

Вінницький національний технічний університет  
(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматики  
(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук  
(повна назва кафедри (предметної, циклової комісії))

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА:**

на тему:

**«Інформаційна технологія проведення онлайн-опитувань»**

Виконав: студент 2-го курсу, групи  
2КН-22м

спеціальності 122 «Комп'ютерні науки»  
(шифр і назва напрямку підготовки, спеціальності)

Тютюнов В. П.  
(прізвище та ініціали)

Керівник: к.т.н., доцент кафедри КН

Колодний В. В.  
(прізвище та ініціали)

«07» 12 2023 р.

Опонент: к.т.н., доцент кафедри САІТ

Крижановський Є. М.  
(прізвище та ініціали)

«07» 12 2023 р.

Допущено до захисту

Завідувач кафедри КН

д.т.н., проф. Яровий А. А.  
(прізвище та ініціали)

«08» 12 2023 р.

Вінниця ВНТУ – 2023 рік

Вінницький національний технічний університет  
Факультет інтелектуальних інформаційних технологій та автоматизації  
Кафедра Комп'ютерних наук  
Рівень вищої освіти II-й (магістерський)  
Галузь знань – 12 «Інформаційні технології»  
Спеціальність – 122 «Комп'ютерні науки»  
Освітньо-професійна програма – «Системи штучного інтелекту»



**ЗАТВЕРДЖУЮ**  
Завідувач кафедри КН  
Д.т.н., проф. Яровий А. А.  
«29» 08 2023р.





### ЗАВДАННЯ

#### НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Тютюнову Владиславу Петровичу

1. Тема роботи: Інформаційна технологія проведення онлайн-опитувань  
керівник роботи: Колодний В.В. к.т.н., доц. каф. КН  
( прізвище, ім'я, по батькові, науковий ступінь, вчене звання)  
затверджені наказом вищого навчального закладу “18” 09 2023 року №247
2. Термін подання студентом роботи: 13.11.2023
3. Вихідні дані до роботи: Кількість альтернатив – 2-10; кількість можливих градацій при опитуванні – до 10; кількість опитаних – довільна; час проведення опитування – довільний; реалізація інформаційної технології – у вигляді загальнодоступного сайту; мова програмування – об'єктно-орієнтована.
4. Зміст текстової частини (перелік питань, які потрібно розкрити): Аналіз сучасних програмних засобів для проведення онлайн-опитувань. Проектування інформаційної технології проведення онлайн-опитувань. Програмна реалізація інформаційної технології проведення онлайн-опитувань. Тестування розробленого програмного забезпечення. Розробка інструкції користувача.
5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень): Загальна структурна схема, діаграма класів, діаграма прецедентів, діаграма діяльності, алгоритми функціонування модулю створення опитування та модулю анкетування, результат роботи розробленого програмного забезпечення.


6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1, 2, 3	Колодний В.В., к.т.н., доц. каф. КН		
4	Ратушняк О.Г., к.т.н., доц. каф. ЕПВМ		

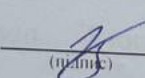
7. Дата видачі завдання 29.08.2023

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного роботи	Строк виконання етапів роботи	Примітка
1	Аналіз сучасних програмних засобів для проведення онлайн-опитувань	01.09.23-07.09.23	
2	Проектування інформаційної технології проведення онлайн-опитувань	08.09.23-24.09.23	
3	Програмна реалізація інформаційної технології проведення онлайн-опитувань	25.09.23-15.10.23	
4	Оформлення пояснювальної записки і графічного матеріалу	21.10.23-29.10.23	
5	Попередній захист	30.10.23-05.11.23	
6	Остаточний захист	06.11.23-10.11.23	

Студент   
(підпис)

Тютюнов В.П.  
(прізвище та ініціали)

Керівник роботи   
(підпис)

Колодний В.В.  
(прізвище та ініціали)

## АНОТАЦІЯ

УДК 004.8

Тютюнов В. П. Інформаційна технологія проведення онлайн-опитувань. Магістерська кваліфікаційна робота за спеціальності 122 – комп'ютерні науки, освітня програма – системи штучного інтелекту. Вінниця: ВНТУ, 2023. 122 с.

На укр. мові. Бібліогр.: 26 назв; рис.: 44; табл.: 3.

Дана магістерська кваліфікаційна робота присвячена розробці інформаційної технології проведення онлайн-опитувань. Були розглянуті особливості проведення онлайн-опитувань та програми аналоги. Було розглянуто та обрано типи шкал оцінювання та їх візуальні представлення. Архітектура розробленої інформаційної технології складається з трьох модулів та десяти класів. Було спроектовано інформаційну технологію проведення онлайн-опитувань, написано мовою програмування TypeScript з використанням таких технологій як: Angular, Firebase, HTML, SCSS. В порівнянні з програмами аналогами, розроблена інформаційна система надає можливість створення власної шкали оцінювання та покращеного аналізу результатів за допомогою візуального відображення з використанням кольорових схем.

Графічна частина складається з 6 плакатів.

У економічному розділі розраховано суму витрат на розробку та виготовлення нового технічного рішення, яка складає 160939 гривень, спрогнозовано орієнтовану величину витрат по кожній з статей витрат, розраховано чистий прибуток, термін окупності витрат для виробника 1 рік та економічний ефект для споживача при використанні даної розробки.

Ключові слова: онлайн-опитування; візуальне ранжування; кольорова шкала; ефективність опитувань; аналіз результатів опитувань.

## **ABSTRACT**

Tiutiunov V. P. Information technology for conducting online surveys. Master's thesis in the specialty 122 – computer science, educational program – artificial intelligence systems. Vinnytsia: VNTU, 2023. 122 p.

In Ukrainian language. Bibliographer: 26 titles; fig.: 44; tables: 3.

This master's thesis is devoted to the development of information technology for conducting online surveys. Peculiarities of conducting online surveys and similar programs were considered. Types of rating scales and their visual representations were considered and selected. The architecture of the developed information technology consists of three modules and ten classes. Information technology for conducting online surveys was designed, written in the TypeScript programming language using such technologies as: Angular, Firebase, HTML, SCSS. In comparison with similar programs, the developed information system provides an opportunity to create your own rating scale and improve the analysis of results with the help of visual display using color schemes.

The graphic part consists of 6 posters.

The economic section calculates the amount of costs for the development and manufacture of a new technical solution, which is 160939 UAH, predicted the estimated cost of each of the cost items, calculated net profit, payback period for the manufacturer 1 year and the economic effect for consumers using this development.

Keywords: online survey; visual ranking; color scale; effectiveness of surveys; analysis of survey result

## ЗМІСТ

ВСТУП .....	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ПРОВЕДЕННЯ ОНЛАЙН-ОПИТУВАНЬ	7
1.1 Особливості проведення онлайн-опитувань .....	7
1.2 Огляд відомих реалізацій програм для проведення онлайн-опитувань .....	8
1.3 Шкали оцінювання в опитування .....	19
1.4 Візуальне представлення результатів .....	21
1.5 Постановка задачі на розробку інформаційної технології проведення онлайн-опитувань. ....	22
1.6 Висновок до розділу 1 .....	23
2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ ОНЛАЙН-ОПИТУВАНЬ.....	24
2.1 Розробка структури інформаційної технології проведення онлайн- опитувань .....	24
2.2 Використання мови моделювання UML в процесі проектування інформаційної технології проведення онлайн-опитувань.....	26
2.3 Моделювання інформаційної технології проведення онлайн-опитувань з використанням мови UML.....	27
2.4 Розробка алгоритмів функціонування інформаційної технології проведення онлайн-опитувань .....	30
2.5 Розробка шкал для інформаційної технології проведення онлайн- опитувань .....	33
2.6 Математична модель інформаційної технології проведення онлайн- опитувань .....	34
2.7 Висновок до розділу 2 .....	37
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ ОНЛАЙН-ОПИТУВАНЬ.....	38
3.1 Обґрунтування вибору інструментів розробки .....	38
3.2 Основні оператори мови програмування TypeScript .....	42

	3
3.3 Особливості середовища Visual Studio Code .....	43
3.4 Програмна реалізація інформаційної технології проведення онлайн-опитувань .....	43
3.5 Тестування розробленої інформаційної технології проведення онлайн-опитувань .....	46
3.6 Приклад практичного використання інформаційної технології проведення онлайн-опитувань .....	57
3.7 Висновки до розділу 3 .....	60
4 ЕКОНОМІЧНА ЧАСТИНА .....	61
4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки .....	61
4.2 Розрахунок витрат на проведення науково-дослідної роботи .....	65
4.3 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором .....	71
4.4 Висновки до розділу 4 .....	75
ВИСНОВКИ .....	76
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	78
Додаток А (обов'язковий) Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень .....	82
Додаток Б (обов'язковий) Лістинг програми .....	83
Додаток В (обов'язковий) Ілюстративна частина .....	105
Додаток Г (довідниковий) Інструкція користувача .....	113
Додаток Д (довідниковий) Довідка впровадження .....	122

## ВСТУП

**Актуальність теми дослідження.** Актуальність проведення досліджень у сфері онлайн-опитувань сьогодні стає все більш важливою в умовах стрімкого розвитку цифрового суспільства. Зростаюча кількість користувачів Інтернету та широке використання мобільних технологій створюють свої власні унікальні виклики та можливості для галузі досліджень. Перехід від традиційних методів збору даних до онлайн-опитувань стає важливим етапом в еволюції досліджень, оскільки дозволяє збирати дані в режимі реального часу, мінімізуючи витрати та забезпечуючи гнучкість у виборі аудиторії для дослідження.

Враховуючи активне використання соціальних мереж, електронної пошти та інших каналів онлайн-комунікації, важливо висвітлити сучасні тенденції та технічні новинки у проведенні опитувань, що дозволяє збирати більш об'єктивні та репрезентативні дані. Активна конкуренція в інформаційному просторі зумовлює необхідність пошуку ефективних інструментів аналізу громадської думки та оцінки відгуків, що робить актуальним розвиток та вдосконалення онлайн-опитувань для найбільш точного та достовірного відображення реальних поглядів та уподобань суспільства.

**Зв'язок роботи з науковими програмами, планами, темами.** Магістерська робота виконана відповідно до напрямку наукових досліджень кафедри комп'ютерних наук Вінницького національного технічного університету 22 К1 «Моделі, методи, технології та пристрої інтелектуальних інформаційних систем управління, економіки, навчання та комунікацій» та плану наукової та навчально-методичної роботи кафедри.

**Мета та завдання дослідження.** Метою дослідження магістерської кваліфікаційної роботи є розширення функціональних можливостей програмного забезпечення для проведення онлайн-опитувань.

Для досягнення поставленої мети необхідно виконати такі задачі:

- аналіз сучасних технологій проведення онлайн-опитування;



- аналіз переваг та недоліків програм аналогів проведення онлайн-опитування;
- розробка алгоритму проведення онлайн-опитування;
- програмна реалізація розробленого алгоритму проведення онлайн-опитування;
- тестування розробленої програми проведення онлайн-опитування та аналіз результатів його роботи;
- економічне підтвердження розробленої програми.

**Об’єкт дослідження** – це процес проведення онлайн-опитувань.

**Предмет дослідження** – це програмні засоби для проведення онлайн-опитувань.

**Наукова новизна одержаних результатів** полягає в такому:

Було вперше запропоновано інформаційну технологію проведення онлайн-опитувань, яка відрізняється від існуючих використанням моделі кольорових схем та їх поєднання з вербально-числовими шкалами, що дозволяє підвищити точність результатів опитування.

**Практичне значення отриманих результатів** полягає в такому:

1. Розроблено алгоритм функціонування модулю анкетування;
2. Розроблено алгоритм функціонування модулю створення опитування;
3. Розроблено програмний модуль анкетування;
4. Розроблено програмний модуль модулю створення опитування.

Розроблена інформаційна технологія використовується в програмному забезпеченні для проведення онлайн-опитувань в компанії «ІТ-СЕРВІС-ВІН» (м. Вінниця) та готується до впровадження у вигляді готового продукту.

**Достовірність теоретичних положень** магістерської кваліфікаційної роботи підтверджується коректністю постановки завдання, посиланнями на сучасні наукові публікації та літературні джерела, порівнянням результатів з відомими програмами аналогами, експериментальними дослідженнями тестування програмної реалізації інформаційної технології проведення онлайн-опитувань.

**Особистий внесок здобувача.** Усі результати, що наведені у магістерській кваліфікаційній роботі, отримані самостійно. У роботах, опублікованих у співавторстві, автору належать такі результати: [1] – розробка системи підтримки проведення онлайн-опитувань з конструюванням індивідуальних шкал.

### **Апробація результатів роботи**

Результати роботи були апробовані на науково-практичній конференції «МОЛОДЬ В НАУЦІ: ДОСЛІДЖЕННЯ, ПРОБЛЕМИ, ПЕРСПЕКТИВИ» (МН-2024). Подана заявка на отримання свідоцтва про реєстрацію авторського права на твір (комп'ютерну програму) для сайту VerbVisPoll [2].

### **Публікації**

За результатами бакалаврської дипломної роботи опублікована теза доповіді на конференції «ІІ Науково-технічна конференція підрозділів Вінницького національного технічного університету (НТКП ВНТУ) (факультет інтелектуальних інформаційних технологій та автоматизації (2022))» (м. Вінниця, Україна) [3].

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ПРОВЕДЕННЯ ОНЛАЙН-ОПИТУВАНЬ

## 1.1 Особливості проведення онлайн-опитувань

Анкетування є важливим інструментом у дослідженнях та ефективним методом для аналізу поглядів та тенденцій різних груп респондентів [3]. Сучасний розвиток технологій встановлює нові стандарти, і онлайн-опитування стають необхідністю у проведенні наукових досліджень.

Онлайн-опитування – це розміщення питань анкети в електронному вигляді та наданні респонденту віддаленого доступу до неї. Можна навести такі переваги онлайн-опитування:

- Економія ресурсів: Перехід до онлайн-формату дозволяє значно ефективніше використовувати як час, так і фінансові ресурси, що стає ключовим аспектом в епоху інформаційних технологій.

- Широке охоплення: Можливість проведення опитувань в електронному вигляді дозволяє охоплювати різні соціальні групи та географічні регіони, роблячи дослідження більш репрезентативним і доступним.

- Рівень довіри: Онлайн-опитування вирішують проблему відсутності прямого контакту з інтерв'юером, забезпечуючи респондентам високий рівень конфіденційності та, таким чином, сприяючи більш відкритому висловленню думок [1].

- Контроль: Застосування фільтрів та обов'язкових полів у онлайн-опитуваннях допомагає уникнути неповних анкет, забезпечуючи якісний збір даних та їх аналіз.

- Комунікація з респондентами: Отримані результати можуть слугувати засобом зворотного зв'язку, що сприяє взаєморозумінню та є додатковим стимулом для участі респондентів у наступних етапах досліджень [5].

Ці особливості визначають необхідність ретельного вивчення при розробці інформаційної технології проведення онлайн-опитувань, щоб врахувати потреби та зручність користувачів у цифровому дослідженні.

Не зважаючи на очевидні переваги, слід відзначити, що використання онлайн-опитувань не виключає можливості їх поєднання з традиційними методами. Комбінування анкетування з іншими формами дослідження може підвищити об'єктивність результатів та розширити обсяг отримуваної інформації. Такий гібридний підхід розкриває нові перспективи для наукового співтовариства та практичних застосувань у соціальних та наукових дослідженнях.

Також існують виклики, що потребують уважного вивчення та управління. Серед них - проблеми безпеки даних, неоднозначність у сприйнятті певних питань, а також ризики втрати зв'язку з аудиторією. Вирішення цих аспектів стане ключем до подальшого успішного розвитку онлайн-опитувань та їх інтеграції в сучасні дослідження.

Технологічний прогрес не тільки визначає формат проведення опитувань, але й активно впливає на саму методологію досліджень. Нові програмні продукти для онлайн-опитувань стають інструментами для науковців, який розширює можливості та дозволяє реалізувати більш складні завдання в рамках наукових досліджень.

## **1.2 Огляд відомих реалізацій програм для проведення онлайн-опитувань**

Розглянемо наступні реалізації для проведення онлайн-опитувань: Microsoft Forms, SurveyMonkey, Google Forms, Survio та CollExpConstr. Це сайти для проведення онлайн-опитувань різної широти використання.

Microsoft Forms - це інструмент для створення опитувань та анкет в електронному вигляді, який надається в рамках екосистеми Microsoft 365. Він відрізняється простотою використання та інтеграцією з іншими сервісами

Microsoft, такими як SharePoint і Teams. Користувачі можуть швидко створювати опитування, використовуючи різні типи питань, і аналізувати результати в зручному форматі. Незважаючи на свою обмежену функціональність порівняно з іншими аналогами, Microsoft Forms відмінно підходить для швидких опитувань та збору даних в офісному середовищі. Сторінка створення опитування Google Forms зображена на рисунку 1.1.

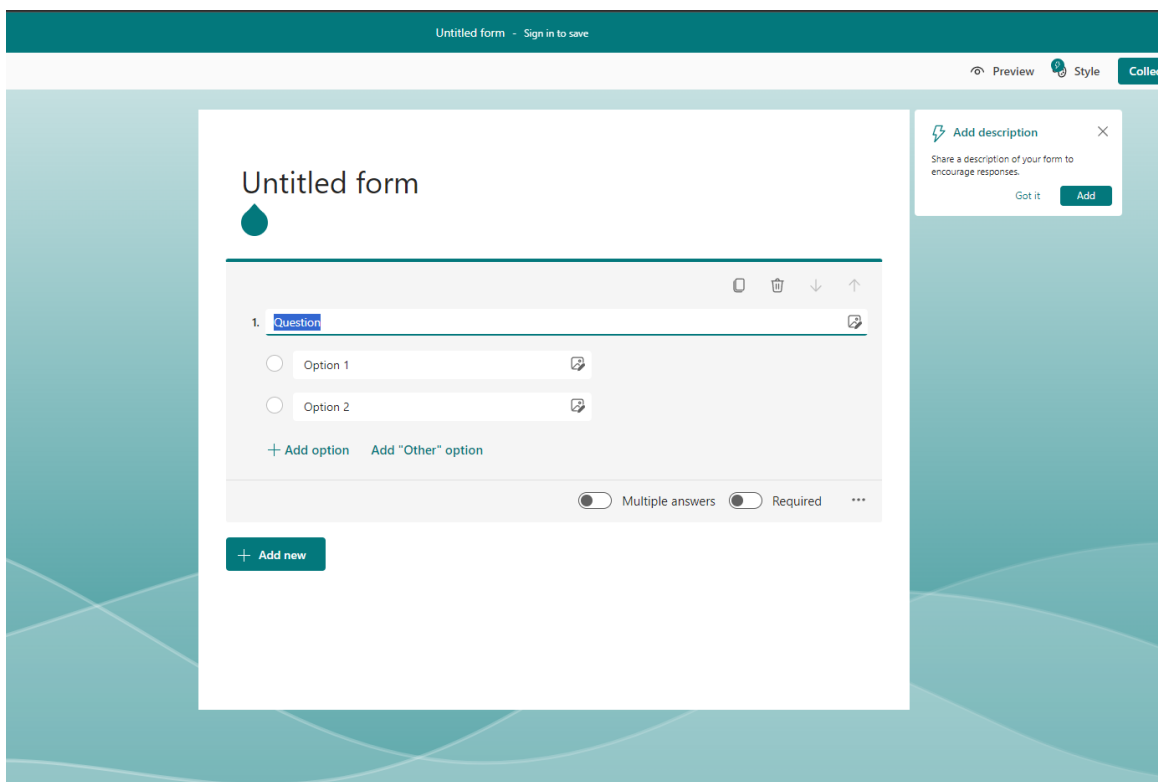


Рисунок 1.1 – Сторінка створення опитування Microsoft Forms

На рисунку 1.2 зображена сторінка проходження опитування. Бачимо, що застосунок має зручний та інтуїтивний інтерфейс на який потрібно орієнтуватись в подальшій розробці. Також на рисунку 1.3 зображена сторінка результатів опитування, на якій можна побачити графіки візуалізації кількості обраних альтернатив для кожного питання. Ідея візуалізації є ключовою в цій роботі, але в Microsoft Forms кольори обираються у випадковому порядку, тобто їх не можна обрати при створенні опитування, через що вони не можуть виконувати репрезентативну роль.

Untitled quiz

1. Question 2

Option 1

Option 2

2. Question 2

Option 1

Option 2

**Submit**

Never give out your password. [Report abuse](#)

This content is created by the owner of the form. The data you submit will be sent to the form owner. Microsoft is not responsible for the privacy or security practices of its customers, including those of this form owner. Never give out your password.

Powered by Microsoft Forms |  
The owner of this form has not provided a privacy statement as to how they will use your response data. Do not provide personal or sensitive information. | [Terms of use](#)

Рисунок 1.2 – Сторінка проходження опитування в Microsoft Forms

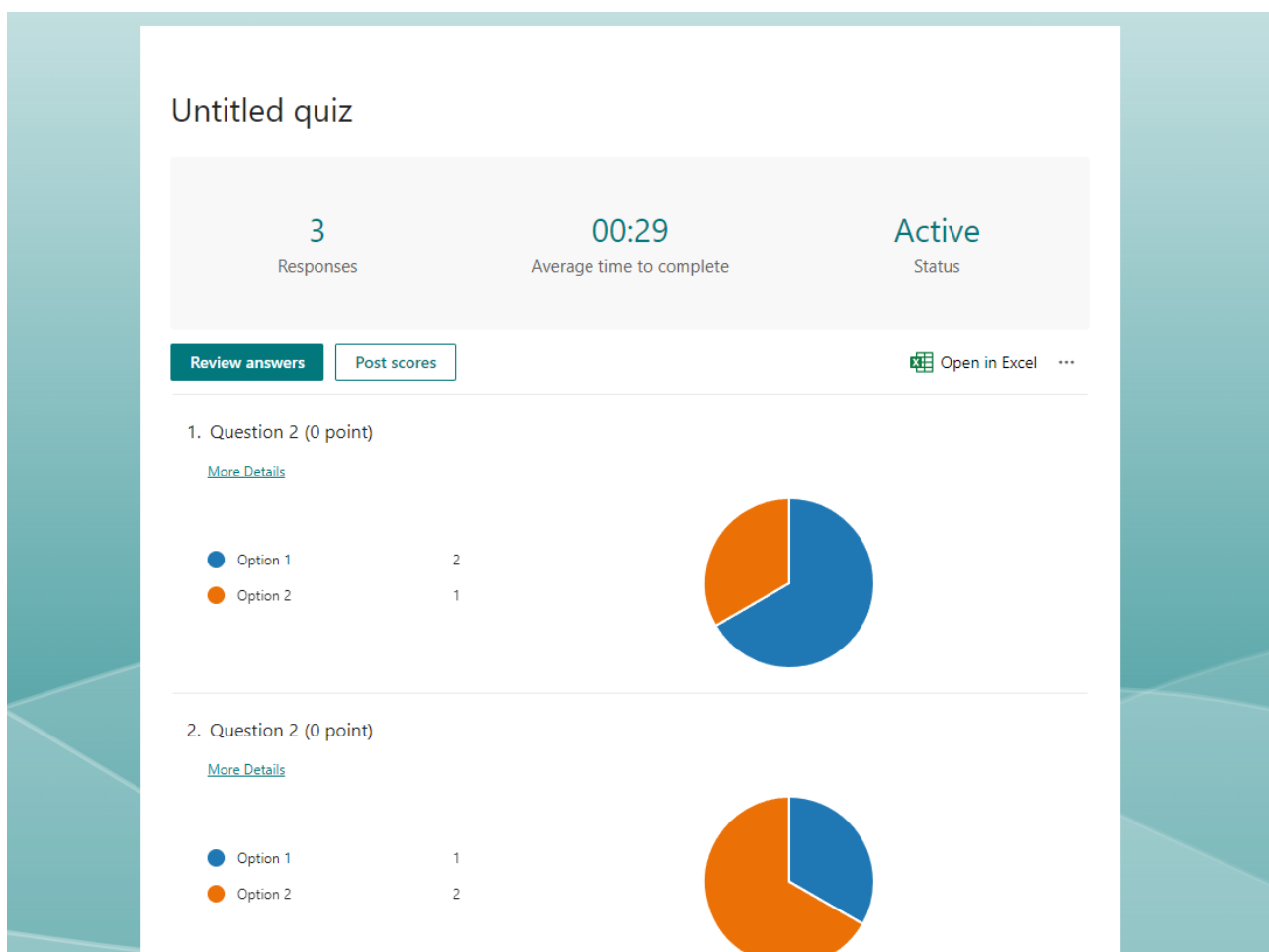


Рисунок 1.3 – Сторінка результатів опитування Microsoft Forms

SurveyMonkey – це веб-сервіс для створення та проведення онлайн-опитувань. Відомий своєю простотою використання та широким набором функцій для налаштування анкет. SurveyMonkey дозволяє користувачам створювати різноманітні типи питань, використовувати власні шаблони опитувань та отримувати аналіз результатів у зручному вигляді. Сервіс також надає можливість інтеграції з іншими популярними платформами та інструментами для полегшення процесу збору та аналізу даних. Незалежно від того, чи це опитування для бізнесу, освіти чи досліджень, SurveyMonkey забезпечує зручний і ефективний спосіб отримання відгуків та даних від респондентів.. Сторінки створення та проходження опитування SurveyMonkey зображені на рисунках 1.4 та 1.5.

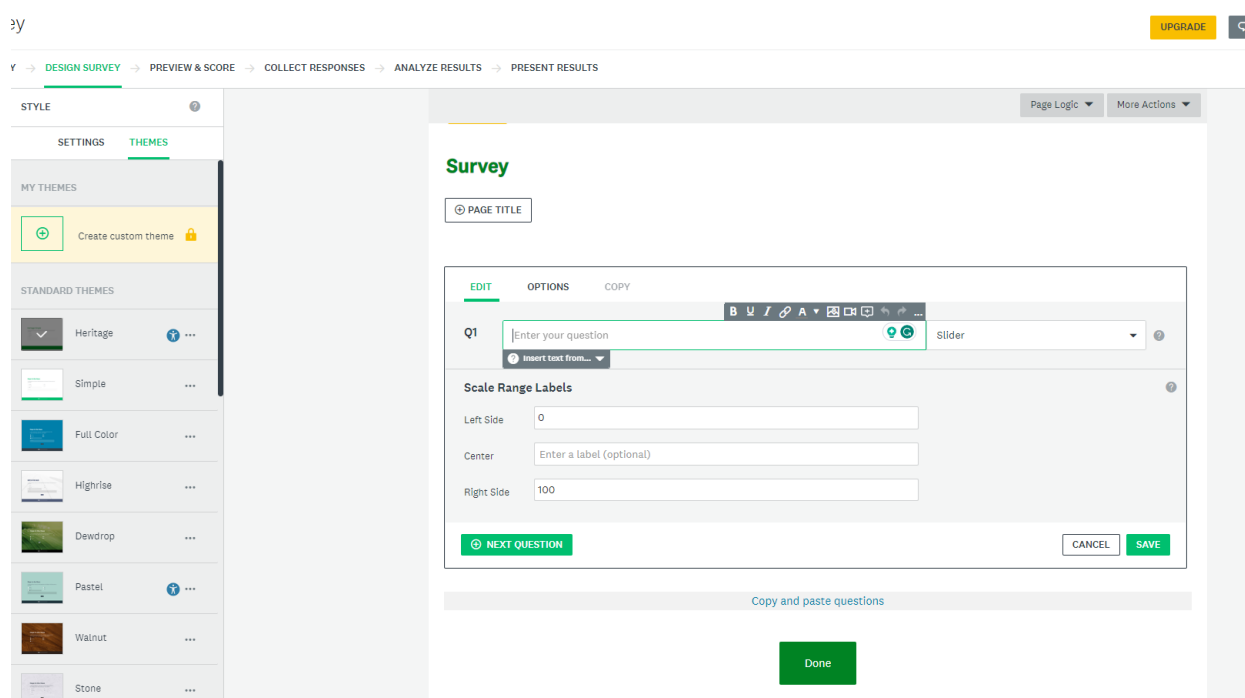


Рисунок 1.4 – Сторінка створення опитування SurveyMonkey

### Type your title here

Write a description of your survey here. Select any question below to change it. Then add questions as needed.

1. Type a multiple choice question here.

- Type your first answer choice.
- Type another here. A respondent can only choose one answer.
- Use the + or - icon on the right to add or remove an answer choice.

2. Type a checkbox question here. Respondents can choose more than one answer.

- Type an answer choice here.
- Type another answer choice.
- Want to offer a "None of the above" option? Look for the checkbox below.

3. Ask a question that requires a short text response here.

Done

### Рисунок 1.5 – Сторінка проходження опитування SurveyMonkey

Сторінка результатів опитування в SurveyMonkey дає можливість для більше детального аналізу ніж Microsoft Forms, але в той же час виглядає неінтуїтивною та незручно для звичайного користувача. Крім того, кольори також не мають жодного сенсу. Сторінка результатів опитування SurveyMonkey зображена на рисунку 1.6.



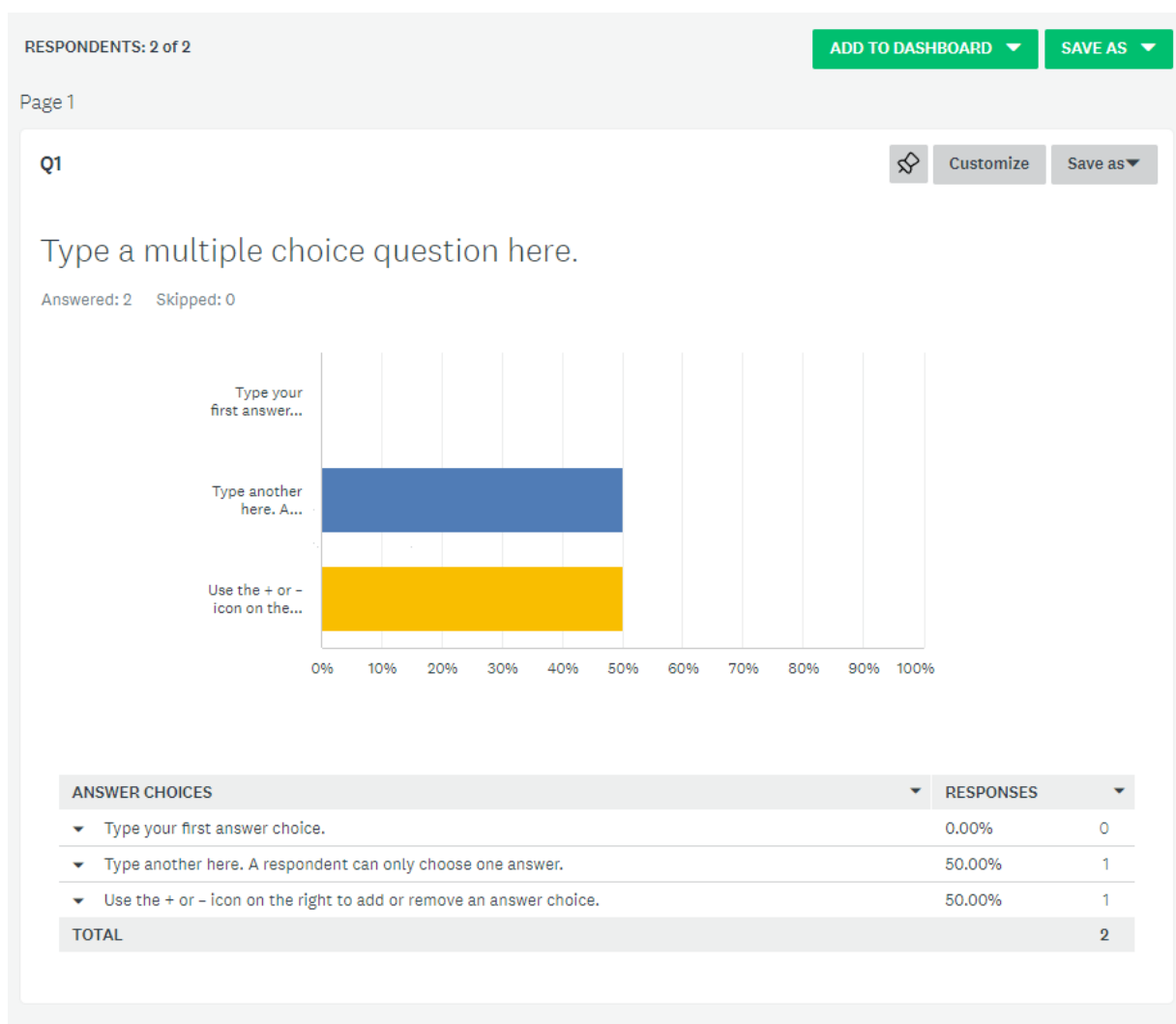


Рисунок 1.6 – Сторінка результатів опитування SurveyMonkey

Google Forms – найпопулярніший сайт для онлайн-анкетування. Він є досить простим у використанні, реалізує зручний та інтуїтивно зрозумілий дизайн. У ньому можна створювати широкий спектр типів питань, такі як: питання з одним варіантом відповіді, питання з декількома варіантами відповіді, питання з вільним текстом. Також можна створити питання з правильною відповіддю, що дозволяє проводити не тільки опитування, а й тестування [6]. Серед мінусів Google Forms можна визначити важкість, майже неможливість, змінення мови користування та відсутність детального аналізу результатів опитування. Сторінки створення та проходження опитування Google Forms зображені на рисунках 1.7 та 1.8.

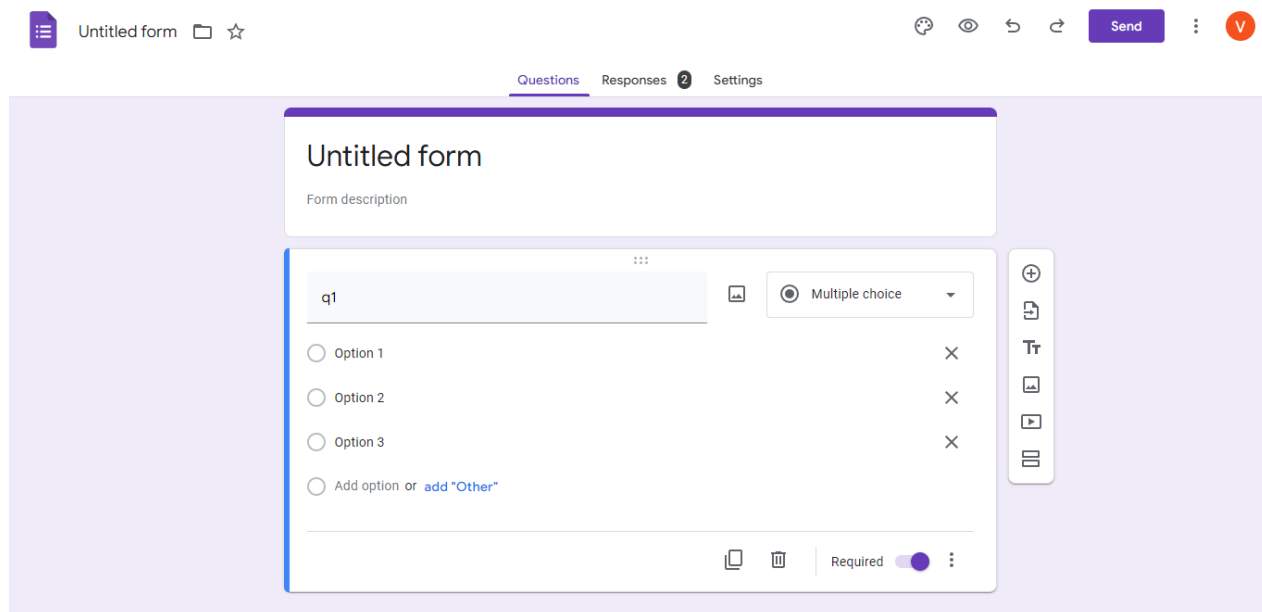


Рисунок 1.7 – Сторінка створення опитування Google Forms

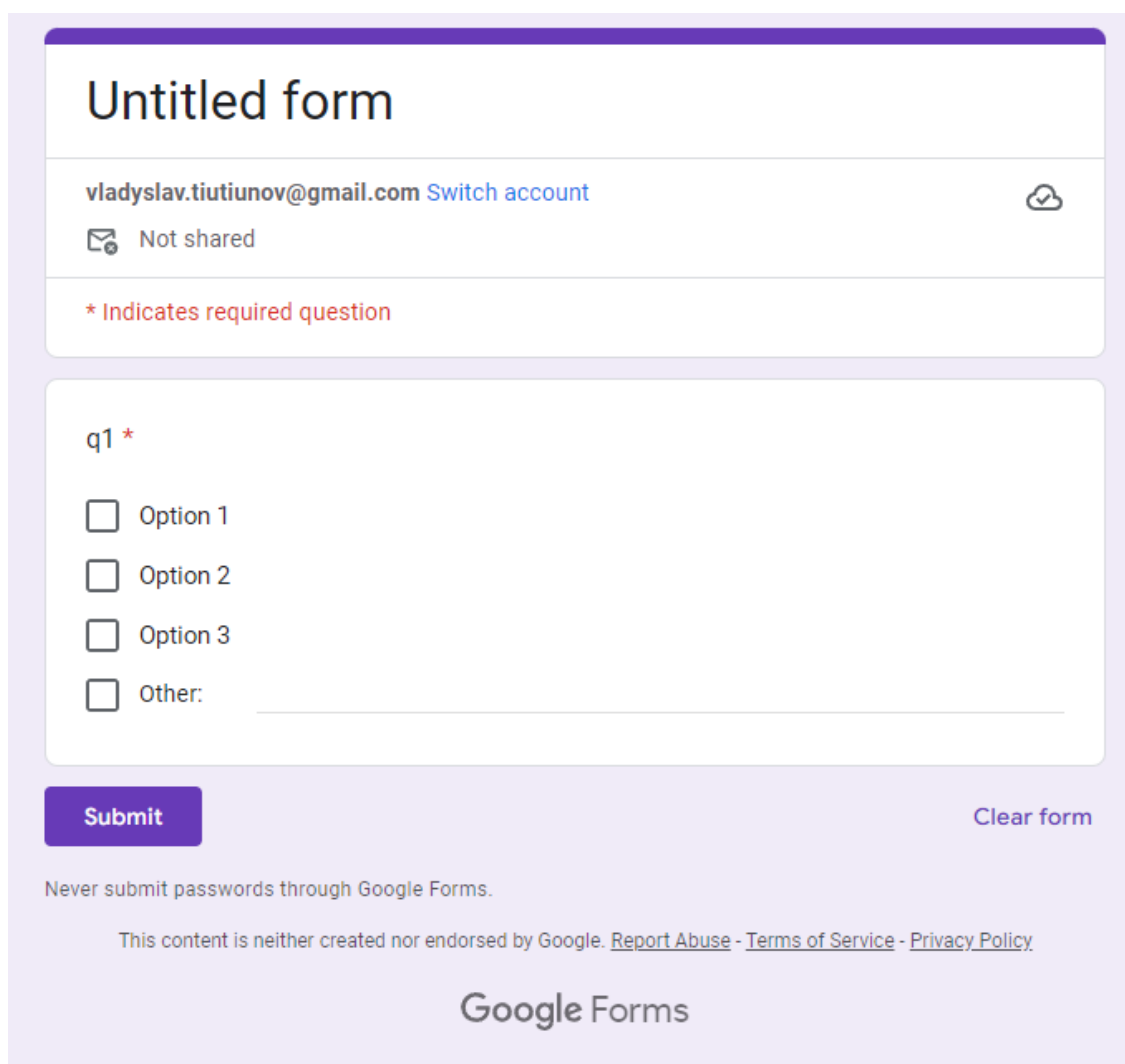


Рисунок 1.8 – Сторінка проходження опитування Google Forms

Сторінка результатів опитування Google Forms має зручний інтерфейс та корисні функції, наприклад різні опції поширення, але, в той же час, не надає достатніх інструментів для аналізу результатів опитування. Крім того візуалізація представлена лише у вигляді графіків без застосування кольорових схем. Сторінка результатів опитування Google Forms зображена на рисунку 1.9.

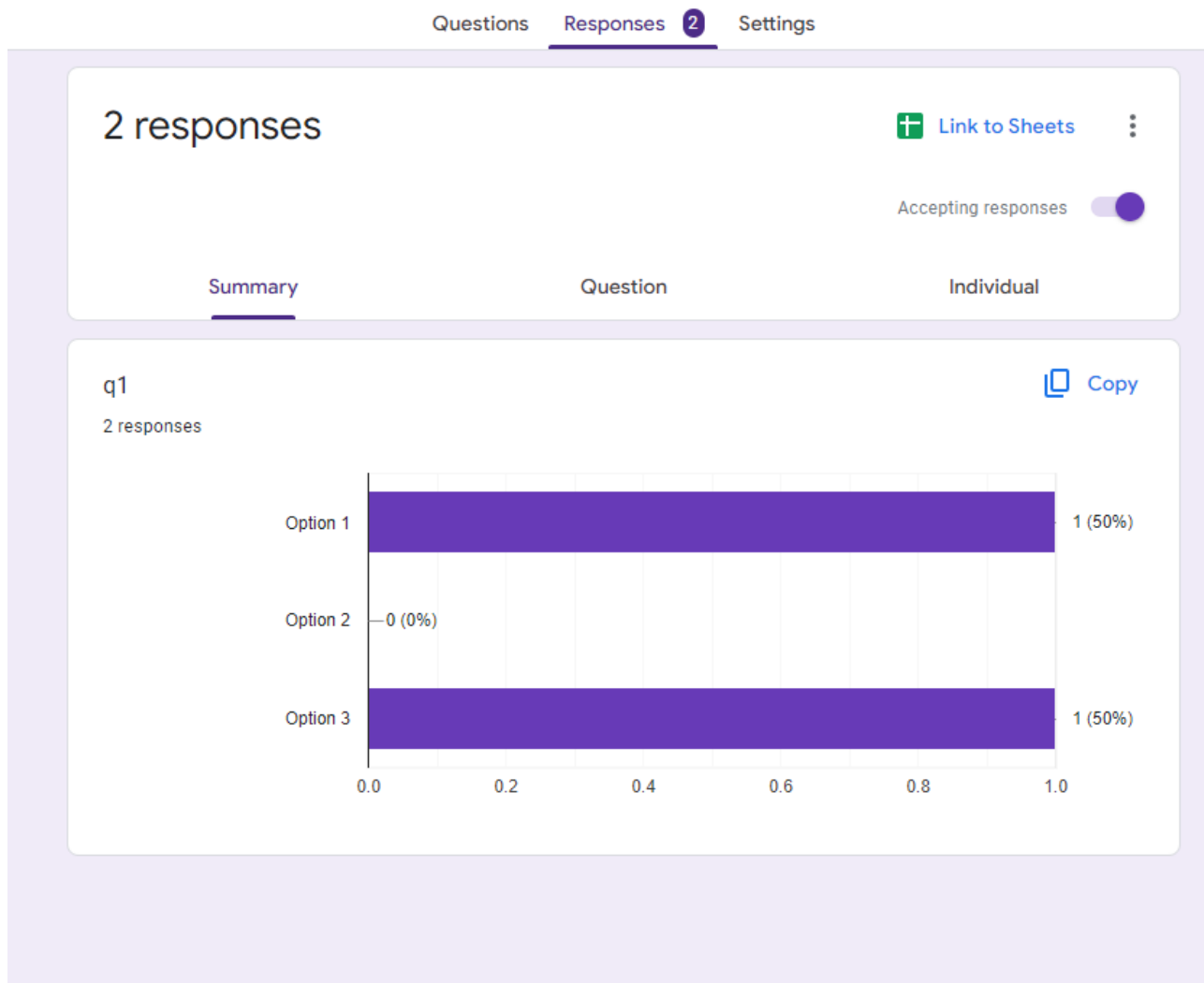


Рисунок 1.9 – Сторінка результатів опитування Google Forms

Survio – більш цільовий продукт, напрямлений на користувачів які зацікавленні у проведенні онлайн-опитування з детальним аналізом результатів. Серед плюсів можна визначити гнучкість у створенні питань, простоту використання та можливість детального аналізу результатів опитування.

Мінусами є обов'язкова реєстрація та просування покупки преміум-акаунту [7]. Сторінки створення та проходження опитування Survio зображені на рисунках 1.10 та 1.11.

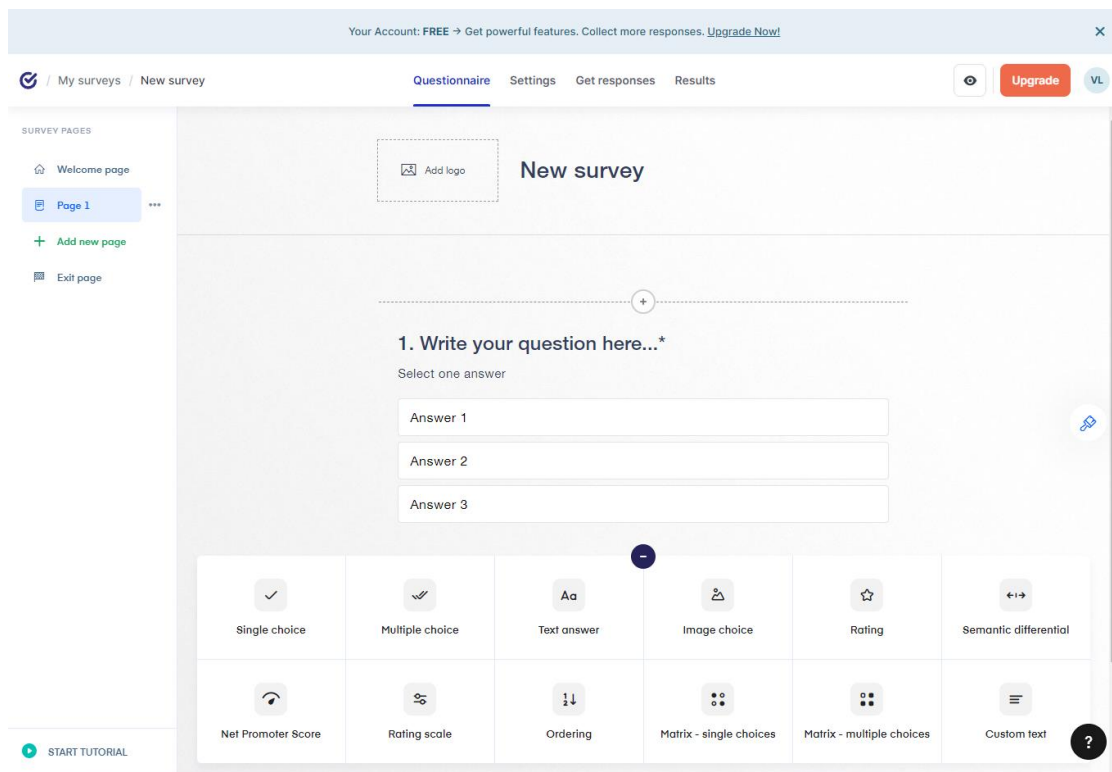


Рисунок 1.10 – Сторінка створення опитування Survio

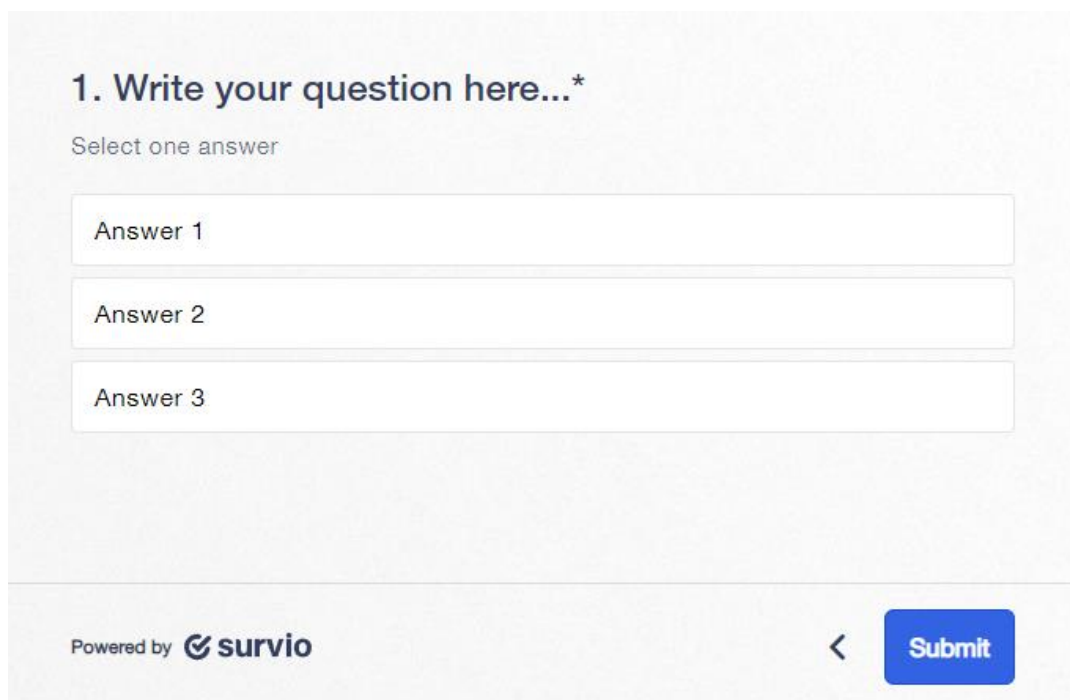


Рисунок 1.11 – Сторінка проходження опитування Survio

Survio також надає зручні інструменти роботи з результатами опитування, але не надає інструментів аналізу результатів опитування крім одного графіку. Сторінка результатів опитування Survio зображено на рисунку 1.12.

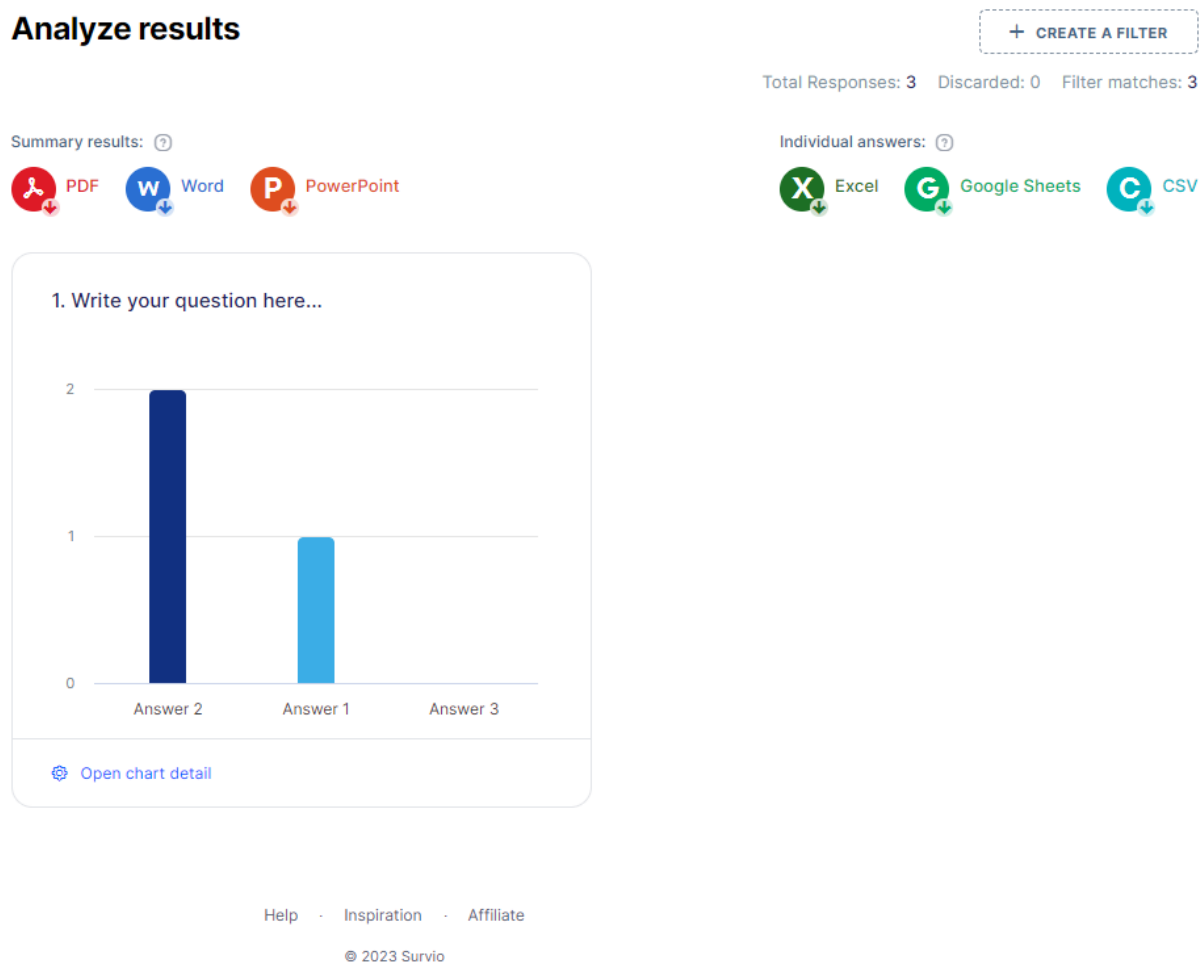


Рисунок 1.12 – Сторінка результатів опитування Survio

CollExpConstr – сайт, розроблений студентами ВНТУ, націлений на проведення онлайн-опитувань серед студентів. На сайті є можливість для створення питань з дуже гнучким вибором шкали, що дозволяє налаштувати оцінювання питань під свої потреби. Також на ресурсі присутній досить детальний аналіз результатів опитування. Серед мінусів можна визначити обов'язкову реєстрацію та неможливість змінення мови [5]. Сторінка створення опитування CollExpConstr зображена на рисунку 1.13.

Кількість альтернатив (1-15):

2

Альтернатива 1

Альтернатива 2

**Шкала для оцінювання:**

Категорично проти	-2	<span style="color: red;">■</span>
Проти	-1	<span style="color: lightcoral;">■</span>
Байдуже	0	<span style="color: yellow;">■</span>
За	1	<span style="color: lightgreen;">■</span>
Категорично за	2	<span style="color: green;">■</span>

**Конструктор шкал**

Назва градації	Значення	Колір
<input type="text"/>	<input type="text"/>	<input type="text" value="■"/>
<input type="text"/>	<input type="text"/>	<input type="text" value="■"/>

Рисунок 1.13 – Сторінка створення опитування CollExpConstr

У порівнянні з попередніми програмами аналогами, CollExpConstr пропонує шкали з застосуванням кольорових схем, але має обмежені можливості аналізу даних.

Складемо таблицю 1.1 – порівняння розглянутих реалізацій програм для проведення онлайн опитувань:

Таблиця 1.1. Порівняння відомих реалізації програм для проведення онлайн-опитування.

	Детальний аналіз результатів	Створення шкали оцінювання	Відсутність реєстрації	Візуальне представлення результатів
Microsoft Forms	-	-	+	-
SurveyMonkey	-	-	-	+
CollExpConstr	+	+	-	+
Google Forms	-	-	+	-
Survio	+	-	-	-

Отже, проаналізувавши популярні програм для проведення онлайн-опитувань, можна зробити висновки щодо вимог до реалізації програмного забезпечення, а саме: детальний аналіз результатів опитування, можливість створення шкали оцінювання, візуальне представлення результатів та відсутність реєстрації.

### 1.3 Шкали оцінювання в опитування

Під час проведення онлайн-опитувань виникає важлива задача вибору шкал для оцінювання відповідей респондентів. Шкали є ключовим елементом, який визначає спосіб вимірювання та подання результатів. Розглянемо які є шкали оцінювання та їхнє практичне використання у контексті онлайн-опитувань:

– Вербальна шкала:

Респондентам пропонується вибрати слово або фразу, яка найкраще відображає їхню думку чи почуття. Наприклад, для вимірювання задоволеності вибором можуть бути слова від "дуже незадоволений" до "дуже задоволений".

– Шкала Лікерта:

Різновид вербальної шкали, в якому вимірюється ступінь погодження або непогодження респондента з певним твердженням. Це один з найпоширеніших типів шкал. Зазвичай використовується п'яти- або семирівнева шкала, де респондент обирає ступінь згоди чи незгоди, наприклад: повністю не згоден; не згоден; байдуже; згоден; повністю згоден.

– Семантичний диференціал:

Це шкала, яка вимірює відчуття або оцінку об'єкта за допомогою протилежних слів або фраз, розташованих на кінцях шкали. Наприклад, шкала може включати пари слів, такі як "сильно - слабо" чи "позитивно - негативно".

– Шкала частоти:

Респондентам пропонується обрати частоту подій, виражену у числовій формі. Наприклад, "скільки разів на тиждень ви використовуєте конкретний продукт?" з можливими варіантами від "жодного разу" до "щоденно".

– Числова шкала:

Шкала, яка складається з набору чисел, які відображають оцінку респондента відповідно альтернативі. Наприклад: 1; 2; 3; 4; 5.

Кожен тип шкали має свої переваги і недоліки. Деякі з них надають більш деталізовану інформацію, тоді як інші є більш простими для аналізу та розуміння. Важливо ретельно підходити до вибору шкал для забезпечення точності та релевантності результатів дослідження.

Важливим аспектом є правильний вибір шкал для конкретного дослідження. Дослідники повинні враховувати характер питань, тип даних, які потрібно отримати, та відповідність шкали конкретній аудиторії. Неправильний вибір може спричинити спотворення результатів та невірне розуміння відповідей респондентів. Неправильний вибір може призвести до спотворення результатів та невірного розуміння відповідей респондентів. Наприклад, використання складних або неінтуїтивних шкал може призвести до помилок та неточностей у відповідях, що вплине на достовірність результатів.



Отже, ми розглянули найпопулярніші типи шкали оцінювання, та проаналізували важливість правильного вибору шкали, та наслідків при поганому виборі, що допоможе успішно розробити та застосувати шкали у подальшому дослідженні.

#### 1.4 Візуальне представлення результатів

Важливим етапом аналізу отриманих результатів опитувань є їхнє візуальне представлення. Застосування кольорових схем для шкал оцінювання може значно полегшити сприйняття результатів та зробити аналіз більш доступним для респондентів та дослідників. Розглянемо приклади використання кольорових представлень на прикладі вже відомих нам шкал.

Однією з ефективних стратегій є використання кольорових позначень для числових шкал типу «-2; -1; 0; 1; 2» або шкали Лікерта. Наприклад, відтінки від червоного до зеленого можуть слугувати інтуїтивним індикатором від негативного до позитивного значення. Це полегшує сприйняття даних та швидко передає основний зміст відповіді. Приклад використання кольорової схеми зображено на рисунку 1.14.

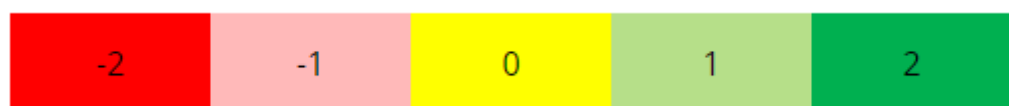


Рисунок 1.14 – Приклад використання кольорової схеми для візуального представлення шкали оцінювання

У випадку порядкових шкал, таких як «1; 2; 3; 4; 5», можна використовувати відтінки одного кольору або плавний перехід від світлого до темного. Наприклад, від світло-зеленого до темно-зеленого. Це дозволяє візуально виділити різницю між рівнями оцінок та надає користувачеві чітке уявлення про певні аспекти відповідей. Приклад візуального представлення порядкової шкали зображено на рисунку 1.15.



Рисунок 1.15 – Приклад використання кольорової схеми для візуального представлення порядкової шкали оцінювання

Використання кольорів для візуального представлення результатів опитувань може значно покращити сприйняття інформації та полегшити процес аналізу. Важливо дотримуватися стандартів дизайну та уникати агресивних кольорових рішень, щоб забезпечити зручність та об'єктивність в сприйнятті результатів опитувань, а також враховувати психологічний вплив кольорів на сприйняття інформації. Деякі кольори можуть асоціюватися з певними емоціями або станами. Наприклад, червоний може вказувати на негатив або увагу, а зелений - на позитив або успіх.

### **1.5 Постановка задачі на розробку інформаційної технології проведення онлайн-опитувань.**

Ціллю даної роботи є розробка інформаційної технології проведення онлайн-опитувань. Застосунок розробляється для використання у браузері.

Додаток має реалізовувати такі функції: створення онлайн-опитувань, створення власної шкали опитування, приєднання до існуючого опитування, можливість відповіді на поставленні питання, перегляд детальних результатів опитування, зміну мови користування. Також необхідно провести тестування даної системи на наявність помилок, безпечність та коректність роботи.

Отже, для виконання поставленого завдання необхідно побудувати структурну схему програмного забезпечення, алгоритми функціонування основних модулів програми, побудувати UML-діаграми, які включатимуть основні модулі, додати функції, властивості та методи до кожного класу, а також описати математичну модель та розробити шкали оцінювання для подальшого

використання у системі. Поступово під час розробки розширювати можливості програми та тестувати її.

## **1.6 Висновок до розділу 1**

Отже, у даному розділі було проведено аналіз предметної області проведення онлайн опитувань, а саме – розглянуто особливості проведення онлайн-опитувань, проведено огляд відомих програм-аналогів. Також було проведено аналіз різних типів шкал оцінювання для подальшого вибору оптимального варіанту та розглянуто можливість візуального представлення результатів опитування за допомогою кольорових шкал. Доведено важливість правильного вибору шкали оцінювання та візуалізації результатів опитування. В результаті сформульовано постановку задачі на розробку інформаційної системи.

## **2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ ОНЛАЙН-ОПИТУВАНЬ**

### **2.1 Розробка структури інформаційної технології проведення онлайн-опитувань**

На етапі розробки програмного забезпечення дуже важливо розробити його правильну структуру (архітектуру). Розробка структурної схеми програми є одним з найважливіших етапів у процесі розробки програмного забезпечення з наступних причин:

- неправильний вибір архітектури веде до ризику зриву всього проекту у майбутньому;
- даний етап є базовим для процесу розробки;
- добре продумана архітектура дозволяє легко модифікувати програмний продукт, якщо відбудуться зміни вимог щодо нього.

Під архітектурою розуміється сукупність компонентів програми, і навіть зв'язки й методи організації інформаційного обміну з-поміж них. Першим завданням, яке необхідно вирішити при розробці структурної схеми системи, є завдання визначення складових її компонентів [8].

Інформаційна технологія проведення онлайн-опитувань можна умовно розділити на три частини з різними функціями та рівнем доступу для користувачів: модуль анкетування, модуль обробки результатів, модуль створення опитувань.

За принципом безсерверних обчислень, всі частини програми зв'язані в один веб-застосунок та звертаються до однієї бази даних без розробки власного серверу . Така архітектура дозволяє зосередитись на розробці корисної функціональності, а не на адмініструванні серверу [9]. Структурна схема інформаційної технології проведення онлайн-опитувань зображена на рис 2.1.

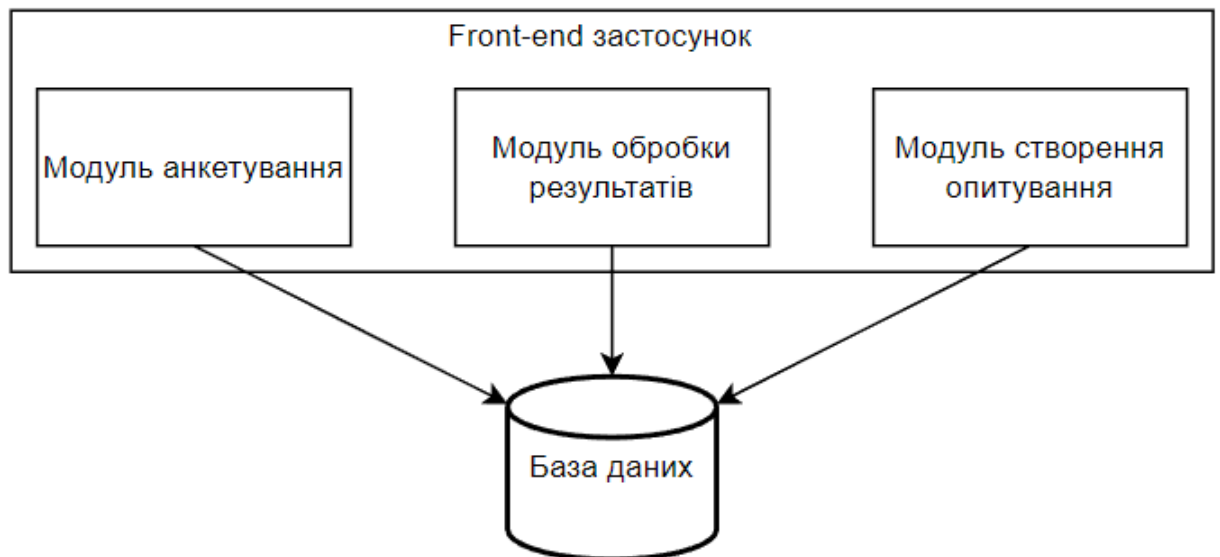


Рисунок 2.1 – Структурна схема інформаційної технології проведення онлайн-опитувань

Модуль анкетування містить такі функції:

- видача питань із опитувальника;
- автоматична перевірка даних на коректність;
- збереження даних в базу даних.

Модуль обробки результатів відповідає за отримання результатів опитування з бази даних та побудови графіків і діаграм на їх основі.

Модуль створення опитування містить такі функції:

- налаштування опитування;
- автоматична перевірка даних на коректність;
- збереження налаштувань опитування у базу даних.

Отже, у цьому підрозділі було розроблено структурну схему інформаційної технології проведення онлайн-опитувань, розглянуто функціонал основних модулів програми.

## 2.2 Використання мови моделювання UML в процесі проектування інформаційної технології проведення онлайн-опитувань

В процесі проектування інформаційної технології проведення онлайн-опитувань використано універсальну мову моделювання UML.

UML — уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування та є невід'ємною частиною уніфікованого процесу розробки інформаційної технології проведення онлайн-опитувань. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, яка називається UML-моделлю [10].

Під час проектування мовою UML створюють зазвичай 3 моделі:

- модель класів, що описує структуру об'єктів системи і їх індивідуальність по відношенню до інших класів. Ціль моделі класів полягає в тому, що охопити ті важливі концепції, які важливі для нашої системи;
- модель станів, що описує аспекти об'єктів, зв'язані часом та послідовними операціями, тобто події, що пов'язані зі зміною станів. Модель станів охоплює питання управління і посиляється на інші моделі;
- модель взаємодії описує взаємодію між об'єктами, тобто кооперацію об'єктів для забезпечення поведінки їх як єдиного цілого.

Крім основних трьох моделей, ще існує модель взаємодії між моделями, адже кожна модель посиляється одна на одну, наприклад, модель станів та взаємодії посиляються на модель класів [10].

Отже, у програмному модулі для проведення онлайн-опитувань буде використана мова UML, щоб промоделювати роботу системи та краще зрозуміти ті зв'язки, що виникнуть під час розробки різних модулів. Також це значно заощадить час та зусилля для проектування.

## 2.3 Моделювання інформаційної технології проведення онлайн-опитувань з використанням мови UML

Моделювання програми у даній роботі полягає у створенні та аналізі програми, структури модульних компонентів системи та програмній реалізації модулів програми.

При моделюванні була розроблена діаграма класів програми, що представлена на рисунку 2.2.

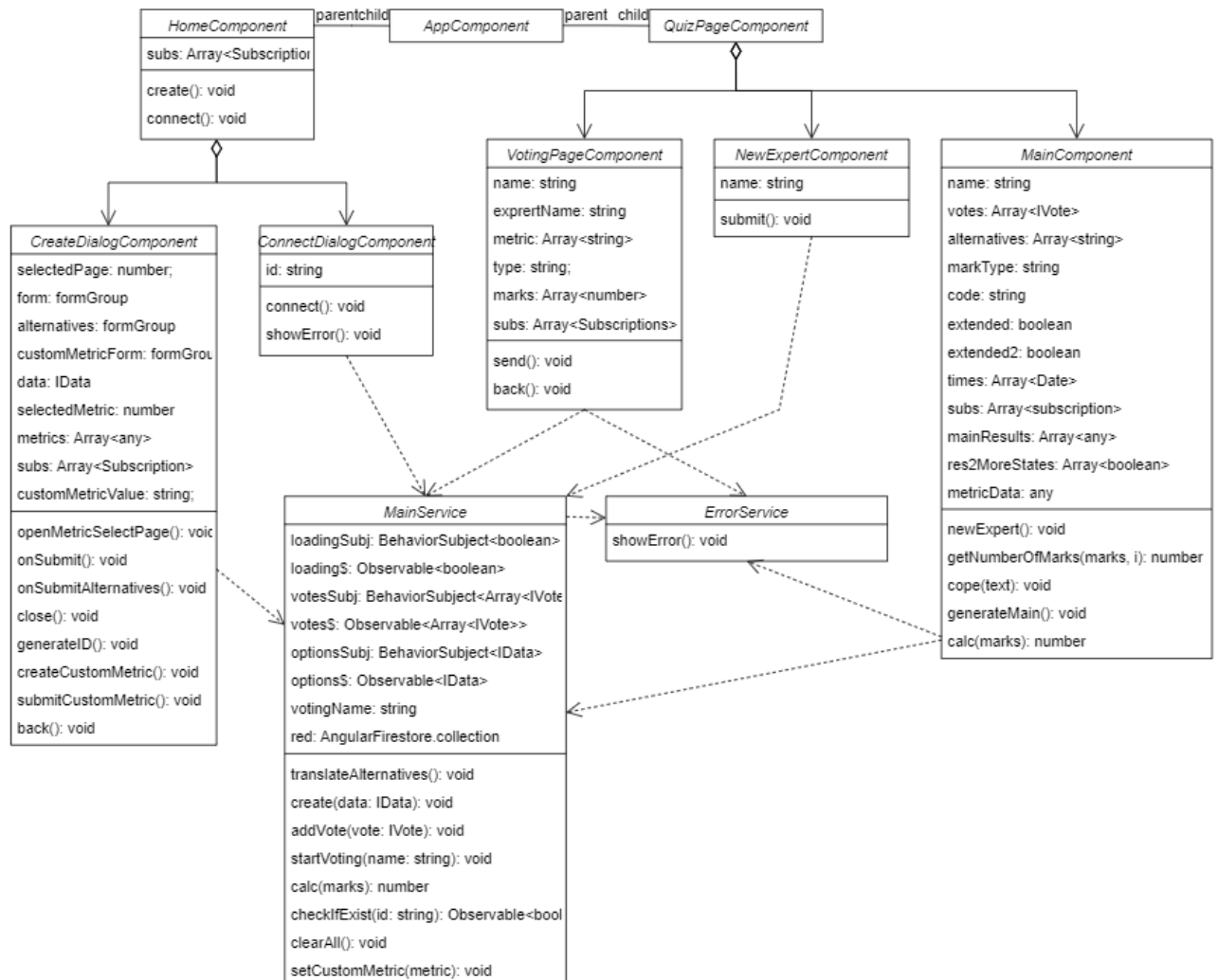


Рисунок 2.2 – Діаграма класів інформаційної технології для проведення онлайн-опитувань

На діаграмі можна побачити основні класи програми, такі як: AppComponent, QuizPageComponent, HomeComponent, CreateDialogComponent,

ConnectDialogComponent, VotingPageComponent, NewExpertComponent, MainComponent, MainService, ErrorService та їхні залежності та взаємодії.

Клас AppComponent – головний клас програми, він відповідає за ініціалізацію всіх інших класів.

Клас QuizPageComponent – клас, який відповідає за ініціалізацію сторінок результатів та проходження опитування.

Клас HomeComponent – відповідає за відображення головної сторінки.

Клас CreateDialogComponent – відповідає за відображення діалогового вікна створення нового опитування.

Клас ConnectDialogComponent – відповідає за відображення діалогового вікна під'єднання до існуючого опитування.

Клас VotingPageComponent – відповідає за відображення сторінки для проходження опитування.

Клас NewExpertComponent – відповідає за відображення діалогового вікна для створення нового експерта.

Клас MainComponent – відповідає за відображення результатів опитування.

Клас MainService – клас, який відповідає за всі основні обчислення та функції у додатку.

Клас ErrorService – відповідає за відображення повідомлень про помилки на сайті.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання [11]. Діаграму прецедентів інформаційної технології проведення онлайн-опитувань наведено на рисунку 2.3.



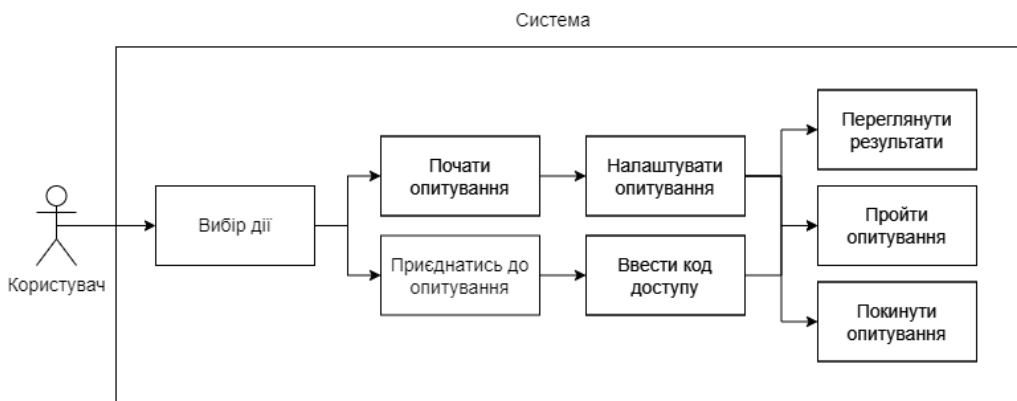


Рисунок 2.3 – Діаграма прецедентів інформаційної технології для проведення онлайн-опитувань

Діаграма діяльності – візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій [10]. Діаграму діяльності наведено на рисунку 2.4.

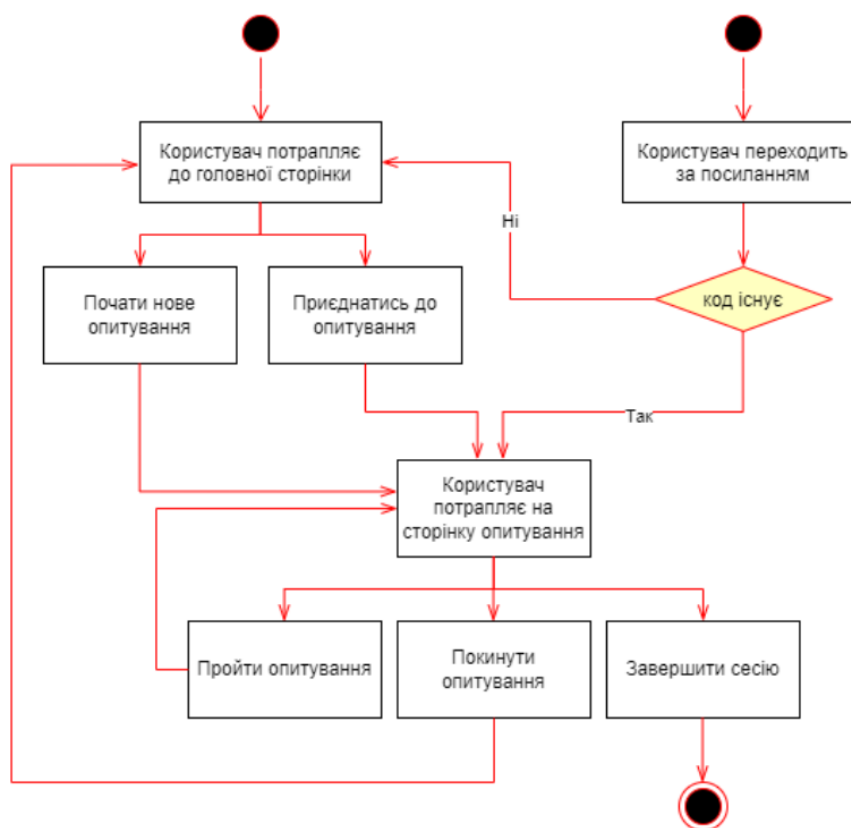


Рисунок 2.4 – Діаграма діяльності інформаційної технології для проведення онлайн-опитувань

Отже, у даній системі використано 10 основних класів, кожен з яких виконує певну роль для роботи користувача з сайтом, побудовано діаграму класів, прецедентів та діяльності.

## **2.4 Розробка алгоритмів функціонування інформаційної технології проведення онлайн-опитувань**

Алгоритм функціонування – це сукупність правил, що ведуть до правильного виконання технічного процесу в системі [12]. Алгоритм будь-якої складності можна реалізувати за допомогою трьох базових алгоритмічних структур: слідування, розгалуження та повторення.

Слідування – це форма організації команд в алгоритмі, під час якої дії виконуються послідовно, тобто одна за одною, без пропусків та повторень.

Розгалуження – це форма організації команд, коли залежно від виконання або невиконання певної умови виконується одна з двох послідовностей команд. Умову можна подати у вигляді запитання, сформульованого так, що допускає лише одну з двох відповідей: «так» або «ні». Якщо умова істинна, то виконуються команди з гілки «так», а якщо умова хибна, то команди з гілки «ні».

Повторення (цикл) – це форма організації команд, в якій одна й та сама послідовність дій виконується декілька разів, залежно від певної умови. Серія команд, що повторюється під час кожної ітерації циклу – це тіло циклу. Команди тіла циклу виконуються до тих пір, поки виконується задана умова. Під час виконання алгоритму спочатку перевіряється умова і якщо вона істинна, то тіло циклу виконується і відбувається повернення на перевірку умови [13].

Розглянемо схеми функціонування програмних модулів створення опитування та анкетування. Модуль створення опитування матиме форму для введення даних для налаштування опитування та можливість виконання HTTPS запиту з подальшим збереження у базу даних. Алгоритм функціонування модулю створення опитування зображено на рисунку 2.5.



Рисунок 2.5 – Алгоритм функціонування модулю створення опитування

Модуль анкетування буде мати простіший алгоритм. Так як модуль відповідатиме лише за введення даних – алгоритм не матиме жодних умов або

циклів. Алгоритм функціонування модулю анкетування зображено на рисунку 2.6.



Рисунок 2.6 – Алгоритм функціонування модулю анкетування

Отже, у цьому підрозділі було розглянуто особливості побудови алгоритмів функціонування, їх основні елементи. Було побудовано алгоритми функціонування модулів анкетування та створення опитування.

## 2.5 Розробка шкал для інформаційної технології проведення онлайн-опитувань

Для успішної реалізації інформаційної технології проведення онлайн-опитувань, було вирішено розробити та впровадити дві різновиди шкал оцінювання, що відповідатимуть задачам дослідження та задовільнять більшість потреб користувача. Кожна з шкал використовується для отримання відповідей респондентів та враховує різні аспекти їхнього ставлення до поставлених питань.

Шкала 1 – Комбінована шкала типу Лікерта з числовими значеннями. Ця шкала включає у себе елементи числової послідовної шкали та Лікерт-шкали. Респондентам пропонується вибрати один із п'яти варіантів відповіді, які охоплюють весь спектр можливих позицій щодо поставленого питання. Категорії "Категорично проти" та "Категорично за" дозволяють виражати екстремальні погляди, тоді як "Байдуже" покриває середину спектра, а "Проти" та "За" представляють менш екстремальні думки. Для можливості обчислення та детального аналізу результатів – відповідно до кожного варіанту відповіді – встановлено числові значення – від -2 до 2. Також, за для візуального відображення результатів, було обрано класичну кольорову схему – від темно-червоного до темно-зеленого. Шкала зображена на рисунку 2.7.

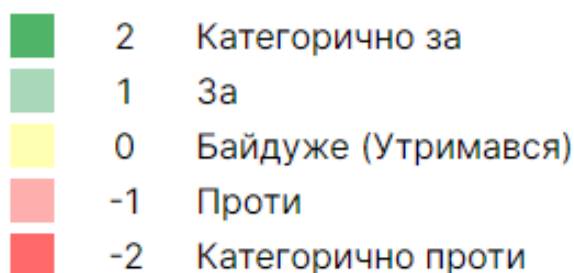


Рисунок 2.7 – Комбінована шкала типу Лікерта з числовими значеннями

Шкала 2 – Числова послідовна шкала – базується на простому числовому відображенні відповідей. Респондентам пропонується обрати число від 1 до 5, що відповідає їхньому ставленню до поставленого питання. Для візуального

відображення обрано відтінки нейтрального синього кольору, де темніший колір позначає вищий бал, а світліший – нижчий. Шкала зображена на рисунку 2.8.

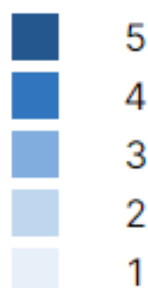


Рисунок 2.8 – Числова послідовна шкала

Ця шкала є простою та зрозумілою, що полегшує відповідь респондентів. Крім того, вона дозволяє отримати кількісні дані для більш точного аналізу.

## 2.6 Математична модель інформаційної технології проведення онлайн-опитувань

Для більше детального аналізу результатів опитування, необхідно змодельовати формули статистичних обчислень результатів. Для обчислення було обрано наступні значення: загальна згода, загальна незгода, індиферентність, полярність думок, конфронтаційність, лобізм, антилобізм, поміркованість, одностайність. Розглянемо кожне значення окремо на прикладі розробленої комбінованої шкали типу Лікерта з числовими значеннями:

- Загальна згода відображає відсоток позитивних відповідей, тобто «категорично за» та «за». Обчислюється за формулою:

$$GA = \frac{Nm_4 + Nm_5}{Nm},$$

де GA – загальна згода;

$Nm_4$ ,  $Nm_5$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

- Загальна незгода відображає відсоток негативних відповідей, тобто «категорично проти» та «проти». Обчислюється за формулою:

$$GDA = \frac{Nm_1 + Nm_2}{Nm},$$

де  $GDA$  – загальна незгода;

$Nm_1, Nm_2$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

- Індиферентність відображає відсоток нейтральних відповідей, в нашому випадку це «байдуже(утримався)». Обчислюється за формулою:

$$IN = \frac{Nm_3}{Nm},$$

де  $IN$  – індиферентність;

$Nm_3$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

- Полярність думок відображає відсоток протилежних відповідей. Обчислюється за формулою:

$$POT = \frac{\min((Nm_1 + Nm_2), (Nm_4 + Nm_5))}{Nm},$$

де  $POT$  – полярність думок;

$Nm_1, Nm_2, Nm_4, Nm_5$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

- Конфронтаційність, як і полярність думок, відображає відсоток протилежних відповідей, але тільки відносно категоричних відповідей, тобто «категорично за» та «категорично проти». Обчислюється за формулою:

$$C = \frac{\min(Nm_1, Nm_5)}{Nm},$$

де  $C$  – конфронтаційність;

$Nm_1, Nm_5$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

- Лобізм відображає відсоток категорично позитивних відповідей. Обчислюється за формулою:

$$L = \frac{Nm_5}{Nm},$$

де  $L$  – лобізм;

$Nm_5$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

- Антилобізм відображає відсоток категорично негативних відповідей. Обчислюється за формулою:

$$AL = \frac{Nm_1}{Nm},$$

де  $AL$  – антилобізм;

$Nm_1$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

- Поміркваність відображає відсоток некатегоричних відповідей, таких як «проти», «байдуже» та «за». Обчислюється за формулою:



$$M = \frac{Nm_2 + Nm_3 + Nm_4}{Nm},$$

де  $M$  – поміркованість;

$Nm_2, Nm_3, Nm_4$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

- Одностайність відображає на скільки одностайно респонденти надали позитивні або негативні варіанти відповідей. Обчислюється за формулою:

$$M = \frac{|(Nm_1 + Nm_2) - (Nm_4 + Nm_5)|}{Nm},$$

де  $M$  – Одностайність;

$Nm_1, Nm_2, Nm_4, Nm_5$  – кількість відповідей по шкалі;

$Nm$  – загальна кількість відповідей.

## 2.7 Висновок до розділу 2

Отже, в даному розділі було спроектовано інформаційну технологію проведення онлайн-опитувань. Було розроблено структурну схему інформаційної технології, за допомогою мови моделювання UML було змодельовано діаграму класів інформаційної системи, діаграму прецедентів та діаграму діяльності. Було розроблено алгоритми функціонування інформаційної технології проведення онлайн-опитувань. Також було розроблено шкали для використання в інформаційній технології проведення онлайн-опитувань та було представлено математичну модель.

### 3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ ОНЛАЙН-ОПИТУВАНЬ

#### 3.1 Обґрунтування вибору інструментів розробки

Для розробки програмного забезпечення, потрібно визначитись з вибором таких інструментів, як: мова програмування, фреймворк, база даних, додаткові сервіси. Так як, метою роботи є розробка програмного забезпечення, яке повинно виконуватись і браузері, для розробки актуального веб-застосунку потрібно обрати серед двох мов програмування, а саме JavaScript та TypeScript.

JavaScript – це легковажна, інтерпретована або JIT-компілювана, об'єктно-орієнтована мова з функціями першого класу. Найширше застосування знаходить як мова сценаріїв веб-сторінок, але також використовується і в інших програмних продуктах, наприклад, node.js або Apache CouchDB. JavaScript це прототипно-орієнтована, мультипарадигменна мова з динамічною типізацією, яка підтримує об'єктно-орієнтований, імперативний та декларативний стилі програмування [14].

JavaScript запускається на стороні клієнта Інтернету, який може використовуватися для створення сценаріїв поведінки веб-сторінки при тих чи інших обставинах. JavaScript легко вивчити, а також це потужна скриптова мова, що широко використовується для розробки застосунків будь-якої складності.

В свою чергу, TypeScript - це мова програмування з відкритим вихідним кодом для розробки програм великого рівня. TypeScript був розроблений одним із технологічних гігантів - Microsoft у 2012 році. Фактичною причиною цієї розробки TypeScript була робота з великомасштабними програмами [15].

TypeScript — це статично скомпільована мова програмування для написання чіткого та короткого коду JavaScript. Він виконує ті ж цілі, що і JavaScript, і може використовуватися як для клієнтських, так і для серверних програм. Крім того, бібліотеки JavaScript також сумісні з TypeScript. Оскільки

TypeScript є розширеною версією JavaScript, весь код JavaScript сумісним з TypeScript.

Основними перевагами на користь TypeScript є:

- Типізація даних. TypeScript дозволяє задавати типи даних, що значно полегшує розробку та підтримку програмного забезпечення. Типізація даних робить код більш читабельним та передбачуваним, допомагає швидше знаходити помилки.

- ООП. TypeScript реалізує інструменти для роботи з ООП, а саме модифікатори доступу, властивості-аксесори (get, set), інтерфейси, тощо [15].

Зважаючи на перераховані особливості та переваги, TypeScript обрано для розробки програмного забезпечення.

Далі потрібно обрати фреймворк, за допомогою якого можна буде створити ефективний застосунок.

Фреймворк - це програмна платформа, що визначає структуру програмної системи. Програмне забезпечення, що полегшує розробку та об'єднання різних компонентів великого програмного проекту. Фреймворк відрізняється від поняття бібліотеки тим, що бібліотека може бути використана в програмному продукті як набір підпрограм близької функціональності, не впливаючи на архітектуру програмного продукту і не накладаючи на неї жодних обмежень. У той час як фреймворк диктує правила побудови архітектури додатка, задаючи на початковому етапі розробки поведінку за умовчанням - "каркас", який потрібно буде розширювати та змінювати відповідно до зазначених вимог [16].

На сьогоднішній день, існує два найпопулярніші фреймворки для розробки на TypeScript – React та Angular.

React — відкритий JavaScript фреймворк для створення інтерфейсів користувача, який покликаний вирішувати проблеми часткового оновлення вмісту веб-сторінки, з якими стикаються в розробці односторінкових застосунків. Розробляється компанією Meta.

Angular – це фреймворк від компанії Google для створення клієнтських програм. Насамперед він націлений на розробку SPA-застосунків (Single Page

Application), тобто односторінкових додатків. У цьому плані Angular є спадкоємцем іншого фреймворку AngularJS. У той же час Angular це не нова версія AngularJS, а принципіально новий фреймворк [17].

Angular надає таку функціональність як двостороннє зв'язування, що дозволяє динамічно змінювати дані в одному місці інтерфейсу при зміні даних моделі в іншому, шаблони, маршрутизація і так далі.

Основною перевагою Angular є слідування принципам ООП, у той час як розробники React поділились на два табори – функціонального та об'єктно орієнтованого програмування. Крім того, React, який є фактично бібліотекою, не накладає необхідних обмежень на архітектуру застосунку, що створює ситуацію в якій різні групи розробників можуть писати застосунки за зовсім різними принципами, що ускладнює підтримку цих проектів.

Angular реалізує багато корисного функціоналу, який у інших фреймворках, зазвичай, приходиться реалізовувати власноруч, або використовувати зовнішні бібліотеки. Також, потужним інструментом є наявність механізму ін'єкції залежностей (Dependency Injection). Він дозволяє зручно передавати у застосунку будь-які дані, перетворювати та перевизначати їх у потрібних частинах. Тому ми можемо робити архітектуру додатків більш простою та гнучкою: зрозумілий потік даних, мінімальна пов'язаність коду, легкість при тестуванні або заміні залежностей [17].

Також необхідно обрати інструмент, за допомогою якого можна уникнути реалізації серверного додатку та легко зв'язатися з базою даних. Найпопулярнішим сервісом, що реалізує такий функціонал є Firebase.

Firebase – продукт компанії Google, призначений для підтримки та розвитку додатків будь-якого типу. За допомогою Firebase можна отримати доступ до безлічі корисних функцій, а саме: сервіс автентикації, база даних, сховище даних, хостинг, хмарні функції, тощо. Для реалізації інформаційної технології проведення онлайн-опитувань нам знадобляться хостинг та база даних Firestore. Cloud Firestore – це гнучка, масштабована база даних для розробки мобільних, веб-додатків та серверів від Firebase та Google Cloud. Як і

база даних Firebase Realtime, вона синхронізує дані між клієнтськими програмами за допомогою прослуховувачів у реальному часі та пропонує автономну підтримку для мобільних пристроїв та Інтернету, щоб ви могли створювати програми, які працюють незалежно від затримки в мережі або підключення до Інтернету. Cloud Firestore також пропонує безшовну інтеграцію з іншими продуктами Firebase та Google Cloud, включаючи хмарні функції [18].

Ключові переваги Cloud Firestore:

- Гнучкість. Модель даних Cloud Firestore підтримує гнучкі ієрархічні структури даних що дозволяє зберігати дані в документах, організованих у колекції. Документи можуть містити складні вкладені об'єкти на додаток до вкладених колекцій.
- Оновлення у реальному часі. Як і база даних реального часу, Cloud Firestore використовує синхронізацію даних для оновлення даних на будь-якому підключеному пристрої. Однак він також призначений для ефективного виконання найпростіших одноразових запитів на вибірку.
- Виразні запити. У Cloud Firestore можна використовувати запити для отримання окремих, певних документів або для отримання всіх документів у колекції, які відповідають параметрам вашого запиту. Ваші запити можуть включати кілька пов'язаних фільтрів та поєднувати фільтрацію та сортування. Вони також індексуються за замовчуванням, тому продуктивність запитів є пропорційною розміру набору результатів, а не набору даних [18].

Отже, на основі проведеного дослідження, для реалізації інформаційної технології проведення онлайн-опитувань було обрано мову програмування TypeScript, фреймворк Angular, та сервіс безсерверної розробки Firebase. Дані інструменти дозволять розробити гнучкий додаток, легкий у використанні та підтримці. Firebase надасть можливість використання NoSQL бази даних без розробки серверного застосунку.

## 3.2 Основні оператори мови програмування TypeScript

Дії програми виражаються в операторах. До них відносяться оголошення змінних, присвоювання значень, виклик методів, прохід по колекціях, умовні переходи, тощо. Порядок виконання операторів в програмі називається потоком керування або потоком виконання. Потік керування може відрізнятись при кожному старті програми, в залежності від реакції програми на вхідні дані, які вона отримує під час виконання. Оператор може складатися з одного рядку коду, що закінчується крапкою з комою або з набору однорядкових операторів в блоці. Блок операторів обмежується фігурними дужками ({}), і може містити вкладені блоки [19]. Типи операторів мови програмування TypeScript наведені в таблиці 3.1.

Таблиця 3.1 – Типи операторів TypeScript

Категорія	Опис
Оператори оголошення	Оператори оголошення представляють собою нову змінну або константу. <code>let y = 21; const str = "Text"; var d = 2.33;</code>
Оператори вибору	Оператори вибору дозволяють розгалуження коду, в залежності від однієї або кількох умов. <code>if</code> , тернарний оператор;
Оператори ітерацій	Оператори ітерацій дозволяють перегляд колекцій або повторення одного і того ж набору операторів до виконання заданої умови. <code>do</code> , <code>for</code> , <code>foreach</code> , <code>while</code> , <code>map</code> ;
Оператори переходу	Оператори переходу виконують передачу керування іншому розділу коду <code>break</code> , <code>continue</code> , <code>default</code> , <code>return</code> , <code>yield</code> ;
Оператори обробки виключень	Оператори обробки виключень дозволяють акуратно відновитися після виключних ситуацій, що виникають пі час виконання. <code>throw</code> , <code>try</code> , <code>catch</code> , <code>then</code> ;
Оператор <code>await</code>	Оператор для керування в асинхронних методах;

### **3.3 Особливості середовища Visual Studio Code**

Visual Studio Code – середа розробки, розповсюджується компанією Microsoft. Середа розробки призначена для створення додатків з використанням різних мов програмування, серед яких є TypeScript.

Visual Studio Code забезпечує авто доповнення аналізу коду, навігацію по коду, відлагодження, інтеграцію систем управління версіями (зокрема, Git). Перевагою інтегрованою середі розробки Visual Studio Code є робота з проектами (у тому числі повний рефакторинг коду TypeScript). Система має гнучке налаштування від кольорової схеми і шрифту до розміщення панелей.

Visual Studio Code підтримує усі існуючі стандарти TypeScript в залежності від встановленої версії, а також такі популярні фреймворки, як Angular та React. Має вбудований термінал, який можна налаштувати під будь-яку оболонку [20].

Ще однією великою перевагою Visual Studio Code є те, що він розповсюджується безплатно.

Таким чином, Visual Studio Code є однією з найкращих середовищ розробки для програмування на мові TypeScript.

### **3.4 Програмна реалізація інформаційної технології проведення онлайн-опитувань**

Першим етапом розробки є створення Angular проекту за допомогою команди Angular CLI – “ng new” та створення файлової структури модулів та компонентів. Файлова структура Angular-застосунку зображена на рисунку 3.1.

Далі було розроблено зовнішній вигляд усіх компонентів за допомогою HTML та CSS. HTML та CSS файли з’єднуються з файлом компоненту за допомогою параметрів декоратору Component. Приклад задання шаблону та стилів компоненту зображений на рисунку 3.2.

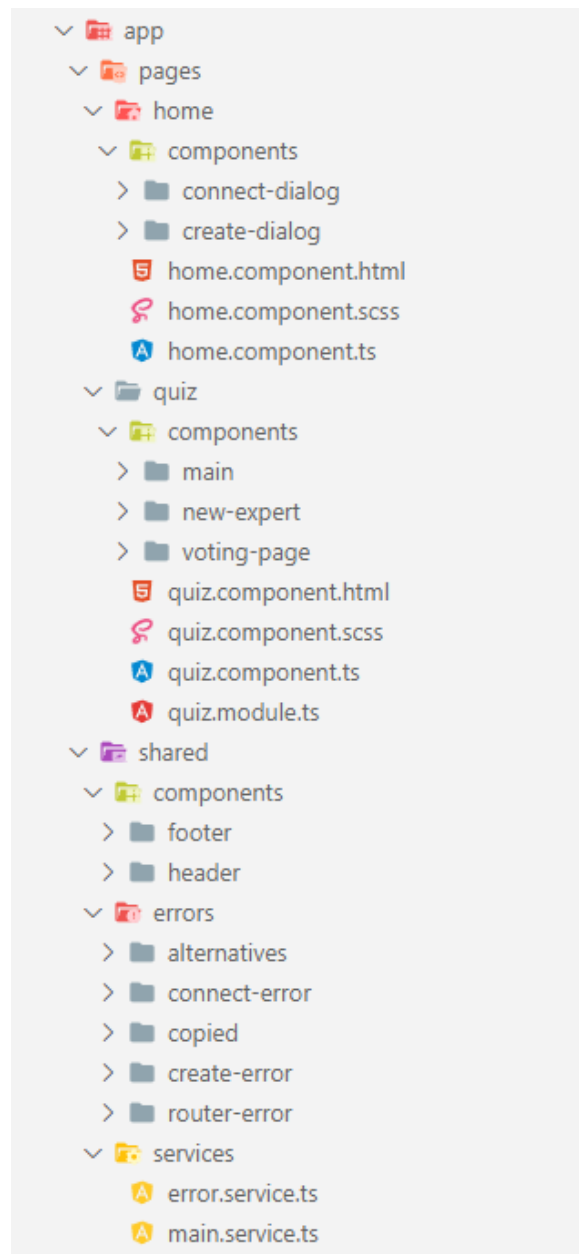


Рисунок 3.1 – Файлова структура Angular-застосунку

```

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.scss']
})

```

Рисунок 3.2 – Приклад задання шаблону та стилів компоненту

Наступним кроком необхідно налаштувати маршрутизацію застосунку. Цей крок допоможе перетворити наш веб-сайт на справжній односторінковий додаток. З використанням такого підходу, застосунок працює набагато



ефективніше, також користувач не буде бачити перезавантаження сторінки при зміні сторінки, що покращує користувацький досвід.

Для налаштування маршрутизації Angular має RouterModule, який потрібно імпортувати в модулі найвищого рівня, передаючи структуру маршрутизації застосунку. Результат налаштування маршрутизації зображено на рисунку 3.3.

```
const routes: Routes = [
  { path: '', redirectTo: 'home', pathMatch: 'full' },
  { path: 'home', component: HomeComponent },
  { path: 'quiz', loadChildren: () => import('./pages/quiz/quiz.module').then(m => m.QuizModule) },
];

unknown, 14 months ago | 1 author (unknown)
@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
```

Рисунок 3.3 – Результат налаштування маршрутизації застосунку

Після реалізації класів та алгоритмів функціонування інформаційної технології проведення онлайн опитування, розроблених у другому розділі, необхідно налаштувати з'єднання з сервісом Firebase. Для цього, на офіційному сайті Firebase потрібно зареєструвати новий додаток, та виконати кроки з інструкції.

Налаштувавши Cloud Firestore, напишемо запити до бази даних. Для виконання запитів, за допомогою сервісу AngularFirestore потрібно визначити посилання на потрібну колекцію, після чого можна виконати на зміну цієї колекції. Приклад запиту на оновлення поля “votes” зображений на рисунку 3.4.

```
this.ref.doc(id).update({ votes });
```

Рисунок 3.4 – Приклад запиту на оновлення поля “votes”

Отже, у цьому підрозділі було реалізовано програмне забезпечення, показано основні моменти налаштування додатку. Показано приклад виконання запиту до бази даних Firestore.

### 3.5 Тестування розробленої інформаційної технології проведення онлайн-опитувань

Протестуємо функціонал розробленого програмного забезпечення з метою перевірки працездатності його роботи, коректності отриманих результатів. Доступ до сайту можна отримати за посиланням <https://verbvispoll.web.app>. Перейшовши за посиланням, потрапляємо на головну сторінку. Головна сторінка додатку зображена на рисунку 3.5.

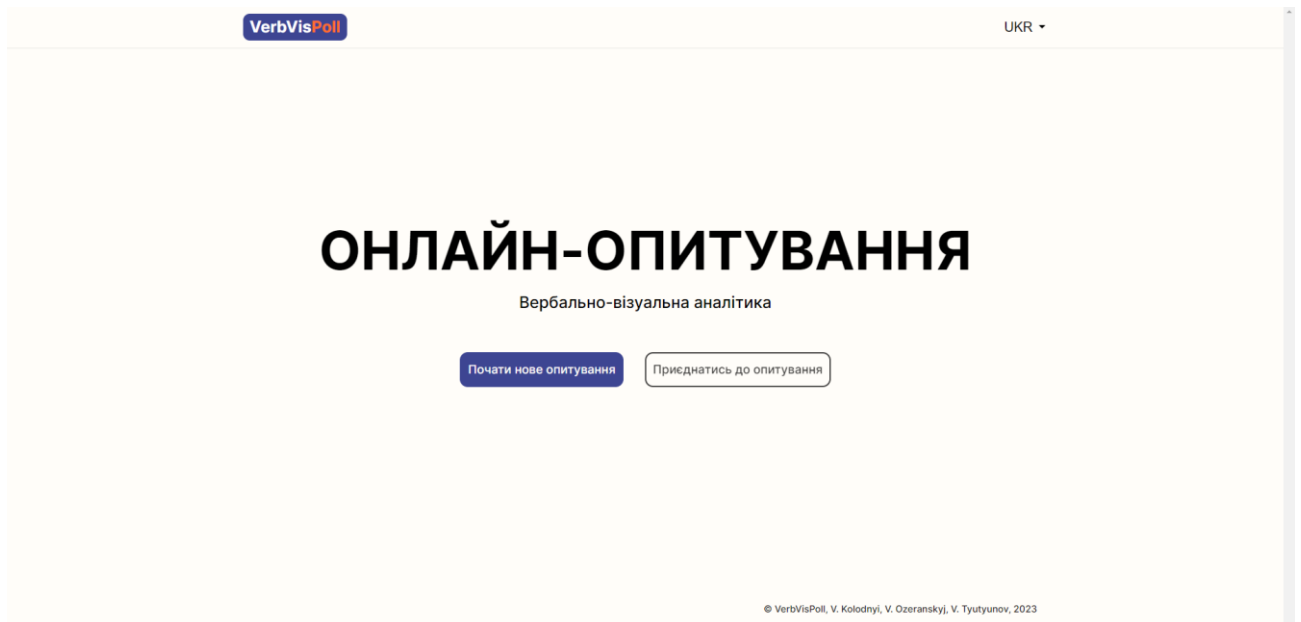


Рисунок 3.5 – Головна сторінка сайту

Перевіримо функціонування створення нового опитування. Для цього натиснемо на кнопку «Почати нове опитування», після чого відкриється модальне вікно з формою налаштування майбутнього опитування. Модальне вікно зображене на рисунку 3.6.

**Нове опитування**

Назва \*

Найкращий фільм Крістофера Нолана

Опис

Код доступу (5-20) \*

best\_nolan\_film

Далі

Рисунок 3.6 – Модальне вікно з формою базового налаштування онлайн-опитування

Ввівши дані, необхідні дані, натиснемо на кнопку «Далі». Після цього ми попадаємо на меню вибору типу шкали оцінювання. Меню вибору типу шкали оцінювання зображено на рисунку 3.7.

←

**Тип шкали оцінювання:**

Категорично проти	Проти	Байдуже (Утримався)	За	Категорично за
1	2	3	4	5

Конструктор шкал

Далі

Рисунок 3.7 – Меню вибору типу шкали онлайн-опитування

Обравши «Конструктор шкал» попадемо на наступний екран де можна створити шкалу яка налічує до 10 градацій, які складаються з назви, балу та кольору. Конструктор шкал зображений на рисунок 3.8.

←

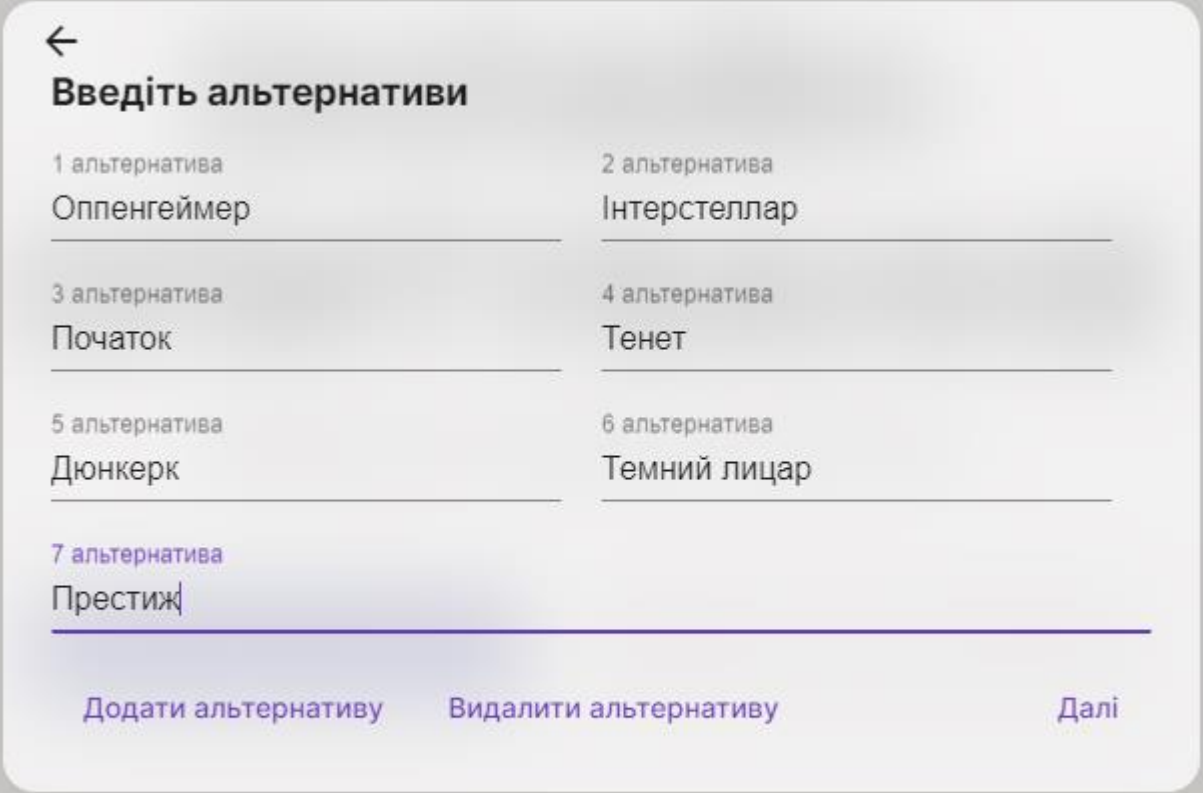
**Конструктор шкал** Додати градацію +

Назва градації	Бал	Колір	×	Назва градації	Бал	Колір	×
1 градація	0	■	×	2 градація	0	■	×
3 градація	0	■	×	4 градація	0	■	×
5 градація	0	■	×	6 градація	0	■	×
7 градація	0	■	×	8 градація	0	■	×
9 градація	0	■	×	10 градація	0	■	×

Далі

Рисунок 3.8 – Конструктор шкал

Повернемось назад та оберемо вже існуючу шкалу з 5 градацій. Після вибору шкали потрапляємо на форму введення альтернатив. Форма введення альтернатив зображена на рисунку 3.9.



←  
**Введіть альтернативи**

1 альтернатива  
Оппенгеймер

2 альтернатива  
Інтерстеллар

3 альтернатива  
Початок

4 альтернатива  
Тенет

5 альтернатива  
Дюнкерк

6 альтернатива  
Темний лицар

7 альтернатива  
Престиж

Додати альтернативу    Видалити альтернативу    Далі

Рисунок 3.9 – Форма введення альтернатив онлайн-опитування

Ввівши необхідні дані та натиснувши «Далі» потрапляємо на сторінку опитування. Поки що вона не містить результатів, так як опитування не було пройдено жодного разу. Сторінка результатів онлайн-опитування зображена на рисунку 3.10.

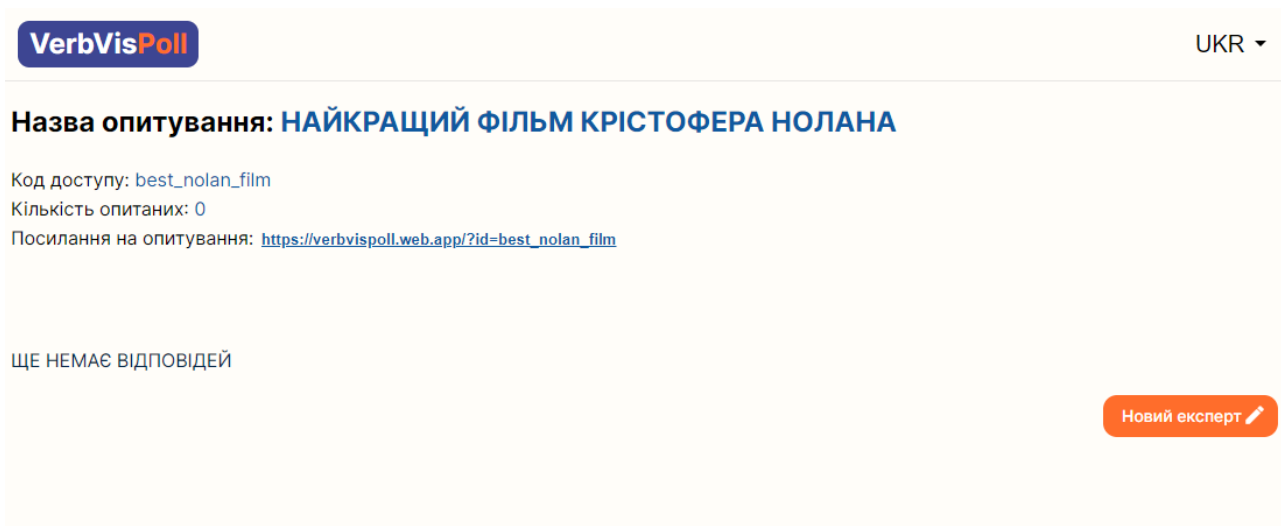


Рисунок 3.10 – Сторінка результатів онлайн-опитування

Для проходження опитування натиснемо кнопку «Новий експерт», після чого відкриється модальне вікно для введення даних експерта. Модальне вікно для введення даних експерта зображене на рисунку 3.11.

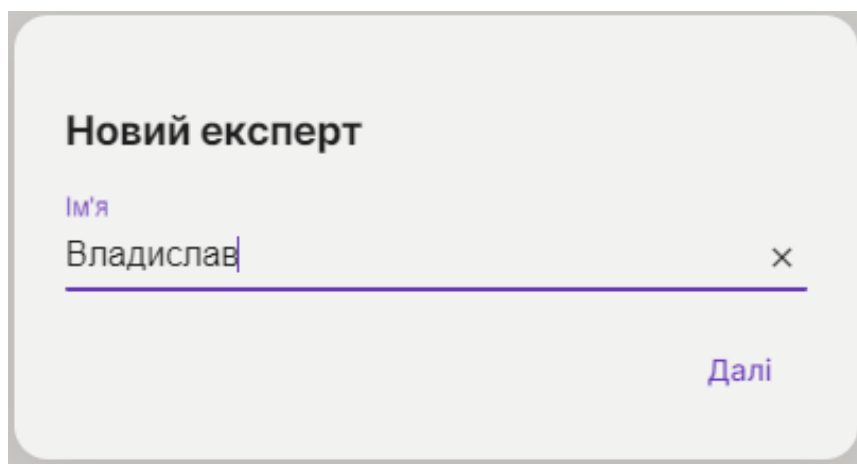


Рисунок 3.11 – Модальне вікно для введення даних експерта

Ввівши необхідні дані та натиснувши на кнопку «Далі» потрапимо на сторінку проходження опитування. Сторінка проходження опитування зображена на рисунку 3.12.

**VerbVisPoll** UKR ▾

**Експерт Владислав**

Вкажіть, чи подобаються вам представлені фільми

<b>Оппенгеймер</b>				
Категорично проти	Проти	<b>Байдуже (Утримався)</b>	За	Категорично за
<b>Інтерстеллар</b>				
Категорично проти	<b>Проти</b>	Байдуже (Утримався)	За	Категорично за
<b>Початок</b>				
Категорично проти	Проти	Байдуже (Утримався)	<b>За</b>	Категорично за
<b>Тенет</b>				
Категорично проти	<b>Проти</b>	Байдуже (Утримався)	За	Категорично за
<b>Дюнкерк</b>				
<b>Категорично проти</b>	Проти	Байдуже (Утримався)	За	Категорично за
<b>Темний лицар</b>				
Категорично проти	Проти	Байдуже (Утримався)	<b>За</b>	Категорично за
<b>Престиж</b>				
Категорично проти	Проти	<b>Байдуже (Утримався)</b>	За	Категорично за

**Підтвердити**

Рисунок 3.12 – Сторінка проходження онлайн-опитування

Обравши варіанти відповідей та натиснувши кнопку «підтвердити», потрапляємо назад на сторінку результатів. Для більшої наочності, пройдем опитування ще декілька разів. Сторінка опитування після проходження опитування зображена на рисунку 3.13.

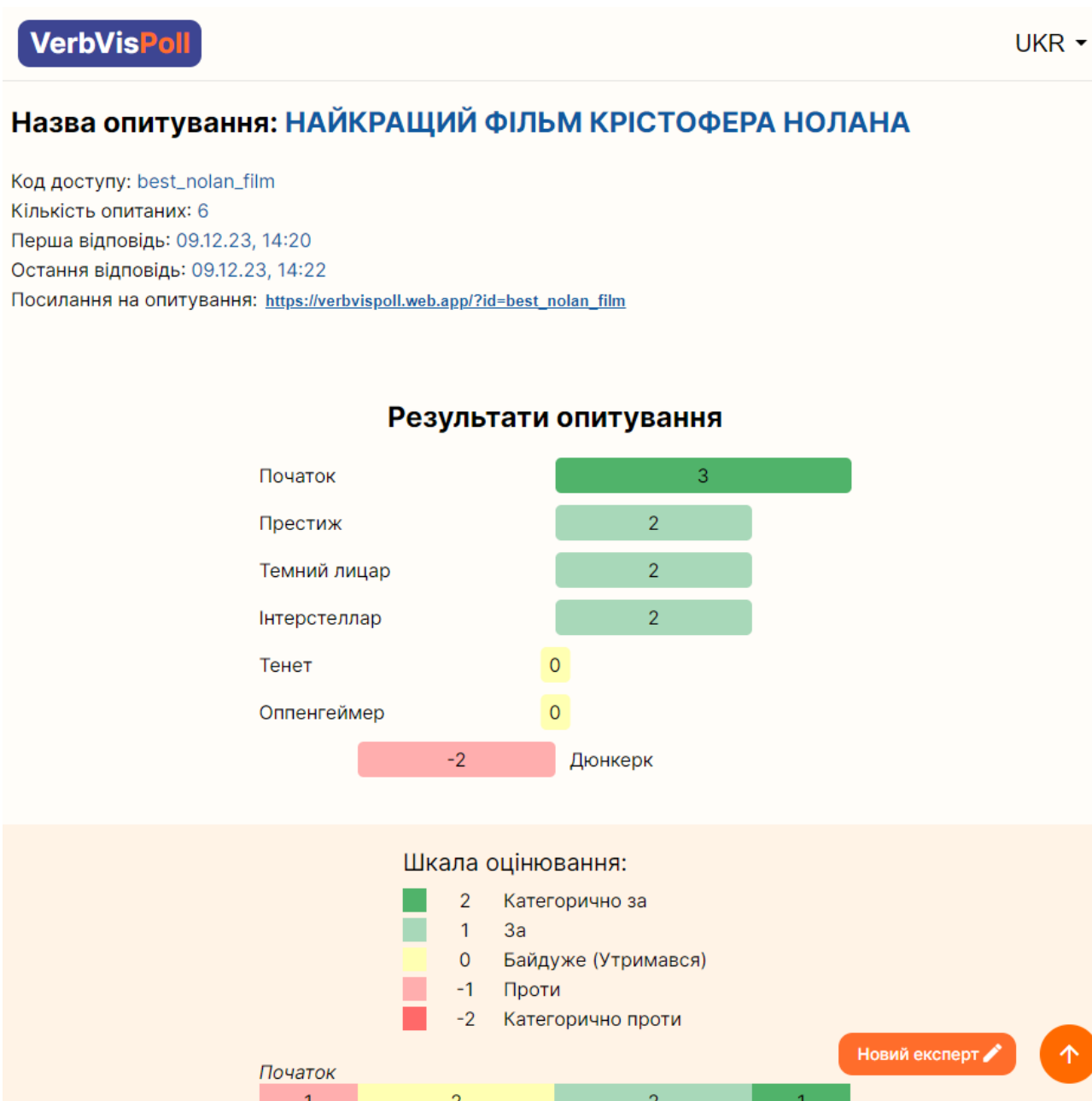


Рисунок 3.13 – Сторінка результатів онлайн-опитування з наявними результатами

Щоб побачити більш детальні результати, необхідно натиснути кнопку «Детальніше». Компонент з детальними результатами зображений на рисунку 3.14.



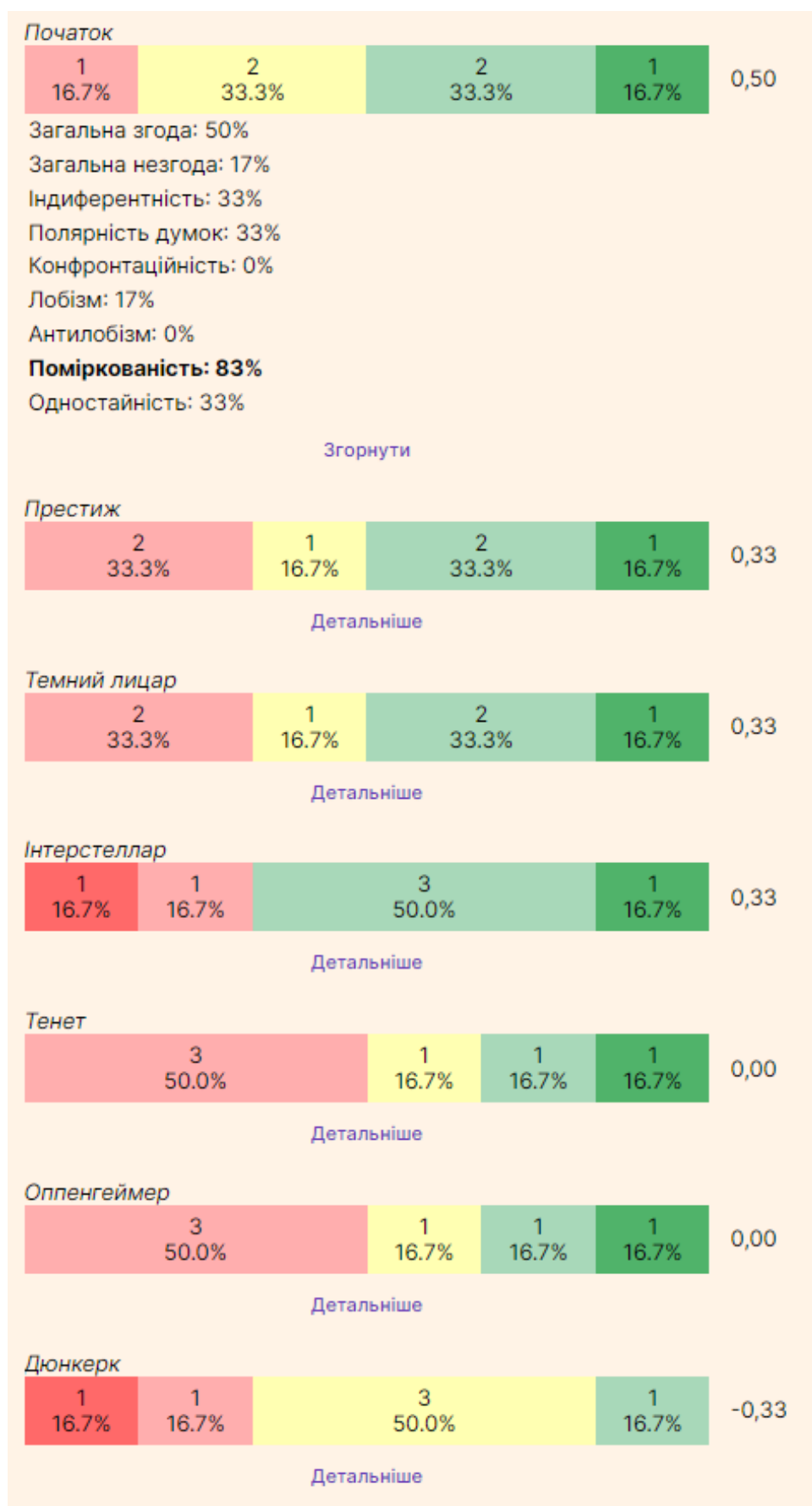


Рисунок 3.14 – Компонент з детальними результатами онлайн-опитування

Натиснувши на кнопку «Мапа переважань» можна побачити таблицю результатів з відповідність відповідей кожного експерта до кожного питання. Мапа переважань зображена на рисунку 3.15.

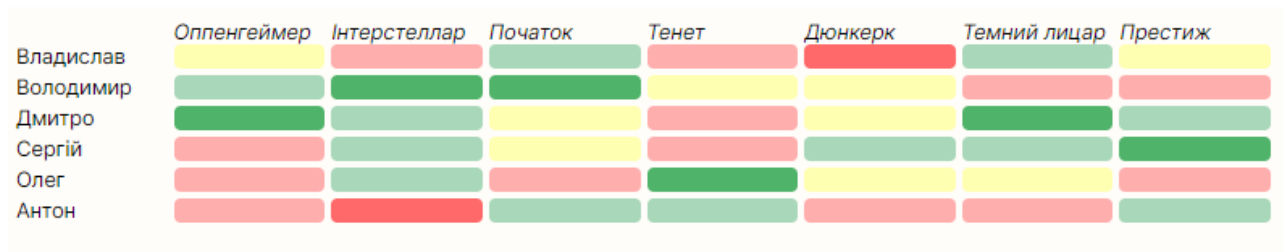


Рисунок 3.15 – Мапа переважань результатів онлайн-опитування

Також важливо перевірити наявність та правильність наших записів у базі даних. Колекція даних онлайн-опитування зображена на рисунку 3.16.

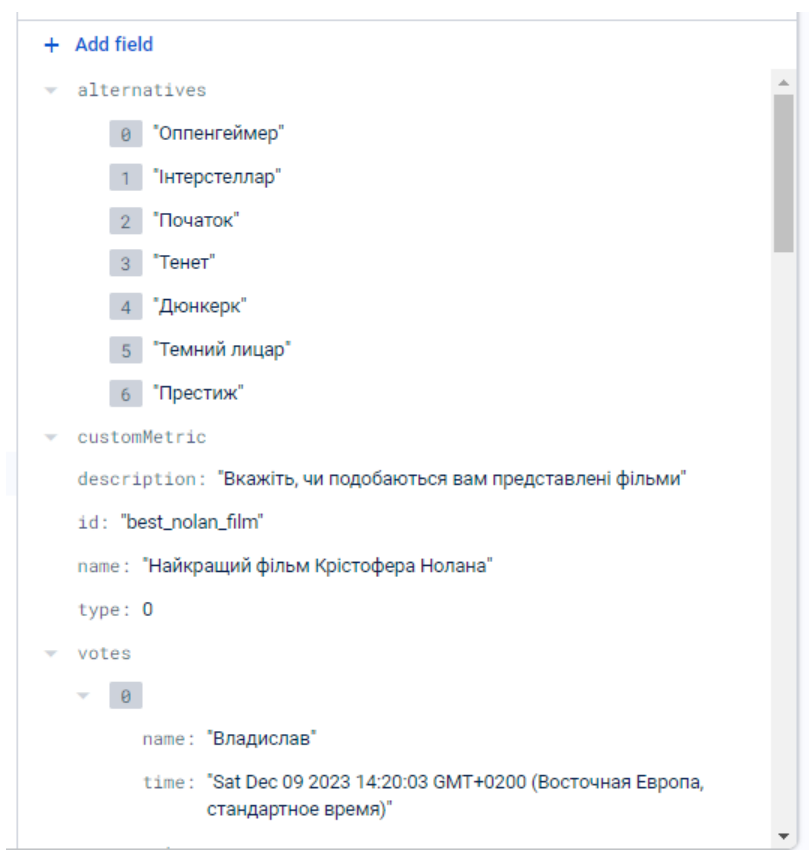


Рисунок 3.16 – Колекція даних проведеного онлайн-опитування

Як бачимо, усі введені дані збережено.

Залишилось протестувати можливість приєднання до існуючого онлайн опитування. Покинемо наше опитування та натиснемо кнопку «Приєднатись до опитування» на головній сторінці. Модальне вікно приєднання до опитування зображено на рисунку 3.17.

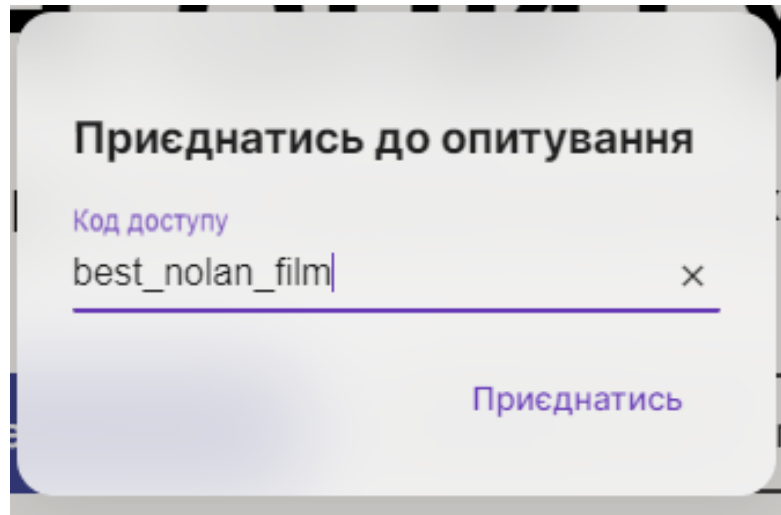


Рисунок 3.17 – Модальне вікно приєднання до онлайн-опитування

Ввівши код доступу, та натиснувши кнопку «Приєднатись», потрапляємо на існуюче тестове опитування. Результат приєднання до існуючого онлайн-опитування зображено на рисунку 3.18.

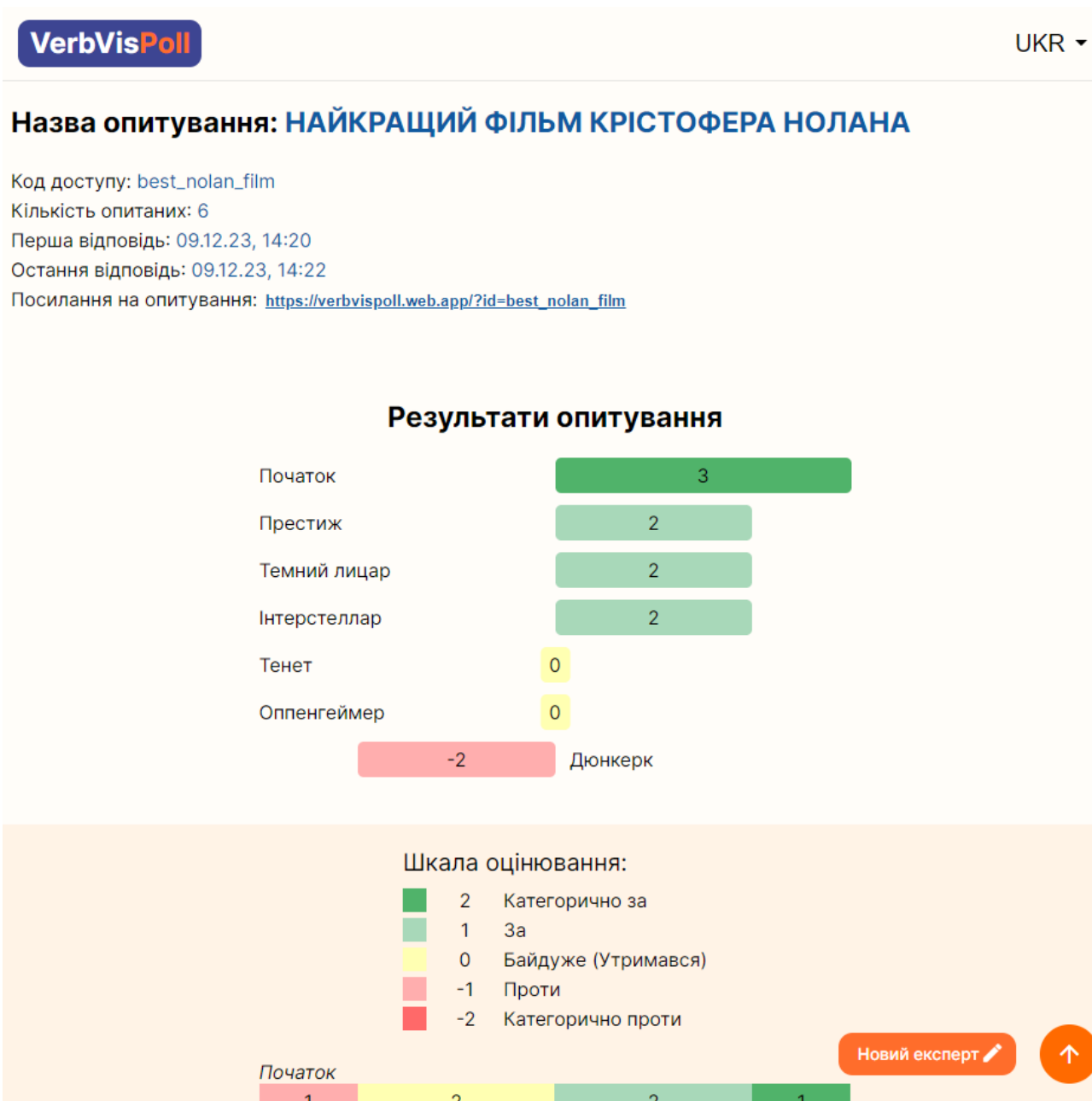


Рисунок 3.18 – Результат приєднання до існуючого тестового онлайн-опитування

Отже, у цьому підрозділі було протестовано увесь функціонал розробленого програмного забезпечення та доведено його готовність для використання рядовим користувачем.

### 3.6 Приклад практичного використання інформаційної технології проведення онлайн-опитувань

Результати магістерської кваліфікаційної роботи реалізовані у вигляді сайту, доступному за <https://verbvispoll.web.app/>, та активно використовується, зокрема, в навчальному процесі на кафедрі Комп'ютерних наук ВНТУ при викладанні дисципліни "Інформаційні технології прийняття рішень".

Прикладом практичного застосування розробленого програмного забезпечення є опитування під назвою «МОЇ ПРІОРИТЕТИ В АЙТІ» та кодом доступу «it\_priorities». У даному опитуванні студенти провели дослідження що до перебігу хвороби та її наслідків. В опитуванні була створена власна шкала оцінювання: 3 - Дуже хочу!; 1 - Було б не погано; 0 - Працюю за оплату; 0 - Працюю без оплати; 0 – Байдуже; -1 - Не хочу; -3 Нізащо!. Візуальне представлення шкали оцінювання зображено на рисунку 3.19.

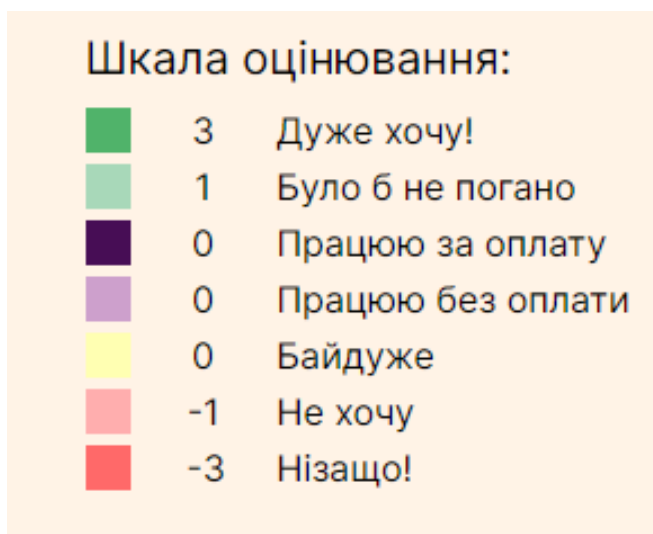


Рисунок 3.19 – Візуальне представлення шкали оцінювання

Також, в опитуванні були задані такі альтернативи: Front-end; Back-end; UI/UX; Data analytics; QA; PM; Працювати не в айтї; Взагалі не працювати. Результати опитування зображено на рисунку 3.20. Мапу

переважань зображено на рисунку 3.21. Детальні результати опитування зображено на рисунку 3.22.

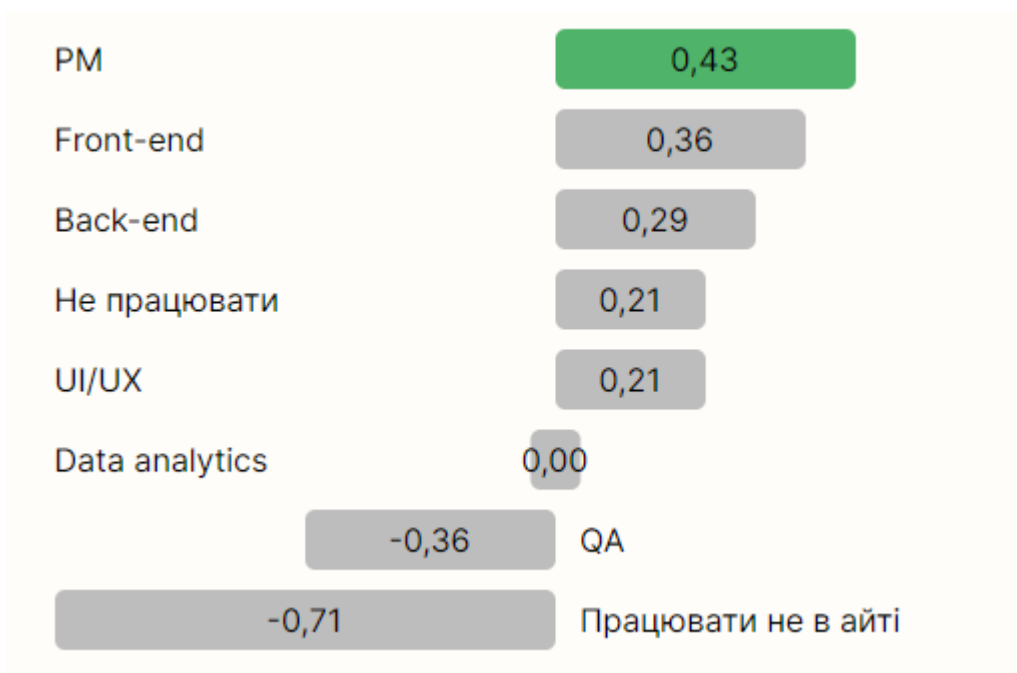


Рисунок 3.20 – Результати опитування «МОЇ ПРІОРИТЕТИ В АЙТІ»

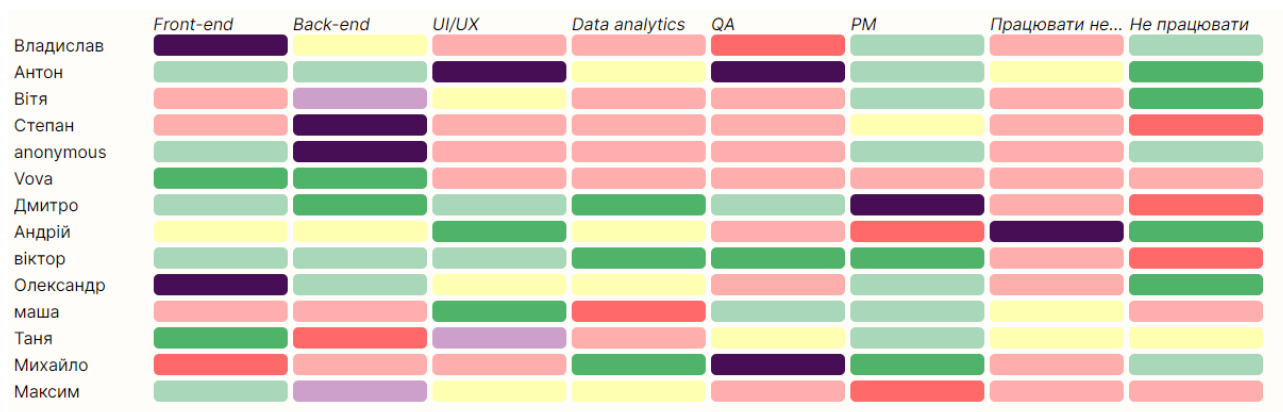


Рисунок 3.21 – Мапа переважань опитування «МОЇ ПРІОРИТЕТИ В АЙТІ»

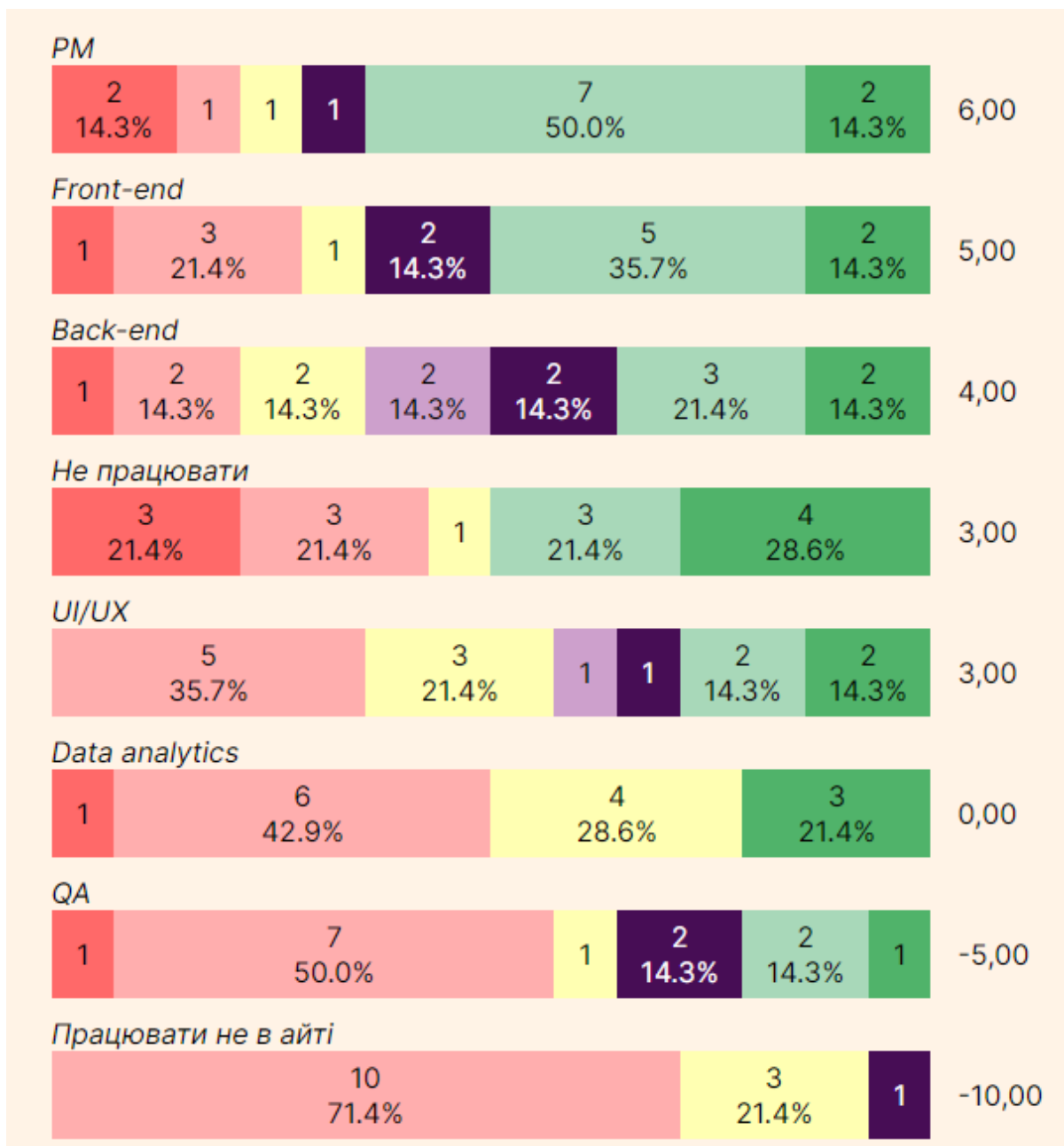


Рисунок 3.22 – Детальні результати опитування «МОЇ ПРІОРИТЕТИ В АЙТІ»

Також, відповідно до проведених досліджень, порівняємо розроблене програмне забезпечення з раніше розглянутими програмами аналогами за наступними критеріями: детальний аналіз результатів, можливість створення шкали оцінювання, відсутність реєстрації, та візуальне представлення результатів. Порівняння наведено у таблиці 3.2.

Таблиця 3.2 – Порівняння програм для проведення онлайн-опитувань

	Детальний аналіз результатів	Створення шкали оцінювання	Відсутність реєстрації	Візуальне представлення результатів
Microsoft Forms	-	-	+	-
SurveyMonkey	-	-	-	+
CollExpConstr	+	+	-	+
Google Forms	-	-	+	-
Survio	+	-	-	-
VerbVisPoll	+	+	+	+

Отже, відповідно до порівняльної таблиці програм для проведення онлайн-опитувань, можна зробити висновок про те що було розширено функціонал програм для проведення онлайн-опитувань, що свідчить про досягнення мети дослідження.

### 3.7 Висновки до розділу 3

Отже, у третьому розділі було обґрунтовано вибір основних інструментів розробки, оглянуто основні оператори мови програмування TypeScript та особливості середовища розробки VisualStudioCode. Було реалізовано та протестовано програмне забезпечення для проведення онлайн опитування, доведено його готовність для використання рядовим користувачем. Також, представлено приклад використання розробленого програмного забезпечення в реальному житті та шляхом порівняння розробленого додатку з програмами аналогами, доведено досягнення мети дослідження.



## 4 ЕКОНОМІЧНА ЧАСТИНА

Для успішного впровадження науково-технічної розробки вирішально важливим є її відповідність сучасним стандартам науково-технічного прогресу та врахування економічних аспектів. Ключовим етапом цього процесу є проведення оцінки економічної ефективності отриманих результатів науково-дослідної роботи.

Дослідження, яке представлено у магістерській роботі та присвячене розробці та аналізу "Інформаційної технології проведення онлайн-опитувань", віднесено до проектів, спрямованих на виведення на ринок. Виведення на ринок може бути визначено під час виконання самої роботи і розглядається як етап комерціалізації науково-технічної розробки. Цей напрямок розглядається як пріоритетний, оскільки отримані результати можуть бути корисними для різних зацікавлених сторін і приносити економічні вигоди.

Проте для успішної реалізації цього процесу вирішальним є здійснення пошуку зацікавленого інвестора, який виявить інтерес до втілення даного проекту, та переконання його у доцільності вкладання інвестицій у цю розробку. З цією метою були визначені наступні етапи виконання робіт:

1. Проведення комерційного аудиту науково-технічної розробки, включаючи визначення науково-технічного рівня та комерційного потенціалу.
2. Розрахунок витрат на реалізацію науково-технічної розробки.
3. Проведення розрахунку економічної ефективності впровадження та комерціалізації науково-технічної розробки для потенційного інвестора, а також обґрунтування економічної доцільності комерціалізації з точки зору інвестора.

### **4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки**

Метою проведення комерційного і технологічного аудиту дослідження за темою "Інформаційна технологія проведення онлайн-опитувань" є розширення функціональних можливостей програмного забезпечення для захисту файлів.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в таблиці 4.1 [26].

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездатність продукту в реальних
Ринкові переваги (недоліки)					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в	Технічні та споживчі властивості продукту значно кращі,
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					

Продовження таблиці 4.1

8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки наведені таблиці 4.2. Для оцінки науково-технічного рівня і комерційного потенціалу розробки експертами було запрошено трьох незалежних експертів з Вінницького національного технічного університету, кафедри «Комп'ютерні науки»: Колодний В. В. к.т.н., доцент, к.т.н., професор, Арсенюк І. Р., к.т.н., доцент, Озеранський В.С., к.т.н., доцент.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	Колодний В. В.	Арсенюк І. Р.	Озеранський В.С.
	Бали:		
1. Технічна здійсненність концепції	4	3	3
2. Ринкові переваги (наявність аналогів)	2	2	2
3. Ринкові переваги (ціна продукту)	4	4	4
4. Ринкові переваги (технічні властивості)	4	3	2
5. Ринкові переваги (експлуатаційні витрати)	4	4	4
6. Ринкові перспективи (розмір ринку)	3	2	2
7. Ринкові перспективи (конкуренція)	3	3	3
8. Практична здійсненність (наявність фахівців)	4	4	4
9. Практична здійсненність (наявність фінансів)	4	4	4
10. Практична здійсненність (необхідність нових матеріалів)	4	4	4
11. Практична здійсненність (термін реалізації)	4	4	4
12. Практична здійсненність (розробка документів)	4	4	4
Сума балів	СБ <sub>1</sub> =44	СБ <sub>2</sub> =41	СБ <sub>3</sub> =40
Середньоарифметична сума балів $СБ_c$	$\overline{СБ} = \frac{\sum_1^3 СБ_i}{3} = \frac{44 + 41 + 40}{3} = 42$		

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в таблиці 4.3 [26].

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ, розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вище середнього
21...30	Середній
11...20	Нижче середнього
0...10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою "Інформаційна технологія проведення онлайн-опитувань" становить 42 бали, що, відповідно до таблиці 4.3 рівень комерційного потенціалу розробки високий, що свідчить про комерційну важливість проведення даних досліджень.

Результатом магістерської кваліфікаційної роботи є веб-сайт для проведення онлайн-опитувань. Користувачами цієї розробки можуть бути науковці, бізнес-аналітики, урядові органи, освітні установи, соціологи та будь-які інші особи або організації, які потребують проведення опитувань та аналізу даних через онлайн-платформу.

#### 4.2 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи на тему "Інформаційна технологія проведення онлайн-опитувань", під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням

конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці.

#### Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників ( $Z_o$ ) розраховуємо у відповідності до посадових окладів працівників, за формулою [26]:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.1)$$

де  $k$  – кількість посад дослідників залучених до процесу досліджень;

$M_{ni}$  – місячний посадовий оклад конкретного дослідника, грн;

$t_i$  – число днів роботи конкретного дослідника, дн.;

$T_p$  – середнє число робочих днів в місяці,  $T_p=21$  дні.

$$Z_o = 18000 \cdot 5 / 21 = 4091 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 4.4.

Таблиця 4.4 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату, грн
Керівник проекту	18000	818,2	5	4091
Інженер-програміст	20000	909,1	35	31818
Всього				35909

#### Додаткова заробітна плата дослідників та робітників

Додаткову заробітну плату розраховуємо як 10 ... 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$Z_{\text{дод}} = (Z_o + Z_p) \cdot \frac{H_{\text{дод}}}{100\%}, \quad (4.2)$$

де  $H_{\text{дод}}$  – норма нарахування додаткової заробітної плати. Прийmemo 11%.

$$Z_{\text{дод}} = (35909) \cdot 11 / 100\% = 3950 \text{ грн.}$$

Нарахування на заробітну плату дослідників та робітників розраховуємо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$Z_n = (Z_o + Z_p + Z_{доо}) \cdot \frac{H_{zn}}{100\%} \quad (4.3)$$

де  $H_{zn}$  – норма нарахування на заробітну плату. Приймаємо 22%.

$$Z_n = (35909+3950) \cdot 22 / 100\% = 8769 \text{ грн.}$$

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за темою "Інформаційна технологія проведення онлайн-опитувань".

Витрати на матеріали ( $M$ ), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^n H_j \cdot C_j \cdot K_j - \sum_{j=1}^n B_j \cdot C_{ej}, \quad (4.4)$$

де  $H_j$  – норма витрат матеріалу  $j$ -го найменування, кг;

$n$  – кількість видів матеріалів;

$C_j$  – вартість матеріалу  $j$ -го найменування, грн/кг;

$C_{ej}$  – вартість відходів  $j$ -го найменування, грн/кг.

Проведені розрахунки зведемо до таблиці 4.5.

Таблиця 4.5 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 шт., грн	Норма витрат, шт.	Вартість витраченого матеріалу, грн
Папір А4	170	1	170
Ручка	30	1	30
Диск оптичний CD	17	1	17
Flesh-пам'ять 64	346	1	346
Всього			563
З врахуванням коефіцієнта транспортування			619,3

Витрати на комплектуючі ( $K_e$ ), які використовують при проведенні НДР на тему "Інформаційна технологія проведення онлайн-опитувань" відсутні.

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення в роботі відсутні.

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення. Програмні засоби, які були використанні при написанні магістерської роботи є безкоштовними.

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_б}{T_е} \cdot \frac{t_{вик}}{12}, \quad (4.5)$$

де  $Ц_б$  – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$  – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_е$  – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$A_{обл} = (25000 \cdot 1) / (2 \cdot 12) = 1041,67 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці 4.6.



Таблиця 4.6 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Комп'ютер	25000	2	1	1041,67
Приміщення лабораторії	250000	20	1	1041,67
Всього				2083,33

Витрати на силову електроенергію ( $B_e$ ) розраховуємо за формулою:

$$B_e = \sum_{i=1}^n \frac{W_{yi} \cdot t_i \cdot C_e \cdot K_{eni}}{\eta_i}, \quad (4.6)$$

де  $W_{yi}$  – встановлена потужність обладнання на визначеному етапі розробки, кВт;

$t_i$  – тривалість роботи обладнання на етапі дослідження, год;

$C_e$  – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), прийmemo  $C_e = 7,5$  грн;

$K_{eni}$  – коефіцієнт, що враховує використання потужності,  $K_{eni} < 1$ ;

$\eta_i$  – коефіцієнт корисної дії обладнання,  $\eta_i < 1$ .

$$B_e = 0,25 \cdot 240 \cdot 7,5 \cdot 0,5 / 0,8 = 281,25 \text{ грн.}$$

До статті «Службові відрядження» дослідної роботи на тему "Інформаційна технологія проведення онлайн-опитувань" належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов'язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень.

Витрати за статтею «Службові відрядження» розраховуємо як 20...25% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{cv} = (Z_o + Z_p) \cdot \frac{H_{cv}}{100\%}, \quad (4.7)$$

де  $H_{cv}$  – норма нарахування за статтею «Службові відрядження», прийmemo  $H_{cv} = 20\%$ .

$$B_{cv} = (35909) \cdot 20 / 100\% = 7181,82 \text{ грн.}$$

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» відсутні.

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_v = (Z_o + Z_p) \cdot \frac{H_{iv}}{100\%}, \quad (4.8)$$

де  $H_{iv}$  – норма нарахування за статтею «Інші витрати», прийmemo  $H_{iv} = 50\%$ .

$$I_v = (35909) \cdot 50 / 100\% = 17954,55 \text{ грн.}$$

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуємо як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{nzb} = (Z_o + Z_p) \cdot \frac{H_{nzb}}{100\%}, \quad (4.9)$$

де  $H_{нзв}$  – норма нарахування за статтею «Накладні (загальновиробничі) витрати», прийємо  $H_{нзв} = 100\%$ .

$$B_{нзв} = (35909) \cdot 100 / 100\% = 35909,09 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи на тему "Інформаційна технологія проведення онлайн-опитувань". розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{заг} = Z_o + Z_p + Z_{одд} + Z_n + M + K_v + B_{спец} + B_{прз} + A_{обл} + B_e + B_{св} + B_{сп} + I_v + B_{нзв}. \quad (4.10)$$

$$B_{заг} = 35909 + 3950 + 8769 + 619,3 + 2083,33 + 281,25 + 7181,82 + 17954,55 + 35909,09 \\ = 112657,43 \text{ грн.}$$

Загальні витрати  $ЗВ$  на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховується за формулою:

$$ЗВ = \frac{B_{заг}}{\eta}, \quad (4.11)$$

де  $\eta$  - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, прийємо  $\eta = 0,7$ .

$$ЗВ = 112657,43 / 0,7 = 160939,18 \text{ грн.}$$

#### **4.3 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором**

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів тієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку.

Результати дослідження проведені за темою "Інформаційна технологія проведення онлайн-опитувань" передбачають комерціалізацію протягом 3-х років реалізації на ринку.

В цьому випадку основу майбутнього економічного ефекту будуть формувати:

$\Delta N$  – збільшення кількості споживачів яким надається відповідна інформаційна послуга у періоди часу, що аналізуються;

$N$  – кількість споживачів яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки, прийmemo 1 особа

$C_o$  – вартість послуги у році до впровадження інформаційної системи, прийmemo 3000,00 грн;

$\pm \Delta C_o$  – зміна вартості послуги від впровадження результатів, прийmemo зростання на 500,00 грн.

Можливе збільшення чистого прибутку у потенційного інвестора  $\Delta \Pi_i$  для кожного із 3-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [26]:

$$\Delta \Pi_i = (\pm \Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\mathcal{G}}{100}\right), \quad (4.12)$$

де  $\lambda$  – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2021 році ставка податку на додану вартість складає 20%, а коефіцієнт  $\lambda = 0,8333$ ;

$\rho$  – коефіцієнт, який враховує рентабельність інноваційного продукту).  
Прийmemo  $\rho = 40\%$ ;

$\mathcal{G}$  – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2023 році  $\mathcal{G} = 18\%$ ;

Збільшення чистого прибутку 1-го року:

$$\Delta \Pi_1 = (1 \cdot 500 + 3000 \cdot 650) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 388715,7 \text{ грн.}$$

Збільшення чистого прибутку 2-го року:

$$\Delta \Pi_2 = (1 \cdot 500 + 3000 \cdot (650 + 600)) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 747865,94 \text{ грн.}$$

Збільшення чистого прибутку 3-го року:

$$\Delta\Pi_3 = (1 \cdot 500 + 3000 \cdot (650 + 600 + 550)) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 1076707 \text{ грн.}$$

Приведена вартість збільшення всіх чистих прибутків  $\Pi\Pi$ , що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$\Pi\Pi = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1 + \tau)^t}, \quad (4.13)$$

де  $\Delta\Pi_i$  – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

$T$  – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

$\tau$  – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні,  $\tau = 18\%$ ;

$t$  – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$$\begin{aligned} \Pi\Pi &= 388715,7 / (1 + 0,18)^1 + 747865,94 / (1 + 0,18)^2 + 1076707 / (1 + 0,18)^3 = \\ &= 1469273,51 \text{ грн.} \end{aligned}$$

Величина початкових інвестицій  $PV$ , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{инв} \cdot 3B, \quad (4.14)$$

де  $k_{инв}$  – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо  $k_{инв} = 2$ ;

$3B$  – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо грн. 160939,18

$$PV = k_{инв} \cdot 3B = 2 \cdot 160939,18 = 321878,37 \text{ грн.}$$

Абсолютний економічний ефект  $E_{abc}$  для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{abc} = ПП - PV \quad (4.15)$$

де  $ПП$  – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 1469273,51 грн;

$PV$  – теперішня вартість початкових інвестицій 321878,37 грн.

$$E_{abc} = ПП - PV = 1469273,51 - 321878,37 = 1147395,14 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій  $E_g$ , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$E_g = \sqrt[T_{жс}]{1 + \frac{E_{abc}}{PV}} - 1, \quad (4.16)$$

де  $E_{abc}$  – абсолютний економічний ефект вкладених інвестицій, грн;

$PV$  – теперішня вартість початкових інвестицій, грн;

$T_{жс}$  – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримання позитивних результатів від її впровадження, 3 роки.

$$E_g = \sqrt[3]{1 + \frac{E_{abc}}{PV}} - 1 = (1 + 1147395,14 / 321878,37)^{1/3} - 1 = 1,01.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій  $\tau_{min}$ :

$$\tau_{min} = d + f, \quad (4.17)$$

де  $d$  – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні  $d = 0,1$ ;

$f$  – показник, що характеризує ризикованість вкладення інвестицій, приймемо 0,25.

$\tau_{\min} = 0,1 + 0,25 = 0,35 < 1,01$  свідчить про те, що внутрішня економічна дохідність інвестицій  $E_g$ , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою "Інформаційна технологія проведення онлайн-опитувань" доцільно.

Період окупності інвестицій  $T_{ок}$  які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_g}, \quad (4.18)$$

де  $E_g$  – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок} = 1 / 1,01 = 1 \text{ р.}$$

$T_{ок} < 3$ -х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

#### 4.4 Висновки до розділу 4

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою "Інформаційна технологія проведення онлайн-опитувань" становить 42 бали, що, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки високий).

Також термін окупності становить 1 рік, що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою "Інформаційна технологія проведення онлайн-опитувань".

## ВИСНОВКИ

При виконанні магістерської кваліфікаційної роботи було розроблено інформаційну систему проведення онлайн-опитувань.

Розширення функціональних можливостей програмного забезпечення для проведення онлайн-опитувань полягає в можливості створення власної шкали опитування з використанням моделі кольорових схем. Інформаційна технологія реалізована у вигляді веб-ресурсу, забезпечуючи користувачам широкі можливості для аналізу результатів опитування.

Було проведено аналіз сучасних технологій проведення онлайн-опитувань, що допомогло визначити критерії до інформаційної технології, а саме можливість детального аналізу результатів опитування, створення власної шкали оцінювання, візуального представлення результатів та відсутність реєстрації.

Згідно з розглянутими критеріями, було проаналізовано переваги та недоліки програм аналогів проведення онлайн-опитувань, то було проведено порівняння з розробленим застосунком.

Розроблено алгоритм проведення онлайн-опитування, на основі якого було розроблено інтерфейс програми та діаграму діяльності користувача.

Була проведена програмна реалізація алгоритму проведення онлайн-опитувань. У результаті було розроблено кросплатформенний браузерний застосунок, створений за допомогою мови програмування TypeScript та таких технологій як HTML, SCSS, Angular та Firebase.

Було проведено тестування інформаційної технології проведення онлайн-опитувань. Аналіз результатів роботи показує що в створеній інформаційній технології є можливість створення альтернатив кількістю до десяти, можливих градацій кількістю до десяти, довільною кількістю опитаних та довільним часом проведення опитування. Також розроблена інформаційна технологія представлена у вигляді загальнодоступного сайту та розроблена за допомогою об'єктно-орієнтовної мови.



Було визначено рівень комерційного потенціалу розробки за темою "Інформаційна технологія проведення онлайн-опитувань" становить 42 бали, що, свідчить про комерційну важливість проведення даних досліджень. Загальні витрати на завершення науково-дослідної роботи та оформлення її результатів складає 160939,18 грн. Також було визначено, що термін окупності становить 1 рік, що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок

У порівнянні з розглянутими програмами-аналогами, розроблена інформаційна технологія дає можливість для детального аналізу результатів, можливості створення шкали оцінювання, відсутність реєстрації, та візуального представлення результатів. Тобто мета магістерської кваліфікаційної роботи досягнута – було розширено функціональні можливості програмного забезпечення для проведення онлайн-опитувань.

## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Тютюнов В.П. Колодний В.В, "Система підтримки проведення онлайн-опитувань з конструюванням індивідуальних шкал" в Матеріали конференції «Молодь в науці: дослідження, проблеми, перспективи», Вінниця, 2023. [Електронний ресурс]. Режим доступу: <https://conferences.vntu.edu.ua/index.php/mn/mn2024/paper/view/19331>. Дата звернення: Грудень. 2023.
2. Загальнодоступна версія веб-застосунку VerbVisPoll [Електронний ресурс] – Режим доступу до ресурсу: <https://verbvispoll.web.app/>.
3. Тютюнов В.П., Колодний В.В., "Розробка сайту для проведення онлайн-опитувань" в Матеріали конференції «LI Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації (2022)», Вінниця, 2022. [Електронний ресурс]. Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2022/paper/view/15515>  
Дата звернення: Грудень. 2023.
4. Колодний В. В. Рекомендації з покращення процедур, методів та інфомаційних технологій прийняття рішень [Електронний ресурс] / В. В. Колодний // Матеріали L науково-технічної конференції підрозділів ВНТУ, Вінниця, 10-12 березня 2021 р. – Електрон. текст. дані. – 2021. – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki2021/paper/view/11670/9803>.
5. Колодний В. В. Інтерактивна система визначення важливості критеріїв на основі аналізу трирівневих ранжувань [Текст] / В. В. Колодний, В. Зубко // Збірник матеріалів конференції «ІНТЕРНЕТОСВІТА-НАУКА-2010», Вінниця, 28 вересня - 03 жовтня 2010 р. – Вінниця : ВНТУ, 2010.
6. Система прийняття рішень на основі методу аналізу тернарних трирівневих ранжувань / Зубко В. В.: XLIV науково-технічна конференція професорсько-викладацького складу, співробітників та студентів університету –

Вінниця, ВНТУ, 2015. – [Електронний ресурс]. – Режим доступу: <http://conf.vntu.edu.ua/allvntu/2015/initki/txt/zubko.pdf>

7. Колодний, Володимир Застосування гештальт-ранжувань для виявлення переваг ОПП [Текст] / В. Колодний, В. Зубко // Proceedings of the tenth international scientific-practical conference «InternetEducation-Science» (IES-2016), Vinnytsia, 11-14 October, 2016. - Vinnytsia : VNTU, 2016.

8. Інформаційна технологія для візуалізації та виявлення переважань / В. В. Зубко // «ІНТЕРНЕТ-ОСВІТА-НАУКА-2016» : Збірник матеріалів конференції. – Вінниця : ВНТУ, 2016.

9. Колодний В. В. Інформаційна технологія візуального моделювання та обробки тернарних гештальт-ранжувань [Текст] / В. В. Колодний, Д. С. Кудрявцев // Інформаційні технології та комп'ютерна інженерія. – 2018.

10. Колодний В. В. Інформаційна технологія колективного прийняття рішень з вибором індивідуальних інструментів оцінювання альтернатив [Текст] / В. В. Колодний, Д. О. Зелінська, В. А. Гірдвайніс // Інформаційні технології та комп'ютерна інженерія. – 2021. – № 1.

11. More ways to build and share Google Forms [Електронний ресурс] – Режим доступу до ресурсу: <https://workspaceupdates.googleblog.com/2014/09/more-ways-to-build-and-share-google.html>

12. Survio. About Us [Електронний ресурс] – Режим доступу до ресурсу: <https://www.survio.com/en/about-us>

13. Розробка структурної схеми програми [Електронний ресурс] – Режим доступу до ресурсу: <https://mognovse.ru/rih-1-razrabotka-strukturnoj-shemi-programmi.html>

14. Serverless computing [Електронний ресурс] – Режим доступу до ресурсу: [https://en.wikipedia.org/wiki/Serverless\\_computing](https://en.wikipedia.org/wiki/Serverless_computing)

15. М.Петрик, О.Петрик, «Моделювання програмного забезпечення». – Програмна інженерія, 2015.

16. UML для бізнес-моделювання: для чого потрібні діаграми процесів, Олександр Марголін. [Електронний ресурс] – Режим доступу до ресурсу: <https://evergreens.com.ua/ua/articles/uml-diagrams.html>
17. Алгоритм функціонування [Електронний ресурс] – Режим доступу до ресурсу: <https://lektsii.net/3-164876.html>
18. Structured program theorem [Електронний ресурс] – Режим доступу до ресурсу: [https://en.wikipedia.org/wiki/Structured\\_program\\_theorem](https://en.wikipedia.org/wiki/Structured_program_theorem)
19. JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <https://learn.javascript.ru/intro>
20. The Good and the Bad of TypeScript [Електронний ресурс] – Режим доступу до ресурсу: <https://www.altexsoft.com/blog/typescript-pros-and-cons/>
21. Software Framework [Електронний ресурс] – Режим доступу до ресурсу: [https://en.wikipedia.org/wiki/Software\\_framework](https://en.wikipedia.org/wiki/Software_framework)
22. Angular [Електронний ресурс] – Режим доступу до ресурсу: <https://metanit.com/web/angular2/1.1.php>
23. Tamplin, James. Firebase is Joining Google!. [Електронний ресурс] – Режим доступу до ресурсу: <https://firebase.blog/posts/2014/10/firebase-is-joining-google>
24. Microsoft TypeScript: Can the father of C# save us from the tyranny of JavaScript? [Електронний ресурс] – Режим доступу до ресурсу: <https://firebase.blog/posts/2014/10/firebase-is-joining-google>
25. Learn to code with Visual Studio Code [Електронний ресурс] – Режим доступу до ресурсу: <https://code.visualstudio.com/learn>
26. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с

## **ДОДАТКИ**

## Додаток А (обов'язковий)

**Протокол перевірки кваліфікаційної роботи на наявність текстових  
запозичень**

Назва роботи: Інформаційна технологія проведення онлайн-опитувань

Тип роботи: магістерська кваліфікаційна робота  
(БДР, МКР)

Підрозділ кафедра комп'ютерних наук, ФІІТА  
(кафедра, факультет)

**Показники звіту подібності Unicheck**


Оригінальність 86,3% Схожість 13,7%

**Аналіз звіту подібності (відмітити потрібне):**

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

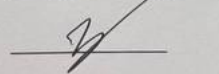
Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи



Тютюнов В.П.

Керівник роботи

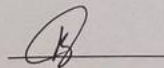


Колодний В.В.

**Опис прийнятого рішення**

Магістерську кваліфікаційну роботу допущено до захисту

Особа, відповідальна за перевірку



Озеранський В.С.

## Додаток Б (обов'язковий)

### Лістинг програми

```
@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.scss']
})
export class AppComponent implements OnInit {
  title = 'tpr';
  loading$: Observable<boolean>;

  constructor(private mainService: MainService, private router: Router, private errorService: ErrorService,
    private snackBar: MatSnackBar) {
    this.loading$ = this.mainService.loading$;
  }

  ngOnInit(): void {
    let id = localStorage.getItem('id');

    if (id) {
      this.mainService.findById(id);
    }

    this.router.events.pipe(filter(e => e instanceof NavigationEnd), take(1))
      .subscribe((res: NavigationEnd) => {
        if (res.url.includes('?') && res.url.includes('=')) {
          let id = res.url.split('?').pop().split('=')?.pop();

          if (id) {
            this.mainService.findById(id, this.showError.bind(this));
          }
        }
      })
  }

  showError() {
    this.snackBar.openFromComponent(ConnectErrorComponent, {
      duration: 5000,
    });
  }
}
```

```

}

export interface IData {
  name: string;
  description: string;
  id: string;
  // count: number;
  type: number;
  alternatives: Array<string>;
  votes?: Array<IVote>;
  customMetric?: any;
}

export interface IVote {
  name: string;
  votes: Array<string | number>;
  time?: string;
}

@Injectable({
  providedIn: 'root'
}))

export class MainService {
  private adminSubj = new BehaviorSubject(false);
  public admin$ = this.adminSubj.asObservable();
  private loadingSubj = new BehaviorSubject(false);
  public loading$ = this.loadingSubj.asObservable();
  public defaultMetrics = [
    {
      type: 'centered',
      values: ['Категорично проти', 'Проти', 'Байдуже (Утримався)', 'За', 'Категорично за'],
    },
    {
      type: 'ladder',
      values: ['1', '2', '3', '4', '5']
    }
  ];
  public deletedMetrics = [
    {

```



```

    type: 'centered',
    values: ['Дуже погано', 'Погано', 'Байдуже (Не знаю)', 'Добре', 'Дуже добре'],
  }
];
public metrics = [];
private votesSubj = new BehaviorSubject<Array<IVote>>([]);
public votes$ = this.votesSubj.asObservable().pipe(filter(r => !!r));
private optionsSubj = new BehaviorSubject<IData | 'empty'>(undefined);
public options$ = this.optionsSubj.asObservable().pipe(filter(r => !!r));
public votingName = "";
private ref = this.firestore.collection('interviews');
constructor(
  private router: Router,
  private firestore: AngularFirestore,
  private dialog: MatDialog,
  private translate: TranslateService
) {
  this.metrics = [...this.defaultMetrics, ...this.deletedMetrics];
  this.votes$.pipe(take(1)).subscribe(res => {
    if (res.length && this.optionsSubj.value !== 'empty' && this.optionsSubj.value?.id) {
      this.ref.doc(this.optionsSubj.value?.id).update({ votes: res });
    }
  });
}
public translateAlternatives(): void {
  this.metrics[0].values = this.translate.instant('alts')[0];
  this.metrics[1].values = this.translate.instant('alts')[1];
  // this.metrics[2].values = this.translate.instant('alts')[2];
}
public create(data: IData): void {
  this.router.navigate(['/quiz']);
  this.optionsSubj.next(data);
  this.ref.doc(data.id).set(data);
  localStorage.setItem('id', data.id);
  this.findById(data.id);
}
}

```

```

public addVote(vote: IVote): void {
  this.router.navigate(['/quiz']);
  this.ref.get().pipe(take(1)).subscribe(res => {
    const data: any = res.docs.map(doc => doc.data());
    const coll = data.find(el => el.id === (this.optionsSubj.value as any).id);
    const votes = coll.votes;
    let i = 0;
    votes.forEach(el => {
      if (el.name === `${vote.name}${i ? ' (' + i + ')' : ''}`) {
        i++;
      }
    });
    if (i > 0) {
      vote.name += ' (' + i + ')';
    }
    votes.push(vote);
    this.votesSubj.next(votes);
    if (this.optionsSubj.value !== 'empty') {
      this.ref.doc(this.optionsSubj.value.id).update({
        votes
      });
    }
  });
}

public startVoting(name: string): void {
  this.votingName = name.trim();
  this.router.navigate(['/quiz/voting']);
}

public findById(id: string, callback?): void {
  this.loadingSubj.next(true);
  this.ref.get().pipe(take(1)).subscribe(res => {
    const data: any = res.docs.map(doc => doc.data());
    const coll = data.find(el => el.id === id);
    this.loadingSubj.next(false);
    if (coll) {
      this.votesSubj.next(coll.votes);
    }
  });
}

```

```

    this.optionsSubj.next(coll);
    this.router.navigate(['quiz']);
    localStorage.setItem('id', id);
    this.dialog.closeAll();
  } else if (callback) {
    this.optionsSubj.next('empty');
    callback();
  }
});
}

public remove(id: string): Promise<any> {
  this.loadingSubj.next(true);
  return this.ref.doc(id).delete().finally(() => {
    this.loadingSubj.next(false);
  });
}

public checkIfExist(id: string): Observable<boolean> {
  return this.ref.get().pipe(map(res => {
    const data: any = res.docs.map(doc => doc.data());
    const coll = data.find(el => el.id === id);
    if (coll) {
      return true;
    }
    return false;
  }));
}

public clearAll(): void {
  this.votesSubj.next([]);
  this.optionsSubj.next(null);
  localStorage.removeItem('id');
}

public secret(): void {
  this.adminSubj.next(true);
}

public setCustomMetric(customMetric): void {
  this.metrics[100] = {

```

```

        type: 'custom',
        values: customMetric
    };
}
}

@Component({
    selector: 'app-main',
    templateUrl: './main.component.html',
    styleUrls: ['./main.component.scss'],
})
export class MainComponent implements OnInit, OnDestroy {
    public name: string;
    public description: string;
    public votes: Array<IVote>;
    public alternatives: Array<string>;
    public markType: string;
    public code: string;
    public extended = false;
    public extended2 = false;
    public times = [];
    private subs = [];
    public bars = [];
    public math: Math = Math;
    public mainResults = [];
    public res2MoreStates = [
        false,
    ];
    public metricData;
    public betCustomColor;
    constructor(
        private dialog: MatDialog,
        private mainService: MainService,
        private errorService: ErrorService,
        private router: Router
    ) {

```

```

let id = localStorage.getItem('id');
if (!id) {
  this.errorService.showRouteError();
  this.router.navigate(['home']);
}
this.subs.push(
  this.mainService.options$.subscribe((res) => {
    if (res === 'empty') {
      this.errorService.showRouteError();
      this.router.navigate(['home']);
      return;
    }
    if (res.type === 100) {
      this.mainService.setCustomMetric(res.customMetric);
    }
    this.name = res.name;
    this.description = res.description;
    this.alternatives = res.alternatives;
    this.markType = this.mainService.metrics[res.type].type;
    this.metricData = this.mainService.metrics[res.type];
    if (this.metricData.type === 'custom') {
      const maxValue = Math.max(...this.metricData.values.map(v => v.value));
      const maxColor = this.metricData.values.find(v => v.value === maxValue).color;
      this.betCustomColor = maxColor;
    }
    this.code = res.id;
    this.subs.push(
      this.mainService.votes$.subscribe((res) => {
        this.bars = [];
        this.votes = res;
        this.times[0] = res[0]?.time;
        this.times[1] = res[res.length - 1]?.time;
        this.alternatives.forEach((alt, i) => {
          this.bars.push({ i, marks: [] });
          this.votes.forEach((vote, j) => {
            this.bars[i].marks.push(vote.votes[j]);
          });
        });
      });
    );
  });
);

```

```

    });
    this.bars[i].marks.sort();
  });
  this.generateMain();
})
);
})
);
}
}
ngOnDestroy(): void {
  this.subs.forEach((sub) => {
    sub.unsubscribe();
  });
}
public newExpert(): void {
  this.dialog.open(NewExpertComponent, {
    scrollStrategy: new NoopScrollStrategy(),
  });
}
public getNumberOfMarks(marks, i) {
  return marks.filter((mark) => mark === i).length;
}
public copy(text) {
  navigator.clipboard.writeText(text);
  this.errorService.showCopied();
}
public extend() {
  this.extended = !this.extended;
}
public extend2() {
  this.extended2 = !this.extended2;
}
public generateMain() {
  let arr = [];
  let max = -100000;

```

```

let min = 100000;
if (this.markType === 'custom') {
  this.bars.forEach((v, i) => {
    if (!this.bars[i]) {
      return;
    }
    let mark = this.getCustomBarTotal(this.bars[i].marks) / this.bars[i].marks.length;
    if (mark > max) {
      max = mark;
    }
    if (mark < min) {
      min = mark;
    }
    arr.push({ mark });
  });
  arr = arr.map((v, i) => {
    return {
      ...v,
      name: this.alternatives[i],
    };
  });
  arr.forEach((v, i) => {
    if (arr[i].mark > 0) {
      arr[i].per = Math.abs(
        (v.mark * 100) / Math.max(Math.abs(max), Math.abs(min))
      );
    } else if (arr[i].mark < 0) {
      arr[i].per = Math.abs(
        (v.mark * 100) / Math.max(Math.abs(max), Math.abs(min))
      );
    }
  });
  this.mainResults = arr.sort((a, b) => a.mark - b.mark);
  const best = this.mainResults[this.mainResults.length - 1];
  if (best.mark > 0) {
    const allBests = this.mainResults.filter(v => v.mark === best.mark);
  }
}

```

```

    allBests.forEach(v => v.customColor = this.betCustomColor);
  }
  return;
}
this.bars.forEach((v, i) => {
  if (this.bars[i]) {
    let mark =
      this.getNumberOfMarks(this.bars[i].marks, 0) *
      (this.markType === 'centered' ? -2 : -2) +
      this.getNumberOfMarks(this.bars[i].marks, 1) *
      (this.markType === 'centered' ? -1 : -1) +
      this.getNumberOfMarks(this.bars[i].marks, 2) *
      (this.markType === 'centered' ? 0 : 0) +
      this.getNumberOfMarks(this.bars[i].marks, 3) *
      (this.markType === 'centered' ? 1 : 1) +
      this.getNumberOfMarks(this.bars[i].marks, 4) *
      (this.markType === 'centered' ? 2 : 2);
    if (mark > max) {
      max = mark;
    }
    if (mark < min) {
      min = mark;
    }
    arr.push({ mark });
  }
});
arr = arr.map((v, i) => {
  return {
    ...v,
    name: this.alternatives[i],
  };
});
if (this.markType === 'centered') {
  arr.forEach((v, i) => {
    if (arr[i].mark > 0) {
      arr[i].per = Math.abs(

```



```

    (v.mark * 100) / Math.max(Math.abs(max), Math.abs(min))
  );
} else if (arr[i].mark < 0) {
  arr[i].per = Math.abs(
    (v.mark * 100) / Math.max(Math.abs(max), Math.abs(min))
  );
}
if (arr[i].mark === 0) {
  arr[i].color = 2;
} else if (arr[i].mark === max) {
  arr[i].color = 4;
} else if (arr[i].mark === min) {
  arr[i].color = 1;
} else if (arr[i].mark > 0) {
  arr[i].color = 3;
} else if (arr[i].mark < 0) {
  arr[i].color = 1;
}
});
} else if (this.markType === 'ladder') {
arr.forEach((v, i) => {
  if (arr[i].mark > 0) {
    arr[i].per = Math.abs(
      (v.mark * 100) / Math.max(Math.abs(max), Math.abs(min))
    );
  } else if (arr[i].mark < 0) {
    arr[i].per = Math.abs(
      (v.mark * 100) / Math.max(Math.abs(max), Math.abs(min))
    );
  }
  if (arr[i].mark === 0) {
    arr[i].color = 2;
  } else if (arr[i].mark === max) {
    arr[i].color = 4;
  } else if (arr[i].mark === min) {
    arr[i].color = 2;
  }
});
}

```

```

    } else if (arr[i].mark > 0) {
        arr[i].color = 2;
    } else if (arr[i].mark < 0) {
        arr[i].color = 2;
    }
});
}
this.mainResults = arr;
}
public toTop() {
    document.body.scroll(0, 0);
}
public calcZZ(marks) {
    return Math.round(
        ((this.getNumberOfMarks(marks, 3) + this.getNumberOfMarks(marks, 4)) /
        marks.length) *
        100
    );
}
public calcZNZ(marks) {
    return Math.round(
        ((this.getNumberOfMarks(marks, 0) + this.getNumberOfMarks(marks, 1)) /
        marks.length) *
        100
    );
}
public calcIN(marks) {
    return Math.round((this.getNumberOfMarks(marks, 2) / marks.length) * 100);
}
public calcPD(marks) {
    return Math.round(
        ((2 *
        this.math.min(
            this.getNumberOfMarks(marks, 0) + this.getNumberOfMarks(marks, 1),
            this.getNumberOfMarks(marks, 3) + this.getNumberOfMarks(marks, 4)
        )) /

```

```
marks.length) *
    100
);
}
public calcCON(marks) {
    return Math.round(
        ((2 *
            this.math.min(
                this.getNumberOfMarks(marks, 0),
                this.getNumberOfMarks(marks, 4)
            )) /
            marks.length) *
            100
        );
}
public calcLO(marks) {
    return Math.round((this.getNumberOfMarks(marks, 4) / marks.length) * 100);
}
public calcALO(marks) {
    return Math.round((this.getNumberOfMarks(marks, 0) / marks.length) * 100);
}
public calcPO(marks) {
    return Math.round(
        ((this.getNumberOfMarks(marks, 1) +
            this.getNumberOfMarks(marks, 2) +
            this.getNumberOfMarks(marks, 3)) /
            marks.length) *
            100
        );
}
public calcOS(marks) {
    if (this.markType === 'ladder') {
        return Math.round(
            (Math.max(
                this.getNumberOfMarks(marks, 0),
                this.getNumberOfMarks(marks, 1),
```

```

        this.getNumberOfMarks(marks, 2),
        this.getNumberOfMarks(marks, 3),
        this.getNumberOfMarks(marks, 4)
    ) /
    marks.length) *
    100
);
} else if (this.markType === 'centered') {
    return Math.round(
        (Math.abs(
            this.getNumberOfMarks(marks, 0) +
            this.getNumberOfMarks(marks, 1) -
            this.getNumberOfMarks(marks, 3) -
            this.getNumberOfMarks(marks, 4)
        ) /
        marks.length) *
        100
    );
}
return 0;
}

public getCustomBarTotal(marks): number {
    const gradations = this.metricData.values;
    const values = marks.map((mark, i) => {
        return gradations[mark].value;
    });
    return values.reduce((a, b) => a + b, 0);
}

public getTextColorForBackground(backgroundColor) {
    if (!backgroundColor) {
        return '#000000';
    }
    // Convert the input color to RGB format
    let r, g, b;

```

```

if (backgroundColor.startsWith('#')) {
  backgroundColor = backgroundColor.slice(1);
}
if (backgroundColor.length === 3) {
  r = parseInt(backgroundColor[0] + backgroundColor[0], 16);
  g = parseInt(backgroundColor[1] + backgroundColor[1], 16);
  b = parseInt(backgroundColor[2] + backgroundColor[2], 16);
} else if (backgroundColor.length === 6) {
  r = parseInt(backgroundColor.slice(0, 2), 16);
  g = parseInt(backgroundColor.slice(2, 4), 16);
  b = parseInt(backgroundColor.slice(4, 6), 16);
} else {
  throw new Error('Invalid color format');
}
// Calculate the brightness of the background color
const brightness = (r * 299 + g * 587 + b * 114) / 1000;
// Return white or black based on the brightness
return brightness > 128 ? '#000000' : '#ffffff';
}
public getMetricDataReverse() {
  const copy = JSON.parse(JSON.stringify(this.metricData.values));
  return copy.reverse();
}
}

@Component({
  selector: 'app-create-dialog',
  templateUrl: './create-dialog.component.html',
  styleUrls: ['./create-dialog.component.scss']
})
export class CreateDialogComponent implements AfterViewInit, OnDestroy {
  public selectedPage = 0;
  public form: FormGroup;
  public alternativesForm: FormGroup | undefined;
  public data: IData | undefined;
  public selectedMetric = -1;

```

```

public metrics = [];
private subs = [];
public customMetricValue;
constructor(
  private mainService: MainService,
  public dialog: MatDialog,
  private errorService: ErrorService,
  private translateService: TranslateService,
  private cdr: ChangeDetectorRef
) {
  this.mainService.translateAlternatives();
  this.metrics = this.mainService.defaultMetrics;
  this.form = new FormGroup({
    name: new FormControl("", [Validators.required, Validators.maxLength(50)]),
    description: new FormControl("", [Validators.maxLength(500)]),
    id: new FormControl("", [Validators.required, Validators.minLength(5), Validators.maxLength(20)]),
    // count: new FormControl('2', [Validators.required, Validators.max(10), Validators.min(1)]),
  });
}
ngAfterViewInit(): void {
  document.querySelector('mat-button-toggle:first-child').dispatchEvent(new Event('click'));
}
ngOnDestroy(): void {
  this.subs.forEach(sub => {
    sub.unsubscribe();
  });
}
public openMetricSelectPage(e): void {
  e.preventDefault();
  this.selectedPage = 1;
}
public onSubmit(e): void {
  e.preventDefault();
  if (this.form.value.name.trim().length === 0 || this.form.value.id.trim().length === 0) {
    this.form.controls.name.setValue(this.form.value.name.trim());
    this.form.controls.id.setValue(this.form.value.id.trim());
  }
}

```

```

return;
}
this.subs.push(this.mainService.checkIfExist(this.form.value.id.trim()).subscribe(exist => {
  if (!exist) {
    this.data = {
      name: this.form.value.name.trim(),
      // count: this.form.value.count,
      description: this.form.value.description,
      id: this.form.value.id.trim(),
      type: this.selectedMetric,
      alternatives: [],
      votes: [],
      customMetric: {
        ...this.customMetricValue
      }
    };
    const values = this.alternativesForm?.value ? Object.values(this.alternativesForm?.value) : [];
    this.alternativesForm = new FormGroup({});
    const length = values.length > 2 ? values.length : 2;
    for (let i = 0; i < length; i++) {
      const text = values[i] || (i + 1) + ' ' + this.translateService.instant('home.alt');
      this.alternativesForm.addControl(
        `${i}`,
        new FormControl(`${text}`, [Validators.required, Validators.maxLength(50)])
      );
      this.data.alternatives.push("");
    }
  } else {
    this.errorService.showCreateError();
    this.selectedPage = 0;
  }
}));
this.selectedPage = 3;
}
public addAlternative(e): void {
  e.preventDefault();

```

```

const values = Object.values(this.alternativesForm?.value);
this.alternativesForm = new FormGroup({});
this.data.alternatives.push("");
for (let i = 0; i < this.data.alternatives.length; i++) {
  const text = values[i] || (i + 1) + ' ' + this.translateService.instant('home.alt');
  this.alternativesForm.addControl(
    `${i}`,
    new FormControl(`${text}`, [Validators.required, Validators.maxLength(50)])
  );
}
}

public removeAlternative(e): void {
  e.preventDefault();
  const values = Object.values(this.alternativesForm?.value);
  this.alternativesForm = new FormGroup({});
  this.data.alternatives.pop();
  for (let i = 0; i < this.data.alternatives.length; i++) {
    const text = values[i] || (i + 1) + ' ' + this.translateService.instant('home.alt');
    this.alternativesForm.addControl(
      `${i}`,
      new FormControl(`${text}`, [Validators.required, Validators.maxLength(50)])
    );
  }
}

public onSubmitAlternatives(): void {
  for (const i in this.alternativesForm?.value) {
    for (const j in this.alternativesForm?.value) {
      if (this.alternativesForm?.value[i] === this.alternativesForm?.value[j] && i !== j) {
        this.errorService.showAltErrors();
        return;
      }
    }
  }
  if (this.alternativesForm.value[i].trim().length === 0) {
    this.alternativesForm.controls[i].setValue("");
    return;
  }
}

```



```

    this.data.alternatives[i] = this.alternativesForm?.value[i];
  }
  this.selectedPage = 1; }
public close() {
  this.dialog.closeAll();
}
public generateID(e) {
  e.preventDefault();
  var result = "";
  var characters = 'ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789';
  var charactersLength = characters.length;
  for (var i = 0; i < 12; i++) {
    result += characters.charAt(Math.floor(Math.random() * charactersLength));
  }
  this.form.controls.id.setValue(result);
}

public createCustomMetric(): void {
  this.selectedPage = 2;
}
public submitCustomMetric(customMetric): void {
  console.log(customMetric);
  if (!customMetric) {
    this.selectedPage = 1;
    return;
  }
  this.customMetricValue = customMetric;
  this.selectedMetric = 100;
  this.data.type = this.selectedMetric;
  this.data.customMetric = this.customMetricValue;
  this.mainService.create(this.data);
  this.dialog.closeAll();
}
public finish(): void {
  this.data.type = this.selectedMetric;

```

```

    this.mainService.create(this.data);
    this.dialog.closeAll();
}
public back(index): void {
    this.selectedPage = index;
}
}
@Component({
    selector: 'app-custom-metric',
    templateUrl: './custom-metric.component.html',
    styleUrls: ['./custom-metric.component.scss'],
})
export class CustomMetricComponent implements OnInit {
    @Output() submitForm = new EventEmitter<any>();
    colors: string[] = [
        '#ff0000',
        '#00ff00'
    ];
    public form = new FormGroup({
        gradations: new FormArray([
            new FormGroup({
                name: new FormControl('1 ' + this.translateService.instant('home.config.gradation'),
[Validators.required]),
                color: new FormControl('#000000', [Validators.required]),
                value: new FormControl(0, [Validators.required]),
            }),
            new FormGroup({
                name: new FormControl('2 ' + this.translateService.instant('home.config.gradation'),
[Validators.required]),
                color: new FormControl('#000000', [Validators.required]),
                value: new FormControl(0, [Validators.required]),
            }),
        ]),
    });
    get gradations() {
        return this.form.controls.gradations as any;
    }
}

```

```

}
get isGradationsDisabled() {
    return this.gradations.controls.length >= 10;
}
get isGradationDeleteDisabled() {
    return this.gradations.controls.length <= 2;
}
constructor(private translateService: TranslateService) {}
ngOnInit(): void {}
public addGradation(): void {
    if (this.isGradationsDisabled) {
        return;
    }
    (this.form.controls.gradations as FormArray).push(
        new FormGroup({
            name: new FormControl(
                (this.gradations.length + 1).toString() + ' ' + this.translateService.instant('home.config.gradation')
            ),
            color: new FormControl('#000000'),
            value: new FormControl(0),
        })
    );
}
public deleteGradation(index: number): void {
    if (this.isGradationDeleteDisabled) {
        return;
    }
    (this.form.controls.gradations as FormArray).removeAt(
        index
    );
}
public onChangeColor(color: string, index: number): void {
    this.gradations.controls[index].get('color').setValue(color);
    this.colors[index] = color;
}
}

```

```
public submit(): void {  
    const sortedGradations = this.form.value.gradations.sort((a, b) => {  
        return a.value - b.value;  
    });  
    this.submitForm.emit(sortedGradations);  
}  
public back(): void {  
    this.submitForm.emit(null);  
}  
}
```

## Додаток В (обов'язковий)

## ІЛЮСТРАТИВНА ЧАСТИНА

## ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ПРОВЕДЕННЯ ОНЛАЙН-ОПИТУВАНЬ

Виконав: студент 2-го курсу,  
групи 2КН-22м  
спеціальності 122 «Комп'ютерні науки»  
(шифр і назва напрямку підготовки, спеціальності)

Гютюнов В.П.

(прізвище та ініціали)

Керівник: к.т.н., доцент каф. КН

Колодний В.В.

(прізвище та ініціали)

« 07 » 12 2023 р.

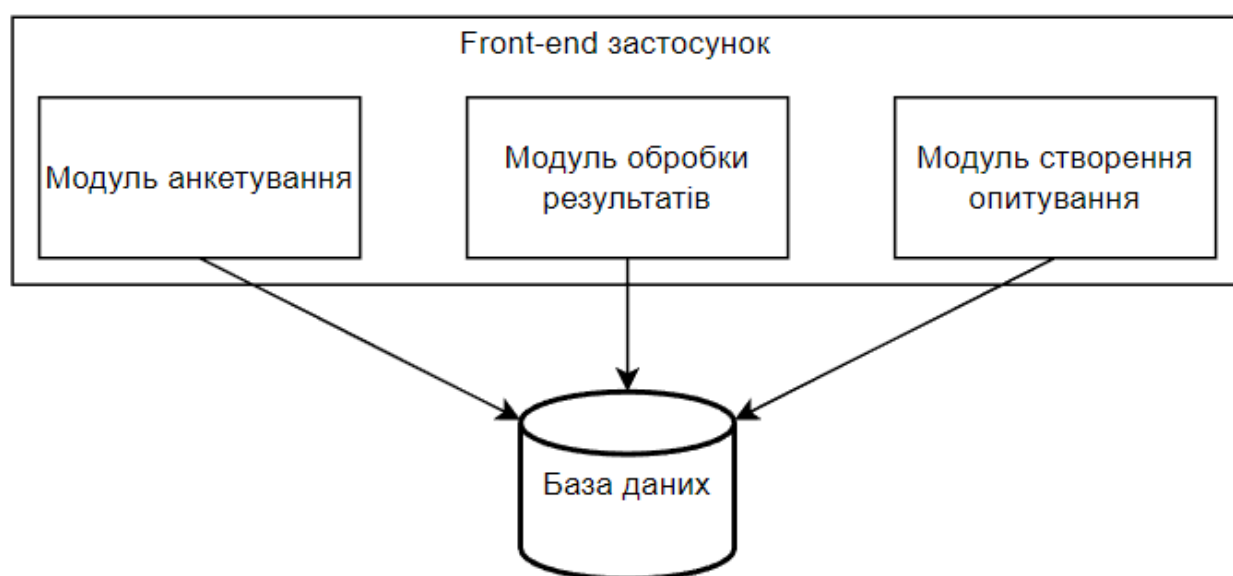


Рисунок В.1 – Структурна схема програмного забезпечення для сайту проведення онлайн-опитувань

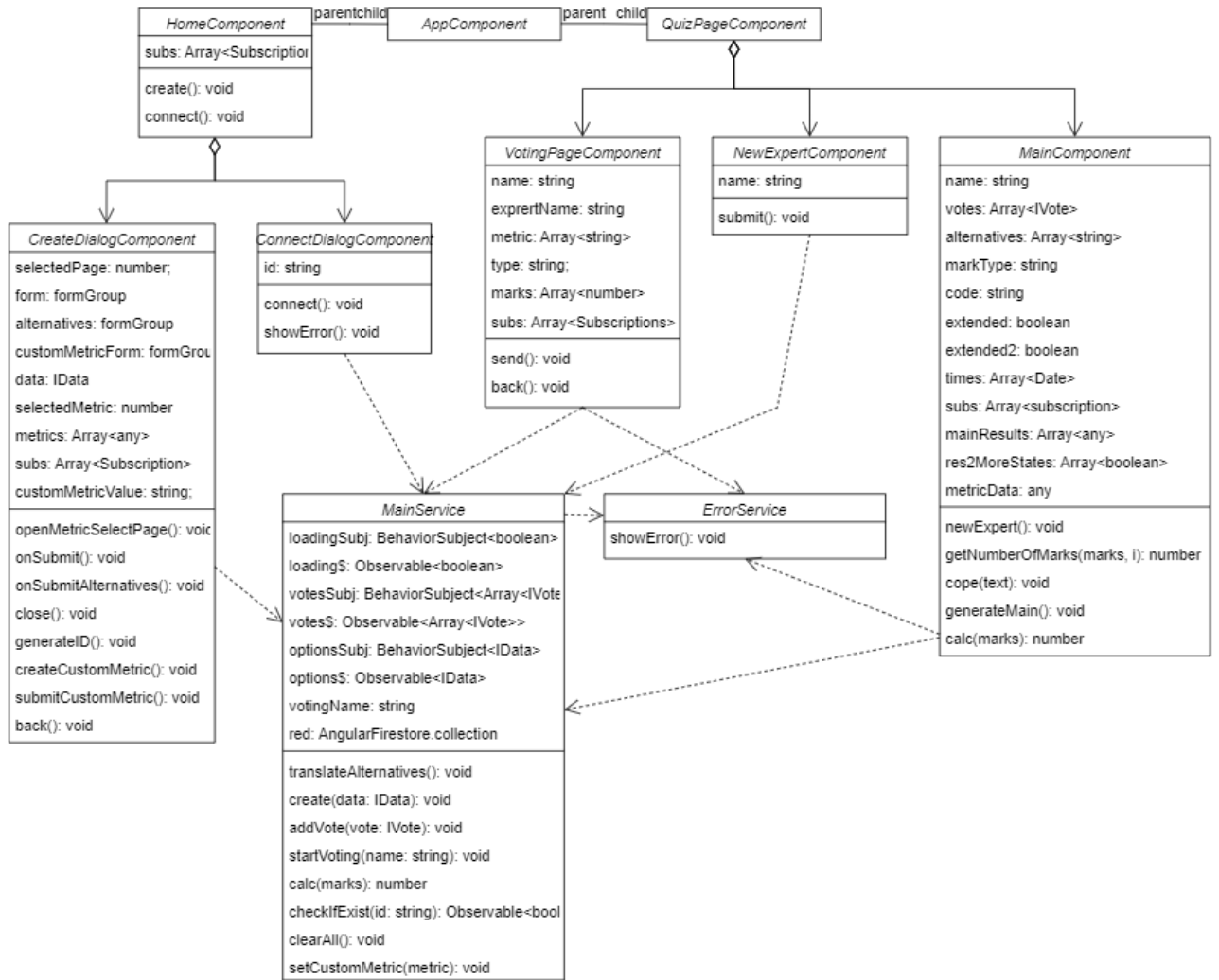


Рисунок В.2 – Діаграма класів сайту для проведення онлайн-опитувань

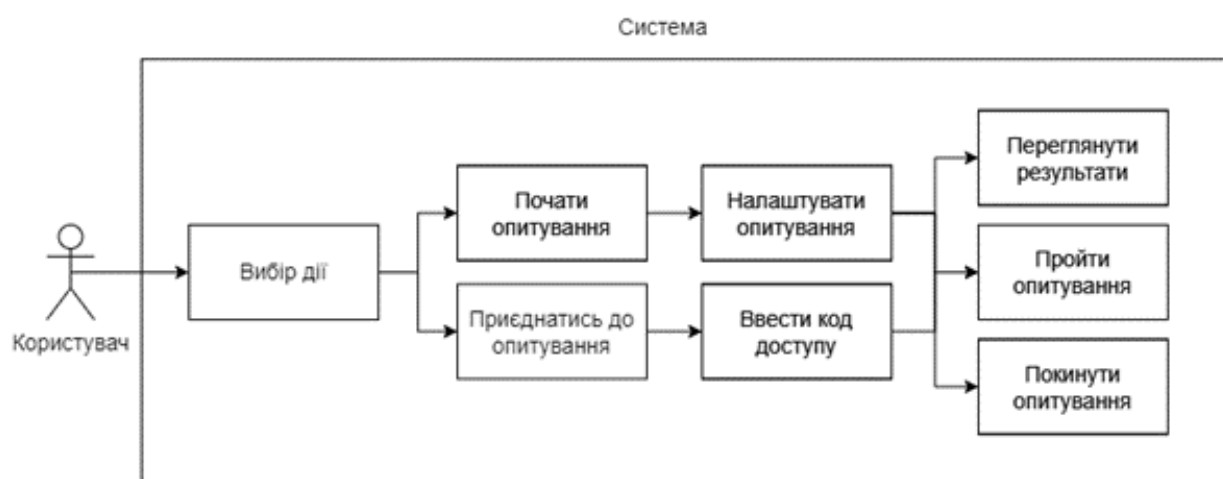


Рисунок В.3 – Діаграма прецедентів сайту для проведення онлайн-опитувань



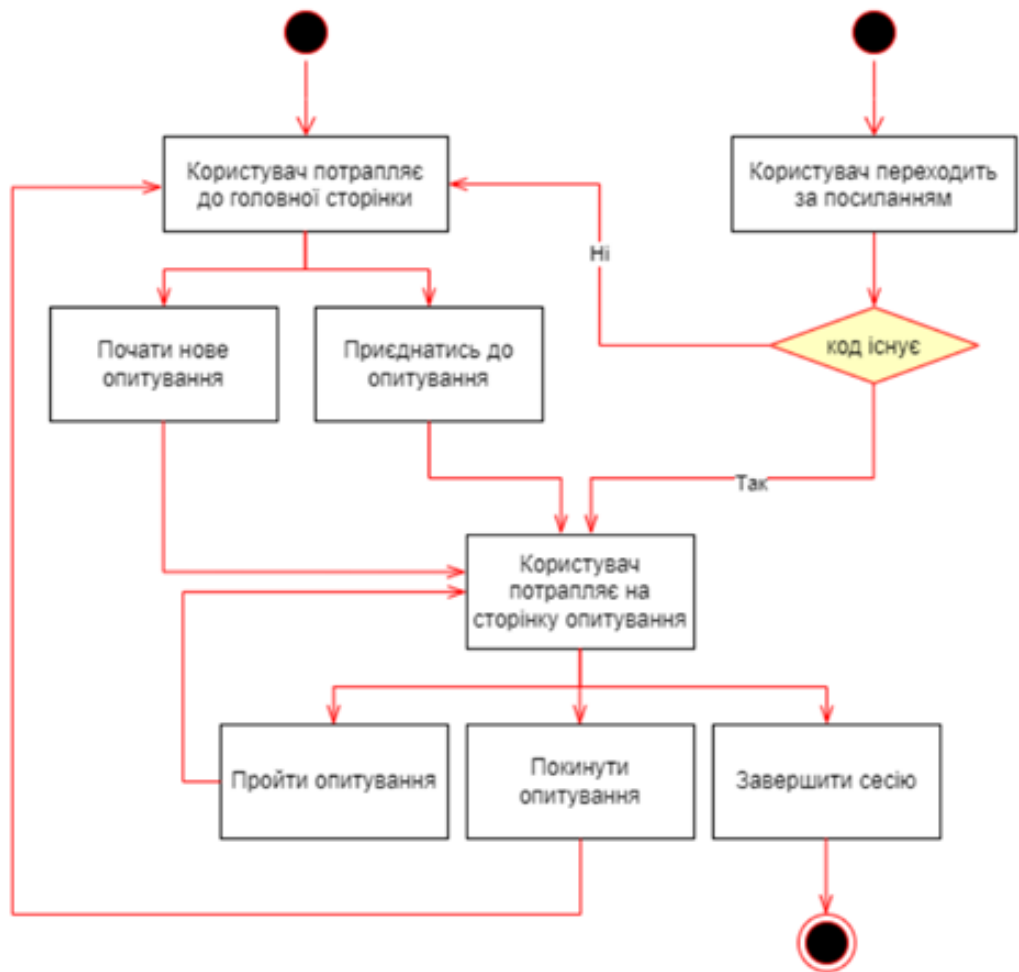


Рисунок В.4 – Діаграма діяльності сайту для проведення онлайн-опитувань

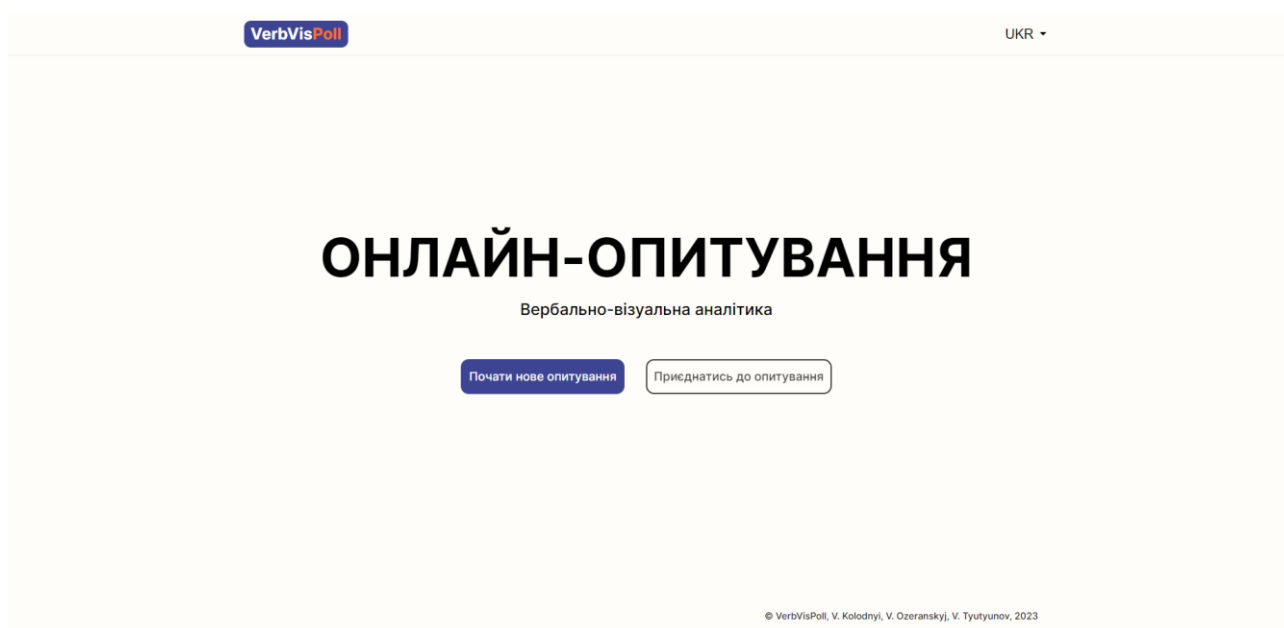









Рисунок В.5 – Головна сторінка інформаційної технології проведення онлайн-опитувань

←

**Конструктор шкал** Додати градацію +

Назва градації	Бал		×	Назва градації	Бал		×
Дуже хочу!	3		×	Було б не погано	1		×
Байдуже	0		×	Не хочу	-1		×
Нізащо!	<input type="text" value="-3"/>		×	Працюю без оплати	0		×
Працюю за оплату	0		×				

Далі

Рисунок В.6 – Форма створення власної шкали опитування

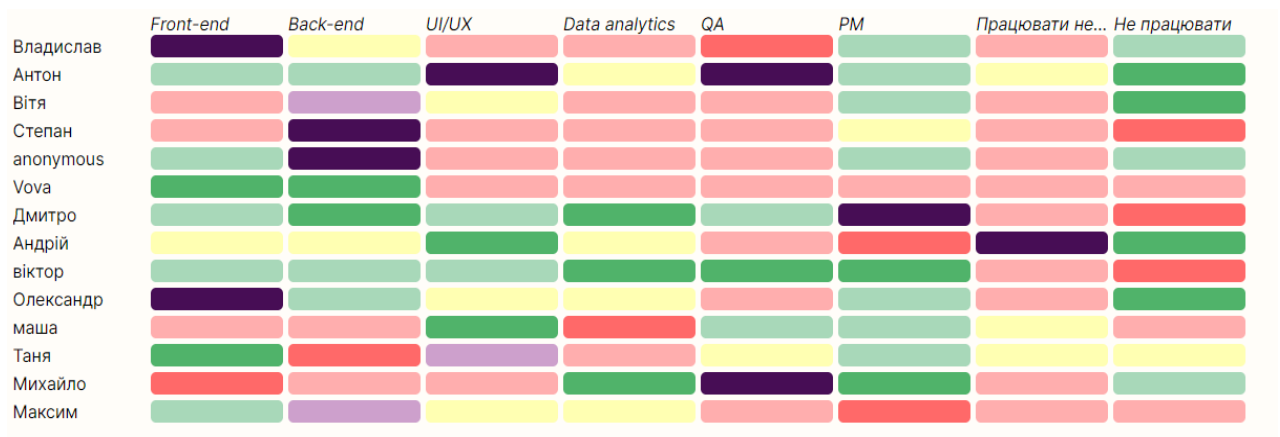


Рисунок В.7 – Приклад мапи переважань результатів опитування

## Додаток Г (довідниковий) Інструкція користувача

### 1. Початкова сторінка

На початковій сторінці можна створити нове опитування або приєднатись до існуючого натиснувши на відповідні кнопки. Також на верхній панелі додатку є меню для вибору мови. Доступні мови: українська та англійська.

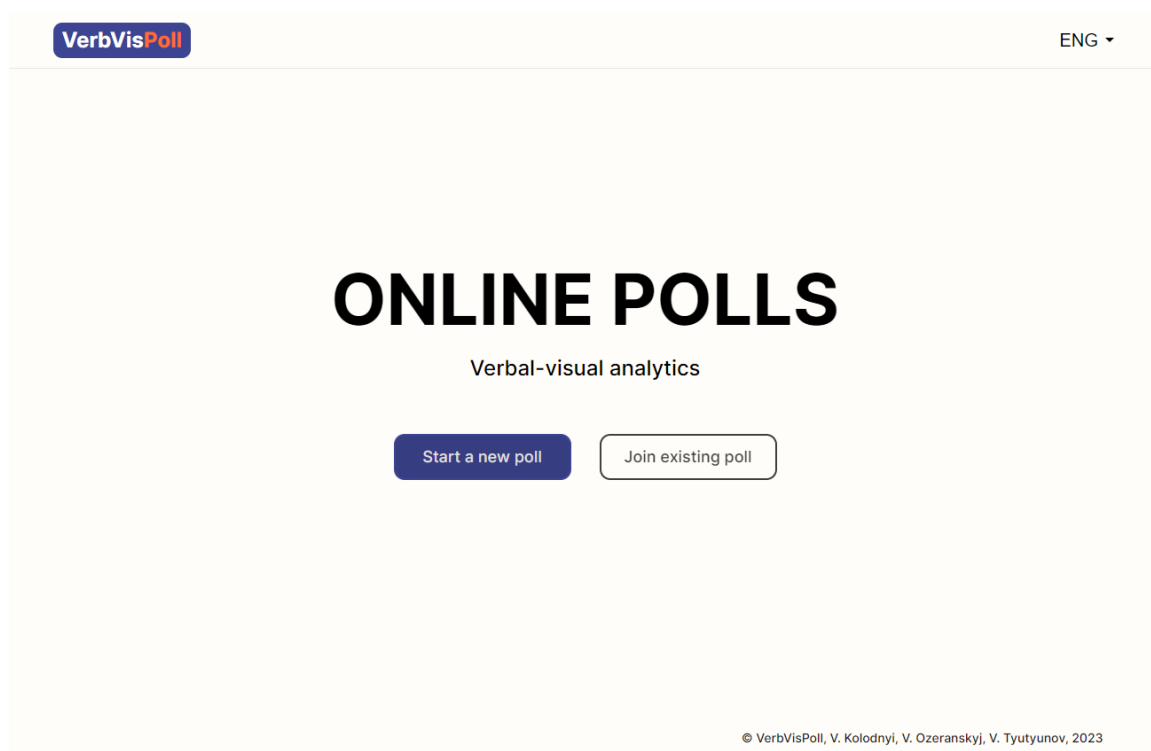

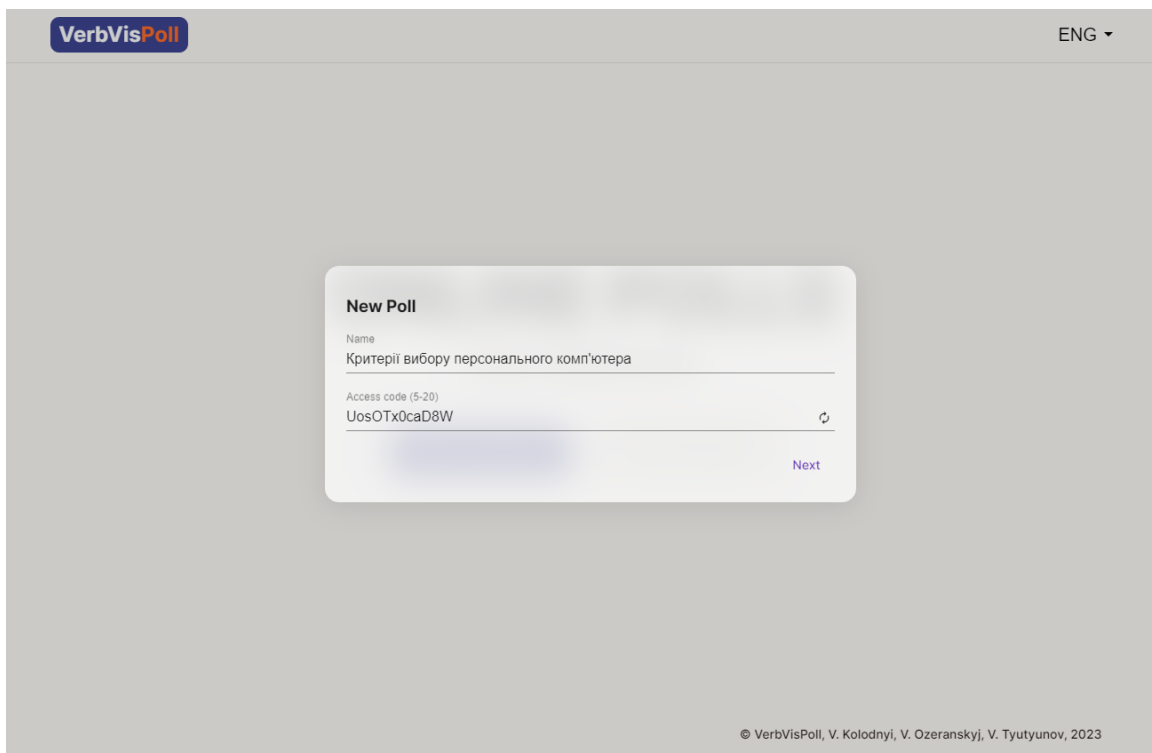


Рисунок Г.1 – Початкова сторінка додатку

### 2. Створення опитування

Після вибору пункту «Почати нове опитування» на головній сторінці, відкриється вікно з налаштуваннями опитування. У полі «Назва» потрібно ввести назву вашого опитування, наприклад «Критерії вибору персонального комп'ютера». У полі «Код доступу» потрібно ввести набір символів, за допомогою якого інші користувачі зможуть приєднатись до вашого опитування.

Також можна натиснути на кнопку  для створення випадкового коду доступу. Після введення даних натисніть «Далі».



The screenshot shows a web interface for creating a new poll. At the top left is the 'VerbVisPoll' logo, and at the top right is a language selector 'ENG'. The main content is a 'New Poll' form with two input fields. The first field is labeled 'Name' and contains the text 'Критерії вибору персонального комп'ютера'. The second field is labeled 'Access code (5-20)' and contains the alphanumeric string 'UosOTx0caD8W'. A refresh icon is positioned to the right of the access code field. A 'Next' button is located at the bottom right of the form. At the bottom right of the page, there is a copyright notice: '© VerbVisPoll, V. Kolodnyi, V. Ozeranskyj, V. Tyutyunov, 2023'.

Рисунок Г.2 – Форма створення нового опитування

Наступною відкриється форма для введення назв альтернатив. У кожне поле потрібно ввести назви альтернатив, наприклад критерії при виборі персонального комп'ютера. Можна додати та видалити альтернативи. Мінімальна кількість альтернатив – 2, максимальна – 10. Після введення натисніть «Далі».

VerbVisPoll ENG ▾

←  
Enter alternatives

Alternative 1  
Зовнішній вигляд

Alternative 2  
Потужність

Alternative 3  
Операційна система

Alternative 4  
Розмір

Alternative 5  
Бренд

Add alternative Remove alternative Next

© VerbVisPoll, V. Kolodnyj, V. Ozeranskyj, V. Tyutyunov, 2023

Рисунок Г.3 – Форма введення назв альтернатив

Далі відкриється форма вибору типу шкали опитування. На вибір представлені 2 загальні шкали та можливість створити власну шкалу.

VerbVisPoll ENG ▾

←  
Rating scale type:

Strongly against	Against	Indifferent (Abstain)	For	Strongly for
1	2	3	4	5
New scale				

Next

© VerbVisPoll, V. Kolodnyj, V. Ozeranskyj, V. Tyutyunov, 2023

Рисунок Г.4 – Форма вибору шкали опитування

VerbVisPoll ENG ▾

←  
New scale

Choose a point-and-color structure:

● -2 -1 0 1 2  
○ 1 2 3 4 5

Enter the names of the gradations

2	Дуже важливо
1	Важливо
0	Нейтрально
-1	Не важливо
-2	Зовсім не важливо

Next

© VerbVisPoll, V. Kolodnyi, V. Ozeranskyj, V. Tyutyunov, 2023

Рисунок Г.5 – Форма створення власної шкали

При створенні власної шкали, вам потрібно обрати кольорову модель та ввести назви градацій.

Після створення, ви попадете на сторінку опитування. Тут вказана назва опитування, код доступу, кількість опитаних та посилання на дане опитування. Натиснувши на це посилання, воно буде скопійоване у ваш буфер обміну – тепер ви можете поширити його для проходження.



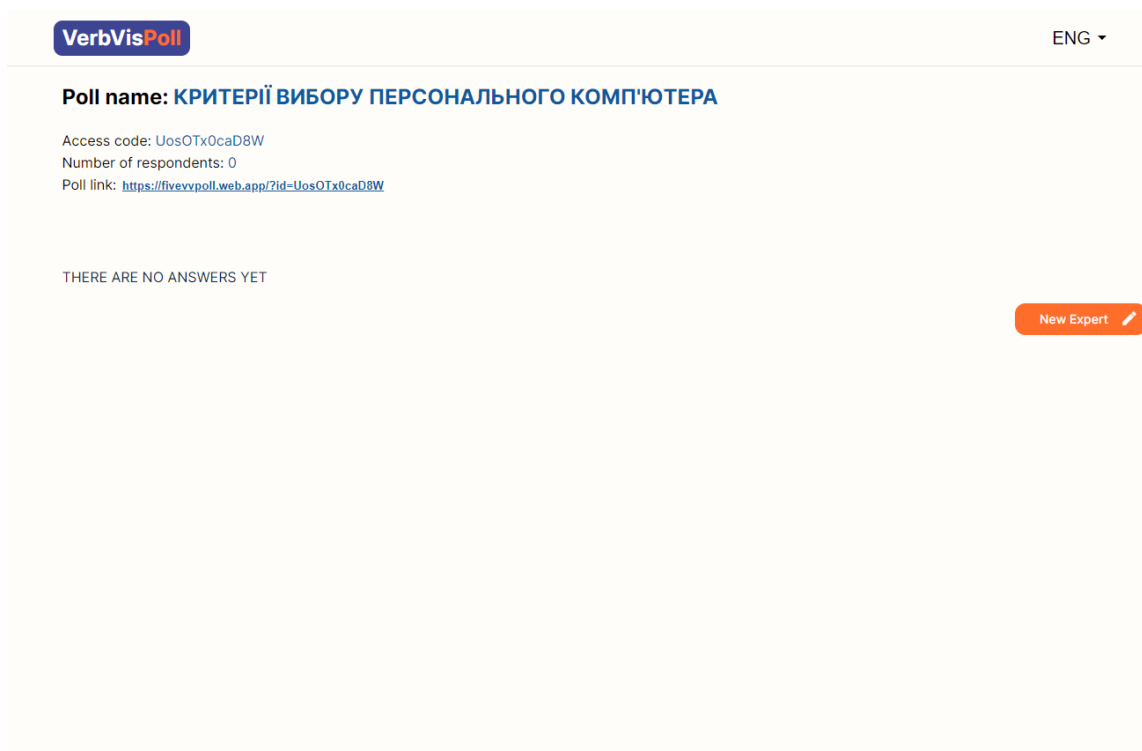


Рисунок Г.6 – Вікно нового опитування

### 3. Приєднання до опитування

Для того щоб приєднатися до опитування потрібно на головній сторінці натиснути на кнопку «Приєднатись до опитування», після чого буде відкрита форма для введення коду доступу відповідного опитування. Код доступу вам може надати користувач, який створив опитування, або користувач, який вже приєднався до опитування. Також вам можуть надати посилання на опитування, перейшовши по якому у вас одразу відкриється відповідне опитування.

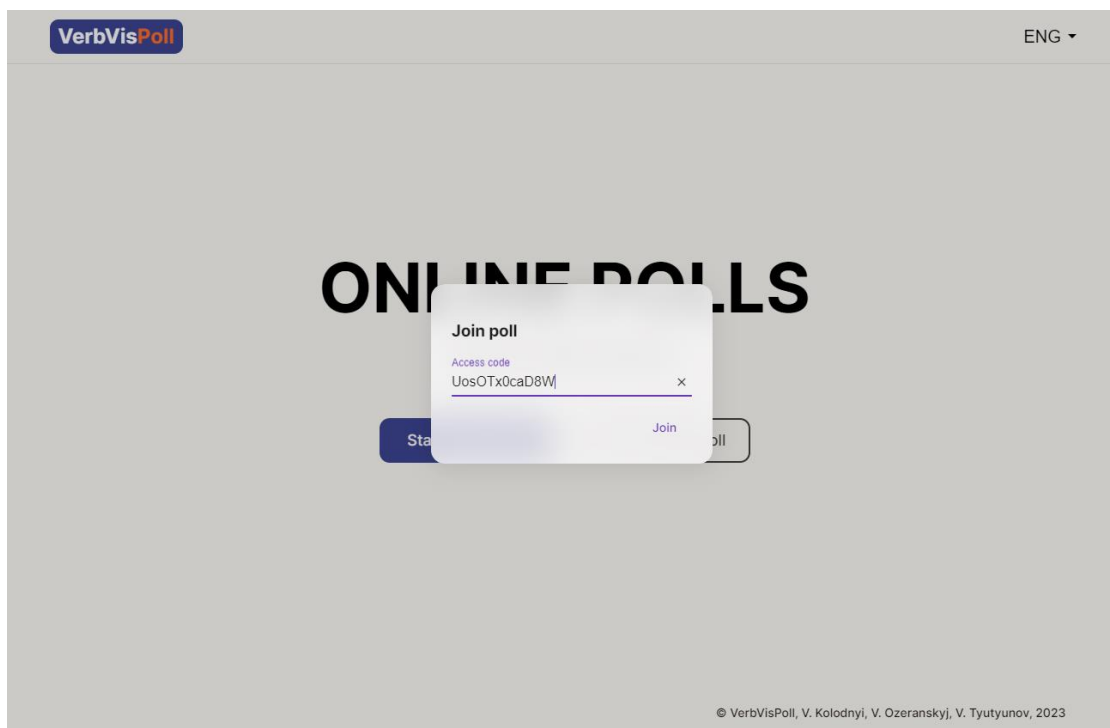


Рисунок Г.7 – Форма приєднання до опитування

Після приєднання, для проходження опитування потрібно натиснути на кнопку «Новий експерт», після чого ввести своє ім'я у відповідне поле.

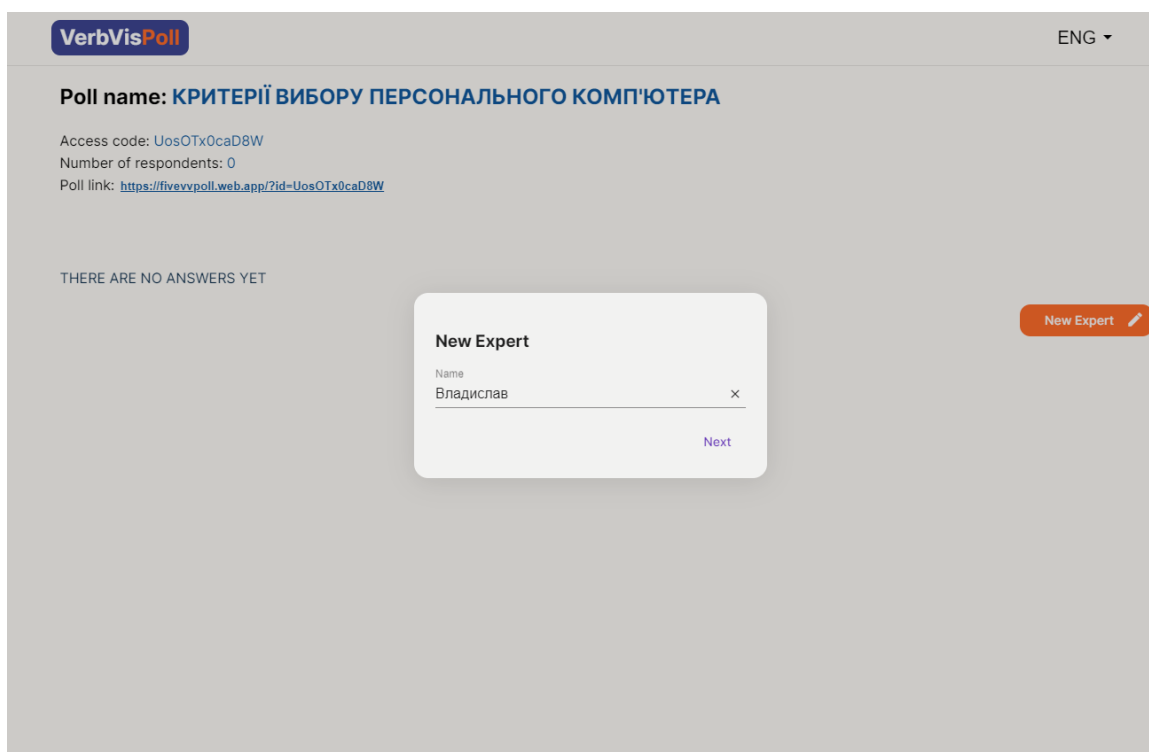


Рисунок Г.8 – Поле для введення ім'я нового експерта

Наступним відкриється вікно з альтернативами та панеллю оцінок для кожної альтернативи. На даній сторінці потрібно оцінити кожну альтернативу, наприклад, як показано на рисунку 9, оцінити важливість кожного критерію у відповідній шкалі. Після оцінювання натисніть «Підтвердити» для збереження результату.

VerbVisPoll ENG ▾

Expert Владислав

Зовнішній вигляд

Зовсім не важливо Не важливо **Нейтрально** Важливо Дуже важливо

Потужність

Зовсім не важливо Не важливо Нейтрально Важливо **Дуже важливо**

Операційна система

Зовсім не важливо Не важливо Нейтрально Важливо **Дуже важливо**

Розмір

Зовсім не важливо **Не важливо** Нейтрально Важливо Дуже важливо

Бренд

**Зовсім не важливо** Не важливо Нейтрально Важливо Дуже важливо

Confirm

© VerbVisPoll, V. Kolodnyi, V. Ozeranskyj, V. Tyutyunov, 2023

Рисунок Г.9 – Сторінка оцінювання альтернатив

Після цього ви знову попадете на сторінку опитування з оновленим результатом. При наявності хоча б одної відповіді на опитування, на сторінці з'явиться графік з результатами опитування (рисунок 10).

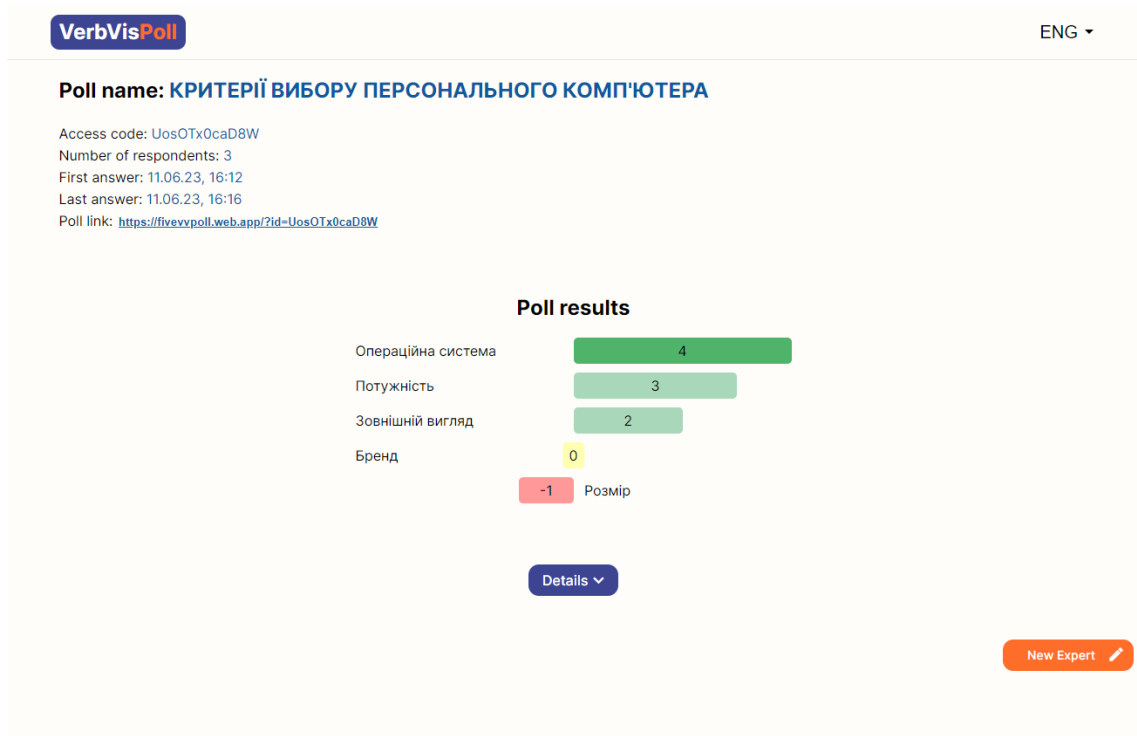


Рисунок Г.10 – Сторінка опитування з графіком результату

Також з'явилась кнопка «Детальніше» після натискання якої з'явиться графік з виставлених оцінок для кожної альтернативи (рисунок 11).



Рисунок Г.11 – Графік оцінювання альтернатив

Також з'явиться кнопка «Мапа переважань», після натискання на яку відкриється Мапа переважань, на якій можна побачити оцінку кожного користування для кожної альтернативи.

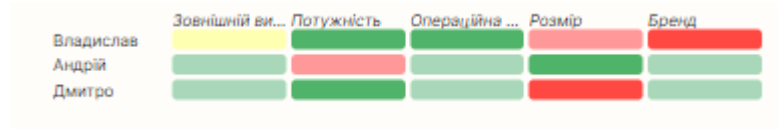


Рисунок Г.12 – Мапа переважань

Щоб побачити оцінку користувача відповідно до шкали оцінювання, наведіть курсор на відповідну клітинку (рис. 13).

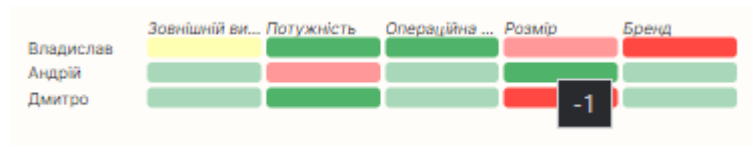


Рисунок Г.13 – Мапа переважань з підказкою на конкретній оцінці

## Додаток Д (довідниковий)

## Довідка впровадження

**«ІТ-СЕРВІС-ВІН»**

Товариство з обмеженою відповідальністю

Вих. № 6/н  
Від «04» грудня 2023 року

21001, м. Вінниця, вул. Замостянська, 15

Довідка дана студенту групи 2КН-22м Тютюнову Владиславу Петровичу про те, що результати його магістерської кваліфікаційної роботи на тему «Інформаційна технологія проведення онлайн-опитувань» використані в програмному забезпеченні для проведення онлайн-опитувань. За результатами роботи програмне забезпечення у вигляді готового продукту планується до впровадження.

Директор ТОВ



Стус В. О.