, так як дозволяють виявити об'єкти, які не можуть бути знайдені останніми [4].

1.2 Огляд існуючих програмних продуктів, що вирішують аналогічні завдання

Окрім програмного продукту, який представлений в даному проєкті, існують і інші продукти, які спроектовані для вирішення подібних проблем. Далі наведено декілька прикладів подібних продуктів.

1.2.1 Програмний продукт Google новини

Google Новини (англ. Google News) – безкоштовний агрегатор новин, який наданий і управляється компанією Google Inc. Сервіс розроблювався протягом трьох років з 2002 до 2005 року, окрім нього був створений застосунок для вибору найактуальніших новин з тисяч публікацій по всьому світу.

Система підбирає рекламу, ґрунтуючись на інформації, яку користувач додає до облікового запису. Окрім цього система також використовує дані аналізу інтересів користувача. Інтерфейс сервісу зображений на рисунку 1.1.

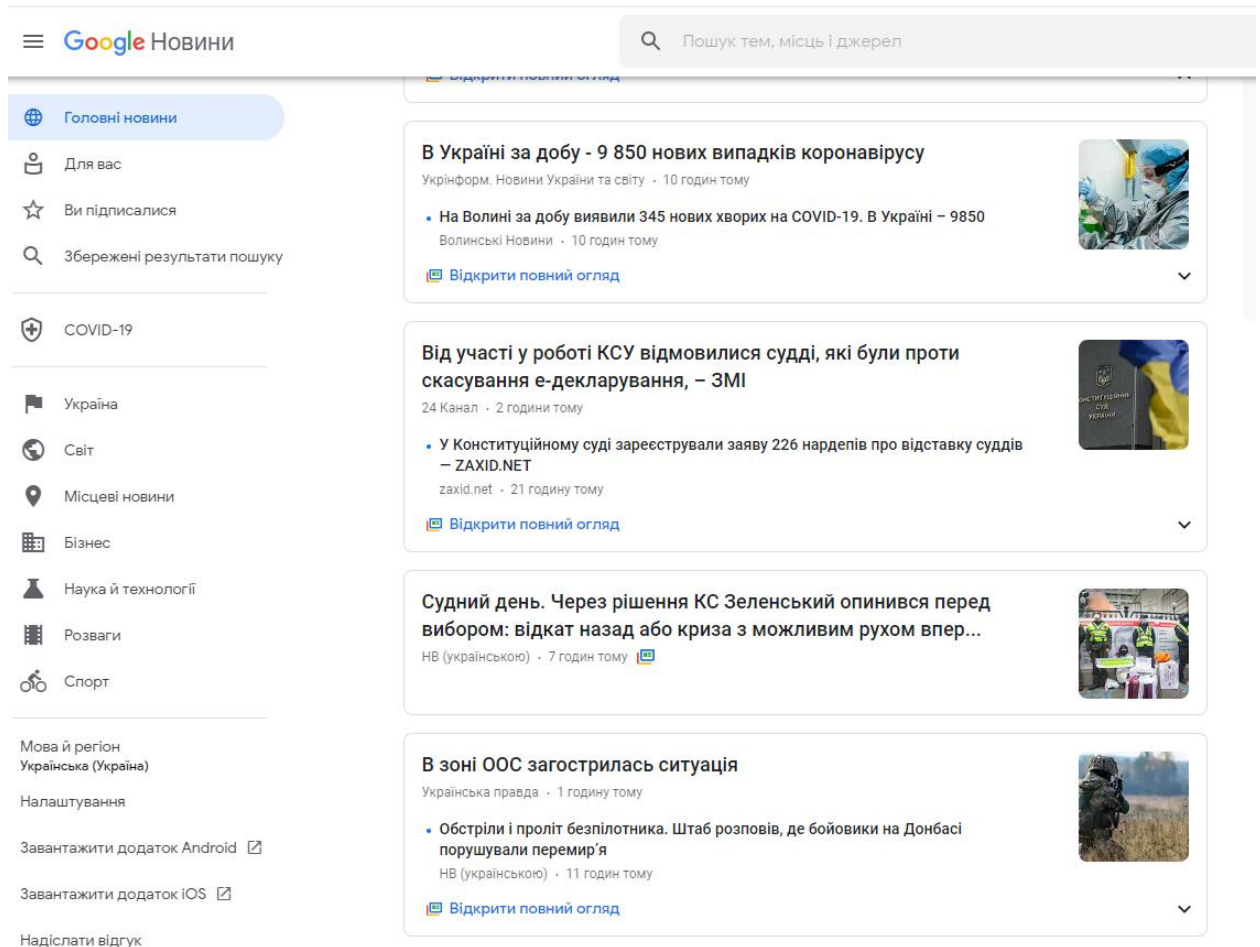


Рисунок 1.1 – Інтерфейс сервісу Google новини

Переваги програмного продукту Google новини:

- інтегрованість в усі смартфони на ОС Android;
- зручний інтерфейс мобільної версії.

Недоліки даного продукту наступні:

- наявність реклами;
- незручний інтерфейс веб-версії.

1.2.2 Програмний продукт News360

News360 – це персоналізований додаток для збору новин для смартфонів, планшетів та веб-браузерів. Він намагається дізнатися інтереси користувача, аналізуючи їх взаємодію з новинами в додатку та

використовуючи семантичний аналіз, щоб створити графік інтересів для кожного користувача. Додаток має аудиторію понад 4 мільйони користувачів.

Спочатку додаток був випущений для iPhone в кінці 2010 року, News360 тепер доступний для всіх основних мобільних платформ, з версіями для iPad, iPhone, Android, Windows 8, Windows Phone та веб-браузерів. Інтерфейс браузерної версії додатку зображений на рисунку 1.2.

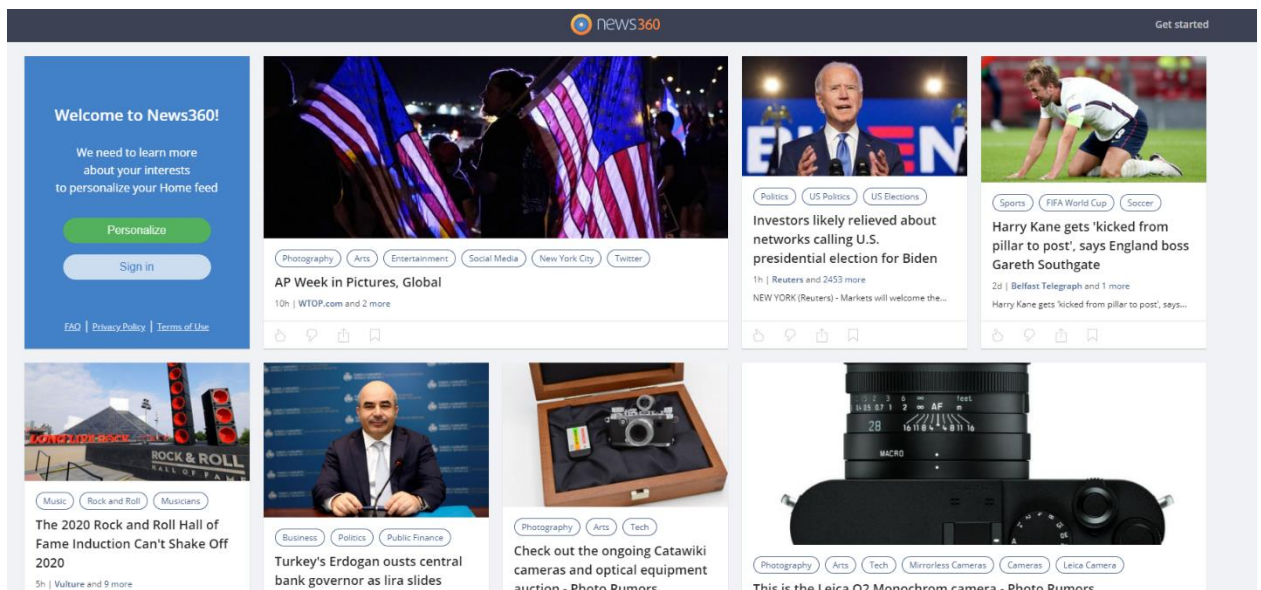


Рисунок 1.2 – Інтерфейс програми News360

Переваги програмного продукту News360:

- наявність як веб версії так і мобільної;
- велика база даних;
- ефективний алгоритм підбору новин.

Недоліки даного продукту наступні:

- відсутність української мови.

1.2.3 Програмний продукт Flipboard

Flipboard – компанія агрегатор новин та агрегатор соціальних мереж. Цей застосунок було вперше випущено в липні 2010 року. Він агрегує контент

із соціальних медіа, новин, сайтів для обміну фотографіями і відображає їх у форматі журналу. Читачі також можуть зберігати історії у журналах Flipboard. Окрім мобільних застосунків на iOS та Android, доступна також браузерна та десктопна версія Flipboard. У серпні 2018 року сервісом користувались 145 мільйонів людей щомісячно. Інтерфейс застосунку Flipboard зображений на рисунку 1.3.

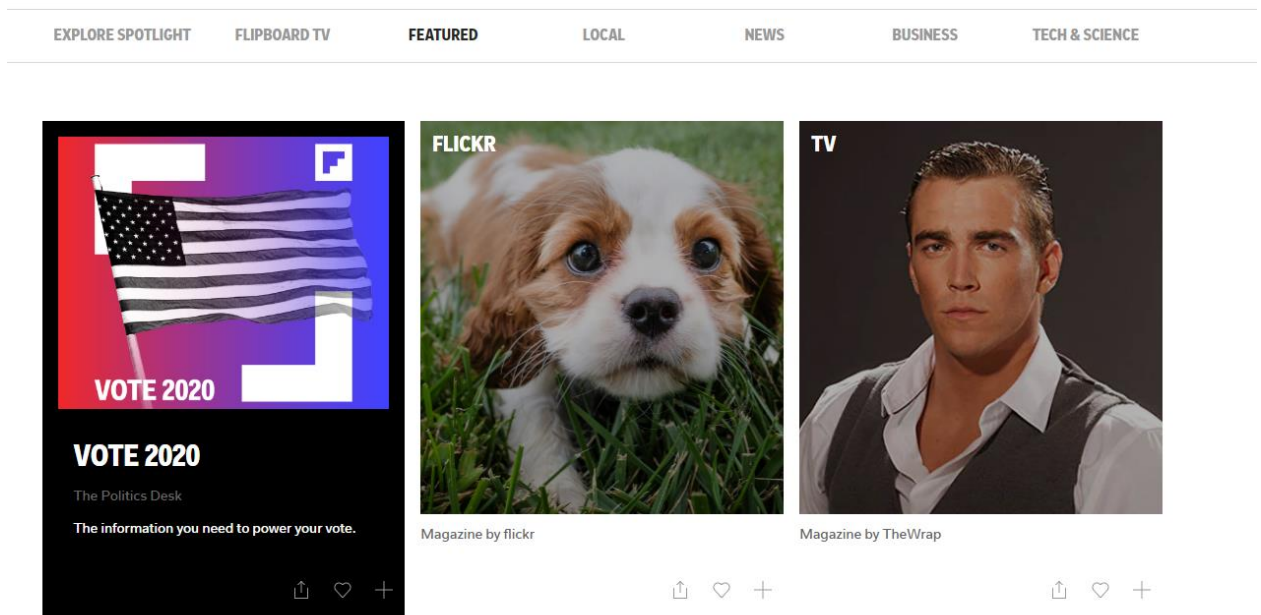


Рисунок 1.3 – Інтерфейс програми Flipboard

Переваги програмного продукту Flipboard:

- присутність на усіх актуальних платформах;
- інтеграція з соціальними мережами;
- зручний для користувача інтерфейс.

Недоліки даного продукту наступні:

- відсутність програмно реалізованої української мови.

1.3 Огляд методів рекомендаційних систем

У центрі будь-якої рекомендаційної системи знаходиться так звана матриця переваг. Це матриця, по одній з осей якої відкладені всі клієнти сервісу (Users), а по іншій – об'єкти рекомендації (Items). На перетині деяких пар (user, item) дана матриця заповнена оцінками (Ratings) – це показник зацікавленості користувача в даній темі, виражений за заданою шкалою. Приклад подібної матриці зображений на рисунку 1.4 [5].

				
John 	5	1	3	5
Tom 	?	?	?	2
Alice 	4	?	3	?

Рисунок 1.4 – Приклад рекомендаційної матриці

На прикладі фільмів показані інтереси різних користувачів. Користувачі зазвичай оцінюють лише невелику частину об'єктів інтересу, що є в каталозі, і завдання рекомендаційної системи – узагальнити цю інформацію і передбачити ставлення клієнта до інших об'єктів, про які нічого не відомо. Іншими словами потрібно заповнити всі незаповнені клітинки.

Поняття «цікавості» теж суб'єктивне. Деяким користувачам потрібні речі тільки з їхньої улюбленої категорії (conservative recommendations), а

хтось, навпаки, більше відгукується на нестандартні товари або групи товарів (risky recommendations). Наприклад, відеохостинг може рекомендувати користувачеві тільки нові серії улюбленого серіалу, а може періодично рекомендувати йому нові шоу або взагалі нові жанри.

Призначені для користувача оцінки можна отримати двома способами:

- явно (explicit ratings) – користувач сам ставить рейтинг товару, залишає відгук, лайкає сторінку;
- неявно (implicit ratings) – користувач явно своє ставлення не виражає, але можна зробити непрямий висновок з його дій: купив товар – значить він йому подобається, довго читав опис – значить є інтерес і т.п.

Звичайно, явні переваги краще, оскільки користувач сам говорить про те, що йому сподобалося. Однак на практиці далеко не всі сайти надають можливість явно виражати свій інтерес, та й не всі користувачі мають бажання це робити. Найчастіше використовуються відразу обидва типи оцінок і добре доповнюють один одного [6].

1.4 Постановка завдання роботи

Програмний продукт (ПП) повинен реалізувати метод, за допомогою якої буде ефективно заповнюватись рекомендаційна матриця для кожного окремого користувача в залежності від тем які його цікавлять. Дану інформацію програмний продукт повинен збирати в ході користування користувачем сайтом, тобто читанні новин.

1.5 Висновки за розділом 1

В першому розділі було розглянуто найпопулярніші ПП, які використовуються для освітлення останніх новин, і в яких є рекомендаційна

система. Було розглянуто їх переваги та недоліки, а також було доведено актуальність цієї теми.

На основі проаналізованих аналогів, а також існуючих методів вирішення завдання було сформовано основні функціональні вимоги до розроблюваного алгоритму.

2 МАТЕРІАЛИ І МЕТОДИ

2.1 Вибір підходу розробки

Рекомендаційні системи можуть ґрунтуватись на трьох різних підходах, які відрізняються початковим об'єктом для розгляду. Система може бути орієнтованою на зміст (Content-based), орієнтована на користувача (User-based) або на елемент (Item-based).

2.1.1 Метод, орієнтований на зміст(Content-Based)

Персональні рекомендації передбачають максимальне використання інформації про самого користувача, і в першу чергу про його попередні покупки або перегляди. Одним з перших з'явився підхід content-based filtering. В рамках даного підходу опис елементу (content) зіставляється з інтересами користувача, отриманими з його попередніх оцінок. Чим більше стаття цим інтересам відповідає, тим вище оцінюється потенційна зацікавленість користувача [7].

Неструктуровані ознаки описуються за допомогою векторів в просторі слів (Vector-Space model). Кожен елемент такого вектора – ознака, потенційно що характеризує інтерес користувача. Аналогічно, стаття на порталі новин – вектор в тому ж просторі.

Під час взаємодії користувача з системою, векторні описи переглянутих ним статей об'єднуються (підсумовуються і нормалізуються) в єдиний вектор і, таким чином, формується вектор його інтересів. Далі достатньо знайти статтю, опис якої найближче до вектору інтересів, тобто вирішити задачу пошуку k найближчих сусідів.

Не всі елементи однаково значущі: наприклад, союзні слова, очевидно, не несуть ніякого корисного навантаження. Тому при визначенні числа співпадаючих елементів в двох векторах все вимірювання потрібно

попередньо зважувати по їх значимості. Дану задачу вирішує перетворення тексту в TF-IDF (від англ. TF – term frequency, IDF – inverse document frequency).

TF-IDF – статистична міра, яка використовується для оцінки важливості слова в контексті документа, що є частиною колекції документів або корпусу. Вага деякого слова пропорційний частоті вживання цього слова в документі і обернено пропорційна частоті вживання слова в усіх документах колекції. Формула 2.1 описує процес пошуку найважливіших слів.

$$w_{x,y} = tf_{x,y} * \log \left(\frac{N}{df_x} \right), \quad (2.1)$$

де tf – частота слова x в описі;

df – кількість елементів, які мають в собі слово;

N – Загальна кількість елементів [8].

Для виявлення оцінки вимірюється евклідова відстань між сформованими векторами.

2.1.2 Метод, орієнтований на користувача (User-Based)

Даний клас систем почав активно розвиватися в 90-і роки XX ст. В рамках даного підходу рекомендації генеруються на підставі інтересів інших схожих користувачів. Такі рекомендації є результатом «колаборації» безлічі користувачів [9].

Класична реалізація алгоритму заснована на принципі k найближчих сусідів. Тобто для кожного користувача система буде шукати k найбільш схожих на нього (в термінах переваг) і доповнювати інформацію про користувача відомими даними по його сусідам. На рисунку 2.1 наведено схему роботи методу, орієнтованого на користувача, формула 2.2 використовується для відповідного розрахунку.

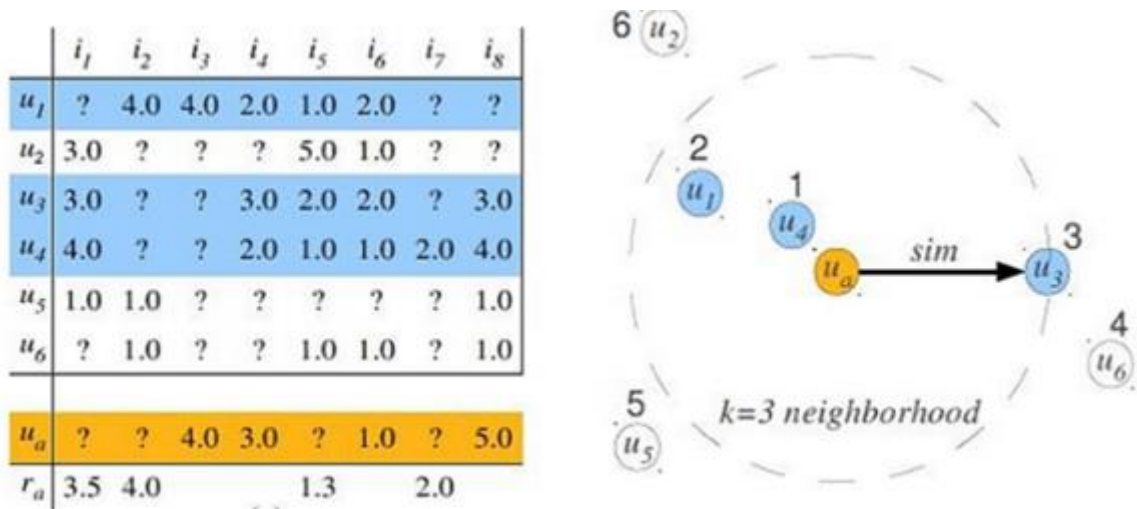


Рисунок 2.1 Реалізація методу, орієнтованого на користувача

$$r_{aj} = \frac{1}{\sum_{i \in N(a)} s_{ai}} \sum_{i \in N(a)} s_{ai} r_{ij}, \quad (2.2)$$

де N – кількість випадків;

a – кількість сусідів;

S – загальна кількість.

Для того, щоб алгоритм був ефективний, важливо щоб виконувалося кілька припущень.

- смаки людей не змінюються часом (або змінюються, але для всіх однаково);
- якщо смаки людей збігаються, то вони збігаються у всьому.

Наприклад, якщо два клієнта люблять одні фільми, то імовірно книги їм теж подобаються однакові. Так часто буває, коли рекомендовані товари однорідні (наприклад, тільки фільми). Якщо ж це не так, то у пари клієнтів цілком можуть збігатися переваги в їжі, а політичні погляди бути прямо протилежними – в таких ситуаціях ефективність алгоритму знижується.

2.1.3 Метод, орієнтований на елемент(Item-Based)

Підхід Item-based є природною альтернативою класичному підходу User-based, і майже повністю його повторює, за винятком одного моменту – застосовується він до транспонованої матриці переваг. Тобто шукає близькі товари, а не користувачів. Даний метод має багато переваг а саме:

- спрощує пошук найближчого сусіда;
- має більшу точність в оцінці близькості товарів;
- не вразлива до зміни смаку користувачів.

Системи колаборативної фільтрації зазвичай застосовують двоступеневу схему :

Цей алгоритм, винайдений Amazon включає в себе наступні кроки:

- будується матриця, визначальні відносини між парами предметів, для знаходження схожих предметів;
- використовуючи побудовану матрицю і інформацію про користувача, будуються прогнози його оцінок [10].

Також існує інша форма колаборативної фільтрації, яка ґрунтується на прихованому спостереженні звичайної поведінки користувача (на противагу явного, який збирає оцінки). У цих системах система спостерігає, як діє даний користувач, і як – інші (яку музику вони слухали, які відео подивилися, які композиції придбали), і використовує отримані дані, щоб передбачити поведінку користувача в майбутньому, або передбачити, як користувач бажав би вступити при наявності певної можливості. Ці передбачення повинні бути складені відповідно до бізнес-логікою, так як наприклад, марно пропонувати кому-небудь купити музичний файл, який у нього вже є [11].

2.2 Обґрунтування вибору

Для реалізації системи рекомендацій багатокористувацького порталу новин було вирішено обрати систему колаборативної фільтрації, орієнтовану на користувача.

2.3 Вибір алгоритму для реалізації системи

Для реалізації такої рекомендаційної системи необхідно використати алгоритм який би міг з легкістю знайти подібні до прочитаних користувачем статті. Для його реалізація найчастіше використовується алгоритм пошуку k найближчих сусідів.

2.4 Алгоритм пошуку k найближчих сусідів

Алгоритм K-Nearest Neighbours (KNN) – це контрольований алгоритм машинного навчання, який простий в реалізації і в той же час має можливість створювати надійні класифікації. Одним з найбільших переваг алгоритму k найближчих сусідів є те, що це лінива здатність до навчання, Це означає, що модель не вимагає навчання і може отримати право на класифікацію даних, на відміну від інших споріднених їм елементів машинного навчання, таких як метод опорних векторів, регресія, і багаторівневе сприйняття [12].

Для навчання класифікатора необхідно мати набір об'єктів, для яких є заздалегідь визначені класи. Ця множина називається навчальною вибіркою, її розмітка проводиться вручну, із залученням фахівців у досліджуваній області.

Для вирішення задачі класифікації алгоритм k найближчих сусідів порівнює евклідові відстані між точками та класифікує задану точку

Далі наведений опис алгоритму k найближчих сусідів.

Для кожного терміну $t \in$ список (id, x) , де x – значення координати по даному терміну: $X(t) = [(id1, x1), (id2, x2) \dots (idm, xm)]$, де $x1 \leq x2 \dots \leq xm$.

Три кроки алгоритму k найближчих сусідів для класифікації нового тексту:

- крок 1. Система формує масив з новинами за останні 5 днів;
- крок 1.1. Система обчислює координати (TF-IDF) текстів;
- крок 2. Для кожного терміну (t, x) система шукає найближче до x значення на $X(t) = [(id1, x1), (id2, x2) \dots (idm, xm)]$, використовуючи метод двійкового пошуку;
- крок 2-1. $i = 1; j = m$;
- крок 2-2. Поки $i < j$;
- крок 2-2-1. $k = (i + j) \text{ div } 2$;
- крок 2-2-1. Якщо $x_k < x$, то $i = k + 1$;
- крок 2-2-2. Якщо $x_k \geq x$, то $j = k$;
- після кроку 2-2 x_k є найближче значення до x на $X(t)$;
- крок 3. Система розглядає $2*k$ текстів навколо заданого (idk, xk) на $X(t)$, вважаючи відстань між текстами, система проводить порівняння і оновлення до найближчих сусідів даного тексту [13].

2.5 Бібліотека PHP machine learning

Для реалізації рекомендаційної системи можна використати бібліотеку «PHP machine learning» для мови програмування PHP. Дана бібліотека містить всі технології, які використовуються в машинному навчанні. До них належать задачі класифікації, регресії та кластеризації.

Оскільки метод пошуку k найближчих сусідів реалізований в даній бібліотеці, для вирішення задачі класифікації можна використати клас k -Nearest Neighbors, який реалізований в даній бібліотеці.

Одним з недоліків алгоритму, представленого в даній бібліотеці, є його неточність. Для вирішення даної проблеми необхідно модифікувати цей алгоритм.

2.6 Висновки за розділом 2

В даному розділі було обрано підхід до розробки програмного продукту, також цього було обрано метод k найближчих сусідів як найкращий варіант для створення рекомендаційної системи для веб-порталу новин. Для реалізації такої системи була обрана бібліотека PHP machine learning. Окрім цього була пояснена доцільність покращення алгоритму k найближчих сусідів.

3 РОЗРОБКА АРХІТЕКТУРИ ПРОГРАМИ

3.1 Нормалізація даних

Через проблему різної оцінки даних користувачами перед роботою з даними система повинна нормалізувати ці дані для подальшої роботи. Існує декілька можливостей для нормалізації:

- центрування (mean-centering) – з оцінок користувача просто віднімається його середня оцінка;
- стандартизація (z-score) – на додаток до центрування система ділить оцінку на стандартне її відхилення у користувача;
- подвійна стандартизація – перший раз система нормує оцінки користувача, другий раз – оцінки новини.

На інтернет порталах новин «оцінками» можуть виступати теми, на які користувач заходить найчастіше [14].

«Схожість» або кореляцію переваг декількох користувачів або декількох новин можна рахувати по різному. Для цього програма повинна порівняти два вектори. Зробити це можна наступними способами [15].

3.1.1 Коефіцієнт кореляції Пірсона

Кореляція Пірсона – класичний коефіцієнт, який цілком можна застосувати і при порівнянні векторів. Формула 3.1 описує розрахунок коефіцієнту кореляції Пірсона:

$$p = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}, \quad (3.1)$$

де x_i – значення змінної x ;

y_i – значення змінної y ;

$\bar{x}$ – середнє арифметичне для змінної x ;

$\bar{y}$ – середнє арифметичне для змінної y .

Основний мінус даного методу в тому, що коли перетин за оцінками низький, кореляція може бути високою просто випадково.

3.1.2 Коефіцієнт кореляції Спірмана

Коефіцієнт кореляції Спірмана – міра лінійного зв'язку між випадковими величинами. На відміну від кореляції Пірсона, кореляція Спірмена є ранговою, тобто для оцінки сили зв'язку використовуються не чисельні значення, а відповідні їм ранги. Коефіцієнт інваріантний по відношенню до будь-якого монотонного перетворення шкали вимірювання(3.2) [16].

$$p = 1 - \frac{6 \sum d_i^2}{n(n^2-1)}, \quad (3.2)$$

де n – кількість ознак;

d – різниця між рангами за двома змінними.

3.2 Оцінювання якості роботи системи

Основна перешкода при розробці систем рекомендацій – вибір метрик для оптимізації. Це може бути складно, тому що в багатьох випадках мета – Не рекомендувати все ті ж елементи, з якими користувач взаємодіяв раніше.

3.2.1 Статистичні метрики

Вони використовуються для оцінки точності методу фільтрації шляхом порівняння прогнозованих рейтингів безпосередньо з фактичним рейтингом

користувачів. Середня абсолютна помилка (MAE), середньоквадратична помилка (RMSE) і кореляція зазвичай використовуються в якості статистичних метрик. Метод MAE є найбільш популярним і широко використовуваним – це міра відхилення рекомендації від фактичної вартості користувача. Прикладами таких метрик можуть виступати MAE і RMSE [17].

3.2.2 Метрики MAE і RMSE

Чим нижче значення MAE (формула 3.3) і RMSE (формула 3.4), тим точніше механізм рекомендацій прогнозує призначені для користувача рейтинги. Ці метрики зручні, коли рекомендації засновані на прогнозуванні рейтингу або кількості транзакцій. Вони дають уявлення про те, наскільки точні прогнози системи і, в свою чергу, наскільки точні рекомендації.

$$MAE = \frac{1}{N} \sum |PB - act| \quad (3.3)$$

$$RMSE = \sqrt{\frac{1}{N} \sum (PB - act)^2}, \quad (3.4)$$

де PB – результат який очікується;

act – актуальний результат.

3.2.3 Метрики підтримки прийнятих рішень

Корисними метриками серед них є Precision (точність) і Recall (повнота). Вони допомагають користувачам вибирати продукти, які більш схожі серед доступного набору продуктів. Метрики розглядають процедуру прогнозування як бінарну операцію, яка відрізняє хороші елементи від тих, які не є хорошими. Розрахунки точності та повноти відображено на формулах 3.5 та 3.6

$$P = \frac{rv}{ra} \quad (3.5)$$

$$r = \frac{rv}{ri}, \quad (3.6)$$

де rv – кількість підходящих рекомендацій;

ra – загальна кількість рекомендацій;

ri – загальна кількість елементів.

3.3 Розроблений метод

Для оптимізації алгоритму побудови персональних рекомендацій необхідно переглянути принцип роботи алгоритму k найближчих сусідів

Більшість моделей алгоритму k найближчих сусідів використовують евклідову або манхеттенську відстань в якості показника відстані до точки. Ці показники прості і добре працюють в різних ситуаціях.

Однак метрики відстані, нечасто використовують косинусну подібність між створеними векторами, так як вона порушує нерівність трикутника, і не працює з негативними даними.

Однак цю систему доцільніше використати в рекомендаційній системі багатокористувацького веб порталу новин із застосуванням косинусної подібності векторів. Нормалізація даних за допомогою косинусної подібності векторів збільшує точність роботи алгоритму k найближчих сусідів на 17%. Недоліком такого підходу є підвищений час роботи програми оскільки розрахунок цієї величини потребує більших підрахунків, ніж пошук стандартної евклідової відстані між векторами. Однак оскільки всі розрахунки будуть проходити на серверній частині веб сайту, то даним недоліком можна знехтувати. Розрахунок косинусної відстані наведений у формулі 3.7.

$$\text{cosine}(x, y) = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}}, \quad (3.7)$$

де x – перший показник;

y – другий показник.

3.4 Можливі проблеми системи

Для реалізації системи рекомендацій для багатокористувацького веб-порталу новин потрібно ознайомитись з можливими проблемами які можуть виникнути в ході роботи алгоритмів. Далі наведено ці можливі проблеми:

- розрідженість даних: як правило, більшість комерційних рекомендаційних систем заснована на великій кількості даних, в той час як більшість користувачів не ставить оцінки товарам. В результаті цього матриця «предмет-користувач» виходить дуже великою і розрідженою, що представляє проблеми при обчисленні рекомендацій;

- проблема холодного старту: нові предмети або користувачі представляють велику проблему для рекомендаційних систем. Частково проблему допомагає вирішити підхід, заснований на аналізі вмісту, так як він покладається не на оцінки, а на атрибути, що допомагає включати нові предмети в рекомендації для користувачів. Однак проблему з наданням рекомендації для нового користувача вирішити складніше;

- масштабованість: зі збільшенням кількості користувачів в системі, з'являється проблема масштабованості. Чим більше користувачів та різних статей для них тим потужніша система повинна бути на сервері;

- синонімія: так називається тенденція схожих і однакових предметів мати різні імена. Більшість рекомендаційних систем не здатні виявити ці приховані зв'язки і тому відносяться до цих предметів як до різних;

– білі ворони. До «білих ворон» відносяться користувачі, чия думка постійно не збігається з більшістю інших. Через унікальність смаку їм неможливо щось рекомендувати. Однак, такі люди мають проблеми з отриманням рекомендацій і в реальному житті, тому даною проблемою можна знехтувати [18].

3.5 Висновки за розділом 3

В ході проєктування архітектури системи було проаналізовано основні підходи до нормалізації даних а також до оцінки якості створеної рекомендаційної системи. Окрім того був розроблений більш оптимізований алгоритм k найближчих сусідів і пояснено доцільність використання в рекомендаційній системі косинусної подібності.

4 РОЗРОБКА ПРОГРАМИ

4.1 Опис основних етапів розробки

Рекомендаційна система – підклас системи фільтрації інформації, яка будує рейтинговий перелік об'єктів, яким користувач може надати перевагу. Для цього використовується інформація про інтереси користувача.

Для правильної роботи системи потрібно розбити її на 2 підсистеми. Система, яка збирає дані від користувача, тобто навчає алгоритм і підсистема яка буде використовувати зібрані дані для виведення персональних рекомендацій [19].

4.2 Процес навчання системи

4.2.1 Опис класу «Learning»

Даний клас призначений для навчання системи. Для навчання системи кожен користувач який буде переглядати статтю на інформаційному порталі новин повинен бути внесений в базу даних.

Цей клас має такі параметри:

- текст статті;
- унікальний ідентифікатор користувача;
- ідентифікатор категорії статті.

В якості системи керування базами даних можна використовувати MySQL.

Переваги MySQL:

- простота у використанні;
- великий функціонал;
- масштабованість;
- швидкість.

Недоліки MySQL:

- недостатня надійність;
- низька швидкість розробки.

Всі ці фактори роблять MySQL хорошим варіантом для навчання алгоритму а також зберігання даних.

Для навчання в системі керування базами даних буде створена таблиця «Recommendations». Дана таблиця буде містити такі поля:

- унікальний ідентифікатор користувача який прочитав статтю;
- унікальний ідентифікатор статті;
- категорія статті;
- текст статті у виді TF-IDF.

Створення таблиці для фази навчання алгоритму зображено на рисунку 4.1.

The screenshot shows the MySQL database management interface. At the top, there is a toolbar with icons for 'Переглянути' (View), 'Структура' (Structure), 'SQL', 'Пошук' (Search), 'Вставити' (Insert), 'Експорт' (Export), 'Імпорт' (Import), 'Привілеї' (Privileges), and 'Операції' (Operations). Below the toolbar, the 'Структура' (Structure) tab is selected. The table name is 'Recommendations'. The table has four columns: 'id' (INT), 'user_id' (INT), 'article_id' (INT), and 'Text' (INT). The 'Додати' (Add) button is set to 1, and the 'стовець(і)' (column(s)) button is visible. Below the column list, there are sections for 'Коментарі до таблиці:' (Comments for table:), 'Порівняння:' (Comparison:), 'Тип таблиць:' (Table type:), and 'Визначення розділів:' (Define partitions:). The 'Тип таблиць:' section shows 'InnoDB' as the selected engine.

Рисунок 4.1 – Створення таблиці в системі MySQL

Після створення таблиці для запису алгоритм, який буде перетворювати дані класу в створену таблицю при кожному переході користувача на певну статтю. Для цього потрібно при завантаженні сторінки зі статтею створити метод, який буде створювати новий запис в базі даних. Однак перед тим як робити запис потрібно обробити виключення.

4.2.2 Захист від дублювання інформації

Оскільки один користувач може прочитати одну статтю декілька разів, система може внести запис про прочитання користувачем однієї статті декілька разів. Це може порушити роботу системи рекомендацій. Для того щоб уникнути такої ситуації перед занесенням в базу даних алгоритм робитиме запит на вибірку з такими даними. Запис у базу даних буде здійснений тільки якщо попередній запит повернув нульовий результат.

4.2.3 Захист від запису неавторизованих користувачів

Оскільки рекомендаційна система буде працювати лише для авторизованих користувачів, то необхідно перед новим записом перевірити наявність авторизованого користувача, для цього в мові програмування PHP існує глобальний масив «Session» та «Cookie». Алгоритм має перевіряти за допомогою них чи авторизований користувач.

4.3 Процес використання отриманих знань

Під час роботи сесії для створення персональних рекомендацій система буде аналізувати переваги користувача за допомогою методу k найближчих сусідів. Для мови програмування PHP даний метод реалізований в бібліотеці PHP Machine Learning.

Для формування рекомендацій використовується клас «Recommend». Даний клас призначений для виведення записів з бази даних про перегляди одного користувача за останні 5 днів, а також пошук подібних до них статей за допомогою методу k найближчих сусідів.

4.3.1 Отримання даних

Для отримання даних про всі перегляди одного користувача використовується метод `getData()`.

Даний метод робить запит мовою SQL до бази даних та формує масив текстів для конкретного користувача.

4.3.2 Перетворення даних

Для перетворення тексту в формат TF-IDF використовується метод `transform()`. Цей метод використовується для того, щоб виділити ключові слова в останніх статтях, які переглянув цей користувач для подальшої роботи за допомогою методу пошуку k найближчих сусідів.

Окрім цього після перетворення тексту за допомогою метода TF-IDF система вираховує косинусну подібність для покращення роботи алгоритму. Для реалізації використовується метод `cosin()`.

4.3.3 Використання методу найближчих сусідів

Далі для створення персональних рекомендацій використовується алгоритм k найближчих сусідів. Для реалізації даного алгоритму використовується клас `KNearestNeighbours`. Цей клас представлений у бібліотеці `Php machine learning`. Даний алгоритм має дві фази:

Фаза навчання реалізована за допомогою метода `train()`. Вона навчає алгоритм для подальшого розпізнавання. В ній використовується масив текстів в форматі TF-IDF, який був отриманий за допомогою методу `getData()` і перетворений за допомогою метода `transform()`.

Фаза розпізнавання реалізована за допомогою метода `predict()`. Цей метод в якості параметру приймає тексти, для яких потрібно знайти

найближчих сусідів. В якості відповідей буде прийматися унікальний ідентифікатор статті, яким вона пронумерована в базі даних.

Для виведення рекомендацій користувачу використовується метод `show()`. Він приймає всі ідентифікатори статей і робить запит за допомогою мови SQL у базу даних. Результати будуть виведені в спеціальне вікно на веб-порталі новин

4.4 Порівняння оригінального алгоритму і створеного

Для перевірки роботи алгоритму було протестовано його точність роботи, для створення персональних рекомендацій за різними темами. У тестуванні використовувалась математична метрика MAE. Рисунок 4.2 показує точність роботи алгоритму для новин з категорією «Політика».

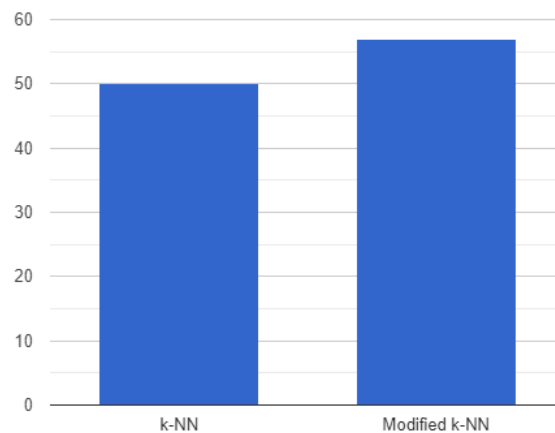


Рисунок 4.2 – Порівняння алгоритмів за категорією «Політика»

Як можна побачити на рисунку модифікований алгоритм точніший за оригінальний на 7% в категорії «Політика».

Аналогічно було проведено порівняння алгоритмів в категорії «Спорт». Результати порівняння зображені на рисунку 4.3.

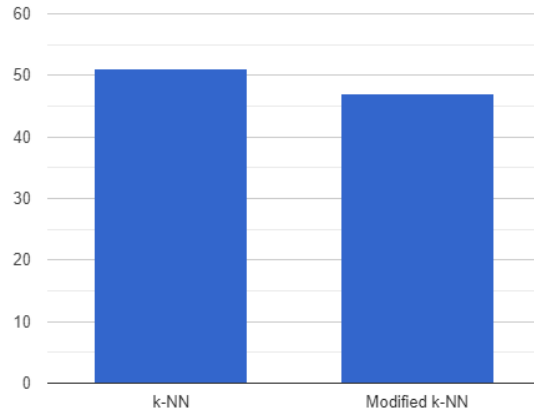


Рисунок 4.3 – Порівняння алгоритмів за категорією «Спорт»

Як можна побачити на рисунку, в категорії «Спорт» модифікований алгоритм виявився менш ефективним ніж оригінальний на 4%. Таким самим чином було протестовано систему в категорії «Тварини». Результати показано на рисунку 4.4.

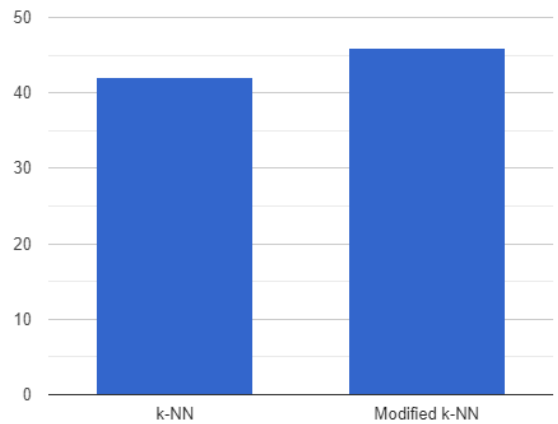


Рисунок 4.4 – Порівняння алгоритмів за категорією «Тварини»

За даною категорією модифікований алгоритм показав більш точні результати на 3%.

Таким чином можна стверджувати що використання алгоритму косинусної подібності збільшує точність роботи рекомендаційної системи в середньому на 4%.

4.5 Висновки за розділом 4

Програмний продукт було створено на основі мови програмування RНР а також бібліотеки RНР Machine Learning. В якості алгоритма класифікації використовується алгоритм пошуку k найближчих сусідів. Завдяки ньому і формуються персональні рекомендації. Окрім цього була покращена точність роботи алгоритму на 4% завдяки використанню математичного метода косинусної подібності.

5 ЕКСПЛУАТАЦІЯ ТА ТЕСТУВАННЯ ПРОГРАМИ

5.1 Керівництво програміста

5.1.1 Призначення програми

Рекомендаційна система багатокористувацького веб-порталу новин призначена для створення рекомендацій користувачам на основі їхніх інтересів.

Працювати з веб-проектом можливо з будь-якого комп'ютера, який має доступ в інтернет.

5.1.2 Функції, виконувані програмою

Функцією програми є створення персональних рекомендацій.

5.1.3 Умови, необхідні для застосування програми

Вимоги до складу периферійних приладів включають системний блок з усіма його складовими, клавіатуру та монітор.

Вимоги до параметрів периферійних приладів:

- IBM PC сумісним з процесором 80386 і вище;
- ОС Windows XP/Vista/7/8/10;
- ОЗУ більше 32 Мбайт;
- 64 МБ відеопам'яті і вище.

Для стабільної роботи програми необхідна актуальна версія браузера.

В перелік задач, виконуваних програмістом, повинні входити:

- задача підтримки працездатності технічних засобів;
- задача підтримки працездатності системних програмних засобів – серверу;
- задача підтримки працездатності програми.

5.1.4 Опис основних характеристик програми

Програма працює лише під час роботи веб сайту.

5.1.5 Опис основних особливостей програми

Програма має користувацький інтерфейс і не надає користувачу можливості налаштування і зміни своїх параметрів.

Для запуску програми потрібно виконати декілька кроків:

- упевнитися у відповідності характеристик ЕОМ, на яку встановлюється програма, до мінімальних системних вимог програми;
- запустити веб-браузер;
- ввести в адресний рядок назву веб-сайту.

5.1.6 Вхідні дані

Вхідними даними до програми є дані, які користувач вносить не напряму а через перегляд статей .

5.1.7 Вихідні дані

Вихідними даними програми є створені для кожного користувача персональні рекомендації.

5.1.8 Повідомлення

При будь яких змінах в базі даних виводиться повідомлення наведене у рисунку 5.1.

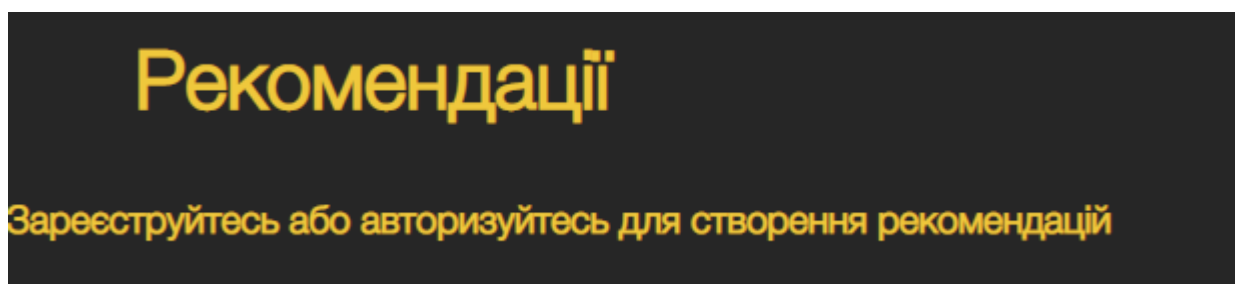


Рисунок 5.1 – Повідомлення користувачу

5.2 Керівництво користувача

5.2.1 Призначення та функції програми

Програмний засіб призначений для людей, які прагнуть ознайомитись з актуальними новинами.

Використовувати програму може абсолютно будь-яка людина з мінімальними навичками володіння комп'ютером.

Програма має такі функції:

- запам'ятовування останніх переглянутих статей користувачем;
- пошук подібних статей;
- виведення персональних рекомендацій.

5.2.2 Умови виконання програми

Далі представлені основні вимоги до системи, необхідні для стабільної роботи і функціональності. В характеристиках вказані мінімальні параметри.

Системні вимоги:

- процесор з тактовою частотою 1.2 GHz;
- 256 Мб та більше RAM;
- вільне місце на ПЗУ: 100 Мб;
- архітектура системи з розрядністю 32- або 64-біт;
- операційна система: Windows (з версії XP та вище);

- стандартні пристрої вводу-виводу (клавіатура, миша, монітор);
- веб-браузер з підтримкою HTML 5.

5.2.3 Виконання програми

Для запуску програми необхідно виконати наступні кроки:

- запустити веб-браузер;
- в адресній стрічці ввести посилання на сайт.

Подальша робота з програмою здійснюється за допомогою миші та клавіатури через інтерфейс програми

5.3 Керівництво системного програміста

5.3.1 Призначення та функції програми

Програмний засіб призначений для людей, які прагнуть прочитати новини, які цікавлять саме їх.

Використовувати програму може абсолютно будь-яка людина з мінімальними навичками володіння комп'ютером. Програма має такі функції:

- навчання системи за допомогою користувацьких переглядів;
- підбір подібних статей;
- формування персональних рекомендацій;

5.3.2 Структура програми

Структура програми наведена на рисунку 5.2 та 5.3.

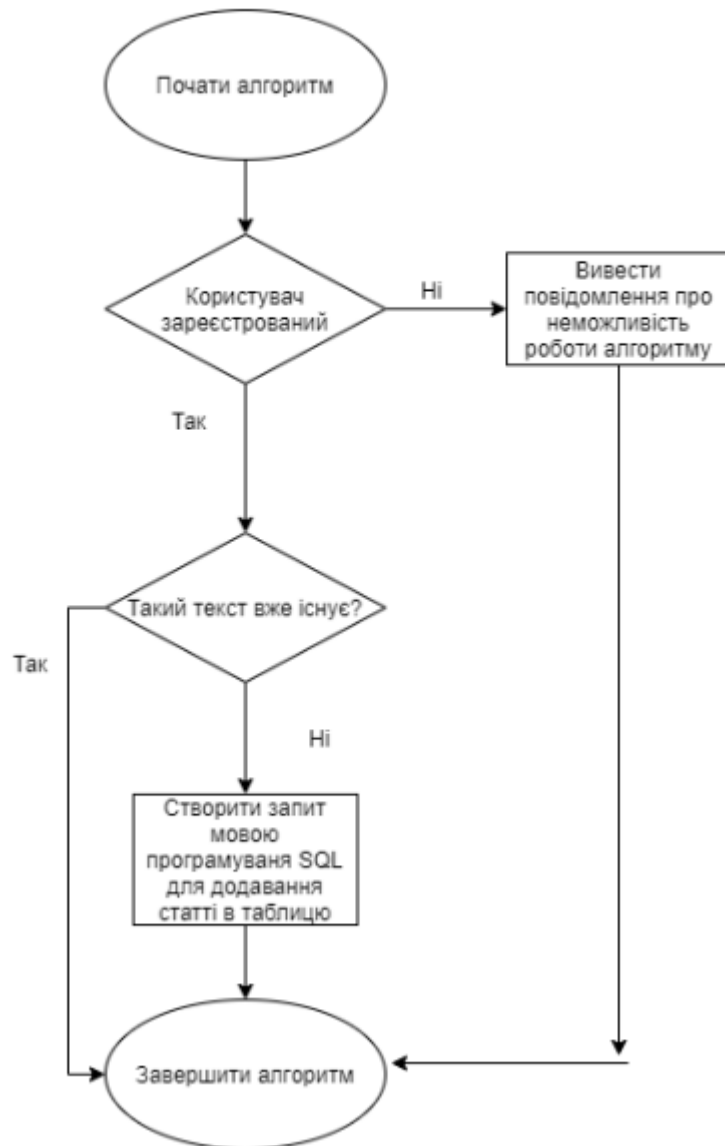


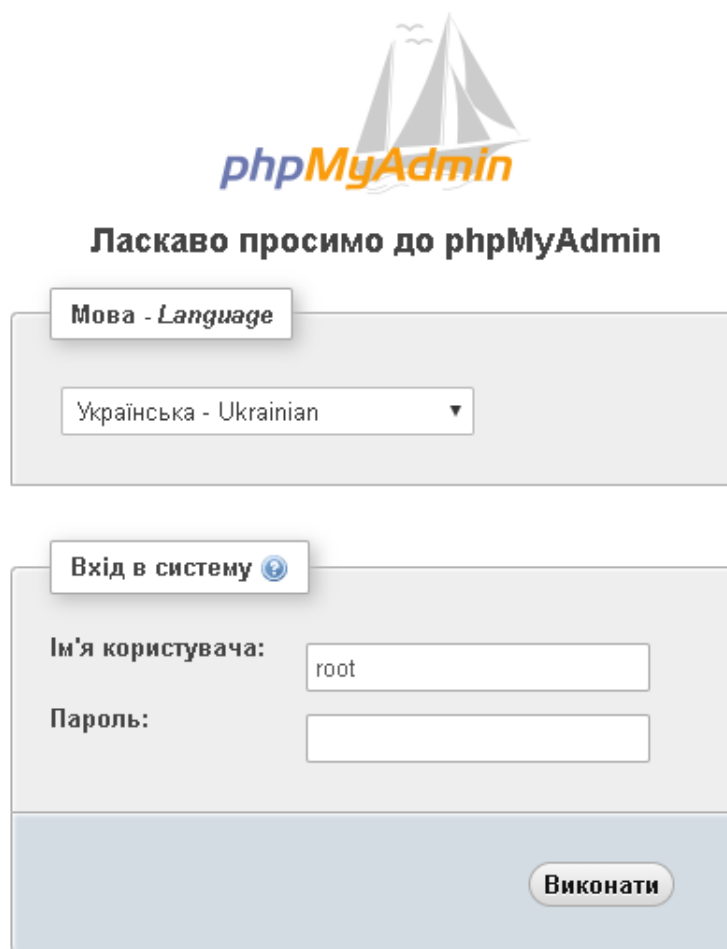
Рисунок 5.2 – Структура програми під час фази навчання



Рисунок 5.3 – Структура програми під час фази розпізнавання

5.3.3 Налаштування програми

Для коректного налаштування програми системному програмісту потрібно увійти в обліковий запис PhpMyAdmin. Інтерфейс входу до PhpMyAdmin зображено на рисунку 5.4.



phpMyAdmin

Ласкаво просимо до phpMyAdmin

Мова - *Language*

Українська - Ukrainian ▼

Вхід в систему ⓘ

Ім'я користувача: root

Пароль:

Виконати

Рисунок 5.4 – Вхід до PhpMyAdmin

Після входу до облікового запису потрібно створити нову базу даних з назвою «Recommend» (рис. 5.5).

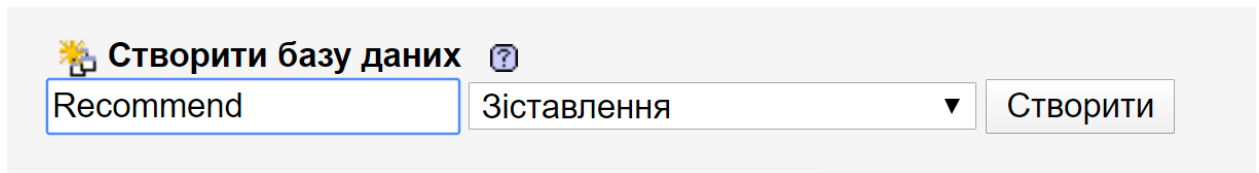


Рисунок 5.5 – Створення бази даних

Для роботи програми необхідно імпортувати базу даних під назвою Recommend в PhpMyAdmin за допомогою команди «Імпорт»

Результатом налаштувань повинна стати повністю функціональна база даних.

5.3.4 Перевірка

Гарантом правильності працездатності системи є відсутність помилок при роботі веб-проєкту.

5.3.5 Додаткові можливості

Даний програмний продукт не обладнаний додатковими можливостями, окрім тих, які були заявлені раніше.

5.4 Тестування програми

Для тестування системи і перевірки її працездатності було створено 5 тестових випадків на багатокористувацькому веб-порталі новин. Далі наведений опис кожного тестового випадку.

5.4.1 Тестовий випадок №1

Цей тестовий випадок імітує людину, яка цікавиться новинами політики а також її вплив на життя громадян. Для тестування було обрано новини, в яких є такі ключові слова «Трамп», «Зеленський» та «Вибори». Результати тестування зображені на рисунку 5.6.

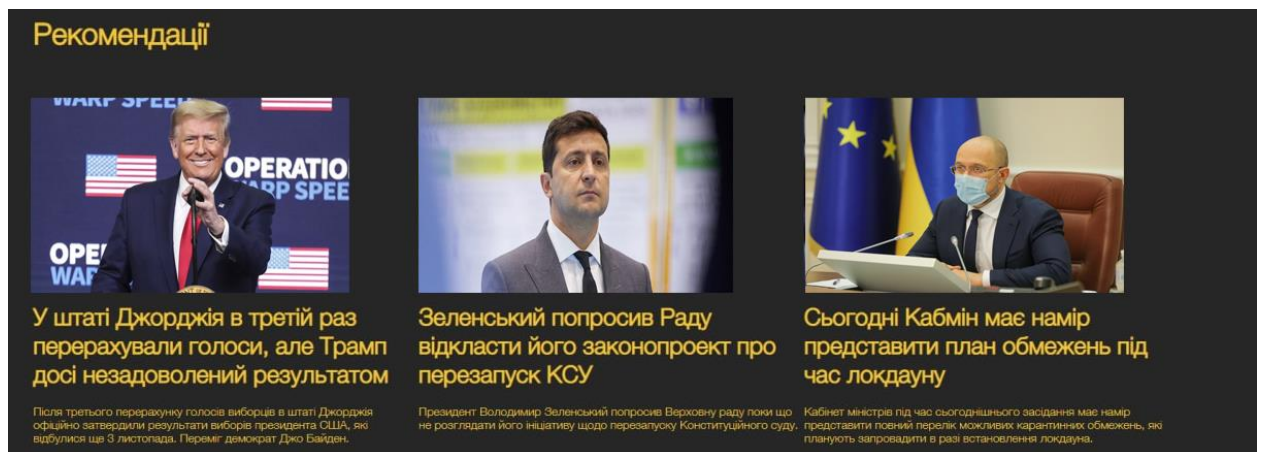


Рисунок 5.6 – Тестовий випадок №1

Як видно на зображенні, рекомендовані новини співпадають з інтересами такого користувача.

5.4.2 Тестовий випадок №2

Цей тестовий випадок імітує людину, яка цікавиться новинами про спорт, до яких належать такі категорії, як «Футбол», «Бокс» та інші види спорту. Результат тестування і сформовані персональні рекомендації зображені на рисунку 5.7.



Рисунок 5.7 – Тестовий випадок №2

Як можна побачити на рисунку, результат не повністю відповідає інтересам користувача. В створених для цього користувача персональних рекомендаціях є новина за темою «Політика». Причиною такого може бути те, що в новинах за темою «Спорт» часто фігурують слова «Програв» або «Здобув перемогу». Такі слова використовуються і в інших галузях, тому вони також можуть потрапити в персональні рекомендації навіть якщо користувач не цікавився такою темою напряду. Цей тест свідчить про недосконалість алгоритму k найближчих сусідів.

5.4.3 Тестовий випадок №3

Цей тестовий випадок описує користувача який не цікавиться якоюсь конкретною темою в новинах. Він може зацікавитись новинами на будь яку тему. Для проведення тестування новини, які було додано у вибірку були випадкові тож немає конкретних ключових слів за якими склалися б персональні рекомендації. Результат тесту показаний на рисунку 5.8.

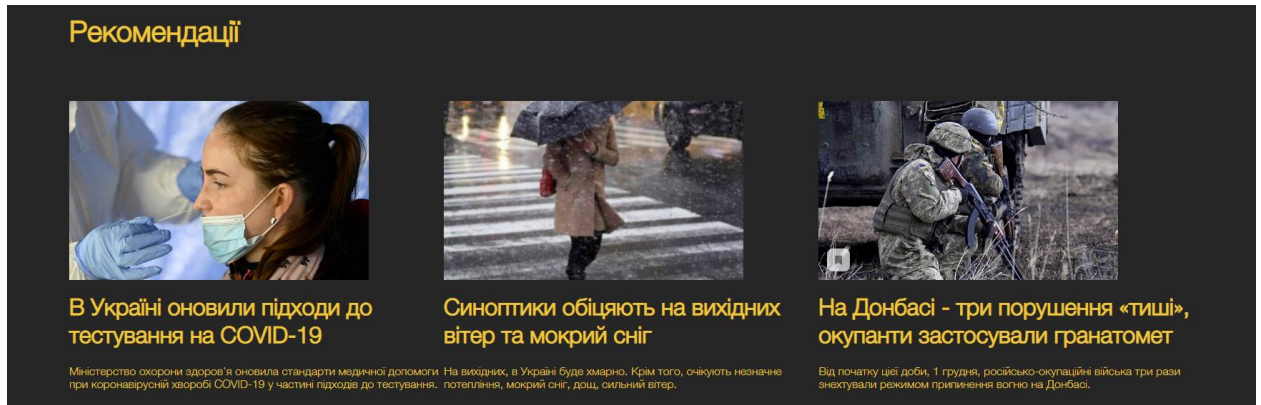


Рисунок 5.8 – Результат тестового випадку № 3

Як можна побачити на рисунку 5.8 у разі перегляду користувачем всіх новин з різною тематикою алгоритм створення персональних рекомендацій також працює, однак ефективність і точність роботи алгоритму знижується оскільки важко зрозуміти які теми будуть цікаві саме цьому користувачу.

5.4.4 Тестовий випадок №4

Цей тестовий випадок імітує ситуацію, коли користувач тільки створив обліковий запис на веб порталі новин проте досі не переглядав жодної статті. Метою даного тестового випадку є перевірка роботи алгоритму при відсутності вхідних даних. Позитивним результатом буде повідомлення про неможливість створення персональних рекомендацій через відсутність даних про інтереси користувача. Результат тестового випадку № 4 зображений на рисунку 5.9.

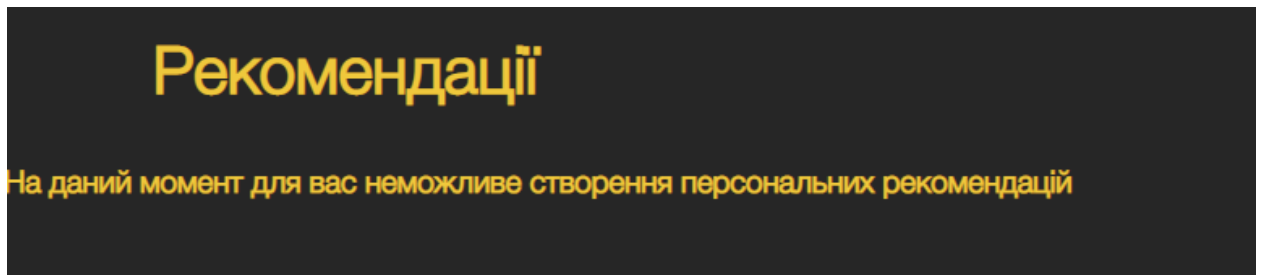


Рисунок 5.9 – Тестовий випадок № 4

Як видно на рисунку, при нестачі даних для системи алгоритм не може працювати, оскільки сформувати персональні рекомендації на основі відсутніх даних неможливо. Саме тому замість новин виводиться таке повідомлення.

5.4.5 Тестовий випадок №5

Цей тестовий випадок імітує користувача, який ще не створив обліковий запис на веб-порталі новин. При такому сценарії алгоритм не повинен працювати взагалі. Результат робот зображений на рисунку 5.10.

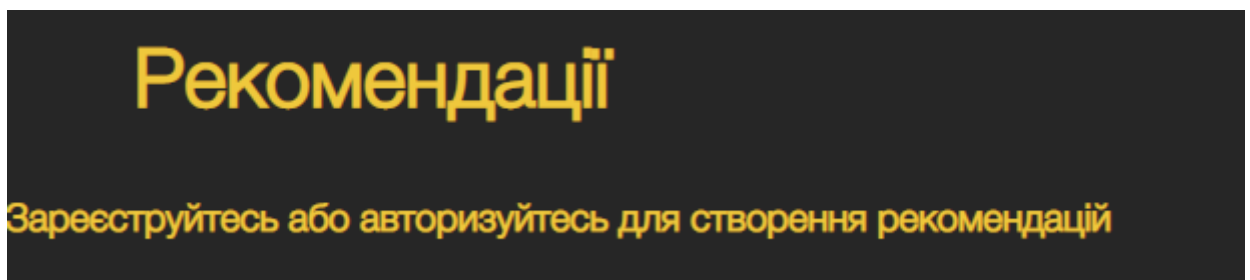


Рисунок 5.10 – Результат тестового випадку №5

Система не може працювати, якщо в ній немає активного користувача, оскільки саме на основі користувача та його унікального ідентифікатора і працює алгоритм створення персональних рекомендацій. Саме тому в даному тестовому випадку було виведено це повідомлення.

5.5 Висновки за розділом 5

В даному розділі було описано керівництво користувача, системного програміста та програміста. Окрім цього було проведено тестування різних сценаріїв використання розробленого програмного продукту. Результати тестів свідчать про те, що створений алгоритм працює, але все одно має певні недоліки.

6 ОРГАНІЗАЦІЙНО-ЕКОНОМІЧНА ЧАСТИНА

6.1 Планування розробки системи

Приводиться економічне обґрунтування програмної реалізації системи, яка буде виводити персональні рекомендації.

Цей програмний продукт має зробити зручнішим перегляд новин для користувачів, дозволить читати різні статті в зручному вигляді. Система буде навчатися, завдяки даним від користувачів та використовувати ці дані

У даний час ринок подібного програмного забезпечення має схожі аналоги від кількох відомих брендів [20].

Очікується отримання прибутку за рахунок реклами на веб сайті.

6.2 Планування розробки

Весь комплекс розробки можна розділити на етапи. Для кожного етапу вказується трудомісткість, кількість виконавців і тривалість робіт. У розробці приймають участь програміст та консультант. Розробка була розпочата 2 вересня і повинна бути виконана до 5 листопада 2020 року. Етапи розробки продукту наведені в таблиці 6.1.

Таблиця 6.1 – Етапи розробки продукту

Назва роботи	Трудомісткість		Виконавці	Тривалість, днів
	люд.- дні	% до підсумк у		
1. Аналіз предметної області	8	12,5	Програміст, консультант	4

Продовження таблиці 6.1

Назва роботи	Трудовісткість		Виконавці	Тривалість, днів
	люд.- дні	% до підсумк у		
2. Визначення вимог до програмного продукту	4	6,25	Програміст, консультант	2
3. Аналіз принципів побудови експертних систем	9	14,06	Програміст	9
4. Дослідження методів вирішення задач	8	12,5	Програміст	8
5. Побудова моделі системи	5	7,81	Програміст	5
6. Створення програмного коду	22	34,38	Програміст	22
7. Тестування і налагодження програмного продукту	4	6,25	Програміст	4
8. Складання програмної документації	4	6,25	Програміст, консультант	2
Разом	64	100		56

6.3 Визначення витрат на розробку

Для визначення витрат на розробку складаються наступні елементи:

- основна заробітна плата;
- додаткова заробітна плата;
- відрахування на єдиний соціальний внесок;

- матеріали і комплектуючі вироби;
- витрати на спеціальне обладнання;
- накладні витрати [21].

Для розрахунку заробітної плати маємо на увазі, що за час, витрачений на розробку системи кількість робочих днів дорівнює 56.

Тривалість робіт визначають за формулою:

$$T_{\text{ц}} = \frac{Q}{R}, \quad (6.1)$$

де $T_{\text{ц}}$ – кількість днів;

Q – трудомісткість;

R – кількість робітників (люд.).

6.3.1 Розрахунок основної заробітної плати

Проводиться розрахунок основної заробітної плати, який наведено у таблиці 6.2.

Таблиця 6.2 – Основна заробітна плата

Посада виконавця	Кількість, чол.	Місячний оклад, грн	Кількість місяців	Сума ЗП, грн
Програміст	1	8900	2,5	17800
Консультант	1	11300	0,5	5 650
Підсумок	2			23450

6.3.2 Розрахунок додаткової заробітної платні

Додаткова заробітна плата дорівнює 10% від основної заробітної плати працівника і розраховується за формулою:

$$ЗП_{\text{дод}} = ЗП_{\text{осн}} \cdot 0,1. \quad (6.2)$$

Після підстановки величини основної заробітної плати в дану формулу, отримується:

$$ЗП_{\text{дод}} = 23450 \cdot 0,1 = 2345 \text{ грн.}$$

6.3.3 Відрахування на єдиний соціальний внесок

Відрахування на єдиний соціальний внесок обчислюються як 22% і від основної та додаткової заробітної плати. Вони розраховуються за формулою 6.3:

$$ВІД = (ЗП_{\text{осн}} + ЗП_{\text{дод}}) \cdot 0,22, \quad (6.3)$$

де $ЗП_{\text{осн}}$ – основна заробітна плата (грн.),

$ЗП_{\text{дод}}$ – додаткова заробітна плата (грн.).

$$ВІД = (23450 + 2345) \cdot 0,22 = 5674,9 \text{ грн.}$$

Таким чином додаткові соціальні відрахування дорівнюють 5674,9 грн.

6.3.4 Визначення витрат на матеріали

Витрати на матеріали розраховуються за формулою:

$$M = \sum_{i=1}^n (Ц_i N_i (1 + K_{\text{т.з.}}) - Ц_{io} N_{io}), \quad (6.4)$$

де M – витрати на матеріали, покупні напівфабрикати та комплектуючі (грн.);

$K_{т.з.}$ – коефіцієнт, що враховує транспортно–заготівельні витрати;

C_i – ціна і–го найменування чи комплектуючого (грн.);

N_i – потреба в і–му матеріалі, напівфабрикаті, комплектуючих;

$C_{іо}$ – ціна зворотних відходів і–го найменування матеріалу (грн.);

$N_{іо}$ – кількість зворотних відходів і–го найменування;

n – кількість найменувань матеріалів.

Витрати на матеріали для розробки наведені в таблиці 6.3.

Таблиця 6.3 – Витрати на матеріали

Найменування, вид, тип, марка	Одиниця виміру	Ціна за одиницю вим., грн.	Кількість од. вим.	Сума витрат, грн.
Канцелярія	Набір	250	1	250
Картридж для принтера	шт.	550	1	400
Папір формату А4 для принтера	Листів	0,15	300	85
USB Flash накопичувач об'ємом 16Gb	шт.	170	1	170
Всього				905

$$M = 1,05 \cdot 905 = 950,25 \text{ грн.}$$

З урахуванням можливих транспортних витрат (5% від вартості матеріалів), сумарна вартість матеріалів становить 950,25 грн.

6.3.5 Витрати на ПК

Дана стаття включає в себе витрати на придбання, транспортування, монтаж а таке налагодження нестандартного обладнання. Окрім цього необхідно розрахувати витрати на оплату машинного часу ЕОМ (амортизаційні витрати на обслуговування та вартість спожитої електроенергії) [22].

Розрахуємо витрати на обслуговування ЕОМ. Амортизаційні відрахування розраховуються за формулою:

$$A = \Phi_B \frac{H_a}{100}, \quad (6.5)$$

де Φ_B – балансова вартість обчислювальної техніки (грн.);

H_a – норма амортизаційних відрахувань на повне відновлення обчислювальної техніки (%).

Підставивши у формулу 6.5 значення балансової вартості використовуваних ЕОМ HP 250 G7, маніпулятору “миша” та принтеру HP LaserJet Pro M15a (разом 30000 грн.) отримаємо :

$$A = 30000 \cdot 0,25 = 7500 \text{ грн.}$$

Таким чином амортизаційні відрахування становлять 7500 грн.

Для розрахунку вартості експлуатації обладнання можна використати формулу:

$$C_{\Sigma} = N_H \Phi_{\text{еф}} K_{\text{зч}} K_{\text{зп}} C_e, \quad (6.6)$$

де N_H – номінальна потужність ЕОМ, кВт;

$\Phi_{\text{еф}}$ – річний ефективний фонд часу роботи ЕОМ;

$K_{\text{зм}}$ – середній коефіцієнт завантаження за часом;

$K_{\text{зп}}$ – коефіцієнт завантаження за потужністю;

C_e – ціна одного кВт*год. Електроенергії, грн./(кВт*год.).

Річний фонд часу роботи ЕОМ (персонального комп'ютера працівника) становить 1800 годин. Середні коефіцієнти завантаження за часом та за потужністю становлять 0.6 та 0.9.

$$C_{\text{з1}} = 0,2 \cdot 1800 \cdot 0,8 \cdot 0,6 \cdot 2,11 = 364,6 \text{ грн.}$$

Заробітна плата обслуговуючого персоналу розраховується за наступною формулою:

$$ЗП_{\text{обсл}} = \Phi ЗР_{\text{р}} (1 + K_{\text{від}}) \frac{t_{\text{обсл}}}{\Phi_{\text{еф.обсл}}}, \quad (6.7)$$

де $\Phi ЗР_{\text{р}}$ – річний фонд заробітної плати (основної та додаткової) обслуговуючих робіт (грн.);

$K_{\text{від}}$ – коефіцієнт відрахувань на соціальне страхування і до інших фондів (%);

$t_{\text{обсл}}$ – час протягом року, необхідний на технічне обслуговування ЕОМ, (ч/рік);

$\Phi_{\text{еф. обсл.}}$ – річний ефективний фонд часу обслуговуючого персоналу.

Місячна заробітна плата обслуговуючого персоналу становить 5300 грн., а річний фонд заробітної плати становить 48600 грн. Річний ефективний фонд робочого часу обслуговуючого ПК робітника становить 1800 годин/рік. В той час як на обслуговування одного ПК витрачається по 3 години на місяць, що на рік становить 36 годин. З таких даних можна зробити висновок, що:

$$ЗП_{\text{обсл}} = 48600 \cdot (1 + 0.22) \cdot \frac{36}{1800} = 1185,84 \text{ грн.}$$

Стаття «Поточний ремонт обладнання» дорівнює 3% від балансової вартості обладнання. В даному випадку ця стаття становить 900 грн.

Стаття «Інші витрати» приймається рівною 5% від суми всіх попередніх статей витрат на утримання та експлуатацію устаткування. Сума всіх попередніх статей дорівнює 9950 грн.,

Таким чином інші витрати становлять 497,52 грн.

Отже далі наведений кошторис витрат на експлуатацію ПК (таблиця 6.4).

Таблиця 6.4 – Кошторис витрат

Найменування статей витрат	Сума, грн.
Амортизація обладнання	7500
Експлуатація ПК (окрім витрат на поточний ремонт)	364,6
Заробітна плата основна та додаткова обслуговуючих робітників з відрахуваннями на соціальні заходи	1185,84
Поточний ремонт обладнання	900
Інші витрати	497,52
Разом	10447,96

6.3.6 Витрати на розробку

Витрати на оплату машинного часу ЕОМ для написання й налагодження програми вираховується за формулою 6.8:

$$C_{\text{мс}} = P_{\text{екс}} t_{\text{мч}}, \quad (6.8)$$

де $C_{\text{мс}}$ – витрати на сплату машинного часу (грн.);

$P_{\text{екс}}$ – експлуатаційні витрати на один час машинного часу ЕОМ;

$t_{\text{мо}}$ – машинний час ЕОМ для написання та налагодження програмного засобу.

$$P_{\text{екс}} = \frac{10447.96}{1800} = 5,80 \text{ грн./машино-час.}$$

ЕОМ може експлуатуватися 55 днів в одну зміну, що становить у сумі 440 годин. Отже витрати на сплату машинного часу можна розрахувати як:

$$C_{\text{мс}} = 5,80 \cdot 440 = 2553 \text{ грн.}$$

6.3.7 Витрати на сервер

У якості сервера використовується HP ProLiant DL 380 G7. Якщо підставити у формулу 6.5 значення балансової вартості сервера:

$$A = 11000 \cdot 0,25 = 2750 \text{ грн.}$$

Таким чином амортизаційні відрахування становлять 2750 грн.

За формулою (6.6) річний фонд часу роботи ЕОМ (сервера) становить 8760 годин. Середні коефіцієнти завантаження за часом та за потужністю становлять 0.6 та 0.9.

$$C_{\Sigma 2} = 0,2 \cdot 8760 \cdot 0,8 \cdot 0,6 \cdot 2,11 = 1\,774,4 \text{ грн.}$$

Місячна заробітна плата обслуговуючого персоналу становить 5300 грн., а річний фонд заробітної плати становить 63600 грн. Річний ефективний фонд робочого часу обслуговуючого ПК робітника становить 1800 годин/рік. На обслуговування одного ПК витрачається по 2 години на місяць, що на рік становить 24 години. З чого виходить, що:

$$ЗП_{\text{обсл}} = 63600 \cdot (1 + 0.22) \cdot \frac{24}{1800} = 1034 \text{ грн.}$$

Стаття «Поточний ремонт обладнання» приймається рівною 3% від балансової вартості обладнання і складає 330.

Стаття «Інші витрати» приймається рівною п'яти відсоткам від суми всіх попередніх статей витрат на утримання та експлуатацію

устаткування [23]. Сума всіх попередніх статей дорівнює 7 074 грн., 5% від суми становлять 354 грн.

Далі у таблиці 6.5 наведено кошторис витрат на утримання та експлуатацію серверу на рік.

Таблиця 6.5 – Кошторис витрат

Найменування статей витрат	Сума, грн.
Амортизація обладнання	2750
Експлуатація (окрім витрат на поточний ремонт)	1774,4
Заробітна плата основна та додаткова обслуговуючих робітників з відрахуваннями на соціальні заходи	1034
Поточний ремонт обладнання	480
Інші витрати	301,92
Разом	6340,32

Для розробки системи також існує потреба в експлуатації сервера протягом 56 діб. Таким чином витрати на сплату машинного часу становлять:

$$C_{mc} = \frac{56}{365} \cdot 6340 = 972,7 \text{ грн.}$$

6.3.8 Накладні витрати

Накладними витратами називаються витрати на загальне управління, загальногосподарські потреби, що включає заробітну плату апарату управління, канцелярські витрати і т.д., також утримання та експлуатацію будівель [24]. Накладні витрати включають у вартість розробки програми непрямым шляхом – у відсотках до основної заробітної плати розробників. В даному випадку накладні витрати становлять 40% до основної заробітної плати розробників, що складає 9380 грн.

Результати визначення витрат на розробку програми приведені в таблиці 6.6.

Таблиця 6.6 – Вартість розробки програми

Стаття витрат	Сума (грн.)	Питома вага до підсумку (%)
Основна заробітна плата	23450	51,96
Додаткова заробітна плата	2345	5,2
ССВ	5674,9	12,58
Вартість матеріалів	905,25	2,01
Витрати на ПК	2553	5,66
Витрати на сервер	1034	1,82
Накладні витрати	9380	20,79
Разом	45342,15	100,00

6.3.8 Розрахунок експлуатаційних витрат

Далі наведений розрахунок Витрат, пов'язаних з експлуатацією програми протягом року. Процес показаний на формулі 6.9:

$$I = P_{\text{екс.кмс}} \cdot T_{\text{кмс}} + \frac{S_{pn}}{n \cdot T_c}, \quad (6.9)$$

де $P_{\text{екс.кмс}}$ – експлуатаційні витрати, що припадають на 1 ч машинного часу КТС споживача програмного продукту, грн / машино–год;

$T_{\text{кмс}}$ – сумарний машинний час КТС протягом року, потрібний даному споживачеві програми для задач, що вирішуються за її допомогою, машино–год;

S_{pn} – сумарні витрати на розробку програми, грн.;

n – кількість споживачів даної програми;

T_c – термін служби даної програми до її морального зносу, років.

$$И = 6340 + \frac{45342}{1 \cdot 7} = 12817 \text{ грн.}$$

6.3.9 Розрахунок витрат на адміністратора

Місячна заробітна плата адміністратора становить 6300 грн., а річний фонд заробітної плати становить:

$$ЗП_{\text{адм}} = 6300 \cdot 12 \cdot 1,22 = 92232 \text{ грн.}$$

6.4 Розрахунок капітальних вкладень

Обсяг капітальних вкладень споживача програми K , пов'язаних з її розробкою і впровадженням, визначають за формулою 6.10:

$$K = K_{nn} + K_{n\phi}, \quad (6.10)$$

де K_{nn} – виробничі витрати, грн. ;

$K_{n\phi}$ – капітальні вкладення в виробничі фонди, необхідні для впровадження програми, грн.

Виробничі витрати включають витрати, пов'язані з розробкою і налагодженням програмного, математичного та інформаційного забезпечення, проєктуванням системи. Виробничі витрати визначають за формулою 6.11:

$$K_{nn} = \frac{S_{pn}}{n} + K_{np.nn}, \quad (6.11)$$

де S_{pn} – сумарні витрати на розробку або придбання програмного продукту, грн;

n – кількість споживачів даної програми;

$K_{np.m}$ – інші виробничі витрати, грн.

Капітальні вкладення в виробничі фонди включають в себе витрати на придбання, установку, монтаж і налагодження комплексу технічних засобів (КТЗ) з урахуванням можливого використання для вирішення завдань для яких призначена програма, яка розробляється, а також для завдань, що вирішуються за допомогою інших програм. Питомі капітальні вкладення в виробничі фонди, що припадають на частку даного програмного продукту визначають за формулою 6.12:

$$K_{n\phi} = \frac{K_{ктс} \cdot T_{ктс}}{\Phi_{эф.ктс}}, \quad (6.12)$$

де $K_{ктс}$ – витрати на придбання, установку, монтаж і налагодження КТЗ, грн. ;

$T_{ктс}$ – машинний час КТЗ, потрібне даному споживачеві програми для завдань, що вирішуються за допомогою даної програми, машино-год / рік;

$\Phi_{эф.ктс.}$ – річний ефективний фонд часу роботи КТЗ, машино-год / рік.

$$K = \frac{45\,342}{1} + \frac{11\,000 \cdot 8\,760}{8\,760} = 56\,342 \text{ грн.}$$

6.5 Оцінка економічної ефективності

Створення даного застосунку направлено на отримання додаткового прибутку за рахунок реклами яку користувач буде бачити під час перебування на сайті.

За рахунок реклами яка коштує 0,85 грн за перегляд(очікується близько 500 переглядів на день. Тобто 15000 переглядів на місяць або 180000 на рік).

Прибутком будуть вважатися кошти, отримані за рахунок переглядів реклами за винятком заробітної плати адміністратору та експлуатаційних витрат. Адміністратор буде контролювати правильність функціонування системи та за необхідності вноситиме зміни [25].

Прибуток розраховується за формулою (6.13).

$$\Pi = \Pi_r * \text{Пост} - 3\Pi_{\text{адм}} - \text{И} \quad (6.13)$$

$$\Pi = 0,85 \cdot 180000 - 92232 - 12817 = 47951 \text{ грн.}$$

Коефіцієнт ефективності вкладених коштів в програму E і термін їх окупності $T_{\text{ок}}$ розраховують за формулами 6.14 та 6.15:

$$E = \frac{\Pi}{K} \quad (6.14)$$

$$T_{\text{ок}} = \frac{K}{\Pi} \quad (6.15)$$

Таким чином за формулами 6.14 та 6.15:

$$E = 47951 / 56342 = 0,85$$

$$T_{\text{ок}} = 56342 / 47951 = 1,11 \text{ року.}$$

6.6 Висновки за розділом 6

Враховуючи всі чинники і проаналізувавши економічну складову можна зробити висновок про економічну доцільність розробки даного

програмного забезпечення, адже окупність продукту відбудеться за 1,11 року, що менше за допустиме значення часу на окупність продукту, яке дорівнює 6,67 року.

7 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ

7.1 Аналіз потенційних небезпек

Використання програмного додатку, що був розроблений в ході виконання дипломної роботи передбачає застосування обчислювальної техніки (персональної ЕОМ, периферійних пристроїв, тощо). На операторів ЕОМ впливають ряд небезпечних і шкідливих виробничих чинників. Ці чинники визначені в ДСанПіН «Гігієнічна класифікація праці за показниками шкідливості та небезпечності факторів виробничого середовища, важкості та напруженості трудового процесу» №0472-14 [26].

Проаналізувавши всі потенційні небезпеки з якими можуть стикатися працівники, що працюють з електронно-обчислювальними машинами, було продумано низку заходів які можуть забезпечити безпеку працівників при роботі з ЕОМ та їх комплектуючими

Усі потенційні небезпеки, що виникають при роботі в офісі або обчислювальному центрі, при роботі з обчислювальною технікою можна поділити на такі групи:

- фізичні;
- хімічні;
- психофізіологічні;
- санітарно-гігієнічні;
- групи небезпек, що пов'язані з порушенням правил пожежної безпеки;
- небезпеки, пов'язані з невідповідністю персоналу у випадку надзвичайних ситуацій.

До потенційних небезпек фізичного характеру можна віднести механічні травмування, що можуть статися в наслідок невідповідного планування робочого простору виробничого приміщення. Окрім цього

можливі ураження електричним струмом під час роботи за комп'ютером або використанні периферійної комп'ютерної техніки. Це може статися внаслідок порушення правил безпеки під час роботи або використанні несправного обладнання. Оскільки робота за комп'ютером супроводжується значним статичним навантаженням на опорно-руховий апарат а також на органи зору. Професійним захворюванням фахівців, чия праця пов'язана з довготривалим використанням ЕОМ (інженери-програмісти, оператори ЕОМ, тощо) є синдром зап'ястного каналу, або тунельний синдром зап'ястя, який може бути наслідком хронічної травми. Причиною цього захворювання є тривала робота з мишею: постійна напруга і здавлювання приводить до мікротравм, здавлювання нерва прилеглими оточуючими тканинами, через що виникає набряк.

Потенційні небезпеки психофізіологічного характеру тісно пов'язані з незадовільною організацією робочого місця (недостатнє врахуванням вимог ергономіки робочого місця, зокрема нераціональне розташування комп'ютерного та периферійного обладнання, захаращеність робочої зони, столи та стільці можуть бути підібрані неправильно відносно параметрів людини) або виробничого процесу (надлишкові комунікації під час робіт, що потребують зосередженості та значних розумових зусиль, тощо).

Потенційні небезпеки санітарно-гігієнічного характеру можуть складати ті, що виникають в наслідок порушення вимог санітарно-гігієнічних норм до приміщення, в якому безпосередньо здійснюється виробничий процес. До них можуть належати недостатня або підвищена освітленість робочої зони, недостатній повітрообмін, високий рівень шумів внаслідок роботи комп'ютерних складових тощо.

До групи небезпек, що пов'язані з порушенням правил пожежної безпеки, можна віднести потенційні небезпеки, що пов'язані з недотриманням або неналежним дотриманням правил пожежної безпеки при плануванні виробничого приміщення, виробничого процесу або під час використання

пожежонебезпечного обладнання чи устаткування (комп'ютерної техніки, периферійних пристроїв, тощо).

Нарешті, до потенційних небезпек, що пов'язані з проявами наслідків надзвичайних ситуацій, можна віднести ті, що пов'язані з невідповідністю персоналу в умовах надзвичайних ситуацій (наприклад, під час пожежі або стихійного лиха) [27].

7.2 Заходи із забезпечення безпеки

Програмне забезпечення, що розроблюється передбачає роботу у приміщенні команди розробників. Розміри приміщення – $10 \times 4 \times 3$ м.

Загалом площа приміщення складає $10 \times 4 = 40 \text{ м}^2$, а об'єм – $10 \times 4 \times 3 = 120 \text{ м}^3$.

При роботі у такому приміщенні, не зважаючи на інші небезпечні фактори, найбільшу небезпеку для співробітників становить електричний струм. Тому велику роль відіграє забезпечення як організаційних, так і технічних заходів щодо роботи технічного обладнання для забезпечення безпеки розробників.

Для запобігання нещасним випадкам було впроваджено ДНАОП 0.00-4.12-99. «Типове положення про навчання з питань охорони праці». Так з усіма працівниками, прийнятими на роботу, а також у період роботи проводяться навчання, інструктажі з питань охорони праці (вступний, первинний, повторний, позаплановий, цільовий). Окрім цього працівники підприємств при прийнятті на роботу і періодично в процесі роботи, а також вихованці, учні і студенти під час навчально-виховного процесу повинні проходити навчання і перевірку знань згідно з вимогами Типового положення. Забороняється допуск без володіння знаннями та навичками описаними в типовому положенні. Працівники мають вміти надавати першу медичну допомогу потерпілим від нещасних випадків, а також бути усвідомлені про

правила поведження під час виникнення аварій, пожеж та правил електробезпеки.

Для перевірки знань з питань охорони праці працівників існує іспит, який проводиться за екзаменаційними білетами у вигляді усного опитування або шляхом тестування на автоекзаменаторі з наступним усним опитуванням.

Для забезпечення безпечної роботи кожний комп'ютер має власний блок живлення. Для перетворення змінного струму мережі у постійний, комп'ютери використовують блоки живлення (5 В, 12 В), потужністю від 350 до 1000 Вт. Оскільки усі прилади при підключенні до електричної мережі яка заземляється, використовують систему TN, обчислювальні пристрої у приміщенні відносяться до категорії I захисту від ураження електричним струмом. Окрім цього, аби уникнути ураження, корпуси всіх електричних пристроїв виготовляються з неструмопровідних матеріалів, а живлення здійснюється спеціальним кабелем.

Для забезпечення безпечності у приміщенні всі кабелі сховані в спеціальні коробки, розетки розташовані на висоті одного та 0,3 метри від підлоги, електрична мережа вмикається і вимикається за допомогою пускової апаратури. Відповідно до міжнародного стандарту СЕ зазначені міри електробезпеки відповідають вимогам, пропонованим побутовим приладам.

Опір захисного заземлення в приміщенні становить 3 Ом (за стандартом – не більше 4 Ом) згідно з "Правилами улаштування електроустановок". Крім того, у приміщенні також встановлені автоматичні вимикачі, пристрій заземлення контуру згідно з ГОСТ 12.2.007.0–75 «Вироби електротехнічні. Загальні вимоги безпеки».

Для підготовки робочого місця до роботи, слід вжити у вказаному порядку таких технічних заходів:

- здійснити необхідні відключення і вжити заходів, що перешкоджають помилковому або самочинному ввімкненню комутаційної апаратури;

- вивісити заборонні плакати на приводах ручного і на ключах дистанційного керування комутаційною апаратурою;
- перевірити відсутність напруги на струмовідних частинах, які слід заземлити для захисту людей від ураження електричним струмом;
- встановити заземлення (ввімкнути заземлювальні ножі, встановити переносні заземлення);
- обгородити, за необхідності, робочі місця або струмовідні частини, що залишилися під напругою, і вивісити на огороженнях плакати безпеки. Залежно від місцевих умов, струмовідні частини обгородити до чи після їх заземлення.

Під час оперативного обслуговування електроустановки двома і більше працівниками в зміну перелічені в цьому пункті заходи мають виконувати два працівники. В разі одноособового обслуговування їх може виконувати одна особа. Для запобігання ураження людини струмом в розглянутому приміщенні використовується захисне заземлення, окремі розетки по 220, а також ізолююча – лінолеум підлога.

Оскільки робоче приміщення обладнане персональними комп'ютерами з візуальними дисплейними терміналами (моніторами), то повинні дотримуватися ергономічні норми згідно з ДСТУ 7234:2011. «Дизайн і ергономіка».

Рациональне планування робочого місця повинно забезпечувати: найкраще розміщення знарядь і предметів праці, не допускати загального дискомфорту, зменшувати стомлюваність працівника, підвищувати його продуктивність праці. Під час організації робочого місця доцільно виключити або мінімізувати всі фізіологічні неприродні і незручні положення тіла.

Загальні вимоги дизайну та ергономіки» такі:

- забезпечення зручності використання обладнання призначенням;
- опанованість обладнання, обслугованість обладнання, гігієнічність обладнання і середовища робочої зони;

- безпека обладнання;
- композиційна організованість елементів обладнання.

Крім того, конструкція виробничого обладнання повинна забезпечувати фізичні навантаження, які б не перевищували б 1046,7 кДж/год (250 ккал/год). Конструкція виробничого обладнання повинна забезпечувати можливість організації трудового процесу, яка б мінімізувала монотонність праці обмеженням повторень простих трудових дій, їх кількості, а також тривалості безперервного пасивного спостереження за перебігом виробничого процесу або його частини .

Для роботи у нічний період часу багато програм та інтегрованих середовищ розробки надають дизайн «темної» теми, а також фільтр синього кольору що дозволяє зменшити шкідливий вплив надмірної яскравості екранів [28].

7.3 Заходи з забезпечення виробничої санітарії та гігієни праці

Вирішення питань виробничої санітарії та професійної гігієни в нашій країні забезпечує закон України “Про забезпечення санітарного та епідемічного благополуччя населення”, який регулює суспільні відносини, що виникають у сфері забезпечення санітарного та епідемічного благополуччя, визначає відповідні права та обов’язки державних органів, підприємств, установ, організацій та громадян, встановлює порядок організації державної санітарно-епідеміологічної служби і здійснення державного санітарно-епідеміологічного нагляду.

Заходи щодо забезпечення виробничої санітарії та гігієни праці для обчислювального центру обладнаного ПК з ВДТ розроблені відповідно до вимог Державних санітарних норм та правил «Гігієнічна класифікація праці за показниками шкідливості та небезпечності факторів виробничого середовища, важкості та напруженості трудового процесу», МЮУ 06.05.2014 р. за

№ 472/25249, ДСанПіН 3.3.2.007-98 «Державні стандартні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» і НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин».

Робота програмістів в обчислювальному центрі операторської роботи, відносять до категорії 1а, тобто до неї належать роботи, що виконуються сидячи і не потребують фізичного напруження, робоче місце не змінюється, робота проводиться у холодний та теплий періоди року, тому рекомендується підтримувати оптимальні показники мікроклімату.

Мікрокліматичні умови виробничих приміщень характеризуються такими показниками:

- температура повітря;
- відносна вологість повітря;
- швидкість руху повітря.

За ступенем впливу на тепловий стан людини мікрокліматичної умови поділяють на оптимальні та допустимі.

Для робочої зони виробничих приміщень встановлюються оптимальні та допустимі мікрокліматичні умови з урахуванням важкості виконуваної роботи та періоду року. При одночасному виконанні в робочій зоні робіт різної категорії важкості рівні показників мікроклімату повинні встановлюватись з урахуванням найбільш чисельної групи працівників. Норми мікроклімату для операторської роботи в різні періоди року відображені в таблиці 7.1.

Таблиця 7.1 – Норми мікроклімату в приміщенні

Період Року	Категорія робіт	Температура повітря, °С	Відносна вологість повітря, %	Швидкість руху повітря, м/с
Холодний	Легка	22–24	40–60	0,1
Теплий	Легка	23–25	40–60	0,1-0,2

В обчислювальному центрі передбачено: устрій системи опалення приміщення для забезпечення необхідної температури повітря в холодний період року, а також кондиціонування приміщення в теплий період. відповідно ДБН В.2.5-67:2013 «Опалення, вентиляція та кондиціонування» [29].

Проведені дослідження мікрокліматичних умов на комп'ютеризованих робочих місцях показали, що зимою значення відносної вологості повітря часто можуть нижчими за встановлені норми і становити в середньому 30–40%. Це призводить не лише до надмірного висихання слизових оболонок очей, носа, горла, а й до нагромадження зарядів статичної електрики, що утворюються в процесі роботи комп'ютера. Для усунення таких проблем, в приміщенні передбачена наявність 3 зволожувачів повітря, які допомагають підняти вологість до 60%. Кількість зволожувачів зумовлена тим, що один такий апарат розраховано на приміщення площею до 15 м².

У теплий період року для підтримки нормальної температури повітря в приміщенні використовуються кондиціонери. Для збільшення площі обсягу в приміщенні наявні два кондиціонера, які розташовані у протилежних частинах обчислювального центру. Такі дії дозволяють підтримувати температуру повітря не вище 24°C. Кількість кондиціонерів зумовлена тим, що один такий апарат розраховано на кондиціонування приміщення площею до 18 м².

Вимоги до освітлення регламентовані ДБН В.2.5-28:2018, згідно з якими освітлення у приміщеннях з ЕОМ суміщене, при якому недостатнє за нормами природне освітлення доповнене штучним. Природне освітлення бічне, одностороннє.

В обчислювальному центрі коефіцієнт природної освітленості (КПО) має становити в середньому 1,1% (1,2% – у сонячний день та 1% – у хмарну погоду), це відповідає до вимогам ДБН В.2.5-28:2018. Для забезпечення відносної постійності природного освітлення вікна обладнанні сонцезахисними металевими жалюзі з системою регулювання. В таких жалюзі коефіцієнт відбиття становить 0,7. Для більшої зручності робочі місця

розташовано так, щоб у поле зору не потрапляли вікна або світлі поверхні світильників.

Найбільш прийнятними для приміщень, де працюють ПК, є люмінесцентні лампи ЛБ (білого світла) і ЛТБ (тепло-білого світла) потужністю 20, 40 або 80 Вт.

Далі наведені розрахунки кількості ламп, необхідних для освітлення приміщення розміром $10 \times 4 \times 3\text{ м}$ освітленістю $E_p = 400\text{лк}$. Коефіцієнт відбиття стелі 70% і стін 50%. Для освітлення використовуються люмінесцентні лампи типу ЛБ в світильниках ЛПО.

Знаходимо індекс приміщення

$$i = \frac{A \times B}{h \times (A + B)} = \frac{10 \times 4}{(3 - 0,7) \times (10 + 4)} = 0,95. \quad (7.1)$$

Прийнявши коефіцієнт запасу $k = 1,6$ і коефіцієнт нерівномірності освітлення можна розрахувати

$$z = \frac{E_{cp}}{E_{min}} = 1,1. \quad (7.2)$$

При індексі $i=0,95$ результат становить $\eta=40\%$.

Світильники розміщено в два ряди. $N_p = 2$

Визначаємо необхідний світловий потік ламп в кожному ряду:

$$\Phi_p = \frac{E_n \times S \times z \times k}{N_p \times \eta} = \frac{400 \times 40 \times 1,1 \times 1,6}{2 \times 0,4} = 35200\text{лм}. \quad (7.3)$$

Якщо в світильнику встановити по дві лампи ЛБ ($n = 2$) потужністю 40 Вт і світловим потоком $\Phi_{\text{л}} = 2500\text{ лм}$, то необхідне число світильників в ряду складе

$$N_p = \frac{\Phi_p}{n \cdot \Phi_n} = \frac{35200}{2 \cdot 2500} = 7. \quad (5.4)$$

Згідно із технічним паспортом обладнання (ПК, монітори, принтери, клавіатури (при натисканні), маніпулятори типу «миша» (при натисканні)), що використовується для роботи, рівень шуму складає: 43 дБА. Згідно із ДСН 3.3.6-039-99 нормоване значення рівня шуму більше, ніж фактичне, тож додаткових робіт для усунення надмірного шуму не проводилося. Роботи в даному приміщенні з таким обладнанням відповідають нормативним вимогам.

Окрім цього на програміста, в період роботи за дисплеєм, здійснюється інтенсивне електромагнітне випромінювання, яке може стати причиною професійних захворювань. Комп'ютери потрапляють за визначенням в 5-й діапазон (низькі частоти).

Також не меншою проблемою при роботі за комп'ютером складає електро-магнітне випромінювання (ЕМВ). Джерелами ЕМВ є не тільки дисплеї, сконструйовані на основі ЕПТ, але і рідкокристалічні та плазмені монітори, високочастотні перетворювачі джерела живлення портативних комп'ютерів.

Монітори на основі рідких кристалів не генерують шкідливих випромінювань, властивих звичайним моніторам, у них відсутнє рентгенівське випромінювання і електростатичне поле. За низької напруги мережі живлення такі дисплеї не мають електростатичного поля, але перемінні електричні і перемінні магнітні поля існують. ЕМП рідкокристалічних (РК) моніторів не перевищують допустимих норм, але тільки на відстані 50 см і більш від екрана. На менших відстанях може спостерігатися перевищення норм. Якщо до того ж монітори і системні блоки не заземлені, то ЕМВ на робочому місці користувачів комп'ютерів від моніторів на основі ЕПТ і РК моніторів не відрізняються. Обладнання і організація робочого місця в обчислювальному центрі мають забезпечувати відповідність конструкції всіх елементів робочого місця та їх взаємного розташування ергономічним вимогам з урахуванням характеру і особливостей трудової діяльності [30].

Площа на одне робоче місце з ПК складає по 8 м², а об'єм – по 28 м³. Приміщення має природне та штучне освітлення. Робочі місця розташовано так, щоб природне світло падало збоку.

Площа робочого місця користувача ПК повинна становити не менше 4,5 м² (об'ємні норми на одну людину – не менше 20 м³). У приміщеннях повинна проводитися щоденне вологе прибирання і систематичне провітрювання після кожної години роботи. Шумляче обладнання (друкарські пристрої, сканери, сервери і тому подібні), рівні шуму якого перевищують нормативні, повинні розміщуватися за межами робочих місць співробітників.

Робота з комп'ютером характеризується значною розумовою напругою і нервово-емоційним навантаженням операторів, високою напруженістю зорової роботи і достатньо великим навантаженням на м'язи рук при роботі з клавіатурою ЕОМ. Велике значення має раціональна конструкція і розташування елементів робочого місця, що важливе для підтримки оптимальної робочої пози людини-оператора.

В приміщенні обчислювального центру наявні 2 медичні аптечки першої допомоги для екстрених ситуацій.

Окрім цього приміщення обладнано шафами для зберігання документів, магнітних дисків, полицями, стелажми та тумбами.

Колір приміщень і меблів сприяє створенню комфортних умов для зорового сприйняття.

Загальний план обчислювального центру та робочих місць наведено на рисунку 7.1.

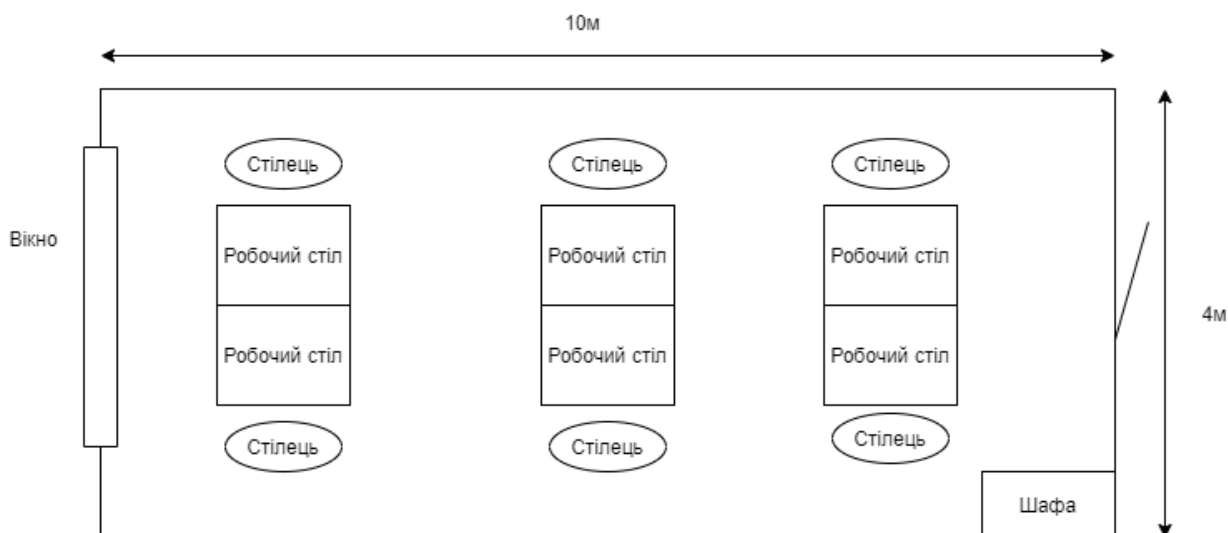


Рисунок 7.1 – Загальна схема обчислювального центру

7.4 Заходи безпеки у надзвичайних ситуаціях

7.4.1 Заходи з пожежної безпеки

Розробка програмного забезпечення супроводжується використанням не лише комп'ютерної техніки та необхідних меблів, а також веденням відповідної документації для чого в приміщенні додатково встановлюється необхідне електрообладнання (принтери) та організовується зберігання паперів. Також за необхідності може бути встановлене і додаткове електрообладнання (проектори, кухонні прилади тощо). Вся техніка, що може знаходитися в приміщенні, меблі, а також супутні матеріали, при неправильній організації пожежної безпеки можуть спричинити пожежу. Для мінімізації ризику виникнення пожежі в обчислювальному центрі існує необхідність впровадження правил та заходів з пожежної безпеки.

Планування та впровадження заходів пожежної безпеки здійснено у відповідності до Закону України Про пожежну безпеку, Правил пожежної безпеки в Україні та вимогам відповідних нормативних актів:

- меблі та обладнання необхідно розміщувати таким чином, щоб забезпечувався вільний евакуаційний прохід до дверей виходу з приміщення

(завширшки не менше 1 м). Евакуаційні шляхи та виходи необхідно постійно утримувати вільними, нічим не захащувати;

- електромережі, електроприлади і апаратуру експлуатувати тільки у справному стані з урахуванням вказівок та рекомендацій підприємств-виготовлювачів. У разі виявлення пошкоджень електромереж, вимикачів, розеток та інших електровиробів слід негайно вимкнути їх та вжити необхідних заходів щодо приведення в пожежобезпечний стан;

- документи, папір та інші горючі матеріали слід зберігати на відстані не менше 1 м від електрощитів; 0,5 м від електросвітильників; 0,6 м від сповіщувачів автоматичної пожежної сигналізації та 0,15 м від приладів центрального водяного опалення;

- засоби протипожежного захисту слід утримувати у справному стані.

Усі працівники повинні вміти користуватись наявними вогнегасниками, іншими первинними засобами пожежогасіння, знати місце їх знаходження.

Відстань від найбільш віддаленого місця приміщення до місця розташування вогнегасника не повинна перевищувати 20 м. [30].

Згідно НАПБ.03.001-2004 "Типові норми належності вогнегасників" тип і необхідну кількість вогнегасників визначають згідно з наступними параметрами:

- граничну захисну площу;
- клас пожежі горючих речовин і матеріалів;
- категорію приміщень і будівель за вибухопожежною і пожежною небезпекам.

При плануванні заходів з пожежної безпеки враховано, що обчислювальний центр, згідно НАПБ Б.03.002-2007 «Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою» відноситься до категорії «Д», а клас можливої

пожежі, відповідно до ДСТУ 4297:2004 «Пожежна техніка. Технічне обслуговування вогнегасників. Загальні технічні вимоги», визначається як «А» (горіння твердих речовин) та «Е» (горіння електроустановок, що перебувають під напругою). Враховуючи площу приміщення, що складає 42 м², згідно НАПБ Б.03.001-2004 «Типові норми належності вогнегасників», приміщення обладнане двома порошковими вогнегасниками ВП-5.

Для підвищення рівня протипожежної безпеки було розроблено графік повірки вогнегасників і призначено відповідальну особу.

Окрім цього, згідно вимог ДБН В.2.5-56:2014 «Системи протипожежного захисту», приміщення обладнане пожежною сигналізацією, що використовує комбіновані теплові мультисенсорні датчики та димові іонізаційні датчики, підключені до сигналізаційної димової пожежної установки типу СДПУ-1.

Для обробки стін, підлоги та стелі приміщення використано негорючі полімерні матеріали, що зменшує ризики швидкого розповсюдження вогню у разі виникнення пожежі.

З організаційних заходів щодо забезпечення пожежної безпеки відповідно до вимог ДБН В.1.1-7:2016 «Пожежна безпека об'єктів будівництва. Загальні вимоги» розроблено план евакуації персоналу в разі виникнення пожежі і розміщено в доступному для перегляду місці. Окрім цього було проведено інструктаж з персоналом щодо дотримання правил пожежної безпеки у приміщенні і вміння користуватися наявними засобами пожежогасіння. Для перевірки навичок персоналу під час пожежі було розроблено план навчань (з використанням учбових пожежних тривог).

Ці міри мінімізують вірогідність пожежі а в разі її виникнення, мінімізують шкоду яку вона може нанести.

7.4.2 Заходи з цивільного захисту. Вплив електромагнітного імпульсу на елементи виробництва

Електромагнітний імпульс (ЕМІ) – це короткий сплеск електромагнітної енергії. Джерело такого імпульсу може бути природним або штучним і може виглядати як випромінювання, електричне або магнітне поле або проводиться електричний струм, в залежності від джерела. ЕМІ-перешкоди зазвичай руйнують або пошкоджують електронне обладнання, а при більш високих рівнях енергії потужна ЕМІ-подія, така як удар блискавки, може пошкодити фізичні об'єкти, такі як будівлі і конструкції.

Переважно розглядають два види електромагнітних імпульсів:

- ядерний ЕМІ – імпульс, що виникає під час ядерного вибуху;
- ЕМІ від розряду блискавки – імпульс, що виникає під час електричного розряду в атмосфері.

Окрім цих джерел існують і інші джерела, але вони генерують імпульси значно меншої потужності.

Електромагнітний імпульс індукує високу електричну напругу в електромережах, електричному і електронному обладнанні. Зростання напруженості спричиняє раптове зростання електричної напруги і виділення великої кількості тепла, внаслідок чого зазнають пошкоджень електронні елементи, електричні кола і навіть лінії електропередачі. Високі напруги також можуть призвести до пробію електричної ізоляції.

Зміна властивостей іоносфери Землі, викликана ЕМІ призводить до появи перешкод у радіозв'язку.

Енергія ЕМІ може передаватися в будь-якій з чотирьох форм:

- електричне поле;
- магнітне поле;
- електромагнітне випромінювання;
- електропровідність;

Імпульс електричної енергії завжди буде супроводжуватися імпульсом магнітної енергії. Зазвичай у імпульсі домінує або електрична, або магнітна форма.

Загалом, випромінювання діє лише на великі відстані, а магнітне та електричне поля діють на невеликі відстані.

Вражаюча дія електромагнітного імпульсу обумовлена виникненням наведених напруг і струмів в різних провідниках. Дія ЕМІ виявляється, перш за все, по відношенню до електричної та радіоелектронної апаратури. Найбільш уразливі лінії зв'язку, сигналізації та управління. При цьому може статися пробій ізоляції, пошкодження трансформаторів, псування напівпровідникових приладів і т.п. Висотний вибух здатний створити перешкоди в цих лініях на дуже великих площах.

Таким чином при виникненні електромагнітного імпульсу на виробництві існує ризик виходу з ладу всіх електронних, радіоелектронних пристроїв та обчислювальної техніки. Незважаючи на те що ЕМІ не спричиняє прямої дії на людину, вихід з ладу електронних засобів може спричинити травму співробітників.

Сигнали ЕМІ високого рівня можуть становити загрозу безпеці людини. В таких обставинах слід уникати прямого контакту з проводом під напругою. Коли це відбувається, наприклад, при дотику до електричного прилада або іншого сильно зарядженого об'єкту, слід подбати про те, щоб звільнити об'єкт, а потім розрядити тіло через високий опір, щоб уникнути ризику небезпечного ударного імпульсу при відступаючи. Дуже висока напруженість електричного поля може викликати пробою в повітрі і потенційно смертельний струм дуги, аналогічний протіканню блискавки, але напруженість електричного поля до 200 кВ / м вважається безпечною.

До методів захисту від ЕМІ можна віднести спеціальні автоматичні вимикаючі прилади (розрядники), які в разі скачків напруги знеструмлюють

підприємство, екрани, заземлення та інші міри для забезпечення безпечного струму

7.5 Висновки за розділом 7

В ході проєктування системи було створено необхідні умови праці для операторів обчислювальної техніки. Окрім цього було вжито заходів безпеки для співробітників згідно сучасних норм України з охорони праці а також оптимізовано ергономіку робочих місць.

ВИСНОВКИ

Під час виконання дипломної роботи була розроблена система рекомендацій для багатокористувацького веб-порталу новин. Було вирішено такі задачі:

- проведено аналіз предметної області;
- проаналізовано існуючі аналоги;
- проаналізовано існуючі мови програмування;
- виконано проєктування додатку;
- розроблено програмний продукт;
- проведено тестування;
- розраховано вартість розробки і введення програмного продукту;
- продумана охорона праці під час експлуатації програмного забезпечення.

Було оптимізовано алгоритм пошуку k -найближчих сусідів для підвищення точності роботи.

Розроблене програмне забезпечення призначене для рекомендацій новин.

Для реалізації програмного продукту була використана мова програмування PHP, та бібліотека PHP ML, окрім цього були описані особливості обраної мови програмування.

В результаті виконання дипломної роботи з використанням методів об'єктно-орієнтованого програмування було спроектовано алгоритм, який повністю реалізує поставлену задачу.

ПЕРЕЛІК ДЖЕРЕЛ І ПОСИЛАННЯ

1. DIGITAL 2020: GLOBAL DIGITAL OVERVIEW [Electronic resource]. – Режим доступу : <https://datareportal.com/reports/digital-2020-global-digital-overview>
2. Михайлуша, О.О. Журналістика та публіцистика: спільне і відмінне [Текст] / О.О.Михайлуша. – К.: НАУ, 2011. – 17 с.
3. Манукян, В.І. Журналістика крізь призму права. Журналісту про право і правах. Правознавцеві про журналістику [Текст] / В.І. Манукян. – К.: Право, 2010. – 384 с.
4. Антонов-Овсієнко, А. Журналістика XXI. Нові ЗМІ и свобода слова [Текст] / А. Антонов-Овсієнко. – К.: – Folio, 2015. – 301с.
5. Andrew, R. The CSS3 Anthology: Take Your Sites to New Heights [Text] / R. Andrew. – SitePoint, 2012. – 443 с.
6. AlexSoft. Майстер-Самовчитель PHP в задачах і прикладах [Текст] / AlexSoft. – AlexSoft, 2009. – 420 с.
7. Aronson, L. HTML Manual of Style: A Clear, Concise Reference for Hypertext Markup Language (Including HTML5) 4th Edition. [Text] / L. Aronson. – Addison-Wesley, 2010. – 336 с.
8. Aaron, A. HTML 5 & CSS3 Genius Guide Vol.3 [Text] / A.Aaron. – Imagine Publishing, 2016. – 180 с.
9. PHP, C#, Python – коротка характеристика трьох мов програмування [Електрон. ресурс]. – Режим доступу: <https://www.internet-technologies.ru/articles/php-csharp-python-harakteristika-yazykov-programmirovaniya.html>
10. Бринк, Х. – Машинне навчання [Текст] / Хорайзон Бринк, Джозеф Річардс Марк Феверофл. – СПб. : Пітер, 2017. – 326 с.
11. Джоел, Г. – Data Science. Наука про дані з нуля [Текст] / Г.Джоел– СПб.: Пітер, 2014. – 336 с.
12. Люгер, Д.Ф. Штучний інтелект:стратегії вирішення складних

проблем [Текст] / Д.Ф.Люгер.– К.: Центр учбової літератури, 2016. – 250 с.

13. Як працюють рекомендаційні системи [Електрон. ресурс]. – Режим доступу : <https://neurohive.io/ru/osnovy-data-science/rekomendatelnye-sistemy-modeli-i-ocenka/>.

14. Анатомія рекомендаційних систем [Електрон. ресурс]. – Режим доступу : <https://habr.com/ru/company/lanit/blog/420499/>.

15. Михайлин, І.Л. Основи журналістики [Текст] / І.Л.Михайлин. – К.: Центр учбової літератури, 2011. – 496 с.

16. Falk, K. Practical Recomender Systems [Text] / Kim Falk USA: Wanning, 2015. – 268 с.

17. Ways to Evaluate Regression Models [Electronic resource]. – Режим доступу : <https://towardsdatascience.com/ways-to-evaluate-regression-models-77a3ff45ba70>.

18. Метрики в задачах машинного навчання [Електрон. ресурс]. – Режим доступу : <https://habr.com/ru/company/ods/blog/328372/>.

19. Габрієлян, Т.О. Бренд в графічному дизайні: концептуалізація, візуалізація, ідентифікація [Текст] / Т.О.Габрієлян. – Сімферополь : Антиква, 2018. – 228 с.

20. Коробко, О.В. Охорона праці і безпека в надзвичайних ситуаціях [Текст] / О.В.Коробко – Запоріжжя : ЗНТУ, 2016. – 102 с.

21. Kuhn, M. Applied Predictive Modeling [Text] / M. Kuhn, K. Johnson, 2013. – 251 с.

22. Шостак, М. І. Новинна журналістика. Новини преси: підручник і практикум для бакалаврату та магістратури / М. І. Шостак. – 2-е вид., Доп. – М: Юрайт, 2016. – 237 с.

23. Посібник з PHP [Електрон. ресурс]. – Режим доступу : <https://htmlacademy.ru/tutorial/php>.

24. Бойчик, І.М. Економіка підприємств: навч посібник / І. М. Бойчик, П. С. Харів, М. І. Хопчан. – Львів: Сполом. – 2009. – 212 с.

25. Вовчак, О. Д. Інвестування: навч. посіб. / О. Д. Вовчак. – Л.: Новий

Світ-2000, 2008. – 544 с.

26. Економіка підприємства: підручник: / За ред. С.Ф. Покропивного. – Вид.2-ге, – К.: КНЕУ, 2013. – 528 с.

27. Єрмошенко, М. М. Аналіз і оцінка інвестиційних проектів: навч. посіб. / М. М. Єрмошенко, І. О. Плужніков. – К.: Нац. акад. управл., 2006. – 156 с.

28. Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом: ДСТУ Б В.2.5-82:2016. – [Чинний від 2017-04-01]. – К.: ДП УкрНДНЦ, 2016. – 109 с. – (Державний Стандарт України) Правила безпечної експлуатації електроустановок: НПАОП 40.1-1.01-97. – [Чинний від 1997-10-06]. – К.: Держнаглядохоронпраці, 1997. – 97 с.

29. Правила безпечної експлуатації електроустановок: НПАОП 40.1-1.01-97. – [Чинний від 1997-10-06]. – К.: Держнаглядохоронпраці, 1997. – 97 с.

30. Санітарні норми мікроклімату виробничих приміщень: ДСН 3.3.6.042-99. – [Чинний від 1999-12-01]. – К.: МОЗ України, 1999. – 106 с.

ДОДАТОК А
Технічне завдання

A.1 Вступ

A.1.1 Назва розроблюваного програмного забезпечення

Назва розроблюваного програмного забезпечення: «Рекомендаційна система для багатокористувацького веб-порталу новин».

A.1.2 Призначення та область застосування

Система призначена для перегляду рекомендованих для кожного користувача новин.

A.2 Підстави для розробки

Підставою для розробки є завдання на дипломну роботу на тему «Рекомендаційна система для багатокористувацького веб-порталу новин».

A.3 Призначення розробки

Розробка призначена для створення персональних рекомендацій.

A.4 Вимоги до програмного виробу

A.4.1 Вимоги до функціональних характеристик

Алгоритм має виконувати такі функції:

- навчання системи за допомогою переглядів користувача;
- виведення рекомендацій на основі інтересів користувача.

До вхідних даних програми відноситься дані користувача, дані про статтю з якою йде взаємодія .

Вихідні дані програми організовані у вигляді інтерфейсу програми, який бере інформацію з бази даних та структурує її.

Інтерфейс користувача повинен забезпечити просту й ефективну взаємодію користувача з програмним продуктом.

A.4.2 Вимоги до надійності

Програма повинна розпізнавати ситуації взаємного нерозуміння між програмою і користувачем, обробляти збійні ситуації шляхом підказок користувачу. У випадку критичних помилок коректна робота системи повинна відновлюватися після перезавантаження системи.

A.4.3 Умови експлуатації

Кліматичні умови експлуатації, при яких повинні забезпечуватися задані характеристики автоматизованої системи, повинні задовольняти вимогам, що пред'являються до технічних засобів у частині умов їх експлуатації.

Користувач системи повинен мати базові навички володіння комп'ютером (ноутбуком) з операційною системою Windows 7 або вище та з веб-браузером.

A.4.4 Вимоги до складу та параметрів технічних засобів

Для відсутності проблем функціонування системи на комп'ютері користувача повинні виконуватися такі умови до апаратної та програмної частини:

- процесор із тактовою частотою 2 GHz та більше;
- можливість підтримки компонентів Microsoft .Net 4.5.2;

- оперативна пам'ять 1 ГБ та більше;
- вільний простір на жорсткому диску не менше, ніж 15 Мб.

А.4.5 Вимоги до інформаційної та програмної сумісності

Вимоги до інформаційних структур, методів рішення та інструментів розробки не висуваються.

А.4.6 Вимоги до маркування та упакування

Програмний засіб зберігатиметься у вигляді папки з необхідними для роботи програми файлами (або у вигляді архіву .zip або .rar).

А.4.7 Вимоги транспортування та збереження

Програмний засіб розповсюджуватиметься будь-яким способом у вигляді папки з необхідними для роботи програми файлами (або у вигляді архіву .zip або .rar).

Додаток Б
Текст розробленої програми

Б.1 Лістинг index.php

```

<?php
include"/includes/config.php";
$page_name="Головна";
include"/includes/database.php";
include"/includes/headerf.php";
include"/includes/ml.php";

$res = mysqli_query($connect, "SELECT * FROM Article");
$ress = mysqli_query($connect, "SELECT * FROM Article ORDER BY Views ASC ");
$last = mysqli_query($connect, "SELECT * FROM Article ORDER BY Date DESC LIMIT
0,3");
$pop = mysqli_query($connect, "SELECT * FROM Article ORDER BY Likes DESC LIMIT
0,3");
$categ= mysqli_query($connect, "SELECT * FROM Category LIMIT 0,3");
?>

$rec= new Recommend;
$rec->getData();
$rec->transform();

<div class="art-pop">
<h2>Рекомендації</h2>
<divclass="art-pop-parent">
<?while($arp=mysqli_fetch_assoc($pop)){?>
<divclass="art-pop-child">

<br>
<div class="art-pop-child-img"> <h1> <a href ='/article.php?id=?php echo $arp['id'] ?>'>
" alt=""> </a> </h1> </div> <br>

<divclass="art-pop-child-content">

```

```

        <h1> <a href ='/article.php?id=<?php echo $arp['id'] ?>'><?php echo $arp['Name']; ?>
</a></h1>

        <?php$text=strip_tags($arp['Text']);
        echo"<p>".mb_substr($text,0,100,'utf-8')."</p>";?>

</div>

<?
}

```

Б.2 Лістинг database.php

```

        <?php    $connect    =    mysqli_connect(    "localhost","root",    "",    "BlogSQL");
if($connect==false){
    echo"Error:Cannot    connect    to    database.Reason:    <br>";
    echomysqli_connect_error();
}

```

Б.3 Лістинг learning.php

```

?php
include"/includes/config.php";
include"/includes/database.php";
include"/includes/headerf.php";
include"/includes/ml.php";
?>

<?php
class Learning {
    public $text = 'aMemberVarMemberVariable';
    public $Name = 'aMemberFunc';
    public $date = 'aMemberFunc';
}

```

```

function toDb(){
$userdata=$_POST['user'];
$articledata=$_POST['article'];

if($_POST["isliked"]==0){
    mysqli_query($connect,"INSERT INTO Likes(Article_id,User_id) VALUES
('$articledata','$userdata')");
    mysqli_query($connect,"UPDATE Recommendations SET Articles = Articles+1 WHERE
(id=$articledata)");
}
else
{
    mysqli_query($connect,"DELETE FROM Likes WHERE (Article_id=$articledata) AND
(User_id=$userdata)");
    mysqli_query($connect,"UPDATE Article SET Likes = Likes-1 WHERE (id=$articledata)");
}

'; if(mysqli_query($connect,"INSERT INTO Recomendations(Name , Text, Date, Views, Likes
,Category_id) VALUES ('$name','$Text','$date','0','0','$cate')")){

    if(mysqli_query($connect,"INSERT INTO User(Nickname , Password , EMail,
Role_id, Avatar) VALUES ('$login','$password','$email','3','avatar.jpeg')")){

        $data=mysqli_query($connect, "SELECT * FROM User WHERE
User.Nickname='$login' AND User.Password='$password'");
        $user=mysqli_fetch_assoc($data);

        if(isset($_POST['remember'])){
            setcookie('id',$user['id'],time()+3600*24*21);
            setcookie('username',$user['Nickname'],time()+3600*24*21);
            setcookie('role_id',$user['Role_id'],time()+3600*24*21);

```

```

        setcookie('Avatar',$user['Avatar'],time()+3600*24*21);
        $r_id = $user['Role_id'];
        $query = mysqli_query($connect, "SELECT * FROM Role WHERE
Role.id='$r_id'");
        $r_name = mysqli_fetch_assoc($query);
        setcookie('role_name',$r_name['Role'],time()+3600*24*21);
    }

    $_SESSION['id'] = $user['id'];
    $_SESSION['username'] = $user['Nickname'];
    $_SESSION['role_id'] = $user['Role_id'];
    $_SESSION['Avatar']=$user['Avatar'];
    $r_id = $user['Role_id'];
    $query = mysqli_query($connect, "SELECT * FROM Role WHERE
Role.id='$r_id'");
    $r_name = mysqli_fetch_assoc($query);
    $_SESSION['role_name'] = $r_name['Role'];

    return 0
}
}

```

Б.4 Лістинг article.php

```

<?php
include"/includes/config.php";
include"/includes/database.php";

$id=$_GET['id'];
$userid=$_SESSION["id"];
mysqli_query($connect,"UPDATE Article SET Views = Views+1 WHERE (id=$id)");

$res2 = mysqli_query($connect, "SELECT * FROM Article WHERE Article.id=$id");

```

```

if(isset($_SESSION['id'])){

    mysqli_query($connect,"INSERT INTO Recommendations(user_id,article_id)
VALUES('$userid','$id')");
    $reslike=mysqli_query($connect,"SELECT * FROM Recommendations WHERE
Article_id=$id AND User_id=$userid");
    $datalike=mysqli_num_rows($reslike);}
$lea = new Learning;
$lea->toDb
$ar2=mysqli_fetch_assoc($res2);
$page_name=$ar2["Name"];

include"/includes/headerf.php";
?>
<br>
<div class="art-img">
    ">
    <h1> <?php echo $ar2['Name']; ?> </h1>
</div>

$res = mysqli_query($connect, "SELECT * FROM Article WHERE
Recommend.Category_id=$cat_id");

$page = mysqli_query($connect, "SELECT * FROM Category WHERE Category.id=$cat_id");

$page_r = mysqli_fetch_assoc($page);
$page_name=$page_r["Category"];

include "/includes/headerf.php";?>

<h1><?php echo "Всі статті з категорії:". $page_name ?></h1>

```

Б.5 Лістинг recommend.php

```

?php
include"/includes/config.php";
include"/includes/database.php";
include"/includes/headerf.php";
include"/includes/ml.php";
?>

<?php
class Recommend {
    public $sample;

    function getData(){
        $res = mysqli_query($connect, "SELECT * FROM Recommend WHERE
        Recomend.User_id=$SESSION[“user”]");
        return $res
    }

    function transform($res){
        $transformer = new TfIdfTransformer($res);
        $cosine= sqrt(array_sum($res))/ pow(sqrt(array_sum($res)));
        $transformer->transform($cosine);

        return $cosine
    }
}

```

ДОДАТОК В
Слайди презентації

Запорізький національний технічний університет.
Кафедра програмних засобів

Дипломна робота магістра на тему: «Система рекомендацій для багатокористувацького веб-порталу новин»

Підготував ст.гр.КНТ-119м:
Керівник: к.т.н доцент

МАКАРЕНКО Р.І.
КОЛПАКОВА Т.О.

Рисунок В.1 – Слайд 1

Мета, об'єкт, предмет і задачі дослідження

- Об'єкт дослідження - дослідження та програмна реалізація рекомендаційної системи для порталу новин.
- Предмет дослідження - програмний застосунок для рекомендацій новин на основі вподобань користувача.
- Мета роботи - створення програмної рекомендаційної системи підбору новин для кожного читача. Удосконалення методу к найближчих сусідів.

2

Рисунок В.2 – Слайд 2

Аналіз предметної області

На сьогоднішній день для людей є можливість в будь-якому місці в будь-який час дізнаватися актуальні новини маючи під рукою лише доступ в інтернет. Для вирішення цієї проблеми розробники в усьому світі намагаються створити додатки за допомогою яких це можна робити швидко і зручно. Окрім цього існує потреба в виокремленні новин для різних користувачів за предметом їхніх інтересів. Для вирішення цієї задачі існують рекомендаційні системи

3

Рисунок В.3 – Слайд 3

Завдання до роботи

- ▶ Аналіз програмних аналогів для визначення вимог до розроблюваного програмного забезпечення;
- ▶ дослідження методів створення персональних рекомендацій;
- ▶ проектування програмної системи;
- ▶ реалізація програмного забезпечення;
- ▶ тестування програмного забезпечення.

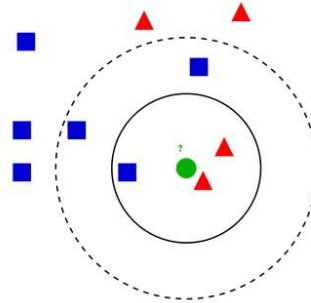
4

Рисунок В.4 – Слайд 4

Матеріали і методи

Метод k-найближчих сусідів(KNN) - метричний алгоритм для автоматичної класифікації об'єктів або регресії.

У разі використання методу для класифікації об'єкт присвоюється того класу, який є найбільш поширеним серед k сусідів даного елемента, класи яких вже відомі.



5

Рисунок В.5 – Слайд 5

Оптимізація алгоритму

- ▶ Звичайний алгоритм k найближчих сусідів використовує евклідову або манхеттенську відстань;
- ▶ розроблений метод використовує косинусну подібність, що збільшує точність класифікації.

$$\text{cosine}(x, y) = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}}$$

6

Рисунок В.6 – Слайд 6

Схема роботи програмного продукту ч.1

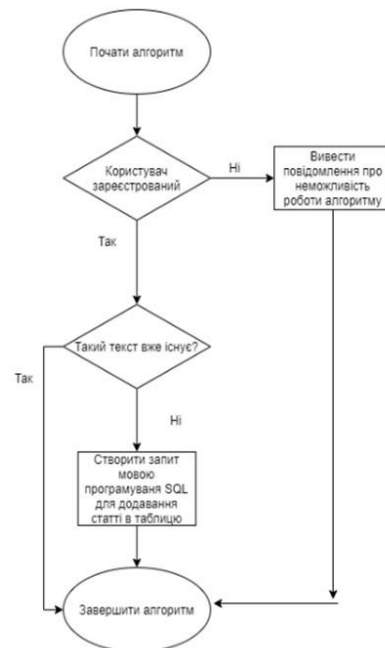


Рисунок В.7 – Слайд 7

Схема роботи програмного продукту ч.2



Рисунок В.8 – Слайд 8

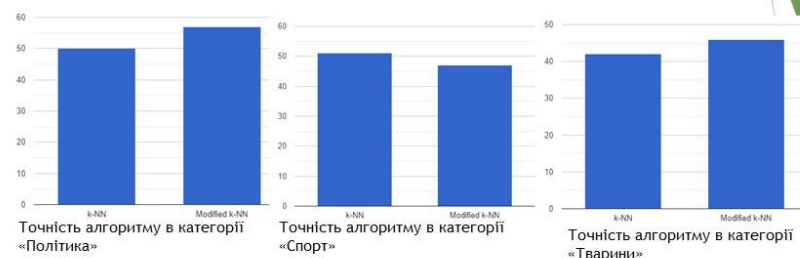
Графічний інтерфейс



9

Рисунок В.9 – Слайд 9

Порівняння алгоритмів



10

Рисунок В.10 – Слайд 10

Висновки

- ▶ На основі проаналізованих аналогів було сформовано основні функціональні вимоги до розроблюваної програми;
- ▶ Був покращений метод к найближчих сусідів розроблений метод з використанням косинусної подібності;
- ▶ була повністю реалізована функціональність системи на основі сформованих вимог;
- ▶ розроблений програмний продукт був успішно протестований.
- ▶ Отримані результати говорять про те, що розроблене програмне забезпечення цілком відповідає поставленій меті розробки.

11

Рисунок В.11 – Слайд 11