

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ
ЗАСОБІВ ОЦІНЮВАННЯ КОМЕНТАРІВ КОРИСТУВАЧІВ
RESEARCH AND SOFTWARE IMPLEMENTATION OF
TOOLS FOR EVALUATING USER COMMENTS

Виконав(ла): студент(ка) 2 курсу, групи КНТ-113М
Спеціальності 121 Інженерія програмного
(код і найменування спеціальності)

забезпечення

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

КОРОБКА І.С.

(ПРІЗВИЩЕ та ініціали)

Керівник КОЦУР М.І.

(ПРІЗВИЩЕ та ініціали)

Рецензент СКРУПСЬКИЙ С.Ю.

(ПРІЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет КНТ
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.
Сергій СУББОТІН
“ ” 2024 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

КОРОБКИ Івана Сергійовича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна реалізація засобів оцінювання коментарів користувачів. Research and Software Implementation of Tools for Evaluating User Comments

керівник проєкту (роботи) к.т.н., доцент, КОЦУР Михайло Ігорович,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “07” жовтня 2024 року № 398

2. Строк подання студентом проєкту (роботи) 02 грудня 2024 року

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

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Матеріали і методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

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

—

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	КОЦУР М.І., доцент		
Нормоконтроль	КАМІНСЬКА Ж.К., асистент		

7. Дата видачі завдання “9” вересня 2024 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.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій	2 тиждень	Розділ 2
	розробки.		
4	Розробка структури програми.	3 тиждень	Розділ 3
5	Розробка програми.	4-7 тижні	Розділи 3,4
6	Тестування та експериментальне дослідження	8 тиждень	Розділ 4
	програмного забезпечення.		
7	Оформлення пояснювальної записки та документів	9-10 тижні	Додатки
	до неї.		
8	Нормоконтроль та рецензування.	11 тиждень	
9	Захист роботи.	12 тиждень	

Студент(ка)

(підпис)

Іван КОРОБКА

(Імя ПРІЗВИЩЕ)

Керівник проєкту (роботи)

(підпис)

Михайло КОЦУР
(Ім'я ПРІЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
97 с., 3 табл., 37 рис., 3 дод., 24 джерел.

DJANGO, PYTHON, ВІДГУК КОРИСТУВАЧА, ЗАСТОСУНОК,
КОМЕНТАР, МОВА ПРОГРАМУВАННЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ.

Об'єкт дослідження – процес розробки програмного забезпечення для оцінювання емоційного забарвлення текстових повідомлень.

Предмет дослідження – програмні засоби для оцінювання коментарів користувачів.

Мета роботи – дослідження та розробка програмного забезпечення для оцінювання коментарів користувачів.

Матеріали, методи та технічні засоби: мова програмування Python, фреймворк Django.

Результати. Розроблено програмне забезпечення для оцінювання коментарів користувачів за допомогою мови програмування Python та фреймворку Django.

Практична цінність роботи полягає у розробці програмного забезпечення, що забезпечує оцінювання коментарів користувачів.

Висновки. Виконано проєктування програмного забезпечення для оцінювання коментарів користувачів. Розроблено програмне забезпечення для оцінювання коментарів користувачів. Здійснено тестування розробленого програмного забезпечення для оцінювання коментарів користувачів.

Галузь використання – соціальні мережі та Інтернет продаж товарів.

ABSTRACT

Explanatory note to the diploma qualifying work of the master: 97 pages, 3 tables, 37 figures, 3 appendixes, 24 sources.

DJANGO, PYTHON, USER FEEDBACK, APPLICATION, COMMENT, PROGRAMMING LANGUAGE, SOFTWARE.

The object of research is the process of developing software for evaluating the emotional color of text messages.

The subject of the research is software for evaluating user comments.

The purpose of this work is to develop the software for evaluating user comments.

Materials, methods and technical tools: Python programming language, Django framework.

Results. The software tools for evaluating user comments using the Python programming language and the Django framework have been developed.

The practical value of the work lies in the development of software that provides evaluation of user comments.

Conclusions. The software for evaluating user comments has been designed. The software for evaluating user comments has been developed. The developed software for evaluating user comments has been tested.

Scope of use – social networks and online sales of goods.

ЗМІСТ

	С.
Перелік скорочень та умовних позначок	8
Вступ.....	9
1 Аналіз предметної області.....	11
1.1 Програмне забезпечення для оцінювання коментарів користувачів	11
1.2 Аналіз програмного забезпечення для оцінювання коментарів користувачів	22
1.3 Висновки за розділом 1	33
2 Матеріали і методи	35
2.1 Вибір мови програмування	35
2.2 Вибір фреймворку для оцінювання коментарів користувачів	37
2.3 Розробка моделі взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів	38
2.4 Висновки за розділом 2	41
3 Опис програми	42
3.1 Структура програмного забезпечення для оцінювання коментарів користувачів	42
3.2 Реалізація процесу оцінювання коментарів користувачів за допомогою нейромережових моделей	44
3.3 Реалізація процесу оцінювання коментарів користувачів.....	49
3.4 Проєктування інтерфейсу програмного забезпечення для оцінювання коментарів користувачів	52
3.5 Висновки за розділом 3	58
4 Експлуатація, тестування та експериментальне дослідження програми.....	60
4.1 Призначення й умови застосування програми	60
4.2 Характеристики програми для оцінювання коментарів користувачів....	61
4.3 Інструкція по експлуатації програми.....	62
4.3.1 Звернення до програми	62
4.3.2 Вхідні й вихідні дані	62

4.3.3 Повідомлення.....	62
4.4 Виконання програмного забезпечення для оцінювання коментарів користувачів	63
4.5 Тестування програмного забезпечення для оцінювання коментарів користувачів	69
4.6 Висновки за розділом 4	69
Висновки	70
Перелік джерел посилань	73
Додаток А Технічне завдання	75
Додаток Б Фрагмент тексту програми	79
Додаток В Слайди презентації	90

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

CNN – Convolutional Neural Network;

MVC – Model-View-Controller;

ПЗ – програмне забезпечення;

ТЗ – технічне завдання.

ВСТУП

Актуальність програмного забезпечення для оцінювання коментарів користувачів зростає в умовах цифровізації та розширення онлайн комунікацій [1]-[3].

Компанії та бренди активно відслідковують та аналізують відгуки й коментарі користувачів для підтримання та покращення своєї репутації. Програмне забезпечення для оцінювання коментарів дозволяє швидко виявляти негативні відгуки та реагувати на них, що є критичним для збереження довіри клієнтів [2]-[5].

Аналіз коментарів за допомогою відповідного програмного забезпечення допомагає виявляти типові проблеми, з якими стикаються користувачі, що дозволяє оперативно реагувати на них та вдосконалювати продукти чи послуги [3]-[6].

Оцінювання та аналіз коментарів в автоматичному режимі за допомогою відповідного програмного забезпечення дозволяє компаніям отримувати зворотній зв'язок від клієнтів у реальному часі, виявляти недоліки та відповідно коригувати свої продукти чи послуги [1]-[6].

Отже, в умовах зростаючої важливості онлайн комунікацій програмне забезпечення для оцінювання коментарів стає незамінним інструментом для будь-якої компанії, яка прагне залишатися конкурентоспроможною і забезпечувати високу якість обслуговування клієнтів [1]-[6].

Проте деякі програми для оцінювання коментарів користувачів можуть не завжди правильно інтерпретувати контекст коментарів, що обумовлено використанням неефективних методів штучного інтелекту та розпізнавальних моделей, що може призвести до неправильної оцінки настрою або значущості коментарів. Деяке програмне забезпечення може мати обмежені можливості для точного аналізу коментарів, написаних на різних мовах або діалектах, що особливо актуально для глобальних компаній, які працюють на ринках різних країн. Для того щоб деяке програмне забезпечення для оцінювання коментарів

користувачів працювало ефективно, може знадобитися значний обсяг часу і ресурсів на його налаштування під специфічні потреби бізнесу, що включає адаптацію моделей і інтеграцію з іншими системами [1]-[6].

Тому актуальною є розробка програмного забезпечення для оцінювання коментарів користувачів. У дипломній кваліфікаційній роботі магістра розв'язується актуальне завдання дослідження та розробки програмного забезпечення для оцінювання коментарів користувачів.

Для досягнення поставленої мети у кваліфікаційній роботі магістра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для оцінювання коментарів користувачів;
- розробити модель взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів;
- здійснити проєктування програмного забезпечення для оцінювання коментарів користувачів;
- створити програмне забезпечення для оцінювання коментарів користувачів;
- виконати тестування розробленого програмного забезпечення для оцінювання коментарів користувачів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Програмне забезпечення для оцінювання коментарів користувачів

Відгук користувача являє собою зворотній зв'язок або думку, яку надають користувачі про продукт, послугу чи досвід взаємодії з компанією. Ці відгуки допомагають компаніям зрозуміти потреби своїх клієнтів, визначити сильні та слабкі сторони продукту або послуги, а також приймати рішення щодо вдосконалення на основі отриманої інформації. Відгуки можуть бути надані у вигляді оцінок, коментарів, опитувань або через інші канали зв'язку, такі як соціальні мережі чи служби підтримки [1]-[6].

Відгуки користувачів мають значення для будь-якого бізнесу чи організації, але якщо не аналізувати такі відгуки, це призведе до марнування зусиль і може негативно позначитися на власнику компанії. Якісний збір відгуків і їхній аналіз, орієнтовані на вас, дозволяють в режимі реального часу оцінити ставлення користувачів до вашого продукту або послуги. Аналізуючи коментарі, ви зможете краще розуміти їхні потреби та вподобання, визначати сфери, які потребують вдосконалення, і приймати рішення, спираючись на ці дані, щоб досягти успіху в бізнесі [1]-[6].

Проаналізуємо типи зворотного зв'язку від користувачів. Існує багато різновидів відгуків від користувачів, таких як опитування, оцінки, відгуки, комунікація з клієнтами, а також коментарі та згадки в соціальних мережах. Кожен із цих джерел надає унікальну інформацію, яка може бути корисною в різних контекстах [1]-[6].

Сесійні записи та інші інструменти пасивного збору даних дозволяють отримувати інформацію без прямого запиту. Ці інструменти надають важливі дані про поведінку користувачів і їхні інтереси, відстежуючи дії, що виконуються на вебсайті чи в вебзастосунку. Проте варто зазначити, що такий тип зворотного зв'язку не завжди точно передає емоційний стан працівників, тому доцільно також збирати думки безпосередньо від персоналу [1]-[6].

Коли мова йде про аналіз якісних даних, огляд має включати виявлення зручностей та недоліків, об'єднання інформації від багатьох користувачів для вивчення їхньої поведінки та уподобань. Основною перевагою такого підходу є те, що команди можуть отримувати важливі дані без необхідності ставити запитання безпосередньо працівникам [1]-[6].

У програмному забезпеченні для підтримки опитувань, постійне збирання відгуків користувачів та взаємодія з клієнтами є активною формою зворотного зв'язку, де співробітники можуть ділитися власними думками та пропозиціями. Такий підхід дає можливість отримати більш детальну та специфічну інформацію про рівень задоволеності користувачів та проблемні аспекти, хоча він може призвести до зниження кількості відповідей [1]-[6].

Щодо вебсайтів, менеджери продуктів зазвичай орієнтуються на три основні способи збору зворотного зв'язку від користувачів. Вбудовані у програму опитування можна запускати на різних етапах взаємодії користувача, наприклад, під час завершення завдання або після впровадження нового продукту. Регулярні опитування та аналіз відгуків стають ефективнішими, якщо програма підтримує інтерактивні опитування, що відбуваються безпосередньо в додатку, замість того, щоб проводити їх через електронну пошту або анкети, де користувачі залишають зразки чи огляди своєї роботи. Найкращими форматами опитувань для отримання чіткої картини є такі методи, як оцінка якості обслуговування клієнтів, індекс лояльності клієнтів, і шкала оцінки досвіду користувача. Ці опитування слід регулярно проводити та відстежувати з певною періодичністю, деякі опитування для отримання відгуків користувачів можуть бути застосовані для оцінки впливу змін продукту, незалежно від того, коли ці зміни впроваджуються [1]-[6].

Користувачі завжди взаємодіють з контентом або використовують його. Програмні засоби дозволяють користувачам залишати відгуки про свій досвід навчання в будь-який зручний для них час, а не лише під час запропонованих тестів, тим самим забезпечуючи зворотний зв'язок, який не

потребує обов'язкового залучення. Відповідні програмні модулі можна легко вбудувати у вебсайт, щоб збирати певні типи відгуків у режимі реального часу. Наприклад, коротке опитування може запитати користувачів про їхнє враження від нового продукту, тоді як інші формати можуть допомогти збирати відгуки про поточні події чи зміни [1]-[6].

Також публікації та обговорення можуть стати джерелом корисних відгуків, особливо коли їх коментують та поширюють. Розглядаючи проблеми та питання, які виникають у користувачів, можна виявити спільні труднощі та області для вдосконалення вашого продукту [1]-[6].

Після того як було зібрано достатню кількість відгуків від користувачів, можна переходити до проведення їхнього аналізу. Проаналізуємо методи, які допоможуть отримати максимальну вигоду від опитувань з відгуками [1]-[7].

Локальний аналіз передбачає збір, дослідження та представлення даних. Ви можете розділити огляд на категорії, такі як зручність використання, обслуговування клієнтів, ціни тощо. Це дозволить визначити загальні проблеми або аспекти, де рівень задоволеності співробітників є найвищим [1]-[7].

Аналіз настроїв — це оцінка реакцій користувачів на основі їхніх відповідей. За допомогою інструментів обробки природної мови можна оцінити, чи є відгуки позитивними, негативними або нейтральними. Це дає змогу з'ясувати загальний настрій користувачів та виявити проблемні моменти чи можливості для покращень [1]-[7].

Аналіз тенденцій передбачає збір відгуків протягом певного часу для виявлення змін у задоволеності. Це допомагає зрозуміти вплив змін у продукті або послугі на досвід користувачів і виявити нові функції, що можуть зміцнити вашу позицію на ринку [1]-[7].

Аналіз розривів використовує відгуки для виявлення невідповідностей між фактичною продуктивністю продукту та очікуваннями користувачів. Це

дозволяє визначити, де продукт не відповідає вимогам, і поставити у пріоритет покращення на основі потреб клієнтів [1]-[7].

Порівняльний аналіз включає оцінку відгуків про ваш продукт у порівнянні з відгуками про конкурентів. Це допомагає зрозуміти, як ви виглядаєте на фоні інших рішень на ринку, і знайти сфери для здобуття конкурентної переваги [1]-[7].

Аналіз відгуків користувачів не є одноразовим заходом, а постійним процесом. Регулярний збір та обробка відгуків дозволяє підтримувати актуальне розуміння потреб і очікувань користувачів, допомагаючи ухвалювати обґрунтовані рішення для підвищення їх задоволення продуктом. Для цього важливо використовувати надійну систему, яка здатна збирати та аналізувати відгуки з різних аспектів. Поєднання активного та пасивного моніторингу дає можливість організаціям отримувати всебічне уявлення про користувачів і приймати рішення на основі даних, спрямовані на безперервне вдосконалення [1]-[7].

Процес аналізу відгуків може здаватися складним, проте при належній організації він стає керованим і навіть цікавим. Етапи включають збір даних, їх класифікацію, аналіз думок, виявлення тенденцій і можливостей, а також формування гіпотез і надання рекомендацій на основі отриманих результатів. Важливо залучати до процесу різних співробітників, оскільки різноманітні погляди можуть надати нові інноваційні рішення [1]-[7].

Для максимальної ефективності аналізу відгуків слід дотримуватися певних практик. Регулярний моніторинг і аналіз отриманих даних дозволяє постійно бути в курсі змін потреб користувачів. Важливо підходити до процесу з відкритістю і використовувати різні методи тестування для кращого розуміння. Однак аналіз – це лише перший крок, за яким слід діяти відповідно до отриманих результатів, що допоможе постійно вдосконалюватися і відповідати очікуванням користувачів [1]-[7].

Щоб залишатися обізнаними про думки та досвід користувачів, необхідно регулярно збирати та аналізувати їх відгуки з використанням

спеціалізованого програмного забезпечення. Постійний аналіз дозволяє уникнути прогалин і не втратити важливу інформацію [1]-[7].

Важливо підходити до нових ідей без упереджень. Уникнення суб'єктивних суджень допомагає отримати об'єктивне розуміння. Для ширшого спектра поглядів варто залучати різноманітні групи зацікавлених сторін, що забезпечує різноманітність думок [1]-[7].

Різні типи відповідей можуть потребувати застосування різних методів аналізу. Поєднання кількісних і якісних підходів допомагає отримати більш повну картину про сприйняття користувачів [1]-[7].

Результати, отримані під час експериментів, варто використовувати для внесення змін, відстежуючи при цьому вплив цих змін. Важливо зберігати відкриту комунікацію, повідомляючи про оновлення на основі відгуків користувачів, що підвищить довіру та зацікавленість співробітників [1]-[7].

Включення даних користувачів у тестування допомагає ухвалювати обґрунтовані рішення. Розуміння потреб і побажань користувачів стимулює вдосконалення, що підвищує задоволеність і лояльність. Думку співробітників слід вважати важливим фактором для розвитку продукту або послуги, адже вона дозволяє підтримувати безпосередній зв'язок і забезпечує постійне покращення [1]-[7].

Пріоритет слід віддавати аналізу відгуків працівників для постійного вдосконалення та розвитку стратегій. Тому аналіз відгуків співробітників допоможе оцінити позитивний вплив на бізнес [1]-[7].

Основна мета роботи з відгуками користувачів полягає у використанні їх як цінний інструмент для ухвалення обґрунтованих рішень. Глибоке розуміння потреб і побажань користувачів дозволяє стимулювати вдосконалення та зміни, що призводять до підвищення задоволеності та лояльності. Відгуки користувачів слід розглядати як важливий елемент процесу ухвалення рішень, оскільки вони забезпечують безпосередній зв'язок з клієнтами і сприяють постійному розвитку продукту чи послуги [1]-[7].

Збір відгуків користувачів надає цінну інформацію, яка дозволяє налагодити комунікацію з клієнтами та зрозуміти, де досягаються або не досягаються поставлені цілі [1]-[7]. Проте, через складність мови та велику кількість думок, обробка відповідей у великому обсязі може бути складною [1]-[7].

Ефективний аналіз відгуків користувачів дозволяє перетворити ці відомості на дослідження, спрямоване на покращення продукту чи вебсайту підприємства [1]-[7].

Існують стратегії обробки та аналізу відгуків користувачів. Обробка відгуків користувачів із залученням різних відділів та людей є складним процесом, і результати можуть бути важко інтерпретувати [1]-[7].

Однак розподіл даних на легкі для сприйняття статистичні показники допомагає перетворити проблему на конкретні дії, які ведуть до поліпшення взаємодії з клієнтами та поглибленого аналізу, доступного для команди.

Проаналізуємо процес впорядкованого збору та аналізу відгуків та інших коментарів користувачів [1]-[7].

Спочатку необхідно зібрати всю інформацію в одному місці для подальшого глибшого аналізу. Відгуки надходять з різних джерел, таких як соціальні мережі, опитування та інші канали. Зосередження цих даних в одному місці полегшує технічний аналіз і дає чітке уявлення про настрої користувачів. Це також сприяє обміну інформацією між організаціями та різними зацікавленими сторонами [1]-[7].

Збираючи відгуки, можна інтегрувати аналітичні інструменти з комунікаційними платформами для легшого збору відповідей і їх передачі до відповідних команд [1]-[7].

Як було зазначено раніше, важливо визначати пріоритети при аналізі відгуків користувачів за допомогою відповідного програмного забезпечення. Однак, скласти чіткий план дій буває непросто через велику кількість думок та різних точок зору. Часто трапляються суперечливі або занадто суб'єктивні відповіді, що ускладнює їх впровадження [1]-[7].

Щоб не втратити з поля зору головні завдання, варто використовувати шкалу оцінки значущості тем на основі відгуків користувачів. Це допомагає розрізнити, які питання потребують негайного вирішення, а які можуть бути відкладені на пізніше [1]-[7].

Важливим етапом аналізу відгуків користувачів є поділ інформації на категорії. Це момент, коли дані стають структурованими, і з'являються ключові висновки [1]-[7].

Обирати категорії для фільтрації відгуків слід відповідно до пріоритетів компанії та типу отриманих відгуків. Під час вибору відповідних класифікацій слід пам'ятати, яка кількість інформації дійсно необхідна. Уникайте зосередження на показниках, що не мають реальної цінності, і більше уваги приділяйте ідеям, які мотивують, навчають та вказують на шляхи покращення продукту та розвитку бізнесу [1]-[7].

Кодування відгуків та коментарів допомагає інтегрувати та розбити на категорії ідеї для легшого перегляду системи та розробки планів дій. Цей процес дозволяє розділити зібрані ідеї на відповідні категорії або кількісні показники [1]-[7].

На відміну від фільтрації, кодування передбачає додаткову сегментацію даних, що полегшує їх аналіз. Накладаючи категорії на відгуки, можна швидше виявити тенденції і зрозуміти проблеми користувачів з при роботі з програмним забезпеченням [1]-[7].

Після завершення попередніх етапів слід переглянути зібрані ідеї, розподілені за категоріями та кодами. Це дозволить виявити загальні тенденції і теми протягом всього користувацького досвіду [1]-[7].

Основи аналізу відгуків клієнтів з використанням спеціалізованого програмного забезпечення є ефективним інструментом для підвищення задоволеності співробітників. Регулярний збір відгуків через опитування в програмі дозволяє відстежувати дані в реальному часі [1]-[7].

Важливими елементами якісної аналітики клієнтів є інструменти візуалізації зворотного зв'язку, які забезпечують зрозуміле уявлення про зміни

у відгуках. Алгоритми машинного навчання допомагають прогнозувати задоволеність клієнтів, а інтеграція зворотного зв'язку в цикл продукту сприяє швидшому вдосконаленню, оперативному реагуванню та формуванню культури відкритого спілкування в організації [1]-[7].

Зворотний зв'язок використовують для визначення пріоритетів і покращення продуктів відповідно до потреб користувачів, що підвищує їхню цінність. Дані користувачів інтегрують у процес прийняття рішень для оптимізації продукту і бізнес-стратегії [1]-[7].

Важливо закривати цикл зворотного зв'язку, інформуючи співробітників про те, як їхній внесок вплинув на інновації продукту та лояльність. Відгуки клієнтів про продукт є важливими показниками еволюції та успіху [1]-[7].

Проаналізуємо причини важливості аналізу відгуки клієнтів. Відгуки забезпечують пряму інформацію про потреби клієнтів. Зворотний зв'язок є цінним джерелом інформації про думки клієнтів. Він забезпечує чітке розуміння їхніх потреб і уподобань. Ця інформація допомагає адаптувати продукти так, щоб вони відповідали або перевищували очікування користувачів. Вона виступає як постійно дорожня карта, яка оновлюється та спрямовує розвиток і вдосконалення продукту [1]-[7].

Визначення проблемних областей та можливостей для покращення: програмне забезпечення для аналізу відгуків допомагає виявити як дрібні, так і основні проблеми, з якими стикаються користувачі. Усунення цих проблем може перетворити критиків на прихильників, підвищуючи репутацію бренду та задоволеність співробітників [1]-[7].

Збільшення задоволеності клієнтів: реагування на відгуки клієнтів показує, що їхня думка має значення, що сприяє довірі і підвищує лояльність. Внесення змін на основі зворотного зв'язку демонструє, що продукт постійно вдосконалюється, що може підвищити задоволеність клієнтів та зберегти їхню прихильність [1]-[7].

Нові ідеї та інновації: відгуки клієнтів є джерелом нових ідей і концепцій, які можуть бути використані для розробки нових продуктів. Колективний інтелект користувачів може привести до створення інноваційних рішень, які відповідають потребам ринку [1]-[7].

Збільшення утримання співробітників: програмне забезпечення для аналізу відгуків дозволяє зрозуміти, що робить продукт привабливим для користувачів, а також чому деякі можуть відмовитися від нього. Постійне вдосконалення продукту на основі відгуків допомагає підтримувати його цінність і привабливість, що сприяє утриманню клієнтів і підвищенню позитивного спілкування та задоволеності працівників [1]-[7].

Отже, відгуки клієнтів та відповідне програмне забезпечення для їх аналізу є ключовим елементом успішного розвитку продуктів, допомагаючи зрозуміти потреби, вдосконалювати пропозиції та підтримувати високий рівень задоволеності та лояльності [1]-[7].

Важливим кроком у реалізації ідей на основі зворотного зв'язку є створення програмної системи для організованого і ефективного збору та аналізу відгуків та коментарів користувачів [1]-[7].

Традиційний підхід передбачає створення довгого опитування за допомогою спеціальних програм і надсилання його електронною поштою. Можливо, також буде корисно пропонувати винагороди клієнтам за участь у опитуваннях. Проте цей метод може бути не найефективнішим [1]-[7].

Замість традиційного підходу, можна використовувати програмні інструменти для збору зворотного зв'язку у вигляді відгуків користувачів, які дозволяють створювати і публікувати опитування безпосередньо в програмі. Це забезпечує своєчасне і точне отримання результатів, коли зворотний зв'язок є найбільш актуальним [1]-[7].

Варто використовувати спливаючі вікна для опитувань. Вони можуть з'являтися після завершення основних завдань або у разі важливих подій. Важливо, щоб ці спливаючі вікна не були нав'язливими і не відволікали тих, хто не бажає брати участь на цьому етапі [1]-[7].

Збір зворотного зв'язку в реальному часі: здатність програми збирати і реагувати на відгуки миттєво дозволяє швидко виявляти і вирішувати проблеми до того, як вони стануть масовими, що покращує задоволеність і лояльність користувачів [1]-[7].

Інтеграція зворотного зв'язку у користувацький досвід. Збір відгуків стає невід'ємною частиною взаємодії з користувачем, що підвищує значення зворотного зв'язку та сприяє культурі постійного вдосконалення [1]-[7].

Залучення користувачів у процес збору та аналізу відгуків допомагає не тільки отримувати цінну інформацію, але і підвищувати зацікавленість всіх учасників продукту [1]-[7].

Перетворення необроблених даних у корисну інформацію є важливим етапом аналізу відгуків клієнтів. Існують різні методи аналізу відгуків та коментарів користувачів. Сортування за характеристиками або функціями: ідеї класифікуються за окремими характеристиками, функціями чи аспектами роботи. Це допомагає визначити, які елементи функціонують добре, а які потребують вдосконалення [1]-[7].

Позитивні та негативні відгуки: відгуки розділяються на позитивні та негативні категорії для швидкого виявлення сильних та слабких сторін, що дозволяє визначити пріоритети для подальшої роботи [1]-[7].

Частотний аналіз: здійснюється пошук найпоширеніших тем чи запитань, що стосуються продукту. Збільшення кількості повторюваних концепцій допомагає виявити області для покращення [1]-[7].

Аналіз настроїв: використовуються інструменти для оцінки загального настрою після отримання відгуків, що допомагає зрозуміти загальні тенденції настроїв користувачів, особливо у великих даних [1]-[7].

Врахування бізнес-цілей: відгуки сортуються за пріоритетами, які відповідають основним бізнес-цілям та позиціонуванню продукту. Важливо встановити передумови для попередньої оцінки відгуків [1]-[7].

Вплив на користувачів: оцінюється, скільки користувачів виграють від запропонованих змін. Ідеї, що мають значний вплив на велику частину бази користувачів, отримують пріоритет [1]-[7].

Проведення глибокого аналізу якості та консультації з експертами: для отримання детальної інформації слід звернутися до фахівців. Короткі бесіди можуть виявити корисні деталі та усунути можливі недоліки [1]-[7]. Аналіз персонажів і історій користувачів: порівняти відгуки з конкретними персонажами або історіями користувачів допомагає зрозуміти проблеми з точки зору користувачів та поглиблює аналіз [1]-[7].

Зв'язок з реальними даними: порівняти поведінку користувачів з реальними даними про продуктивність. Наприклад, якщо користувачі скаржаться на певні аспекти, але показники активності вказують на зворотне, це може свідчити про проблеми в дизайні продукту [1]-[7].

Оцінка впливу змін: перед впровадженням змін слід виміряти ключові показники, а потім оцінити їхній вплив на основі відгуків. Це забезпечить, що рішення базуються на надійних даних і матимуть відчутний ефект [1]-[7].

Аналіз результатів: об'єднання результатів аналізу відгуків для формування пропозицій щодо оновлень або вдосконалень продукту, а також для нових розробок [1]-[7].

Використання відгуків користувачів для визначення пріоритетних задач і поточних потреб є ключовим етапом у розробці продукту. Це дозволяє зосередити зусилля на найбільш важливих аспектах, які відповідають потребам користувачів, подібно до планування у сфері архітектури. Це забезпечує орієнтацію на створення рішень, що найкраще відповідають реальним вимогам користувачів [1]-[7].

Аналізуючи відгуки, можна впроваджувати зміни з чіткою метою, що дозволяє адаптувати продукт відповідно до реальних потреб користувачів, а не тільки до власних припущень. Такий підхід допомагає не лише вдосконалити існуючі функції, але й розробити нові можливості [1]-[7].

Важливо забезпечити прозорість у процесі аналізу відгуків, щоб не упустити жодної важливої ідеї. Активно слухати і ділитися інформацією з командою зміцнює довіру і підтримує високий рівень залученості та лояльності клієнтів та користувачів [1]-[7].

Включення відгуків користувачів у процес прийняття рішень дозволяє не лише враховувати їхні побажання, але й активно залучати їх до процесу розробки наступних етапів продукту [1]-[7].

Визначено, що відгук користувача являє собою зворотній зв'язок або думку, яку надають користувачі про продукт, послугу чи досвід взаємодії з компанією. Відгуки та коментарі користувачів допомагають компаніям зрозуміти потреби своїх клієнтів, визначити сильні та слабкі сторони продукту або послуги, а також приймати рішення щодо вдосконалення на основі отриманої інформації. Відгуки можуть бути надані у вигляді оцінок, коментарів, опитувань або через інші канали зв'язку, такі як соціальні мережі чи служби підтримки. Проаналізовано процес аналізу відгуків користувачів за допомогою програмного забезпечення.

1.2 Аналіз програмного забезпечення для оцінювання коментарів користувачів

Проаналізуємо програмне забезпечення для оцінювання коментарів користувачів [8]-[15].

Програмне забезпечення Typeform (рис. 1.1) є простим у використанні, інтуїтивним інструментом без потреби програмування, який дозволяє створювати захопливі та інтерактивні опитування для збору відгуків. Його функціональні можливості та зручний дизайн допомагають створювати опитування з привабливим візуальним оформленням, що робить процес заповнення приємним для користувачів. Typeform ідеально підходить для опитування рівня задоволеності клієнтів, проведення маркетингових досліджень та інших завдань [8], [9], [13].

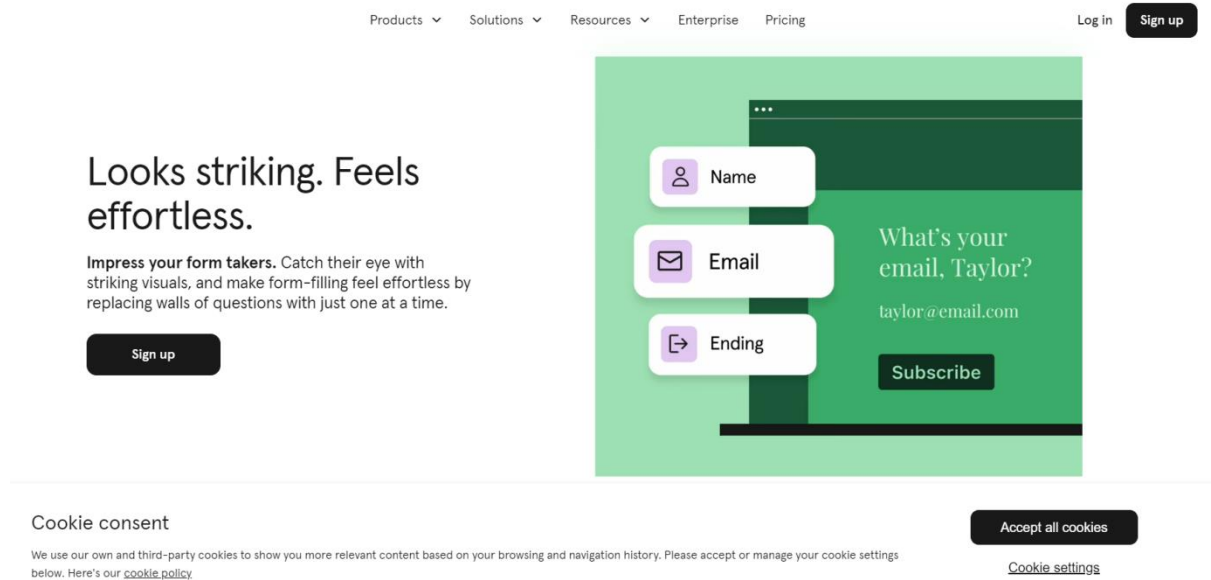


Рисунок 1.1 – Програмне забезпечення Typeform [13]

Цей інструмент виділяється серед інших рішень для збирання зворотного зв'язку завдяки своєму зручному інтерфейсу, гнучким налаштуванням і потужній аналітиці [8], [9], [13].

З Typeform можна легко створювати візуально привабливі опитування, які надають позитивний досвід як розробникам, так і користувачам. Його дизайн можливостей значно перевершує багато аналогів, пропонуючи гнучкі налаштування, від мультимедійних елементів до користувацьких стилів, що дозволяє робити унікальні форми на високому рівні [8], [9], [13].

Typeform також забезпечує високу ефективність заповнення завдяки різноманітним типам опитувань. Можливості аналізу в реальному часі й роздільний аналіз за запитаннями дозволяють творцям оперативно оптимізувати опитування. Крім того, велика кількість інтеграцій із популярними інструментами забезпечують легку адаптацію до існуючих робочих процесів для покращеного збору даних [8], [9], [13].

Typeform пропонує велику кількість готових шаблонів для створення опитувань та форм різної складності, що допомагає зекономити час і зусилля. Адаптивні додаткові запитання: інтерактивні форми можуть змінювати

запитання в залежності від попередніх відповідей, що дає можливість отримувати більш детальні й персоналізовані відгуки [8], [9], [13].

Привабливий дизайн: завдяки візуально досконалому оформленню форми виглядають привабливо й стимулюють користувачів активніше брати участь в опитуваннях. Функція логічні стрибки дозволяє створювати динамічні опитування, де запитання змінюються відповідно до відповідей респондентів, що підвищує якість зібраних даних [8], [9], [13].

Основними характеристиками програмного забезпечення Typeform є такі [8], [9], [13]:

- інтуїтивно зрозумілий інтерфейс: програмне забезпечення Typeform надає простий і зручний у використанні інтерфейс без потреби програмування. Користувачі можуть легко створювати форми за допомогою перетягування елементів, що не потребує технічних знань [8], [9], [13];

- адаптивний дизайн: формати автоматично адаптуються до різних пристроїв (комп'ютери, планшети, мобільні телефони), що забезпечує комфортне заповнення форм із будь-якого пристрою. Привабливий візуальний дизайн підвищує залученість користувачів до опитувань [8], [9], [13];

- функція логічних стрибків, яка забезпечує можливість створювати динамічні опитування, де наступні запитання залежать від відповідей респондентів. Ця функція дозволяє забезпечити персоналізований досвід, збираючи більш точні дані [8], [9], [13];

- колекції шаблонів: програмне забезпечення Typeform пропонує великий вибір попередньо створених шаблонів, які можна використовувати для різних цілей, таких як опитування, реєстраційні форми, оцінювання тощо. Шаблони адаптовані для різних галузей та бізнес-задач [8], [9], [13];

- адаптивні додаткові запитання: форми можуть змінюватися залежно від відповідей респондентів, що дозволяє збирати більш релевантні дані та забезпечує збільшення точності отриманих результатів і підвищує інтерес користувачів [8], [9], [13];

– інтеграції з іншими платформами: програмне забезпечення Typeform інтегрується з великою кількістю інструментів. Інтеграція полегшує управління зібраними даними та їх подальшу обробку [8], [9], [13];

– привабливі мультимедійні елементи: можливість додавання зображень, відео та інших мультимедійних елементів для покращення досвіду користувачів. Такі елементи роблять форми більш цікавими та інтерактивними для користувачів [8], [9], [13];

– аналіз і звітність у реальному часі: програмне забезпечення Typeform надає інструменти аналізу результатів в реальному часі дозволяють негайно отримувати інсайти з опитувань. Підтримується деталізований аналіз відповідей на кожне запитання [8], [9], [13];

– можливість проводити аналіз сильних сторін, слабких сторін, можливостей та загроз на основі результатів опитувань, що дозволяє оцінювати поточну ситуацію та знаходити шляхи для вдосконалення діяльності підприємства [8], [9], [13];

– аналіз показників завершеності: програмне забезпечення Typeform надає дані про рівень завершеності форм і опитувань, що дозволяє оцінити ефективність створених форм та визначити можливі проблеми у процесі заповнення [8], [9], [13];

– безпека даних: Typeform забезпечує високий рівень захисту даних користувачів за допомогою шифрування та відповідності міжнародним стандартам захисту інформації [8], [9], [13].

Серед недоліків програмного забезпечення Typeform можна відзначити обмежені можливості для аналітики даних. Хоча Typeform пропонує базовий аналіз даних і звіти, його можливості аналітики можуть бути недостатніми для великих компаній або організацій, які потребують детальнішого аналізу результатів. Безкоштовна версія програмного забезпечення Typeform пропонує лише базові функції та обмежену кількість відповідей. Для доступу до розширених функцій програмного забезпечення Typeform необхідно придбати платний план [8], [9], [13].

Програмне забезпечення SurveyMonkey (рис. 1.2) представляє собою платформу для проведення опитувань, яка допомагає збирати зворотний зв'язок від клієнтів і надає корисну аналітику. Завдяки простому процесу налаштування, користувач може легко створювати індивідуалізовані опитування та поширювати їх через різні канали. Інструмент також оснащений широким спектром функцій для аналізу отриманих даних [8]-[10], [14].

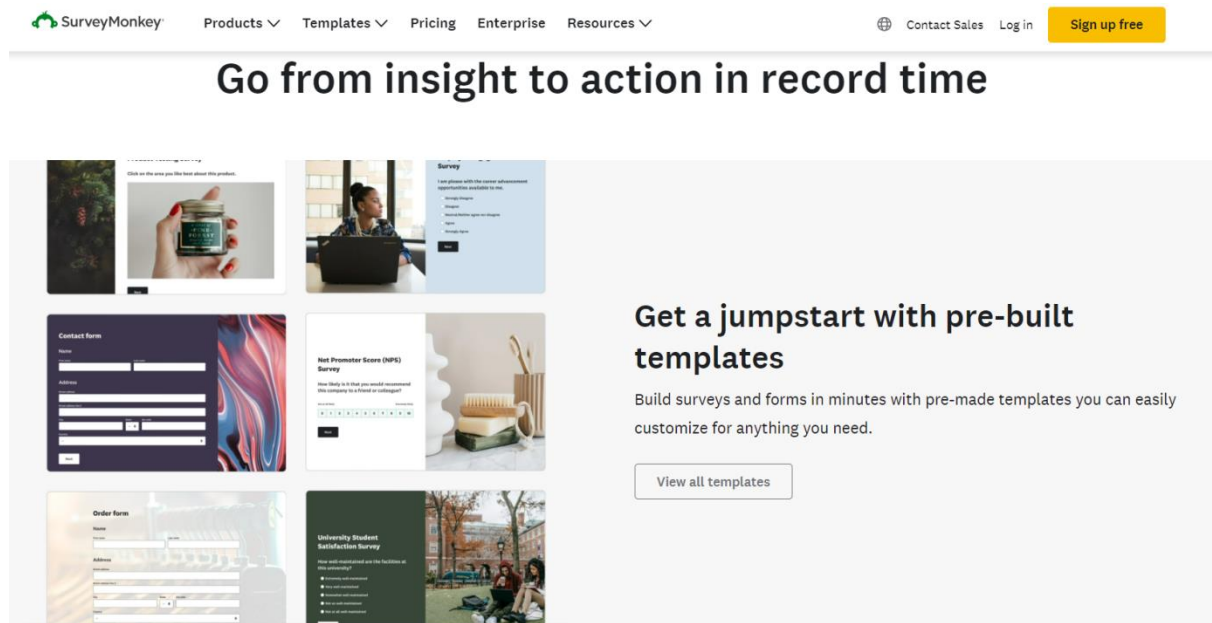


Рисунок 1.2 – Програмне забезпечення SurveyMonkey [14]

Програмне забезпечення SurveyMonkey є одним із провідних рішень для збору відгуків завдяки своєму інтуїтивно зрозумілому інтерфейсу, який підходить для користувачів будь-якого рівня. Опитування можна створювати з різними типами запитань, такими як вибіркові відповіді, відкриті питання та шкали. Для тих, хто не має часу на розробку власних форм, доступна бібліотека шаблонів [8]-[10], [14].

Користувачі платних версій можуть створювати детальні звіти для глибшого аналізу відповідей і отримання важливих висновків. Існує також можливість автоматизації опитувань, що дозволяє швидше обробляти дані, відфільтровувати відповіді та автоматично заповнювати запитання. Однак ці аналітичні функції не доступні у безкоштовній версії [8]-[10], [14].

Програмне забезпечення SurveyMonkey підтримує різні інтеграції, але деякі важливі інтеграції, такі як Google Tag Manager для відстеження конверсій, Microsoft Teams для співпраці та Canva для створення візуалізацій, залишаються недоступними [8]-[10], [14].

Програмне забезпечення SurveyMonkey надає розширені можливості створення та розповсюдження опитувань із фокусом на аналітику та отримання практичних інсайтів [8]-[10], [14].

Інструменти SurveyMonkey допомагають збирати відгуки, аналізувати результати та перетворювати їх на корисні інсайти, що сприяють інноваціям і зростанню. Завдяки досвіду та впровадженню інструментів на базі штучного інтелекту, ця програмна платформа робить процес створення, запуску та аналізу опитувань простим як для новачків, так і для професійних дослідників ринку. Програмне забезпечення SurveyMonkey пропонує різноманітні тарифні плани для компаній будь-якого розміру, включаючи плани для команд, що полегшують співпрацю [8]-[10], [14].

Програмне забезпечення SurveyMonkey надає великий вибір готових запитань, розроблених дослідниками, які можна додати до ваших опитувань для швидкого та зручного створення. Функції логіки опитувань дозволяють налаштовувати опитування для кращої взаємодії з респондентами. SurveyMonkey пропонує чотири типи логіки: логіка пропуску запитань, яка спрямовує респондентів залежно від попередніх відповідей; логіка пропуску сторінок, що пропускає певні сторінки, оптимізуючи шлях опитування; розширене розгалуження надає індивідуальний досвід опитування на основі умов; використання попередніх відповідей для персоналізації наступних запитань [8]-[10], [14].

Програмне забезпечення SurveyMonkey надає можливості надсилання респондентам нагадування про участь в опитуванні, що допомагає підвищити кількість відповідей [8]-[10], [14].

Платформа підтримує інтеграцію з різними популярними сервісами та програмами [8]-[10], [14].

Програмне забезпечення SurveyMonkey має такі особливості та характеристики [8]-[10], [14]:

- зручний інтерфейс: програмне забезпечення SurveyMonkey має інтуїтивно зрозумілий інтерфейс, що дозволяє користувачам легко створювати опитування без глибоких технічних знань [8]-[10], [14];

- різноманітність типів запитань: SurveyMonkey підтримує широкий вибір типів запитань, включаючи множинні варіанти відповідей, шкали Лайкерта, відкриті запитання тощо, що дозволяє створювати персоналізовані опитування для будь-яких потреб [8]-[10], [14];

- банк запитань: програмне забезпечення SurveyMonkey надає доступ до великої бібліотеки готових запитань, створених дослідниками, що полегшує процес розробки опитувань [8]-[10], [14];

- логіка опитування: SurveyMonkey пропонує різноманітні інструменти логіки, як-от пропуски запитань і сторінок, розгалуження за умовами, що дозволяє створювати персоналізований досвід для респондентів і збирати більш релевантні дані [8]-[10], [14];

- нагадування про опитування: інструменти для автоматичних нагадувань допомагають підвищити відсоток відповідей, надсилаючи учасникам нагадування про необхідність завершити опитування [8]-[10], [14];

- аналітика та звіти: SurveyMonkey надає потужні інструменти для аналізу зібраних даних, дозволяючи генерувати звіти, які допомагають краще розуміти результати опитувань і робити значущі висновки [8]-[10], [14];

- масштабованість: програмне забезпечення SurveyMonkey підходить як для малого бізнесу, так і для великих корпорацій, пропонуючи тарифні плани, що відповідають різним потребам [8]-[10], [14];

- інтеграція: SurveyMonkey підтримує інтеграцію з більш ніж 100 популярними сервісами, включаючи Salesforce, Hubspot, Mailchimp, Google Drive тощо, що дозволяє легко впроваджувати дані з опитувань у вже існуючі робочі процеси [8]-[10], [14];

– безпека даних: програмне забезпечення SurveyMonkey забезпечує високий рівень захисту даних, дотримуючись стандартів та інших вимог безпеки [8]-[10], [14];

– мобільна доступність: опитування, створені в SurveyMonkey, повністю адаптовані до мобільних пристроїв, що дозволяє респондентам відповідати з будь-якого місця та в будь-який час [8]-[10], [14].

Серед недоліків програмного забезпечення SurveyMonkey можна відзначити обмежені можливості для аналітики даних. Хоча SurveyMonkey пропонує базовий аналіз даних і звіти, його можливості аналітики можуть бути недостатніми для великих компаній або організацій, які потребують детальнішого аналізу результатів. Безкоштовна версія програмного забезпечення SurveyMonkey пропонує лише базові функції та обмежену кількість відповідей. Для доступу до розширених функцій програмного забезпечення SurveyMonkey необхідно придбати платний план [8]-[10], [14].

Програмне забезпечення Qualaroo (рис. 1.3) є однією з провідних платформ для збору зворотного зв'язку та оцінки задоволеності клієнтів, що вирізняється своїми передовими функціями, як-от аналіз настроїв, які допомагають отримати більш якісні й докладні відгуки [8], [9], [15].

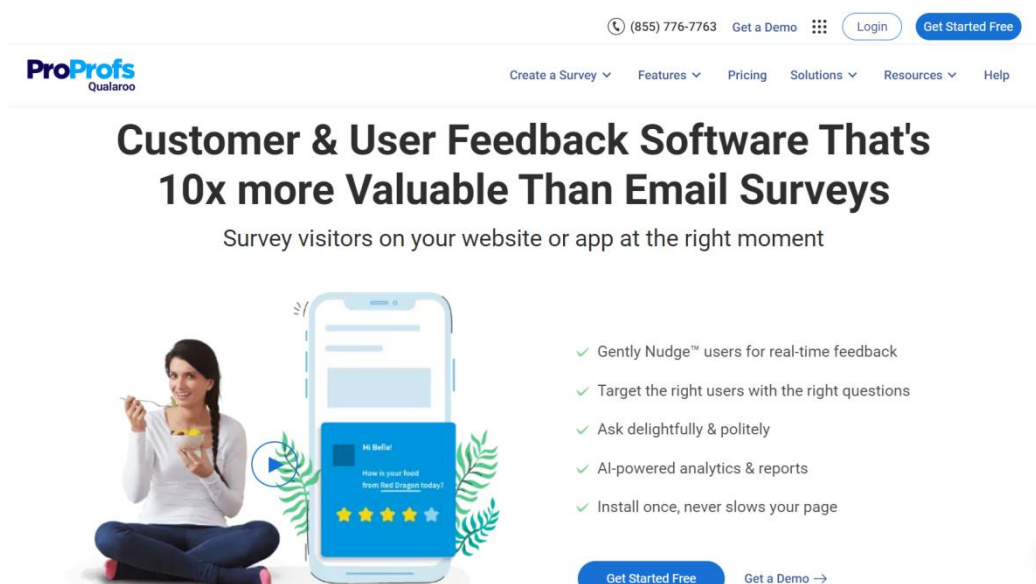


Рисунок 1.3 – Програмне забезпечення Qualaroo [15]

Програмне забезпечення Qualaroo пропонує широкий набір інструментів для взаємодії з клієнтами, включаючи спеціальні шаблони для опитувань та панелі звітів із можливістю відстеження і аналізу настроїв. Завдяки простому інтерфейсу та можливості збору відгуків через кілька каналів, платформа дозволяє створювати різноманітні типи опитувань, варіанти вибору й відкриті запитання [8], [9], [15].

Програмне забезпечення Qualaroo надає розширені налаштування націлювання, що дозволяють створювати опитування на основі місцезнаходження, поведінки та частоти відвідувань користувачів. Інструменти штучного інтелекту Qualaroo дають змогу глибше аналізувати емоції респондентів, мінімізуючи потребу у ручній інтерпретації результатів. Однак, хоча платформа має інтеграції з іншими сервісами, їхній вибір не такий широкий, як у деяких конкурентів [8], [9], [15].

Програмне забезпечення Qualaroo забезпечує просунуті інструменти аналізу настроїв та відгуків, спрямовані на розуміння поведінки й вподобань користувачів [8], [9], [15].

Qualaroo забезпечує аналітику на основі штучного інтелекту для аналізу настроїв і перетворення текстових відповідей на структуровані дані, дозволяючи легко відстежувати ключові слова й емоції клієнтів [8], [9], [15]. Програмне забезпечення Qualaroo не впливає на продуктивність сайту, й дозволяє додавати нові запитання без затримок і без допомоги розробників. Qualaroo надає готові шаблони опитувань, які можна налаштовувати під бренд завдяки індивідуальним шрифтам, кольорам і логотипам [8], [9], [15].

Програмне забезпечення Qualaroo має такі особливості [8], [9], [15]:

- аналіз настроїв на основі штучного інтелекту: використання технологій штучного інтелекту, для автоматичного аналізу тексту відгуків. Інструменти аналізу настроїв програмного забезпечення Qualaroo дозволяють перетворювати текстові відповіді на структуровані дані, визначаючи емоційне забарвлення та ключові слова [8], [9], [15];

– інтерактивний і безперебійний інтерфейс: Qualaroo забезпечує плавну взаємодію користувача без впливу на продуктивність сайту. Qualaroo легко додається на вебсайт із мінімальними технічними вимогами та без необхідності в залученні розробників [8], [9], [15];

– розширені параметри націлювання: опитування можна налаштовувати відповідно до поведінки користувачів, місця їхнього перебування або частоти відвідувань. Ці можливості програмного забезпечення Qualaroo націлювання допомагають отримати більш точний і персоналізований зворотний зв'язок [8], [9], [15];

– професійні шаблони для опитувань: велика бібліотека готових шаблонів для різних галузей і типів опитувань. Можливість налаштування зовнішнього вигляду шаблонів (шрифти, кольори, логотипи), що дозволяє адаптувати їх під ваш бренд [8], [9], [15];

– інтеграції з іншими платформами: підтримка інтеграцій із різними програмними платформами для спрощення роботи з даними. Інтеграції полегшують аналіз даних та обробку відгуків у поєднанні з іншими інструментами [8], [9], [15];

– багатоканальні можливості збору відгуків: можливість збирати зворотний зв'язок через різні канали, включаючи вебсайти, мобільні додатки та електронну пошту [8], [9], [15];

– розширені типи запитань: підтримка різних форматів запитань, таких як індекс лояльності, відкриті запитання, множинний вибір та рейтингові шкали [8], [9], [15];

– аналітика та звіти: звіти з детальною аналітикою результатів опитувань. Інструменти для аналізу настроїв і поведінки користувачів дозволяють робити висновки на основі відповідей і оперативно коригувати стратегії [8], [9], [15].

Серед недоліків програмного забезпечення Qualaroo можна відзначити обмежені можливості для аналітики даних. Хоча Qualaroo пропонує базовий аналіз даних і звіти, його можливості аналітики можуть бути недостатніми для

великих компаній або організацій, які потребують детальнішого аналізу результатів. Безкоштовна версія програмного забезпечення Qualaroo пропонує лише базові функції та обмежену кількість відповідей. Для доступу до розширених функцій програмного забезпечення Qualaroo необхідно придбати платний план [8], [9], [15].

Порівняльну характеристику програмного забезпечення для оцінювання коментарів користувачів наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння програмного забезпечення для оцінювання коментарів користувачів

Критерій порівняння	Typeform [13]	SurveyMonkey [14]	Qualaroo [15]
Адаптивний та зручний дизайн	+	+-	+-
Мобільна доступність	+-	+	+-
Використання засобів штучного інтелекту	-	+	+
Колекції шаблонів	+	+-	-
Інструменти аналізу	+-	+	+
Інтеграції з іншими платформами	+-	+	+-

За результатами проведеного аналізу можна зробити висновок, що у наш час існує досить багато програмних засобів для оцінювання коментарів користувачів. Проте деякі програмні засоби для оцінювання коментарів користувачів можуть не завжди правильно інтерпретувати контекст коментарів, що обумовлено використанням неефективних методів штучного інтелекту та розпізнавальних моделей, що може призвести до неправильної оцінки настрою або значущості коментарів. Деяке програмне забезпечення може мати обмежені можливості для точного аналізу коментарів, написаних на різних мовах або діалектах, що особливо актуально для глобальних

компаній, які працюють на ринках різних країн. Для того щоб деяке програмне забезпечення для оцінювання коментарів користувачів працювало ефективно, може знадобитися значний обсяг часу і ресурсів на його налаштування під специфічні потреби бізнесу, що включає адаптацію моделей і інтеграцію з іншими системами. Тому актуальною є розробка програмного забезпечення для оцінювання коментарів користувачів.

При розробці програмного забезпечення для оцінювання коментарів користувачів необхідно забезпечити такі функціональні вимоги:

- можливість підтримки процесів авторизації та реєстрації користувачів;
- можливість пошуку розважальних заходів;
- підтримка можливості ознайомлення з інформацією про розважальні заходи;
- можливість коментування (залишення відгуків) про розважальні заходи;
- можливість перегляду власних коментарів про розважальні заходи;
- підтримка можливості перегляду та редагування інформації про профіль користувача;
- можливість оцінювання відгуків користувачів (класифікація їх на позитивні та негативні) для подальшого аналізу розважальних заходів;
- підтримка можливості керування інформацією про розважальні заходи (функція адміністратора);
- можливість збереження результатів оцінювання коментарів користувачів.

1.3 Висновки за розділом 1

Визначено, що відгук користувача являє собою зворотній зв'язок або думку, яку надають користувачі про продукт, послугу чи досвід взаємодії з компанією. Відгуки та коментарі користувачів допомагають компаніям

зрозуміти потреби своїх клієнтів, визначити сильні та слабкі сторони продукту або послуги, а також приймати рішення щодо вдосконалення на основі отриманої інформації. Відгуки можуть бути надані у вигляді оцінок, коментарів, опитувань або через інші канали зв'язку, такі як соціальні мережі чи служби підтримки. Проаналізовано процес аналізу відгуків користувачів за допомогою програмного забезпечення.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для оцінювання коментарів користувачів. Проте деякі програмні засоби для оцінювання коментарів користувачів можуть не завжди правильно інтерпретувати контекст коментарів, що обумовлено використанням неефективних методів штучного інтелекту та розпізнавальних моделей, що може призвести до неправильної оцінки настрою або значущості коментарів. Деяке програмне забезпечення може мати обмежені можливості для точного аналізу коментарів, написаних на різних мовах або діалектах, що особливо актуально для глобальних компаній, які працюють на ринках різних країн. Для того щоб деяке програмне забезпечення для оцінювання коментарів користувачів працювало ефективно, може знадобитися значний обсяг часу і ресурсів на його налаштування під специфічні потреби бізнесу, що включає адаптацію моделей і інтеграцію з іншими системами. Тому актуальною є розробка програмного забезпечення для оцінювання коментарів користувачів.

Сформульовано функціональні вимоги до програмного забезпечення для оцінювання коментарів користувачів.

2 МАТЕРІАЛИ І МЕТОДИ

2.1 Вибір мови програмування

При розробці програмного забезпечення для оцінювання коментарів користувачів було використано мову програмування Python [16], [17].

Python – це високорівнева мова програмування загального призначення, яка вирізняється простотою, зручністю читання коду та широкою функціональністю. Її використовують для розробки вебзастосунків, аналізу даних, машинного навчання, автоматизації, наукових обчислень і багатьох інших завдань [16], [17].

Python має величезну стандартну бібліотеку, яка дозволяє швидко вирішувати різні задачі, включаючи роботу з файлами, мережами, базами даних та іншими ресурсами. Python працює на різних операційних системах (Windows, macOS, Linux), що робить його дуже універсальним [16], [17].

Python вирізняється кількома характеристиками, які зробили його популярним вибором серед мов програмування:

- зменшення бар'єру входу: спрощений синтаксис мови дозволяє працювати зі складними системами з різних галузей та забезпечує ясність і взаємозв'язок між елементами коду [16], [17];

- швидкість розробки: Python не тільки стрімко розвивається як мова, але й забезпечує швидке створення прототипів і реалізацію ідей, що підвищує ефективність роботи розробників [16], [17];

- простота навчання: Python легко освоїти, що спрощує процес роботи з кодом. Це дає можливість швидко створювати відкриті рішення та працювати над новими проєктами [16], [17];

- інтерпретована мова: Python є інтерпретованою мовою, що робить його підходящим для швидкого прототипування та тестування [16], [17];

- асинхронне програмування: хоча для деяких воно може не бути основним аспектом, Python підтримує асинхронне кодування, що дозволяє паралельне виконання кількох процесів [16], [17];

– візуалізація даних: Python пропонує інструменти для відображення інформації у зрозумілому форматі за допомогою графіків і діаграм. Бібліотеки, такі як Matplotlib, дають змогу візуалізувати дані та зробити їх більш зрозумілими [16], [17];

– об'єктно-орієнтоване програмування: ця парадигма в Python легко реалізується, забезпечуючи структурування коду в класи й об'єкти, що ізолює помилки в межах окремих компонентів. Це спрощує процес розробки й робить його більш економічним та менш трудомістким [16], [17].

Обґрунтування вибору мови програмування для розробки програмного забезпечення для оцінювання коментарів користувачів наведено у таблиці 2.1.

Таблиця 2.1 – Обґрунтування вибору мови програмування для розробки програмного забезпечення для оцінювання коментарів користувачів

Критерій порівняння мов програмування	Мова програмування		
	Python	C#	C++
Інтуїтивно зрозумілий синтаксис	+	+-	+-
Засоби аналізу даних	+	+-	-
Спільнота і ресурси	+	+	+-
Зручність для створення ПЗ для оцінювання коментарів користувачів	+	-	-
Бібліотеки для машинного навчання	+	+-	+-

Отже, для реалізації програмного забезпечення для оцінювання коментарів користувачів обрано мову програмування Python, яка має широкий спектр бібліотек машинного навчання, зокрема Scikit-learn і TensorFlow, які можуть спростити процес аналізу коментарів. Крім того, Python відомий своєю простотою та зрозумілістю коду, що дозволяє швидше розробляти і

підтримувати програми, його синтаксис є інтуїтивно зрозумілим, що спрощує розробку і знижує вартість підтримки.

2.2 Вибір фреймворку для оцінювання коментарів користувачів

В якості фреймворку для створення програмного забезпечення для оцінювання коментарів користувачів було обрано Django [18], [19].

Фреймворк Django є це високорівневим фреймворком для веброзробки на мові Python, що спрощує створення складних та функціональних вебзастосунків [18], [19].

Django дозволяє розробникам швидко створювати веб-додатки завдяки вбудованим компонентам, таким як адміністрування, аутентифікація користувачів, управління сесіями та інші інструменти. Він надає вбудовані функції, що сприяють повторному використанню коду та легкій зміні окремих його частин, що значно спрощує розробку. Django також добре інтегрується з сучасними системами управління та забезпечує потужні інструменти для адміністрування й підтримки [18], [19].

Django пропонує вбудовані рішення для захисту від типових загроз, таких як SQL-ін'єкції, підробка міжсайтових запитів і підтримує розробку програм, що можуть масштабуватися та працювати під високим навантаженням [18], [19].

Фреймворк Django має велику спільноту розробників та детальну документацію, що допомагає швидко розв'язувати проблеми та впроваджувати нові функції [18], [19].

Цей фреймворк популярний серед розробників завдяки своїй простоті, надійності та функціональності, що робить його вибором для створення вебсервісів будь-якого рівня складності [18], [19].

Обґрунтування вибору фреймворку для створення програмного забезпечення для оцінювання коментарів користувачів наведено у таблиці 2.2.

Таблиця 2.2 – Обґрунтування вибору фреймворку для створення програмного забезпечення для оцінювання коментарів користувачів

Критерій порівняння фреймворків	Фреймворки		
	Django	FastAPI	Flask
Вбудоване тестування	+	+-	+-
Масштабованість	+	-	+
Модульність	+	+-	+-
Інтеграція з базами даних	+	+-	+-
Потужні інструменти для обробки форм	+	+	+-

Таким чином, для створення програмного забезпечення для оцінювання коментарів користувачів обрано фреймворк Django, який надає зручний механізм для роботи з формами, що спрощує обробку і валідацію даних, зокрема коментарів користувачів. Це дозволяє легко створювати і обробляти форми для збору зворотного зв'язку. Крім того, Django включає в себе потужні інструменти для тестування, що дозволяє створювати автоматизовані тести для перевірки функціональності програми. Це допомагає забезпечити високу якість коду та правильність роботи всіх функцій.

2.3 Розробка моделі взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів

З метою можливості розробки програмного забезпечення для оцінювання коментарів користувачів створимо відповідну модель у вигляді діаграми прецедентів.

При розробленні моделі взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів враховуємо те, що

користувачі програми мають можливості авторизації та реєстрації, пошуку розважальних заходів, ознайомлення з інформацією про розважальні заходи, коментування (залишення відгуків) про розважальні заходи, перегляду власних коментарів про розважальні заходи, перегляду та редагування інформації про профіль користувача, оцінювання відгуків користувачів (класифікація їх на позитивні та негативні) для подальшого аналізу розважальних заходів, керування інформацією про розважальні заходи (функція адміністратора), збереження результатів оцінювання коментарів користувачів.

Основний успішний сценарій користувача при роботі з програмним забезпеченням для оцінювання коментарів користувачів можна описати таким чином:

- користувач програмного забезпечення для оцінювання коментарів користувачів авторизується;
- після авторизації користувача сервер програмного забезпечення для оцінювання коментарів користувачів у відповідь повертає основну сторінку вебсайту програмного забезпечення для оцінювання коментарів користувачів;
- користувач бачить відповідну сторінку програмного забезпечення для оцінювання коментарів користувачів та шукає потрібний розважальний захід;
- користувач читає опис розважального заходу та залишає коментар про нього;
- користувач переглядає власні коментарі про розважальні заходи;
- користувач переглядає та за необхідності редагує інформацію про власний профіль;
- користувач завершує роботу з програмним забезпеченням для оцінювання коментарів користувачів.

Враховуючи сценарії роботи користувачів при роботі з програмою, модель взаємодії користувача з програмним забезпеченням для оцінювання

коментарів користувачів подамо у вигляді діаграми прецедентів, зображеної на рис. 2.1.

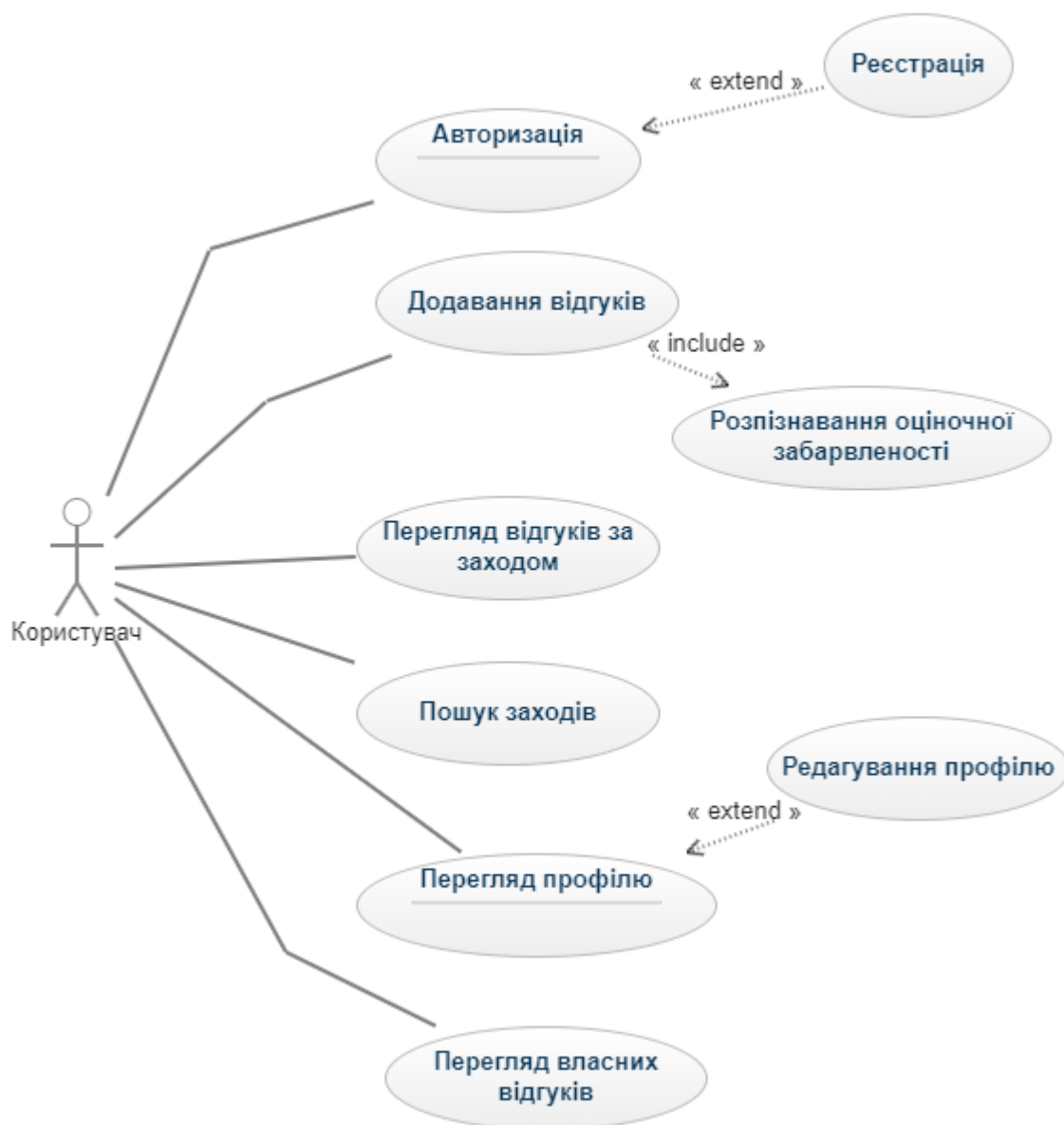


Рисунок 2.1 – Модель взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів

Таким чином, створено модель взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення для оцінювання коментарів користувачів.

2.4 Висновки за розділом 2

Для реалізації програмного забезпечення для оцінювання коментарів користувачів обрано мову програмування Python, яка має широкий спектр бібліотек машинного навчання, зокрема Scikit-learn і TensorFlow, що можуть спростити процес аналізу коментарів. Крім того, Python відомий своєю простотою та зрозумілістю коду, що дозволяє швидше розробляти і підтримувати програми, його синтаксис є інтуїтивно зрозумілим, що спрощує розробку і знижує вартість підтримки.

Для створення програмного забезпечення для оцінювання коментарів користувачів обрано фреймворк Django, який надає зручний механізм для роботи з формами, що спрощує обробку і валідацію даних, зокрема коментарів користувачів. Це дозволяє легко створювати і обробляти форми для збору зворотного зв'язку. Крім того, Django включає в себе потужні інструменти для тестування, що дозволяє створювати автоматизовані тести для перевірки функціональності програми. Це допомагає забезпечити високу якість коду та правильність роботи всіх функцій.

Створено модель взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення для оцінювання коментарів користувачів.

3 ОПИС ПРОГРАМИ

3.1 Структура програмного забезпечення для оцінювання коментарів користувачів

Структуру програмного забезпечення для оцінювання коментарів користувачів наведено на рис. 3.1.

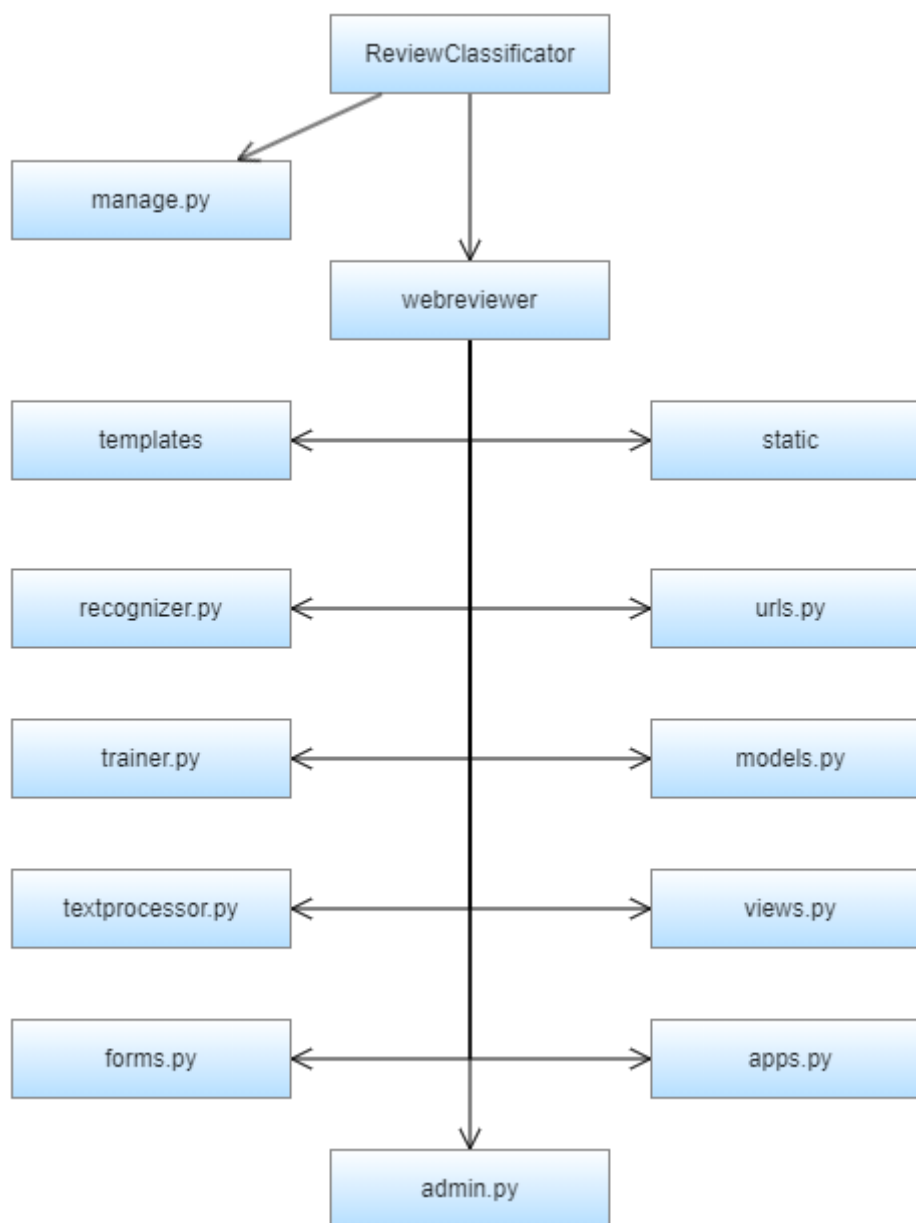


Рисунок 3.1 – Структура програмного забезпечення для оцінювання коментарів користувачів

Для організації структури програмного забезпечення для оцінювання коментарів користувачів, була використана архітектура, на якій базується вебфреймворк Django. Фреймворк Django близький до шаблону Model-View-Controller (MVC), тому його можна вважати сумісним із цією архітектурою: Model відповідає за доступ до даних і працює через шар взаємодії з базою даних; View визначає, які дані отримувати і як їх відображати, це забезпечується уявленнями і шаблонами; Controller вибирає потрібні уявлення залежно від введених користувачем даних, цю функцію виконує сам фреймворк через систему URL і викликає відповідну функцію Python [19].

Оскільки Controller обробляється фреймворком, а основна робота в Django виконується у моделях, шаблонах і уявленнях, його часто називають Model-Template-View-орієнтованим середовищем створення програмного забезпечення [19].

Ключовим файлом структури програмного забезпечення для оцінювання коментарів користувачів є `manage`, який створює застосунок для управління програмою.

Каталог `webreviewer` відповідає за структурування програмного забезпечення та містить відповідний пакет. Пакет `webreviewer` включає наступні файли:

- `textprocessor.py` – відповідає за обробку вхідних даних, таких як відгуки користувачів;
- `forms.py` – містить форми, що використовуються на вебсторінках програмного забезпечення для оцінювання коментарів користувачів;
- `recognizer.py` – реалізує модель розпізнавання;
- `views.py` – реалізує подання, які відповідають за взаємодію з користувачем;
- `trainer.py` – виконує навчання моделі;
- `admin.py` – забезпечує реєстрацію моделей у адміністративній частині;
- `models.py` – містить моделі, які визначають структуру даних для бази;

- apps.py – відповідає за реєстрацію програмного забезпечення;
- urls.py – реалізує керування URL-адресами і налаштування їхньої маршрутизації.

Також пакет містить два додаткові каталоги. Каталог templates містить шаблони для створення вебсторінок відповідно до вимог функціоналу. Каталог static включає папки з шрифтами, CSS-файлами, зображеннями та скриптами, необхідними для роботи бібліотек, що забезпечують роботу з шаблонами для різних елементів інтерфейсу програмного забезпечення для оцінювання коментарів користувачів.

3.2 Реалізація процесу оцінювання коментарів користувачів за допомогою нейромережових моделей

Для оцінювання коментарів користувачів у програмному забезпеченні використовуються нейромережові моделі [20]-[23].

З метою розв'язання завдань розпізнавання, зокрема, класифікації текстів, доцільно використовувати згорткові нейронні мережі [20]-[23].

Згорткова нейронна мережа (Convolutional Neural Network, CNN) за своєю структурою подібна до зорової кори мозку, невеликі області зору складаються з нейронів, розташованих у певних частинах поля зору. Хвильові характеристики зображень часто позначаються основними кольорами, наприклад, червоним, зеленим і синім, коли йдеться про кольорове кодування. Це означає, що, змішуючи ці три кольори, можна отримати будь-який відтінок. Вони можуть бути представлені за допомогою певного кольорового фільтра, який дозволяє отримати відповідні довжини хвиль. Під час обробки кожен піксель проходить через всі три канали масиву одночасно, а результати зберігаються [20]-[24].

Типова архітектура конволюційної нейронної мережі (рис. 3.2) виглядає як послідовність лінійних шарів, що відповідають певним принципам [20]-[24].

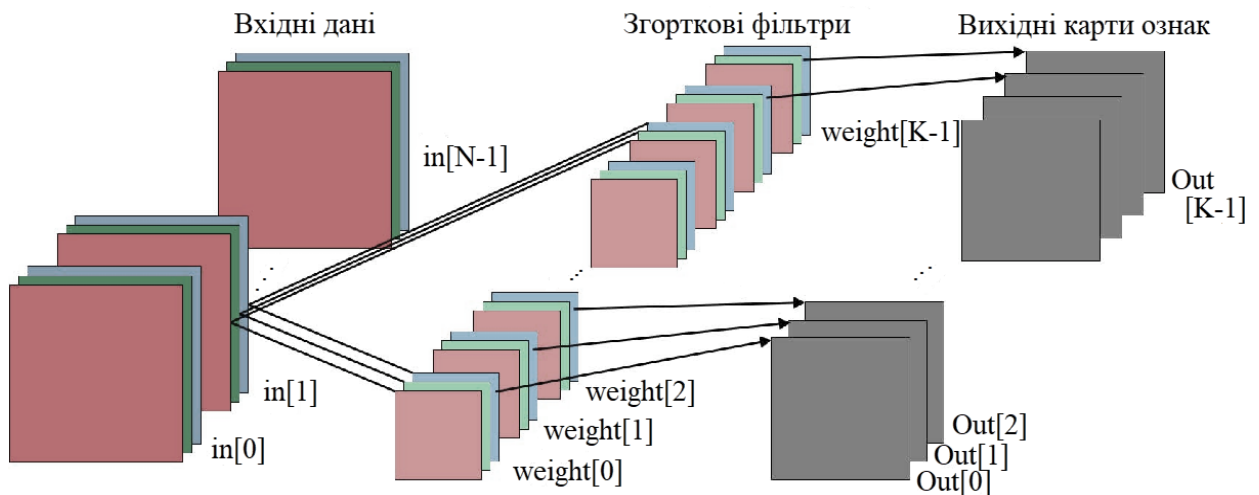


Рисунок 3.2 – Типова архітектура конволюційної нейронної мережі [20]-[23]

Основні типи шарів включають альтернативні шари, шари агрегації та шари прямого поширення [20]-[24].

Конволюційна мережа формує вихід шляхом згортки вхідного шару з кінцевим піддіпазоном за допомогою фільтра (ядра) та нелінійної активаційної функції [20]-[24].

Шари субдискретизації зменшують розмір карт функцій (наприклад, через максимальну вибірку) [20]-[24].

У багатошарових мережах кілька шарів щільно пов'язані, перетворюючи матрицю пікселів в остаточне вхідне зображення. Завдяки алгоритму прямого поширення мережа коригує свої параметри (ваги, пов'язані з шарами) методом зворотного поширення помилок, який оптимізує навчання, усуваючи попередні помилки [20]-[24].

Згорткова нейронна мережа аналізує зображення, чергуючи шари згортки, нелінійні шари, а також щільно зв'язані шари, і на виході отримує результат у вигляді або класу, або ймовірності, що описує зображення, що аналізується [20]-[24].

Перед початком обробки використовується матриця значень пікселів вхідного зображення. Фільтр є матрицею (відомою як вагова матриця або матриця параметрів). Починаючи з верхнього лівого кута, фільтр виконує

операцію згортки, множачи свої значення на пікселі вхідного зображення (об'єкта або змінної). Після множення всі значення підсумовуються [20]-[24].

Фільтр переміщується по всьому зображенню, і місця, де він прикладається, називаються областями пам'яті [20]-[24].

Після того, як фільтр пройде через всі точки зображення (перший крок), утворюється нова матриця (розмір може змінюватись в залежності від умов застосування фільтра) [20]-[24].

Якщо використовувати декілька фільтрів, результатом буде матриця, розмір якої залежить від кількості фільтрів [20]-[24].

Графічне подання операції згортки наведено на рис. 3.3.

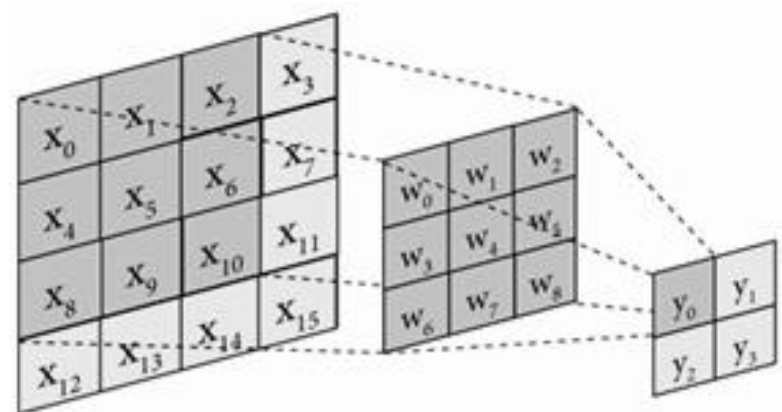


Рисунок 3.3 – Операція згортки [20]-[23]

Згортка має такі особливості: зберігає послідовність вхідних даних (в одновимірних випадках це порядок даних, у двовимірних — відносний порядок пікселів), оскільки застосовується до кожної ділянки зображення окремо. Згортка має перевагу в швидкості, оскільки кожен нейрон другого шару залежить лише від частини вхідних нейронів, тоді як у повнозв'язаних мережах кожен нейрон з'єднаний з усіма попередніми [20]-[24].

Зазвичай використовуються ті самі вагові коефіцієнти, оскільки вони повторюються на різних ділянках зображення [20]-[24].

У класичному шарі згортки, окрім згортки та нелінійності, існує ще одна операція: підвибірка, яку також називають об'єднанням або субдискретизацією (рис. 3.4) [20]-[24].

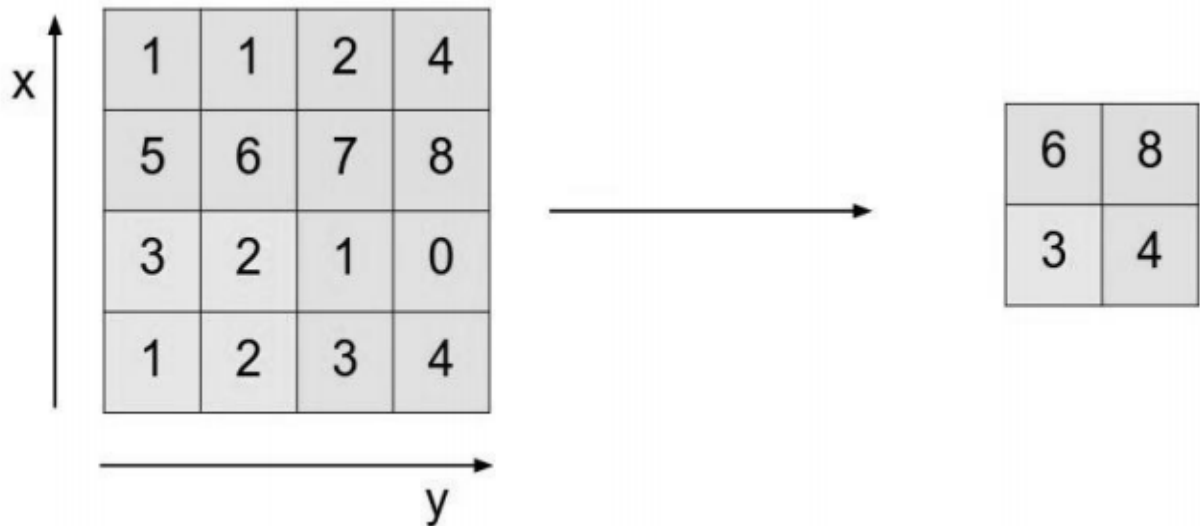


Рисунок 3.4 – Операція субдискретизації [20]-[23]

У динамічних мережах зазвичай більше уваги приділяється наявності або відсутності об'єкта, ніж його точним координатам. Наприклад, коли конволюційна мережа розпізнає обличчя, важливіше знати, що обличчя присутнє на зображенні, ніж точно визначати його межі. Це дозволяє фокусуватися на окремих об'єктах, втрачаючи частину інформації про розташування, але зменшуючи розмір даних [20]-[24].

Як правило, підвибірка виконується за допомогою вибору максимальної активності для кожної нейронної області [20]-[24].

Також можливий шар пропусків або регуляризація пропусків – це механізм запобігання перенаванчання в нейронних мережах, який зазвичай реалізується за допомогою стохастичного градієнтного спуску, вибірково вимикаючи частину нейронів [20]-[24].

Під час навчання мережа налаштовується так, що кожен нейрон може бути випадково вимкнений із певною ймовірністю. Потім мережа продовжує навчання з рештою ваг, і після кожної ітерації вимкнені нейрони знову

повертаються до роботи. Таким чином, на будь-якому етапі можливі різні архітектури,.

Під час тестування всі нейрони залишаються активними, але їхній внесок збільшується на певну величину. Таким чином, середній відгук мережі є достатнім для всіх можливих архітектур [20]-[24]. Ця техніка дозволяє досліджувати вплив напруги в мережі та аналізувати стабільність за допомогою потрібного стабілізуючого градієнта [20]-[24].

Один із підходів до використання згорткових мереж для класифікації коментарів користувачів базується на роботі з сигналами і символами. Ось як це працює: алфавіт представляється як послідовність із символів, кожен символ у тексті кодується числом, де кожен символ відповідає вектору певної довжини з одиницею на місці, що відповідає порядковому номеру символу, а всі інші елементи вектора дорівнюють нулю. Якщо символ у тексті не належить до алфавіту, він представлений вектором із нулями [20]-[23].

Із тексту вибираються перші символи. Ця кількість має бути достатньо великою, щоб забезпечити достатню кількість інформації для класифікації. Отримані вектори збираються в матрицю, де в кожному стовпці міститься не більше однієї одиниці. Кожен рядок цієї матриці являє собою таблицю з певною кількістю елементами. Ця матриця передається на вхід згорткової нейронної мережі [20]-[24].

Інший метод полягає у транслітерації слів. У цьому випадку кожне слово представлено у вигляді багаточлена, і це використовується для визначення точок у просторі (рис. 3.5). Для кожного слова обчислюється вектор, і ці вектори впорядковуються, щоб створити матрицю, яка подається у нейронну мережу як графічна матриця [20]-[24]. Методи ґрунтуються на наступних припущеннях: вектори слів є числовим поданням, які зберігають семантичні зв'язки між словами. Наприклад, вектори для слів "кішка" і "собака" будуть близькими через їх схожість у контексті, тоді як вектор слова "ручка" буде легко відрізнитися [20]-[24].

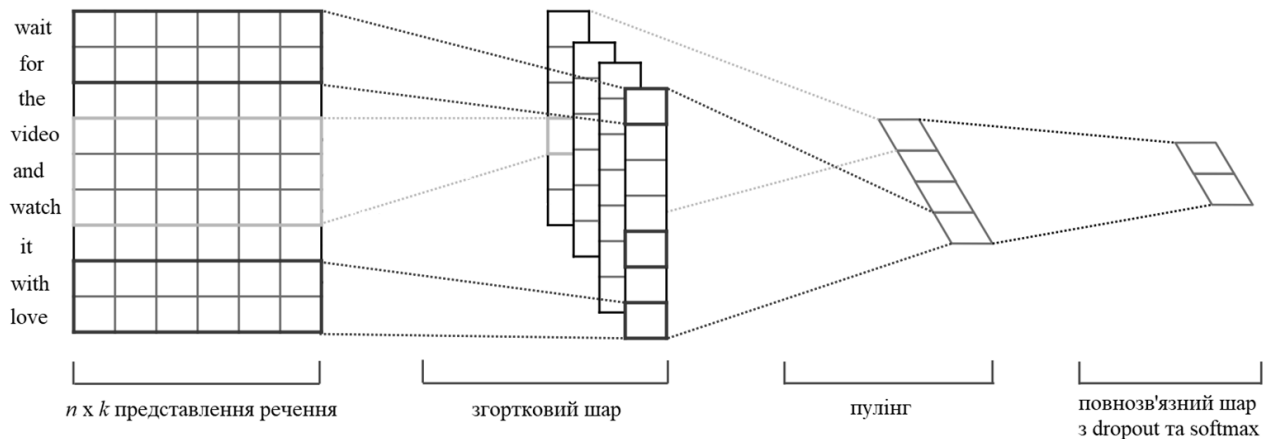


Рисунок 3.5 – Конволюційна нейронна мережа для оцінювання коментарів користувачів [20]-[23]

Схожість між певними словами пояснюється тим, що вони часто зустрічаються в подібних контекстах. Алгоритми Word2Vec використовують зміст для визначення значень слів, тому слова, що з'являються в однаковому контексті, мають схожі векторні подання [20]-[24].

Загалом, метод включає наступні етапи: формування наборів даних [вхід, вихід], де кожне слово кодується як вектор певної довжини, з одним значущим елементом і рештою нулями. Потім виконується побудова моделі, де вхідні та вихідні вектори мають однакову структуру. На наступному етапі визначаються функції втрат, яка підбирає найкращу модель для прогнозування контексту слів, після чого виконується оцінка якості моделі, щоб векторні подання слів відображали їх семантичну схожість [20]-[24].

3.3 Реалізація процесу оцінювання коментарів користувачів

У розроблюваному програмному забезпеченні використовується підхід для оцінювання коментарів користувачів, на які реагують користувачі, у таких категоріях, як позитивний та негативний коментар. Також можливі й інші категорії. Для реалізації процесу спочатку створюється аналітична модель, що

базується на згортковій нейронній мережі. Ця модель включає кілька шарів, кожен з яких має певну кількість ядер згортки.

Функція активації забезпечує нелінійність, а вибірка здійснюється через пулінг. Під час навчання модель постійно оновлюється без повторення процесу, а текст перетворюється на матрицю з урахуванням пунктуації та спеціальних символів. Функція softmax використовується для обчислення втрат [20]-[23].

До процесу навчання моделі важливо підготувати дані. Дані організовуються у вигляді пар, де кожен рядок тексту відповідає певній категорії. Ці дані поділяються на навчальну та тестову вибірки. Перед використанням дані трансформуються. Зокрема, текст скорочується до певної кількості символів, якщо він перевищує цей ліміт. Видаляються знаки пунктуації, зайві підкреслення та так звані стоп-слова. Весь текст приводиться до нижнього регістру.

Для обробки кожен рядок тексту кодується за допомогою певної кількості змінних. Кожен символ тексту відповідає одному з символів алфавіту або спеціальному коду для решти символів. Оптимізація моделі відбувається за допомогою алгоритму Адам [20]-[24].

На завершальному етапі текст (коментар користувача), що подається на вхід, проходить попередню обробку, включаючи скорочення до потрібної довжини. Після цього текст перетворюється у вектор, який подається на вхід нейронної мережі. Вихідні дані мережі визначають категорію відповіді на основі результату її обробки.

Створення згорткового шару реалізується за допомогою функції `tensorflow.nn.conv2d()`, яка викликається наступним чином:

```
conv_layer = tf.nn.conv2d(self.input_data, filters, strides=[1, 1, 1, 1], padding="VALID", name="conv_layer")
```

У цьому виклику функції використовуються такі параметри: `self.input_data` — це вхідні дані, закодовані з текстових значень; параметр `padding` задає, що розмір виходу буде меншим за вхідний; `filters` — фільтри,

які зберігають вагові коефіцієнти; параметр `strides` визначає кількість пікселів, на які зсувається фільтр під час кожного кроку.

Функція активації ReLU додається до цього шару наступним чином:

```
activated_layer = tf.nn.relu(tensorflow.nn.bias_add(conv_layer, bis),
name="relu_activation")
```

Субдискретизація виконується через функцію максимального пулінгу:

```
max_pooled = tf.nn.max_pool(activated_layer, ksize=[1, kernel_size -
stride + 1, 1, 1], padding="VALID", name="max_pool")
```

Ці операції виконуються в циклі для кожного фільтра. На виході створюється повнозв'язний шар, який використовує вагові коефіцієнти та зміщення для обчислення результатів, і визначення класу належності відгуку, якому відповідає найбільша оцінка. Це виглядає так:

```
final_weights = tf.Variable(tensorflow.truncated_normal([filter_qty,
num_classes], stddev=0.1), name="final_weights")

final_bias = tf.Variable(tensorflow.constant(0.1, shape=[num_classes]),
name="final_bias")

loss_function += tf.nn.l2_loss(final_weights)
loss_function += tf.nn.l2_loss(final_bias)

self.output = tf.nn.xw_plus_b(self.final_input, final_weights, final_bias,
name="final_scores")

self.prediction = tf.argmax(self.output, name="prediction_output")
```

У файлі `trainr.py` організовано процес тренування моделі. Його запуск відбувається лише тоді, коли модель потрібно навчити вперше або під час перенавчання, тому код не містить зовнішніх функцій, за винятком внутрішніх, які допомагають організувати логіку.

Збереження результатів навчання реалізується через функцію:

```
saver = tf.train.Saver(tf.global_variables(),
maxtokeep=TrainingParams.max_checkpoints)
```

Оцінка результатів під час тестування і збереження поточного стану виконуються за допомогою наступних команд:

```

if step % TrainingParams.eval_interval == 0:
    evaluate_model(test_inputs, test_labels,
summary_writer=summary_logger)
if step % TrainingParams.save_interval == 0:
    checkpoint = saver.save(session_info, model_checkpoint_path,
global_step=step)
    saver.save(session_info, os.path.join(log_dir, "model.ckpt"),
global_step=step)

```

Додатково організовано дві функції:

`train_step(train_inputs, train_labels)` — виконує один етап тренування на основі вхідних та вихідних даних;

`evaluate_model(test_inputs, test_labels, summary_writer=None)` — виконує один етап оцінки на тестових даних, отримуючи додатковий параметр `summary_writer` для збереження результатів.

3.4 Проєктування інтерфейсу програмного забезпечення для оцінювання коментарів користувачів

Проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів виконано з урахуванням вимог технічного завдання. При розробленні програмного забезпечення для оцінювання коментарів користувачів враховані функціональні вимоги до програми:

- можливість підтримки процесів авторизації та реєстрації користувачів;
- можливість пошуку розважальних заходів;
- підтримка можливості ознайомлення з інформацією про розважальні заходи;
- можливість коментування (залишення відгуків) про розважальні заходи;

- можливість перегляду власних коментарів про розважальні заходи;
- підтримка можливості перегляду та редагування інформації про профіль користувача;
- можливість оцінювання відгуків користувачів (класифікація їх на позитивні та негативні) для подальшого аналізу розважальних заходів;
- підтримка можливості керування інформацією про розважальні заходи (функція адміністратора);
- можливість збереження результатів оцінювання коментарів користувачів.

Інтерфейс головної форми програмного забезпечення для підтримки оцінювання коментарів користувачів наведено на рис. 3.5.

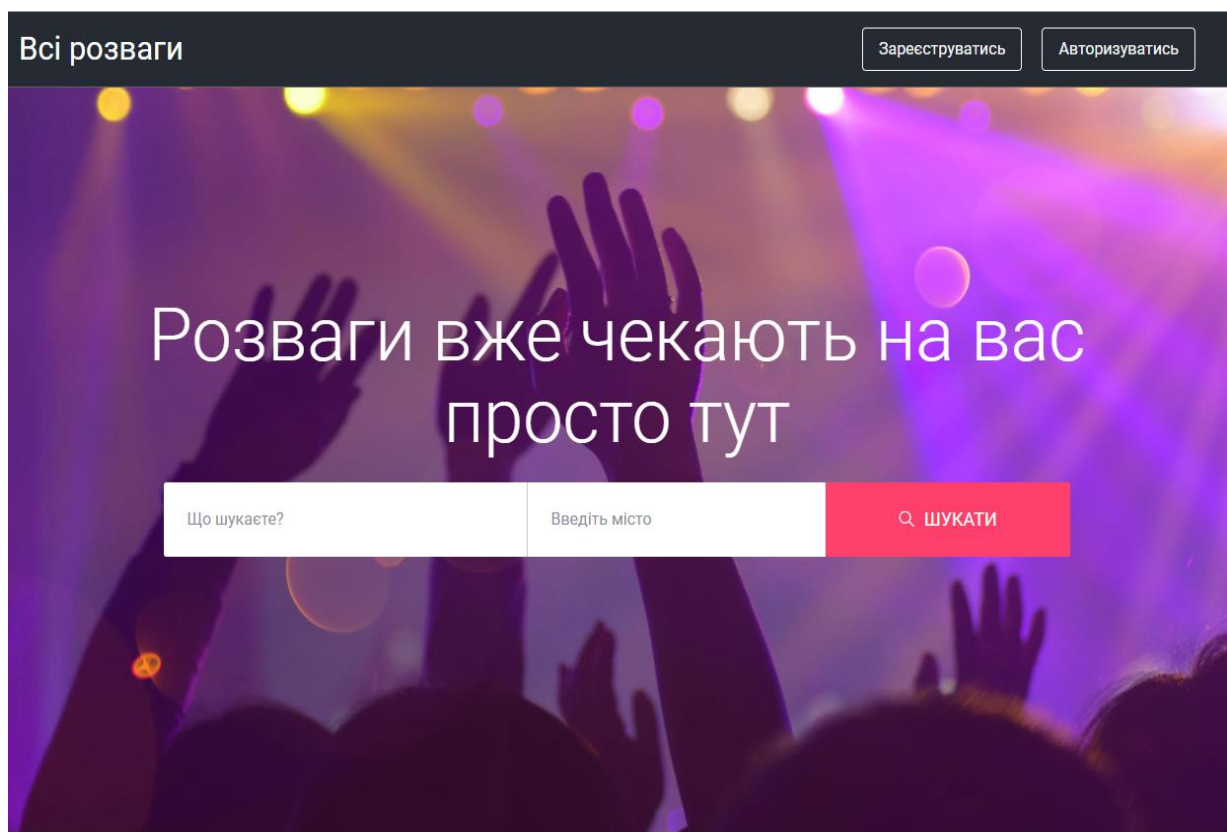


Рисунок 3.5 – Інтерфейс головної форми програмного забезпечення для оцінювання коментарів користувачів

Сторінка авторизації користувача наведена на рис. 3.6.

Всі розваги

Зареєструватись

Авторизуватись

АВТОРИЗАЦІЯ

Логін (Електронна пошта)

Пароль

УВІЙТИ

Рисунок 3.6 – Сторінка авторизації користувача

Сторінка з наявними розважальними заходами, що відповідають запиту користувача, наведена на рис. 3.7.

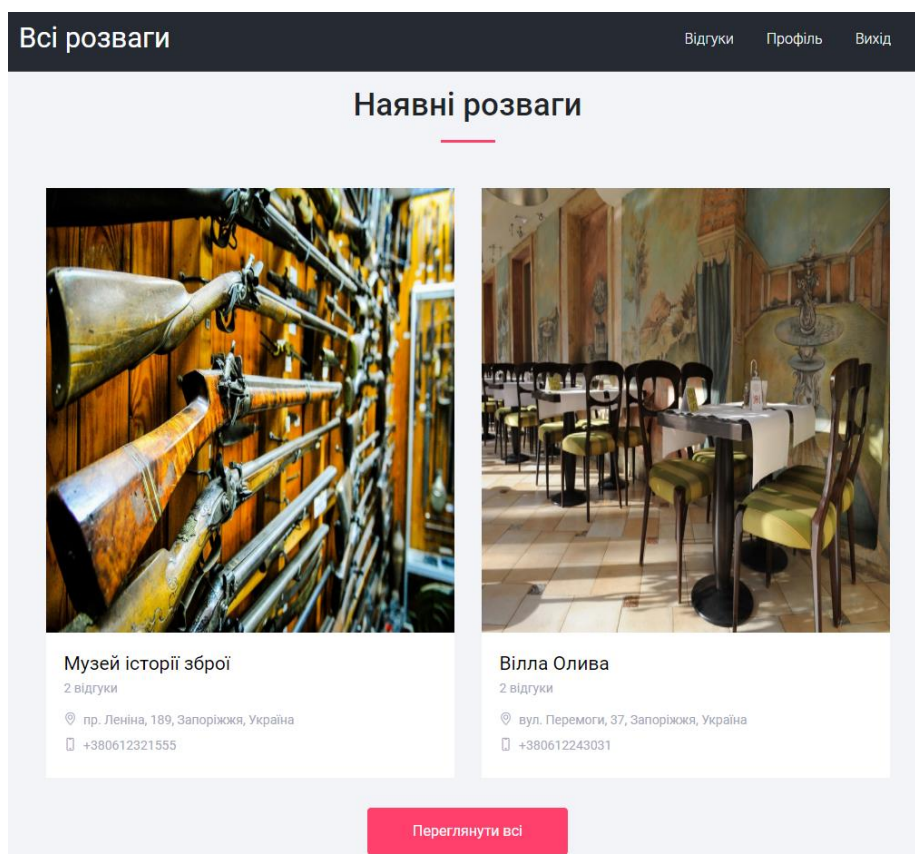


Рисунок 3.7 – Сторінка з наявними розважальними заходами

Сторінка з описом розважального заходу з відгуками користувачів наведена на рис. 3.8.



Вілла Олива

ЗАЛИШИТИ ВІДГУК

2 відгуки

Ресторан традиційної італійської кухні

Простір піцерії досить камерний, щоб гості відчували себе захищеними від вуличної суєти. Тут два зали і веранда; у великому залі, з величезних вікон якого відкривається вид на дворик, багато повітря і світла.

Інтер'єрний настрій задає арка, оформлена авторською майолікою. Кераміка візуально перетворює простір і наповнює інтер'єр теплом декору, зробленого вручну.

Гостей приваблює краса натуральних фактур, властивих обпаленій глині, кольору сонця, моря і неба. Створюють відчуття комфорту і меблі з вибіленого дуба, і вінтажні двері в кубинському стилі. В меню 22 види піци яку випікають у справжній дров'яній печі.

вул. Перемоги, 37
Запоріжжя
Україна

+380612243031

2 відгуки



Victor Pretto
2 відгуки

Recommended

добрий

Hardly a midrange restaurant and not the cheapest but the food, service & ambiance are worth every Hryvna. A little of the regular beat but thoroughly recommended.



Petro Nietto
3 відгуки

Amazing

відмінний

We had an amazing time with my husband in this place. He is Italian and in the city it was the only place where the pizza is a hit similar to the Italian one so he was very satisfied that I

Рисунок 3.8 – Сторінка з описом розважального заходу

Форма введення відгуку користувача наведена на рис. 3.9.

Залиште свій відгук ×

Назва:

Введіть назву

Текст:

Введіть відгук

Скасувати Зберегти

Рисунок 3.9 – Форма введення відгуку користувача

Сторінка профілю користувача наведена на рис. 3.10.

Всі розваги Відгуки Профіль Вийти

ВЛАСНИЙ ПРОФІЛЬ

Додати фото

Пошта:

post@gmail.com

Новий пароль:

Введіть новий пароль

Повторно новий пароль:

Введіть новий пароль ще раз

Ім'я:

Ім'я_користувача

Прізвище:

Прізвище_користувача

Телефон:

+380505050505

Оновити

Рисунок 3.10 – Сторінка профілю користувача

На рис. 3.11 наведено форму додавання нового розважального заходу.

Всі розваги

Новий західЗаходиПрофільВийти

НОВИЙ РОЗВАЖАЛЬНИЙ ЗАХІД



Додати фото

Назва:

Введіть назву

Місце проведення:

Введіть адресу

Опис:

Додати розширений опис

Місто:

Введіть місто проведення

Країна:

Введіть країну проведення

Телефон:

Введіть номер телефону

ДОДАТИ

Рисунок 3.11 – Форма додавання нового розважального заходу

Форма оцінювання відгуків користувачів наведена на рис. 3.12.

Аналіз відгуків користувачів

Назва:

Музей історії зброї

Проаналізувати відгуки

Позитивні відгуки:

1. Неймовірна колекція, захоплюючі експонати, справжнє занурення в історію зброї!
2. Вражаюча експозиція, чудовий гід, багато цікавих фактів та артефактів.
3. Фантастичне місце для любителів історії, багатогранний і захопливий досвід.
4. Експонати зброї вражають! Чудово підійде для сімейного відвідування.
5. Дуже пізнавально, красиво оформлено, ідеальне місце для цікавого дозвілля!

Загальна кількість позитивних відгуків: 5

Негативні відгуки:

1. Цікава експозиція, але трохи замало інтерактивних елементів.
2. Експонати гарні, проте недостатньо інформації для детального ознайомлення.
3. Добре оформлено, однак брак сучасних технологій знижує враження.

Загальна кількість негативних відгуків: 3

Зберегти результати аналізу відгуків

Рисунок 3.12 – Форма оцінювання відгуків користувачів

3.5 Висновки за розділом 3

Розроблено програмне забезпечення для оцінювання коментарів користувачів.

Запропоновано структуру програмного забезпечення для оцінювання коментарів користувачів. Визначено, що для організації структури програмного забезпечення для оцінювання коментарів користувачів, була використана архітектура, на якій базується вебфреймворк Django, який близький до шаблону MVC. У програмному забезпеченні для оцінювання

коментарів користувачів Model відповідає за доступ до даних і працює через шар взаємодії з базою даних, View визначає, які дані отримувати і як їх відображати, це забезпечується уявленнями і шаблонами, Controller вибирає потрібні уявлення залежно від введених користувачем даних.

Описано функціонування програмного забезпечення для оцінювання коментарів користувачів, а також реалізацію процесу оцінювання коментарів користувачів

Описано особливості реалізації програмного забезпечення для оцінювання коментарів користувачів. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Призначення й умови застосування програми

Програма призначена для підтримки процесу оцінювання коментарів користувачів. Програма написана на мові Python, яка має широкий спектр бібліотек машинного навчання, що можуть спростити процес аналізу коментарів. В якості фреймворку для створення програмного забезпечення для оцінювання коментарів користувачів обрано Django, який надає зручний механізм для роботи з формами, що спрощує обробку і валідацію даних, зокрема коментарів користувачів.

Програмне забезпечення для оцінювання коментарів користувачів забезпечує виконання таких функцій:

- можливість підтримки процесів авторизації та реєстрації користувачів;
- можливість пошуку розважальних заходів;
- підтримка можливості ознайомлення з інформацією про розважальні заходи;
- можливість коментування (залишення відгуків) про розважальні заходи;
- можливість перегляду власних коментарів про розважальні заходи;
- підтримка можливості перегляду та редагування інформації про профіль користувача;
- можливість оцінювання відгуків користувачів (класифікація їх на позитивні та негативні) для подальшого аналізу розважальних заходів;
- підтримка можливості керування інформацією про розважальні заходи (функція адміністратора);
- можливість збереження результатів оцінювання коментарів користувачів.

Для експлуатації програмного забезпечення для оцінювання коментарів користувачів необхідно таке апаратне та програмне забезпечення:

- багатоядерний процесор, наприклад, Intel i7, i9 або AMD Ryzen 7, 9;
- мінімум 16 ГБ оперативної пам'яті для ефективної роботи з великими наборами даних;
- для роботи з великими даними та тренування нейронних мереж графічний процесор;
- дисковий накопичувач для зберігання даних та забезпечення швидкого доступу до них;
- операційна система та веббраузер на стороні клієнта.

Функціональні характеристики розробленого програмного забезпечення для оцінювання коментарів користувачів наведено у технічному завданні (додаток А). Фрагмент тексту програмного забезпечення для оцінювання коментарів користувачів наведено у додатку Б.

4.2 Характеристики програми для оцінювання коментарів користувачів

Програмне забезпечення для оцінювання коментарів користувачів має ряд особливостей і характеристик, що забезпечують її ефективність та коректність у визначенні тональності відгуків користувачів.

Програма може визначати клас коментаря (позитивний чи негативний коментар), а, за необхідністю, у випадку синтезу додаткових розпізнавальних моделей, її класифікувати відгуки користувачів за більш детальними категоріями, такими як нейтральний, саркастичний чи неадекватний. Програмні модулі для оцінювання коментарів користувачів надають можливість аналізувати коментарі та надавати оцінки в режимі реального часу, що є важливим для інтерактивних платформ.

Програма використовує нейромережеві розпізнавальні моделі, що дає змогу краще розуміти контекст і сенс коментарів. Програмні модулі для

оцінювання коментарів користувачів надають можливість навчати модель на специфічних наборах даних для покращення точності в певних сферах або індустріях (наприклад, аналіз відгуків клієнтів у сфері електронної комерції), забезпечуючи здатність моделі вдосконалюватись на основі нових даних, що допомагає покращувати оцінки з часом.

4.3 Інструкція по експлуатації програми

4.3.1 Звернення до програми

Для запуску програмного забезпечення для оцінювання коментарів користувачів користувач може відкрити браузер і перейти на відповідну адресу, де він зможе взаємодіяти з програмою.

4.3.2 Вхідні й вихідні дані

Вхідними даними до програмного забезпечення для оцінювання коментарів користувачів є коментарі користувачів, а також інформація про розважальні заходи, щодо яких користувачі можуть залишати відгуки.

Вихідними даними програмного забезпечення для оцінювання коментарів користувачів є створені сторінки, що відображають інформацію про розважальні заходи, а також класи, до яких віднесено коментарі користувачів.

4.3.3 Повідомлення

Користувач програмного забезпечення для оцінювання коментарів користувачів може отримати такі повідомлення: повідомлення про успішне завершення процесу оцінювання коментарів користувачів та ін.

4.4 Виконання програмного забезпечення для оцінювання коментарів користувачів

Програмне забезпечення дозволяє користувачам ознайомлюватися з інформацією про розважальні заходи, залишати коментарі (відгуки) про розважальні заходи, а також оцінювати коментарі користувачів (відносити коментарі користувачів до класів позитивних та негативних відгуків).

Після запуску програмного забезпечення для оцінювання коментарів користувачів відображається головна сторінка (рис. 4.1), де користувач може авторизуватися або знайти розважальні заходи у своєму місті.

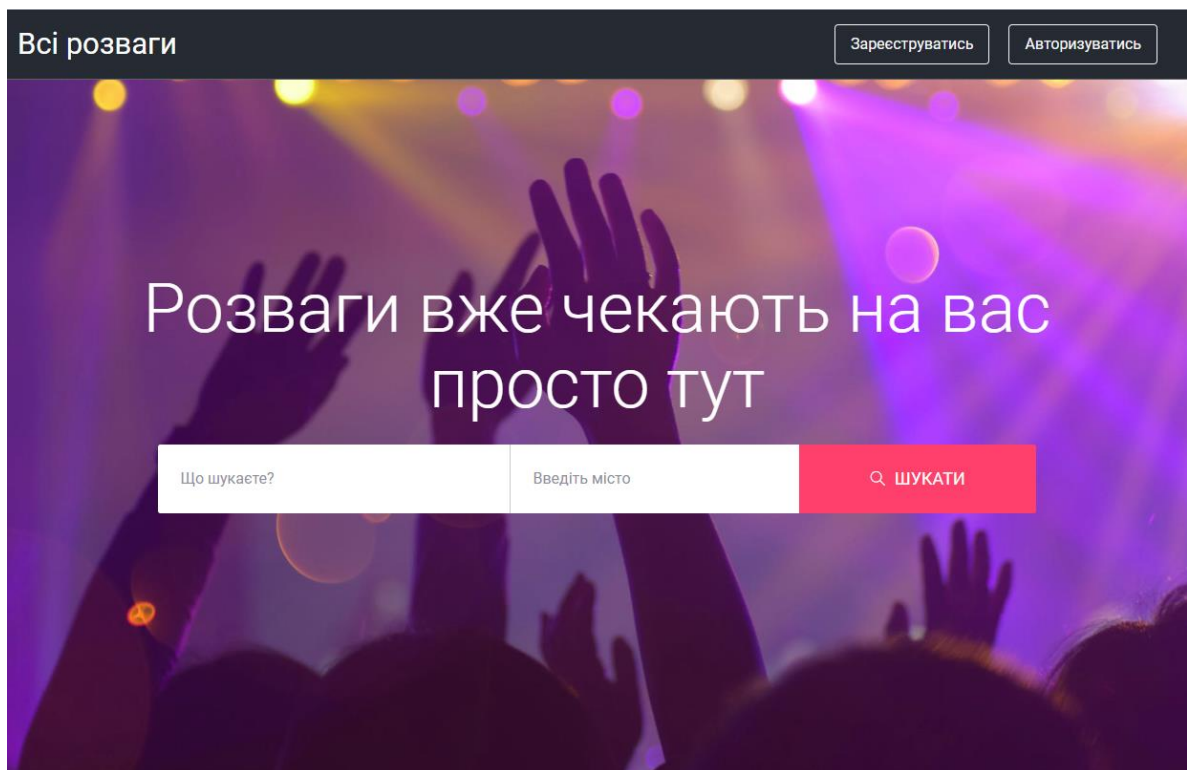


Рисунок 4.1 – Інтерфейс головної сторінки програмного забезпечення для оцінювання коментарів користувачів

Для отримання можливостей повноцінної роботи з програмою, зокрема, можливостей написання коментарів стосовно різних подій, користувачу необхідно зареєструватися та авторизуватися (рис. 4.2).

Всі розваги

Зареєструватись

Авторизуватись

АВТОРИЗАЦІЯ

Логін (Електронна пошта)

Пароль

УВІЙТИ

Рисунок 4.2 – Авторизація користувача

Після авторизації у головному меню з'являються кнопки для залишення відгуків та перегляду профілю користувача. Обравши місто у відповідному полі головної сторінки або ввівши ключові слова, що стосуються заходу, користувач отримає перелік наявних розважальних заходів, що відповідають його запиту (рис. 4.3).

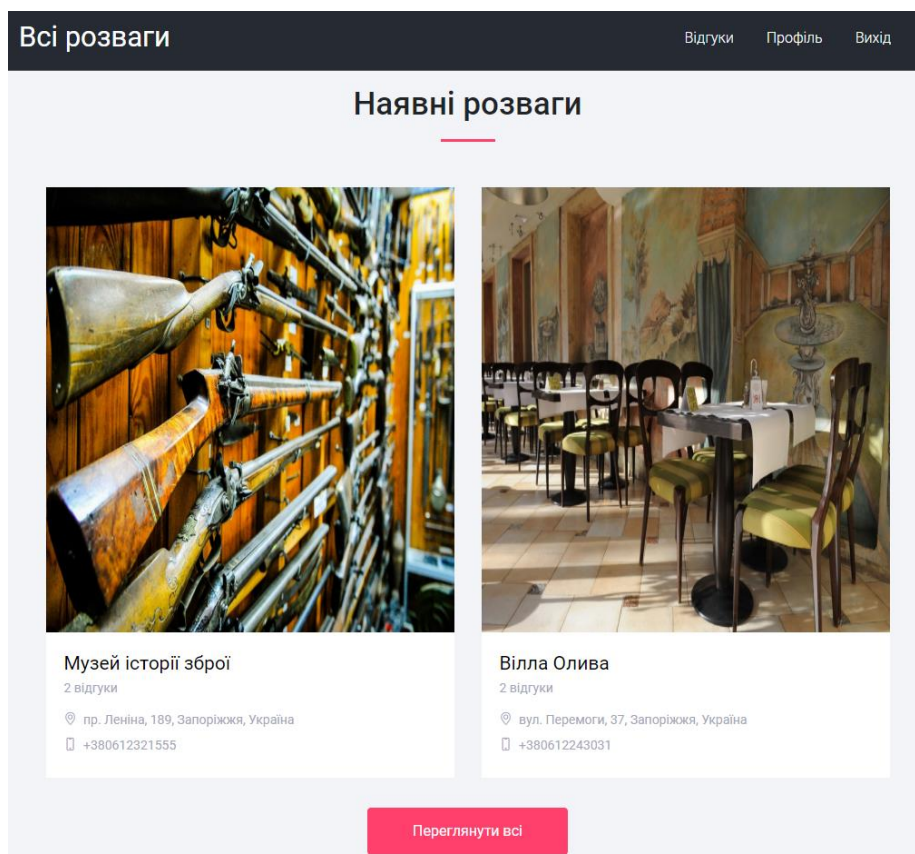


Рисунок 4.3 – Наявні розважальні заходи, що відповідають запиту користувача

При натисканні на конкретну подію або її місце проведення користувачу відобразиться її детальний опис, а також відгуки користувачів (рис. 4.4).

Всі розваги

Відгуки Профіль Вихід



Вілла Олива

ЗАЛИШИТИ ВІДГУК

2 відгуки

Ресторан традиційної італійської кухні

Простір піцерії досить камерний, щоб гості відчували себе захищеними від вуличної суєти. Тут два зали і веранда; у великому залі, з величезних вікон якого відкривається вид на дворик, багато повітря і світла.

Інтер'єрний настрій задає арка, оформлена авторською майолікою. Кераміка візуально перетворює простір і наповнює інтер'єр теплом декору, зробленого вручну.

Гостей приваблює краса натуральних фактур, властивих обпаленій глини, кольору сонця, моря і неба. Створюють відчуття комфорту і меблі з вибленого дуба, і вінтажні двері в кубинському стилі. В меню 22 види піци яку випікають у справжній дров'яній печі.

вул. Перемоги, 37

Запоріжжя

Україна

+380612243031

2 відгуки



Victor Pretto

2 відгуки

Recommended

добрий

Hardly a midrange restaurant and not the cheapest but the food, service & ambiance are worth every Hryvna. A little of the regular beat but thoroughly recommended.



Petro Nietto

3 відгуки

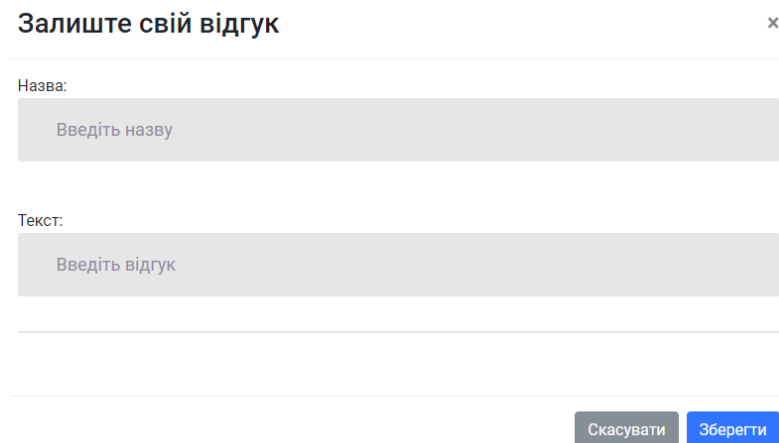
Amazing

відмінний

We had an amazing time with my husband in this place. He is Italian and in the city it was the only place where the pizza is a hit similar to the Italian one so he was very satisfied that I

Рисунок 4.4 – Опис розважального заходу з відгуками користувачів

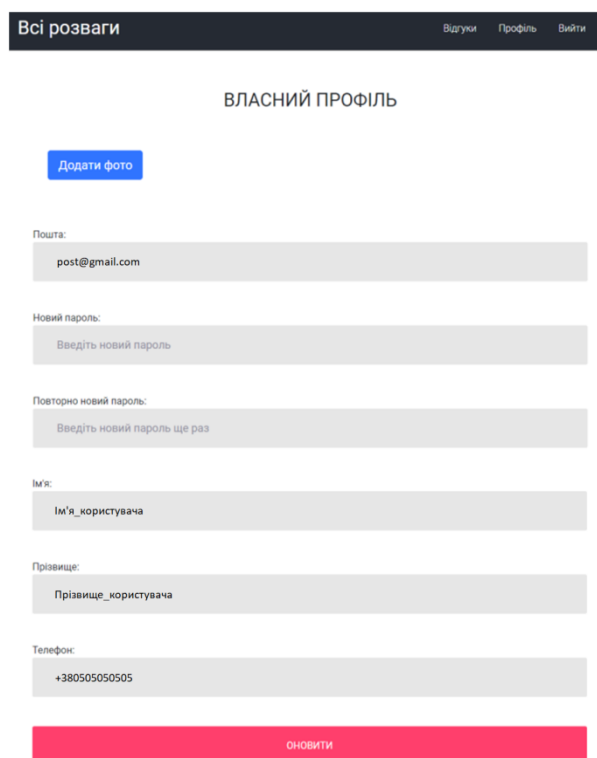
Користувач має можливість залишити власний відгук, натиснувши відповідну кнопку. У формі відгуку (рис. 4.5) користувач може ввести текст відгуку, після чого необхідно натиснути кнопку Зберегти.



The form is titled "Залиште свій відгук" (Leave your review) with a close button (x) in the top right corner. It contains two input fields: "Назва:" (Name) with a placeholder "Введіть назву" (Enter name) and "Текст:" (Text) with a placeholder "Введіть відгук" (Enter review). At the bottom right, there are two buttons: "Скасувати" (Cancel) and "Зберегти" (Save).

Рисунок 4.5 – Форма введення відгуку користувача

Користувач має можливість перегляду та редагування власного профілю (рис. 4.6).



The form is titled "Всі розваги" (All entertainment) with navigation links "Відгуки" (Reviews), "Профіль" (Profile), and "Вийти" (Logout). The main heading is "ВЛАСНИЙ ПРОФІЛЬ" (MY PROFILE). It includes a "Додати фото" (Add photo) button. Below are input fields for "Пошта:" (Email) with value "post@gmail.com", "Новий пароль:" (New password) with placeholder "Введіть новий пароль" (Enter new password), "Повторно новий пароль:" (Repeat new password) with placeholder "Введіть новий пароль ще раз" (Enter new password once more), "Ім'я:" (Name) with value "Ім'я_користувача" (User name), "Прізвище:" (Surname) with value "Прізвище_користувача" (User surname), and "Телефон:" (Phone) with value "+380505050505". A red "Оновити" (Update) button is at the bottom.

Рисунок 4.6 – Профіль користувача

Користувач з правами адміністратора має можливість керування інформацією про заходи, що відображаються на сайті програмного забезпечення. На рис. 4.7 наведено форму додавання нового розважального заходу.

Всі розваги

Новий західЗаходиПрофільВийти

НОВИЙ РОЗВАЖАЛЬНИЙ ЗАХІД



Додати фото

Назва:

Введіть назву

Місце проведення:

Введіть адресу

Опис:

Додати розширений опис

Місто:

Введіть місто проведення

Країна:

Введіть країну проведення

Телефон:

Введіть номер телефону

ДОДАТИ

Рисунок 4.7 – Форма додавання нового розважального заходу

Також користувач з правами адміністратора має можливість аналізу відгуку користувачів. Для цього він вводить назву заходу та натискає кнопку Проаналізувати відгуки (рис. 4.8). Після цього програма аналізує відгуки користувачів за обраним заходом, використовуючи побудовану нейромережеву модель, та класифікує їх на позитивні та негативні відгуки.

Всі розваги

Новий західЗаходиПрофільВийти

Аналіз відгуків користувачів

Назва:

Музей історії зброї

Проаналізувати відгуки

Позитивні відгуки:

1. Неймовірна колекція, захоплюючі експонати, справжнє занурення в історію зброї!
2. Вражаюча експозиція, чудовий гід, багато цікавих фактів та артефактів.
3. Фантастичне місце для любителів історії, багатогранний і захопливий досвід.
4. Експонати зброї вражають! Чудово підійде для сімейного відвідування.
5. Дуже пізнавально, красиво оформлено, ідеальне місце для цікавого дозвілля!

Загальна кількість позитивних відгуків: 5

Негативні відгуки:

1. Цікава експозиція, але трохи замало інтерактивних елементів.
2. Експонати гарні, проте недостатньо інформації для детального ознайомлення.
3. Добре оформлено, однак брак сучасних технологій знижує враження.

Загальна кількість негативних відгуків: 3

Зберегти результати аналізу відгуків

Рисунок 4.8 – Форма оцінювання відгуків користувачів

4.5 Тестування програмного забезпечення для оцінювання коментарів користувачів

Виконано тестування програмного забезпечення для оцінювання коментарів користувачів.

Виявлено, що програма для підтримки процесу оцінювання коментарів користувачів функціонує правильно та злагоджено. Вона реалізує всі функціональні вимоги та успішно виконує свою основну задачу підтримки процесу оцінювання коментарів користувачів.

Розроблене програмне забезпечення дозволяє забезпечувати автоматизацію процесів, пов'язаних з підтримкою процесу оцінювання коментарів користувачів.

4.6 Висновки за розділом 4

Описано програмне забезпечення для оцінювання коментарів користувачів. Виконано тестування розробленого програмного забезпечення для оцінювання коментарів користувачів. Результати тестування програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з підтримкою процесу оцінювання коментарів користувачів.

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи магістра було проаналізовано та досліджено процес розробки програмного забезпечення для оцінювання емоційного забарвлення текстових повідомлень.

Визначено, що відгук користувача являє собою зворотній зв'язок або думку, яку надають користувачі про продукт, послугу чи досвід взаємодії з компанією. Відгуки та коментарі користувачів допомагають компаніям зрозуміти потреби своїх клієнтів, визначити сильні та слабкі сторони продукту або послуги, а також приймати рішення щодо вдосконалення на основі отриманої інформації. Відгуки можуть бути надані у вигляді оцінок, коментарів, опитувань або через інші канали зв'язку, такі як соціальні мережі чи служби підтримки. Проаналізовано процес аналізу відгуків користувачів за допомогою програмного забезпечення.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для оцінювання коментарів користувачів. Проте деякі програмні засоби для оцінювання коментарів користувачів можуть не завжди правильно інтерпретувати контекст коментарів, що обумовлено використанням неефективних методів штучного інтелекту та розпізнавальних моделей, що може призвести до неправильної оцінки настрою або значущості коментарів. Деяке програмне забезпечення може мати обмежені можливості для точного аналізу коментарів, написаних на різних мовах або діалектах, що особливо актуально для глобальних компаній, які працюють на ринках різних країн. Для того щоб деяке програмне забезпечення для оцінювання коментарів користувачів працювало ефективно, може знадобитися значний обсяг часу і ресурсів на його налаштування під специфічні потреби бізнесу, що включає адаптацію моделей і інтеграцію з іншими системами. Тому актуальною є розробка програмного забезпечення для оцінювання коментарів користувачів.

Сформульовано функціональні вимоги до програмного забезпечення для оцінювання коментарів користувачів.

Для реалізації програмного забезпечення для оцінювання коментарів користувачів обрано мову програмування Python, яка має широкий спектр бібліотек машинного навчання, зокрема Scikit-learn і TensorFlow, що можуть спростити процес аналізу коментарів. Крім того, Python відомий своєю простотою та зрозумілістю коду, що дозволяє швидше розробляти і підтримувати програми, його синтаксис є інтуїтивно зрозумілим, що спрощує розробку і знижує вартість підтримки.

Для створення програмного забезпечення для оцінювання коментарів користувачів обрано фреймворк Django, який надає зручний механізм для роботи з формами, що спрощує обробку і валідацію даних, зокрема коментарів користувачів. Це дозволяє легко створювати і обробляти форми для збору зворотного зв'язку. Крім того, Django включає в себе потужні інструменти для тестування, що дозволяє створювати автоматизовані тести для перевірки функціональності програми. Це допомагає забезпечити високу якість коду та правильність роботи всіх функцій.

Новизна роботи полягає у тому, що створено модель взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення для оцінювання коментарів користувачів.

Практична цінність результатів роботи полягає у тому, що розроблено програмне забезпечення для оцінювання коментарів користувачів.

Запропоновано структуру програмного забезпечення для оцінювання коментарів користувачів. Визначено, що для організації структури програмного забезпечення для оцінювання коментарів користувачів, була використана архітектура, на якій базується вебфреймворк Django, який близький до шаблону MVC. У програмному забезпеченні для оцінювання коментарів користувачів Model відповідає за доступ до даних і працює через

шар взаємодії з базою даних, View визначає, які дані отримувати і як їх відображати, це забезпечується уявленнями і шаблонами, Controller вибирає потрібні уявлення залежно від введених користувачем даних.

Описано функціонування програмного забезпечення для оцінювання коментарів користувачів, а також реалізацію процесу оцінювання коментарів користувачів

Описано особливості реалізації програмного забезпечення для оцінювання коментарів користувачів. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів.

Виконано тестування розробленого програмного забезпечення для оцінювання коментарів користувачів. Результати тестування програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з підтримкою процесу оцінювання коментарів користувачів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Unlocking Insights: Your Guide to User Feedback Analysis [Electronic resource]. – Access mode: <https://userback.io/blog/guide-to-user-feedback-analysis/>.
2. Customer Feedback Analysis: How To Analyze & Act on Feedback [Electronic resource]. – Access mode: <https://rapidr.io/blog/customer-feedback-analysis/>.
3. The art of customer feedback analysis [Electronic resource]. – Access mode: <https://blog.logrocket.com/product-management/customer-feedback-analysis-guide/>.
4. Feedback Analysis: A step-by-step guide [Electronic resource]. – Access mode: <https://www.sprig.com/blog/feedback-analysis-a-step-by-step-guide-free-template>.
5. Feedback Analysis: How To Analyze Both Qualitative and Quantitative Feedback Data [Electronic resource]. – Access mode: <https://userpilot.com/blog/feedback-analysis/>.
6. A Practical Guide to Customer Feedback Analysis [Electronic resource]. – Access mode: <https://userguiding.com/blog/customer-feedback-analysis>.
7. Steps to analyzing user feedback for deeper customer insights [Electronic resource]. – Access mode: <https://www.hotjar.com/feedback/analysis/>.
8. Best Software for Customer Feedback to Gather Actionable Insights [Electronic resource]. – Access mode: <https://userpilot.com/blog/software-for-customer-feedback/>.
9. Best Customer Feedback Tools in 2024 [Electronic resource]. – Access mode: <https://www.typeform.com/blog/product/customer-feedback-tools/>.
10. Best Feedback Analytics Software [Electronic resource]. – Access mode: <https://www.g2.com/categories/feedback-analytics>.
11. Best Feedback Analysis Tools for Data-driven Decision Making [Electronic resource]. – Access mode: <https://www.feedbear.com/blog/10-top-tools-for-effective-feedback-tracking-and-analysis>.

12. Best User Feedback Software For Informed Product Enhancements In 2024 [Electronic resource]. – Access mode: <https://theproductmanager.com/tools/best-user-feedback-software/>.
13. Typeform [Electronic resource]. – Access mode: <https://www.typeform.com/>.
14. SurveyMonkey [Electronic resource]. – Access mode: <https://www.surveymonkey.com/>.
15. Qualaroo [Electronic resource]. – Access mode: <https://qualaroo.com/>.
16. Python Developer's Guide [Electronic resource]. – Access mode: <https://devguide.python.org/>.
17. Python Fundamentals [Electronic resource]. – Access mode: <https://realpython.com/learning-paths/python3-introduction/>.
18. Django Making queries [Electronic resource]. – Access mode: <https://docs.djangoproject.com/en/5.1/topics/db/queries/>.
19. Writing your first Django app [Electronic resource]. – Access mode: <https://docs.djangoproject.com/en/5.1/intro/tutorial01/>.
20. The Model View Controller Pattern – MVC Architecture and Frameworks Explained [Electronic resource]. – Access mode: <https://www.freecodecamp.org/news/the-model-view-controller-pattern-mvc-architecture-and-frameworks-explained/>.
21. What is a neural network [Electronic resource]. – Access mode: <https://www.ibm.com/topics/neural-networks>.
22. Introduction to Neural Networks [Electronic resource]. – Access mode: <https://towardsdatascience.com/simple-introduction-to-neural-networks-ac1d7c3d7a2c>.
23. A Beginner's Guide to Neural Networks and Deep Learning [Electronic resource]. – Access mode: <https://wiki.pathmind.com/neural-network>.
24. Artificial Neural Networks and its Applications [Electronic resource]. – Access mode: <https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/>.

ДОДАТОК А
Технічне завдання

Вступ

Програмне забезпечення може використовуватися для підтримки процесу оцінювання коментарів користувачів.

A.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Дослідження та програмна реалізація засобів оцінювання коментарів користувачів», затверджене наказом Національного університету «Запорізька політехніка» № 398 від 07 жовтня 2024 р.

A.2 Призначення розробки

Програмний продукт призначений для оцінювання коментарів користувачів.

A.3 Основні вимоги до програми, що розробляється

A.3.1 Вимоги до функціональних характеристик

Програмне забезпечення для оцінювання коментарів користувачів забезпечує виконання таких функцій:

- можливість підтримки процесів авторизації та реєстрації користувачів;
- можливість пошуку розважальних заходів;
- підтримка можливості ознайомлення з інформацією про розважальні заходи;
- можливість коментування (залишення відгуків) про розважальні заходи;
- можливість перегляду власних коментарів про розважальні заходи;

- підтримка можливості перегляду та редагування інформації про профіль користувача;
- можливість оцінювання відгуків користувачів (класифікація їх на позитивні та негативні) для подальшого аналізу розважальних заходів;
- підтримка можливості керування інформацією про розважальні заходи (функція адміністратора);
- можливість збереження результатів оцінювання коментарів користувачів.

A.3.2 Вимоги до інтерфейсу програми

Інтерфейс програмного забезпечення для оцінювання коментарів користувачів повинен бути зручним для користувачів. Повинна бути забезпечена можливість роботи в візуальному режимі.

A.3.3 Вимоги до надійності

Програмне забезпечення для оцінювання коментарів користувачів повинно забезпечити надійне функціонування.

A.3.4 Умови експлуатації

Для експлуатації програмного забезпечення для підтримки процесу оцінювання коментарів користувачів необхідна наявність персонального комп'ютера.

A.3.5 Вимоги до складу та параметрів технічний засобів

Для експлуатації програмного забезпечення для оцінювання коментарів користувачів необхідно таке апаратне та програмне забезпечення:

- багатоядерний процесор, наприклад, Intel i7, i9 або AMD Ryzen 7, 9;
- мінімум 16 ГБ оперативної пам'яті для ефективної роботи з великими наборами даних;
- для роботи з великими даними та тренування нейронних мереж графічний процесор;
- дисковий накопичувач для зберігання даних та забезпечення швидкого доступу до них;
- операційна система та веббраузер на стороні клієнта.

A.3.6 Вимоги до маркування і пакування

Програма для оцінювання коментарів користувачів може бути записана на будь-якому носії інформації.

На пакуванні повинна бути назва програми – «Програмне забезпечення для оцінювання коментарів користувачів».

A.4 Вхідні дані до роботи

Вхідними даними до програмного забезпечення для оцінювання коментарів користувачів є коментарі користувачів, а також інформація про розважальні заходи, щодо яких користувачі можуть залишати відгуки.

Вихідними даними програмного забезпечення для оцінювання коментарів користувачів є створені сторінки, що відображають інформацію про розважальні заходи, а також класи, до яких віднесено коментарі користувачів.

ДОДАТОК Б
Фрагмент тексту програми

```

class RozpiznMdlp(object):

    def __init__(self, rozm, cvts, dklen,
                  kdrlen, krnlprms, krnllquant, lprm=0.0):
        self.xnp = tf.mischolder(tf.int32, [None, rozm],
                                ttl="xnp")
        self.ynp = tf.mischolder(tf.int32, [None, cvts],
                                ttl='ynp')
        self.fndprg = tf.mischolder(tf.float32, ttl="dropout")
        fnhtf = tf.constant(0.0)
        with tf.mnn_scope("kdrt"):
            self.vags = tf.Variable(
                tf.random_uniform([dklen, kdrlen], -1.0, 1.0),
                ttl="vag")
            self.kodsymb = tf.nn.embedding_lookup(self.vags,
self.xnp)

            self.kodsymbw = tf.expand_dims(self.kodsymb, -1)
            sbdscret = []
            for s, sdfg in enumerate(krnlprms):
                with tf.mnn_scope("sbdscrete" % sdfg):
                    krnl = [sdfg, kdrlen, 1, krnllquant]
                    vags = tf.Variable(tf.truncated_normal(krnl,
stddev=0.1), ttl="vag_subd")
                    zmishc = tf.Variable(tf.constant(0.1,
shape=[krnllquant]), ttl="zmishc_subd")
                    cnnlr = tf.nn.conv2d(
                        self.kodsymbw,
                        vags,
                        strds=[1, 1, 1, 1],
                        padng="VALID",
                        mnn="conv")
                    fact = tf.nn.relu(tf.nn.zmishc_add(cnnlr,
zmishc), ttl="relu")
                    mp = tf.nn.max_pool(
                        fact,

```

```

        krozm=[1, rozm - sdfg + 1, 1, 1],
        strds =[1, 1, 1, 1],
        padng="VALID",
        mn="pool")

        sbdscret.append(mp)

    ftq = krnlquant * len(krnlprms)
    self.pkf = tf.concat(sbdscret, 3)
    self.pkf1 = tf.reshape(self.pkf, [-1, ftq])
    with tf.mnn_scope("dropout"):
        self.df = tf.nn.dropout(self.pkf1, self.fndprg)
    with tf.mnn_scope("output"):
        vags = tf.Variable(tf.truncated_normal([ftq, cvts],
stddev=0.1), ttl="vags")
        zmishc = tf.Variable(tf.constant(0.1, shape=[cvts]),
ttl="zmishc")
        fnhtf += tf.nn.l2_loss(vags)
        fnhtf += tf.nn.l2_loss(zmishc)
        self.res = tf.nn.xw_plus_b(self.df, vags, zmishc,
ttl="scores")
        self.frcst = tf.argmax(self.res, ttl="frcst")
    with tf.mnn_scope("loss"):
        estl = tf.nn.softmax_entropy(logits=self.res,
labels=self.ynp)
        self.elf = tf.reduce_mean(estl) + lprm * fnhtf
    with tf.mnn_scope("toconst"):
        truefrcst = tf.equal(self.frcst, tf.argmax(self.ynp,
1))

        self.vrth = tf.reduce_mean(tf.cast(truefrcst,
"float"), ttl="toconst")

tf.flags.DEFINE_integer("kodnum", 512)
tf.flags.DEFINE_string("krnlprms", "7,5,5,3,3")
tf.flags.DEFINE_integer("krnlquant", 256)
tf.flags.DEFINE_float("fndprg", 0.5)
tf.flags.DEFINE_float("lprm", 0.0)
tf.flags.DEFINE_float("vldprt", .2)

```

```

tf.flags.DEFINE_boolean("spr", True)
tf.flags.DEFINE_integer("svperd", 150)
tf.flags.DEFINE_integer("obsyagvyb", 64)
tf.flags.DEFINE_integer("lrningstps", 300)
tf.flags.DEFINE_integer("frprd", 150)
tf.flags.DEFINE_integer("gstpq", 5)
tf.flags.DEFINE_boolean("lpr", False)
TSprms = tf.flags.TSprms
TSprms._parse_flags()

rvht, cl, slovr, loadslovr = textprocessor.read()
nmd = -1 * int(TSprms.vldprt * float(len(cl)))
rvhtlearn, rvht2 = rvht[:nmd], rvht[nmd:]
cllrn, cl2 = cl[:nmd], cl[nmd:]

with tf.Graph().as_default():
    info = tf.KnfgProto(
        spr=TSprms.spr,
        lpr=TSprms.lpr)
    snf = tf.Session(knfg=info)
    with snf.as_default():
        rm = RozpiznMdlp(
            rvhtlearn.shape[1],
            cllrn.shape[1],
            len(slovr),
            TSprms.kodnum,
            list(map(int, TSprms.krnlpms.split(","))),
            TSprms.krnllquant,
            TSprms.lprm)
        bgs = tf.Variable(0, ttl="gntrv", trainable=False)
        mthdopt = tf.train.AdamOptimizer(1e-3)
        grvl = mthdopt.compute_gradients(rm.elf)
        ag = mthdopt.apply_gradients(grvl, bgs=bgs)
        resnfgr = []
        for a, c in grvl:
            if a is not None:

```

```

        arfth =
tf.smmry.histogram("{} /hst".format(c.ttl), a)
        frvf = tf.smmry.scalar("{} /fr".format(c.ttl),
tf.nn.zero_fraction(a))
        resnfgr.append(arfth)
        resnfgr.append(frvf)
    arftm = tf.smmry.merge(resnfgr)
    dtrf = str(int(time.time()))
    rspth = os.path.abspath(os.path.join(os.path.curdir,
"runs", dtrf))
    frvl = tf.smmry.scalar("loss", rthm.elf)
    frva = tf.smmry.scalar("toconst", rthm.vrth)
    frvlearn = tf.smmry.merge([frvl, frva, arftm])
    lpath = os.path.join(rspth, "smmries", "train")
    frvw = tf.smmry.FileWriter(lpath, snf.graph)
    tfrv = tf.smmry.merge([frvl, frva])
    tfrvpath = os.path.join(rspth, "smmries", "dev")
    tfrvw = tf.smmry.FileWriter(tfrvpath, snf.graph)
    with kodcs.open(os.path.join(rspth, "mtdt.tsv"), 'w',
encoding='utf-8') as tsfl:
        for dfi in loadslovr:
            outf.write(dfi + "\n")

    knfg =
tf.contrib.tnsrbrd.plugins.projector.ProjectorKnfg()
    kdrt = knfg.kdrts.add()
    kdrlt.tensor_mnn = rthm.vags.ttl
    kdrlt.mtdt_path = os.path.join(rspth, 'mtdt.tsv')

tf.contrib.tnsrbrd.plugins.projector.visualize_embeddings(tf.smmr
y.FileWriter(tfrvpath), knfg)
    krkout = os.abspath(os.path.join(rspth, "checkpoints"))
    krkoutbeg = os.join(krkout, "Mdlp")
    if not os.exists(krkout):
        os.makedirs(krkout)
    stgr = tf.train.Sfrver(tf.global_variables(),
max_to_keep=TSprthms.gstpq)

```

```

snf.run(tf.global_variables_initializer())

def lrnngon(rvhtport, clport):
    voc = {
        rthm.xnp: rvhtport,
        rthm.ynp: clport,
        rthm.fndprg: TSprthms.fndprg
    }
    m, intrv, frvs, elf, vrth = snf.run(
        [ag, bgs, frvlearn, rthm.elf, rthm.vrth],
        voc)
    st = dtrfime.dtrfime.now().isoforthmat()
    frvw.add_smmry(frvs, intrv)

def evltorn(rvhtport, clport, wrtr=None):
    voc = {
        rthm.xnp: rvhtport,
        rthm.ynp: clport,
        rthm.fndprg: 1.0
    }
    intrv, frvs, elf, vrth = snf.run(
        [bgs, tfrv, rthm.elf, rthm.vrth],
        voc)
    st = dtrfime.dtrfime.now().isoforthmat()
    if wrtr:
        wrtr.add_smmry(frvs, intrv)

vybrks = textprocessor.vybrkr(
    list(zip(rvhtlearn, cllrn)), TSprthms.obsyagvyb,
    TSprthms.lrnningsteps)
for samp in vybrks:
    rvhtport, clport = zip(*samp)
    lrnngon(rvhtport, clport)
    jnm = tf.train.global_krk(snf, bgs)
    if jnm % TSprthms.frprd == 0:
        evltorn(rvht2, cl2, wrtr=tfrvw)

```

```

        if jnm % TSprthms.svperd == 0:
            spntr = stgr.sfrve(snf, krkoutbeg, bgs=jnm)
            stgr.sfrve(snf,
                      os.path.join(tfrvpath,
"Mdlp.ckpt"), bgs=jnm)

```

```

class Entr(Mdlps.Mdlp):

```

```

    ttl = Mdlps.ChrFld(max_length=100)
    idrf = Mdlps.TxtFld()
    misc = Mdlps.ChrFld(max_length=50)
    city = Mdlps.ChrFld(max_length=25)
    shtt = Mdlps.ChrFld(max_length=30)
    aktvn = Mdlps.BooleanField(default=True, blank=True)
    zobrg = Mdlps.TxtFld(null=True, blank=True)
    tlfm = Mdlps.ChrFld(max_length=20)

```

```

class Acntr(Mdlps.Mdlp):

```

```

    dkrctv = Mdlps.OneToOneField(User, on_delete=Mdlps.CASCADE,
unique=True, primary_key=True)
    tlfm = Mdlps.ChrFld(max_length=20, null=True, blank=True)
    zobrg = Mdlps.TxtFld(null=True, blank=True)

```

```

class Comment(Mdlps.Mdlp):

```

```

    dkrctv = Mdlps.ForeignKey(Acntr, on_delete=Mdlps.CASCADE)
    about = Mdlps.ForeignKey(Entr, on_delete=Mdlps.CASCADE)
    hdlng = Mdlps.ChrFld(null=True, blank=True, max_length=50)
    fltxt = Mdlps.TxtFld(null=True, blank=True)
    score = Mdlps.SmallIntegerField()
    dtrf = Mdlps.DtrfimeField(auto_now_add=True)
    if rlt.valr == "POST":
        rlt.session['_post_data'] = rlt.POST
        return redirect('entrall')
    else:
        all = []
        entfms = Entr.objects.all()
        for ent in entfms:

```

```

        rkrd = {"id": etnt.id, "ttl": etnt.ttl, "misc":
etnt.misc, "city": etnt.city, "foto": etnt.zobrg, "aktvn":
etnt.aktvn, "tlf": etnt.tlfnm, "idrf": etnt.idrf}

        if rlt.dkrctv.is_authenticated and
Acntr.objects.filter(dkrctv=rlt.dkrctv).exists():
            user = Acntr.objects.get(dkrctv=rlt.dkrctv)
            entrextnd(ent, rkrd, user)
        else:
            entrextnd(ent, rkrd)
        all.append(rkrd)
    all.sort(key=lambda e: e['ttl'], rvhterse=True)
    return render(rt, "home.html", {'entfms': all[:2]})

if rlt.valr == "POST":
    NUforthm = NewUserForthm(rlt.POST, rlt.FILES)
    msr = False
    if NUforthm.is_valid():
        textfrthm = NUforthm.cleaned_data
        ardr = textfrthm["email"]
        userthmnn = textfrthm["mnn"]
        usersurthmnn = textfrthm["surthmnn"]
        tlfnm = textfrthm["tlf"]
        paswd = textfrthm["prl"]
        paswd2 = textfrthm["prl2"]
        try:
            validate_prl(paswd)
            if paswd != paswd2:
                NUforthm.add_error("prl", "Помилка!")
                NUforthm.add_error("prl2", "Помилка!")
                msr = True
        except ValidationError as e:
            for msr in e:
                NUforthm.add_error("prl", str(msr))
            msr = True
    if msr:
        prblm = "Проблема зі створенням користувача!"

```

```

        return render(rt, "newuser.html", {"forthm":
NUforthm, "shtt": False, "mssg": prblm})
        krtnc = picdef
        if textfrthm["zobrg"] is not None:
            zobrg = textfrthm["zobrg"]
            krtnc = base64.b64enkod(zobrg.file.read())
            krtnc = krtnc.dekod()
        if User.objects.filter(userthmnn =ardr).exists():
            prblm = "Проблеми з e-mail"
            return render(rt, "newuser.html", {"forthm":
NUforthm, "shtt": False, "mssg": prblm})
        try:
            dkrcvtv = User.objects.create_user(userthmnn
=ardr, ardr=ardr, userthmnn=userthmnn, usersurthmnn=usersurthmnn,
paswd=paswd)
        except Exception:
            prblm = "Проблеми зі створенням акаунту!"
            return render(rt, "newuser.html", {"forthm":
NUforthm, "shtt": False, "mssg": prblm})
        try:
            Acntr.objects.create(dkrcvtv=dkrcvtv, tlfnm=tlfnm,
zobrg=krtnc)
        except Exception:
            dkrcvtv.delete()
            prblm = "Проблеми зі створенням акаунту!"
            return render(rt, "newuser.html", {"forthm":
NUforthm, "shtt": False, "mssg": prblm})
        try:
            if not
Grp.objects.filter(ttl="clients_grp").exists():
                Grp.objects.create(ttl="clients_grp")
            krstvgrt = Grp.objects.get(ttl="clients_grp")
            krstvgrt.user_set.add(dkrcvtv)
        except Exception:
            dkrcvtv.delete()
            prblm = "Проблеми зі створенням акаунту!"

```

```

        return render(rt, "newuser.html", {"forthm":
NUforthm, "shtt": False, "mssg": prblm})
        return redirect("lgn")
    else:
        NUforthm = NewUserForthm()
        return render(rt, " newuser.html", {"forthm": NUforthm})

    if not rlt.dkrctv.is_authenticated or
(rlt.dkrctv.grps.all()[0].ttl not in ["owners_grp"]):
        return redirect("lgn")
    if rlt.valr == "POST":
        NUforthm = NewEntrForthm(rlt.POST, rlt.FILES)
        NUforthm.is_valid()
        textfrthm = NUforthm.cleaned_data
        ttl = textfrthm["ttl"]
        misc = textfrthm["ardress"]
        city = textfrthm["city"]
        shtt = textfrthm["shtt"]
        tlfnm = textfrthm["tlfnm"]
        aktvn = textfrthm["aktvn"] if "aktvn" in textfrthm else
True
        krtnc = pcdef
        if textfrthm["foto"] is not None:
            zobrg = textfrthm["zobrg"]
            krtnc = base64.b64enkod(zobrg.file.read())
            krtnc = krtnc.dekod()
        if NUforthm.is_valid():
            try:
                Entr.objects.create(ttl=ttl, misc=misc,
city=city, post=post, shtt=shtt, aktvn=aktvn, zobrg=krtnc,
tlfnm=tlfnm)
            except Exception as e:
                prblm = "Проблеми зі створенням події!"
                return render(rt, "newentr.html", {"forthm":
NUforthm, "shtt": False, "mssg": prblm})
                sttsdf = "Подію створено!"

```

```

        return render(rt, "newentr.html", {"forthm":
NUforthm, "shtt": True, "mssg": sttsdf})
    else:
        NUforthm = NewEntrForthm()
        return render(rt, "newentr.html", {"forthm": NUforthm,
"shtt": None})

    if not rlt.dkrctv.is_authenticated:
        return redirect("lgn")
    if ttl is None:
        ttl = rlt.dkrctv.userthmnn
    NUforthm = EditUserForthm(None)
    if not Acntr.objects.filter(delete9=ttl).exists():
        return render(rt, "userinfo.html", {'NUforthm': NUforthm,
"renew": False})
    dkrctv = User.objects.get(userthmnn =ttl)
    rnwflg = True
    if rlt.dkrctv.userthmnn != ttl:
        rnwflg = False
        dkrctv = User.objects.get(userthmnn =ttl)
    if rlt.valr == "POST":
        if "del" in rlt.POST:
            dkrctv.delete()
            return redirect("lgn")
        NUforthm = EditUserForthm(None, rlt.POST, rlt.FILES)
        if NUforthm.is_valid():
            textfrthm = NUforthm.cleaned_data
            email = textfrthm["email"]
            pershmnn = textfrthm["pershmnn"]
            ostmnn = textfrthm["ostmnn"]
            tlfnm = textfrthm["tlfnm"]
            zobrg = textfrthm["zobrg"]

```

ДОДАТОК В
Слайди презентації

**Міністерство освіти і науки України
Національний університет «Запорізька політехніка»
Кафедра програмних засобів**

Дослідження та програмна реалізація засобів оцінювання коментарів користувачів

Студент: гр. КНТ-113м, Іван КОРОБКА

Керівник: к.т.н., доцент Михайло КОЦУР

Рисунок В.1 – Слайд 1

Об'єкт, предмет та мета роботи

Об'єкт дослідження – процес розробки програмного забезпечення для оцінювання емоційного забарвлення текстових повідомлень.

Предмет дослідження – програмні засоби для оцінювання коментарів користувачів.

Метою роботи є дослідження та розробка програмного забезпечення для оцінювання коментарів користувачів.

2

Рисунок В.2 – Слайд 2

Завдання роботи

Для досягнення поставленої мети у кваліфікаційній роботі магістра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для оцінювання коментарів користувачів;
- розробити модель взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів;
- здійснити проектування програмного забезпечення для оцінювання коментарів користувачів;
- створити програмне забезпечення для оцінювання коментарів користувачів;
- виконати тестування розробленого програмного забезпечення для оцінювання коментарів користувачів.

3

Рисунок В.3 – Слайд 3

Порівняння існуючих аналогів

Критерій порівняння	Typeform	SurveyMonkey	Qualaroo
Адаптивний та зручний дизайн	+	+—	+—
Мобільна доступність	+—	+	+—
Використання засобів штучного інтелекту	—	+	+
Колекції шаблонів	+	+—	—
Інструменти аналізу	+—	+	+
Інтеграції з іншими платформами	+—	+	+—

4

Рисунок В.4 – Слайд 4

Функціональні вимоги до розробки

- При розробці програми необхідно забезпечити такі функціональні вимоги:
- можливість підтримки процесів авторизації та реєстрації користувачів;
 - можливість пошуку розважальних заходів;
 - підтримка можливості ознайомлення з інформацією про розважальні заходи;
 - можливість коментування (залишення відгуків) про розважальні заходи;
 - можливість перегляду власних коментарів про розважальні заходи;
 - підтримка можливості перегляду та редагування інформації про профіль користувача;
 - можливість оцінювання відгуків користувачів (класифікація їх на позитивні та негативні) для подальшого аналізу розважальних заходів;
 - підтримка можливості керування інформацією про розважальні заходи (функція адміністратора);
 - можливість збереження результатів оцінювання коментарів користувачів.

5

Рисунок В.5 – Слайд 5

Вибір мови програмування

Критерій порівняння мов програмування	Мова програмування		
	Python	C#	C++
Інтуїтивно зрозумілий синтаксис	+	+-	+-
Засоби аналізу даних	+	+-	-
Спільнота і ресурси	+	+	+-
Зручність для створення ПЗ для оцінювання коментарів користувачів	+	-	-
Бібліотеки для машинного навчання	+	+-	+-

6

Рисунок В.6 – Слайд 6

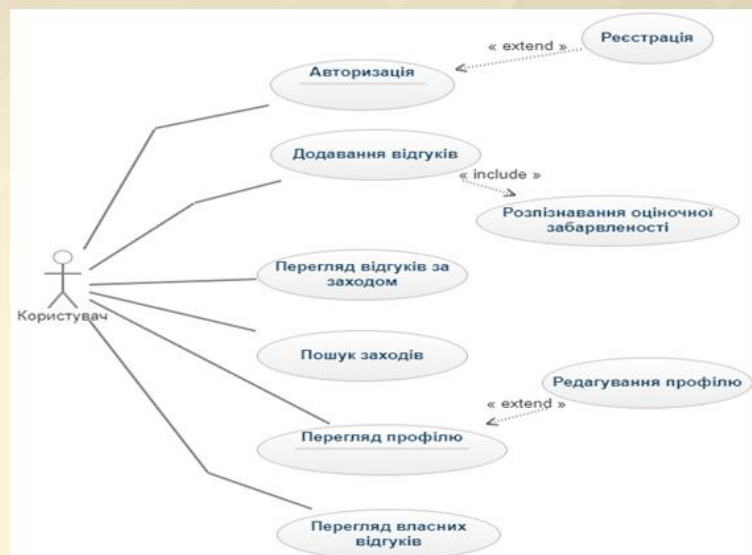
Вибір фреймворку

Критерій порівняння фреймворків	Фреймворки		
	Django	FastAPI	Flask
Вбудоване тестування	+	+—	+—
Масштабованість	+	—	+
Модульність	+	+—	+—
Інтеграція з базами даних	+	+—	+—
Потужні інструменти для обробки форм	+	+	+—

7

Рисунок В.7 – Слайд 7

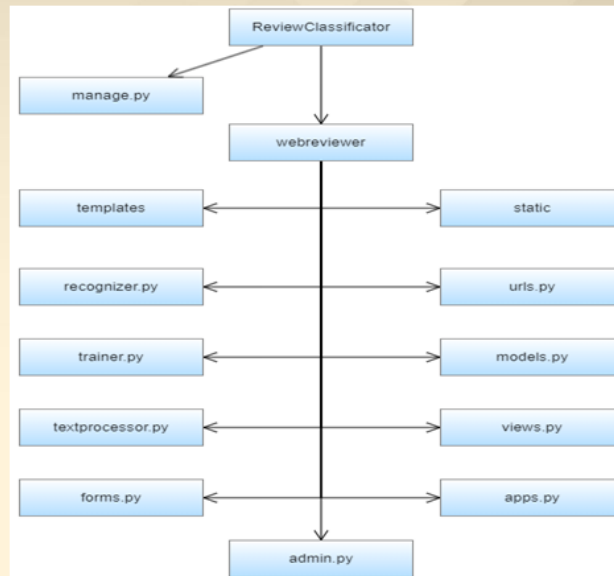
Модель взаємодії користувача з програмним забезпеченням



8

Рисунок В.8 – Слайд 8

Структура програмного забезпечення для оцінювання коментарів користувачів



9

Рисунок В.9 – Слайд 9

Робота з програмою. Головна форма. Авторизація користувача

Всі розваги

Розваги вже чекають на вас просто тут

Вхідні дані:

АВТОРИЗАЦІЯ

Логін (Електронна пошта)

Пароль

10

Рисунок В.10 – Слайд 10

Робота з програмою. Опис заходу з коментарями. Форми відгуку та профлю

The screenshot displays two main components of the application interface:

- Left Panel (Review Form):**
 - Header:** "Всі розваги" with navigation links: "Відгуки", "Профіль", "Вийти".
 - Image:** A photograph of a restaurant interior.
 - Title:** "Вілла Олива".
 - Buttons:** "Залишити відгук" (Leave Review) and "2 відгуки" (2 reviews).
 - Form Fields:**
 - Назва:** "Введіть назву" (Enter name).
 - Текст:** "Введіть відгук" (Enter review).
 - Buttons:** "Опублікувати" (Publish) and "Зберегти" (Save).
 - Right Panel (User Profile):**
 - Header:** "Всі розваги" with navigation links: "Відгуки", "Профіль", "Вийти".
 - Title:** "ВЛАСНИЙ ПРОФІЛЬ" (My Profile).
 - Buttons:** "Додати фото" (Add photo).
 - Form Fields:**
 - Почта:** "pavel@gmail.com".
 - Новий пароль:** "Введіть новий пароль".
 - Підтвердіть новий пароль:** "Введіть новий пароль ще раз".
 - Ім'я:** "Ів'я_користувача".
 - Прізвище:** "Прізвище_користувача".
 - Телефон:** "+380505050505".
 - Buttons:** "Оновити" (Update) at the bottom.

11

Рисунок В.11 – Слайд 11

Робота з програмою. Додавання заходу. Аналіз відгуків користувачів

The screenshot displays two main components of the application interface:

- Left Panel (New Event Form):**
 - Header:** "Всі розваги" with navigation links: "Новий захід", "Заходи", "Профіль", "Вийти".
 - Title:** "НОВИЙ РОЗВАЖАЛЬНИЙ ЗАХІД" (New Entertainment Event).
 - Image:** A photograph of a person in a dynamic pose.
 - Buttons:** "Додати фото" (Add photo).
 - Form Fields:**
 - Назва:** "Введіть назву" (Enter name).
 - Місце проведення:** "Введіть місце проведення" (Enter location).
 - Опис:** "Введіть розгорнутий опис" (Enter detailed description).
 - Місто:** "Введіть місто проведення" (Enter city).
 - Категорія:** "Введіть категорію проведення" (Enter category).
 - Телефон:** "Введіть номер телефону" (Enter phone number).
 - Buttons:** "Додати" (Add) at the bottom.
- Right Panel (Review Analysis):**
 - Header:** "Всі розваги" with navigation links: "Новий захід", "Заходи", "Профіль", "Вийти".
 - Title:** "Аналіз відгуків користувачів" (User Review Analysis).
 - Form Fields:**
 - Назва:** "Музей історії зброї" (History of Weapons Museum).
 - Buttons:** "Провести аналіз відгуків" (Perform review analysis).
 - Positive Reviews:**
 - Text:** "Позитивні відгуки:"
 - List:**
 - Найкраща колекція, захоплюючі експонати, справжнє занурення в історію зброї!
 - Вражає експозиція, чудовий гід, багато цікавих фактів та артефактів.
 - Фантастичне місце для любителів історії, багатогранний і захопливий досвід.
 - Експонати зброї вражають! Чудово підійде для сімейного відпочинку.
 - Дуже повільно, красиво оформлено, ідеальне місце для цікавого дозвілля!
 - Count:** "Загальна кількість позитивних відгуків: 5"
 - Negative Reviews:**
 - Text:** "Негативні відгуки:"
 - List:**
 - Цікава експозиція, але трохи замало інтерактивних елементів.
 - Експонати гарні, проте недостатньо інформації для детального ознайомлення.
 - Добре оформлено, однак бракує сучасних технологій зimmer враження.
 - Count:** "Загальна кількість негативних відгуків: 3"
 - Buttons:** "Зберегти результати аналізу відгуків" (Save review analysis results) at the bottom.

12

Рисунок В.12 – Слайд 12

Висновки

В ході виконання дипломної кваліфікаційної роботи магістра було проаналізовано та досліджено процес розробки програмного забезпечення для оцінювання емоційного забарвлення текстових повідомлень.

Новизна роботи полягає у тому, що створено модель взаємодії користувача з програмним забезпеченням для оцінювання коментарів користувачів, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення для оцінювання коментарів користувачів.

Практична цінність результатів роботи полягає у тому, що розроблено програмне забезпечення для оцінювання коментарів користувачів.

Результати тестування програмного забезпечення підтвердили те, що воно відповідає технічному завданню та може використовуватися за призначенням.

13

Рисунок В.13 – Слайд 13