

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра комп'ютерних наук

Пояснювальна записка

до магістерської кваліфікаційної роботи

на тему «Інформаційна технологія проведення колективних онлайн експертиз»

Виконав: студент 2 курсу,

групи 2КН-20м

спеціальності 122 «Комп'ютерні науки»

Мельник С.М.

Керівник: к. т. н., доц. Колодний В.В.

Рецензент: к. т. н., доц. каф. ПЗ Романюк О.В.

Вінниця

2021

ЗАТВЕРДЖУЮ

Завідувач кафедри КН

д.т.н., проф. Яровий А.А.

(підпис)

« ____ » _____ 2021 року

ЗАВДАННЯ

на магістерську кваліфікаційну роботу на здобуття кваліфікації магістра зі спеціальності 122 – «Системи штучного інтелекту»

08-22.МКР.017.21.471.ПЗ

Магістранта групи 2КН-20м Мельника С.М.

Тема магістерської кваліфікаційної роботи: «Інформаційна технологія проведення колективних онлайн експертиз»

Вихідні дані: кількість альтернатив – 20 шт; кількість експертів – 100 чол; кількість експертних оцінок – 5 шт; кількість користувачів, що можуть користуватись сайтом одночасно – 2000 чол; мова програмування об'єктно орієнтована.

Короткий зміст частин магістерської кваліфікаційної роботи:

1. Графічна: діаграма прецедентів інформаційної технології; діаграма взаємодії клієнтського та серверного модулів; діаграми діяльності модулів інформаційної технології; приклад роботи програми.

2. Текстова (пояснювальна записка): обґрунтування актуальності інформаційної технології; характеристика та аналіз систем аналогів; постановка задачі; розробка алгоритмів функціонування модулів інформаційної технології; розробка загальної структурної схеми інформаційної технології; програмна реалізація інформаційної технології; тестування інформаційної технології.

КАЛЕНДАРНИЙ ПЛАН ВИКОНАННЯ МКР

№ етапу	Назва етапу	Термін виконання		Очікувані результати
		Початок	Кінець	
1	Обґрунтування доцільності розробки інформаційної технології			Розділ 1
2	Проектування інформаційної технології			Розділ 2
3	Реалізація інформаційної технології			Розділ 3
4	Підготовка економічної частини			Розділ 4
5	Апробація результатів дослідження тези доповідей/акт впровадження			тези доповідей/акт впровадження
6	Оформлення пояснювальної записки, графічного матеріалу та презентації			Пояснювальна записка, презентація

Консультанти з окремих розділів магістерської кваліфікаційної роботи

1. Науковий керівник _____

(підпис)

К. Т. Н., доц.

наук. ступінь, вчене звання (посада)

“ _____ ” _____ 20__ р.

Колодний В.В.

ініціали та прізвище

2. Економічна частина _____

(підпис)

канд. екон. наук, проф.

наук. ступінь, вчене звання (посада)

“ _____ ” _____ 20__ р.

Небава М.І.

ініціали та прізвище

Дата попереднього захисту роботи “ _____ ” _____ 20__ р.

Рецензент _____

(підпис)

К. Т. Н., доц. каф. ПЗ

наук. ступінь, вчене звання (посада)

“ _____ ” _____ 20__ р.

Романюк О.В.

ініціали та прізвище

Завдання видав

науковий керівник _____

(підпис)

К. Т. Н., доц.

наук. ступінь, вчене звання (посада)

“ _____ ” _____ 20__ р.

Колодний В.В.

ініціали та прізвище

Завдання отримав магістрант _____

(підпис)

Мельник С.М.

ініціали та прізвище

“ _____ ” _____ 20__ р.

АНОТАЦІЯ

Магістрську кваліфікаційну роботу присвячено розробці інформаційної технології проведення колективних онлайн експертиз. Досліджено предметну область прийняття колективних рішень. Визначено способи прийняття рішень з використанням методів експертного оцінювання альтернатив та оглянуто види відображення результатів. Розроблено структуру та алгоритм роботи інформаційної технології проведення колективних онлайн експертиз. Інформаційну технологію реалізовано програмно в вигляді веб додатку з використанням клієнт-серверної архітектури з використанням мови програмування JavaScript. Розроблена інформаційна технологія дозволяє створити експертизу з власноруч доданими альтернативами і оцінками, та визначити яка альтернатива є кращою згідно з експертними оцінками.

ABSTRACT

The master's qualification work is devoted to the development of information technology for conducting collective online examinations. The subject area of collective decision-making is investigated. Methods of decision-making using methods of expert evaluation of alternatives are identified and the types of reflection of results are examined. The structure and algorithm of information technology of collective online examinations are developed. The information technology is implemented programmatically in the form of a web application using a client-server architecture using the JavaScript programming language. Developed information technology allows you to create expertise with your own added alternatives and assessments, and determine which alternative is the best according to expert assessments.

ЗМІСТ

ВСТУП.....	7
1 ОБГРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ ОНЛАЙН ЕКСПЕРТИЗ	10
1.1 Аналіз предметної області проведення колективних онлайн експертиз.....	10
1.2 Огляд видів відображення результатів проведених онлайн експертиз.....	12
1.3 Аналіз аналогів технології проведення колективних онлайн експертиз	17
1.4 Висновок	20
2 АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ МОДУЛЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ КОЛЕКТИВНИХ ОНЛАЙН ЕКСПЕРТИЗ.....	21
2.1 Математична модель проведення колективних онлайн експертиз.....	21
2.2 Проектування клієнт-серверної взаємодії в інформаційній технології проведення колективних онлайн експертиз	22
2.3 Розробка загальної структурної схеми інформаційної технології проведення колективних онлайн експертиз	25
2.4 Розробка алгоритмів функціонування модулів інформаційної технології проведення колективних онлайн експертиз	28
2.5 Висновок	35
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ КОЛЕКТИВНИХ ОНЛАЙН ЕКСПЕРТИЗ.....	36
3.1 Обґрунтування вибору програмно-апаратної платформи	36
3.2 Обґрунтування вибору мови програмування.....	39
3.3 Реалізація основних модулів інформаційної технології проведення колективних онлайн експертиз.....	50
3.4 Тестування інформаційної технології проведення колективних онлайн експертиз	53
3.5 Висновок	66
4 ЕКОНОМІЧНА ЧАСТИНА.....	67
4.1 Комерційний та технологічний аудит науково-технічної розробки.....	67

4.2 Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи.....	71
4.3 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором.....	76
4.4 Висновок	81
ВИСНОВКИ.....	82
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	83
ДОДАТКИ.....	Ошибка! Закладка не определена.
Додаток А. Інструкція користувача.....	Ошибка! Закладка не определена.
Додаток Б. Лістинг програми	Ошибка! Закладка не определена.
Додаток В. Графічна частина	Ошибка! Закладка не определена.
Додаток Г. Довідка про впровадження	Ошибка! Закладка не определена.

ВСТУП

Актуальність. У сучасному суспільстві, з активно прогресуючими цифровими технологіями, досить актуальним є питання автоматизації процесів, які протягом багатьох років виконувались вручну. Одним із цих процесів є прийняття колективного рішення, що є в міру прийнятним для кожного експерта, та задовольнить мету опитування.

Прийняття колективних експертних рішень – це результат спільної інтелектуальної праці групи людей. Такі рішення приймаються з урахуванням інтересів і позицій всіх членів групи. Безсумнівна перевага колективних рішень в порівнянні з індивідуальними – вища якість і обґрунтованість, через те, що для їх розробки використовується набагато більший обсяг інформації.

В свою чергу використання візуальних елементів, таких як діаграми, графіки та карти є досить важливим. Візуалізація даних забезпечує доступний спосіб побачити та зрозуміти тенденції та закономірності в даних. Тому є актуальним створення інформаційної технології, що поєднає в собі можливість проведення колективної експертизи та надасть якісну та зрозумілу візуалізацію результуючих даних.

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 «Розробка спеціалізованих засобів штучного інтелекту на основі інтелектуального аналізу даних та машинного навчання» та плану наукової та навчально-методичної роботи кафедри.

Мета і завдання дослідження. Метою дослідження є підвищення ефективності процесу проведення колективних онлайн експертиз в умовах віддаленої комунікації між експертами за рахунок розширення функціональних можливостей інформаційної технології.

Для досягнення поставленої мети необхідно виконати такі задачі:

1. Аналіз актуальності інформаційної технології.
2. Аналіз та характеристика систем аналогів.

3. Розробка алгоритмів функціонування усіх програмних модулів.
4. Програмна реалізація інформаційної технології.
5. Тестування інформаційної технології.

Об'єкт дослідження – процес автоматизованого проведення колективної онлайн експертизи та відображення результатів згідно з експертними оцінками.

Предмет дослідження – програмні засоби проведення колективної онлайн експертизи та відображення отриманих експертних оцінок.

Методи дослідження. У роботі використовуються наступні методи наукових досліджень: метод системного аналізу для проведення аналізу структури інформаційної технології, теорія методів експертного оцінювання, методи відображення інформації; методи цифрової обробки інформації; методи взаємодії між клієнт-серверними додатками.

Наукова новизна одержаних результатів полягає в наступному:

– розроблено динамічну архітектуру інформаційної технології, що дозволяє виконувати оцінювання альтернатив експертами та веб орієнтовану інформаційну технологію проведення колективних онлайн експертиз, що забезпечує підвищення мобільності та зручності використання програмного засобу;

– удосконалено підхід проведення колективних онлайн експертиз за рахунок можливості створювати персоналізовану шкалу оцінювання для поточної експертизи, що позитивно вплине на точність результатів.

Практичне значення одержаних результатів полягає в тому, що розроблено мультифункціональну, візуально наповнену інформаційну технологію, що надасть можливість експертам і користувачам технології проводити колективну експертизу в режимі онлайн та аналізувати результати за допомогою візуально представленої інформації.

Достовірність теоретичних положень магістерської кваліфікаційної роботи підтверджується строгістю постановки задач, коректністю використання методів дослідження, експериментальними дослідженнями, тестуванням програмної реалізації інформаційної технології, порівнянням результатів з відомими, та

збіжністю результатів моделювання з результатами, що отримані під час впровадження розроблених програмних засобів.

Особистий внесок здобувача. Усі результати, що наведено у магістерській кваліфікаційній роботі, отримано самотужки. Запропоновано використання методики проведення колективних онлайн експертиз для пришвидшення та покращення прийняття рішень. Створено програмний додаток у вигляді веб орієнтованої інформаційної технології, який реалізує створену методику проведення колективних онлайн експертиз.

Апробація результатів роботи. Результати дослідження було апробовано на конференції «XII Міжнародної науково-практичної конференції «Інтернет-ОсвітаНаука-2020»» (м. Вінниця, Україна, 2020р.) [1].

Результати, одержані в процесі виконання магістерської кваліфікаційної роботи плануються до використання в Нараївській загальноосвітній школі I-III ступенів Краснопільської сільської ради Гайсинського району Вінницької області (Додаток Г).

Публікації. За результатами магістерської кваліфікаційної роботи опубліковано 3 тези доповіді на конференціях «L Науково-технічна конференція підрозділів Вінницького національного технічного університету» [2], «XLIX Науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2020)» [3], «XII Міжнародної науково-практичної конференції «Інтернет-Освіта-Наука-2020»» [1]. Отримано свідоцтво авторського права на твір «Система візуалізації колективної експертизи CollExpert5» [4]. Подано заявку на отримання свідоцтва про реєстрацію авторського права на твір «Інформаційна технологія проведення колективних онлайн експертиз».

1 ОБГРУНТУВАННЯ ДОЦІЛЬНОСТІ РОЗРОБКИ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ ОНЛАЙН ЕКСПЕРТИЗ

1.1 Аналіз предметної області проведення колективних онлайн експертиз

Питання вибору завжди буде актуальним для людини, незважаючи на її вік, стать, національність, освіту і т.д. Життєвий досвід, звичайно, вчить, і з часом приймати рішення стає простіше, але він зовсім не дає ніякої гарантії, що всі рішення без винятку будуть правильними і ефективними. Прийняття рішень – це вміння, для оволодіння яким необхідно докласти чимало зусиль, а також здобути досить велику кількість знань.

Піти ж тут можна різними шляхами: або пізнавати все методом спроб і помилок, витрачаючи на це багато часу і нервів, або пошукати можливість отримати необхідну інформацію в структурованій і компактній формі, тим самим заощадивши свої психічні і тимчасові ресурси. Вибирати стратегію своєї поведінки нам доводиться в більшості життєвих ситуацій, і вибираємо ми завжди, навіть тоді, коли нам здається, що ми цього не робимо. Але навичка прийняття рішень – це вміння свідомо обирати з безлічі варіантів той, який вплине на ситуацію найкращим чином. Говорячи простіше, є певна цільова функція, яка допомагає оцінити «корисність» тієї чи іншої ситуації. Вона може стосуватися не тільки людини, що робить вибір, але і його близьких, друзів, колег або взагалі всього людства. І вміння приймати рішення – це навичка вибору сценарію з максимальним значенням цільової функції. Важливо зауважити, що вибір не завжди правильний об'єктивно – він просто є найкращим на думку конкретної людини [5].

Є ситуації, в яких зробити вибір і прийняти рішення складно. Наприклад, цільова функція може видавати однакові значення, різні варіанти можуть бути рівноцінні, обидва сценарії можуть бути однаково цінні для людини. Інший варіант труднощів з прийняттям рішень виражається в тому, що цільова функція не встановлена. Інакше кажучи, людина просто не знає, чого хоче. Цей випадок набагато складніше, і вимагає більш суворого визначення причин труднощів.

Виходячи зі сказаного, вміння приймати рішення являє собою цілий набір навичок:

- навичка бачення максимальної кількості можливих варіантів рішень;
- навичка розставлення пріоритетів (визначення цільової функції) для кожної конкретної ситуації;
- навичка вибору одного рішення серед безлічі.

Не буде зайвим відзначити, що всі ці функції за людину спочатку (поки вона знаходиться в дитячому віці) виконують дорослі. Але чим старше стає людина, тим більше вибору їй доводиться робити самотійно. І прийнято вважати, що після досягнення повноліття людина вже в змозі приймати будь-які рішення, які визначають її подальше життя.

Людина розвивається далі, і вже вчиться приймати рішення, які стосуються не тільки її, а й інших людей. Тобто вона вчиться приймати рішення за інших, наприклад, коригувати поведінку своїх дітей вдома або керувати співробітниками на роботі. Цей навик теж дуже важливий, але він і більш складний, тому що кількість варіантів вибору розширюється, а цільова функція включає в себе особисті інтереси і інтереси оточуючих [6].

Актуальність цієї теми обумовлена тим, що вміння приймати рішення – це в першу чергу здатність діяти, вирішувати ситуації, вирішувати проблеми і покращувати власне життя. І від того, наскільки людина володіє цією навичкою, можуть залежати як її власні життєві і професійні результати, так і результати інших людей.

Отже, задача що полягає в розробці інформаційної технології проведення колективних онлайн експертиз є актуальною. Веб обієнтована інформаційна технологія забезпечить користувачам швидке та зручне проведення експертизи для великої кількості експертів та альтернатив. А також надасть змогу отримувати результати експертизи в наочному вигляді за допомогою засобів візуалізації.

1.2 Огляд видів відображення результатів проведених онлайн експертиз

Часто люди в колективі мають проблеми комунікації. Трапляються ситуації, що наші думки, які ми намагаємося розповісти іншому співрозмовнику всіх деталях, залишаюся незрозумілими. Справа в тому, що наші слова, різні люди розуміють по різному. У кожного з нас свій особистий, особливий досвід, тому і ставлення до одного і той же може сильно відрізнятись. Навіть одну і ту ж презентацію на конференції слухачі сприймуть по-своєму: запам'ятають різні слайди, придумують різні питання. Виникає запитання, як домогтися бездоганного і однозначного розуміння серед таких різних людей одночасно? Малюнки в блокноті, замальовки на серветці, схеми в інтернеті, що пояснюють відео та інші засоби комунікації дозволяють фіксувати наші думки і доносити інформацію наочно. Проблема комунікації і пошуку істини лежить в основі багатьох звичних нам речей – фотографії, фільмів, сучасної інфографіки або класичного живопису [7].

Візуальні образи сприймаються нами більш однозначно, ніж їх словесний опис. Краще один раз побачити. Більш того, в одну одиницю часу ми можемо побачити набагато більше, ніж можемо почути. Візуальні образи в 60 000 разів сприймаються швидше, ніж слова. Дослідження Массачусетського інституту МІТ від 2014 року показали, що людина може зрозуміти, що зображено на фотографії, подивившись на неї всього 13 мілісекунд.

Згодом стало зрозуміло, що візуалізація інформації може допомогти в рішенні задач і істотно скоротити час пошуку рішення. У XVII столітті, вчені почали візуалізувати метео-дані на картах і вести прості таблиці. Через століття, завдяки Вільяму Плейфер, з'явилися перші наукові способи візуалізації статистичної інформації та економічних даних. Плейфер аргументував, що діаграми передають дані ефективніше, ніж таблиці. Він – творець лінійного графіка, барчарта і кругової діаграми. В наш час, візуалізація допомагає наводити порядок, перетворюючи сирі дані в наочні діаграми і графіки (рис. 1.1).



Рисунок 1.1 – Важливість візуального представлення інформації

Всі способи візуалізації даних – діаграми і графіки – можна згрупувати за питаннями, на які ця візуалізація буде відповідати. Іншими словами, існують різні види порівняння даних. Виділимо найбільші з них на рисунку 1.2.

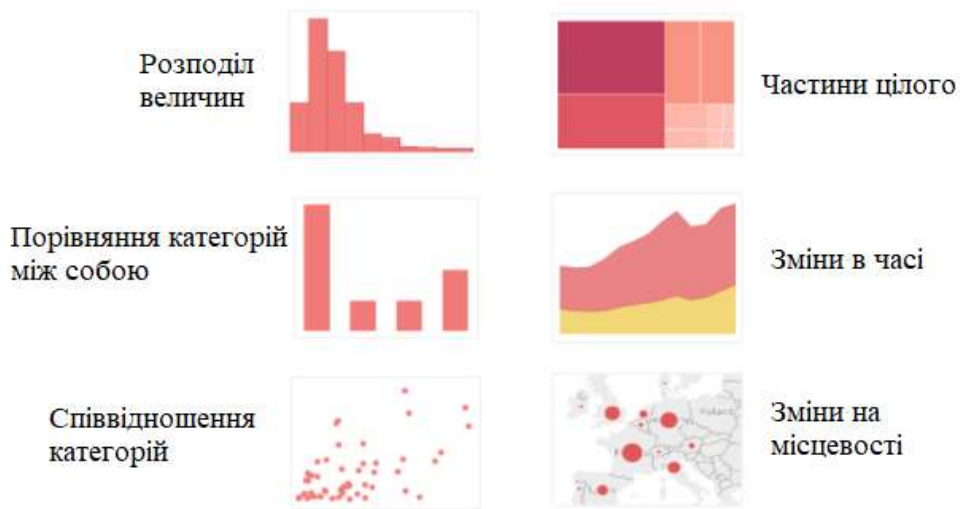


Рисунок 1.2 – Найпопулярніші види порівняння даних

Барчарт або лінійна діаграма забезпечує порівняння категорій між собою. Барчарт або лінійна діаграма – це найпоширеніший і універсальний вид діаграми. Вона відповідає на питання "скільки?". Величина показника виражається за допомогою довжини стовпчиків: чим довше стовпець, тим більше його значення. Стовпчикова діаграма може бути і вертикальною і горизонтальною, в залежності від того, де розташовуються категорії. Діаграма показує значення одного показника по

кожній дискретної категорії, порівнюючи їх один з одним. Якщо потрібно за допомогою лінійної діаграми відобразити тренди в часі, це краще зробити за допомогою лінійного графіка. Барчарти припускають пряме порівняння між окремими періодами. Приклад такої діаграми наведено на рисунку 1.3.

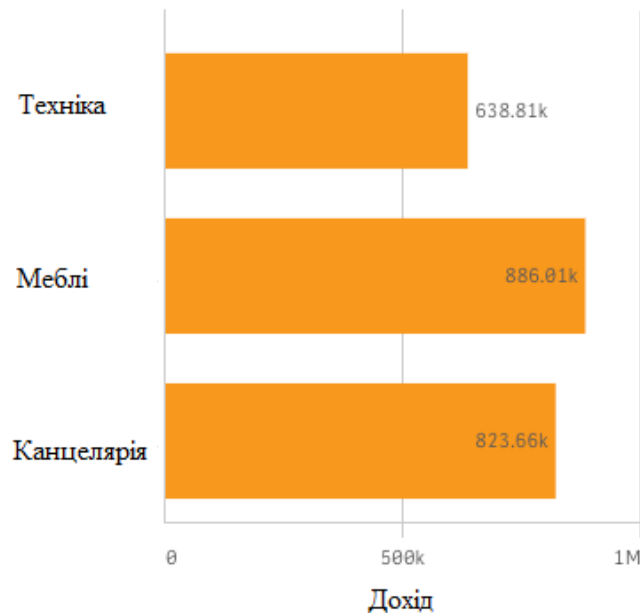


Рисунок 1.3 – Лінійна діаграма

Гістограма – це вид статистичного графіка, який показує розподіл величини. Гістограма показує частоту значень змінної, тобто скільки разів те чи інше значення зустрічається в наборі даних. Візуальним кодуванням тут буде довжина стовпчика – чим він вищий, тим більша частота значення величини. Гістограми використовуються для того, щоб відстежити концентрацію значень, виявити екстремуми, встановити тип статистичного розподілу. При цьому, частотність може показуватися як на інтервалах значень, так і на періоді часу. За змістом, гістограма показує щільність розподілу [8].

Через те, що гістограма показує значення за допомогою стовпців, її часто плутають зі звичайною лінійною діаграмою (барчартом). Незважаючи на зовнішню схожість, ці дві діаграми мають різний зміст і показують різні аспекти даних. У гістограм немає прогалів між стовпцями. Це показує безперервність числового ряду. Гістограму зображено на рисунку 1.4.

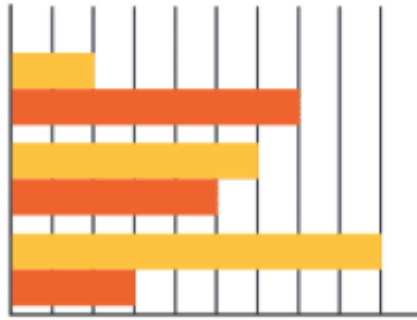


Рисунок 1.4 – Гістограма

Круговою діаграмою називають круг, поділений на сектори. Величина сектора показує, скільки разів та чи інша кількість повторюється в ряді. Приклад кругової діаграми наведено на рисунку 1.5. Кругова діаграма має наступні недоліки:

1. На круговій діаграмі можна відобразити кілька рядів. На графіку або стовбчатій гістограмі можна відобразити будь-яку кількість рядів. Це зручно, так як дозволяє порівнювати ряди між собою. Кругова діаграма дає можливість оцінити тенденцію ряду: тобто найбільш повторювані в ряду елементи.

2. На діаграмі не можна відобразити зміну функції. Для цього можна використовувати тільки графік.

3. На діаграмі не можна наочно зобразити більше 7 елементів. Якщо в ряду 200, 300 і більше елементів, то ефект наочності пропадає. Глядачеві буде просто неможливо розгледіти співвідношення елементів. Тому було виведено максимальне число елементів ряду, після якого ефект наочності кругової діаграми пропадає: це число 7.



Рисунок 1.5 – Кругова діаграма

Діаграма Мекко (яку іноді називають діаграмою Marimekko) – це двомірна діаграма з накопиченням. Крім різної висоти сегментів звичайної діаграми з накопиченням, діаграма Мекко також підтримує різну ширину стовпців. Ширина стовпців масштабується таким чином, що загальна їх ширина дорівнює необхідній ширині діаграми. Щоб зберегти візуальне відношення між шириною різних стовпців, на діаграмі Мекко між стовпцями немає проміжків. Приклад діаграми Мекко наведено на рисунку 1.6.

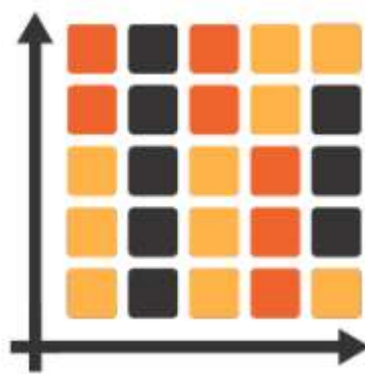


Рисунок 1.6 – Діаграма Мекко

Після перегляду різних методів візуалізації інформації варто обрати такий, що найкраще підійде під поточне завдання. Так як основними функціями інформаційної технології є порівняння отриманих значень та аналіз складових експертизи, то в процесі візуалізації потрібно покладатись на кінцеві набори даних. Тому використаємо наступні методи візуалізації:

- різнобічно орієнтована гістограма, підійде для зіставлення подібності та відмінності між отриманими даними, враховуючи наявність від’ємних значень та великої кількості альтернатив;

- діаграма Мекко, продемонструє всебічний розподіл даних, зробить акцент на нормальних тенденціях та факторах, що можуть їх порушити. Кольорове забарвлення елементів діаграми приверне увагу експертів на негативні й позитивні акценти та підсвідомо вплине на прийняття певного рішення.

1.3 Аналіз аналогів технології проведення колективних онлайн експертиз

Важливим процесом перед проектуванням і розробкою інформаційної системи є швидкий та грамотний аналіз вже існуючих робіт. Слід об'єктивно оцінювати всі недоліки і переваги існуючих систем і зробити грамотні висновки про найкраще застосування розроблюваної технології. Як системи аналоги до технології проведення колективних онлайн експертиз обрано ВМПП та VisPA [9].

1. Візуальний метод попарних порівнянь – ВМПП. ВМПП – це веб орієнтована система, яка дає можливість провести процедуру попарного порівняння альтернатив. Процедура попарного порівняння допомагає експерту визначити уподобання серед об'єктів в процесі порівняння можливих пар з подальшим упорядкуванням об'єктів на основі отриманих результаті порівняння. Для початку роботи з системою потрібно ввести альтернативи. Їх кількість може бути від 4х до 9. Користувач може створити власних перелік альтернатив. В іншому випадку він має змогу скористатись наявними альтернативами, що окреслюють риси особистості. Завершивши введення альтернатив, користувач виконує попарне порівняння альтернатив між собою, пересуваючи повзунок. Цей процес наведено на рисунку 1.7. Результат виконання буде відображено вигляді гістограми, як це показано на рисунку 1.8.

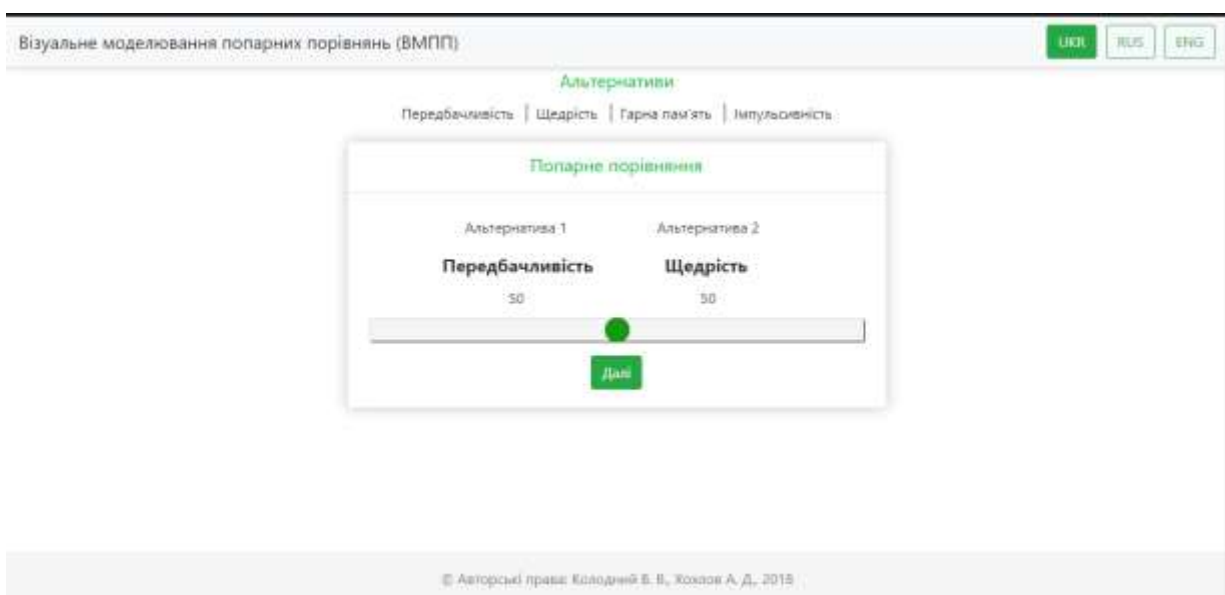


Рисунок 1.7 – Процес попарного порівняння альтернатив

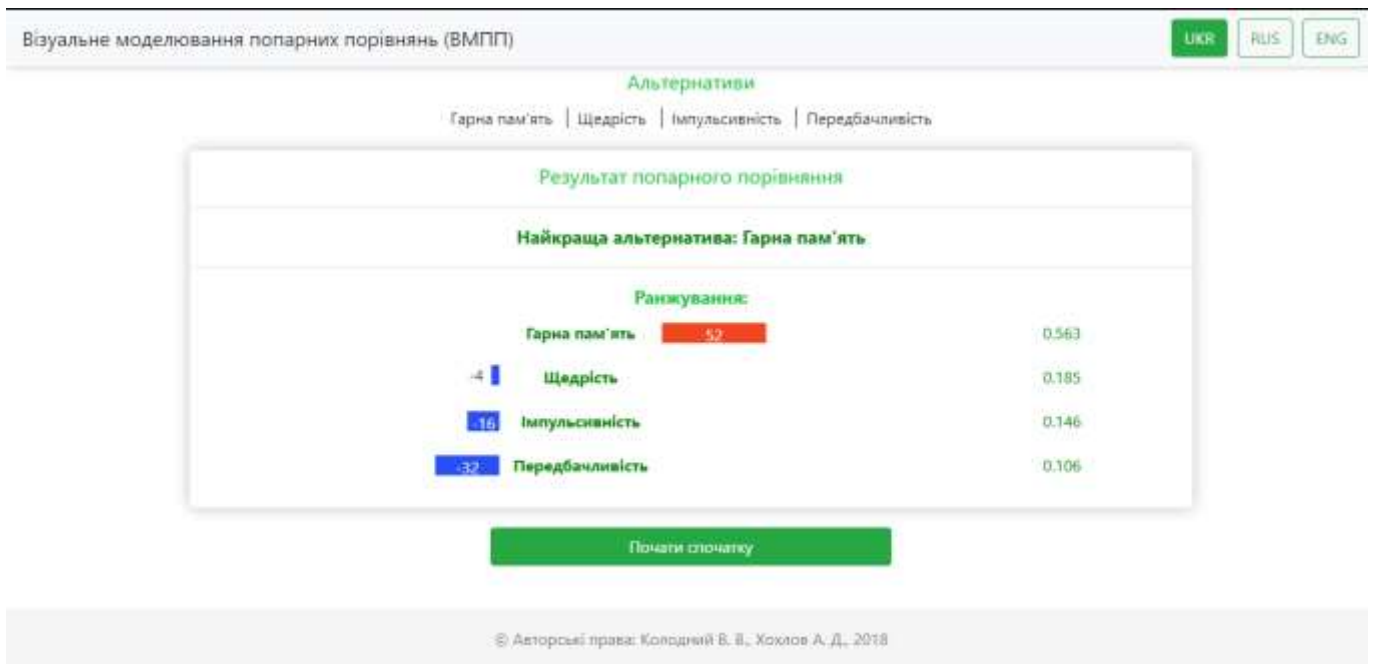


Рисунок 1.8 – Результат попарного порівняння альтернатив

Серед недоліків системи ВМПП є наступні:

- неможливо порівняти більше 9 альтернатив;
- неможливо зберегти результати порівняння;
- неможливо виконувати порівняння альтернатив одночасно декількома експертами.

2. Visual Pairs Of Alternatives (VisPA) – мобільний додаток візуального порівняння альтернатив. Цей додаток виконує некрітеріальне оцінювання альтернатив з використанням гештальт ранжування. Алгоритм роботи наступний: об'єкту прийняття рішень (ОПР) запропоновано створити альтернативи на власну тематику, або використати альтернативи, що визначають риси особистості. Кількість альтернатив обмежена і становить від трьох до дев'яти. Альтернативи порівнюються відбувається попарно, за допомогою проведення повзунка на екрані телефону. Процес показано на рисунку 1.9. Виконавши порівняння всіх альтернатив ОПР одерже результати у вигляді гістограми. Результуючу гістограму наведено на рисунку 1.10. Серед недоліків мобільного додатку VisPA є такі, як відсутня можливість порівняння великої кількості альтернатив. Коли відбувається порівняння максимально можливої кількості альтернатив, цей процес займає велику кількість часу, тому що виконується

повний перебір. Незручність у використанні візуального порівняння альтернатив в тому, що об'єкт прийняття рішень не може прив'язати кількісну оцінку до альтернативи, а лише 'на око' визначає перевагу одного варіанту над іншим. В результаті чого, поява числових оцінок в результуючому ранжуванні є важко зрозумілою.



Рисунок 1.9 – Процес порівняння альтернатив в VisPA

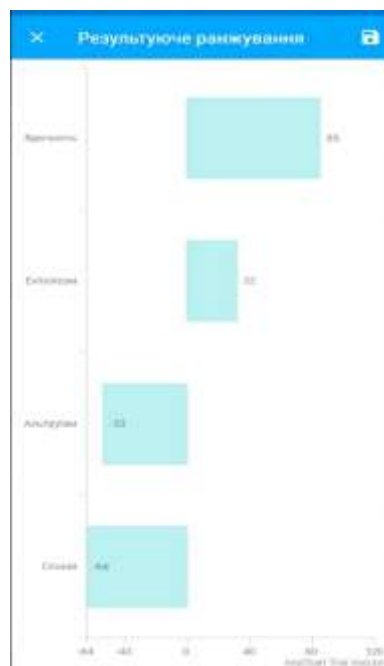


Рисунок 1.10 – Результат порівняння альтернатив в VisPA

Проаналізувавши існуючі аналоги системи, прийнято рішення реалізувати нову інформаційну технологію. Дана технологія буде мультифункціональною та об'єднає в собі такі можливості:

- проведення індивідуальної та колективної експертизи;
- використання програми без попереднього встановлення;
- використання програми з смартфона;
- можливість введення великої кількості альтернатив;
- можливість змінювати кількісь оцінок та задавати їхню вагу;
- можливість збереження результатів експертизи;
- можливість видалення експертиз;
- одночасне відображення результатів порівняння всіх альтернатив.

1.4 Висновок

В результаті виконання поточного розділу виконано наступне:

- проведено аналіз актуальності створення інформаційної технології проведення колективних онлайн експертиз;
- описано, чому вміння прийняття рішення є важливим;
- наведено взаємозв'язок між особистими навичками експерта і процесом прийняття рішень;
- відображено, чому візуальне сприйняття інформації є ефективним і важливим;
- описано різні види візуалізації інформації;
- визначено, які види візуалізації інформації будуть використовуватись в інформаційній технології, що розробляється;
- проаналізовано аналоги технології, що розробляється;
- визначено основні можливості аналогів, їх переваги і недоліки
- описано функціональні можливості, що включатиме в себе інформаційна технологія, що розроблюється.

2 АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ДЛЯ РЕАЛІЗАЦІЇ МОДУЛЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ КОЛЕКТИВНИХ ОНЛАЙН ЕКСПЕРТИЗ

2.1 Математична модель проведення колективних онлайн експертиз

В основі інформаційної технології лежить метод безпосередньої оцінки. Він полягає в тому, що діапазон зміни будь-якої кількісної змінної поділяється на інтервали, кожному з яких надається певна вага, наприклад від 1 до 5. Шкала може містити як позитивні так і негативні оцінки, наприклад -5 до +5. Кожний об'єкт включається в певний інтервал в залежності від його значення. Число інтервалів може бути різним у різних експертів.

В деяких випадках зручніше спочатку вибрати найбільш переважаючий фактор, а потім ранжування.

Сумарні оцінки можуть піддаватись нормалізації. Це дозволяє встановити зв'язок між оцінками які експерти надали об'єктам. Для цього кожна оцінка ділиться на суму оцінок по всім об'єктам.

При проведенні колективної експертизи потрібно отримати усереднену оцінку для кожного фактора. Для цього розраховується середнє арифметичне (M):

$$M = \frac{\sum V}{N},$$

де V – індивідуальна думка кожного експерта;

\sum - знак суми;

n – кількість експертів.

Задача полягає в тому щоб обрати таке рішення з множини A можливих варіантів, що максимально задовольняє переваги експертів.

Систему переваг експертів можна подати у вигляді:

$$P = \langle A, Q, K, F^Q, N \rangle,$$

де Q – множина критеріїв оцінювання якості рішень;

K – множина шкал;

F^Q – відображення множини альтернатив A в множину Q .

N – множина експертів.

Множина A – це сукупність альтернатив, що відповідні можливим способам досягнення мети і не порушують певних обмежень.

Множина критерії Q відображає аспекти мети роботи системи які є суттєвими для експертів.

Множина шкал K відповідає кожному критерію множини шкали Q , в якому вимірюються значення факторів.

Відображення F^Q відповідає кожній альтернативі множини A в просторі критеріїв.

Система переваг P – це елемент який певною мірою зумовлює слабку структурованість задач. Апроксимація системи переваг експертів в більшості випадків дозволяє за певних припущень спростити умови для побудови вирішального правила R або описати механізм вибору альтернативи з множини A .

2.2 Проектування клієнт-серверної взаємодії в інформаційній технології проведення колективних онлайн експертиз

«Клієнт–сервер» (англ. Client-server) – архітектура, в якій мережеві навантаження розподілені між постачальниками послуг, що називаються серверами, і замовниками послуг, що називаються клієнтами.

Фактично клієнт і сервер – це програмне забезпечення. Зазвичай ці програми розташовані на різних обчислювальних машинах і взаємодіють між собою через обчислювальну мережу за допомогою мережевих протоколів, але вони можуть бути розташовані також і на одній машині.

Програми–сервери очікують від клієнтських програм запити і надають їм свої ресурси у вигляді даних (наприклад, завантаження файлів за допомогою HTTP, FTP, BitTorrent, потокове мультимедіа або робота з базами даних) або у вигляді сервісних функцій (наприклад, робота з електронною поштою, спілкування за допомогою систем миттєвого обміну повідомленнями або перегляд web-сторінок у всесвітній павутині) [10].

Оскільки одна програма–сервер може виконувати запити від безлічі програм-клієнтів, її розміщують на спеціально виділеній обчислювальній машині, налаштованій особливим чином, як правило, спільно з іншими програмами-серверами, тому продуктивність цієї машини повинна бути високою. Через особливу роль такої машини в мережі, специфіки її обладнання та програмного забезпечення, її також називають сервером, а машини, які виконують клієнтські програми, відповідно, клієнтами. Загальна клієнт-серверна архітектура зображена на рисунку 2.1.

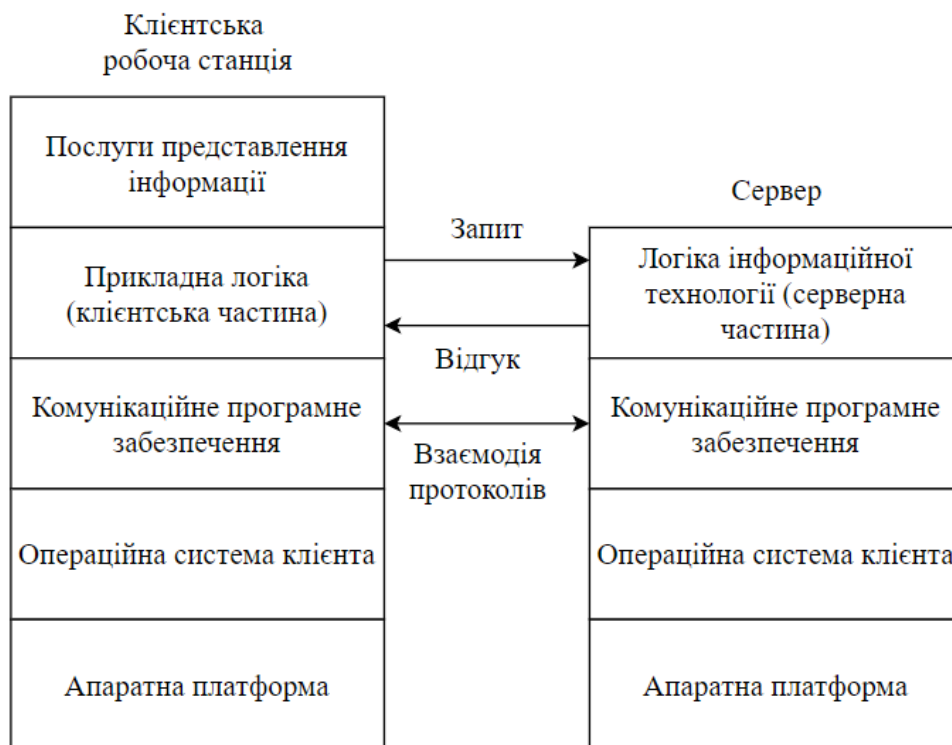


Рисунок 2.1 – Загальна схема клієнт-серверної архітектури

Взаємодія клієнта і сервера забезпечується комунікаційним програмним забезпеченням. Прикладами такого програмного забезпечення є набір протоколів TCP / IP, - протоколи OSI, а також різні фірмові архітектури, на зразок SNA.

Зрозуміло, призначення всього цього програмного забезпечення (протоколів і операційної системи) полягає в наданні бази для розподілених додатків. В ідеальному випадку функція, що виконується додатком, повинна бути розподілена між клієнтом і сервером таким чином, щоб обчислювальні та мережеві ресурси використовувалися оптимально, а користувачі отримали оптимальні можливості для виконання різних

завдань і спільної роботи. В деяких випадках для цього може бути необхідно, щоб більша частина програмного забезпечення виконувалася на сервері, тоді як в інших випадках велика частина логіки може бути реалізована на клієнті.

Нарешті, істотним фактором успіху є метод взаємодії користувача з системою, тобто велике значення має користувальницький інтерфейс клієнтської машини. У більшості систем клієнт-сервер графічному інтерфейсу користувача (Graphical User Interface, GUI) приділяється дуже серйозна увага – він повинен бути простим і зручним, але більш ніж достатньо потужним і гнучким. Таким чином, модуль послуг представлення на робочій станції можна вважати відповідальним за дружній інтерфейс з розподіленими додатками. Модуль послуг представлення не слід плутати з рівнем представлення еталонної моделі OSI. Рівень представлення займається форматуванням даних для їх коректної інтерпретації кожним з двох взаємодіючих комп'ютерів. Модуль послуг представлення має справу з взаємодією користувача і додатка, а також з функціональністю графічного інтерфейсу користувача [11].

Перевагами такої системи є наступні:

- Відсутність дублювання коду програми-сервера програмами-клієнтами.
- Так як всі обчислення виконуються на сервері, то вимоги до комп'ютерів, на яких встановлено клієнт, знижуються.
- Всі дані зберігаються на захищеному сервері, що, як правило, захищений якісніше за більшість клієнтів. На сервері простіше організувати контроль повноважень, з метою делегування доступу до даних тільки клієнтам з відповідними повноваженнями та правами доступу.

Недоліки системи з клієнт-серверною архітектурою:

- Непрацездатність сервера може спричинити непрацездатність всієї обчислювальної мережі мережу. Непрацездатний сервер, це сервер, продуктивності якого не вистачає на обслуговування всіх клієнтів, а також сервер, що знаходиться на ремонті, профілактиці і т. д.
- Підтримка роботи системи вимагає окремого фахівця - системного адміністратора.
- Висока вартість обладнання.

Отже, саме клієнт-серверна модель використовуватиметься під час проектування та розробки інформаційної технології проведення колективних онлайн експертиз. Серверна та клієнтська частини розроблятимуться одночасно, та їх модулі будуть взаємодіяти між собою. В серверній частині будуть представлені наступні модулі: модуль реєстрації, модуль створення експертизи, модуль проведення експертизи, модуль обрахування результатів колективних переважань експертів. В клієнтській частині будуть представлені наступні модулі: модуль реєстрації, модуль створення експертизи, модуль проведення експертизи, модуль відображення результатів проведення колективних онлайн експертиз.

2.3 Розробка загальної структурної схеми інформаційної технології проведення колективних онлайн експертиз

Створення діаграми прецедентів (Use Case Diagram, діаграма використання) є невідемним процесом проектування інформаційних технологій. В єдиній мові моделювання (UML) діаграма використання може підсумовувати деталі користувачів вашої системи (також відомих як актори) та їх взаємодії з системою. Для створення даної діаграми, буде використовуватися набір спеціалізованих символів та роз'ємів. Діаграма ефективного використання допомагає обговорити та представити:

- Сценарії, в яких система або програма взаємодіє з людьми, організаціями чи зовнішніми системами.
- Цілі, яких система чи додаток допомагає досягти цим суб'єктам (відомим як актори).
- Обсяг системи.

Основними елементами діаграми прецедентів є актори, система та цілі.

Актори: Користувачі, які взаємодіють із системою. Актором може бути особа, організація або зовнішня система, яка взаємодіє з вашим додатком або системою. Вони повинні бути зовнішніми об'єктами, які виробляють або споживають дані [12].

Система: конкретна послідовність дій та взаємодії між акторами та системою. Система також може називатися сценарієм.

Цілі: Кінцевий результат більшості випадків використання. Успішна діаграма повинна описувати діяльність та варіанти, які використовуються для досягнення мети.

Діаграма прецедентів не є деталізованою – наприклад вона не моделює порядок виконання кроків. Натомість, діаграма правильного використання зображує високооглядний огляд взаємозв'язку між варіантами використання, суб'єктами та системами. Експерти рекомендують використовувати діаграми варіантів використання для доповнення більш описового текстового випадку використання.

UML – це інструментарій моделювання, який можна використовувати для побудови своїх діаграм. Випадки використання зображені з позначеною овальною формою. Фігурки представляють акторів процесу, а участь актора в системі моделюється лінією між актором та варіантом використання. Щоб зобразити межу системи, намалюйте рамку навколо самого варіанта використання.

Діаграми варіантів використання UML ідеально підходять для:

- Представлення цілей взаємодії системи та користувача.
- Визначення та організація функціональних вимог у системі.
- Визначення контексту та вимог до системи.
- Моделювання основного потоку подій у випадку використання.

Діаграму прецедентів для інформаційної технології проведення колективних онлайн експертиз наведено на рисунку 2.2.

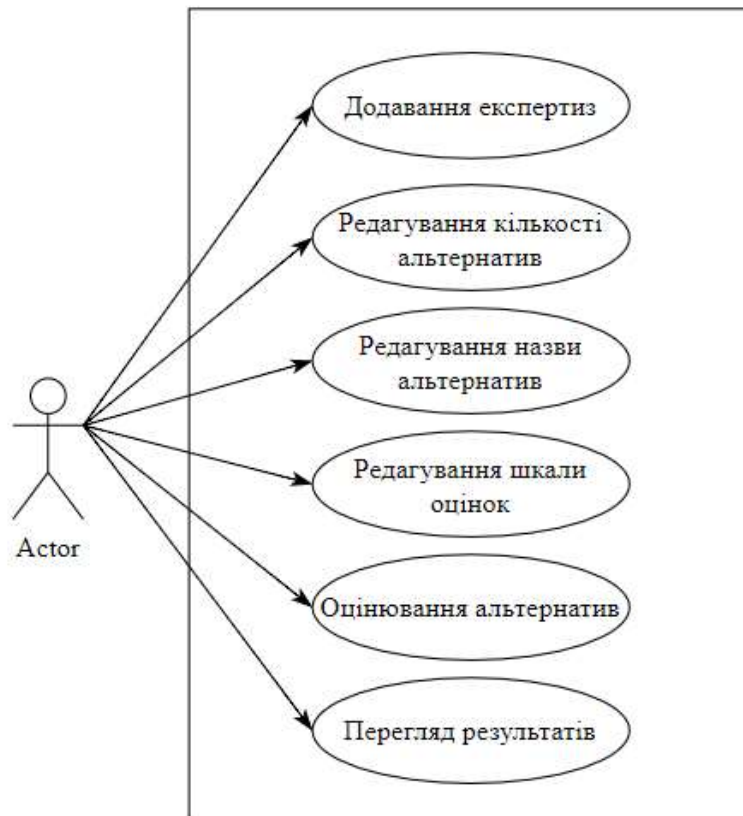


Рисунок 2.2 – UML-діаграма прецедентів

Так як інформаційна технологія проведення колективних онлайн експертиз розробляється у вигляді клієнт-серверної архітектури та містить два паралельних процеси розробки, то обидві частини, як клієнтська так і серверна взаємозалежні один від одного. Тому слід навести схему взаємодії модулів обох частин системи. Загальна структурна схема взаємодії клієнтських модулів та серверної частини зображена на рисунку 2.3. Загальна структурна схема взаємодії серверних модулів та клієнтської частини зображена на рисунку 2.4.

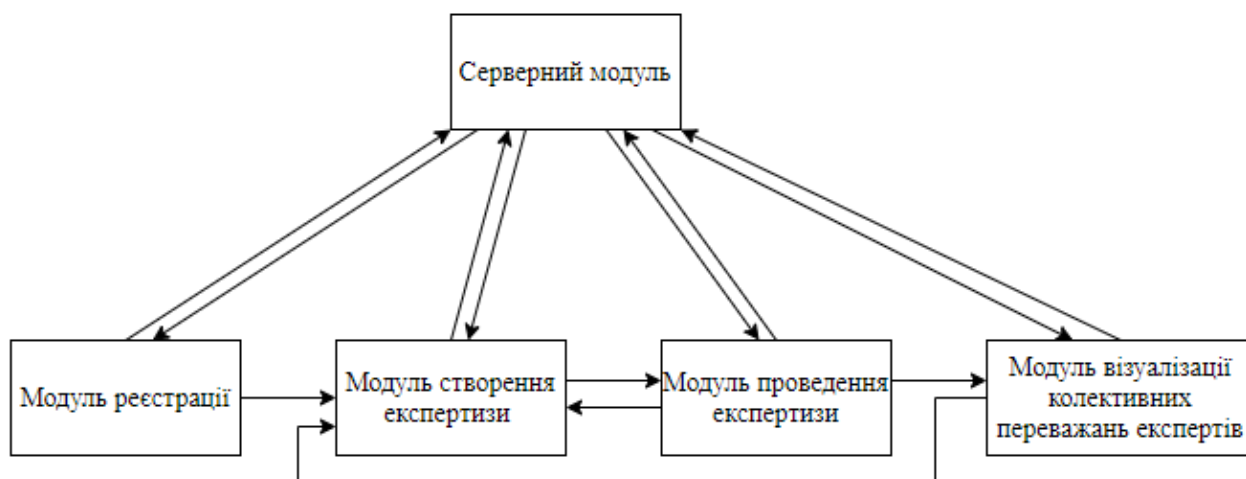


Рисунок 2.3 – Структурна схема взаємодії клієнтських модулів та серверної частини



Рисунок 2.4 – Структурна схема взаємодії серверних модулів та клієнтської частини

2.4 Розробка алгоритмів функціонування модулів інформаційної технології проведення колективних онлайн експертиз

Алгоритми – це ряд інструкцій, які крок за кроком виконуються, щоб зробити щось корисне або вирішити проблему. В обчислювальній техніці алгоритми надають комп'ютерам послідовний посібник щодо виконання дій. Вони складаються з точного переліку інструкцій, які точно визначають, як виконати завдання. Комп'ютерні

алгоритми працюють через вхід і вихід. Вони беруть вхідні дані і застосовують кожен крок алгоритму до цієї інформації для створення результату [13].

Наприклад, пошукова система – це алгоритм, який приймає пошуковий запит як вхідний і шукає у своїй базі даних елементи, які мають відношення до слів у запиті. Потім він виводить результати.

Алгоритми можна проасоціювати з блок-схемами. Введення вказує на кроки та запитання, які потребують послідовного розгляду. Після завершення кожного розділу блок-схеми буде згенерований результат. Алгоритми використовуються у всіх сферах ІТ та обчислень. Вони можуть маніпулювати та обробляти дані та виконувати обчислення чи дії різними способами. Чудовий приклад діючих алгоритмів – це програмне забезпечення для автоматизації. Це тому, що автоматизація працює, дотримуючись встановлених правил для виконання завдань. Ці правила формують алгоритм.

Отже, програмне забезпечення для автоматизації складається з безлічі алгоритмів, які працюють для автоматизації процесів. Алгоритм інформаційної технології проведення колективних онлайн експертиз спрямований на швидкий процес отримання, обробки та повернення даних. А клієнтська частина даної технології забезпечить користувачеві зручний та зрозумілий процес проведення онлайн експертизи. Для покращення ефективності в процесі розробки алгоритмів, інформаційну технологію розділено на модулі, які взаємодітимуть між собою:

- модуль реєстрації;
- модуль створення експертизи;
- модуль проведення експертизи;
- модуль обрахування результатів;
- модуль відображення результатів експертизи.

Діаграму діяльності модуля реєстрації інформаційної технології наведено на рисунку 2.5.

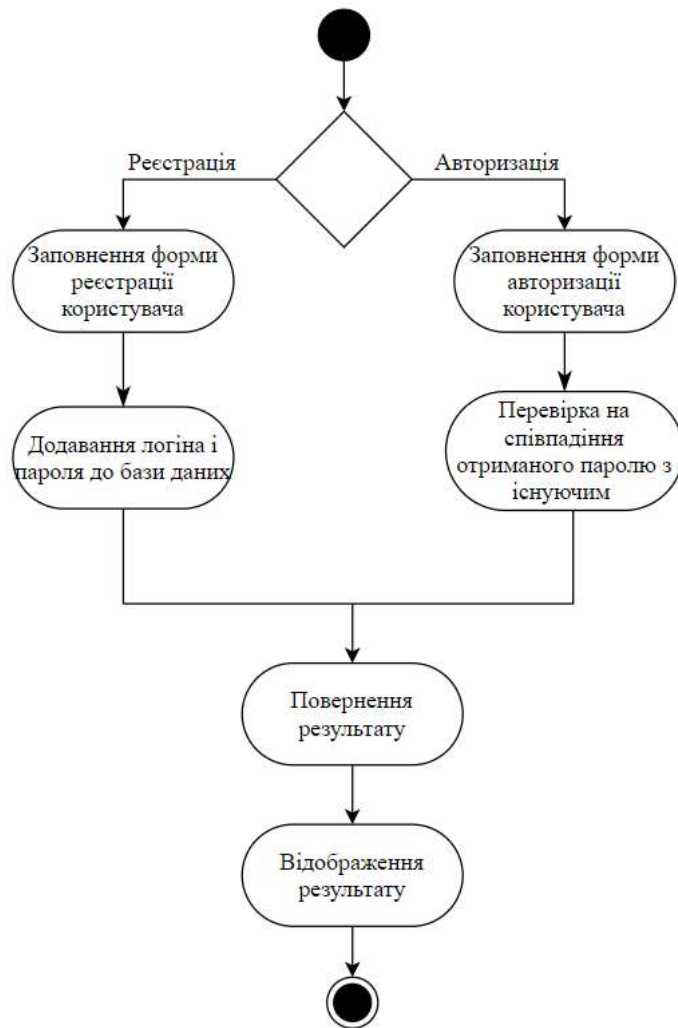


Рисунок 2.5 – Діаграма діяльності модуля реєстрації

Для початку роботи з інформаційною технологією обов'язковою умовою для користувача є реєстрація або авторизація. Відповідно спершу користувач обирає реєстрацію. Далі користувачеві потрібно ввести реєстраційні дані, а саме логін та пароль. Після підтвердження реєстраційних даних ця інформація надсилається для обробки на сервер. З боку сервера, реєстраційні дані додаються в базу даних для подальшого використання. Користувач в свою чергу отримує повідомлення про успішну реєстрацію. Для користування системою, вже зареєстрований користувач повинен авторизуватись. Користувач заповнює реєстраційну форму і дані надсилаються на сервер для обробки. Сервер виконує перевірку на наявність такого користувача в базі даних і якщо авторизаційні дані введено вірно, користувач входить

в систему. Якщо ж інформація про користувача в базі даних відсутня, то пропонується ввести інші реєстраційні дані.

Наступним етапом після реєстрації є перехід до сторінки з вибором подальших дій користувача. Йому пропонується створення власної експертизи або перегляд вже наявних. Діаграму діяльності модуля створення експертизи наведено на рисунку 2.6.

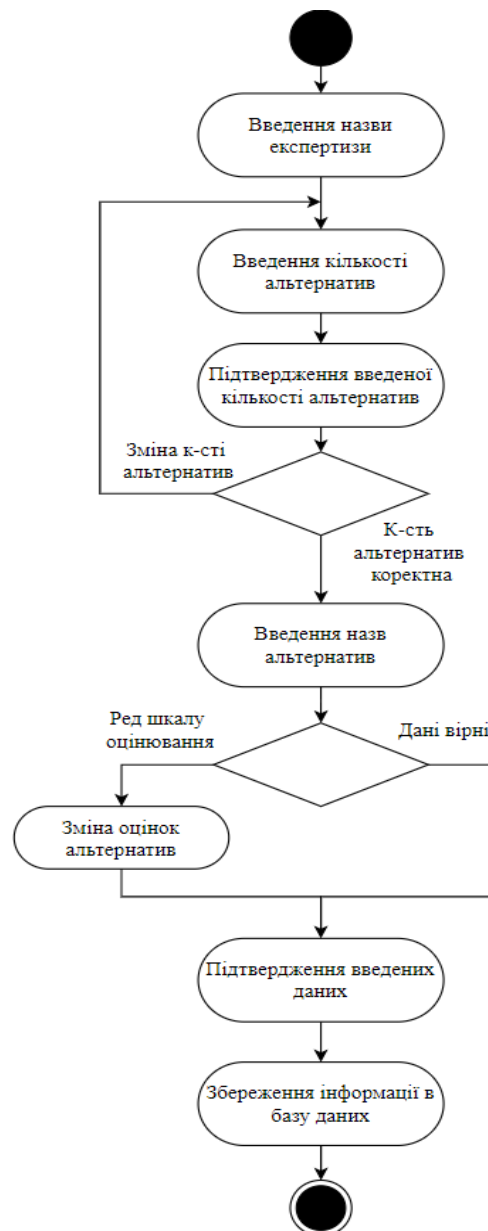


Рисунок 2.6 – Діаграма діяльності модуля створення експертизи

Розглянемо компоненти діаграми детальніше. За допомогою першого компоненту користувачеві вводиться назва експертизи. Далі вибирається кількість альтернатив. Їх кількість може коливатися від 2-х до 20. Введені дані

підтверджуються. Якщо потрібно змінити кількість альтернатив варто увести цю інформацію ще раз та підтвердити. Підтвердивши кількість альтернатив, яка задовольняє вимоги користувача переходимо до етапу заповнення назв альтернатив. Заповнивши назви альтернатив користувач має можливість відредагувати шкалу оцінювання та змінити як вербальні так і числові значення. Після цього дані потрібно підтвердити та відправити на сервер. На сервері введена інформація зберігається в базу даних. Діаграму діяльності модуля проведення експертизи наведено на рисунку 2.7.

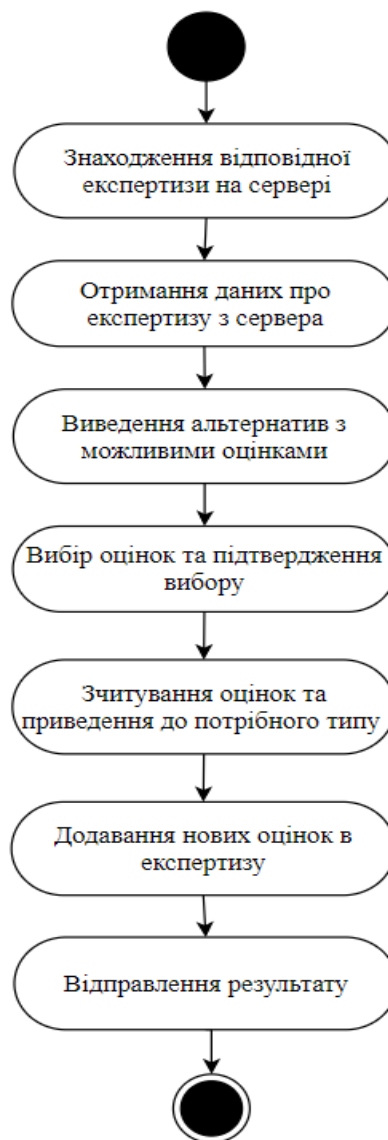


Рисунок 2.7 – Діаграма діяльності модуля проведення експертизи

Опишемо більш детально компоненти діаграми. При переході на конкретну експертизу отримаємо дані про неї з сервера. А саме – назву експертизи, кількість

альтернатив, назви альтернатив та їх оцінки. Продемонструємо користувачеві дані про альтернативи та експертні оцінки. Експерт оцінює кожну з альтернатив відповідно і підтверджує вибір. Введені дані відправляються на сервер для подальшої обробки. Діаграму діяльності модуля обрахування результатів колективних переважань експертів наведено на рисунку 2.8.



Рисунок 2.8 – Діаграма діяльності модуля обрахування результатів експертизи

Розглянемо кожен з компонентів модуля. Перший компонент одержує ідентифікатор експертизи та надсилає його для пошуку інформації про нього в базі даних. Наступний компонент знаходить потрібні дані про експертизу в базі даних:

альтернативи, експерти та їх оцінки. Третій компонент виконує підрахунок експертних оцінок по кожній з альтернатив. Далі дані про оцінки обробляються та аналізуються. Згідно з результатами обрахунку виконується ранжування альтернатив. В результаті виконання ранжування, визначається найкраща альтернатива і порядок відображення інших альтернатив. Оброблені дані надходять в наступний модуль для відображення результатів. Цей модуль зображено на рисунку 2.9.



Рисунок 2.9 – Діаграма діяльності модуля відображення результатів експертизи

Перший компонент модуля відображення результатів експертизи отримує обраховані результати експертизи з сервера для відображення. Першою відображається альтернатива, яка є найкращою. Наступний етап демонструє

гістограму результуючого ранжування альтернатив з їх усередненими числовими значеннями. Далі виводяться гістограми, для кожної альтернативи в експертизі. На гістограмі відображено кількість експертів, що оцінювали альтернативи. Поставлені оцінки підкреслено відповідним кольором. Останнім етапом відображення є матриця переважань на основі альтернатив та експертів. Дана матриця наповнена кольоровою інтерпретацією експертних оцінок.

2.5 Висновок

В результаті виконання другого розділу виконано проектування клієнт-серверної архітектури інформаційної технології проведення онлайн експертиз. Окреслено важливість даної архітектури, наведено загальну схему, визначено її переваги і недоліки.

Розроблено діаграму діяльності в якій визначено загальні можливості актора під час користування програмою. Визначено основні модулі що використовуватимуться в інформаційній технології. Продемонстровано взаємодію модулів клієнтської частини з серверною частиною, а також взаємодію модулів серверної частини з клієнтською.

Розроблено діаграми діяльності кожного з модулів системи. А саме модуль реєстрації, модуль створення експертизи, модуль проведення експертизи, модуль обрахування результатів, модуль відображення результату експертизи.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ПРОВЕДЕННЯ КОЛЕКТИВНИХ ОНЛАЙН ЕКСПЕРТИЗ

3.1 Обґрунтування вибору програмно-апаратної платформи

Програмне забезпечення – це код або набір інструкцій, які вказують комп'ютеру чи апаратному забезпеченню, як працювати. Програмне забезпечення, як правило, є загальним, але його також можна створити на замовлення. Універсальне програмне забезпечення відкрите для ринку, і його специфікації розроблені програмістом. В основному ПЗ розраховано на широкий ринок клієнтів. Індивідуальне програмне забезпечення — це програмне забезпечення, специфікації якого розроблені відповідно до конкретної фірми чи організації. Він відкритий не для всіх. В основному призначений для конкретних бізнес-цілей. Програмне забезпечення в основному класифікується на сім категорій – системне програмне забезпечення, прикладне програмне забезпечення, інженерне/наукове програмне забезпечення, вбудоване програмне забезпечення, програмне забезпечення лінійки продуктів, веб-додатки та програмне забезпечення для штучного інтелекту [14]. Класифікації програмного забезпечення наведено на рисунку 3.1.

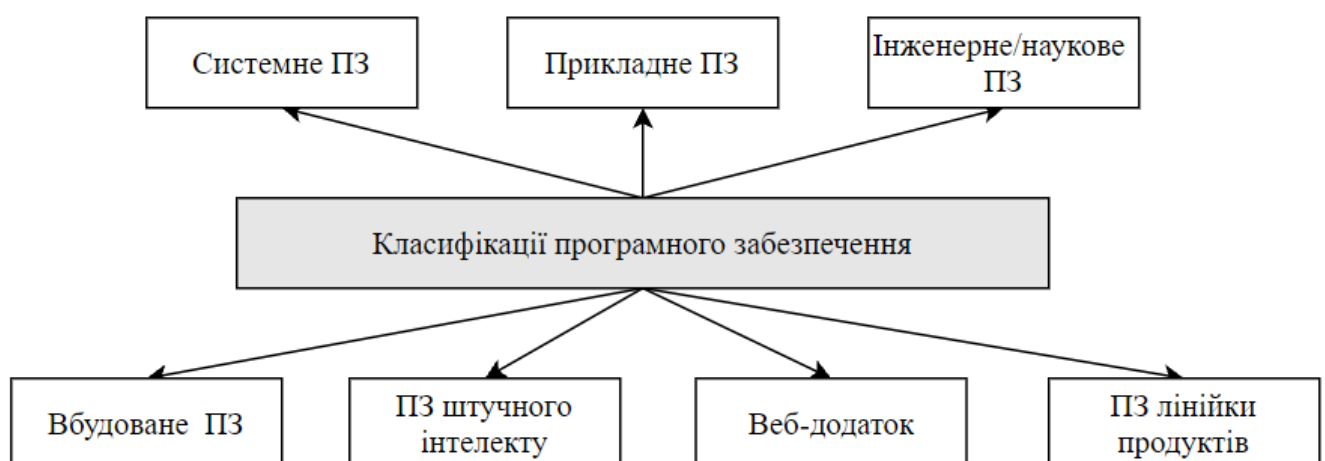


Рисунок 3.1 – Класифікації програмного забезпечення

Розглянемо детальніше різні класифікації програмного забезпечення:

1. Системне програмне забезпечення. Воно безпосередньо взаємодіє з обладнанням комп'ютера. В першу чергу це стосується ефективного управління комп'ютерною системою. Системне програмне забезпечення використовується для розробки нових системних програм і за допомогою завантаження ми можемо зробити їх портативними. Це залежить від машини. Системне програмне забезпечення також класифікується на три категорії: операційна система, яка діє як інтерфейс між користувачем і апаратним забезпеченням і надає різні послуги користувачам. Другий — програмне забезпечення системної підтримки, яке ефективніше керує апаратним забезпеченням. Іншим є програмне забезпечення для розробки системи, яке підтримує середовище розробки програм для користувача.

2. Прикладне програмне забезпечення. Воно призначене для вирішення проблем користувача відповідно до вимог користувача. Прикладне програмне забезпечення може бути загальним або специфічним. Прикладне програмне забезпечення також класифікується на дві категорії. Одна — це програмне забезпечення загального призначення, яке використовується для багатьох завдань і надає багато функцій. Іншим є програмне забезпечення спеціального призначення, яке призначене лише для певної мети. Наприклад, програми користувача. Основна увага приділяється додатку, а не комп'ютерній системі. В першу чергу це стосується вирішення деяких проблем з використанням комп'ютера як інструменту.

3. Інженерне/наукове програмне забезпечення стосується вимог обробки в конкретному додатку. Це програмне забезпечення спеціально використовується для креслення, моделювання, розрахунків навантаження та аналізу інженерно-статистичних даних для інтерпретації та прийняття рішень. Наприклад, CAD (Computer Aided Design), CAM (Computer Aided Manufacturing) і CAE (Computer Aided Engineering). Це програмне забезпечення використовується в галузі механіки, електрики, креслення, інженерії та аналізу. Вони працюють на мейнфреймах, робочих станціях загального призначення та ПК (персональних комп'ютерах) [15].

4. Вбудоване програмне забезпечення. Це програмне забезпечення вбудовується в апаратне забезпечення як частина більших систем для керування різними функціями. Цей тип програмного забезпечення вбудовано в ПЗУ (пам'ять

лише для читання) систем. Наприклад, програмне забезпечення для керування клавіатурою, вбудоване в мікрохвильову піч або пральну машину, де необхідно аналізувати вхідні дані, приймати рішення та вживати заходів, які дозволяють виробу працювати бажаним чином. Це програмне забезпечення також називають інтелектуальним програмним забезпеченням через його продуктивність.

5. Програмне забезпечення лінійки продуктів. Цей тип програмного забезпечення відноситься до методів, засобів і прийомів розробки програмного забезпечення для створення сукупності подібних програмних систем із спільного набору програмних активів з використанням загальних засобів виробництва. Це набір програмного продукту, який має спільні функції, але кожен з них певним чином відрізняється. Наприклад, вони можуть бути розроблені для конкретного замовника або для вбудованого програмного забезпечення (документ Word, електронна таблиця, комп'ютерна графіка, особисті та бізнес-додатки).

6. Веб-додаток. Це програма, доступ до якої здійснюється через веб-браузери через мережу, наприклад інтернет або інтранет. Це також програмна частина для комп'ютера, яка закодована мовою підтримки браузера та запускається у звичайному веб-браузері, щоб зробити програму виконуваною. Перше покоління веб-додатків дозволяє компанії розміщувати інформацію публічно. Таким чином, цю інформацію бачить кожен, хто має веб-браузер і доступ до Інтернету. Проблема першого покоління полягає в тому, що інформація знаходиться в статичному вигляді. Веб-додатки другого покоління дозволяють користувачам виконувати інтерактивні запити до баз даних із веб-додатків. Він характеризується полегшенням спілкування, обміном інформацією, зосередженістю на користувачах і співпраці в WWW. Третє покоління є більш корисним, ніж додатки другого покоління.

7. Програмне забезпечення штучного інтелекту. У цьому програмному забезпеченні використовуються нечислові алгоритми, які використовують дані, згенеровані системою, для вирішення складних задач, які не піддаються процедурам розв'язання задач і вимагають спеціального аналізу та інтерпретації проблеми для її вирішення. Наприклад, штучні нейронні мережі, робототехніка, експертні системи та комп'ютерні ігри. Все це програмне забезпечення може працювати як в режимі

реального часу, так і в автономному режимі. Цим програмним забезпеченням можна безкоштовно надавати спільний доступ за плату через сховище.

Програмно-апаратну платформу також називають цифровою або обчислювальною платформою. Це налаштування, в яких виконується частина програмного забезпечення. Це може бути як операційна система так і інше обладнання. Архітектура комп'ютера, операційна система чи бібліотеки – це деякі концептуальні рівні обчислювальних платформ. Можна сказати, що обчислювальна платформа – це фаза, на якій можуть працювати комп'ютерні програми. Цифрову платформу можна розуміти як обмеження процесу розробки програмного забезпечення, а також як помічник в процесі розробки. Різні платформи надають різноманітні функціональні можливості та обмеження для процесу розробки програмного забезпечення. Подібним чином вони надають функціональні можливості низького рівня, які є помічниками у процесі розробки. Наприклад, операційна система може бути платформою, яка витягує фундаментальні зміни в апаратному забезпеченні та надає загальні команди для дійсних файлів або для отримання мережі [16].

Програмно-апаратні платформи можуть містити апаратне забезпечення лише в умовах невеликих вбудованих систем. Вбудовані системи можуть контактувати лише з апаратним забезпеченням, без операційної системи. Браузер є прикладом веб орієнтованого програмного забезпечення. Браузер, працює на апаратній платформі та операційній системі. Це не стосується програмного забезпечення, що працює всередині браузера.

В результаті аналізу програмно-апаратних платформ прийнято рішення використати веб-додаток, як платформу для розробки інформаційної технології.

3.2 Обґрунтування вибору мови програмування

Перед початком реалізації інформаційної технології слід вибрати для використання правильну мову веб-програмування. Варто проаналізувати чому в конкретному випадку слід вибирати певні мови замість інших. З моменту виникнення Інтернету веб-програмування залишалось популярним і затребуваним навиком. У

суспільстві, яке керується Інтернетом, програмування для Інтернету продовжує свою тенденцію до зростання.

Вкрай важливим є вибір правильної мови веб-програмування, що найкраще підійде під певну задачу. Існує кілька факторів, що визначають, яка мова є найкращою. Ось як вибрати правильну мову веб-програмування для використання. Вибір «правильної» мови може ввести в оману. Це означає, що існують неправильні мови. Хоча деякі мови просто не сприяють реалізації ваших задач у програмуванні, існує багато різних методів досягнення бажаного результату. Особливо це стосується веб-програмування. Існують переважно веб-орієнтовані мови, такі як JavaScript, а також мови загального призначення, як-от Java. Так як прийнято рішення реалізувати інформаційну технологію у вигляді веб-додатку зосередимося на питаннях, які посприяють у визначенні потрібної мови веб-програмування. Кілька мов, зручних для створення веб-додатків наведено в таблиці 3.1.

Таблиця 3.1 – Мови програмування та їх характеристики

Мова програмування	Характеристики
JavaScript	<ul style="list-style-type: none">– об'єктно-орієнтована;– динамічна;– структурована;– має багато бібліотек;– не потребує компілятора
Ruby	<ul style="list-style-type: none">– зручний синтаксис, схожий на мову;– стислий;– Framework Ruby on Rails;– мультипарадигма
PHP	<ul style="list-style-type: none">– широко використовується;– багато ресурсів;– динамічна;– гнучке середовище

Розглянемо детальніше кожен з вищеписаних мов програмування.

Мова програмування PHP існує вже більше двох десятиліть і зарекомендувала себе як потужне і надійне рішення, здобувши армію прихильників і шанувальників. Однак ми повинні визнати, що ця мова кодування поступово втрачає свою популярність на користь новіших і оптимізованих аналогів. Назва PHP означає Hypertext Preprocessor і позначає мову сценаріїв на стороні сервера, що означає, що програми, написані на ньому, працюють на веб-серверах і не залежать від веб-браузера. Проте з роками сфера його використання змінилася, і сьогодні мова кодування PHP входить до числа найкращих і найпопулярніших інструментів програмування для веб-розробки завдяки своїм багатьом перевагам. Це вважається дуже ефективною технологією, яка пропонує зручний процес розробки з багатьма додатковими інструментами, які допомагають йому. Насправді, згідно з індексом популярності мови програмування (PYPL), PHP є п'ята за популярністю мова кодування в світі.

Мова PHP в основному використовується для веб-розробки, і вона справді хороша у цій області. Хоча спочатку вона використовувалась для створення динамічних веб-сторінок, розробники вважають за краще використовувати цю мову сценаріїв для створення серверної частини веб-додатків. Однак для початку PHP є мовою загального призначення, тому, якщо потрібно, вона може мати інші реалізації. Наприклад, можна створювати настільні програми за допомогою PHP. Більше того, починаючи з версії 5, PHP підтримує об'єктно-орієнтоване програмування, пропонуючи цілий новий набір можливостей [17].

Універсальність цієї мови сценаріїв є результатом її чудової здатності поєднуватися з іншими мовами програмування. Наприклад, розробники можуть писати розширення до PHP, використовуючи мову C, що дозволяє додати ще більше функцій. Крім того, PHP має велику кількість доступних бібліотек і фреймворків, які ще більше розширюють його можливості. Найпопулярнішими прикладами таких фреймворків є Laravel, Symfony, Phalcon, Zend Framework і Yii.

Два з найвідоміших прикладів програмного забезпечення, написаного на PHP – це Facebook і Wordpress. Wordpress є найпопулярнішою системою керування вмістом в Інтернеті: серед усіх веб-сайтів, які використовують такі системи, близько 48% з

них використовують Wordpress. Завдяки великій кількості плагінів, як вбудованих, так і сторонніх, Wordpress підходить практично для будь-якої ролі. Ви можете використовувати його для створення блогу, фотогалереї, інтернет-магазину, порталу новин та багатьох інших типів веб-сайтів.

Facebook є ще одним очевидним прикладом того, що може робити PHP. Ця соціальна мережа вже давно перетворилася на складне середовище, яке має різноманітний спектр функцій, включаючи обмін миттєвими повідомленнями, рекламу, ведення блогів, презентацію новин, відтворення відео та інші, але все ще базується на коді PHP, тісно пов'язаному з іншими веб-технологіями [18].

Популярність мови PHP є логічним результатом її численних переваг, які роблять її потужним та ефективним інструментом розробки. Розглянемо переваги PHP:

1. Великий вибір доступних спеціалістів. Популярність PHP породила численну спільноту розробників, частина з яких може бути потенційними кандидатами на роботу. Велика кількість наявних спеціалістів призводить до високої конкурентоспроможності та меншої затребуваної заробітної плати, що вигідно для зниження витрат на розробку. Крім того, цю мову досить легко вивчити та впровадити, тому навіть молодші розробники зазвичай можуть ефективно реалізувати базову функціональність програми.

2. Велика документація. Доступно багато навчальних посібників, посібників та інших довідкових матеріалів, які полегшують розробку та можуть стати допомогою та джерелом натхнення у важких ситуаціях. Ці матеріали також безцінні для початківців програмістів, що полегшують процес навчання з поступово зростаючими труднощами. І, як було описано вище, вивчення PHP є порівняно легким, хоча й не таким легким, як вивчення Python, наприклад, що ще більше додає його популярності.

3. Покращена швидкість завантаження. Використання PHP прискорює завантаження сторінок веб-сайтів у порівнянні з багатьма іншими технологіями веб-розробки. Наприклад, наразі PHP приблизно втричі швидший за Python для більшості сценаріїв використання. У свою чергу, менший час завантаження є важливим фактором рейтингу SEO, який сприяє подальшому просуванню веб-сайту,

забезпечуючи конкурентні переваги. Вища швидкість додатків забезпечує задоволення клієнтів і, у поєднанні з іншими перевагами, допомагає створити та зберегти клієнтську базу.

4. Широкий вибір баз даних. PHP дозволяє підключатися практично до будь-якого типу бази даних. Найпоширенішим вибором є MySQL, головним чином тому, що він безкоштовний, ефективний і популярний серед розробників. Інші надійні варіанти систем керування базами даних, сумісні з PHP – це mSQL, MS-SQL, SQLite, PostgreSQL тощо. Крім того, PHP можна однаково добре використовувати з Elasticsearch, Redis, MongoDB та іншими нереляційними базами даних. Таким чином, розробники не обмежуються використанням конкретної бази даних і можуть вибрати найбільш оптимальну для майбутнього додатка, враховуючи всі важливі фактори.

5. Недороге програмне забезпечення з відкритим кодом. PHP — це безкоштовна технологія, яка дає значну економію бюджету на розробку. Крім того, більшість інструментів розробки, які зазвичай використовуються в поєднанні з PHP, є програмним забезпеченням з відкритим кодом і можуть використовуватися безкоштовно; тим самим вони додатково знижують вартість проекту. Крім того, існують численні фреймворки, такі як Laravel і CodeIgniter, а також різні CMS, такі як Wordpress і Drupal, наприклад, які розширюють функціональність PHP і роблять процес розробки простішим і ефективнішим.

6. Дешевші послуги хостингу. Найпоширенішим сценарієм роботи веб-сайту PHP є стек LAMP. Це означає, що веб-сайт працює на веб-сервері Apache HTTP, розгорнутому в системі Linux, і використовує MySQL як базу даних. Всі ці компоненти безкоштовні, а стек добре перевірений, що передбачає скорочення часу та коштів на розробку.

7. Відмінна комбінованість з HTML. PHP пропонує вбудоване програмування HTML, що є причиною неймовірної синергії між цими двома технологіями. У більшості випадків PHP-скрипт не втручається в HTML-код веб-сторінки, а завершує його, залишаючись у межах, визначених тегами `<?php ?>` [18].

8. Хороша гнучкість. Завдяки гнучкості PHP може ефективно поєднуватися з багатьма іншими мовами програмування, щоб програмний продукт міг

використовувати найбільш ефективну технологію для кожної окремої функції. Крім того, PHP є міжплатформною мовою, що означає, що розробники можуть використовувати будь-яку основну операційну систему – Windows, Linux, macOS – для виконання кодування. Така гнучкість значно полегшує процес розробки, роблячи його швидшим і менш дорогим [19].

9. Сумісність з хмарними сервісами. Нині багато сучасних продуктів, як правило, використовують рішення хмарних обчислень, як-от Amazon Web Services, для різних цілей. Програми, написані на PHP, підтримуються різними хмарними сервісами, такими як AWS Lambda, наприклад. Таким чином, PHP-додаток можна розгорнути на хмарному сервері і досягти відмінної масштабованості та інших корисних ефектів. Більш того, область хмарних обчислень не монополізована іншими мовами кодування, тому PHP зайняв своє місце в таких реалізаціях.

Хоча PHP, безсумнівно, корисний у сфері веб-розробки, він також має кілька недоліків, які не дозволяють йому домінувати в цій області. Розглянемо ці недоліки та дізнаємося, як вони можуть бути шкідливими для майбутнього програмного забезпечення та його бізнес-реалізації. Три основних недоліки PHP:

1. Зниження популярності. Хоча PHP є потужним інструментом, який підтримується великою спільнотою та великою довідковою документацією, існують простіші мови програмування для веб-програм. З цієї причини початківці розробники вважають за краще вивчати JavaScript як свою першу мову і рідко замислюються про додавання PHP до своїх навичок. Зараз PHP домінує в сегменті веб-розробки, але, швидше за все, це зміниться в майбутньому. Згодом кількість фахівців скоротиться, а початківців розробників, які пропонують базові навички за низькою ціною, буде бракувати, тому вартість продуктів, створених на PHP, ймовірно, зросте.

2. Відсутність спеціалізованих бібліотек для сучасних потреб. Наприклад, машинне навчання зараз є гарячою тенденцією, і вона точно збереже свою популярність і в найближчому майбутньому. Незважаючи на те, що PHP має свій набір бібліотек, він не може конкурувати з JavaScript у розробці веб-додатків за допомогою машинного навчання. Наразі PHP не може запропонувати настільки ж швидкі й ефективні альтернативи. Таким чином, якщо вашій програмі потрібна

функціональність ML або може знадобитися це в майбутньому, коли ваш бізнес буде розширюватися, PHP не найкращий вибір.

3. Вади безпеки. Протягом багатьох років побоювання щодо безпеки продуктів на основі PHP все ще зберігаються з кількох причин. Однією з них є природа PHP з відкритим вихідним кодом, що означає, що можливі вразливості коду стають загальновідомими після їх виявлення. Таким чином, принаймні теоретично, протягом періоду між їх виявленням і виправленням у нових версіях мови програмування ці вразливості можуть бути використані будь-яким програмістом зі зловмисним умислом і відповідними навичками. Однак цей недолік можна в рівній мірі віднести до інших технологій з відкритим кодом загалом, оскільки він переважно стосується моделі з відкритим кодом, а не мови PHP зокрема [20].

JavaScript — це дуже потужна мова сценаріїв на стороні клієнта, введена для того, щоб «оживити веб-сторінки». Це дозволяє створювати динамічний вміст для Інтернету. JavaScript — це легка мова з відкритим вихідним кодом і дозволяє використовувати багато платформ. Вона не вимагає компіляції та інтерпретується за допомогою об'єктно-орієнтованих можливостей. Крім того, вона працює з різними іншими мовами програмування. І це є причиною його широкого використання в усьому світі. Багато популярних веб-сайтів та веб-програм, таких як Google, Amazon, PayPal тощо, використовують цю мову. Розширення файлу JavaScript – .js.

Брендан Айх розробив JavaScript у 1995 році, працюючи в Netscape Communications, Java, Scheme і Self, надихнувши його. Оскільки Microsoft стала смертельною загрозою, Netscape розпочав процес стандартизації, щоб запобігти доступу Microsoft до JavaScript. Вони також співпрацювали з Sun Microsystems (зазвичай її називають Sun), щоб зламати монополію Microsoft. Причина, чому партнерство між Sun і Netscape працювало настільки ідеально, полягало в тому, що вони обидва мали одну ціль. JavaScript спочатку був відомий як LiveScript від Netscape і Mocha. Пізніше вони перейменували LiveScript/Mocha на JavaScript. Хоча Java і Javascript звучать схожі, вони не схожі. Вони мають дуже різний синтаксис, семантику та використання. Єдине спільне те, що обидва є торговими марками або зареєстрованими торговими марками Oracle у США та інших країнах.

Через популярність Java в той час спочатку називали JavaScript «молодшим братом Java». Але незабаром JavaScript став незалежним з власним ECMAScript в 1997 році і зовсім не нагадує Java. JavaScript 1.8.5 є останньою версією і включає багато з ECMA-262 Edition 5 відповідності. Розробники випустили його 27 липня 2010 року.

Проаналізуємо навіщо потрібен JavaScript. Більшість програм працюють завдяки взаємодії між клієнтом (пристроєм користувача) і віддаленим сервером. Клієнт запитує дані від сервера. Сервер отримує запит, обробляє його, а потім відповідає відповідно. Відповідь, надіслана назад, має формат, зрозумілий користувачеві, і тому є прийнятним для клієнта. Але цей процес вимагає часу та ресурсів. JavaScript дозволяє перевіряти форми без введення сервера, зменшуючи трафік. Він надає чудові інструменти для більш інтерактивного та зручного веб-сайту. Деякі з основних функцій JavaScript:

- автозаповнення: вікно пошуку дає пропозиції на основі того, що вже ввів користувач;
- перевірка форми: якщо користувачі роблять помилку під час заповнення форми, JavaScript негайно повідомляє їх про помилку, уникаючи повторного заповнення;
- вирішує проблеми з макетом, щоб уникнути накладання елементів на сторінці;
- додає анімацію на сторінку, щоб зробити її більш привабливою.

JavaScript є однопоточним [21]. Це означає, що інструкції виконуються послідовно, по одній. Це можливо за допомогою наступних компонентів:

1. Механізм JavaScript. Це програма, що відповідає за переклад вихідного коду на машинну мову та його виконання на ЦП. Кожен сучасний браузер оснащений механізмом JavaScript. Таким чином, немає необхідності завантажувати будь-яке додаткове програмне забезпечення. Двигун складається з 2 компонентів: куча пам'яті – тут відбувається виділення пам'яті; стек викликів – коли сценарій викликає функцію, інтерпретатор спочатку додає її до стеку викликів, а потім починає обробляти.

2. Час виконання. Механізм JavaScript працює всередині середовища для додаткових функцій, які використовуються під час виконання. Хоча JavaScript є однопоточним, середовище виконання складається з пулу потоків. Це дозволяє JavaScript одночасно працювати у фоновому режимі, поки користувач переглядає, не перериваючи виконання програми.

3. Петля подій. Цей механізм керує всіма потоками з пулу потоків у порядку їх виконання, тобто обробляє зворотні виклики. Зворотний виклик – це фрагмент коду, який виконується, коли відбувається певна подія, тобто клацання мишею. Якщо відбувається подія, середовище поміщає зворотний виклик в обробник подій у циклі подій. Зворотні виклики завжди виконуються по одному.

Нижче наведено основні функції JavaScript:

- Підтримує концепції об'єктно-орієнтованого програмування.
- Не залежить від платформи та чутливий до регістру.
- Надає різні вбудовані функції, такі як alert(), prompt() тощо.
- Можливість обробки винятків.
- Дозволяє використовувати функції з будь-яким ім'ям або без нього. Функція без імені є анонімною функцією.

Застосування JavaScript:

– Веб-розробка. Веб-розробка є окремою мовою для створення веб-сторінок. Він також підтримує зовнішні програми, такі як документи PDF, запущені віджети тощо. Це також додає різні спеціальні ефекти до сторінки, як-от графіку.

– Веб-додатки. Веб-програми взаємодіють з браузером, не надсилаючи повідомлення між браузером і сервером. За допомогою різних фреймворків, доступних на ринку, дуже легко створювати інтерактивні сторінки. Це економить час і зусилля, необхідні розробнику для створення веб-додатку.

– Презентації. Презентація JavaScript надає зручні бібліотеки та рамки для презентацій. Вона забезпечує вишукані теми, які не надто кричущі.

– Серверні програми. Сервер-додатки Javascript також дуже корисний для створення серверних програм і швидший, ніж інші серверні мови. NodeJS — це безкоштовне серверне середовище з відкритим вихідним кодом, яке

використовується для цього. Воно дозволяє генерувати динамічний веб-контент, змінювати базу даних, збирати дані форм тощо.

– Веб-сервери. Веб-сервери NodeJS має вбудований модуль (набір функцій), який дозволяє створити HTTP-сервер. Незважаючи на те, що JavaScript почався як сценарій на стороні клієнта, тепер JavaScript повністю здатний також виконувати сценарії на стороні сервера.

– Ігри. Разом із HTML5, JavaScript допомагає у розробці ігор. Бібліотека EaselJS надає прості рішення для роботи з насиченою графікою. Вона також має API, знайомий для багатьох розробників.

– Використовуючи JavaScript, малювати графіку за допомогою HTML стало легше на веб-сторінці. Полотно є без кордонів або вмісту, тому користувачі можуть створювати власне мистецтво. JavaScript забезпечив користувача середовищем для різних проектів цифрового мистецтва.

– Програми для розумних годинників. JavaScript містить бібліотеку PebbleJS, яка надає прості рішення для роботи з насиченою графікою. Розробник має доступ до багатьох функцій програми, створених для розумних годинників.

– Мобільні програми. JavaScript забезпечує структуру під назвою PhoneGap, яка підтримує мобільні додатки. React Native сьогодні також служить цій же меті. Маючи пристойні знання HTML, CSS і JavaScript, можна створювати чудові програми.

– Машинне навчання. Javascript представив машинне навчання разом з усіма попередніми функціями. Різні фреймворки, такі як Keras.js, BrainJS, Compromise і WebDNN, покращують функціональність JavaScript у сфері AI [22].

Розглянемо переваги JavaScript:

1. Швидкість – JavaScript має тенденцію працювати дуже швидко, оскільки часто запускається безпосередньо в браузері клієнта. Поки він не потребує зовнішніх ресурсів, JavaScript не сповільнюється викликами сервера. Крім того, всі основні браузери підтримують JIT компіляцію для JavaScript, що означає, що немає необхідності компілювати код перед його запуском.

2. Простота – синтаксис JavaScript був натхненний Java і його відносно легко вивчити порівняно з іншими популярними мовами, такими як C++.

3. Популярність – JavaScript є скрізь в Інтернеті, а з появою Node.js все частіше використовується у бекенд. Існує незліченна кількість ресурсів для вивчення JavaScript. Як StackOverflow, так і GitHub демонструють все більшу кількість проєктів, які використовують JavaScript, і очікується, що популярність, яку він отримав за останні роки, лише збільшиться.

4. Сумісність – на відміну від PHP або інших мов сценаріїв, JavaScript можна вставити на будь-яку веб-сторінку. JavaScript можна використовувати в багатьох різних програмах завдяки підтримці інших мов, таких як Perl і PHP.

5. Завантаження сервера – JavaScript є клієнтським, тому він загалом зменшує попит на сервери, і простим додаткам може взагалі не знадобитися сервер.

6. Розширені інтерфейси – JavaScript можна використовувати для створення таких функцій, як перетягування компонентів, таких як повзунки, які значно покращують інтерфейс користувача та роботу сайту.

7. Розширена функціональність – розробники можуть розширити функціональність веб-сторінок, написавши фрагменти JavaScript для сторонніх доповнень, таких як Greasemonkey.

8. Універсальність – існує багато способів використання JavaScript через сервери Node.js. Якщо ви завантажуєте Node.js за допомогою Express, використовуєте базу даних документів, як-от MongoDB, і використовуєте JavaScript на інтерфейсі для клієнтів, можна розробити цілу програму JavaScript від початку до кінця, використовуючи лише JavaScript.

9. Оновлення – з появою ECMAScript 5 (специфікація сценаріїв, яку використовує JavaScript), ECMA International було присвячено оновленню JavaScript щорічно.

Розглянемо недоліки JavaScript:

1. Клієнтська безпека – оскільки код JavaScript виконується на стороні клієнта, помилка і прорахунки можуть іноді бути використані в зловмисних цілях. Через це деякі люди вирішують повністю вимкнути JavaScript.

2. Підтримка браузерів – у той час як серверні скрипти завжди дають той же результат, різні браузери іноді інтерпретувати код JavaScript різному. Ці відмінності

мінімальні, і не доводиться турбуватися про це, поки тестується скрипт у всіх основних браузерах.

Отже, вище було проаналізовано такі мови програмування як PHP, Ruby та JavaScript, описано їх основні характеристики, переваги та недоліки. Згідно цієї інформації прийнято рішення використати для розробки інформаційної технології мову програмування JavaScript через її переваги, такі як швидкість, сумісність та універсальність.

3.3 Реалізація основних модулів інформаційної технології проведення колективних онлайн експертиз

Реалізація інформаційної технології проведення колективних онлайн експертиз виконана в вигляді клієнт серверної архітектури. Клієнт-серверна архітектура, як було розглянуто раніше, є трьохрівневою, так як крім клієнта та сервера інформаційної технології є також сервер бази даних. Для кожної частини використано свою мову: для клієнтської частини це JavaScript, для серверної – Node.js та база даних MongoDB. Підключення клієнта до сервера будується на HTTP протоколі [23].

Протокол передачі гіпертексту (HTTP) є основою всієї павутини і використовується для завантаження веб-сторінок за допомогою гіпертекстових посилань. HTTP — це протокол прикладного рівня, призначений для передачі інформації між мережевими пристроями і працює поверх інших рівнів стеку мережевих протоколів. Типовий потік через HTTP передбачає, що клієнтська машина робить запит до сервера, який потім надсилає повідомлення-відповідь.

З'єднання буде виконуватися тільки про необхідності отримати інформацію з сервера або навпаки надіслати якісь дані на сервер. Через необхідність передавати повний шлях до запиту, для кожного пишеться окрема функція-обробник на Node.js. Фрагмент функції обробника GET запиту наведено нижче. Ця функція працює на сервері наступним чином: отримує ідентифікатор експертизи з параметра запиту та звертається до бази даних з метою отримати інформацію про відповідну експертизу.

Отримавши необхідну інформацію, функція повертає результат для подальшого відображення.

```
app.get('/result/:id', function (req, res) {
  const { id } = req.params;
  getVoteById(id)
    .then(responce => {
      if (responce.experts.length)
        res.render('diagram-page', {vote: responce});
      else
        res.render('no-result');
    });
});
```

Таким чином працює взаємодія клієнт серверної архітектури в процесі роботи інформаційної технології. Перейдемо до окремих частин програмної реалізації, що відповідають за основний функціонал системи. Найважливішим етапом проведення експертизи є створення експертизи та призначення експертних оцінок конкретній експертизі. В процесі опису програмної реалізації розглянемо також взаємодію компонентів клієнта, сервера та бази даних і обмін інформацією між ними. Нижче наведено програмну реалізацію клієнтської частини.

```
$('.js-name-list').on('submit', (e) => {
  e.preventDefault();
  const $names = $('.${SETTINGS.SELECTOR.INPUT_NAME}');
  const votes = this.isMarksChanged ? this.getMarks() : [...DEFAULT_MARKS];
  let candidates = [];

  $names.each((index, element) => {
    candidates.push({
      name: element.value,
      votes
    });
  });

  this.hide();
  addVote({
    candidates,
```

```

    name: window.name,
    createdBy: localStorage.getItem('userId')
  }).then(id => {
    window.location.assign(`${window.location.origin}/vote/${id}`);
  });
});

```

Це функція обробник, яка викликається після того, як користувач заповнив назву експертизи та назви альтернатив. Далі ця функція працює наступним чином:

- зчитується назва експертизи;
- зчитуються назви введених альтернатив з відповідних полів;
- відбувається перевірка на зміну користувачем експертних оцінок;
- якщо оцінки не змінювались, то використовуються значення за замовчуванням;
- якщо оцінки змінено, то значення які було відредаговано, зчитуються для подальшого використання;
- формується тіло запиту, який відправляється на сервер;
- після отримання результату від сервера відбувається перехід до сторінки оцінювання створених альтернатив.

Далі розглянемо програмну реалізацію запиту до сервера та його алгоритм роботи.

```

app.post('/candidates', function(req, res) {
  const { vote } = req.body;

  addVoteToVotes({ ...vote, experts: [] })
    .then(id => {
      res.json(id);
    })
    .catch(error => {
      console.error(error.message);
    });
});

```

Це серверна функція-обробник, що працює з POST методом та отримує дані з клієнтської частини. Дана функція відповідальна за звернення до бази даних для

запису в неї інформації, яку отримано з клієнтської частини. Алгоритм роботи цієї функції наступний:

- отримує з клієнтської частини тіло запиту з даними;
- ініціює запит до бази даних для збереження отриманих даних;
- якщо дані збережено та відповідь від бази успішна, то до клієнта надсилається ідентифікатор створеного запису;
- якщо база даних повертає помилку, то помилка повертається в консоль сервера.

Далі розглянемо програмну реалізацію взаємодії сервера і бази даних.

```
async function addVoteToVotes(element) {
  const client = await MongoClient.connect(url, { useUnifiedTopology: true })
  const db = client.db(databaseName);
  const collection = db.collection(collectionName);
  const Vote = await collection.insertOne(element);
  await client.close();
  return Vote.insertedId;
}
```

Це функція звернення до бази даних, що відповідає за зберігання користувацьких даних в базі. Спочатку створюється зв'язок з клієнтом бази даних по URL адресі. Далі створюється зв'язок з потрібною базою даних. Далі відбувається пошук відповідної колекції та запис даних. Після цього зв'язок з клієнтом бази завершується і повертається ідентифікатор записаних даних.

3.4 Тестування інформаційної технології проведення колективних онлайн експертиз

Інформаційну технологію проведення колективних онлайн експертиз реалізовано як веб орієнтовану систему. Це підвищить попит та зручність використання даної технології, за рахунок того, що користувачеві не потрібно робити додаткових інсталяцій на персональний компютер або смартфон а залишиться лише перейти за веб посиланням та почати використання [24].

Основною метою під час проектування та розробки інформаційної технології було підвищення ефективності процесу проведення колективних онлайн експертиз в умовах віддаленої комунікації між експертами за рахунок розширення функціональних можливостей інформаційної технології. Для того, щоб підтвердити досягнення вищеписаної мети проведемо тестування реалізованої технології. В ході тестування перевіримо наступний функціонал:

1. Створення нової експертизи за класичним сценарієм.
2. Створення нової експертизи зі зміною оцінок.
3. Проведення експертизи.
4. Відображення результатів експертизи.
5. Видалення експертизи.

Після опису основного сценарію тестування перейдемо до його реалізації. Протестуємо створення нової експертизи за класичним сценарієм. Для того, щоб почати створення нової експертизи перейдемо на вкладку «Нова експертиза» як це показано на рисунку 3.2.



Рисунок 3.2 – Перехід до створення нової експертизи

В результаті натискання на кнопку «Нова експертиза» відбувся успішний перехід до процесу створення. Перевіримо чи можливо продовжити створення експертизи не вказавши назву. Перевірку введення пустої назви експертизи наведено

на рисунку 3.3. Як видно на рисунку 3.3. створити експертизу з пустою назвою не вдається.



Рисунок 3.3 – Введення пустої назви експертизи

Введемо назву експертизи та перевіримо чи ця операція відбувається успішно на рисунку 3.4.



Рисунок 3.4 – Введення назви експертизи

Після підтвердження введеної назви відбувся успішний перехід на сторінку для введення необхідної кількості альтернатив. Перехід до введення кількості альтернатив наведено на рисунку 3.5.



Рисунок 3.5 – Введення кількості альтернатив

Ввівши потрібну кількість альтернатив та підтвердивши введені дані відкривається можливість редагування назв альтернатив. Відредагуємо назви альтернатив як це показано на рисунку 3.6 та перевіримо чи відбувається перехід безпосередньо до процесу проведення експертизи.



Рисунок 3.6 – Редагування назв альтернатив

Назви альтернатив успішно вказано та після натискання кнопки «Далі» відбувся очікуваний перехід до сторінки з проведенням експертизи. Сторінку з проведенням експертизи та експертні оцінки наведено на рисунку 3.7.

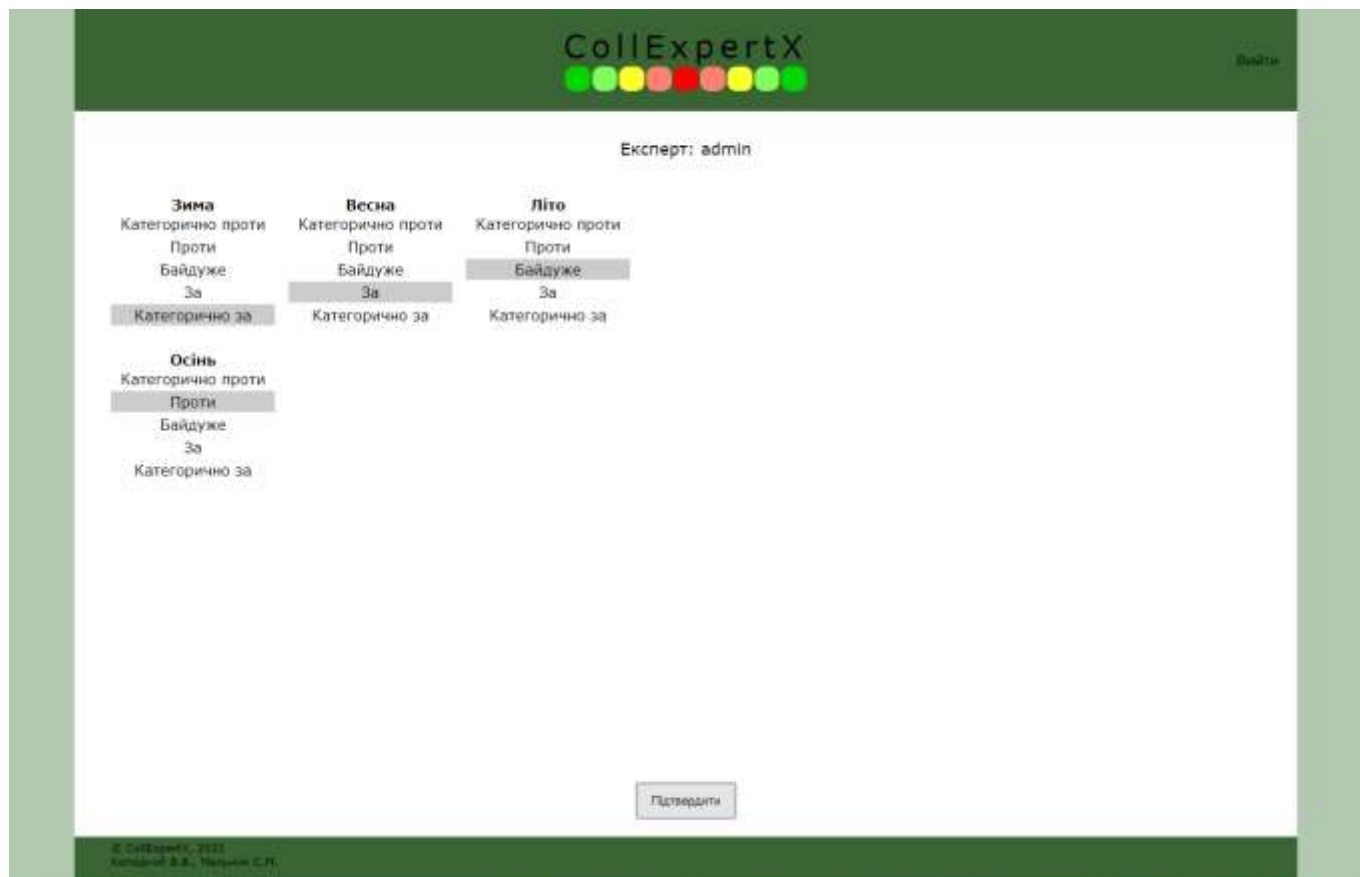


Рисунок 3.7 – Сторінка проведення експертизи з експертними оцінками

Перевірка показала, що в процесі проведення експертизи наявно п'ять експертних оцінок, які збережено в системі за замовчуванням і експерт має змогу успішно їх змінювати в залежності від свого ставлення до певної альтернативи [25]. Перевіримо чи зберігаються результати після натискання кнопки підтвердження та чи відбувається перехід до результатів експертизи. Результат перевірки наведено на рисунку 3.8.



Рисунок 3.8 – Результати експертизи

Результати експертизи отримано, тому можна зробити висновок, що процес створення нової експертизи за класичним сценарієм перевірено та визначено що він працює успішно. Перейдемо до наступного тест-кейсу, а саме створення нової експертизи зі зміною оцінок. Початок створення експертизи до етапу введення назв альтернатив є ідентичним вищеописаному сценарію. Створимо ще одну експертизу під назвою «МКР – тест 2», що включатиме в себе альтернативи: «ніч», «ранок», «день», «вечір» та натиснемо кнопку «Змінити оцінки» як це показано на рисунку 3.9.



Рисунок 3.9 – Перехід до зміни оцінок

Перехід до зміни оцінок відбувся успішно. Тепер вкажимо нові оцінки, а саме «Подобається», «Не подобається» та поставимо числове значення для них 10 та (-10) і кольори жовтий та синій. Перевірку вищеописаного процесу наведено на рисунку 3.10.



Рисунок 3.10 – Зміна оцінок для альтернатив

Як видно на рисунку 3.10 назви, значення та колір оцінок для альтернатив успішно змінюється. Перевіримо функцію додавання та видалення оцінок. Додамо ще дві оцінки, а саме «Байдуже» та «Утримуюсь» з значеннями рівними нулю та сірим кольором. Додавання оцінок наведено на рисунку 3.11.



Рисунок 3.11 – Додавання двох нових оцінок

Дві нові оцінки додано успішно. Тепер виникла ситуація, що дві нових оцінки схожі за сенсом між собою, тому перевіримо процес видалення, видаливши оцінку «Утримуюсь». Перевірку процесу видалення оцінки наведено на рисунку 3.12.



Рисунок 3.12 – Процес видалення оцінки

Процеси додавання і видалення перевірено, вони пройшли успішно. І тепер маємо чотири бажані альтернативи та три новий оцінки. Перейдемо далі, щоб перевірити процес проведення експертизи. Процес проведення експертизи та додавання експертних оцінок наведено на рисунку 3.13.



Рисунок 3.13 – Процес проведення експертизи

Процес проведення експертизи з новими експертними оцінками успішний. Експерт має змогу вибирати альтернативи згідно з власними переважаннями. Підтвердивши оцінки експертів, перевіримо перехід до сторінки результатів на рисунку 3.14.

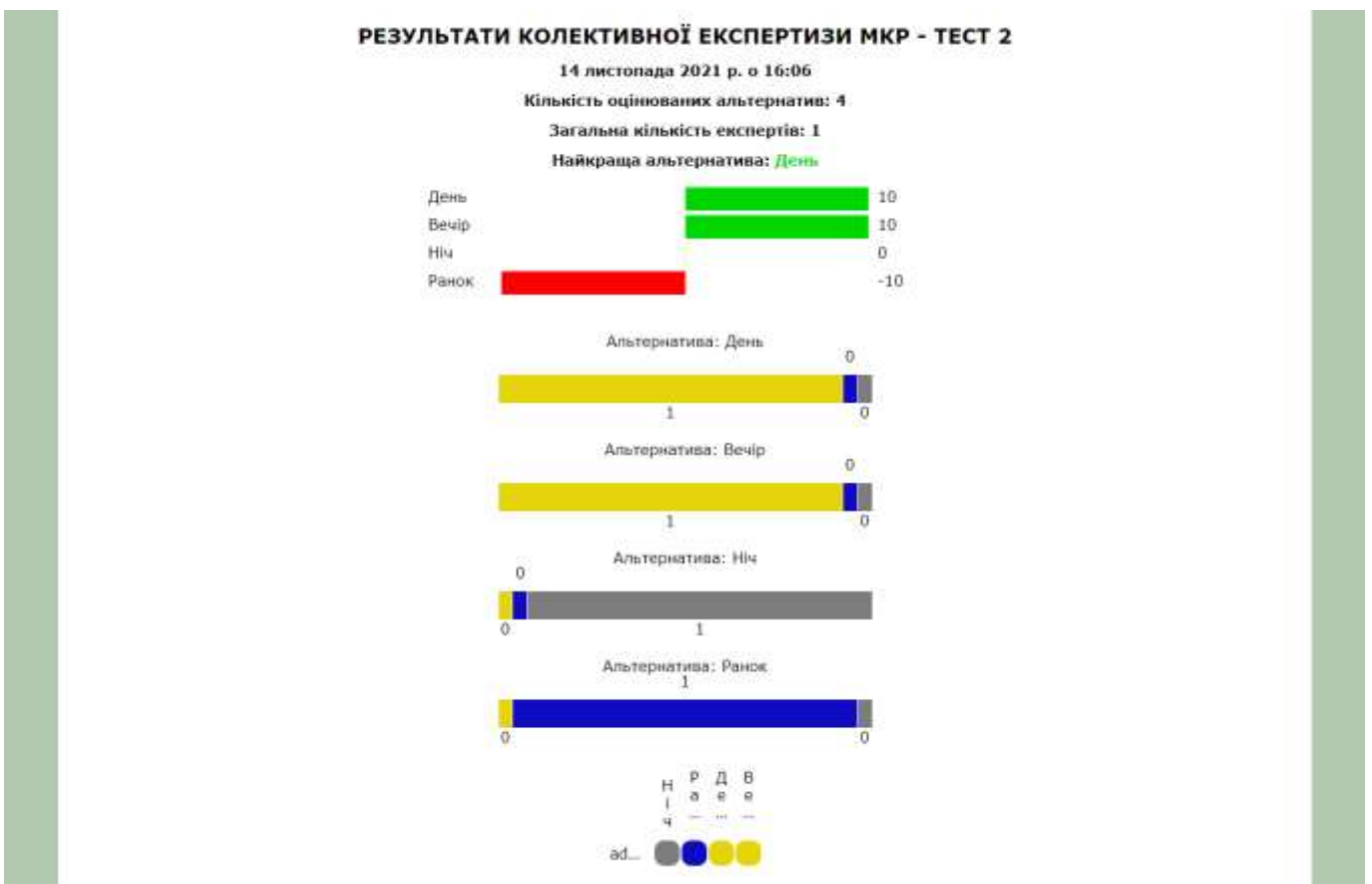


Рисунок 3.14 – Перехід до сторінки результатів

Перехід до сторінки результатів виконано успішно, результати експертизи відображаються як і очікувалося. Тому можна зробити висновок, що перевірка створення нової експертизи зі зміною оцінок успішна. В ході виконання перших двох перевірок було включено процес проведення експертизи та відображення результатів експертизи. Як видно в попередніх перевірках, обидва процеси успішно працюють як для класичного сценарію створення експертизи, та і для експертизи з відредагованими експертними оцінками.

Останній пункт, який варто перевірити це видалення експертиз. Важливо відмітити, що користувач не повинен мати можливість видалити експертизи, які він не створював. Тому, перейдемо до сторінки з експертизами натиснувши кнопку «Експертизи», як це показано на рисунку 3.15. Сторінку з переліком експертиз наведено на рисунку 3.16.

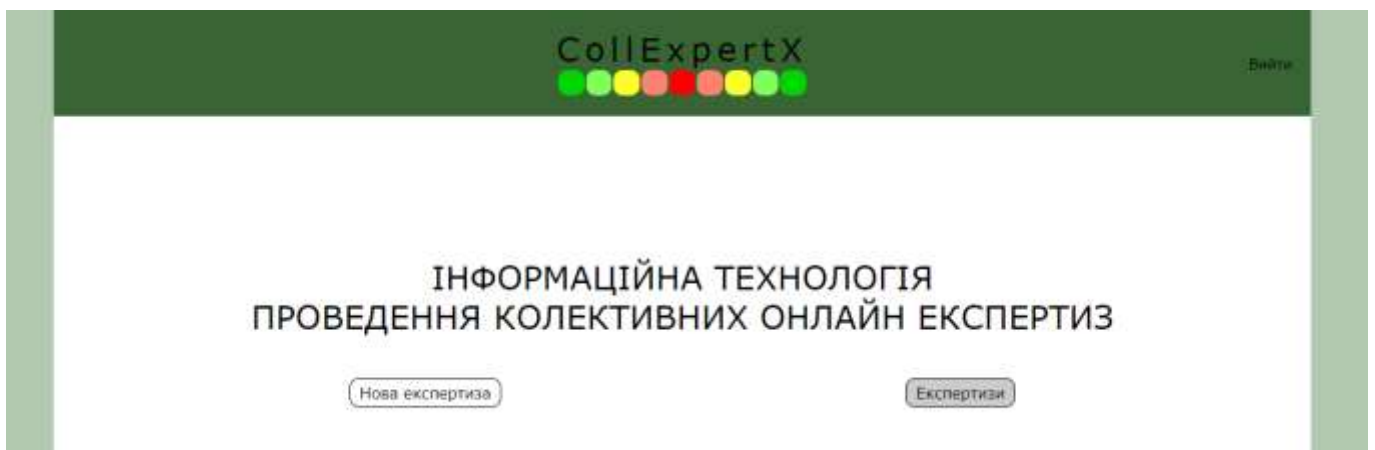


Рисунок 3.15 – Перехід до сторінки з експертизами



Рисунок 3.16 – Сторінка з експертизами

Перейшовши до сторінки з експертизами бачимо перелік доступних експертиз. Серед доступних експертиз, кнопка видалити доступна лише для певної частини з них. Це пов'язано з тим, що активний користувач може видалити лише частину експертиз, а саме ті, що створив власноруч. Видалимо дві зайвих експертизи під назвою «МКР – тест 2» натиснувши на кнопку видалити які це показано на рисунку 3.17.

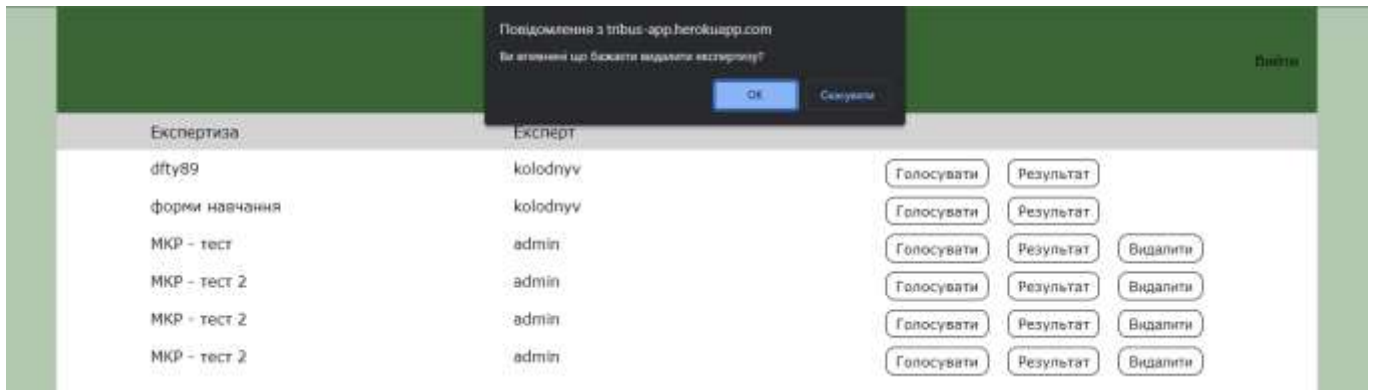


Рисунок 3.17 – Видалення експертизи

Після натискання на кнопку видалення, користувач отримує попередження про видалення експертизи, щоб запобігти випадковому видаленню важливої інформації. Після підтвердження видалення експертизи, перевіримо чи вона зникла з переліку експертиз. Результат перевірки видалення експертизи показано на рисунку 3.18



Рисунок 3.18 – Результат перевірки видалення експертизи

Як бачимо на рисунку 3.18 видалення непотрібних експертиз пройшло успішно, тому робимо висновок, що функціонал який відповідає за видалення експертиз працює коректно. Тому, можна зробити висновок, що тестування функціоналу інформаційної технології завершено успішно.

Перейдемо до тестування ефективності розробленої інформаційної технології. Перевірку ефективності виконано за допомогою порівняння розробленої інформаційної технології й систем-аналогів – ВМПП та VisPA. Для порівняння залучено три групи експертів по 10 чоловік. Експертами виступають студенти університету та учні старшої школи. До перевірки залучено в загальному 15 учнів та 15 студентів. Кожна з груп містить в рівній кількості студентів та учнів, задля об'єктивності тестування.

Групою №1 перевірено мобільний додаток VisPA, групою №2 досліджено систему ВМПП і відповідно групою №3 протестовано розроблену інформаційну технологію – CollExpertX. Перед створеними групами експертів стоїть однакова задача – визначитися з місцем, де можна відпочити на вихідних з класом (групою). Так як в системі ВМПП максимальна кількість альтернатив обмежена та дорівнює 9, зробимо кількість альтернатив рівною для всіх груп експертів. Тому, експертам потрібно вибрати найкращу альтернативу серед 9 наступних: поїздка в музей, відвідування старовинного замку, екскурсія історичними пам'ятками міста, поїздка на природу за місто, похід в аквапарк, похід в кіно, відвідування майстеркласу з гончарного мистецтва, кулінарний майстерклас, візит в науково-розважальну лабораторію. По закінченню колективних експертиз, кожен учасник групи повинен виставити оцінку ефективності процесу. Оцінки мають бути в межах від 1 до 10, де 10 – повністю задоволений, та базуватися на трьох аспектах – час витрачений на визначення найкращої альтернативи, зручність використання реалізованої технології та задоволення від отриманого загального результату. З отриманих оцінок від експертів, виведено середню оцінку ефективності використання кожної з систем та порівняно їх між собою.

Група №1, що відповідальна за мобільний додаток VisPA, в оцінила його ефективність в 7 балів. Основними перевагами даного додатку визначено його

доступність з мобільного пристрою та зручність проведення індивідуальної експертизи. Головним недоліком визначено відсутність чіткого розуміння числової інтерпретації альтернатив та важкість зведення фінальних результатів після кожної індивідуальної експертизи.

Група №2 оцінила ефективність системи ВМПП в 6 балів. Серед переваг визначено наявність числової інтерпретації важливості альтернатив, зручність та наочність процесу експертизи. Серед недоліків група відмітила, що проведення експертизи забирає багато часу при великій кількості альтернатив, так як виконується порівняння кожен з кожним. Також систему визначили малоефективною для великої групи експертів, так як експертиза проводиться індивідуально і по черзі, й отримання фінального результату потребує додаткових зусиль.

Інформаційну технологію CollExpertX оцінено групою №3 в 9 балів. Значними перевагами даної технології для групи виявились наступні:

- можливість чітко визначити свою позицію та вказати своє ставлення до альтернативи;
- процес визначення найкращої альтернативи не займає багато часу, так як альтернативи оцінюються одночасно за один підхід;
- зручно використовувати для великого колективу, так як технологія надасть чітке та обдумане фінальне рішення, що заощадить людський ресурс.

Серед недоліків визначено відсутність можливості редагування експертизи та оцінок після її збереження.

Отже, ефективність роботи розробленої технології на 23% вища, ніж аналог VisPA та на 34% вища, ніж аналог ВМПП.

Результати тестування повністю задовольняють мету розробки інформаційної технології, а підвищення ефективності проведення експертизи за допомогою розширення функціональних можливостей порівняно з системами аналогами. Увесь функціонал який відрізняє інформаційну технологію від систем аналогів реалізовано та протестовано з успішними результатами. За рахунок цього ефективність процесу проведення колективних онлайн експертиз в умовах віддаленої комунікації значно покращується, а якість отриманих результатів зростає.

3.5 Висновок

В результаті виконання третього розділу магістерської кваліфікаційної роботи виконано обґрунтування вибору програмно-апаратної платформи для розробки інформаційної технології. Було проаналізовано сім кваліфікацій програмно-апаратного забезпечення та прийнято рішення використовувати веб-додаток.

Далі було проведено обґрунтування вибору мови програмування. В ході аналізу мов програмування виділено PHP, Ruby та JavaScript. Наведено основні області їх застосування та проаналізовано переваги та недоліки. Визначено, що для розробки поточної інформаційної технології найбільш доцільно використати мову програмування JavaScript.

Здійснено реалізацію основних модулів інформаційної технології. В ході роботи, реалізовано клієнт-серверну архітектуру інформаційної технології. А також описано взаємодію клієнт-серверних компонентів та алгоритми функціонування основних модулів.

Також виконано тестування розробленої інформаційної технології. В процесі тестування перевірено основний функціонал згідно з описаними тест-кейсами. Всі перевірки пройшли успішно, розроблена інформаційна технологія працює коректно та є на 28.5% ефективнішою за системи аналоги.

4 ЕКОНОМІЧНА ЧАСТИНА

4.1 Комерційний та технологічний аудит науково-технічної розробки

Метою даного розділу є проведення технологічного аудиту, в даному випадку нового програмного продукту інформаційної технології проведення колективних онлайн експертиз. Особливістю програми є те, що данна технологія поєднує в собі можливість проведення колективної експертизи та надає якісну та зрозумілу візуалізацію результуючих даних, тому що прийняття колективних експертних рішень це важливий процес. Рішення потрібно приймати з урахуванням інтересів і позицій всіх членів групи, так як це підвищить їх якість та обґрунтованість завдяки обробці великої кількості інформації.

Аналогами розробки є візуальний метод попарних порівнянь – ВМПП. ВМПП – це веб орієнтована система, яка дає можливість провести процедуру попарного порівняння альтернатив. Процедура попарного порівняння допомагає експерту визначити уподобання серед об'єктів в процесі порівняння можливих пар з подальшим упорядкуванням об'єктів на основі отриманих результаті порівняння. Приблизна ціна 3500\$.

Ще одним аналогом є Visual Pairs Of Alternatives – VisPA, мобільний додаток візуального порівняння альтернатив. Цей додаток виконує некритеріальне оцінювання альтернатив з використанням гештальт ранжування, за ціною 5000\$.

Для проведення комерційного та технологічного аудиту залучають не менше 3-х незалежних експертів. Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням п'ятибальної системи оцінювання за 12-ма критеріями, у відповідності із табл. 4.1.

Таблиця 4.1 – Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

Бали (за 5-ти бальною шкалою)					
Критерій	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 роботоздатність продукту в реальних умовах
Ринкові переваги					
2	Багато аналогів на малому ринку	Ринкові п 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 аналогів	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 аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
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	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
Практик на здійсненність					
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

Продовження табл. 4.1

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 виробництві
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

Усі дані по кожному параметру занесено в таблиці 4.2

Таблиця 4.2 – Результати оцінювання комерційного потенціалу розробки

Критерії оцінювання	ПІБ експертів		
	Експерт 1	Експерт 2	Експерт 3
	Бали		
Технічна здійсненність концепції	4	4	4
Наявність аналогів на ринку	3	3	2
Цінова політика	4	4	4
Технічні та споживчі властивості виробу	3	3	4
Експлуатаційні витрати	3	4	4
Ринок збуту	4	3	4
Конкурентоспроможність	3	2	4
Фахівці з технічної і комерційної реалізації	4	3	4

Продовження табл. 4.2

Фінансування	4	4	3
Матеріально-технічна база	2	4	3
Термін реалізації ідеї	4	4	3
Супровідна документація	4	3	4
Сума	42	41	43
Середньоарифметична сума балів	$(42+41+43) / 3 = 42$		

За даними таблиці 4.2 можна зробити висновок щодо рівня комерційного потенціалу даної розробки. Для цього доцільно скористатись рекомендаціями, наведеними в таблиці 4.3.

Таблиця 4.3 - Рівні комерційного потенціалу розробки

Середньоарифметична сума балів СБ , розрахована на основі висновків експертів	Рівень комерційного потенціалу розробки
0 - 10	Низький
11 -20	Нижче середнього
21 -30	Середній
31 -40	Вище середнього
41 -48	Високий

Як видно з таблиці, рівень комерційного потенціалу розроблюваного нового програмного продукту є високим, що досягається за рахунок того, що удосконалено процес проведення колективних онлайн експертиз за рахунок можливості створювати персоналізовану шкалу оцінювання для поточної експертизи, що позитивно вплине на точність результатів. Створено веб орієнтовану інформаційну технологію проведення колективних онлайн експертиз, що забезпечує підвищення мобільності та зручності використання програмного засобу.

4.2 Прогнозування витрат на виконання науково-дослідної (дослідно-конструкторської) роботи

4.2.1 Основна заробітна плата розробників, яка розраховується за формулою:

$$Z_o = \frac{M}{T_p} \cdot t, \quad (4.1)$$

де М – місячний посадовий оклад конкретного розробника (дослідника), грн.;

T_p – число робочих днів в місяці, 23 днів;

t – число днів роботи розробника (дослідника).

Результати розрахунків зведемо до таблиці 4.1.

Таблиця 4.1 – Основна заробітна плата розробників

Найменування посади	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи	Витрати на заробітну плату, грн.
Керівник проекту	30000	1304,35	40	52173,913
Інженер	27000	1173,91	40	46956,522
Всього				99130,43

Так як в даному випадку розробляється програмний продукт, то розробник виступає одночасно і основним робітником, і тестувальником розроблюваного програмного продукту.

4.2.2 Додаткова заробітна плата розробників, які приймали участь в розробці обладнання.

Додаткова заробітна плата прийнято розраховувати як 12 % від основної заробітної плати розробників та робітників:

$$Z_d = Z_o \cdot 12 \% / 100 \% \quad (4.2)$$

$$Z_d = (99130,43 \cdot 12 \% / 100 \%) = 11895,65 \text{ (грн.)}$$

4.2.3 Нарахування на заробітну плату розробників.

Згідно діючого законодавства нарахування на заробітну плату складають 22 % від суми основної та додаткової заробітної плати.

$$H_3 = (Z_o + Z_d) \cdot 22 \% / 100\% \quad (4.3)$$

$$H_3 = (99130,43 + 11895,65) \cdot 22 \% / 100 \% = 24425,74 \text{ (грн.)}$$

4.2.4. Оскільки для розроблювального пристрою не потрібно витратити матеріали та комплектуючі, то витрати на матеріали і комплектуючі дорівнюють нулю.

4.2.5 Амортизація обладнання, яке використовувалось для проведення розробки.

Амортизація обладнання, що використовувалось для розробки в спрощеному вигляді амортизація обладнання, що використовувалась для розробки розраховується за формулою:

$$A = \frac{Ц}{T_{в}} \cdot \frac{t_{вик}}{12} \text{ [грн.]} \quad (4.4)$$

де Ц – балансова вартість обладнання, грн.;

T – термін корисного використання обладнання згідно податкового законодавства, років

$t_{вик}$ – термін використання під час розробки, місяців

Розрахуємо, для прикладу, амортизаційні витрати на комп'ютер балансова вартість якого становить 24000 грн., термін його корисного використання згідно податкового законодавства – 2 роки, а термін його фактичного використання – 1,74 міс.

$$A_{обл} = \frac{24000}{2} \times \frac{1,74}{12} = 1739,13 \text{ грн.}$$

Аналогічно визначаємо амортизаційні витрати на інше обладнання та приміщення. Розрахунки заносимо до таблиці 4.2. Для розрахунку амортизації нематеріальних ресурсів використовується формула:

$$A_{н.р.} = Ц_{н.р.} * H_a * \frac{t_{вик.}}{12} \quad (4.5)$$

Норму амортизації H_a приймемо за 15 %.

Таблиця 4.2 – Амортизаційні відрахування матеріальних і нематеріальних ресурсів для розробників

Найменування обладнання	Балансова вартість, грн.	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн.
Комп'ютер (Lenovo Ideapad I340)	24000	2	1,74	1739,130
Приміщення	720000	20	1,74	5217,391
Ліцензійна ОС, та спеціалізовані ліцензійні нематеріальні ресурси	6350	-	1,74	138,043
Всього				7094,57

4.2.6 Тарифи на електроенергію для непобутових споживачів (промислових підприємств) відрізняються від тарифів на електроенергію для населення. При цьому тарифи на розподіл електроенергії у різних постачальників (енергорозподільних компаній), будуть різними. Крім того, розмір тарифу залежить від класу напруги (1-й або 2-й клас). Тарифи на розподіл електроенергії для всіх енергорозподільних компаній встановлює Національна комісія з регулювання енергетики і комунальних послуг (НКРЕКП). Витрати на силову електроенергію розраховуються за формулою:

$$V_e = V \cdot П \cdot \Phi \cdot K_{п}, \quad (4.6)$$

де V – вартість 1 кВт-години електроенергії з ПДВ для 1 класу підприємства, $V = 2,01$ грн./кВт;

Π – встановлена потужність обладнання, кВт. $\Pi = 0,7$ кВт;

Φ – фактична кількість годин роботи обладнання, годин.

K_{Π} – коефіцієнт використання потужності, $K_{\Pi} = 0,9$.

$$B_e = 0,9 \cdot 0,7 \cdot 8 \cdot 40 \cdot 2,01 = 405,216 \text{ (грн.)}$$

4.2.7 Інші витрати та загальновиробничі витрати.

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» розраховуються як 50...100% від суми основної заробітної плати дослідників:

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

де H_{ib} – норма нарахування за статтею «Інші витрати».

$$I_e = 99130,43 \cdot 125\% / 100\% = 123913 \text{ (грн.)}$$

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» розраховуються як 100...150% від суми основної заробітної плати дослідників:

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

де $H_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати».

$$H_{нзв} = 99130,43 * 140 \% / 100 \% = 138783 \text{ (грн.)}$$

4.2.8 Витрати на проведення науково-дослідної роботи.

Сума всіх попередніх статей витрат дає загальні витрати на проведення науково-дослідної роботи:

$$\begin{aligned} B_{заг} &= 99130,43 + 11895,65 + 24425,74 + 7094,57 + 405,22 + 123913 + 138783 = \\ &= 405647,26 \text{ грн.} \end{aligned}$$

4.2.11 Розрахунок загальних витрат на науково-дослідну (науково-технічну) роботу та оформлення її результатів.

Загальні витрати на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються ZB , визначається за формулою:

$$ZB = \frac{B_{заг}}{\eta} \text{ (грн)}, \quad (4.9)$$

де η – коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи.

Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то $\eta=0,1$; технічного проектування, то $\eta=0,2$; розробки конструкторської документації, то $\eta=0,3$; розробки технологій, то $\eta=0,4$; розробки дослідного зразка, то $\eta=0,5$; розробки промислового зразка, то $\eta=0,7$; впровадження, то $\eta=0,9$. Оберемо $\eta = 0,7$, так як розробка, на даний момент, знаходиться на стадії промислового зразка:

$$ZB = 405647,26 / 0,7 = 579496 \text{ грн.}$$

4.3 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором

В ринкових умовах узагальнювальним 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. Саме зростання чистого прибутку забезпечить потенційному інвестору надходження додаткових коштів, дозволить покращити фінансові результати його діяльності, підвищить конкурентоспроможність та може позитивно вплинути на ухвалення рішення щодо комерціалізації цієї розробки.

Для того, щоб розрахувати можливе зростання чистого прибутку у потенційного інвестора від можливого впровадження науково-технічної розробки необхідно:

а) вказати, з якого часу можуть бути впроваджені результати науково-технічної розробки;

б) зазначити, протягом скількох років після впровадження цієї науково-технічної розробки очікуються основні позитивні результати для потенційного інвестора (наприклад, протягом 3-х років після її впровадження);

в) кількісно оцінити величину існуючого та майбутнього попиту на цю або аналогічні чи подібні науково-технічні розробки та назвати основних суб'єктів (зацікавлених осіб) цього попиту;

г) визначити ціну реалізації на ринку науково-технічних розробок з аналогічними чи подібними функціями.

При розрахунку економічної ефективності потрібно обов'язково враховувати зміну вартості грошей у часі, оскільки від вкладення інвестицій до отримання прибутку минає чимало часу. При оцінюванні ефективності інноваційних проектів передбачається розрахунок таких важливих показників:

- абсолютного економічного ефекту (чистого дисконтованого доходу);
- внутрішньої економічної дохідності (внутрішньої норми дохідності);
- терміну окупності (дисконтованого терміну окупності).

Аналізуючи напрямки проведення науково-технічних розробок, розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором можна об'єднати, враховуючи визначені ситуації з відповідними умовами.

4.3.1 Розробка чи суттєве вдосконалення програмного засобу (програмного забезпечення, програмного продукту) для використання масовим споживачем.

В цьому випадку майбутній економічний ефект буде формуватися на основі таких даних:

$$\Delta\Pi_i = (\pm\Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.10)$$

де $\pm\Delta C_o$ – зміна вартості програмного продукту (зростання чи зниження) від впровадження результатів науково-технічної розробки в аналізовані періоди часу;

N – кількість споживачів які використовували аналогічний продукт у році до впровадження результатів нової науково-технічної розробки;

C_o – основний оціночний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки, $C_o = C_b \pm \Delta C_o$;

C_b – вартість програмного продукту у році до впровадження результатів розробки;

ΔN – збільшення кількості споживачів продукту, в аналізовані періоди часу, від покращення його певних характеристик;

λ – коефіцієнт, який враховує сплату податку на додану вартість. Ставка податку на додану вартість дорівнює 20%, а коефіцієнт $\lambda = 0,8333$.

ρ – коефіцієнт, який враховує рентабельність продукту;

ϑ – ставка податку на прибуток, у 2021 році $\vartheta = 18\%$.

Припустимо, що при прогнозованій ціні 38000 грн. за одиницю виробу, термін збільшення прибутку складе 3 роки. Після завершення розробки і її вдосконалення, можна буде підняти її ціну на 1000 грн. Кількість одиниць реалізованої продукції також збільшиться: протягом першого року – на 150 шт.,

протягом другого року – на 400 шт., протягом третього року на 700 шт. До моменту впровадження результатів наукової розробки реалізації продукту не було:

$$\Delta\Pi_1 = (0*1000 + (38000 + 1000)*150)*0,8333*0,38*(1 - 0,18) = 1480099,941 \text{ грн.}$$

$$\Delta\Pi_2 = (0*1000 + (38000 + 1000)*(150+400))*0,8333*0,38*(1 - 0,18) = 5569849,777 \text{ грн.}$$

$$\Delta\Pi_3 = (0*1000 + (38000 + 1000)*(150+400+700))*0,8333*0,38*(1 - 0,18) = 12658749,494 \text{ грн.}$$

Отже, комерційний ефект від реалізації результатів розробки за три роки складе 19708699,21 грн.

4.3.2 Розрахунок ефективності вкладених інвестицій та періоду їх окупності.

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

$$\Pi\Pi = \sum_1^T \frac{\Delta\Pi_i}{(1+\tau)^t}, \quad (4.11)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої науково-дослідної (науково-технічної) роботи, грн;

T – період часу, протягом якою виявляються результати впровадженої науково-дослідної (науково-технічної) роботи, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,05 \dots 0,15$;

t – період часу (в роках).

Збільшення прибутку ми отримаємо починаючи з першого року:

$$\text{ПП} = (1480099,941/(1+0,1)^1)+(5569849,777/(1+0,1)^2)+(12658749,494/(1+0,1)^3) = 1345545,40 + 4603181,63 + 9510705,86 = 15459432,89 \text{ грн.}$$

Далі розраховують величину початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки. Для цього можна використати формулу:

$$PV = k_{инв} * ZB, \quad (4.12)$$

де $k_{инв}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові заходи тощо; зазвичай $k_{инв}=2...5$, але може бути і більшим;

ZB – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

$$PV = 2 * 579496 = 1158992,17 \text{ грн.}$$

Тоді абсолютний економічний ефект $E_{абс}$ або чистий приведений дохід (NPV , *Net Present Value*) для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{абс} = \text{ПП} - PV, \quad (4.13)$$

$$E_{абс} = 15459432,89 - 1158992,17 = 14300440,72 \text{ грн.}$$

Оскільки $E_{абс} > 0$ то вкладання коштів на виконання та впровадження результатів даної науково-дослідної (науково-технічної) роботи може бути доцільним.

Для остаточного прийняття рішення з цього питання необхідно розрахувати внутрішню економічну дохідність або показник внутрішньої норми дохідності (*IRR, Internal Rate of Return*) вкладених інвестицій та порівняти її з так званою бар'єрною ставкою дисконтування, яка визначає ту мінімальну внутрішню економічну дохідність, нижче якої інвестиції в будь-яку науково-технічну розробку вкладати буде економічно недоцільно.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій E_e . Для цього використаємо формулу:

$$E_e = \sqrt[T_{ж}]{1 + \frac{E_{abc}}{PV}} - 1, \quad (4.14)$$

$T_{ж}$ – життєвий цикл наукової розробки, роки.

$$E_e = \sqrt[3]{(1 + 14300440,72/1158992,17) - 1} = 1,372$$

Визначимо мінімальну ставку дисконтування, яка у загальному вигляді визначається за формулою:

$$\tau = d + f, \quad (4.15)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2021 році в Україні $d = (0,09...0,14)$;

f – показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05...0,5)$.

$$\tau_{\min} = 0,14 + 0,05 = 0,19.$$

Так як $E_e > \tau_{\min}$, то інвестор може бути зацікавлений у фінансуванні даної наукової розробки.

Розрахуємо термін окупності вкладених у реалізацію наукового проекту інвестицій за формулою:

$$T_{ок} = \frac{1}{E_г}, \quad (4.16)$$

$$T_{ок} = 1 / 1,372 = 0,73 \text{ р.}$$

Оскільки $T_{ок} < 3$ -х років, а саме термін окупності рівний 0,73 роки, то фінансування даної наукової розробки є доцільним.

4.4 Висновок

Економічна частина даної роботи містить розрахунок витрат на розробку нового програмного продукту, сума яких складає 579496 гривень. Було спрогнозовано орієнтовану величину витрат по кожній з статей витрат.

Також розраховано чистий прибуток, який може отримати виробник від реалізації нового технічного рішення, розраховано період окупності витрат для інвестора та економічний ефект при використанні даної розробки.

В результаті аналізу розрахунків можна зробити висновок, що розроблений програмний продукт за ціною дешевший за аналог і є висококонкурентоспроможним. Період окупності складе близько 0,73 роки.

ВИСНОВКИ

У магістерській кваліфікаційній роботі проведено дослідження, що присвячене проблемі прийняття колективних рішень в умовах віддаленої комунікації та розроблено інформаційну технологію проведення колективних онлайн експертиз. Проаналізувавши предметну область визначено, що задача яка полягає в розробці інформаційної технології проведення колективних онлайн експертиз є актуальною. За рахунок віддаленого доступу це забезпечить користувачам швидке та зручне проведення експертизи для великої кількості експертів та альтернатив.

Виконано проектування та розробку програмного забезпечення інформаційної технології. Опираючись на поставлені задачі, визначено структуру програмного продукту, розроблено та описано алгоритм роботи інформаційної технології. Виконано обґрунтування вибору мови програмування JavaScript, основними її перевагами є: універсальність, швидкість, кросплатформеність та сумісність.

Проведено тестування розробленої інформаційної системи. Тестування підтвердило, що система працює відповідно до поставлених завдань. Порівнюючи розроблену інформаційну технологію з аналогами, визначено що функціонал значно збагатився за рахунок додавання можливості створення нових експертиз; можливості редагування шкали оцінювання; можливості одночасного відображення результатів порівняння. Збагачення функціоналу позитивно впливало на підвищення підвищення ефективності процесу проведення колективних онлайн експертиз на 28.5% порівнянно з системами-аналогами. Це підтверджує досягнення мети магістерської кваліфікаційної роботи.

Наукова новизна результатів дослідження полягає в розробленій динамічну архітектурі інформаційної технології та удосконаленому процесі проведення колективних онлайн експертиз за рахунок можливості створювати персоналізовану шкалу оцінювання для поточної експертизи, що позитивно впливає на точність результатів. Результати досліджень пройшли експериментальне випробовування і впроваджені у в Нараївській загальноосвітній школі I-III ступенів Краснопільської сільської ради Гайсинського району Вінницької області.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мельник С.М. Аналіз підходів до реалізації комп'ютеризованої системи візуалізації колективних переважань експертів [Текст] / Мельник С.М., Петришин С.І., Яцько А.А. // Матеріали конференції «XII Міжнародної науково-практичної конференції «Інтернет-Освіта-Наука-2020»: Тез. доп. – Вінниця, 2020. – с.2.

2. Мельник С.М. Аналіз особливостей інформаційної технології візуалізації колективних переважань експертів CollExpert5 [Текст] / Мельник С.М., Колодний В.В. // Матеріали конференції «L Науково-технічна конференція підрозділів Вінницького національного технічного університету (2021)»: Тез. доп. – Вінниця, 2021. – с.2.

3. Мельник С.М. Розробка клієнтської частини клієнт-серверної системи візуалізації колективної експертизи CollExpert5 [Текст] / Мельник С.М., Петришин С.І., Колодний В.В. // Матеріали конференції «XLIX Науково-технічна конференція факультету інформаційних технологій та комп'ютерної інженерії (2020)»: Тез. доп. – Вінниця, 2020. – с.2.

4. Колодний В.В. Комп'ютерна програма «Система візуалізації колективної експертизи CollExpert5» / В.В. Колодний, С.М. Мельник, А.А. Яцько // Свідоцтво про реєстрацію авторського права на твір №96492. – Рішення від 03.03.2020.

5. Колодний В. В. Некритеріальне оцінювання альтернатив/ В. В. Колодний, В. В. Зубко // «ІНТЕРНЕТОСВІТА-НАУКА-2016»: Збірник матеріалів конференції . Вінниця : ВНТУ, 2016. - С. 43-44.

6. L. Bianchi, L. M. Gambardella et M. Dorigo, An ant colony optimization approach to the probabilistic traveling salesman problem, PPSN-VII, Seventh International Conference on Parallel Problem Solving from Nature, Lecture Notes in Computer Science, Springer Verlag, Berlin, Allemagne, 2002.

7. Data visualization beginner's guide: a definition, examples, and learning resources [Електронний ресурс] – Режим доступу до ресурсу: <https://www.tableau.com/learn/articles/data-visualization>.

8. The Top 10 Types of Data Visualization Made Simple [Електронний ресурс] – Режим доступу до ресурсу: <https://boostlabs.com/10-types-of-data-visualization-tools/#The-15-Most-Common-Types-of-Data-Visualization-Formats>.

9. Юхимчук С.В., Колодний В.В., Зарезенко Д.П. Програмна реалізація системи підтримки прийняття рішень, що базується на методі дискретно-неперервного аналізу розподілу корисності. - Вісник ВПІ. – 2009. – №2. – С. 70-77.

10. Девятков С.С. Проектирование программного обеспечения с использованием стандартов UML 2.0 и SysML 1.0 / С.С. Девятков // 115 Прикладная информатика. – М.: Негосударственное образовательное частное учреждение высшего образования Московский финансово промышленный университет Синергия. – 2006. – №6 2– С. 48-63.

11. Berson A. Client/server architecture / Alex Berson. – New York, NY: McGraw-Hill, Inc. Professional Book Group, 1992. – 452 с.

12. S. Iredi, D. Merkle et M. Middendorf, Bi-Criterion Optimization with Multi Colony Ant Algorithms, Evolutionary Multi-Criterion Optimization, First International Conference (EMO'01), Zurich, Springer Verlag, pages 359 — 372, 2001.

13. Липаев В.В. Надежность и функциональная безопасность комплексов программ реального времени / В.В. Липаев. Институт системного программирования Российской академии наук – М: 2013. 176 с.

14. Гусятников В.Н. Стандартизация и разработка программных систем / В.Н. Гусятников, А.И. Безруков – М: Финансы и статистика, 2010, 288 с.

15. Колодний В. В. Мобільний застосунок візуального порівняння альтернатив VisPA / В. В. Колодний, П. О. Зоря // “Інформаційні технології та комп’ютерна інженерія”. Вінниця: ВНТУ, 2019. - С.3

16. Розробка інтелектуальних систем підтримки прийнятих рішень для діагностики, керування та оптимізації технічних та біологічних об’єктів: звіт про НДР (заключний) / Наук. кер. В.Д. Дмитрієнко. – Х.:НТУ «ХП», 2012

17. PHP (Hypertext Preprocessor) Definition [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://techterms.com/definition/php>.

18. Solutions M. Advantages and Disadvantages of PHP Frameworks [Електронний ресурс] / Mindfire Solutions. – 2018. – Режим доступу до ресурсу: <https://medium.com/@mindfiresolutions.usa/advantages-and-disadvantages-of-php-frameworks-c046d50754e5>.

19. TechTarget Contributor. What is Ruby? - Definition from WhatIs.com [Електронний ресурс] / TechTarget Contributor. – 2019. – Режим доступу до ресурсу: <https://whatis.techtarget.com/definition/Ruby>.

20. Nick Gorbikoff. What are the advantages and disadvantages of Ruby? [Електронний ресурс] / Nick Gorbikoff. – 2018. – Режим доступу до ресурсу: <https://www.quora.com/What-are-the-advantages-and-disadvantages-of-Ruby>.

21. What is JavaScript? JavaScript - Overview. [Електронний ресурс] – Режим доступу до ресурсу: https://www.tutorialspoint.com/javascript/javascript_overview.htm.

22. An Introduction to JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <https://javascript.info/intro>.

23. Колодний В. В. Інформаційна технологія виявлення переважань в неструктурованих задачах прийняття рішень / В. В. Колодний, В. В. Зубко // “Інформаційні технології та комп’ютерна інженерія”. Вінниця: ВНТУ, 2015. - С.3.

24. Marshinov. Юнит-тестирование для чайников [Електронний ресурс] / Marshinov. – 2013. – Режим доступу до ресурсу: <https://habr.com/ru/post/169381/>.

25. Препроцессор Sass. Полное руководство и зачем он нужен [Електронний ресурс] // Stas Bagretsov. – 2018. – Режим доступу до ресурсу: <https://medium.com/@stasonmars/%D0%BF%D1%80%D0%B5%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80-sass>.