

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

Факультет інформаційних технологій
Кафедра інтелектуальних технологій

**ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
БАКАЛАВРА
НА ТЕМУ:**

Система підтримки дистанційного проведення конкурсів

Галузь знань **12 «Інформаційні технології»**

Спеціальність **122 «Комп'ютерні науки»**

Освітня програма **«Аналітика даних»**

Освітній рівень: бакалавр

Виконала: студентка 4 курсу, групи Анд-41

_____ Кузьменко А.О. _____
(прізвище та ініціали)

Керівник _____ Циганок В.В. _____
(прізвище та ініціали)
_____ проф., д.т.н., с.н.с. _____
(науковий ступінь, звання)

Випускна кваліфікаційна робота бакалавра допущена до захисту
рішенням кафедри *інтелектуальних технологій*
Протокол №_13__від_05.06.2023 р.
зав. кафедра _доц. Іларіонов О.Є.

Київ - 2023

АНОТАЦІЯ

Кузьменко Анастасія Олегівна виконала випускню кваліфікаційну роботу на тему “Система підтримки дистанційного проведення конкурсів” за спеціальністю 122 - “Комп’ютерні науки”, освітня програма - “Аналітика даних”.

У випускній кваліфікаційній роботі проведено аналіз наявних проблем при проведенні конкурсів, розроблено веб-орієнтований додаток для дистанційного проведення конкурсів, який забезпечує визначення ваг критеріїв для оцінювання робіт, а також, проводити оцінювання членам журі за сформованою множиною критеріїв. Використовуються лише відносні оцінки проектів щодо інших прослуханих. Застосування багатокритеріального оцінювання дозволяє швидко визначити переможців конкурсу та звести до мінімуму можливість маніпуляцій у ході оцінювання.

Ключові слова: дистанційні конкурси, підтримка прийняття рішень, відносні оцінки, багатокритеріальне групове оцінювання, веб-орієнтований додаток.

Summary

The degree project: «Support system for remote conducting of contests» has completed by Kuzmenko Anastasiia specialty 122 – «Computer Sciences», educational program - "Data Analytics".

In this graduation existing problems of remote conducting of competitions are analyzed, a web application for remote conducting of competitions was developed, which ensures the determination of criteria weights for evaluation of works, as well as evaluation of jury members based on a set of criteria. Only the relative evaluations of the projects in relation to the other auditioned ones are used. The use of multi-criteria evaluation allows you to quickly determine the winners of the competition and minimize the possibility of manipulation during the evaluation.

Keywords: remote competitions, decision-making support, relative estimations, multi-criteria group evaluation, web-based application.

ЗМІСТ

АНОТАЦІЯ	2
ВСТУП	4
РОЗДІЛ 1. ОСОБЛИВОСТІ ЗАДАЧІ ДИСТАНЦІЙНОГО ПРОВЕДЕННЯ КОНКУРСІВ	7
1.1 Технології супроводу проведення конкурсів, що застосовуються	7
1.1.1 Теперішній процес проведення конкурсу	7
1.1.2 Недоліки наявних технологій супроводу конкурсів	7
1.1.3 Планування проектування програмної системи позбавленої недоліків притаманних наявним системам	9
1.2 Пропозиції щодо вибору засобів розроблення додатку для супроводження дистанційних конкурсів	9
1.3 Пропозиції щодо вибору методів, необхідних для реалізації технології проведення дистанційних конкурсів	10
1.3.1 Метод аналізу ієрархій	10
1.3.2 Метод Борда	13
1.4 Вимоги до розробки та постановка задачі проектування	14
РОЗДІЛ 2. РОЗРОБКА СИСТЕМИ УПРАВЛІННЯ БАЗОЮ ДАНИХ ТА ОПИС РОЗРОБЛЕНОГО ВЕБ-ДОДАТКУ	16
2.1 Архітектура розробленого веб-додатку	16
2.2 Інформаційне забезпечення. Аналіз потоків даних та розробка системи управління базою даних	19
2.3 Програмне забезпечення	24
2.4 Реалізація методу аналізу ієрархій	25
2.5 Висновки до другого розділу	28
РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ ПІДТРИМКИ ПРОВЕДЕННЯ КОНКУРСІВ	29
3.1 Опис використаних бібліотек	29
3.2 Специфікація програмного модулю та опис функцій	30
3.3 Підключення методу аналізу ієрархій	32
3.3 Опис програмної реалізації	32
3.4 Результат тестування розробленої системи	35
3.5 Аналіз тестування розробленої системи підтримки	44
ВИСНОВКИ	47
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	49
ДОДАТОК А	51

ВСТУП

В сучасному світі особливо після появи коронавірусу з'явилася необхідність в переході усіх можливих заходів в дистанційний режим. Розглядаючи теперішню ситуацію в Україні, тобто повномасштабну війну, необхідність переходу проведення конкурсів у дистанційний режим є дуже велика. Є багато людей, які до початку війни проживали на території України та хочуть прийняти участь в конкурсі або ж є експерти, які були змушені змінити своє місце проживання на закордон. Також через загрозу обстрілів, більшість людей, які також приймають участь в конкурсі або ж є експертами, прагнуть залишатися вдома. Тому було прийняте рішення розробити систему підтримки дистанційного проведення конкурсів та створити веб-орієнтований додаток. Виходячи з вищепереліченого, актуальність теми кваліфікаційної роботи зумовлена тим, що створення системи підтримки дистанційного проведення конкурсів є забезпечення безпеки учасників та/або журі конкурсу в період повномасштабної війни, забезпечення ефективного проведення конкурсу. Також, розроблена система допоможе уникнути урахувань недостовірних оцінок та можливих маніпуляцій зі сторони членів журі при оцінюванні, що призводять до викривлення результатів конкурсів.

Провівши аналіз наукових джерел та інформації від учасників конкурсів, членів журі та апеляційних комісій, було виявлено недоліки у наявній системі проведення конкурсів, що викликало необхідність розробки.

Об'єктом дослідження є процес дистанційного проведення конкурсів. Об'єкт дослідження передбачає дослідження і аналіз особливостей, викликів та можливостей, що пов'язані з організацією та проведенням конкурсів в онлайн-середовищі. Досліджується, які аспекти потрібно враховувати при перенесенні конкурсів у віртуальне середовище, як забезпечити об'єктивність, взаємодію та безпеку учасників, а також ефективність процесу оцінювання та визначення переможців.

Предметом дослідження є розробка веб-додатку для підтримки дистанційного проведення конкурсів. Предмет дослідження включає розробку програмного забезпечення, яке забезпечує віртуальне проведення конкурсів, включаючи реєстрацію учасників, надання можливостей для представлення конкурсних матеріалів, організацію процесу оцінювання та визначення переможців.

Метою випускної кваліфікаційної роботи є розробка веб-орієнтованого додатку для дистанційного проведення конкурсів, з уникненням всіх наявних недоліків наявних технологій. Така програмна система має сприяти зручності, доступності та ефективності проведення конкурсів у віддалених умовах, а також об'єктивізувати результати конкурсів та підвищити рівень довіри суспільства до них.

Створення веб-додатків стає більш популярним, через те що економить час на реалізацію, спрямоване на полегшення роботи. Тому використання веб-додатку для дистанційного проведення конкурсів на даний момент є актуальною темою.

Досягнення поставленої мети передбачає реалізацію таких завдань:

- 1) розглянути наявні системи та технології проведення конкурсів у дистанційному режимі;
- 2) визначити принцип роботи та наявні недоліки відповідних систем;
- 3) здійснити порівняльний аналіз систем та платформ для проведення дистанційних конкурсів ;
- 4) проаналізувати наявні методи, що застосовуються при реалізації системи дистанційного проведення конкурсів та обрати необхідні з них для застосування;
- 5) створити веб-орієнтований додаток для дистанційного проведення конкурсів, який буде надавати різний функціонал в залежності від ролі користувача у конкурсі. Для учасників конкурсу: завантажувати конкурсні роботи та переглядати рейтинг з оцінками та місцями,

для експертів: перегляд завантажених конкурсних робіт, оцінювання робіт та перегляд рейтингу.

РОЗДІЛ 1.ОСОБЛИВОСТІ ЗАДАЧІ ДИСТАНЦІЙНОГО ПРОВЕДЕННЯ КОНКУРСІВ

1.1 Технології супроводу проведення конкурсів, що застосовуються

1.1.1 Теперішній процес проведення конкурсу

На теперішній час конкурси здебільшого проходять наступним чином:

- конкурсна комісія заздалегідь складає умови конкурсу та колективно формує множину критеріїв для оцінювання проектів;
- комісія оголошує умови проведення конкурсу проектів (оприлюднені умови, як правило, включають строки проведення, категорії майбутніх учасників, правила подання, регламент, тематики, бажані результати, строки виконання проектів тощо)
- на призначену дату конкурсу формується склад журі, яким надаються повноваження оцінювати кожний поданий на конкурс проект за сформованою множиною критеріїв
- у ході конкурсу заслуховуються презентації учасників конкурсу
- кожен з членів журі після закінчення конкурсу надає і оприлюднює свої оцінки кожного проекту за заданими критеріями
- оцінки всіх членів журі узагальнюються і отримується рейтинг проектів, на основі якого і визначаються переможці конкурсу.

Незважаючи на, здавалось би, простий та надійний спосіб традиційного проходження конкурсів, цій технології притаманний цілий ряд недоліків.

1.1.2 Недоліки наявних технологій супроводу конкурсів

Ряд характерних недоліків присутні в традиційних технологіях переважно через приділення недостатньої уваги як організації конкурсу, так і методам, що покладаються в основу прийняття рішень щодо оцінювання та відбору переможців.

Розглянемо докладніше виявлені недоліки у відповідності з ходом проведення традиційного конкурсу:

- при складанні множини критеріїв для оцінювання конкурсних робіт часто це робиться суб'єктивно лише одиницями або, навіть однією особою із організаторів конкурсу, що призводить до порушення відкритості конкурсу та ставить під сумнів справедливість результатів;
- сформована множина критеріїв не перевіряється на взаємну незалежність, і у випадку наявних залежних критеріїв призводить до порушень при узагальненні;
- критерії, що пропонуються для оцінювання журі, у переважній більшості конкурсів вважаються рівними за вагою, що призводить до того, що результуючі оцінки не повністю відображають якість конкурсної роботи і може призвести до викривлення результатів конкурсу;
- оцінювання, зазвичай, членами журі проводиться не анонімно, що призводить до небажаних впливів на оцінки признаних авторитетів серед спільноти (так званий “ефект диктатора”);
- членам журі пропонується оцінювати за критеріями усі подані на конкурс роботи, навіть якщо член журі не прослуховував презентацію конкурсної роботи і, як наслідок, це призводить до недостовірної оцінки;
- оцінювання робіт за критеріями, зазвичай пропонується виконувати у бальній шкалі (максимальна кількість балів пропонується, часто, не обґрунтованою). Це призводить до можливості маніпуляцій при оцінюванні недоброчесними членами журі, коли кількість максимум кількості балів обрано невідповідним до кількості членів журі і в такому випадку один член журі необґрунтовано завищивши або заниживши бальну оцінку може знівелювати думку усієї чисельної групи журі.

1.1.3 Планування проектування програмної системи позбавленої недоліків притаманних наявним системам

Для уникнення вище описаних недоліків при розробці веб-орієнтованого додатку було вирішено:

- Зробити оцінювання конкурсних робіт членами журі повністю анонімним, що приводить до неможливості небажаних впливів на підсумкові оцінки;
- Оцінювання роботи відбувається за допомогою методу Борда, що виключає необґрунтоване заниження або завищення оцінок. Максимальна кількість “балів” відповідає кількості членів журі;
- Член журі не має можливості оцінювати роботу, яку не встиг прослухати або ознайомитись з роботою в інший спосіб у ході проведення конкурсу, , крім того, при наявності конфлікту інтересів з авторами оцінюваної роботи;
- У сформованій множині критеріїв самі критерії не є взаємозалежними між собою, що виключає викривлення при агрегації з застосуванням середнього результатів конкурсу;
- Використати метод аналізу ієрархій для можливості визначити ваги множини критеріїв для оцінювання конкурсних робіт.

1.2 Пропозиції щодо вибору засобів розроблення додатку для супроводження дистанційних конкурсів

Задля уникнення наявних недоліків організації дистанційних конкурсів пропонується технологія, втілена в програмний застосунок, яка дозволить значно підвищити організаційний рівень проведення конкурсів та збільшити рівень довіри до їх результатів.

1.2.1 Вибір виду програмних систем, що пропонуються для застосування

Для розробки системи підтримки конкурсів було обрано створення веб-орієнтованого додатку. Існує декілька причин вибору веб-орієнтованого додатка:

- 1) простота доступу до додатка. Користувач, який має комп'ютер, підключений до мережі Інтернет, може використовувати web-додаток;
- 2) вся особиста інформація користувачів, їхні дані знаходяться у хмарному сховищі. Їх захист, збереження забезпечує хостинг-провайдер;
- 3) можливістю роботи через браузер, який є у кожного користувача;
- 4) простота створення.

1.3 Пропозиції щодо вибору методів, необхідних для реалізації технології проведення дистанційних конкурсів

Охарактеризуємо наявні методи, що використовуються при проведенні конкурсів.

1.3.1 Метод аналізу ієрархій

У випускній кваліфікаційній роботі використовується метод аналізу ієрархій для визначення рейтингу робіт, поданих на конкурс за заданими критеріями. Метод аналізу ієрархій [6] - це математичний інструмент системного підходу до вирішення складних проблем прийняття рішень. Метод був розроблений відомим американським математиком Томасом Сааті. Метод аналізу ієрархій застосовується для розв'язання задач, що передбачають декомпозицію з використанням ієрархій та синтез шляхом знаходження відношення через судження. У загальному вигляді постановка задачі, що розв'язується включає мету, альтернативи та критерії оцінки альтернатив, тоді потрібно обрати найкращу альтернативу. Після побудови ієрархічної структури (мети – критеріїв – альтернатив) за допомогою парних порівнянь елементів

кожного рівня ми можемо отримати результат, який може бути представлений у вигляді обернено симетричної матриці (матриці парних порівнянь альтернатив), елемент якої a_{ij} характеризує інтенсивність прояву альтернативи (як елемента ієрархії) і щодо альтернативи j у сенсі вибраного фіксованого критерію. Суть методу аналізу ієрархій полягає в узагальненні задачі на верхньому рівні і її подальшій деталізації на нижніх рівнях ієрархії. Верхній рівень визначає головні цілі, а нижні рівні – способи формування та методи розбиття елементів попереднього рівня.

Основні етапи методу аналізу ієрархій:

1) Побудова ієрархічної структури задачі.

Розглядаються домінуючі ієрархії, які будуються з вершини (мета - з точки зору управління) через проміжні рівні (критерії, від яких залежать наступні рівні) до найнижчого рівня, який є, зазвичай, переліком альтернатив.

2) Визначення пріоритетів елементів ієрархічної структури.

Наступним етапом аналізу є визначення пріоритетів, що представляють відносну важливість або перевагу елементів побудованої ієрархічної структури, за допомогою процедури парних порівнянь. Для встановлення відносної важливості елементів ієрархії використовується шкала відношень Сааті. Ця шкала надає експерту можливість ставити у відповідність ступеням переваги одного фактору перед іншим (деяке число). На рисунку 1.3 показано шкалу попарних порівнянь Сааті.

Рівень переваги	Визначення	Пояснення
1	Відсутність переваги	Внесок альтернатив до цілі однаковий
2	Слабка перевага	
3	Посередня перевага	Досвід та судження трохи сприяють одній з альтернатив над іншою
4	Більш ніж посередня перевага	
5	Сильна перевага	Досвід та судження сильно сприяють одній з альтернатив над іншою
6	Більш ніж сильна перевага	
7	Дуже сильна або продемонстрована перевага	Перевага дуже сприятлива до однієї з альтернатив, її домінування продемонстровано на практиці
8	Дуже, дуже сильна перевага	
9	Екстремальна перевага	Докази, що сприяють одній з альтернатив над іншою є найвищим можливим порядком підтвердження
1,1 - 1,9	Значення, близькі до відсутності переваги	Коли альтернативи дуже близькі додавання знаків після коми дозволяє показати наявність різниці

Рисунок 1.3 - Шкала попарних порівнянь Сааті

3) Знаходження вектора пріоритетів

Ранжування елементів, що аналізується з використанням матриці попарних порівнянь, здійснюється на підставі аналізу головних векторів матриці попарних порівнянь.

4) Розрахунок пріоритетів альтернатив

Розрахунок пріоритетів альтернатив є знаходження вектору пріоритетів альтернатив за матрицею попарних порівнянь. Альтернатива з найбільшим значенням пріоритету має найбільшу перевагу.

Переваги методу аналізу ієрархій:

- Схема застосування методу абсолютно не залежить від сфери діяльності;
- Даний метод може служити надбудовою для інших методів;
- Метод відображає природний хід людського мислення;
- В рамках методу немає загальних правил для формування структури моделі ухвалення рішення;
- Збирання даних для підтримки прийняття рішення здійснюється головним чином за допомогою процедури парних порівнянь.

Неділоки методу:

- В рамках методу немає засобів для перевірки достовірності даних;
- Метод дає лише спосіб рейтингування альтернатив, але не має внутрішніх засобів для інтерпретації рейтингів.

Таким чином, можна зробити висновок, що аналіз методу ієрархій потрібен в подальшому розробленні веб-орієнтованого додатку для того, щоб визначити найкращу роботу на конкурсі за допомогою попарних порівнянь.

В розробленій системі метод аналізу ієрархій використовується щоб члени журі могли самостійно визначити критерії та ваги множини критеріїв. Застосування методу аналізу ієрархій в розробленому веб-орієнтованому додатку відбувається за допомогою заповнення членами журі Excel файлу. В цьому файлі є всі необхідні таблиці для знаходження вектору пріоритетів альтернатив. Після заповнення членами журі Excel файлу, створена програма підключається до цього файлу та переносить вектор пріоритетів альтернатив з Excel таблиці в створену таблицю бази даних.

1.3.2 Метод Борда

Метод Борда базується на аналізі індивідуальних переваг членів журі та не враховують таких суб'єктивних факторів як здатність виборця виявити кращого для нього кандидата, оптимізм чи негативізм виборця тощо.

Для підрахунку результатів оцінювання конкурсних робіт було обрано метод Борда. Відповідно до цього методу результати голосування виражаються у вигляді числа балів, набраних кожним із учасників конкурсу. Так, при виборах з учасників кожен експерт ранжує всіх учасників за зменшенням переваги, за перше місце за перевагою учаснику конкурсу присуджується n балів, за друге - $n-1$ бал і т. д., за останнє місце - 1 бал), всі набрані бали учасниками підсумовуються. Відповідно, переможцем конкурсу вважається учасник, який набрав найвищий сумарний бал.

Перевагою методу Борда є неможливість попереднього інтуїтивного підрахунку балів.

Недоліком є те що при оцінюванні об'єктам нечислової природи (місцям у порядку переваги) застосовуються арифметичні операції.

В розробленому веб-орієнтованому додатку метод Борда застосовується для оцінювання членами журі конкурсних робіт. Члени журі ранжують всі конкурсні роботи за зменшенням переваги. Після ранжування всі поставлені бали сумуються та переможцем конкурсу буде той, в кого в сумі найвищий бал.

1.4 Вимоги до розробки та постановка задачі проектування

Задача випускної кваліфікаційної роботи полягає у розробці веб-орієнтованого додатку для дистанційного проведення конкурсів, з уникненням всіх наявних помилок. Перш ніж розробляти веб-орієнтований додаток, було визначено системні, функціональні та нефункціональні вимоги.

Темою роботи є система підтримки дистанційного проведення конкурсів.

Метою випускної кваліфікаційної роботи є розробка веб-орієнтованого додатку для дистанційного проведення конкурсів, з уникненням всіх наявних недоліків наявних технологій. Така програмна система має сприяти зручності, доступності та ефективності проведення конкурсів у віддалених умовах, а також об'єктивізувати результати конкурсів та підвищити рівень довіри суспільства до них.

До математичних методів розв'язання у випускній кваліфікаційній роботі відносяться метод аналізу ієрархій та метод Борда. Метод аналізу ієрархій надає можливість визначення ваг множини критеріїв для оцінювання конкурсних робіт. В свою чергу метод Борда виключає необгрунтоване заниження або завищення оцінок.

Системні вимоги — вимоги, які використовуються для опису характеристик, яким повинен відповідати цифровий пристрій (ПК, гральна консоль, мобільний телефон тощо) для коректної роботи певного програмного забезпечення. Цифровий пристрій має мати мінімальні системні вимоги для коректної роботи розробленого веб-орієнтованого додатку. Основна вимога мати будь-який браузер для користування веб-додатком.

Функціональна вимога - це формулювання того, як система повинна поводитися. Він визначає, що повинна робити система, щоб задовольнити потреби або очікування користувача.[7] Функціональні вимоги для розроблення веб-орієнтованого додатку поділені на дві групи: для учасників конкурсу, для експерт, через те що для цих груп повинні бути різні вимоги. Функціональні вимоги для розроблення веб-орієнтованого додатку:

- 1) Для учасників конкурсу:
 - Реєстрація людини;
 - Завантаження файлу конкурсної роботи;
 - Перегляд оцінок всіх учасників;
- 2) Для експертів:
 - Перегляд завантажених конкурсних робіт;
 - Оцінювання робіт;
 - Перегляд оцінок учасників.

Нефункціональні вимоги – це обмеження або вимоги, що накладаються на систему. Вони визначають атрибут якості програмного забезпечення.[8]

Нефункціональні вимоги для розробленого веб-орієнтованого додатку:

- Веб-додаток повинен мати можливість масштабування за потреби;
- Веб-додаток повинен бути захищена від несанкціонованого доступу;
- Простий в використанні;
- Надійність;
- Можливість працювати з веб-додатком у різних браузерах з мінімальними змінами;
- Здатність обслуговувати велику кількість учасників та журі конкурсу без зниження продуктивності самого веб-додатку.

1.5 Висновок до першого розділу

Отже, за дотриманням всіх вище описаних системних, функціональних та нефункціональних вимог, розроблений в подальшому, веб-орієнтований додаток буде працювати належним чином та виконувати всі функції, які будуть необхідні учасникам та експертам конкурсу. Метод аналізу ієрархій використовується для групової декомпозиції узагальненого критерію, формування та визначення ваг множини критеріїв для оцінювання конкурсних робіт. Метод Борда в випускній кваліфікаційній роботі використовується для проведення оцінювання членами журі конкурсних робіт.

РОЗДІЛ 2. РОЗРОБКА СИСТЕМИ УПРАВЛІННЯ БАЗОЮ ДАНИХ ТА ОПИС РОЗРОБЛЕНОГО ВЕБ-ДОДАТКУ

2.1 Архітектура розробленого веб-додатку

Архітектура веб-орієнтованого додатку для системи підтримки дистанційного проведення конкурсів складається з підсистем взаємодії з учасником конкурсу та експертом конкурсу. Також в архітектуру входить база даних, яка складається з таблиці інформації учасників, таблиці оцінок конкурсних робіт та таблиці векторів пріоритетів. Архітектура представлена на рисунку 2.1.

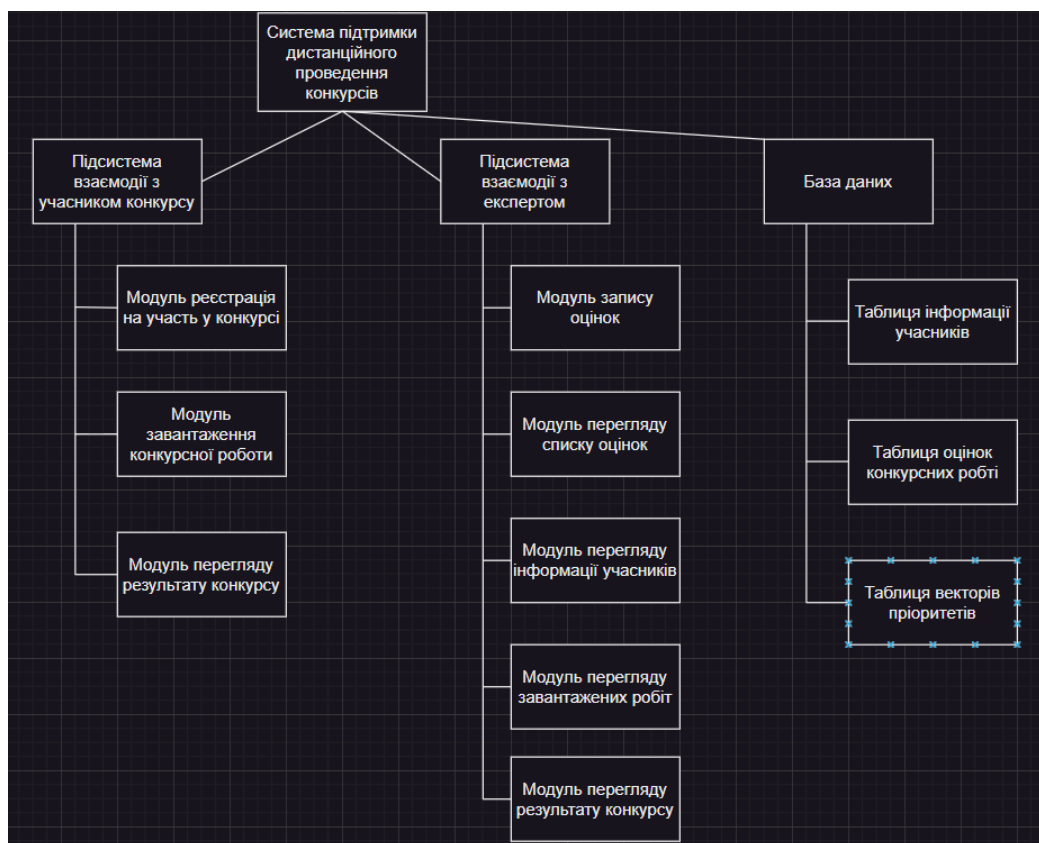


Рисунок 2.1 - Архітектура веб-орієнтованого додатку для системи підтримки дистанційного проведення конкурсів

Підсистема взаємодії з учасником конкурсу та підсистема взаємодії з експертом мають спільну головну сторінку.

До підсистеми взаємодії з учасником конкурсу входить модуль реєстрації на участь у конкурсі, модуль завантаження конкурсної роботи та модуль перегляду результатів конкурсу. Модуль реєстрації на участь у конкурсі надає можливість учасникам конкурсу внести інформацію про себе у форму. Модуль завантаження конкурсної роботи надає можливість учаснику завантажити свою конкурсну роботу, для того щоб експерт міг здійснити її перегляд. Модуль перегляду результатів конкурсу надає учаснику конкурсу можливість переглянути скільки балів набрали всі конкурсні роботи.

До підсистеми взаємодії з експертом наявні такі модулі: модуль запису оцінок, модуль перегляду списку оцінок, модуль перегляду інформації учасників, модуль перегляду завантажених робіт та модуль перегляду результатів конкурсу. Модуль запису оцінок реалізує можливість експертам оцінити конкурсні роботи та записати оцінки в форму. Модуль перегляду оцінок дає змогу експертам переглянути раніше поставлені ними оцінки робіт, які були виставлені на конкурс. Модуль перегляду інформації учасників дозволяє експертам переглянути інформацію, яку ввів учасник про себе. Модуль перегляду завантажених робіт дає можливість переглянути експертам конкурсні роботи, які завантажили учасники самого конкурсу. Модуль перегляду результатів конкурсу дає змогу учаснику та журі конкурсу переглянути скільки балів набрала та чи інші конкурсна робота.

База даних складається з трьох таблиць: таблиця інформації учасників, таблиця оцінок конкурсних робіт та таблиця вектору пріоритетів альтернатив. Таблиця інформації учасників пов'язана з модулем реєстрації на участь у конкурсі та модулем перегляду інформації учасників. У модулі реєстрації на участь у конкурсі інформація з форми вводиться в таблицю інформації учасників, а в модулі перегляду інформації учасників є можливість переглянути введену інформацію з таблиці. Таблиця оцінок конкурсних робіт пов'язана з модулем запису оцінок та модулем перегляду оцінок. В модулі запису оцінок, оцінки з форми потрапляють в таблицю а в модулі перегляду оцінок здійснюється перегляд таблиці оцінок конкурсних робіт. Таблиця вектору

пріоритетів альтернатив містить інформацію яка альтернатива, тобто робота в випадку кваліфікаційної роботи, переможе за обраними критеріями.

Представлення проектних рішень з різних видів забезпечення включає в себе розробку наступних видів забезпечення: інформаційне та програмне забезпечення. Для інформаційного забезпечення проводиться аналіз потоків даних, розробки різних моделей баз даних.

2.2 Інформаційне забезпечення. Аналіз потоків даних та розробка системи управління базою даних

Інфологічна модель – відображає інформацію про предметну область у вигляді, незалежному від СУБД, що використовується. Ця модель відображає інформаційно-логічний рівень абстрагування, який пов'язаний з описом об'єктів предметної області, їх властивостей та взаємозв'язків [9]. На рисунку 2.2 зображена інфологічна модель для розробленої бази даних у вигляді діаграми Чена.

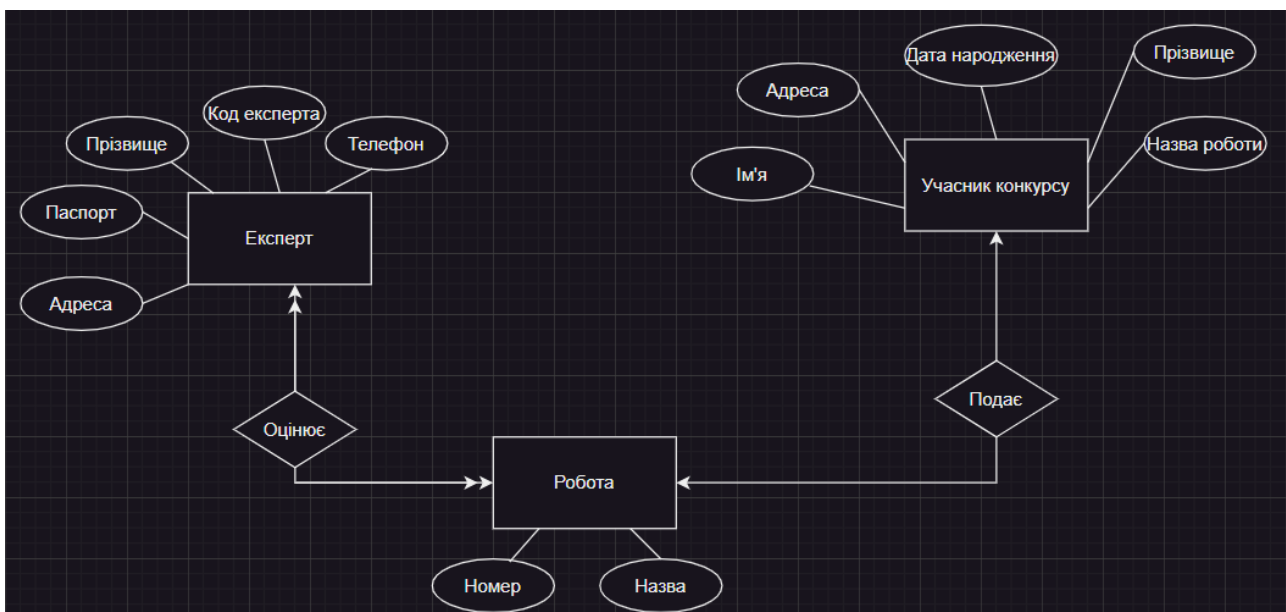


Рисунок 2.2 - Інфологічна модель дистанційного проведення конкурсів у вигляді діаграми Чена

В інфологічній моделі 3 сутності: “Учасник конкурсу”, “Робота”, “Експерти”. “Учасник конкурсу” - це людина, яка подає свою роботу на конкурс. “Робота” - це робота, яку подає учасник на конкурс. “Експерти” - це люди, які ставлять оцінки роботам по критеріям. Сутність “Експерти” має 5 атрибутів. Сутність “Учасник конкурсу” також має 5 атрибутів. Сутність “Робота” має 2 атрибуту. Всі атрибути наведені нижче в таблиці.

Правила цілісності бази даних - це правила, які дають змогу уникнути введення некоректних даних у БД, а також забезпечити можливість зв'язування декількох таблиць. Ці правила можуть бути описані при створенні чи модифікації таблиці. Також нижче в таблиці наведені правила цілісності баз даних в стовпчику “Характеристики атрибуту”.

Таблиця 2.1 - Опис сутностей інфологічної моделі

№ п/п	Назва атрибуту	Характеристики атрибуту		
		Тип значення	Обмеження на довжину	Признак ключового атрибуту
Сутність Експерти				
1	Прізвище	Символьний	100 симв.	
2	Код екперта	Числовий	10 цифр	РК
3	Телефон	Числовий	12 цифр	
4	Адреса	Символьний	50 симв.	
5	Паспорт	Символьно-числове	30 симв.	
Сутність Учасник конкурсу				

1	Прізвище	Символьний	100 симв.	
2	Ім'я	Символьний	100 симв.	
3	Назва роботи	Символьний	100 симв.	РК
4	Дата народження	Числовий	8 цифр	
5	Адреса	Символьний	50 симв.	
Сутність Робота				
1	Номер	Числовий	10 цифр	РК
2	Назва	Символьний	100 симв.	
1	Критерій	Символьний	100 симв.	

Зв'язок — це асоціація між сутностями, при якій, як правило, кожний екземпляр однієї сутності, що називається сутністю-предком, асоційований із довільною кількістю екземплярів іншої сутності, що називається сутністю-нащадком, а кожний екземпляр сутності-нащадка асоційований виключно з одним екземпляром сутності-предка.

Класифікація зв'язків:

- Один — до одного 1 : 1 Кожний учасник має лише одну роботу.
- Один — до багатьох 1 : ∞ або 1 : N Одному екземпляру однієї сутності може відповідати кілька екземплярів іншої сутності.
- Багато — до багатьох ∞ : ∞ або N : N Декілька експертів оцінюють декілька робіт.

Таблиця 2.2 - опис зв'язків між сутностями інфологічної моделі.

№ п/п	Сутності, що утворюють зв'язок	Тип зв'язку	Пояснення
1	Робота-Учасник конкурсу	1...1	Учасник конкурсу може представляти лише одну роботу.
2	Робота-Експерти	N...N	Експерти може оцінювати безліч робіт. Роботу можуть оцінювати декілька експертами.

Логічна модель розроблюваної системи, подається на діаграмі класів (Class diagram), що визначає типи класів системи і різного роду статичні зв'язки між ними. Також на діаграмі зображуються атрибути класів, операції класів та обмеження, що накладаються на зв'язки між класами. Вид та інтерпретація діаграми класів суттєво залежить від рівня абстракції, з яким вона створюється, — класи можуть подавати сутності предметної області в процесі аналізу або елементи програмної системи під час проектування і реалізації [10]. На рисунку 2.3 зображена логічна модель для розробленої бази даних.

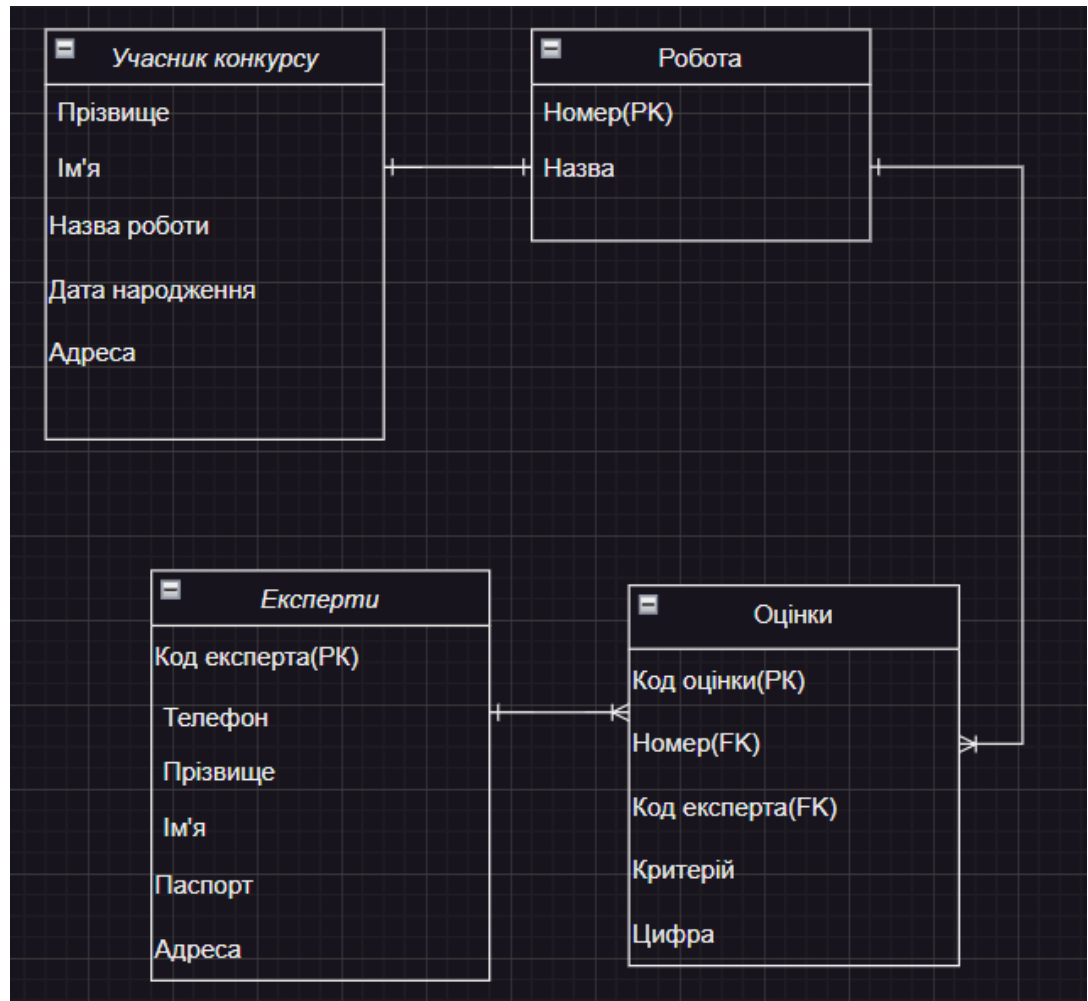


Рисунок 2.3 - Логічна модель бази даних

Фізична модель бази даних містить всі деталі, необхідні конкретної СУБД для створення бази (найменування таблиць і стовпців, типи даних, визначення первинних і зовнішніх ключів і т. п.). Фізична модель будується на основі логічної з урахуванням обмежень, що накладаються можливостями обраної СУБД: Імена сутностей і полів (таблиць і стовпців) повинні бути записані латинськими літерами, без пробілів, починаючи з букви. Для кожного поля необхідно визначити тип даних, які в ньому містяться [11]. На рисунку 2.4 зображена фізична модель для розробленої бази даних.

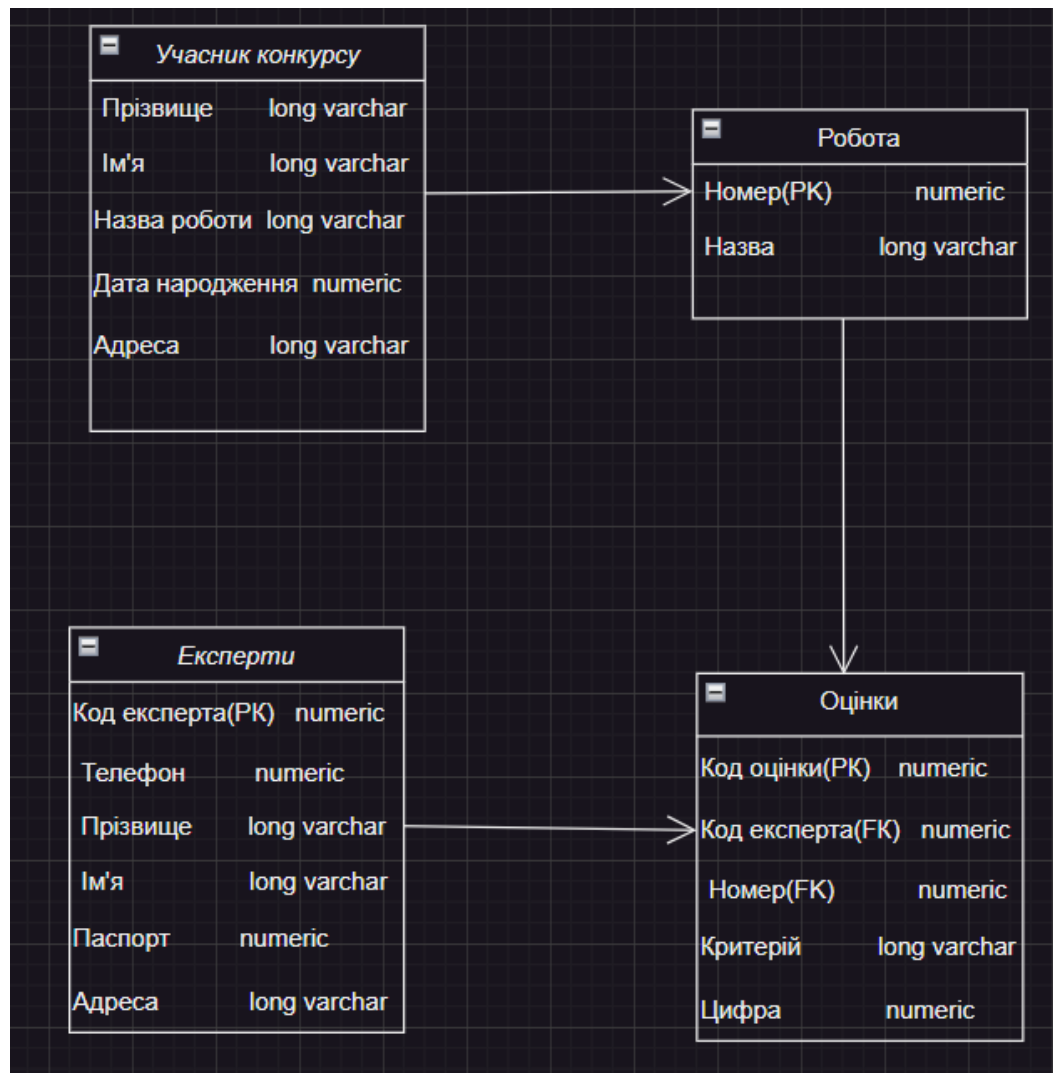


Рисунок 2.4 - Фізична модель бази даних

2.3 Програмне забезпечення

Програмне забезпечення — одна із складових інформаційної системи — сукупність програм і програмних документів, необхідних для експлуатації цих програм. В дипломній роботі буде використовуватися класична модель веб-додатку. Для розробки веб-орієнтованого додатку було обрано використання фреймворку. Для розробки програмного модуля вирішено використати мікрофреймворк Flask, через його зручність та легкість для вивчення.

Розробка веб-орієнтованого відбувається за допомогою мови програмування Python. Python було обрано через її наявності великої кількості модулів, які забезпечують різноманітні додаткові можливості. Середовище

розробки було обрано систему програмування PyCharm. До системи програмування PyCharm входять текстовий редактор, транслятор та налагоджувач. Перевагою PyCharm є підтримка фреймворків, оскільки веб-орієнтований додаток буде розроблятися за допомогою мікрофреймворку Flask. Також PyCharm надає можливість підсвічувати помилки та синтаксис.

2.4 Реалізація методу аналізу ієрархій

Спочатку було обрано критерії та альтернативи. За критерії було взято: Актуальність, оригінальність, логічну структуру, якість укладання та відповідність роботи до теми. За альтернативи - 5 робіт, які були висунуті на конкурс.

Першим кроком було побудова матриць парних порівнянь для критеріїв:

Таблиця 2.3 - таблиця матриці парних порівнянь для критеріїв.

Критерії	Відповідність роботи до теми	Оригінальність	Грамматика	Якість укладання	W	λ_{max}
	K1	K2	K3	K4		
K1	1	0,2	0,3333333333	0,2	0,0716255 9308	5,7678029 11
K2	5	1	0,2	7	0,3248589 201	
K3	3	5	1	2	0,4226139 282	
K4	5	0,14285714 29	0,5	1	0,1809015 587	
Сума	14	6,34285714 3	2,0333333333	10,2		

Також було підраховане власне значення та вектор пріоритетів.

Наступним кроком було побудова матриць парних порівнянь для кожної альтернативи, підрахунок їх векторів пріоритетів та власних значень.

Таблиця 2.4 - таблиця матриць парних порівнянь для кожної альтернативи.

	Робота №1	Робота №2	Робота №3	Робота №4		
Відповідність роботи до теми	A1	A2	A3	A4	W	λ_{max}
A1	1	0,2	0,1666666667	2	0,1017542281	6,121968915
A2	5	1	2	0,25	0,2999502582	
A3	6	0,5	1	5	0,3677024018	
A4	0,5	4	0,2	1	0,2305931119	
Сума	12,5	5,7	3,366666667	8,25		
	Робота №1	Робота №2	Робота №3	Робота №4		
Оригінальність	A1	A2	A3	A4	W	λ_{max}
A1	1	0,3333333333	0,25	4	0,1602656622	6,379661671
A2	3	1	2	0,25	0,284791331	
A3	4	0,5	1	5	0,3370556803	
A4	0,25	4	0,2	1	0,2178873265	
Сума	8,25	5,8333333333	3,45	10,25		

	Робота №1	Робота №2	Робота №3	Робота №4		
Логічна структура роботи	A1	A2	A3	A4	W	λ_{max}
A1	1	0,14285714 29	0,166666667	4	0,1283338 299	7,8665481 29
A2	7	1	5	0,25	0,3720311 009	
A3	6	0,2	1	5	0,2758396 819	
A4	0,25	4	0,2	1	0,2237953 874	
Сума	14,25	5,34285714 3	6,366666667	10,25		
	Робота №1	Робота №2	Робота №3	Робота №4		
Якість укладання	A1	A2	A3	A4	W	λ_{max}
A1	1	0,16666666 67	0,2	4	0,1335456 14	7,6594655 8
A2	6	1	5	0,25	0,3704428 915	
A3	5	0,2	1	5	0,2723713 06	
A4	0,25	4	0,2	1	0,2236401 884	
Сума	12,25	5,36666666 7	6,4	10,25		

Останнім кроком було обчислення вектору пріоритетів альтернатив.

Таблиця 2.5 - таблиця вектору пріоритетів альтернатив.

W*	0,3018405337	0,2858567094	0,2871082343	0,2854717277
----	--------------	--------------	--------------	--------------

В результаті ми отримали, що перша альтернатива, а саме Робота №1, отримала найбільше значення. Це говорить про те що, якщо оцінювати обрані роботи за визначеними критеріями, то робота №1 на першому місці в рейтингу, робота №3 на другому, робота №2 на третьому та робота №4 на четвертому.

Програмна реалізація методу аналізу ієрархій заключається в підключенні файлу Excel до проекту та оперуванні з даними з файлу.

2.5 Висновки до другого розділу

Майбутній веб-орієнтований додаток буде розроблятися за допомогою середовища розробки PyCharm, використовую фреймворк Flask та мову програмування Python. Оцінювання робіт буде проводитися за допомогою методу Борда, щоб виключити необґрунтоване заниження або завищення оцінок. Для виключення рівнозначних ваг, використання методу аналізу ієрархій надає можливість визначення ваг множини критеріїв для оцінювання конкурсних робіт.

РОЗДІЛ 3. ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ ПІДТРИМКИ ПРОВЕДЕННЯ КОНКУРСІВ

3.1 Опис використаних бібліотек

В програмному модулі використовують бібліотеку “sqlite3”. SQLite3 — це бібліотека C, яка надає легку дискову базу даних, яка не потребує окремого серверного процесу та дозволяє отримати доступ до бази даних за допомогою нестандартного варіанту мови запитів SQL. Деякі програми можуть використовувати SQLite для внутрішнього зберігання даних. Також можна створити прототип програми за допомогою SQLite, а потім перенести код у більшу базу даних, таку як PostgreSQL або Oracle.

Також в реалізації програмного модуля було використано функцію “render_template” та клас “request”.

Render_template використовується для створення виводу з файлу шаблону на основі двигуна Jinja2, який знаходиться в папці шаблонів програми.

Клас Request() модуля flask створює об'єкт запиту при кожному зверненні до URL програми, який, у свою чергу, запам'ятовує кінцеву точку endpoint і аргументи, що передаються в функцію-подання.

3.2 Специфікація програмного модулю та опис функцій

Програма складається з одного програмного модуля app.py. В цій програмний модуль входять всі потрібні функції для дистанційного проведення функцій. Опис важливих функцій, які необхідні для розробки веб-орієнтованого додатку дистанційного проведення модулів:

mai(): функція, яка відповідає за під'єднання Excel таблиці до програми та виведення з неї даних в створену таблицю бази даних.

uploadFile(): функція, яка відповідає за завантаження конкурсних робіт учасниками конкурсу в програму. Учасник конкурсу вибирає з свого пристрою

файл, який потрібно завантажити та натискає кнопку. При вірному завантаженні з'являється повідомлення про успішне завантаження конкурсної роботи.

`showData()`: функція, яка відповідає за показ завантажених конкурсних робіт.

`addrec()`: функція, яка відповідає за завантаження оцінок конкурсних робіт в таблицю бази даних. Експерт вводить свої оцінки в html форму та натискає кнопку. При натисненні кнопки відбувається вивантаження введених оцінок з форми та переносяться в таблицю `mark` бази даних.

`list()`: функція, яка відповідає за виведення таблиці `mark` на html сторінку.

Функції `ressum()`, `ressum2()`, `ressum3()`, `ressum4()`, `ressum5()` відповідають за виведення результатів конкурсу по кожній конкурсній роботі.

`partform()`: функція, яка відповідає за завантаження інформації про учасників конкурсних робіт в таблицю `info` бази даних. Учасник конкурсу вводить інформацію про себе, таку як прізвище та ім'я, дату народження, місце проживання та назву роботи в html форму. Після заповнення форми, введена інформація завантажується в створену таблицю `info` в базі даних.

`info()`: функція, яка відповідає за виведення таблиці `info` на html сторінку.

`add()`: функція, яка відповідає за додавання рядку в таблицю. Якщо з'явиться необхідність додати рядок до будь-якої таблиці, то є можливість швидко це зробити, просто ввівши назву таблиці та назву нового рядку.

`delete()`: функція, яка дає змогу видалити необхідний рядок з таблиці. Ввівши назву таблиці та назву рядку та значення звідти, необхідний рядок видаляється.

Розглянемо створені html сторінки веб-орієнтованого додатку:

- 1) `finish.html`: на сторінці виводиться результат оцінювання конкурсної роботи №1;
- 2) `finish2.html`: на сторінці виводиться результат оцінювання конкурсної роботи №2;

- 3) finish3.html: на сторінці виводиться результат оцінювання конкурсної роботи №3;
- 4) finish4.html: на сторінці виводиться результат оцінювання конкурсної роботи №4;
- 5) finish5.html: на сторінці виводиться результат оцінювання конкурсної роботи №5;
- 6) home.html: головна сторінка, на якій є змога перейти на необхідну іншу сторінку;
- 7) infopart.html: сторінка, на якій виводиться таблиця info з бази даних;
- 8) infozapus.html: сторінка, на якій є форма для запису своєї інформації для учасника;
- 9) list.html: сторінка, на якій виводиться таблиця mark з бази даних;
- 10) result.html: сторінка, на якій виводиться повідомлення “Оцінки успішно додані” після натискання кнопки на сторінці student.html;
- 11) resulti.html: сторінка, на якій виводиться повідомлення “Реєстрація пройшла успішно” після натискання кнопки на сторінці infozapus.html;
- 12) show.html: сторінка відповідає за виведення завантажених конкурсних робіт на екран;
- 13) student.html: сторінка, на якій є форма для запису оцінок конкурсних робіт;
- 14) upload1.html: сторінка, на якій відбувається завантаження конкурсної роботи учасником.
- 15) vector.html: сторінка, яка виводить таблицю vector з бази даних, в якій знаходиться вектор пріоритетів альтернатив з методу аналізу ієрархій.

3.3 Підключення методу аналізу ієрархій

Спершу було створено таблицю для даних, які знаходяться в Excel файлі.

Наступним кроком було написання коду для передавання даних з Excel файлу в створену таблицю. На рисунку 3.4 показана створена таблиця та її робота.

Робота_1	Робота_2	Робота_3	Робота_4	Робота_5
0.21338742...	0.20509364...	0.20737884...	0.20201468...	0.20364142...

Рисунок 3.4 - Створена таблиця

Далі було створено html сторінку, в якій знаходиться таблиця з векторами пріоритетів альтернатив, та створено функцію в головному файлі app.py для виведення даних на html сторінку. На рисунку 3.5 зображено вивід html сторінки для таблиця векторів пріоритетів альтернатив.

Вектори пріоритетів

Робота №1	Робота №2	Робота №3	Робота №4	Робота №5
0.2133874259	0.2050936468	0.2073788443	0.2020146892	0.2036414238

[Головна сторінка](#)

Рисунок 3.5 - Результат роботи коду

3.3 Опис програмної реалізації

Першим кроком було створення бази даних, в якій будуть створені таблиці “mark”, “info” та “vector”.

Таблиця “vector” містить вектори пріоритетів альтернатив. Опис створення, підключення та виведення був описаний вище.

Таблиця “mark” містить інформацію щодо виставлених оцінок експертами. Таблиця складається з чотирьох рядків, а саме: Назва критерію, Робота №1, Робота №2, Робота №3, Робота №4, Робота №5. В залежності від потреб, в таблицю можна додавати рядки з новими роботами. Код для створення є в програмі, тобто в подальшому використанні коду програми додавання рядків не є проблемою.

Таблиця “info” містить інформацію про учасника. Таблиця має стовпці “Прізвище та ім’я”, “Дата народження”, “Місце проживання”, “Назва роботи”. Всю цю інформацію учасник має можливість ввести сам через форму на html сторінці.

На рисунку 3.9 показано створену таблицю для інформації учасників.

Прізвище_та_ім'я	Дата_народження	Місце_проживання	Назва_роботи
------------------	-----------------	------------------	--------------

Рисунок 3.9 - Створена таблиця

Другим кроком було створення головної сторінки, в якій є можливість для учасників зареєструватися, завантажити конкурсну роботу та переглянути результат конкурсу. Для експертів є можливість записати оцінки, переглянути їх, переглянути введену інформацію учасниками та переглянути завантажені роботи. На рисунку 3.10 зображено діаграму переходів між екранними вікнами для учасників.

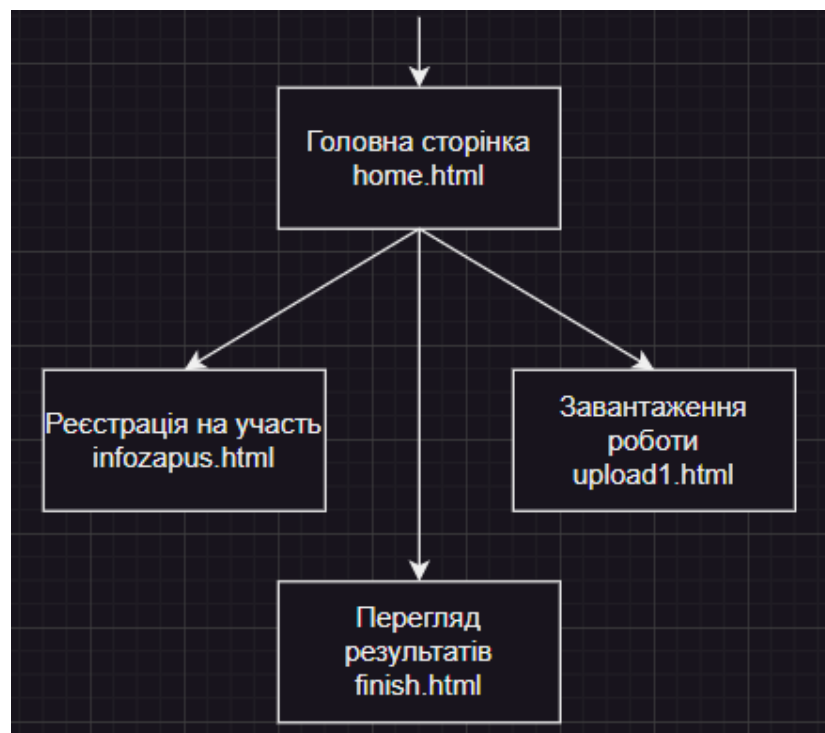


Рисунок 3.10 - Діаграма переходів між екранними вікнами для учасників

На рисунку 3.11 зображено діаграму переходів між екранними вікнами для експертів.

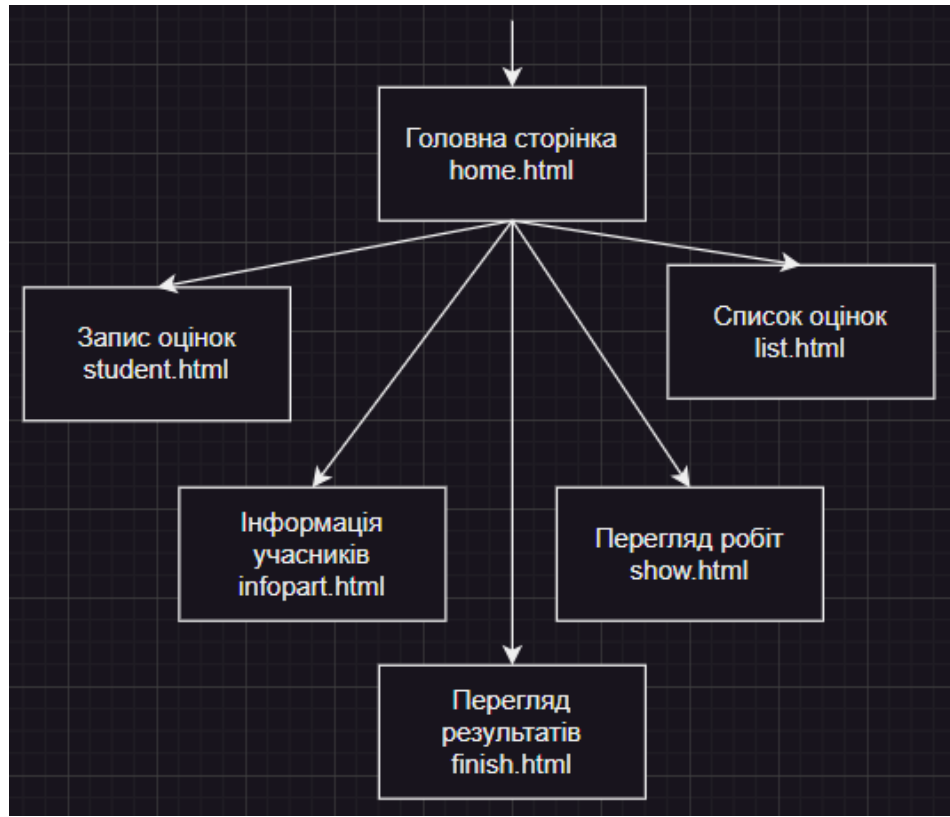


Рисунок 3.11 - Діаграма переходів між екранними вікнами для експертів

Для html сторінки з реєстрацією учасників було створено форму для введення інформації. Також створено функцію `partform()`, яка передає дані з форми в таблицю “info”.

Для завантаження конкурсної роботи, учасник завантажує файл з роботою, який потім зберігається в програмі.

Перегляд результатів конкурсу реалізовано за допомогою підрахунку суми оцінок за всіма критеріями та виведення суми на html сторінку.

Наступним кроком було створення форми для запису оцінок. Далі було створено функцію `addrec()`, яка вставляє дані, які були написані в форму, в таблицю “mark”.

Функція `list()` відповідає за заповнення рядків на html сторінку, в які записані всі дані з таблиці.

За перегляд введеної інформації учасників відповідає функція info().

Перегляд завантажених робіт відбувається за допомогою функції showData().

3.4 Результат тестування розробленої системи

Розроблену програмну систему протестовано на предмет повноти її функціоналу. На рисунку 3.12 та 3.13 представлено головну сторінку розробленого веб-орієнтованого додатку.

Вітаємо вас на головній сторінці

Для учасників

[Реєстрація на участь у конкурсі](#)

[Завантаження роботи](#)

Для експертів

[Записати бали](#)

[Список оцінок](#)

[Вектор пріоритетів](#)

[Інформація учасників конкурсу](#)

[Перегляд робіт](#)

Рисунок 3.12 - Головна сторінка розробленої програми

Спільне

[Перегляд результатів Роботи №1](#)

[Перегляд результатів Роботи №2](#)

[Перегляд результатів Роботи №3](#)

[Перегляд результатів Роботи №4](#)

[Перегляд результатів Роботи №5](#)

Рисунок 3.13 - Головна сторінка розробленої програми

З головної сторінки перейдемо на сторінку реєстрації на участь (див. рисунок 3.14):

Реєстрація

Прізвище та ім'я
Демченко Сергій

Дата народження
19.08.2009

Місце проживання
місто Житомир, вулиця Сх

Назва роботи
Світ мертвий

submit

Рисунок 3.14 - Сторінка для запису інформації учасника

На рисунках 3.15, 3.16 та 3.17 зображено тестування форми для запису інформації учасника.

Реєстрація

Прізвище та ім'я
Іванов Олег

Дата народження
24.11.2004

Місце проживання
місто Тернопіль

Назва роботи
Мій дім

submit

Рисунок 3.15 - Тестування реєстрації

Реєстрація

Прізвище та ім'я
Кулик Аліна

Дата народження
26.12.2002

Місце проживання
місто Чернігів

Назва роботи
Нова історія

submit

Рисунок 3.16 - Тестування реєстрації

Реєстрація

Прізвище та ім'я
Бурцевий Данііл

Дата народження
17.11.2010

Місце проживання
місто Херсон

Назва роботи
Це моє місто

submit

Рисунок 3.17 - Тестування реєстрації

Запишемо вигадану інформацію, для перевірки роботи запису в таблицю та відображення записаних даних. При натисненні на кнопку “submit”, відбувається перехід на наступну сторінку (Рисунок 3.18), на якій виводиться повідомлення про успішний запис інформації та посилання на головну сторінку.

Реєстрація пройшла успішно

[Головна сторінка](#)

Рисунок 3.18 - Сторінка результату реєстрації

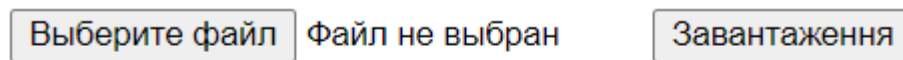
Для перегляду таблиці введених даних, потрібно перейти на сторінку “Інформація учасників конкурсу”. На рисунку 3.19 показано роботу сторінки з інформацією учасників.

Прізвище та ім'я	Дата народження	Місце проживання	Назва роботи
Анастасія Кузьменко	2006-10-26	місто Клів, проспект Червоної Калини	Світ динозаврів
Демченко Сергій	2009-08-19	місто Житомир, вулиця Східна	Світ мертвий
Іванов Олег	2004-11-24	місто Тернопіль	Мій дім
Кулик Аліна	2002-12-26	місто Чернігів	Нова історія
Бурцевий Данііл	2010-11-17	місто Херсон	Це моє місто

[Головна сторінка](#)

Рисунок 3.19 - Введена інформація у таблиці

З головної сторінки перейдемо на сторінку завантаження роботи (див. рисунок 3.20):



Файл не завантажено

Рисунок 3.20 - Сторінка для завантаження роботи

Для завантаження роботи потрібно натиснути на кнопку “Виберіть файл”. Після обрання файлу, натискаємо на кнопку “Завантаження”. Якщо завантаження пройшло успішно напис “Файл не завантажено” перетворюється на “Файл завантажено” зеленого кольору (див. рисунок 3.21).

Файл завантажено

Рисунок 3.21 - Напис про успішне завантаження роботи

На рисунку 3.22 показано текст конкурсної роботи.

Kyiv is the capital of Ukraine. It's one of the oldest cities of Europe. Kyiv is situated on the picturesque banks of the Dnieper river. The population of Kyiv is about 3 million. Kyiv is a large political, industrial and cultural centre. Kyiv is a seat of the higher body of state power of Ukraine, the Verkhovna Rada. Kyiv is a scientific centre. The Academy of Sciences of Ukraine is here. Kyiv's cultural life is rich and varied. There are many theatres, museums, exhibitions in Kyiv. Kyiv's theatres are well-known. The famous National Opera and Ballet Theatre and other theatres are greatly appreciated by the Kyivites and the guests of the city. The most important educational institutions of Ukraine are in Kyiv. The National Ukrainian University named after Shevchenko is in the centre of the capital. Kyiv is famous for its monuments. The monument to Prince Volodymyr in the picturesque park on the Dnieper hills has become a symbol of the ancient city. Kyiv is the Hero-City. In the park of Immortal Glory there is an obelisk in honour of those who fell in the battles of the Second World War. A lot of tourists come to Kyiv. The central street of the capital is Khreshchatyk. There are many fine buildings in Khreshchatyk. Kyiv is especially beautiful in spring when the famous Kyiv chestnut trees are in blossom.

Рисунок 3.22 - Текст в файлі для перевірки виведення

Перегляд файлу відбувається на сторінці “Перегляд робіт”(див. рисунок 3.23):

Робота №1

Kyiv is the capital of Ukraine. It is one of the oldest cities of Europe. Kyiv is situated on the picturesque banks of the Dnieper river. The population of Kyiv is about 3 million. Kyiv is a large political, industrial and cultural centre. Kyiv is a seat of the higher body of state power of Ukraine, the Verkhovna Rada. Kyiv is a scientific centre. The Academy of Sciences of Ukraine is here. Kyiv's cultural life is rich and varied. There are many theatres, museums, exhibitions in Kyiv. Kyiv's theatres are well-known. The famous National Opera and Ballet Theatre and other theatres are greatly appreciated by the Kyivites and the guests of the city. The most important educational institutions of Ukraine are in Kyiv. The National Ukrainian University named after Shevchenko is in the centre of the capital. Kyiv is famous for its monuments. The monument to Prince Volodymyr in the picturesque park on the Dnieper hills has become a symbol of the ancient city. Kyiv is the Hero-City. In the park of Immortal Glory there is an obelisk in honour of those who fell in the battles of the Second World War. A lot of tourists come to Kyiv. The central street of the capital is Khreshchatyk. There are many fine buildings in Khreshchatyk. Kyiv is especially beautiful in spring when the famous Kyiv chestnut trees are in blossom.

Рисунок 3.23 - Завантажена робота учасника

На рисунку 3.24 показано текст конкурсної роботи.

Meet my family. There are five of us – my parents, my elder brother, my baby sister and me. First, meet my mum and dad, Jane and Michael. My mum enjoys reading and my dad enjoys playing chess with my brother Ken. My mum slim and is rather tall. She has long red hair and big brown eyes. She has a very pleasant smile and a soft voice. My mother is very kind and understanding. We are real friends. She is a housewife. As she has three children, she is always busy around the house. She takes care of my baby sister Meg, who is only three months old. My sister is very small and funny. She sleeps, eats and sometimes cries. We all help our mother and let her have a rest in the evening. Then she usually reads a book or just watches TV. My father is a doctor. He is tall and handsome. He has short dark hair and gray eyes. He is a very hardworking man. He is rather strict with us, but always fair. My elder brother Ken is thirteen, and he is very clever. He is good at Maths and always helps me with it, because I can hardly understand all these sums and problems. Ken has red hair and brown eyes. My name is Jessica. I am eleven. I have long dark hair and brown eyes. I am not as clever as my brother, though I try to do my best at school too. I am fond of dancing. Our dancing studio won The Best Dancing Studio 2015 competition last month. I am very proud of it. I also like to help my mother with my little sister very much. Our family is very united. We love each other and always try to spend more time together.

Рисунок 3.24 - Текст в файлі для перевірки виведення

Перегляд файлу відбувається на сторінці “Перегляд робіт”(див. рисунок 3.25):

Meet my family. There are five of us – my parents, my elder brother, my baby sister and me. First, meet my mum and dad, Jane and Michael. My mum enjoys reading and my dad enjoys playing chess with my brother Ken. My mum slim and is rather tall. She has long red hair and big brown eyes. She has a very pleasant smile and a soft voice. My mother is very kind and understanding. We are real friends. She is a housewife. As she has three children, she is always busy around the house. She takes care of my baby sister Meg, who is only three months old. My sister is very small and funny. She sleeps, eats and sometimes cries. We all help our mother and let her have a rest in the evening. Then she usually reads a book or just watches TV. My father is a doctor. He is tall and handsome. He has short dark hair and gray eyes. He is a very hardworking man. He is rather strict with us, but always fair. My elder brother Ken is thirteen, and he is very clever. He is good at Maths and always helps me with it, because I can hardly understand all these sums and problems. Ken has red hair and brown eyes. My name is Jessica. I am eleven. I have long dark hair and brown eyes. I am not as clever as my brother, though I try to do my best at school too. I am fond of dancing. Our dancing studio won The Best Dancing Studio 2015 competition last month. I am very proud of it. I also like to help my mother with my little sister very much. Our family is very united. We love each other and always try to spend more time together.

Рисунок 3.25 - Завантажена робота учасника

На рисунку 3.26 показано текст конкурсної роботи.

Every day in elementary school in America begins at 9.20 a.m. Children have classes till 3.15 p.m. At 12 o'clock children have lunch. Many boys and girls bring their lunch from home. But some of them go for lunch to a school cafeteria. Mrs. Bradley prepares school lunches almost every weekday for her two children. Sometimes she gives the children money and they eat in the school cafeteria. But usually the children prefer to take a lunch from home. This morning Mrs. Bradley is making peanut butter and cheese sandwiches, the children's favorite. She puts two bottles of apple juice for the children to drink. She is going to put the sandwiches, some cherry tomatoes and two bananas in their lunchboxes. The lunchbox is easy for the children to carry to school.

Рисунок 3.26 - Текст в файлі для перевірки виведення

Перегляд файлу відбувається на сторінці “Перегляд робіт”(див. рисунок 3.27):

Every day in elementary school in America begins at 9.20 a.m. Children have classes till 3.15 p.m. At 12 o'clock children have lunch. Many boys and girls bring their lunch from home. But some of them go for lunch to a school cafeteria. Mrs. Bradley prepares school lunches almost every weekday for her two children. Sometimes she gives the children money and they eat in the school cafeteria. But usually the children prefer to take a lunch from home. This morning Mrs. Bradley is making peanut butter and cheese sandwiches, the children's favorite. She puts two bottles of apple juice for the children to drink. She is going to put the sandwiches, some cherry tomatoes and two bananas in their lunchboxes. The lunchbox is easy for the children to carry to school.

Рисунок 3.27 - Завантажена робота учасника

На рисунку 3.28 показано текст конкурсної роботи.

There are different kinds of animals on our planet, and all of them are very important for it. For example, everybody knows that the sharks are dangerous for people, but they are useful for cleaning seawater. There are two types of animals: domestic (or pets) and wild. People keep pets in their homes. And some wild animals are very dangerous. Domestic animals live next to people, whereas wild animals' "homes" are forests, jungles, oceans and so on. Giraffes are very beautiful and unusual animals. They are the tallest land animals in the world. Giraffes can reach a height of 5,5 m and a weight of 900 kg. They are famous for their long necks. But does anybody know that giraffes have a very long tongue? They can even clean the ears with it! Giraffes are usually yellow or light brown with dark stains. Giraffes live in African savannas. They can live from 20 до 30 years. It is interesting to know that giraffes sleep only for a few minutes at a time. They sit down on the ground and bend their long neck down. Giraffes do not hunt. They eat leaves, grass, and fruit. Due to their long neck, they can reach the highest leaves on the trees that other animals cannot eat. You can often meet giraffes in city Zoos. They are very friendly and all the children like them very much.

Рисунок 3.28 - Текст в файлі для перевірки виведення

Перегляд файлу відбувається на сторінці “Перегляд робіт”(див. рисунок 3.29):

There are different kinds of animals on our planet, and all of them are very important for it. For example, everybody knows that the sharks are dangerous for people, but they are useful for cleaning seawater. There are two types of animals: domestic (or pets) and wild. People keep pets in their homes. And some wild animals are very dangerous. Domestic animals live next to people, whereas wild animals' "homes" are forests, jungles, oceans and so on. Giraffes are very beautiful and unusual animals. They are the tallest land animals in the world. Giraffes can reach a height of 5,5 m and a weight of 900 kg. They are famous for their long necks. But does anybody know that giraffes have a very long tongue? They can even clean the ears with it! Giraffes are usually yellow or light brown with dark stains. Giraffes live in African savannas. They can live from 20 до 30 years. It is interesting to know that giraffes sleep only for a few minutes at a time. They sit down on the ground and bend their long neck down. Giraffes do not hunt. They eat leaves, grass, and fruit. Due to their long neck, they can reach the highest leaves on the trees that other animals cannot eat. You can often meet giraffes in city Zoos. They are very friendly and all the children like them very much.

Рисунок 3.29 - Завантажена робота учасника

Далі проведемо тестування запису оцінок. Переходимо на сторінку “Записати бали”(див. рисунок 3.30).

Запис Оцінок

Назва критерію
 ▼
 Робота №1

 Робота №2

 Робота №3

 Робота №4

 Робота №5

Рисунок 3.30 - Сторінка для запису оцінок

На рисунках 3.31, 3.32 та 3.33 зображено тестування форми для запису оцінок конкурсної роботи.

Запис Оцінок

Назва критерію
Граматика

Робота №1
5

Робота №2
2

Робота №3
4

Робота №4
1

Робота №5
3

submit

Рисунок 3.31 - Тестування запису оцінок

Запис Оцінок

Назва критерію
Якість укладання

Робота №1
3

Робота №2
4

Робота №3
2

Робота №4
1

Робота №5
5

submit

Рисунок 3.32 - Тестування запису оцінок

Назва критерію
<input type="text" value="Якість укладання"/>
Робота №1
<input type="text" value="3"/>
Робота №2
<input type="text" value="1"/>
Робота №3
<input type="text" value="5"/>
Робота №4
<input type="text" value="4"/>
Робота №5
<input type="text" value="2"/>
<input type="button" value="submit"/>

Рисунок 3.33 - Тестування запису оцінок

В першому рядку є можливість у виборі за яким критерієм буде проведено оцінювання (див. рисунок 3.34).

Назва критерію
<input type="text" value="Оригінальність"/>
<input type="text" value="Оригінальність"/>
<input type="text" value="Якість укладання"/>
<input type="text" value="Граматика"/>
<input type="text" value="Відповідність до теми"/>

Рисунок 3.34 - Список критеріїв

Після запису оцінок, при натисненні на кнопку submit, експерта переносить на сторінку про успішний запис (див. рисунок 3.35).

Оцінки успішно додані

[Головна сторінка](#)

Рисунок 3.35 - Сторінка результату запису

Для перегляду таблиці введених даних, потрібно перейти на сторінку “Список оцінок”. На рисунку 3.36 зображено введені оцінки конкурсних робіт у форму.

Назва критерію	Робота_1	Робота_2	Робота_3	Робота_4	Робота_5
Якість укладання	2	3	1	5	4
Оригінальність	4	2	5	1	3
Граматика	1	4	2	5	3
Відповідність до теми	3	4	2	5	1
Оригінальність	4	2	5	3	1
Граматика	5	2	4	1	3
Якість укладання	3	4	2	1	5
Відповідність до теми	5	2	4	3	1
Якість укладання	3	1	5	4	2

[Головна сторінка](#)

Рисунок 3.36 - Введена інформація у таблиці

Бачимо, що останній рядок містить введені нами оцінки, отже запис працює правильно.

Для перегляду результатів оцінювання робіт потрібно перейти на окремі сторінки для окремих робіт. Підрахуємо для Роботи 1 суму оцінок: 31 балів. На рисунку 3.37 показано вивід інформації на сторінці “Перегляд результатів Роботи №1”:

Робота_1
31

[Головна сторінка](#)

Рисунок 3.37 - Виведення результату оцінювання роботи №1

Підрахуємо для Роботи 2 суму оцінок: 29 балів. На рисунку 3.38 показано вивід інформації на сторінці “Перегляд результатів Роботи №2”:

Робота_2
29

[Головна сторінка](#)

Рисунок 3.38 - Виведення результату оцінювання роботи №2

Підрахуємо для Роботи 3 суму оцінок: 34 бали. На рисунку 3.39 показано вивід інформації на сторінці “Перегляд результатів Роботи №3”:

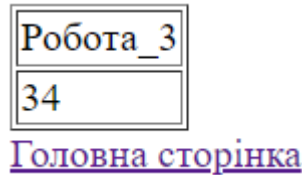


Рисунок 3.39 - Виведення результату оцінювання роботи №3

Підрахуємо для Роботи 4 суму оцінок: 30 балів. На рисунку 3.40 показано вивід інформації на сторінці “Перегляд результатів Роботи №4”:

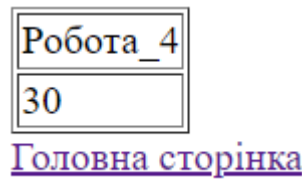


Рисунок 3.40 - Виведення результату оцінювання роботи №4

Підрахуємо для Роботи 5 суму оцінок: 26 балів. На рисунку 3.41 показано вивід інформації на сторінці “Перегляд результатів Роботи №5”:

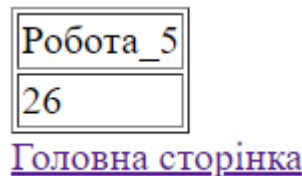


Рисунок 3.41 - Виведення результату оцінювання роботи №5

За результатами оцінок Робота 3 отримує перше місце (34 бали), друге місце отримує Робота 1 (31 балів), третє місце - Робота 4 (30 балів), четверте місце - Робота 2 (29 балів) та останнє місце отримує Робота 5 (26 балів).

Висновок щодо результату тестування розробленої програми: Розроблена програма працює правильно. Всі записані дані зберігаються в створених таблицях та виводяться на html сторінки.

3.5 Аналіз тестування розробленої системи підтримки

Під час тестування розробленого веб-орієнтованого додатку було визначено, що програма працює вірно та відповідно до сформованих

функціональних та нефункціональних вимог. Результати роботи розробленого програмного забезпечення відповідають завданню випускної кваліфікаційної роботи та функціональним вимогам, які були описані в розділі 1.5. Програмне забезпечення надає можливість:

1. Для учасників конкурсу:
 - Реєстрація на участь у конкурсу;
 - Завантаження файлу конкурсної роботи;
 - Перегляд результатів оцінювання робіт.
2. Для експертів:
 - Перегляд завантажених конкурсних робіт;
 - Оцінювання робіт;
 - Перегляд оцінок учасників;
 - Перегляд результатів оцінювання робіт.

Якщо говорити про відповідність нефункціональним вимогам, то розроблене програмне забезпечення є продуктивним. Програмне забезпечення надає користувачу просте та зрозуміле здійснення функціональних вимог. Також розроблений веб-орієнтований додаток надає можливість масштабувати сторінки без значної зміни. На рисунку 3.42 зображена не масштабована таблиця з інформацією учасників.

Прізвище та ім'я	Дата народження	Місце проживання	Назва роботи
Анастасія Кузьменко	2006-10-26	місто Кпів, проспект Червоної Калини	Світ динозаврів
Демченко Сергій	2009-08-19	місто Житомир, вулиця Східна	Світ мертвих
Іванов Олег	2004-11-24	місто Тернопіль	Мій дім
Кулик Аліна	2002-12-26	місто Чернігів	Нова історія

Рисунок 3.42 - Не масштабована таблиця інформації учасників

На рисунку 3.43 показано як масштабується таблиця з інформацією учасників при зміні розміру вікна браузера.

Прізвище та ім'я	Дата народження	Місце проживання	Назва роботи
Анастасія Кузьменко	2006-10-26	місто Київ, проспект Червоної Калини	Світ динозаврів
Демченко Сергій	2009-08-19	місто Житомир, вулиця Східна	Світ мертвих
Іванов Олег	2004-11-24	місто Тернопіль	Мій дім
Кулик Аліна	2002-12-26	місто Чернігів	Нова історія

Рисунок 3.43 - Правильно маштабована таблиця інформації учасників

Опис інструктивних матеріалів користувачу:

Для використання розробленої програми користувачу потрібно лише мати будь-який браузер, який наявний в усіх ноутбуків, комп'ютерів та телефонів. У використанні розробленого веб-орієнтованого додатку у користувача не виникне проблем, через те що на даний момент кожна людина вміє користуватися браузером, а більше для використання нічого не потрібно.

Подальше використання розробленого програмного забезпечення є можливим. Для повноцінного дистанційного проведення конкурсів потрібно допрацювати систему для авторизації. Також якщо є потреба то створення нових html сторінок, наприклад виведення посилань на Zoom конференції для учасників для захисту робіт, якщо є потреба.

ВИСНОВКИ

У випускній дипломній роботі розглянуто процес дистанційного проведення конкурсів. Через появу коронавірусу та початку війни в країні, з'явилася необхідність перевести всі процеси конкурсного відбору в дистанційний режим. Задля безпеки учасників та/або журі конкурсу було розроблено веб-орієнтований додаток, яким можна користуватися в безпечному місці, такі як укриття. Також, проаналізувавши наявний процес проведення конкурсів було виявлено ряд недоліків. Для усунення цих недоліків, розроблена система допоможе уникнути урахувань недостовірних оцінок, маніпуляцій зі сторони експертів та викривлення результатів оцінювання конкурсної роботи. Проаналізувавши недоліки наявних систем запропоновані підходи до усунення недоліків при розробці нової програмної системи.

Розглядаючи варіанти розробки системи підтримки дистанційного проведення конкурсів, було вирішено розробити веб-орієнтований додаток, який буде надавати можливість як учасникам конкурсу так і експертам, які оцінюють роботи, дистанційно проводити свою роботу. Було визначено, що для учасників конкурсу має бути наявна можливість зареєструватися на конкурс, завантажити конкурсну роботу та переглянути результати оцінювання. Для експертів має бути наявна можливість перегляду інформації про учасників конкурсу, запису та перегляду оцінок, перегляду робіт учасників та результат оцінювання конкурсних робіт. В створеному веб-орієнтованому додатку весь цей функціонал розроблено та протестовано.

При реалізації системи супроводу дистанційних конкурсів застосовано метод аналізу ієрархій для групової декомпозиції узагальненого критерію та формування та визначення ваг множини критеріїв для оцінювання конкурсних робіт. Також, у випускній дипломній роботі був використаний метод Борда для групового ординального оцінювання проектів.

Мету роботи, яка полягає в розробці веб-орієнтованого додатку для дистанційного проведення конкурсів, з уникненням всіх наявних недоліків наявних технологій було повністю досягнуто.

Застосування розробленої системи розраховане на подальше повноцінне дистанційне проведення конкурсів. При подальшому вдосконаленні розробленого веб-орієнтованого додатку доцільним буде включити в арсенал методів обробки групових експертних оцінок визначення та підвищення узгодженості у ході експертизи, а також урахування компетентності (інформованості) члена журі у питанні конкурсної роботи, що оцінюється.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Циганок В.В. Дослідження ефективності агрегування експертних оцінок, отриманих методом попарного порівняння. – 2010. – 538-544 с.
2. Циганок В.В. Моделювання експертних оцінок для тестування методів обробки інформації в системах підтримки прийняття рішень. – 2011. – 21-32 с.
3. Циганок В.В. Метод обчислення ваг альтернатив на основі результатів парних порівнянь, проведених групою експертів. Реєстрація, зберігання і обробка даних. – 2008. т.10, №2. – с.121-127.
4. Тоценко В.Г Підтримка прийняття рішень при проведенні великомасштабних конкурсів проектів. Методи і засоби підтримки прийняття рішень. – 1999. – Т.1. – № 2.
5. Система розподіленого збору експертної інформації “Consensus-2”.
6. Саати Томас Метод аналізу ієрархій. The analytic hierarchy process—what it is and how it is used. 1987.
7. Компанія Visure Solutions. Що таке функціональні вимоги: приклади, визначення, повний посібник.
<https://visuresolutions.com/uk/%D0%B1%D0%BB%D0%BE%D0%B3/%D1%84%D1%83%D0%BD%D0%BA%D1%86%D1%96%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%96-%D0%B2%D0%B8%D0%BC%D0%BE%D0%B3%D0%B8/>
8. Компанія Visure Solutions. Що таке нефункціональні вимоги: приклади, визначення, повне керівництво.
<https://visuresolutions.com/uk/%D0%B1%D0%BB%D0%BE%D0%B3/%D0%BD%D0%B5%D1%84%D1%83%D0%BD%D0%BA%D1%86%D1%96%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%96-%D0%B2%D0%B8%D0%BC%D0%BE%D0%B3%D0%B8/>
9. Коцюба А.Ю.Моделі даних. Реляційна модель даних. Інфологічна модель даних. 2015

https://elib.lntu.edu.ua/sites/default/files/elib_upload/BD_2016_3/page5.html

10. Бібліотека підручників та статей Posibniki. Логічна модель системи. 2022.

<https://posibniki.com.ua/post-logichna-model-sistemi>

11. Учбові матеріали та реферати. Фізична модель бази даних. 2018.

<http://um.co.ua/2/2-6/2-64980.html>

ДОДАТОК А

Код програми app.py:

```
import sqlite3

from flask import Flask, render_template, request
import sqlite3 as sql
import openpyxl
import pandas as pd
from flask import *
import os
from werkzeug.utils import secure_filename

UPLOAD_FOLDER = os.path.join('staticFiles', 'uploads')
ALLOWED_EXTENSIONS = {'csv'}
app = Flask(__name__)
app.config["UPLOAD_FOLDER"] = UPLOAD_FOLDER
app.secret_key = 'This is your secret key to utilize session in Flask'

@app.route('/vector')
def mai():
    book = openpyxl.open("метод аналізу ієрархій.xlsx", read_only=True)
    sheet = book.active
    cells = sheet['K14':'O14']
    con = sql.connect("databasemark.db")
    cur = con.cursor()
    for a1, a2, a3, a4, a5 in cells:
        Робота_1 = a1.value
        Робота_2 = a2.value
        Робота_3 = a3.value
        Робота_4 = a4.value
        Робота_5 = a5.value
    cur.execute(
        "INSERT INTO vector (Робота_1,Робота_2,Робота_3,Робота_4,Робота_5)"
        "VALUES(?, ?, ?, ?, ?)", (Робота_1, Робота_2, Робота_3, Робота_4, Робота_5))
    con.commit()
    con.row_factory = sql.Row
    cur = con.cursor()
    cur.execute("select * from vector")
    Rows = cur.fetchall()
    return render_template("vector.html", Rows=Rows)
```

```

@app.route('/')
def home():
    return render_template('home.html')

@app.route('/upload', methods=['GET', 'POST'])
def uploadFile():
    if request.method == 'POST':
        f = request.files.get('file')
        data_filename = secure_filename(f.filename)
        f.save(os.path.join(app.config['UPLOAD_FOLDER'],
                               data_filename))
        session['uploaded_data_file_path'] = os.path.join(app.config['UPLOAD_FOLDER'],
                                                             data_filename)
    return render_template('upload2.html')
return render_template("upload1.html")

@app.route('/show')
def showData():
    data_file_path = session.get('uploaded_data_file_path', None)
    uploaded_df = pd.read_csv(data_file_path, encoding='unicode_escape')
    uploaded_df_html = uploaded_df.to_html()
    return render_template('show.html', data_var=uploaded_df_html)

@app.route('/enternew')
def new_student():
    return render_template('student.html')

@app.route('/addrec', methods=['POST', 'GET'])
def addrec():
    if request.method == 'POST':
        try:
            Назва_критерію = request.form['name']
            Актуальність = request.form['cr1']
            Оригінальність = request.form['cr2']
            Логічна_структура_роботи = request.form['cr3']
            Якість_укладання = request.form['cr4']
            Відповідність_теми_до_роботи = request.form['cr5']

```

```
with sql.connect("databasemark.db") as con:
```

```
    cur = con.cursor()
```

```
        cur.execute(
```

```
            "INSERT INTO mark (Критерій, Робота 1, Робота 2, Робота 3, Робота 4, Робота 5)"
```

```
            "VALUES(?, ?, ?, ?, ?, ?)", (
```

```
                Назва_критерію, Актуальність, Оригінальність, Логічна_структура_роботи, Якість_укладання,
```

```
                Відповідність_теми_до_роботи))
```

```
        con.commit();
```

```
    except:
```

```
        con.rollback()
```

```
    finally:
```

```
        return render_template("result.html")
```

```
    con.close()
```

```
@app.route('/list')
```

```
def list():
```

```
    con = sql.connect("databasemark.db")
```

```
    con.row_factory = sql.Row
```

```
    cur = con.cursor()
```

```
    cur.execute("select * from mark")
```

```
    rows = cur.fetchall();
```

```
    return render_template("list.html", rows=rows)
```

```
@app.route('/resultat')
```

```
def finish():
```

```
    return render_template('resultat.html')
```

```
@app.route('/result')
```

```
def ressum():
```

```
    con = sql.connect("databasemark.db")
```

```
    con.row_factory = sql.Row
```

```
    cur = con.cursor()
```

```
    cur.execute("SELECT Критерій, SUM(Робота 1) "FROM mark ")
```

```

rows = cur.fetchall();
return render_template("finish.html", rows=rows)
@app.route("/result2")
def ressum2():
con = sql.connect("databasemark.db")
con.row_factory = sql.Row
cur = con.cursor()
cur.execute("SELECT Критеріі , SUM(Робота 2) " "FROM mark ")
rows = cur.fetchall();
return render_template("finish2.html", rows=rows)
@app.route("/result3")
def ressum3():
con = sql.connect("databasemark.db")
con.row_factory = sql.Row
cur = con.cursor()
cur.execute("SELECT Критеріі , SUM(Робота 3) " "FROM mark ")
rows = cur.fetchall();
return render_template("finish3.html", rows=rows)
@app.route("/result4")
def ressum4():
con = sql.connect("databasemark.db")
con.row_factory = sql.Row
cur = con.cursor()
cur.execute("SELECT Критеріі , SUM(Робота 4) " "FROM mark ")
rows = cur.fetchall();
return render_template("finish4.html", rows=rows)
@app.route("/result5")
def ressum5():
con = sql.connect("databasemark.db")
con.row_factory = sql.Row
cur = con.cursor()
cur.execute("SELECT Критеріі , SUM(Робота 5) " "FROM mark ")
rows = cur.fetchall();
return render_template("finish5.html", rows=rows)

@app.route("/enterinfo")
def newinfo():
return render_template('infozapus.html')

```

```

@app.route('/partform', methods=['POST', 'GET'])
def partform():
    if request.method == 'POST':
        try:
            Прізвище та імя = request.form['name']
            Дата народження = request.form['numb']
            Місце проживання = request.form['live']
            Назва роботи = request.form['robota']

            with sql.connect("databasemark.db") as con:
                cur = con.cursor()

                cur.execute("INSERT INTO info (Прізвище та імя, Дата народження, Місце проживання, Назва роботи)"
                    "VALUES(?, ?, ?, ?)", (Прізвище та імя, Дата народження, Місце проживання, Назва роботи))

                con.commit();

        except:
            con.rollback()

        finally:
            return render_template("resulti.html")
            con.close()

@app.route('/info-part')
def info():
    con = sql.connect("databasemark.db")
    con.row_factory = sql.Row

    cur = con.cursor()
    cur.execute("select * from info")

    rows = cur.fetchall();
    return render_template("infopart.html", rows=rows)
#додавання рядку в таблицю
def add():
    sqlite_connection = sqlite3.connect('databasemark.db')
    cursor = sqlite_connection.cursor()
    sql_delete_query = "ALTER TABLE mark ADD Робота 6 NUMBER"

```

```
cursor.execute(sql_delete_query)
sqlite_connection.commit()
#видалення рядку з таблиці
def delete():
    sqlite_connection = sqlite3.connect('databasemark.db')
    cursor = sqlite_connection.cursor()
    sql_delete_query = "DELETE from vector where Робота_1 = 0.2133874259"
    cursor.execute(sql_delete_query)
    sqlite_connection.commit()

if name == 'main':
    app.run(debug=True)
```