

НУБІП України

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

15.03 — КМР. 1862 — "С" 2021.11.01.08 ПЗ
НУБІП України

НАСЖИ АРТЕМА ОЛЕГОВИЧА

2022 р.

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

НУБІП України

Факультет інформаційних технологій

УДК
«ПОГОДЖЕНО» «ДОНУСКАЄТЬСЯ ДО ЗАХИСТУ»

Декан факультету

Завідувач кафедри комп'ютерних наук

інформаційних технологій

Глазунова О.Г., д.п.н., професор
Голуб Б.Л., к.т.н., доцент

202_р.

202_р.

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему Система підтримки прийняття рішень з організації
фотовиставок

Спеціальність 122 Комп'ютерні науки

Освітня програма Інформаційні управляючі системи і технології
(код і назва)

(назва)

Орієнтація освітньої програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Гарант освітньої програми
(науковий ступінь та вчене звання) (підпис) (ПІБ)

Керівник магістерської кваліфікаційної роботи

К. т. н., доцент.
(науковий ступінь та вчене звання) (підпис) Дудник Алла Олексіївна (ПІБ)

Виконав

(підпис)

Пасіка Артем Олегович (ПІБ студента)

КИЇВ-2022

ЗМІСТ

ВСТУП	4
1. СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	6
1.1 Постановка завдання	6
1.2 Огляд інформаційних джерел та існуючих рішень	7
2. МОДЕЛЮВАННЯ СИСТЕМИ	11
2.1 Діаграма прецедентів	11
2.2 Діаграма послідовності	14
2.3 Діаграма активності	14
2.4 Структура джерела інформації для проведення інтелектуального аналізу	15
3. РОЗРОБКА СИСТЕМИ	21
3.1 Архітектура системи	21
3.2 Інформаційне забезпечення	22
3.3 Розробка програмного забезпечення	27
4. РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ	37
4.1 Нечітка експертна система	37
4.2 Дослідження використання задач класифікації	47
4.3 Дослідження використання методу асоціативних правил	54
4.4 Дослідження використання алгоритмів кластеризації	58
ВИСНОВКИ	67
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	69
ДОДАТОК А	71
ДОДАТОК Б	73
ДОДАТОК В	75

ВСТУП

Соціальні мережі розвиваються дуже швидко. Вони охоплюють мільйони користувачів та тисячі авторів контенту. Зараз неможливо визначити найкращу пісню в Spotify чи найкрасивіше фото в Instagram. Та це і не потрібно. У кожного свої смаки. Але як бути з платформами, які створені спеціально для змагань, для визначення найкращого? Коли така платформа стає популярною і охоплює велику кількість контент мейкерів, тоді необхідно щось більше, ніж просто ручна оцінка.

Інформаційна система проведення фотоконкурсів може бути таким прикладом. Багато авторів завантажують свої роботи, яких може налічуватись тисячі. Для того, щоб оперативно проводити такі конкурси і не витрачати купу часу на перевірку кожної роботи була створена система підтримки прийняття рішень з організації фотовиставок.

Завдяки такій системі підтримки організатор зможе переглядати та аналізувати інформацію про взаємодію користувачів із системою. На основі цього аналізу можна буде зробити висновки, щодо тем, які цікаві користувачам в конкретний відрізок часу. Цю інформацію можна далі використовувати в маркетингових цілях для показу релевантної реклами або для ідеї створення найбільш актуальної фотовиставки. Але найголовніше, що завдяки оцінкам звичайних користувачів система сама прибере не цікаві роботи з якими було найменше взаємодії.

На основі середньої оцінки зображення, кількості переглядів та коментарів система за допомогою правил поведінки розділить всі роботи за 3 групи: відхилені, автоматично підтверджені зображення та роботи, що потребують додаткового розгляду. І вже організатори (журі) будуть самостійно оцінювати вузьке коло найкращих робіт.

Потрібно розуміти, що така діяльність веде до накопичення дуже значної кількості даних, які мають бути збереженими та доступними для подальшого аналізу. Для звичайної діяльності платформа використовує спосіб збереження даних через БД MySQL.

У подальшому, накопичивши дані, є можливість створення аналітичної складової, яка дозволить робити певні припущення та прогнози щодо діяльності системи, що в свою чергу може бути застосовано для зменшення витрат на підтримку роботи системи, збільшення прибутку, а також покращити досвід взаємодії користувача з системою.

Об'єкт дослідження: оцінка зроблених фотографій для конкурсу.

Предмет дослідження: система підтримки прийняття рішень з організації фотовиставок, яка допоможе оптимізувати процес оцінювання робіт.

Мета роботи: дослідити та проаналізувати ефективність зібраних даних системи підтримки прийняття рішень з організації фотовиставок.

Структура роботи. Магістерський проект складається зі вступу, чотирьох розділів (системного аналізу предметної області, моделювання системи, розробки системи та результатів досліджень), висновків, списку використаних джерел та 3 додатків. Записка складається з 70-ти сторінок. У роботі використано 14 джерел.

1. СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Постановка завдання

Ціль даної магістерської роботи – створити систему підтримки, яка допоможе більш точно та швидко оцінювати зображення в фотоконкурсах. На початку може здатися, що це марна ціль, тому що в інформаційній системі проведення фотоконкурсів є журі, яке і оцінює всі завантажені роботи. Проте, коли система стане загальновідома, то завантажених робіт може бути дуже багато, а наймати велику кількість журі недоцільно з фінансового боку. Для цього і необхідна система підтримки прийняття рішень, що допоможе звузяти коло найкращих фото робіт. А вже після цього в роботу вступить журі.

Першим етапом конкурсів є оцінка зображень звичайними глядачами, а вже після цього на основі таких даних, як середня оцінка користувачів, кількість переглядів та кількість коментарів будуть обрані найкращі роботи.

Розробити систему підтримки прийняття рішень, що проводить аналіз на основі даних користувачів, зібраних у результаті взаємодії з системою з ціллю виявлення переваг користувачів.

Дані мають описувати такі аспекти системи:

- інформація про зображення:

- назва конкурсу;
- середня оцінка зображення;
- дата завантаження зображення;
- кількість переглядів зображення

- категорія зображення

- інформація про користувача:

- вік;
- країна проживання;

- стать;

- дані про оцінювання зображень з боку користувачів. В рамках цієї системи розглядається варіант з оцінкою зображення по шкалі від 0 до

5 балів.

Має бути налаштований процес збору даних в операційну базу даних та подальшою передаючою цих даних в сховище даних, для подальшого аналізу за допомогою технології OLAP.

Проведення аналізу має давати відповідь на наступні питання:

1. Яке зображення має найбільше переглядів за вказаний проміжок часу?
2. В який конкурс завантажується найбільше робіт за вказаний проміжок часу?
3. Хто з авторів має найкращий рейтинг за вказаний проміжок часу?

1.2 Огляд інформаційних джерел та існуючих рішень

Для створення інформаційної системи необхідно переглянути якомога більше існуючих рішень та проаналізувати переваги та недоліки таких систем.

Це допоможе зробити систему конкурентоспроможною та може надати нових ідей для реалізації.

Існує велика кількість систем, які дають можливість завантажувати свої зображення в мережу та ділитися ними з оточуючими. Це загально відомі соціальні мережі та месенджери: Instagram, Telegram, Pinterest, Facebook та інші (рис. 1). Вони мають багатфункціональні системи для різних задач, але не заточені виключно під фотоконкурси.

Наприклад, в Instagram будь-хто може завантажити зображення, яке буде доступне для оцінки, поставити лайк або прокоментувати (рис. 2). Мінуси такої системи для конкурсу в тому, що система оцінювання досить примітивна. Якщо користувачу не сподобалось фото, то йому залишається просто не ставити оцінку. Також, в Instagram немає загального рейтингу по конкурсам і тому, неможливо виявити переможця. Переваги соціальної мережі в порівнянні з інформаційною системою проведення фотоконкурсів в тому, що вона має складні дорого вартісні алгоритми. В випадку з завантаженням зображення Instagram в автоматично перевіряє його на цензуру та моментально публікує (рис. 3). В нашій же системі зображення

спочатку проходить ручну перевірку адміністратором. Це може займати досить багато часу та негативно позначиться на користувацькому досвіді.

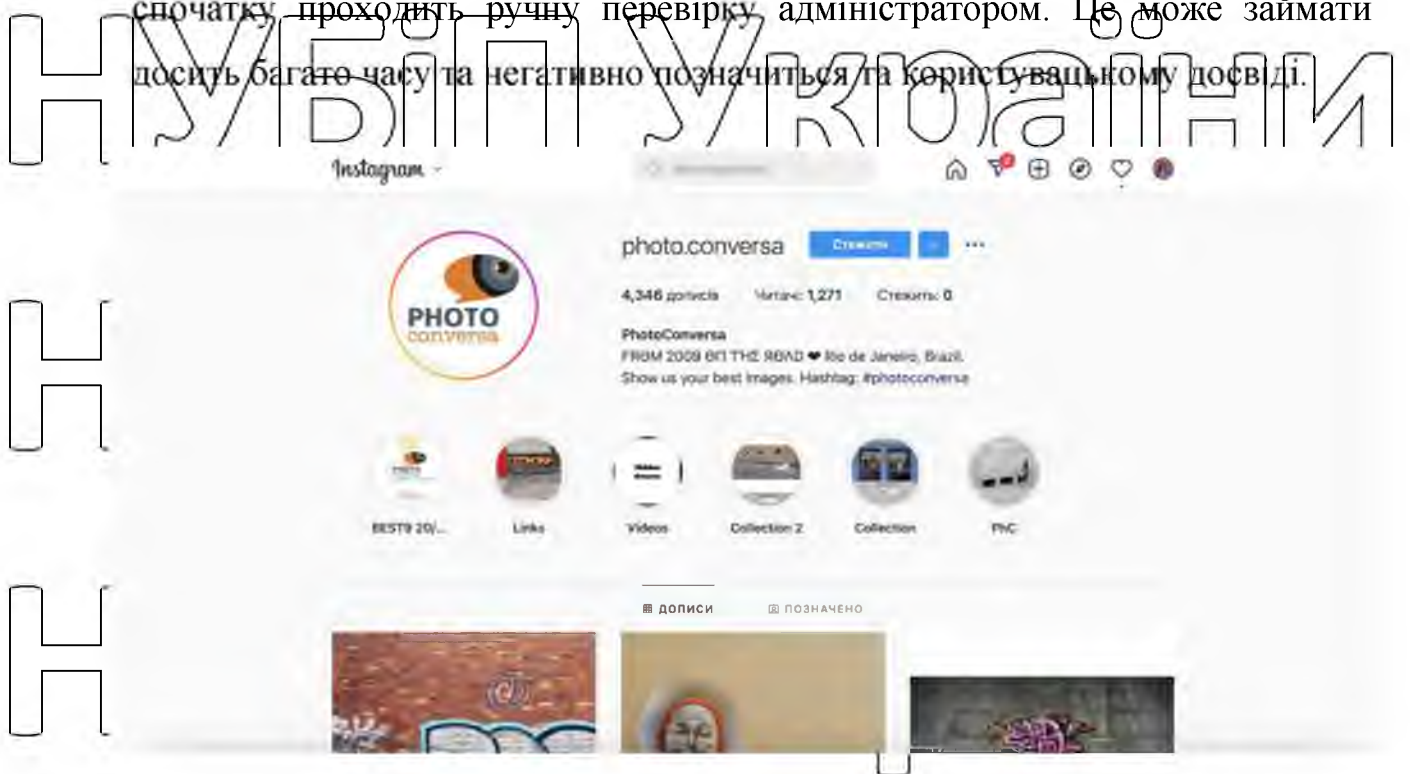


Рис. 1 Приклад аккаунту з фотоконкурсами в Instagram

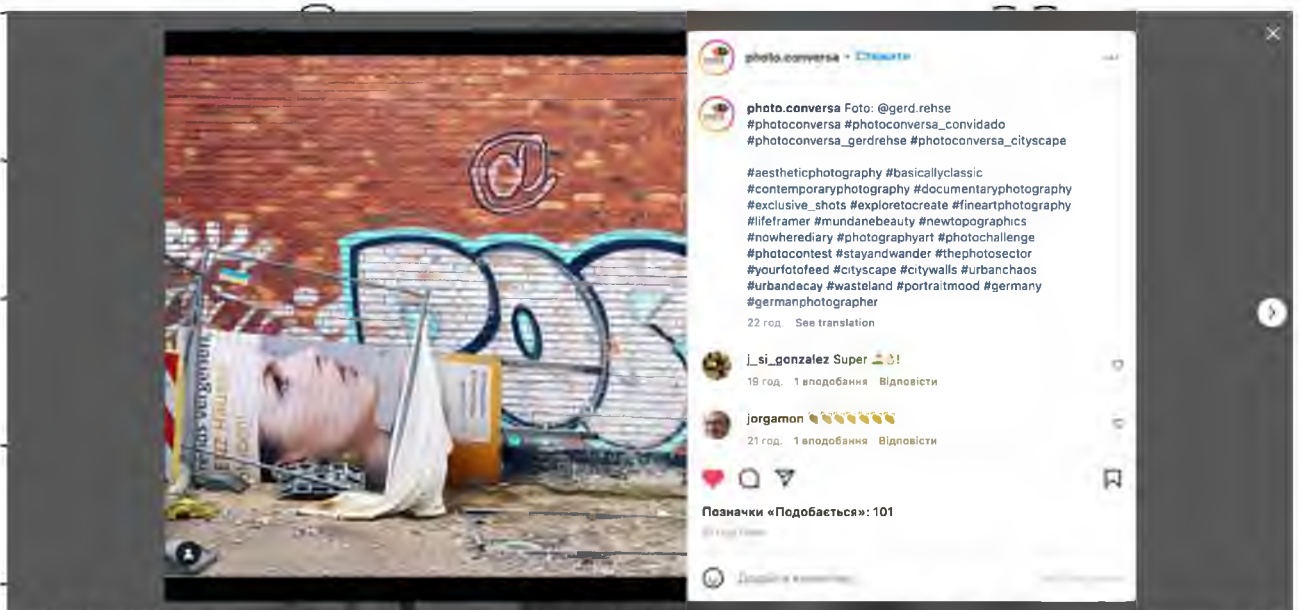


Рис. 2 Система оцінювання в мережі Instagram

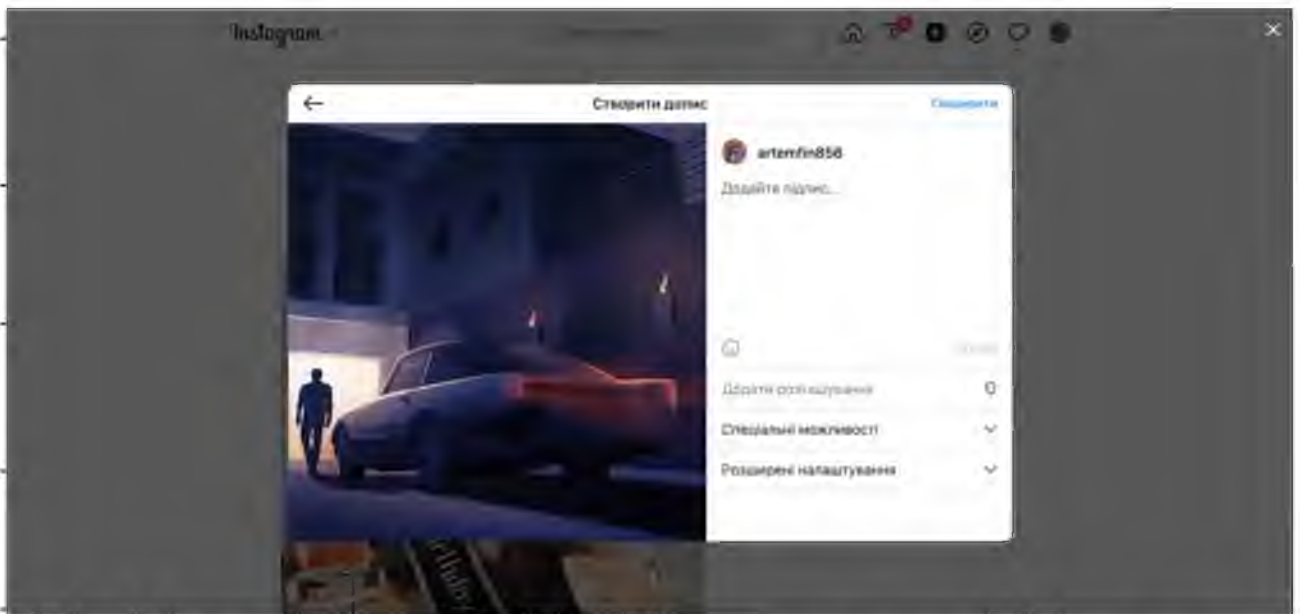


Рис. 3 Процес завантаження зображення в Instagram

В цифровому просторі нашого регіону ніша систем фотоконкурсів майже порожня і тому за прикладом іноземних інформаційних систем досить доцільним є створення системи, яка позиціонується виключно для участі та управління фотоконкурами. На рис. 4 та рис. 5 можна побачити приклад такої системи.

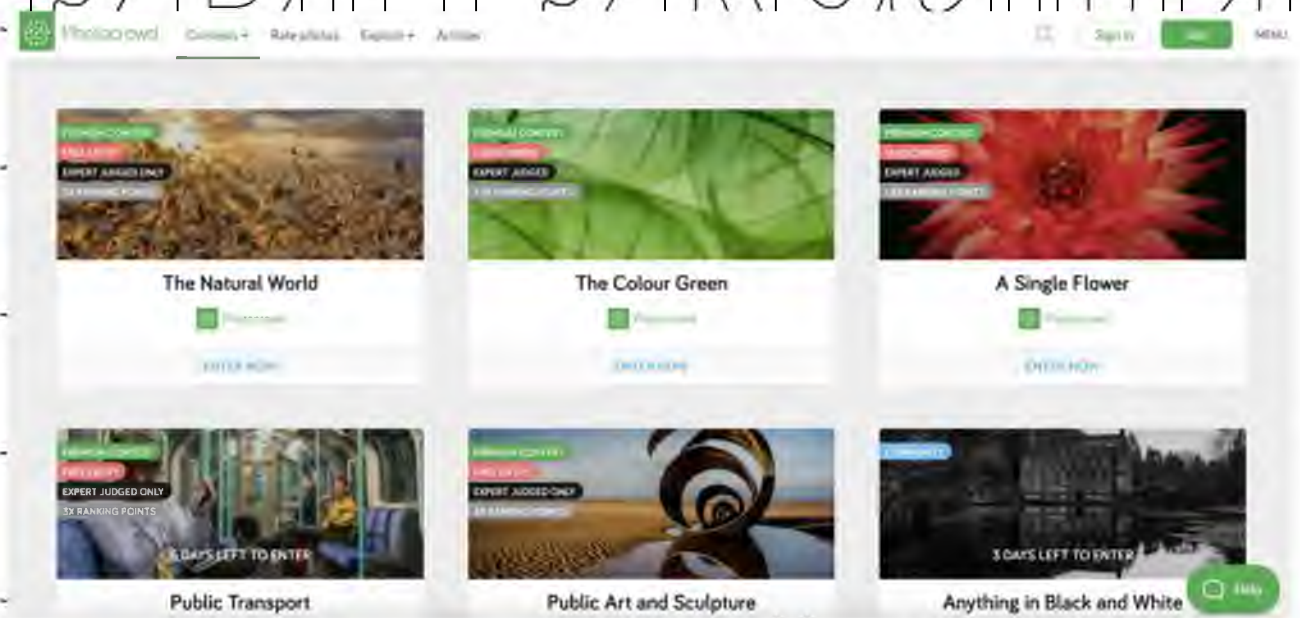


Рис. 4 Приклад системи фотоконкурсів



Рис. 5 Сторінка конкурсу з можливістю завантаження зображення
Створення такої системи не потребує чогось нового. Використовуються загальновідомі мови програмування та технічні засоби.

НУБІП України

НУБІП України

НУБІП України

НУБІП України

2. МОДЕЛЮВАННЯ СИСТЕМИ

2.1 Діаграма прецедентів

Важливим етапом роботи над системою є її попереднє дослідження та аналіз з метою виявлення її особливостей, задач, обмежень, сферу застосування та тощо. Одним з найбільш кращих способів проведення цієї роботи є моделювання системи.

Для моделювання системи було обрано використання об'єктно-орієнтованого підходу до моделювання системи, а якщо точніше побудова моделі за схемою діаграми прецедентів.

Цей спосіб передбачає створення схеми, яка буде включати такі основні компоненти:

1. актори – зовнішні сутності, що не належать системі, але взаємодіють з нею для реалізації якихось власних потреб, або участь якої необхідна в системі для коректного протікання процесів;
2. прецеденти – описує випадки або способи взаємодії акторів з системою з метою досягнення певних цілей цих акторів або коректного протікання процесів;
3. зв'язок між вище вказаними об'єктами, може бути, як між прецедентом та актором, так і між прецедентами, що може надати більш детальне представлення зв'язку між процесами в системі з точки зору акторів.

Перевагою використання цього способу моделювання системи є те, що цей підхід дозволяє розкрити групи користувачів системи, їх вимоги до функціоналу та чітко задати предметну область системи та виокремити головні процеси в ній, які мають бути реалізовані для задоволення потреб користувачів. Діаграма прецедентів для системи підтримки зображена на рис. 6.

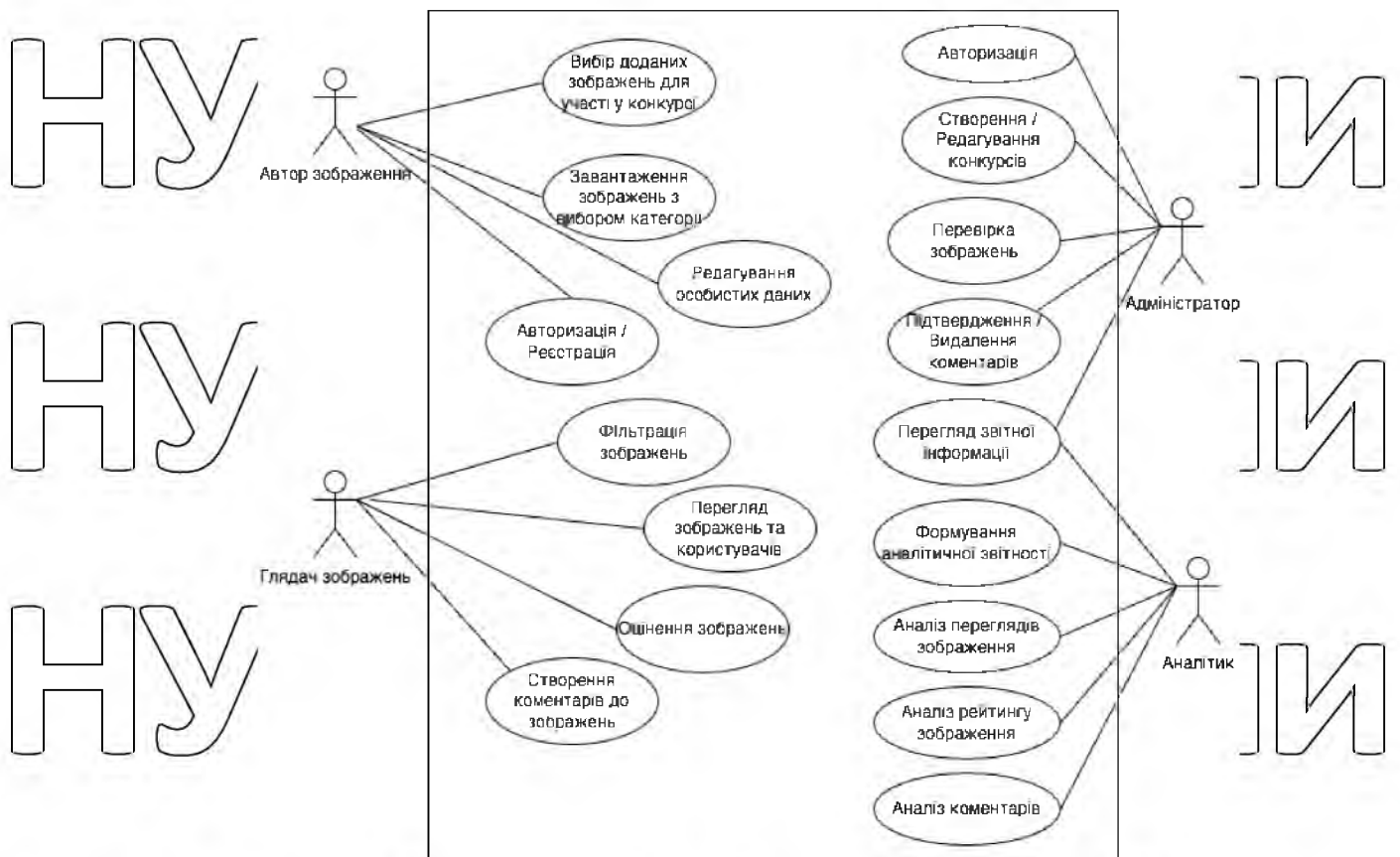


Рис. 6 Діаграма прецедентів

В рамках цієї системи можна виокремити таких акторів:

- Автор зображення – користувач системи, що створює та завантажує фото роботи в систему. Має особистий публічний обліковий запис, де інші користувачі системи можуть переглядати та оцінювати завантажені роботи.
- Глядач зображення – користувач системи, що може не мати облікового запису. Має можливість оцінювати та коментувати зображення авторів.
- Адміністратор – також, організатор фотовиставок. Може створювати, редагувати або видаляти конкурси. Модерувати зображення, користувачів та їх коментарі.
- Аналітик - очікує від системи надання аналітичних даних, щодо користувачів та їх діяльності на платформі.

В інформаційній системі проведення фото конкурсів можна виділити перелік прецедентів:

- авторизація / реєстрація Актором виступає користувач (автор

зображення). Даний прецедент надає можливість користувачу створити особистий кабінет для подальшого завантаження фото робіт.

- завантаження зображення. Актором виступає користувач (автор зображення). Даний прецедент надає можливість обрати категорію та завантажити зображення зі свого пристрою в особистий кабінет системи.

- вибір зображень для конкурсу. Актором виступає користувач (автор зображення). Даний прецедент надає можливість подати раніше завантажені зображення для участі у конкурс.

- редагування особистих даних. Актором виступає користувач (автор зображення). Даний прецедент надає можливість користувачу змінити дані, введені під час реєстрації.

- фільтрація зображень. Актором виступає користувач (глядач зображення). Даний прецедент надає можливість користувачу відфільтрувати всі зображення, що знаходяться в системі, так як він бажає.

- оцінка зображення. Актором виступає користувач (глядач зображення). Даний прецедент надає можливість користувачу оцінити будь-яке, додане в конкурс, зображення.

- створення коментарів. Актором виступає користувач (глядач зображення). Даний прецедент надає можливість користувачу робити відгуки про зображення.

- створення / Редагування конкурсів. Актором виступає адміністратор. Даний прецедент надає можливість створювати або змінювати конкурси.

- перевірка зображень та коментарів. Актором виступає адміністратор. Даний прецедент надає можливість перевірити перед публікацією, відправлени користувачами зображення та коментарі, на цензуру.

- формування аналітичної звітності. Актором виступає аналітик. Даний прецедент формує зібрані, за деякий період часу, дані у звіт.

- перегляд звітної інформації. Актором виступає адміністратор та аналітик. Даний прецедент надає можливість адміністратору переглядати звітну інформацію.

2.2 Діаграма послідовності

Діаграма послідовності – це один із варіантів формалізації сценаріїв використання. Її перевага в тому, що на ранніх стадіях опису сценаріїв можливо з'ясувати склад взаємодіючих компонентів та описати потік повідомлень від одних компонентів до інших. Ці компоненти та потоки повідомлень в подальшому будуть трансформовані в конкретні класи (об'єкти), методи цих об'єктів (якщо говорити термінологією мови Java). Відповідно, одразу ж з'ясується і модель системи подій (Actions), які дані класи (об'єкти) будуть підтримувати та обробляти [1]. На діаграмі послідовності об'єкти розміщені вгорі і від них вниз йдуть вертикальні лінії. Це лінії життя об'єкту. Вона являє собою фрагмент життєвого циклу об'єкта в процесі взаємодії. Кожне повідомлення представляється у вигляді стрілки між лініями життя двох об'єктів.

Діаграму послідовності для даної системи розміщено у додатку А.

2.3 Діаграма активності

При моделюванні поведінки системи виникає необхідність деталізувати особливості алгоритмічної та логічної реалізації операцій. Традиційно з цією метою використовують блок-схеми або структурні схеми алгоритмів. Кожна така схема акцентує увагу на послідовності виконання певних дій (або елементарних операцій), які в сукупності спричиняють до отримання бажаного результату.

Для моделювання процесу виконання операцій у мові UML використовують діаграму діяльності, яка зображується графом, вершинами якого є стани (дій і/або видів діяльності), а дугами – переходи від одного стану (дій/виду діяльності) до іншого стану (дій/виду діяльності) [2].

Діаграма діяльності до контрольного класу "Участь у конкурсі" зображена у додатку Б.

1. Передумови

Передпочатком роботи потоку, повинні бути виконані під-потоки "Авторизація користувача" та "Завантаження зображення до особистого кабінету

користувача".

2. Головний потік

2.1 Користувач обирає конкурс, де хоче прийняти участь

2.2 Вибір зображення, що буде відповідати правилам конкурсу

2.3 Очікування дозволу на публікацію зображення у конкурсі

2.4 Перевірка адміністратором, зображення на відповідність правилам конкурсу

2.5 Публікація зображення і конкурсі

3. Паралельні потоки

3.1 Оцінювання опублікованих зображень користувачами

3.2 Коментування зображень користувачами

3.3 Перевірка коментарів адміністратором

3.4 Оновлення даних на сайті

4. Альтернативні потоки

4.1 Зображення не відповідає правилам конкурсу (перехід до пункту 2.1 головного потоку)

4.2 Коментар не пройшов модерацию (перехід до пункту 3)

2.4 Структура джерела інформації для проведення інтелектуального аналізу

Для проведення аналізу потрібно опрацювати вхідні дані, створити сховище даних, наповнити та розгорнути його.

Для розгортання кубу на першому етапі необхідно визначити джерело даних. На основі джерела даних будуть імпортуватись необхідні дані. Формуємо джерело даних за допомогою модуля Data Source Wizard, де обрано створене сховище даних.

Другий етап побудови куба – це створення уявлення джерела даних. На рис. 7.8 зображено додавання уявлення за допомогою Data Source View Wizard.

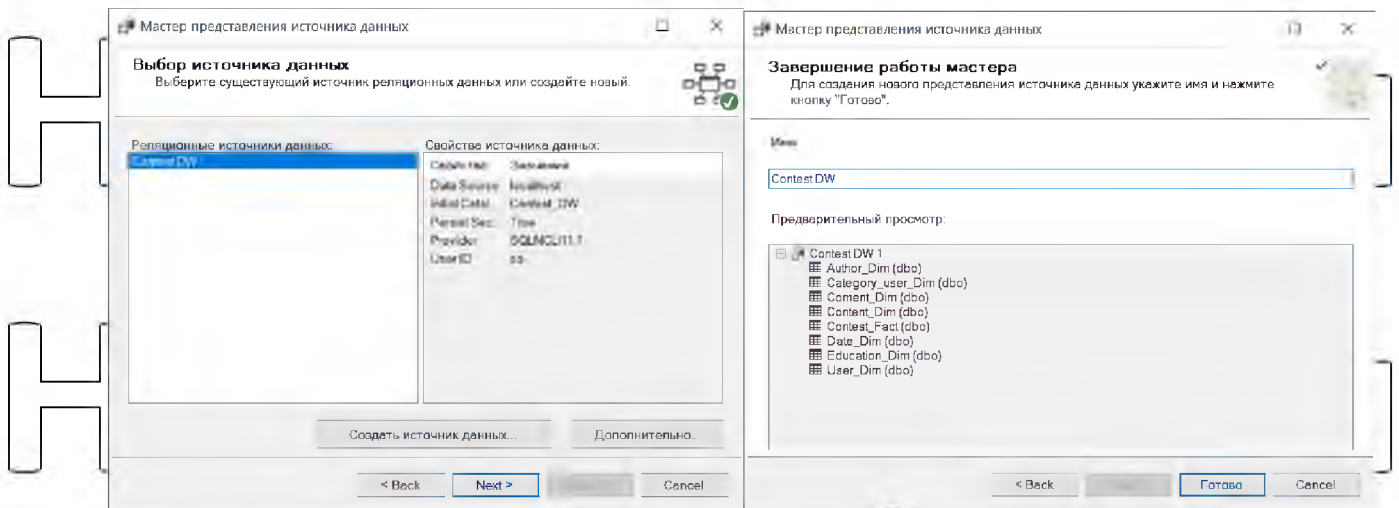


Рис. 7-8 Створення уявлення на основі СД Contest_DW

На основі створеного джерела та уявлення джерел даних необхідно створити виміри для OLAP куба. На рис. 9 зображено процес створення часового виміру.

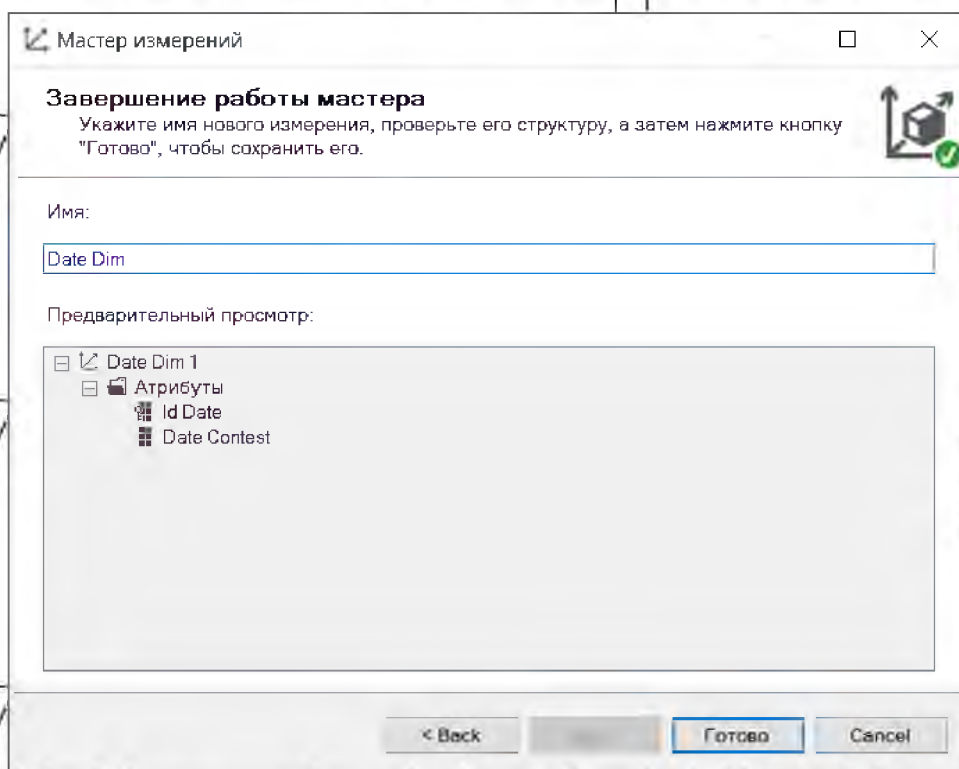


Рис. 9 Створення часового виміру (DateDim)

Наступними вимірами є вимір контенту, створення якого зображено на рис. 10. Цей вимір пов'язаний з вимірами автора та освіти (рис. 11-12), які надалі будуть використовуватись для аналізу фактів.

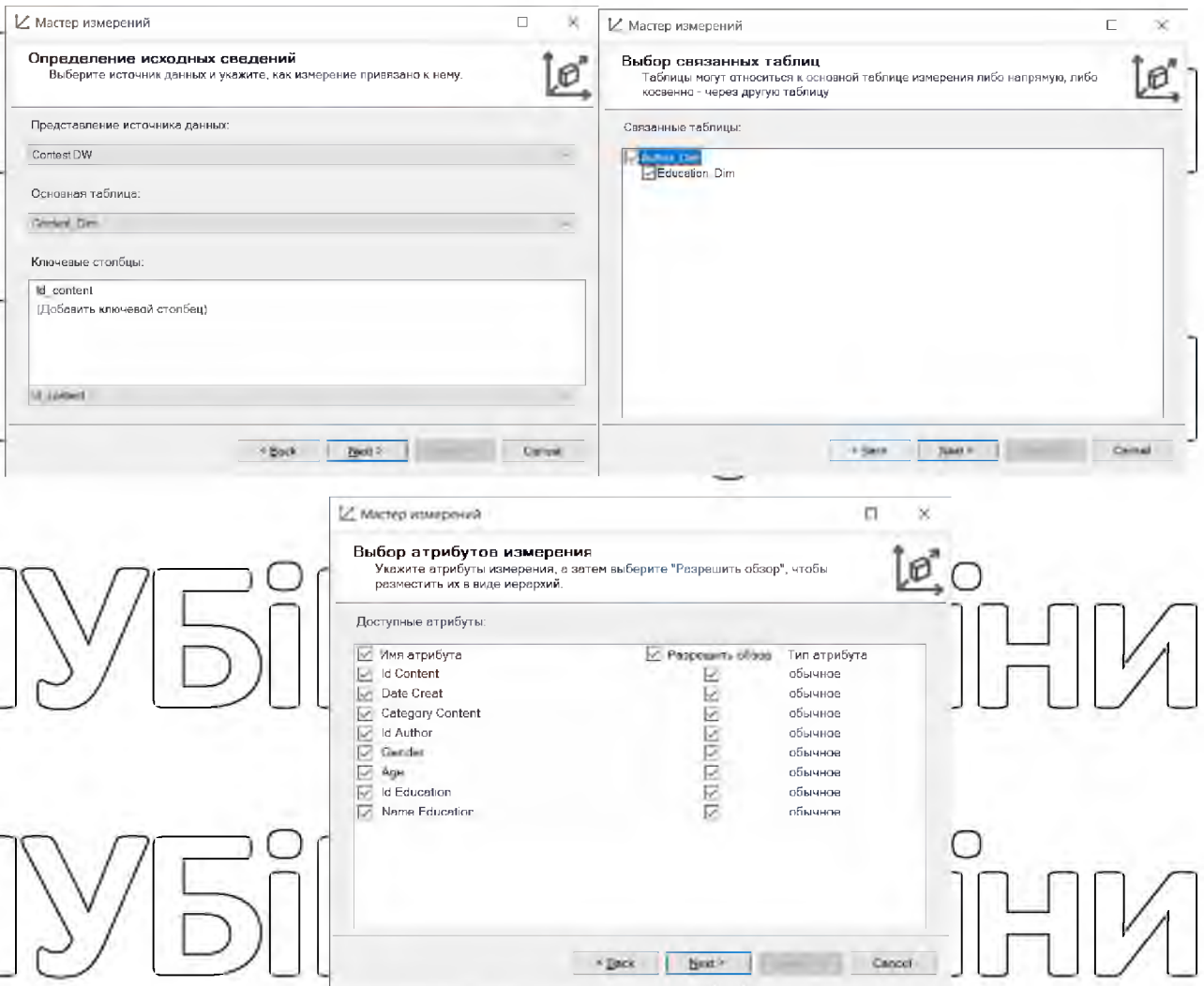


Рис. 10-12 Створення виміру ContentDim та пов'язаних з ним AuthorDim і EducationDim

Наступним виміром є вимір коментарів (рис. 13), який є одним з ключових вимірів. З цим виміром також пов'язані вимір користувача та категорія (рис. 14-15), до якої користувачі відносяться. Надалі ці оцінки користувачів використовуються для аналізу результатів виставки.

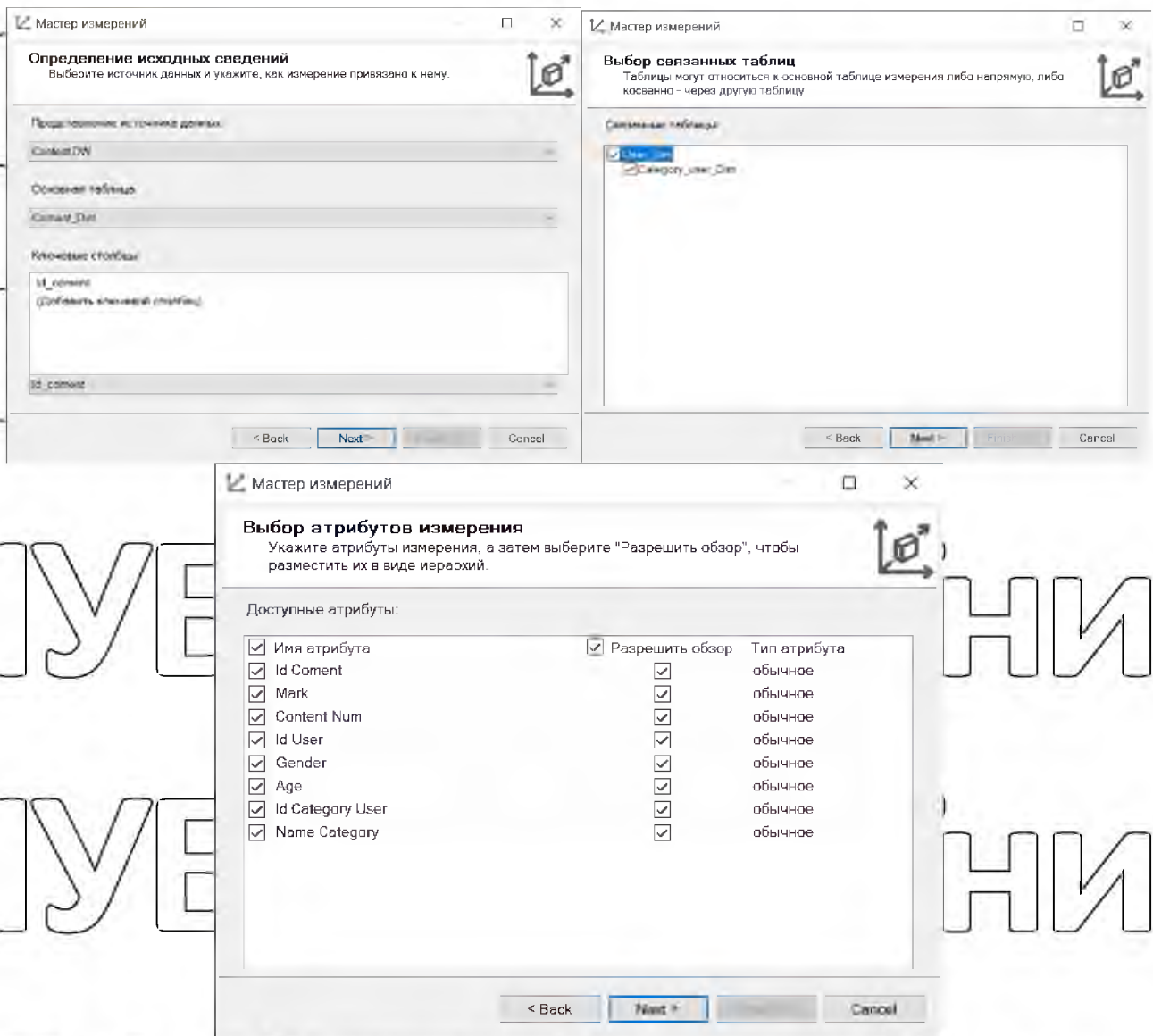


Рис. 13-15 Створення виміру CommentDim та пов'язаних з ним UserDim і CategoryUserDim

Після створення всіх необхідних вимірів наступним кроком є створення кубу OLAP за допомогою майстра кубів (Cube Wizard). На рис. 16 зображено створення кубу. На першому етапі оберемо таблицю фактів, яка виступає основою для створення кубу.

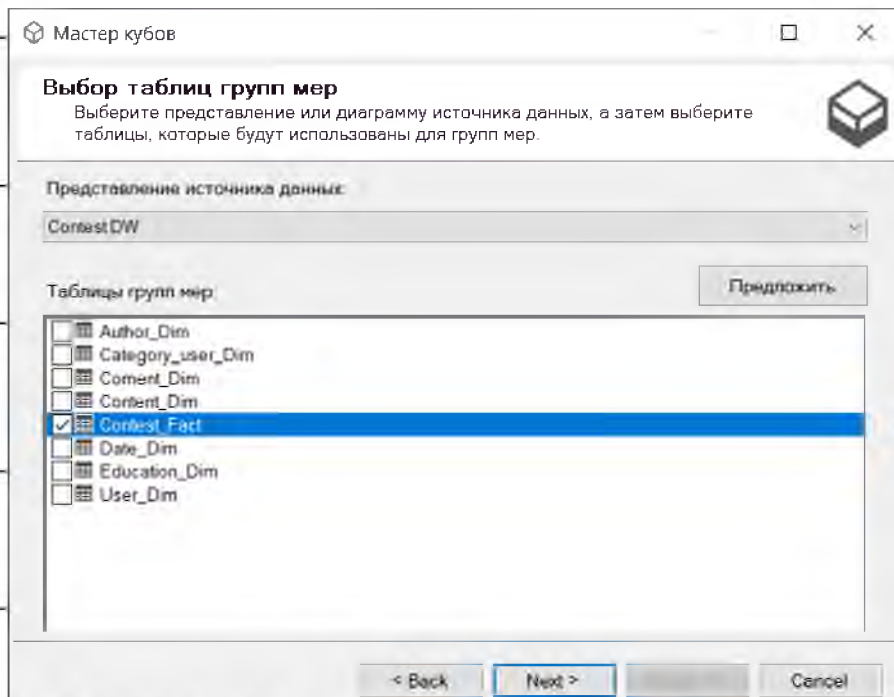


Рис. 16 Вибір таблиці для створення кубу

Наступним етапом йде вибір полів та вимірів, які формуватимуть куб (рис.

17-18).

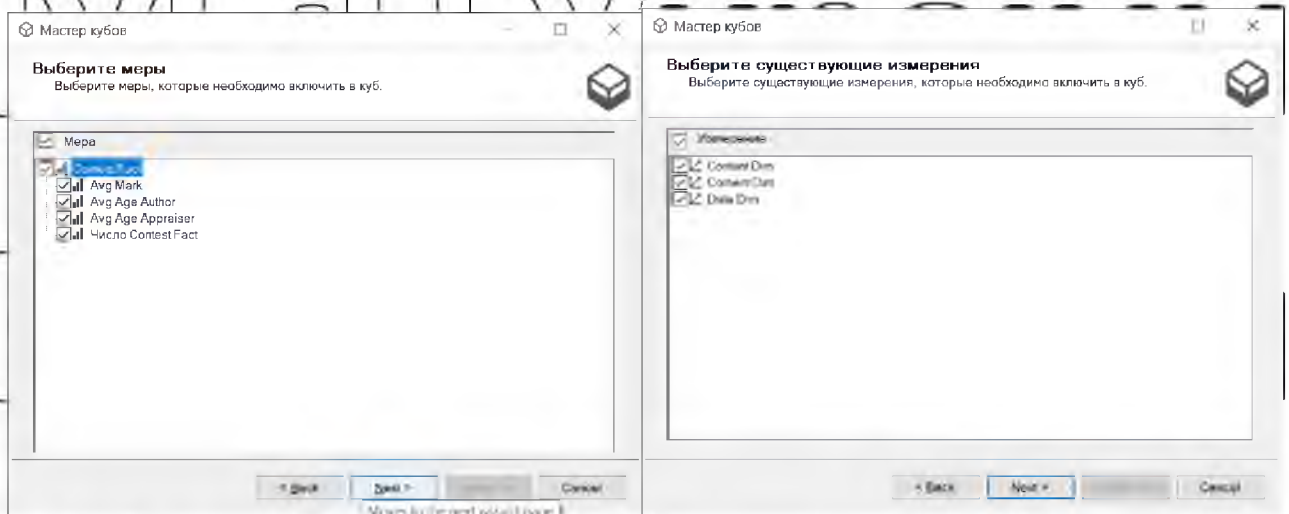


Рис. 17-18 Формування кубу

Після виконання всіх попередніх етапів результатом є створений Contest.DW, що зображений на рис. 19. Він складається з 7 вимірів та однієї таблиці фактів. Серед 7 вимірів 4 з них є консольними і допоможуть у додатковому аналізі отриманих даних у таблиці фактів.

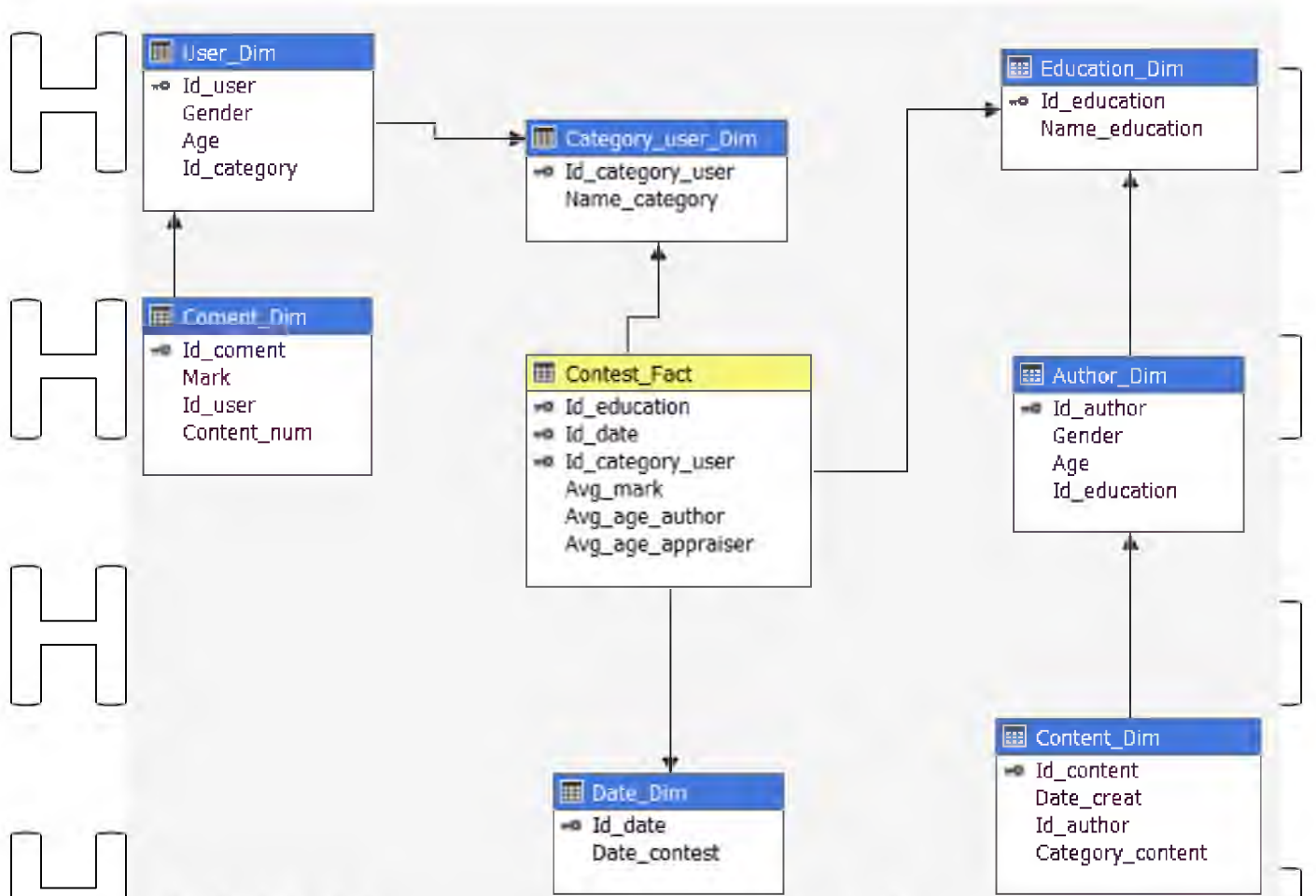


Рис. 19 Створений куб

3. РОЗРОБКА СИСТЕМИ

3.1 Архітектура системи

На рис. 20 зображено архітектуру системи. В ній можна виділити наступні частини:

- сервер (хостинг), на якому розміщена інформаційна система з якою і буде взаємодіяти користувач
- пристрій користувача з підтримкою сучасного веб-браузера через який відбувається взаємодія з системою;
- сервер БД, на якому завантажена обрана СУБД та реалізована оперативна БД;
- сервер СД, на якому реалізоване СД інтелектуальної системи;
- середовище OLAP куба; робоча станція аналітика, на якому встановлений модуль аналітики.

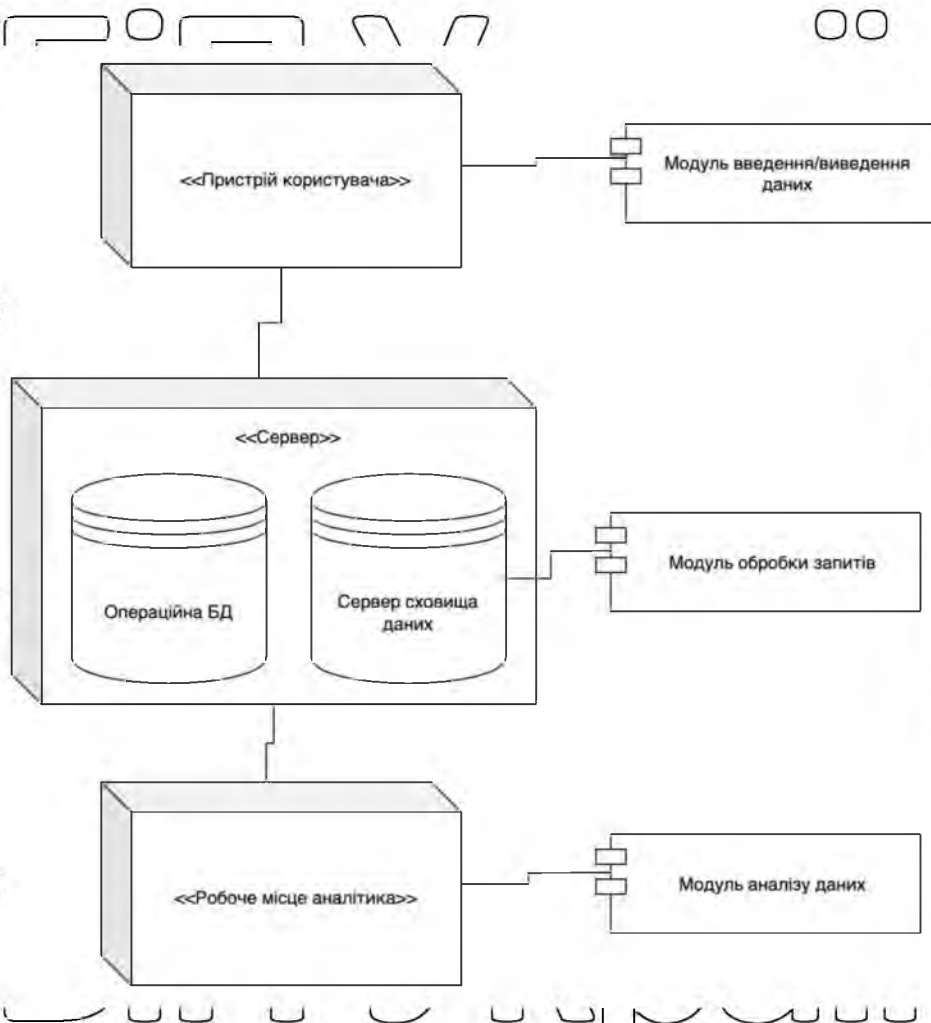


Рис. 20 Топологія системи

3.2 Інформаційне забезпечення

3.2.1 Операційна база даних. Для реалізації інформаційної системи було обрано СУБД MySQL. MySQL – це вільна система керування реляційними базами даних. Вона використовується, в першу чергу, для створення динамічних веб-сторінок, оскільки має чудову підтримку з боку різноманітних мов програмування [3]. Логічну модель представлено на рис. 21.

В діаграмі сутностей системи представлено 7 сутностей:

- **user.** В даній таблиці зберігаються інформація про зареєстрованого користувача, а саме: email, ім'я, пароль в зашифрованому вигляді, логін, дата реєстрації, короткий слоган та url назва фото профілю.
- **rating.** В даній таблиці розміщені всі оцінки, які робили користувачі. В ній містяться атрибути таблиць user та content, як зовнішні ключі та зберігаються такі значення: оцінка зображення та дата оцінювання.
- **content.** Таблиця містить дані про всі завантажені користувачами зображення, а саме: url назви зображення, дата завантаження, короткий опис, кількість переглядів, статус зображення, а також зовнішні ключі таблиць user, contest та category.
- **contest.** В цій таблиці міститься інформація про фотоконкурс, а саме: статус конкурсу, його назва, опис та дата закінчення.
- **category.** В даній таблиці зберігаються дані про категорію зображень, а саме назва категорії.
- **comment.** Таблиця містить зовнішній ключ атрибутів таблиці content. В ній зберігаються такі дані: ідентифікатор користувача, що написав коментар, зміст коментаря, дата створення та його статус.
- **settings.** Тут зберігаються дані для входу адміністратора та деякі налаштування системи. Для зручності зміни цієї інформації все це було зроблено через запити. Тобто, таблиця має два атрибути: set_key (тут зберігаються назви змінних, які в подальшому вказуються при створенні функціоналу системи) та set_value (в даному атрибуті зберігаються значення змінних).

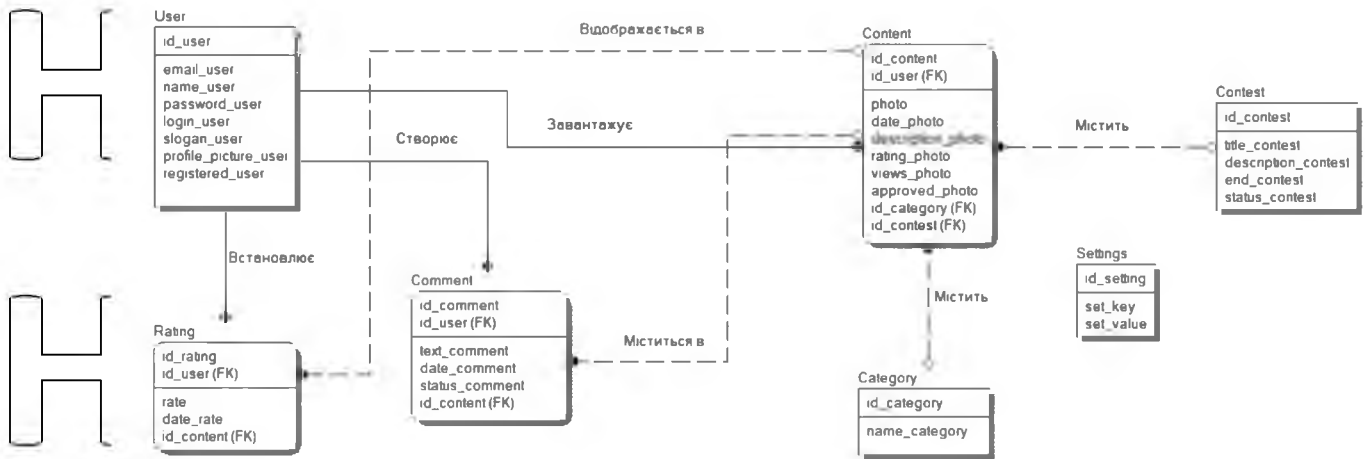


Рис. 21 Логічна модель даних

Для створення таблиць використовується команда «Create table» після якої вказується назва нової таблиці. Далі вказуються значення таблиці та їх типи. NOT NULL вказує на те, що значення не може бути пустим. PRIMARY KEY використовується для однозначної ідентифікації рядків в таблиці. Значення первинного ключа має бути завжди унікальним і не повторюватись. Приклад створення таблиці users на рис. 22

```

CREATE TABLE IF NOT EXISTS `users` (
  `id` int(1) NOT NULL AUTO_INCREMENT,
  `email` varchar(50) NOT NULL,
  `name` varchar(30) NOT NULL,
  `password` varchar(128) NOT NULL DEFAULT '',
  `user` varchar(30) NOT NULL DEFAULT '',
  `registered` timestamp NOT NULL DEFAULT current_timestamp(),
  `slogan` varchar(50) NOT NULL DEFAULT '',
  `profile_picture` varchar(50) NOT NULL DEFAULT '',
  PRIMARY KEY (`id`),
  KEY `email` (`email`),
  KEY `password` (`password`),
  KEY `user` (`user`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;
    
```

Рис. 22 Створення таблиці users

Для поточної системи було створено 7 таблиць. Запити на створення всіх таблиць можна побачити у додатку В.

3.2.2 OLAP-технології. OLAP – технологія оперативної аналітичної

обробки даних, що використовує методи та засоби для збору, зберігання та аналізу багатомірних даних. Основне призначення OLAP-систем – підтримка аналітичної діяльності, довільних запитів користувачів-аналітиків. Ціль OLAP-аналіза – перевірка виникаючих гіпотез [4].

Куб OLAP – структура даних, що забезпечує швидкий аналіз даних за межами можливостей звичайної реляційної БД. Куб може відображати та підсумовувати великі об'єми даних та надавати користувачам доступ до будь-яких даних із можливістю пошуку. Завдяки такій гнучкості структура дозволяє проводити досить складні маніпуляції над даними, а саме об'єднання, фрагментування та обробка по мірі необхідності для вирішення великого спектра питань. Загалом, OLAP куби – ланка, що представляє собою фінальну складову рішення по створенню та обслуговуванню СД [5].

Однією з корисних функцій куба OLAP – дані можуть зберігатись в статистичному вигляді. З точки зору користувача це має такий вигляд, що в кубі уже всі необхідні відповіді, так як структура містить чимало попередньо розрахованих даних.

Загалом є три основних способи реалізації:

- MOLAP (Multidimensional OLAP) – для реалізації багатомірних моделей використовують багатомірні БД;
- ROLAP (Relational OLAP) – для реалізації багатомірних моделей використовують реляційну БД;
- HOLAP (Hybrid OLAP) – для реалізації багатомірних моделей використовують і багатомірні і реляційні БД.

Розглянемо детальніше MOLAP, адже саме така архітектура OLAP-сервера використовується в даному проекті. MOLAP-сервери використовують для зберігання та керування даними багатомірної БД. При цьому дані зберігаються у вигляді впорядкованих багатомірних масивів. Прикладом таких масивів є гіперкуби та полікуби.

Переваги використання багатомірних БД в OLAP-системах:

• пошук та відбір даних відбувається значно швидше, ніж при багатомірному концептуальному погляді на реляційну БД, адже багатомірна БД денормалізована та містить раніше агреговані показники, забезпечуючи оптимізований доступ до заданих даних та не потребуючи додаткових перетворень при переході від множини пов'язаних таблиць до багатомірної моделі;

• багатомірні БД легко справляються із завданнями включення в інформаційну модель різноманітних вбудованих функцій. З іншої сторони, дана реалізація має і кілька недоліків:

• за рахунок денормалізації та попередньої агрегації об'єм даних в багатомірній БД в рази більший в порівнянні з реляційною БД;

• у переважній більшості випадків інформаційний гіперкуб дуже рідкісний, і оскільки дані зберігаються впорядковано, невизначені значення можуть бути видалені лише шляхом вибору оптимального порядку сортування, який дозволяє організувати дані в максимально можливі безперервні групи;

• багатовимірні бази даних чутливі до змін багатовимірної моделі. Отже, при додаванні нового виміру доводиться змінювати структуру всієї бази даних, що тягне за собою багато часу.

3.2.3 Сховище даних. В першу чергу, на рівні операційної бази даних має бути забезпечено отримання актуальної та достовірної інформації, а також, її редагування з мінімальною затримкою та уникненням помилок в цих даних.

Для цих цілей ми проведемо нормалізацію даних, суть якої в розбитті бази даних на набір зв'язаних таблиць кожна з яких буде містити тільки необхідний об'єм даних.

При роботі з такою базою даних в OLTP підході, який зазвичай вимагає заціпити невеликі об'єми даних таке рішення є вдалим і показує високу ефективність (при умові уникнення надмірної нормалізації). Проблеми виникають, коли нам потрібно проводити аналіз даних, який вже буде вимагати використання великого об'єму даних, що в свою чергу призведе до необхідності

з'єднання великої кількості таблиць і призведе до низької ефективності таких операцій.

Тому виникає потреба в створенні так званого сховища даних, яке буде слугувати для накопичення даних, які були отримані з операційних баз даних.

Так, як сховища даних служать для проведення аналітики і будуть вимагати обробку великої кількості агрегованих даних, то при проектуванні цього сховища проводить денормалізація [6].

Мета концепції сховищ даних - прояснити відмінності в характеристиках даних в операційних і аналітичних системах, визначити вимоги даним, що поміщаються в сховище, визначити загальні принципи і етапи його побудови, основні джерела даних, дати рекомендації по рішенняю потенційних проблем, що виникають при їх вивантаженні, очищенні, узгодженні, транспортуванні і завантаженні до цільової базу даних сховища.

Предметом концепції сховищ даних є не аналіз даних, а власне дані, тобто концепція їх підготовки для подальшого аналізу. В той же час концепція сховища даних визначає не просто єдиний логічний погляд на корпоративні дані, а реалізацію єдиного інтегрованого джерела даних [7].

Іншою особливістю сховища є те, що на відміну від операційних баз даних при нормальному використанні дані лише додаються, і ніколи не оновлюються і тим більш видаляються.

Зважаючи на особливості сховищ даних, а також їх значення для системи необхідно мати уявлення, щодо задач для розуміння необхідних вимог до ефективності запитів та збереження даних в сховищу.

Для системи підтримки прийняття рішень з організації фотовиставок було розроблено сховище даних (рис. 23) в середовищі SQL Server.



Рис. 23 Сховище даних

3.3 Розробка програмного забезпечення

Головна сторінка клієнтської частини системи містить в собі навігаційне меню та категорії зображень, які йдуть табами. В кожному табі розміщені зображення відповідні до конкретної категорії. Зображення головної сторінки

на рис. 24. З цієї сторінки користувач може перейти на конкретне зображення,

авторизуватись чи зареєструватись або перейти на будь-яку сторінку, вказану в навігаційному меню

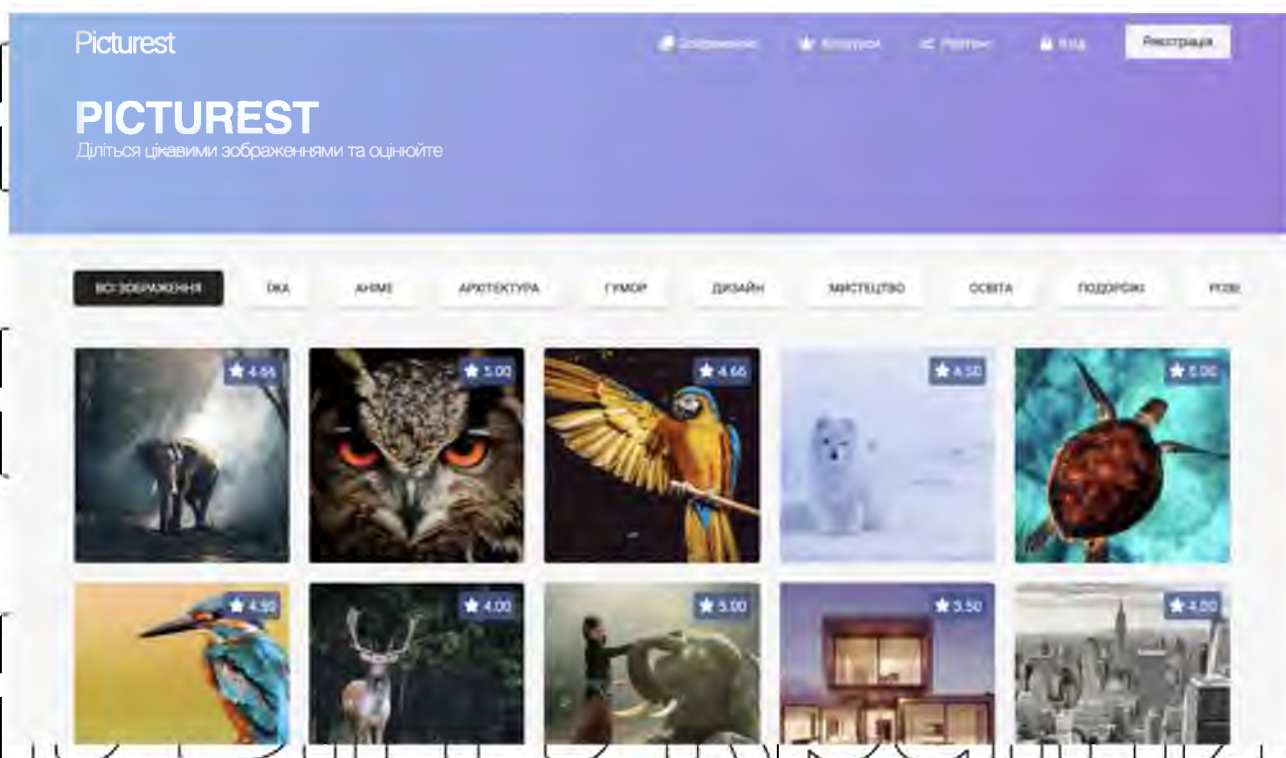


Рис. 24 Головна сторінка

Для того, щоб мати змогу приймати участь у конкурсах і завантажувати свої роботи, автор повинен зареєструватися в системі. Після авторизації автор зможе завантажувати зображення в особистий кабінет, а після цього розміщувати їх у відповідні конкурси (рис. 25-26).

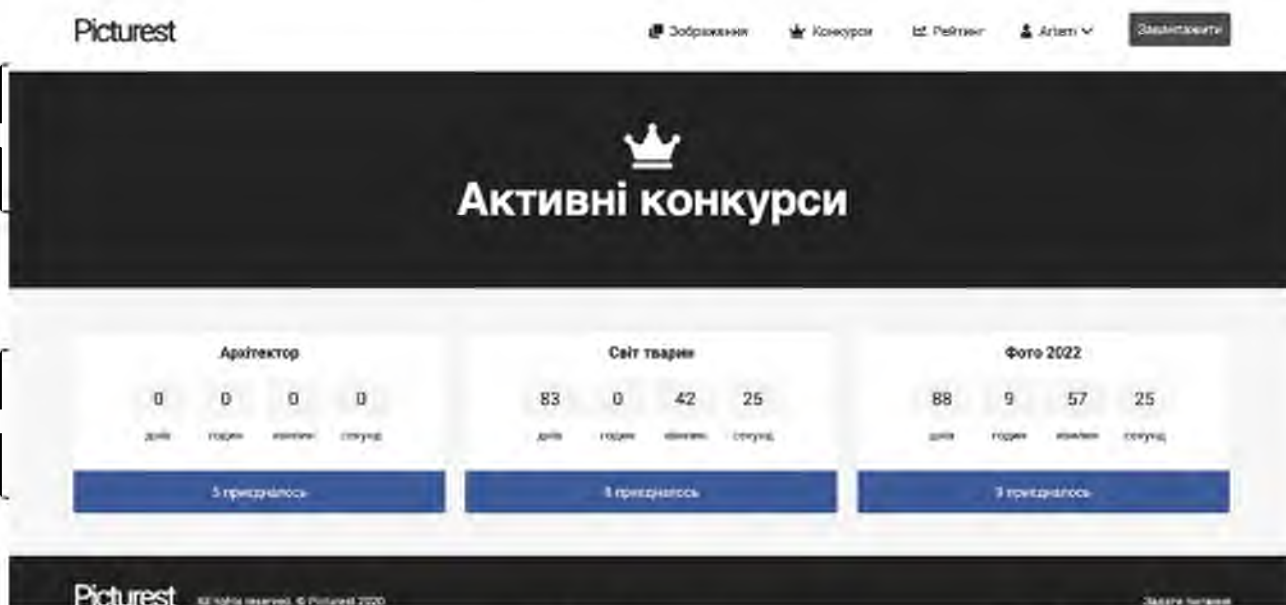


Рис. 25 Вибір конкурсу

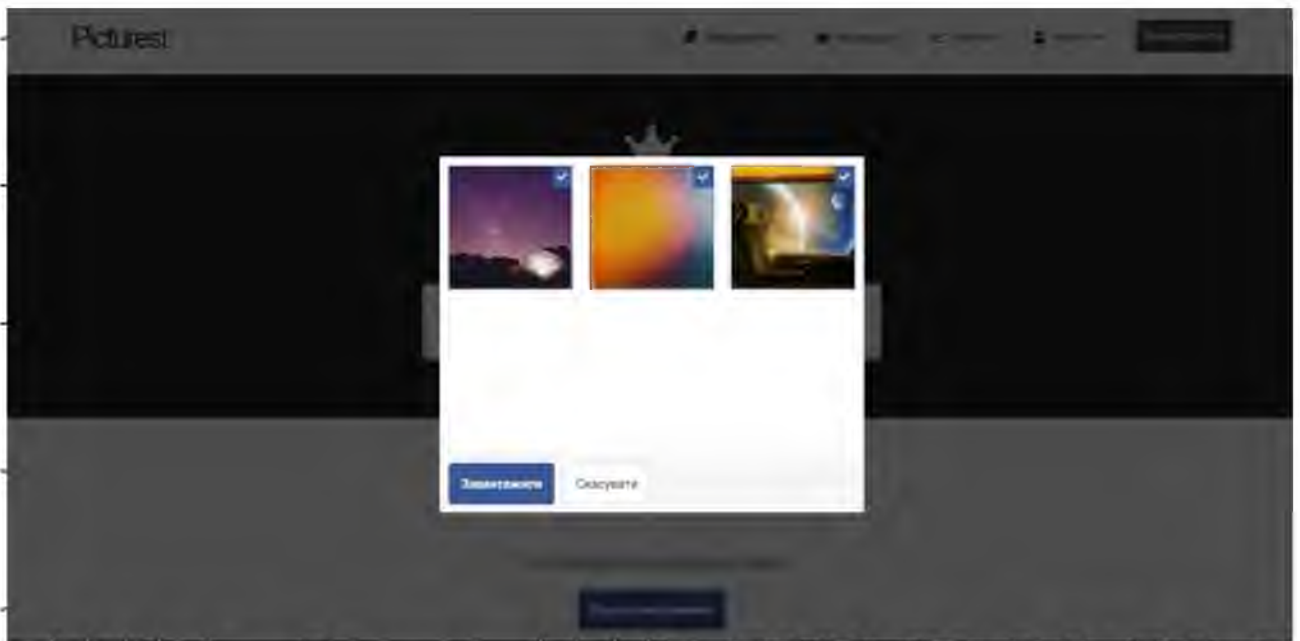


Рис. 26 Вибір зображень для подання на конкурс
Коли зображення приймає участь у конкурсі, його можна оцінити.

Зробити це можливо, якщо перейти на сторінку зображення (рис. 27).



Рис. 27 Оцінення зображення

Це однією функціональною можливістю для користувачів є коментування зображень. Авторизований користувач має можливість залишити відгук під будь-яким зображенням. Як і з завантаженням зображень, після написання коментаря з'явиться повідомлення про публікацію після модератії адміністратором. Цей процес зображено на рис. 28 та рис. 29.

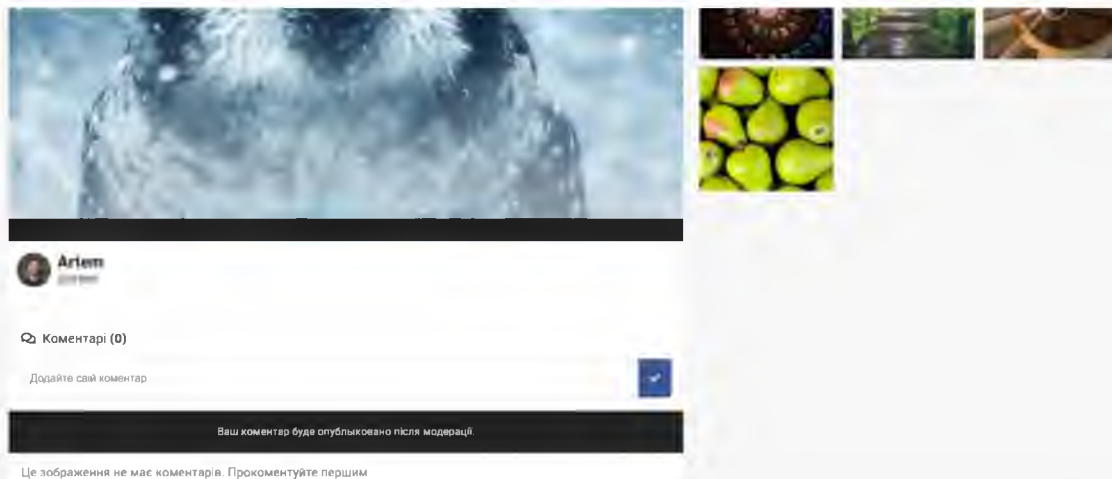


Рис. 28 Повідомлення про модерацію коментаря



Рис. 29 Опубліковані коментарі

На основі таких даних, зроблених користувачами, як середня оцінка, кількість переглядів та кількість коментарів, система підтримки прийняття рішень проводить певну модерацію всіх робіт, завдяки створеному алгоритму поведінки. Зображення з високим результатом автоматично проходить цю модерацію, з середнім рівнем відправляється на додатковий розгляд, а з низьким – автоматично відхиляється для подальшої участі у конкурсі.

Після того, як у зображенні з'явиться хоча б одна оцінка, воно з'явиться на сторінці «Рейтинг». Тут користувач може побачити легальну

інформацію по кожному конкурсу: позицію, середню оцінку, кількість оцінок та кількість переглядів. Також, є можливість фільтрації по конкурсах, категоріях та сортування за рейтингом, кількістю оцінок та переглядів (рис. 30).



Рис. 30 Сторінка «Рейтинг»

Адміністративна панель складається з навігаційного меню та основної частини, де розміщується вся інформація. На кожній сторінці знаходяться дані та елементи управління кожною таблицею БД. Першою сторінкою, після авторизації, яку побачить адміністратор буде «Користувачі» (рис. 31). Тут можна переглянути таку інформацію про користувачів, як: фото профілю, ім'я, логін, електронну пошту та дату реєстрації. Також, адміністратор може редагувати дані користувачів або і зовсім їх видаляти.

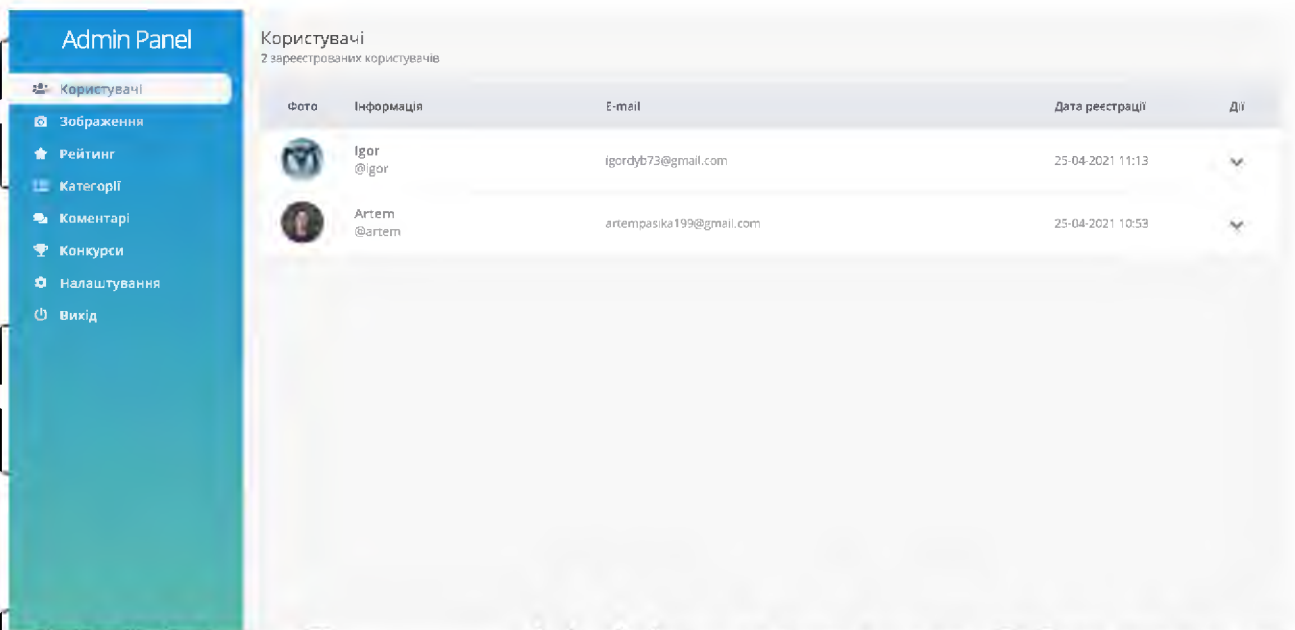


Рис. 31 Сторінка «Користувачі» адміністративної частини системи

На сторінці «Зображення» розміщуються всі роботи авторів, які пропустила система підтримки прийняття рішень. Вони розділяються на 2

вкладки:

- «затверджено» На цій вкладці знаходяться всі зображення, що набрали високий рівень в системі підтримки (рис. 32). Тепер організатор оцінює ці роботи за таким же принципом, як і користувачі (рис. 33). Проте, у організатора є можливість вручну переглянути всі роботи і за необхідності видалити не підходящі.

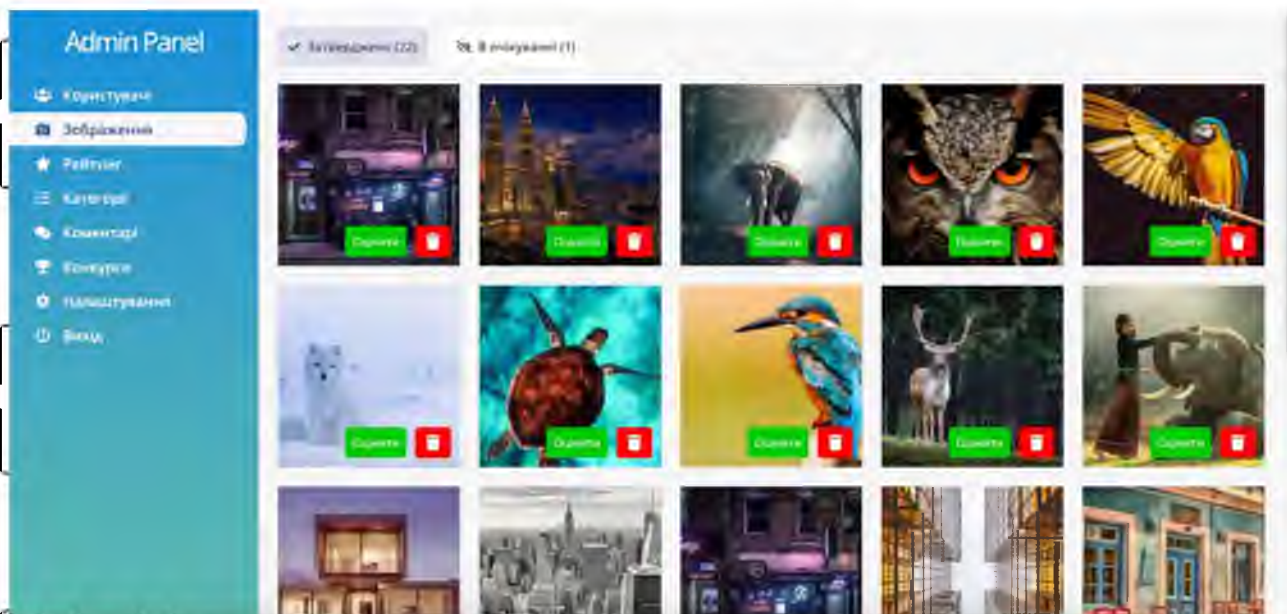


Рис. 32 Сторінка опублікованих зображень

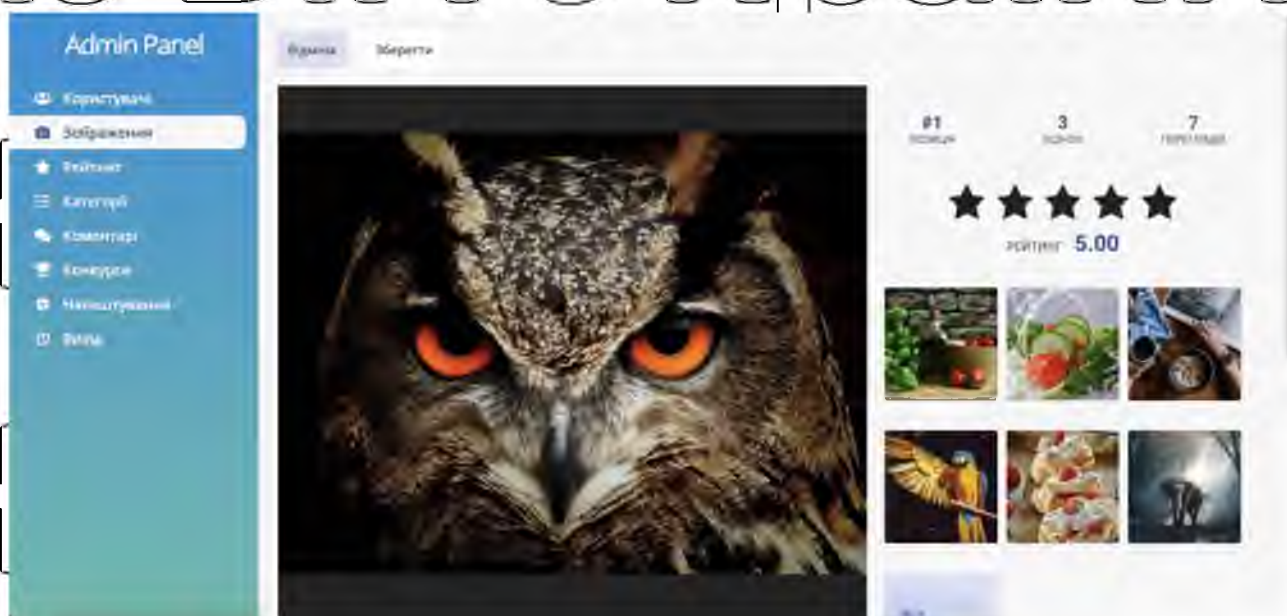


Рис. 33 Сторінка оцінки зображення організатором

• «в очікуванні». Сюди потрапляють зображення, що отримали середній рівень в системі підтримки. Вони потребують ручного розгляду. У організатора є два варіанти: затвердити або видалити зображення. У випадку затвердження, зображення переміститься у першу вкладку.

Процес затвердження зображення на рис. 34.

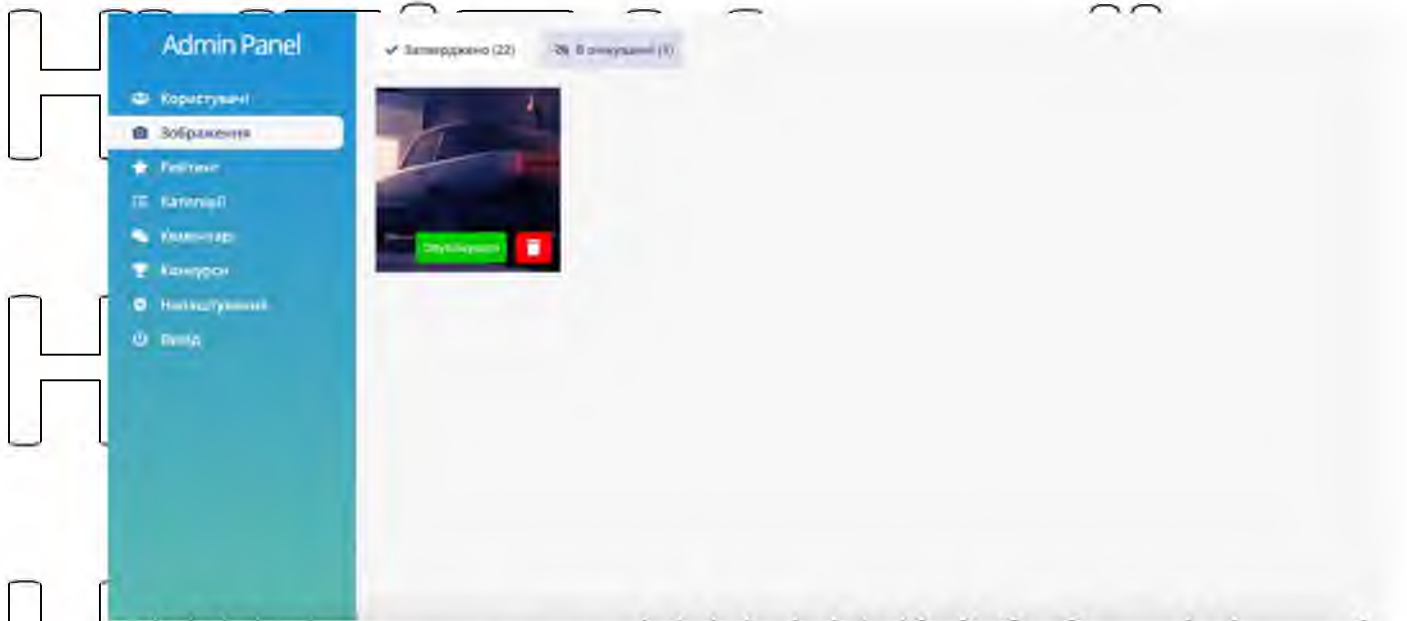


Рис. 34 Сторінка зображень в очікуванні

На сторінці «Рейтинг» виводяться всі оцінки зображень, зроблені користувачами, та час коли вони були зроблені (рис. 35). Також, організатор може видалити всі, або якісь конкретні оцінки.

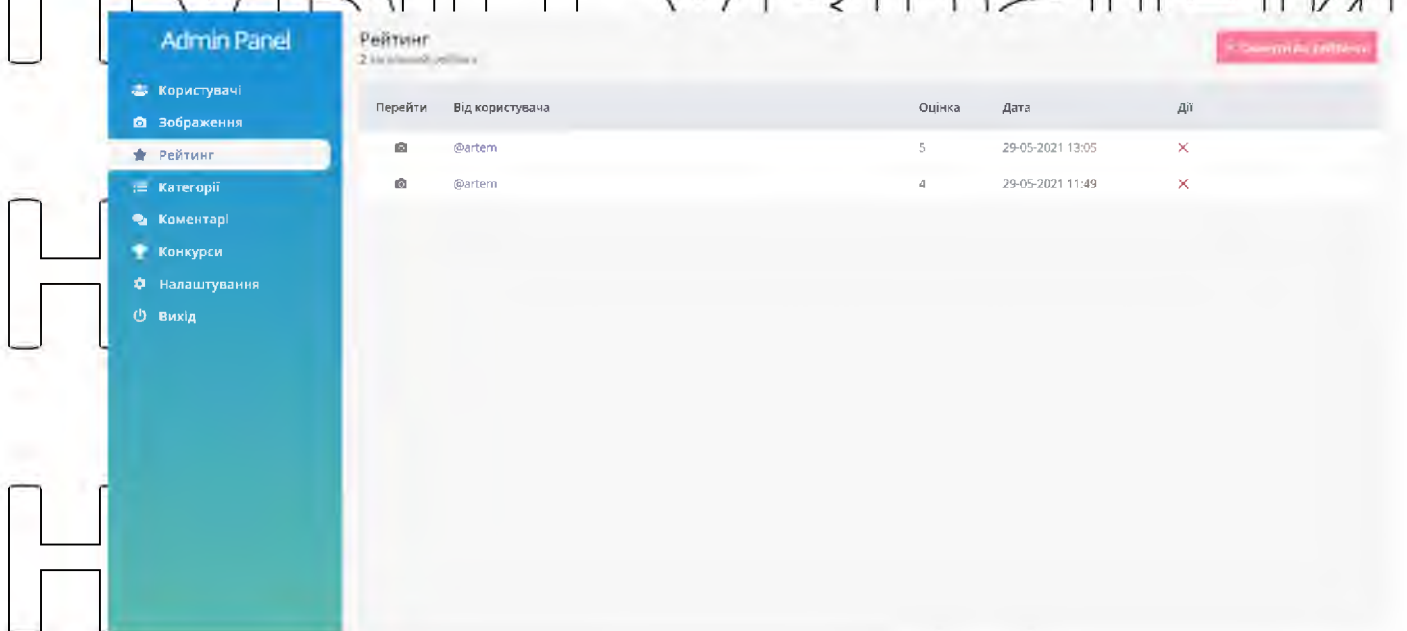


Рис. 35 Сторінка «Рейтинг» адміністративної частини системи

На наступній сторінці можливо створювати нові категорії зображень, редагувати вже існуючі або видалити. На рис. 36 та рис. 37 зображено всі існуючі категорії та форму створення категорії відповідно.



Рис. 36 Сторінка «Категорії»



Рис. 37 Форма створення категорії

Сторінка «Коментарі» структурована за таким же принципом, як і зображення. Існує дві вкладки: «Затверджено» та «В очікуванні». Коментар можна, як видали, так і опублікувати. Процес публікації коментаря на рис. 38.

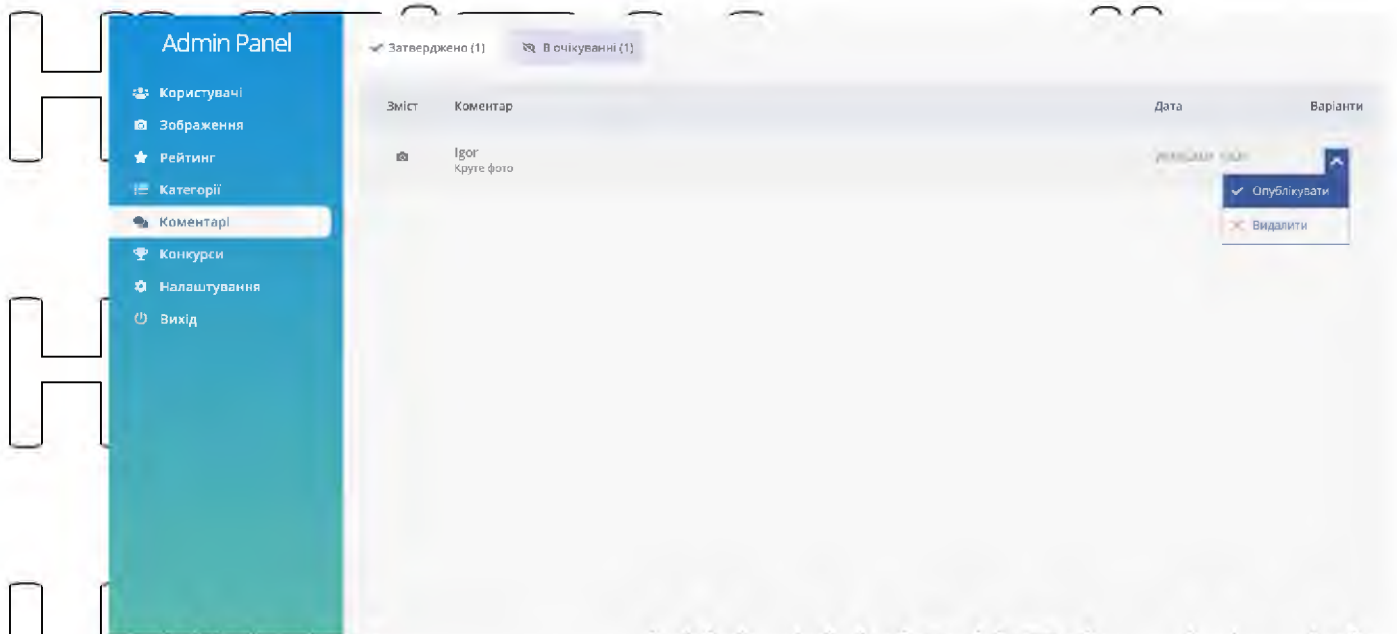


Рис. 38 Процес модерації коментаря

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

4.1 Нечітка експертна система

Експертна система - це програма для комп'ютера, яка оперує зі знаннями в певній предметній області з метою вироблення рекомендацій або вирішення проблем.

Експертна система може повністю взяти на себе функції, виконання яких звичайно вимагає залучення людини-фахівця, або відіграє роль асистента для людини, що ухвалює розв'язок. Інакше кажучи, система (технічна або соціальна), що вимагає ухвалення рішення, може одержати його безпосередньо від програми або через проміжну ланку - людини, яка спілкується із програмою. Той, хто ухвалює рішення, може бути експертом зі своїми власними правами, і в цьому випадку програма може "виправдати" своє існування, підвищуючи ефективність його роботи. Альтернативний варіант -

людей, що працює в співробітництві з такою програмою, може добитися з її допомогою результатів більш високої якості. Загалом кажучи, правильний розподіл функцій між людиною й машиною є однією із ключових умов високої ефективності впровадження експертних систем.

Технологія експертних систем є одним з напрямків нової області дослідження, яка одержала найменування штучного інтелекту (Artificial Intelligence - AI). Дослідження в цій області сконцентровані на розробці й впровадженні комп'ютерних програм, здатних емулювати (імітувати, відтворювати) ті області діяльності людини, які вимагають мислення, певної майстерності й накопиченого досвіду. До них належать завдання прийняття рішень, розпізнавання образів і розуміння людської мови.

Ця технологія вже успішно застосовується в деяких областях техніки й життя суспільства - органічної хімії, пошуку корисних копалин, медичній діагностиці. Перелік типових завдань, які вирішуються експертними системами, включає:

• добування інформації з первинних даних (таких як сигнали, що надходять від гідролокатора);

• діагностика несправностей (як у технічних системах, так і в людському організмі);

• структурний аналіз складних об'єктів (наприклад, хімічних сполук);

• вибір конфігурації складних багатокomпонентних систем (наприклад, розподілених комп'ютерних систем);

• планування послідовності виконання операцій, що приводять до заданої цілі (наприклад, виконувані промисловими роботами).

Система, крім виконання обчислювальних операцій, формує певні міркування й висновки, ґрунтуючись на тих знаннях, якими вона розташовує. Знання в системі представлені, як правило, на деякій спеціальній мові й зберігаються окремо від власне програмного коду, який і формує висновки й міркування. Цей компонент програми прийнято називати базою знань.

При формалізації знань існує проблема, що утрудняє використання традиційного математичного апарата. Це проблема опису понять, що оперують якісними характеристиками об'єктів (багато, мало, сильний, дуже сильний і т.п.).

Ці характеристики звичайно розмиті й не можуть бути однозначно інтерпретовані, однак містять важливу інформацію (наприклад, "одним з можливих ознак грипу є висока температура").

Крім того, у задачах, розв'язуваних інтелектуальними системами, часто доводиться користуватися неточними знаннями, які не можуть бути інтерпретовані як повністю щирі або помилкові (логічні true/false або 0/1). Існують знання, вірогідність яких виражається деякою проміжною цифрою, наприклад 0,7.

Як, не руйнуючи властивості розмитості й неточності, представляти подібні знання формально? Для дозволу таких проблем на початку 70-х років ХХ століття американський математик Лотфи Заде запропонував формальний апарат

нечіткої (fuzzy) алгебри й нечіткої логіки [Заде, 1972]. Пізніше цей напрямок одержав широкое поширення [Орловський, 1981; Аверкий й ін., 1986; Яшин, 1990] і поклало початок одній з галузей ІІІ за назвою м'які обчислення (soft computing).

Л. Заде ввів одне з головних понять у нечіткій логіці - поняття лінгвістичної змінної. Лінгвістична змінна (ЛП) — це змінна, значення якої визначається набором вербальних (тобто словесних) характеристик деякої властивості. Наприклад, ЛП "ріст" визначається через набір (карликовий, низький, середній, високий, дуже високий).

Значення лінгвістичної змінної (ЛП) визначаються через так називані нечіткі множини (НМ), які у свою чергу визначені на деякому базовому наборі значень або базовій числовій шкалі, що має розмірність. Кожне значення ЛП визначається як нечітка множина (наприклад, НМ "низький ріст").

Нечітка множина визначається через деяку базову шкалу B й функцію приналежності НМ — $\mu(x)$, $x \in B$ приймаючого значення на інтервалі $[0, 1]$. Таким чином, нечітка множина B — це сукупність пар виду $(x, \mu(x))$. Часто зустрічається й такий запис:

$$B = \sum_{i=1}^n \frac{x_i}{\mu(x_i)}$$

де x_i — i -те значення базової шкали.

Функція приналежності визначає суб'єктивний ступінь упевненості експерта в тім, що дане конкретне значення базової шкали відповідає обумовленому НМ. Цю функцію не варто плутати з імовірністю, що носить об'єктивний характер і підкоряється іншим математичним залежностям [8].

Для того щоб отримати певні висновки, в середовищі MATLAB створюємо нечітку експертну систему (рис. 39). Визначаємо входні та вихідні нечіткі змінні.

а також правила поведінки. Ціль експертної системи допомогти обрати найкращі фото роботи для подальшої оцінки організаторами. Для цього спочатку з роботами взаємодіють всі окові користувачі. Ця інформація і буде вхідними даними нечіткої експертної системи, а саме: середня оцінка зображення, кількість переглядів та кількість коментарів.

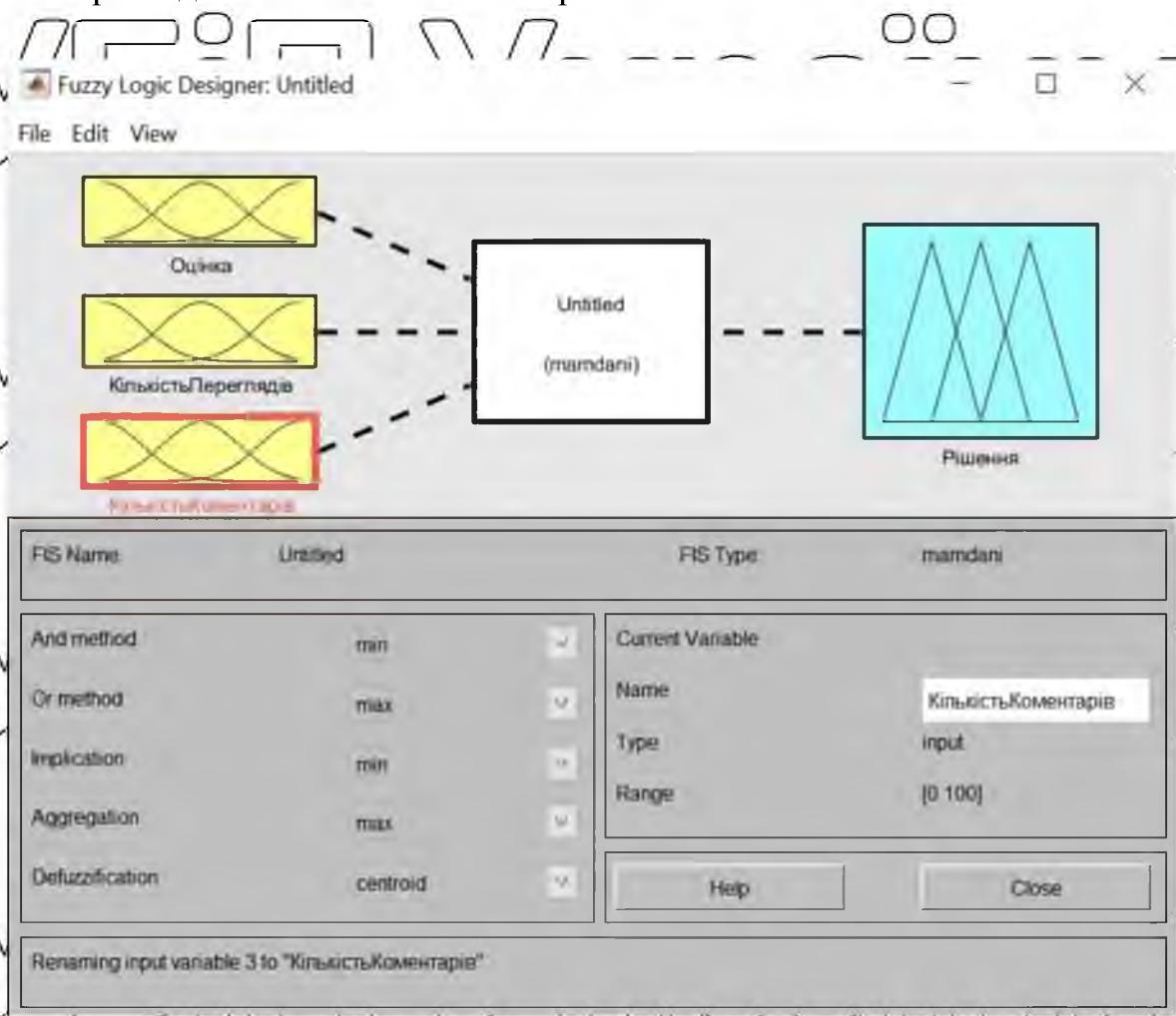


Рис. 39 Нечітка експертна система

При розгляді роботи, користувачі мають можливість оцінити зображення за п'ятибальною шкалою. Для першої вхідної інформації на рис. 40 задаємо діапазон від 1 до 5 балів та створюємо 3 рівні оцінок: низька (від 1 до 3 балів), середня (від 2.5 до 3.7 балів) та висока (від 3.4 до 5 балів).

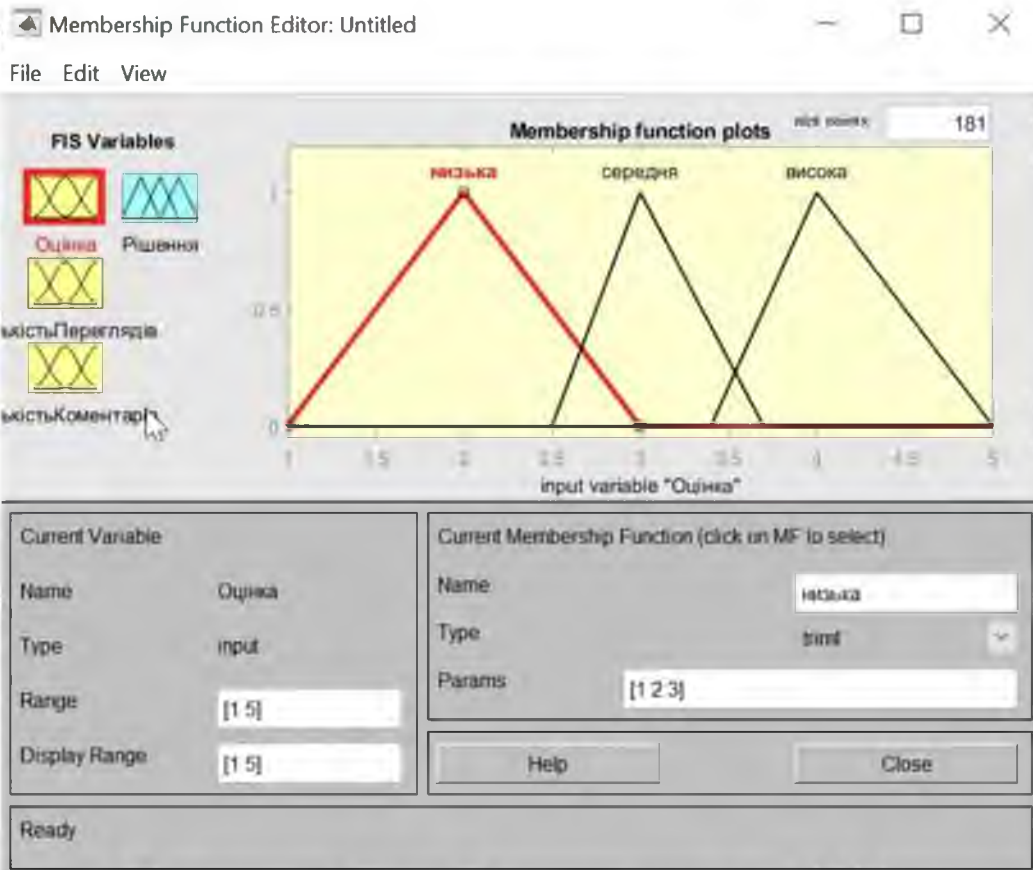


Рис. 40 Вхідна змінна «Середня оцінка зображення»

Користувачі не завжди будуть оцінювати зображення, навіть якщо перейдуть на сторінку з ним. Тому наступною важливою змінною вхідних даних буде кількість переглядів зображення (рис. 41). Задаємо умовний діапазон від 0 до 100 переглядів та розділяємо його на 3 рівні: мало переглядів (від 0 до 40), середня кількість переглядів (від 25 до 65) та багато переглядів (від 60 до 100).

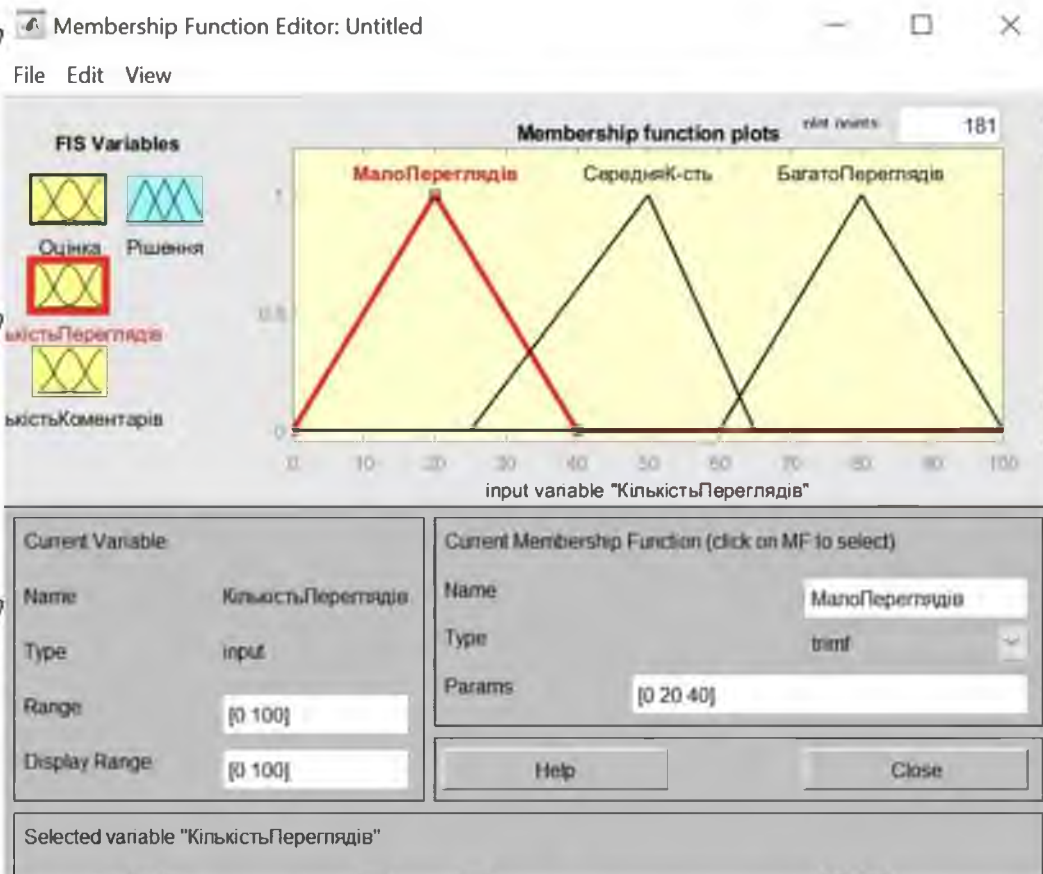


Рис. 41 Вхідна змінна «Кількість переглядів зображення»

Також не менш важливим фактором в виборі найкращих фото робіт є інтерес та активність до них збоку глядачів. Тому третьою вхідною змінною буде кількість коментарів зображення (рис. 42). Так як і в випадку з переглядами, створимо умовний діапазон від 0 до 100 коментарів та розділимо його на 3 рівні: мало коментарів (від 0 до 40), середня кількість коментарів (від 30 до 70) та багато коментарів (від 60 до 100).

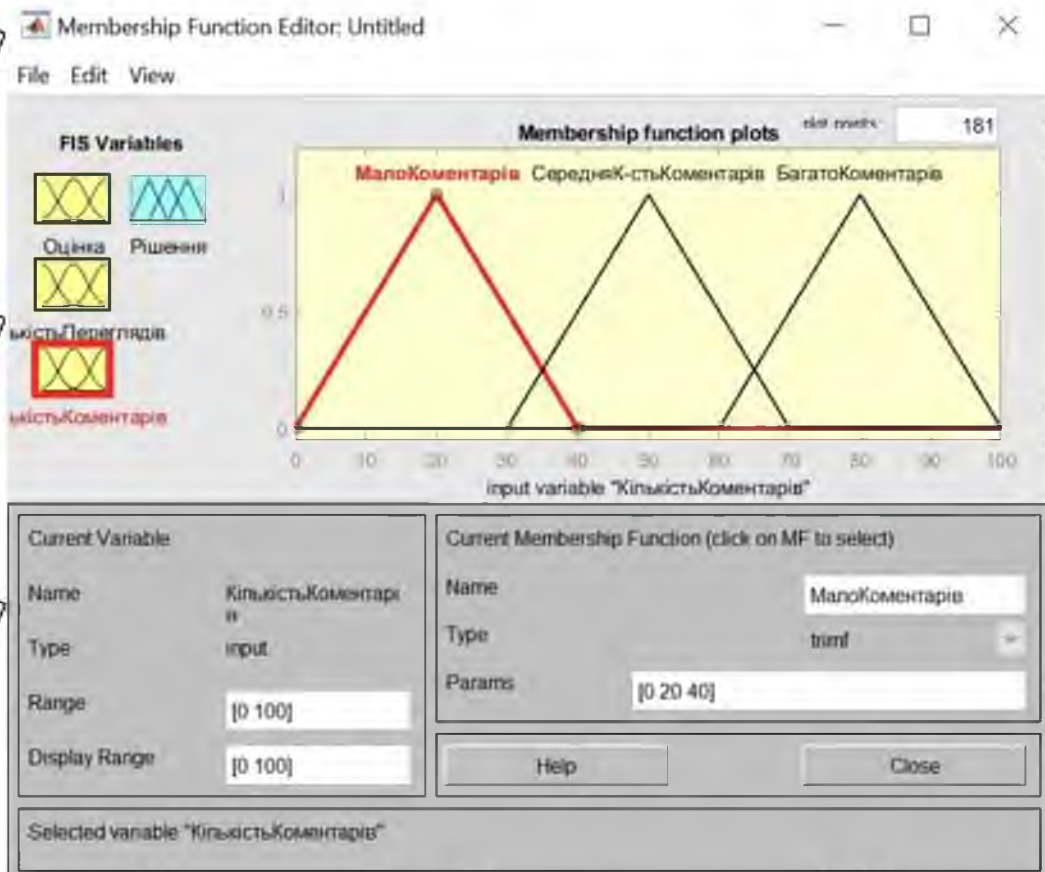


Рис. 42 Вхідна змінна «Кількість коментарів зображення»

Отже, експертна система вже має 5 вхідні змінні, але щоб перейти для створення правил поведінки необхідно створити вихідну інформацію, а саме – рішення. Створюємо умовний діапазон рішень від 0 до 1 (рис. 43). І якщо зображення на основі вхідних даних та створених правил поведінки отримає рішення в діапазоні від 0 до 0.4, то зображення буде відхилено для подальшої участі у конкурсі. Якщо ж діапазон буде від 0.3 до 0.7, тоді зображення буде передане на додатковий розгляд. А в випадку, коли рішення буде від 0.65 до 1, фото робота автоматично буде прийнята.

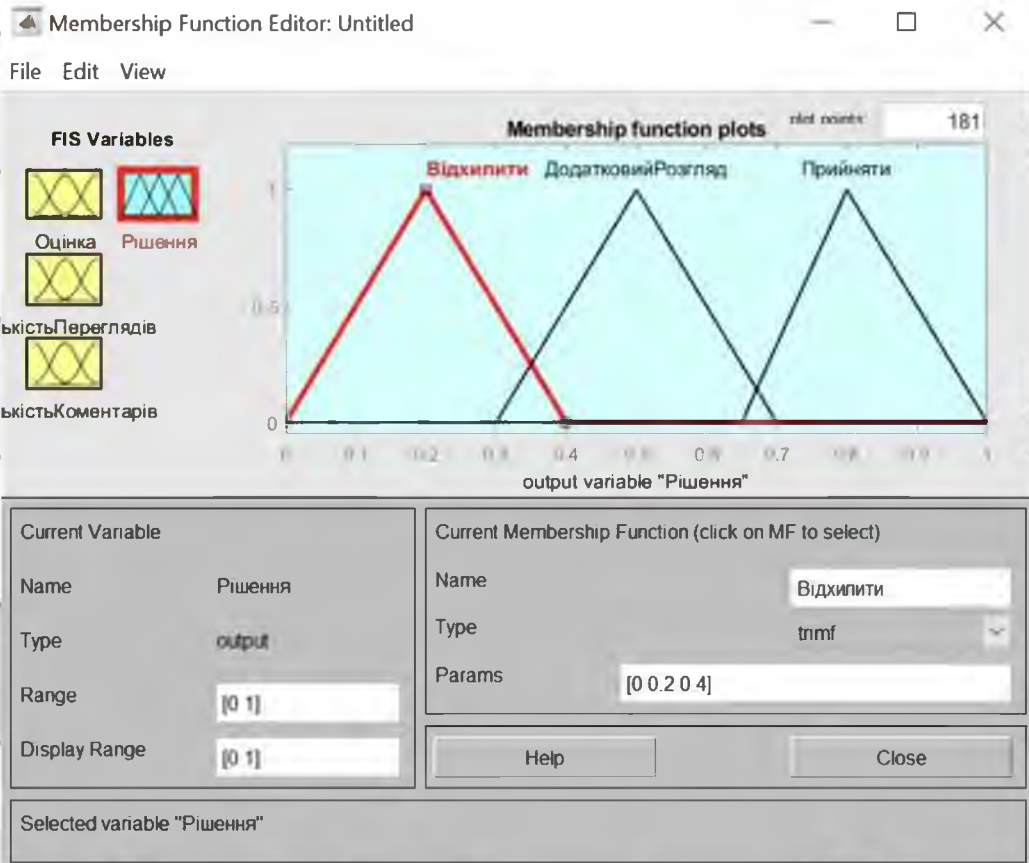


Рис. 43 Вихідна змінна «Рішення»

Тепер, коли система має і вхідну, і вихідну інформацію, залишається лише створити правила її поведінки. Наприклад, у випадку, коли оцінка зображення середня, кількість переглядів мала та кількість коментарів мала, тоді визначимо рішення – відхилити. За таким принципом перебираємо різні комбінації вхідних змінних та визначимо їх рішення. Правила поведінки експертної системи наведено на рис. 44.

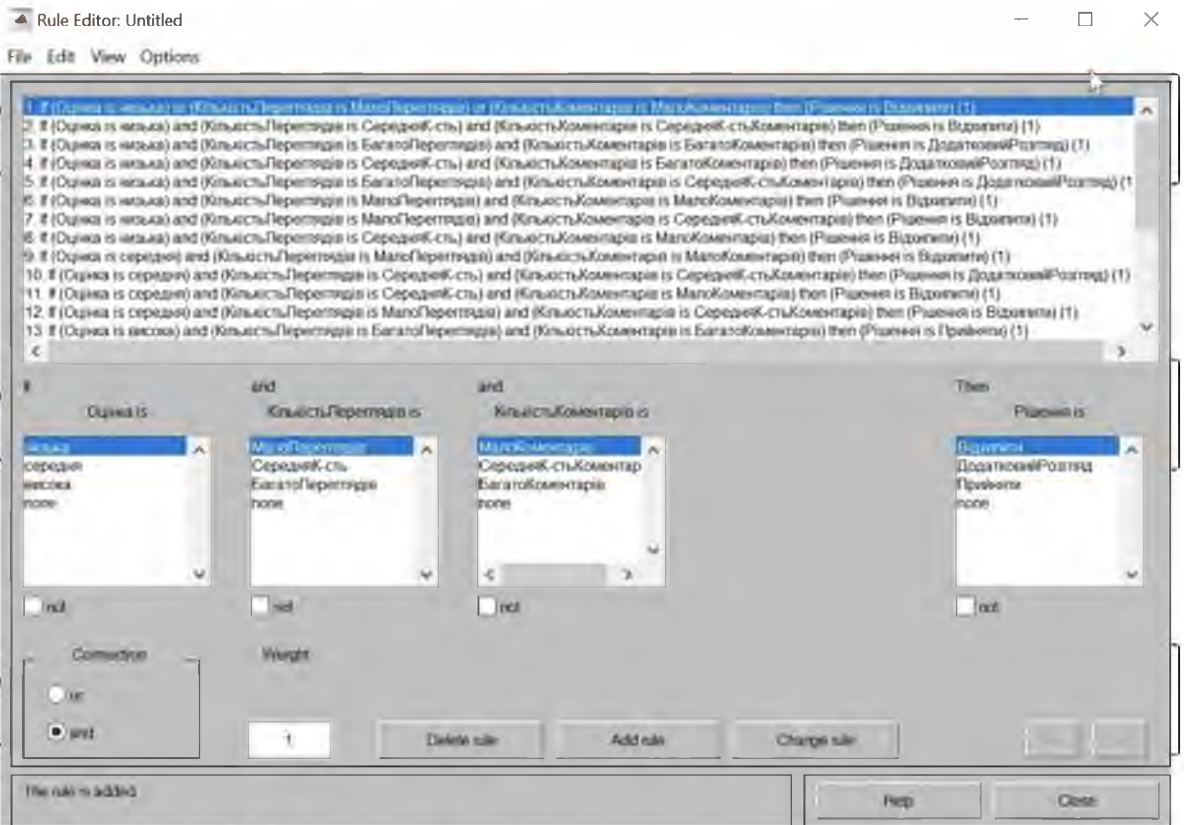


Рис. 44 Правила поведінки нечіткої експертної системи

На рис. 45-47 можна побачити тестування роботи системи. Наприклад, коли середня оцінка зображення 4,24, кількість переглядів – 77, а кількість коментарів – 84, тоді рішення складе 0.81 і спрацюють правила 13 та

21.



Рис. 45 Тестування системи при високих вхідних даних



Рис. 46 Тестування системи при середніх вхідних даних

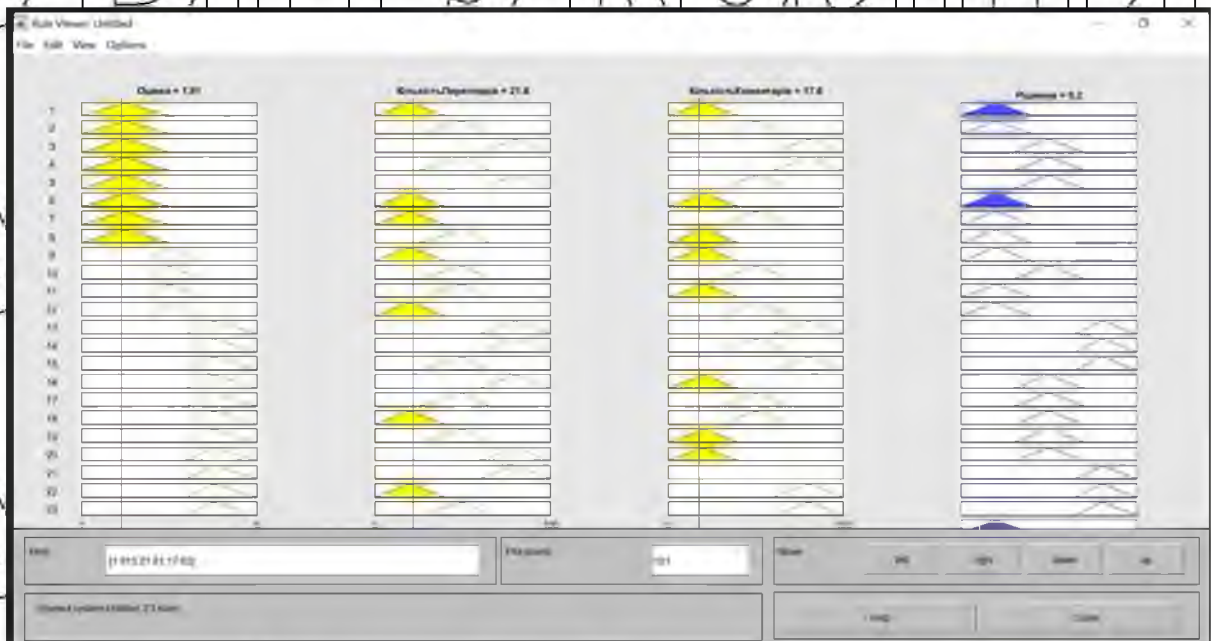


Рис. 47 Тестування системи при низьких вхідних даних

На рис. 48 зображена поверхня всіх створених для експертної системи правил поведінки.

НУБІП України

НУБІП України

НА
НА

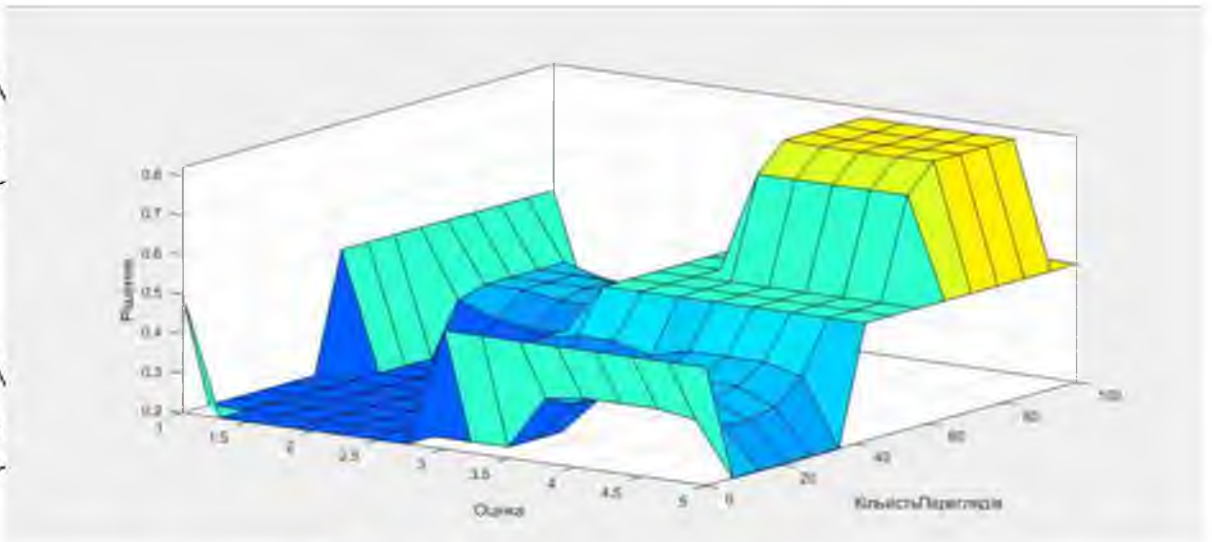


Рис. 48 Поверхня правил поведінки

4.2 Дослідження використання задач класифікації

4.2.1 Використання 1-Rule для класифікації.

Цей метод використовується в алгоритмі послідовного навчання для вивчення правил. Він повертає одне правило, яке охоплює принаймні кілька прикладів. Однак те, що робить його дійсно потужним, так це його здатність створювати зв'язки між наданими атрибутами, отже охоплюючи більший простір гіпотез.

Наприклад

ЯКЩО мати (y, x) і жінка (y), ТО дочка (x, y).

Тут будь-яку людину можна асоціювати зі змінними x і y.

Алгоритм Learn-One-Rule дотримується парадигми жадібного пошуку, коли він шукає правила з високою точністю, але його покриття дуже низьке. Він класифікує всі позитивні приклади для конкретного випадку. Він повертає одне правило, яке охоплює кілька прикладів [9].

Реалізація алгоритму 1-Rule

Для реалізації алгоритму 1-Rule було виділено 2 класи високої ("HIGH") та низької ("LOW") оцінки робіт, показник яких визначається у порівнянні з середньою оцінкою за весь період дослідження.

За допомогою SQL визначені результати алгоритму. Нижче наведено реалізацію та результат у вигляді таблиці.

Спочатку проведено вибірку даних, де визначено середню оцінку та тип оцінки за кожним видом освіти автора та користувача, який оцінював (рис. 49).

```

use Contest_DB
go
DECLARE @avgMark FLOAT;
SELECT @avgMark = AVG(Contest_Fact.Avg_mark)
FROM Contest_Fact;

SELECT Education_Dim.Name_education,
       Category_user_Dim.Name_category,
       Contest_Fact.Avg_mark AS mark,
       @avgMark AS avgHarvest,
       CASE
         WHEN @avgMark > Contest_Fact.Avg_mark
         THEN 'LOW'
         ELSE 'HIGH'
       END AS markClassification
FROM Contest_Fact
JOIN Education_Dim ON Contest_Fact.Id_education = Education_Dim.Id_education
JOIN Category_user_Dim ON Contest_Fact.Id_category_user =
Category_user_Dim.Id_category_user
GROUP BY Education_Dim.Name_education, Category_user_Dim.Name_category,
Contest_Fact.Avg_mark

```

	Name_education	Name_category	mark	avgHarvest	markClassification
1	Вища гуманітарна	Відвідувач	7	7.79310344827586	LOW
2	Вища гуманітарна	Відвідувач	8	7.79310344827586	HIGH
3	Вища гуманітарна	Відвідувач	9	7.79310344827586	HIGH
4	Вища гуманітарна	Відвідувач	10	7.79310344827586	HIGH
5	Вища гуманітарна	Жюрі	8	7.79310344827586	HIGH
6	Вища гуманітарна	Жюрі	9	7.79310344827586	HIGH
7	Вища гуманітарна	Користувач	6	7.79310344827586	LOW
8	Вища гуманітарна	Користувач	7	7.79310344827586	LOW
9	Вища гуманітарна	Користувач	8	7.79310344827586	HIGH
10	Вища професійна	Відвідувач	7	7.79310344827586	LOW
11	Вища професійна	Відвідувач	8	7.79310344827586	HIGH
12	Вища професійна	Відвідувач	9	7.79310344827586	HIGH
13	Вища професійна	Відвідувач	10	7.79310344827586	HIGH
14	Вища професійна	Жюрі	8	7.79310344827586	HIGH
15	Вища професійна	Користувач	7	7.79310344827586	LOW
16	Вища професійна	Користувач	8	7.79310344827586	HIGH
17	Вища технічна	Відвідувач	7	7.79310344827586	LOW
18	Вища технічна	Відвідувач	8	7.79310344827586	HIGH
19	Вища технічна	Відвідувач	9	7.79310344827586	HIGH
20	Вища технічна	Відвідувач	10	7.79310344827586	HIGH
21	Вища технічна	Жюрі	8	7.79310344827586	HIGH
22	Вища технічна	Жюрі	7	7.79310344827586	HIGH

Рис. 49 Вибірка фактів

Наступним етапом було визначення класифікації оцінки та її ймовірність відносно середнього показника серед заданих фактів відповідно до типу освіти (рис. 50) та типу користувача, який оцінював (рис. 51).

```

DECLARE @avgMark FLOAT;
SELECT @avgMark = AVG(Contest_Fact.Avg_mark)
FROM Contest_Fact;

```



```

SELECT Education Dim.Name education,
      (SELECT COUNT(cf.Avg_mark)
      FROM Contest_Fact cf
      WHERE @avgMark > (cf.Avg_mark)
      AND Contest_Fact.Id_education = cf.Id_education) AS
lessThanAvg,
      (SELECT COUNT(cf.Avg_mark)
      FROM Contest_Fact cf
      WHERE @avgMark < (cf.Avg_mark)
      AND Contest_Fact.Id_education = cf.Id_education) AS
biggerThanAvg,
      COUNT(Contest_Fact.Avg_mark) AS
allMarks,
      AVG(Contest_Fact.Avg_mark) AS
avgEducMark,
      @avgMark AS
avgMark,
CASE
  WHEN @avgMark > AVG(Contest_Fact.Avg_mark)
  THEN 'LOW'
  ELSE 'HIGH'
END
markClassification,
      ROUND(CAST(CAST((SELECT COUNT(cf.Avg_mark)
      FROM Contest_Fact cf
      WHERE @avgMark > (cf.Avg_mark)
      AND Contest_Fact.Id_education = cf.Id_education) AS FLOAT) /
      CAST(COUNT(Contest_Fact.Avg_mark) AS FLOAT) AS FLOAT) * 100, 2) AS
lessProbability,
      ROUND(CAST(CAST((SELECT COUNT(cf.Avg_mark)
      FROM Contest_Fact cf
      WHERE @avgMark < (cf.Avg_mark)
      AND Contest_Fact.Id_education = cf.Id_education) AS FLOAT) /
      CAST(COUNT(Contest_Fact.Avg_mark) AS FLOAT) AS FLOAT) * 100, 2) AS
biggerProbability
FROM Contest_Fact
JOIN Education_Dim ON Contest_Fact.Id_education = Education_Dim.Id_education
GROUP BY Education_Dim.Name_education, Contest_Fact.Id_education

```

	Name_education	lessThanAvg	biggerThanAvg	allMarks	avgEducMark	avgMark	markClassification	lessProbability	biggerProbability
1	Вища професійна	12	21	33	7.75757575757576	7.79310344827586	LOW	36.36	63.64
2	Вища технічна	8	17	25	7.8	7.79310344827586	HIGH	32	68
3	Вища гуманітарна	11	22	33	7.78787878787879	7.79310344827586	LOW	33.33	66.67
4	Повна середня	8	17	25	7.84	7.79310344827586	HIGH	32	68

Рис. 50 Класифікація відповідно до освіти

```

DECLARE @avgMark FLOAT;
SELECT @avgMark = AVG(Contest_Fact.Avg_mark)
FROM Contest_Fact;

SELECT Category_user Dim.Name_category,
      (SELECT COUNT(cf.Avg_mark)
      FROM Contest_Fact cf
      WHERE @avgMark > (cf.Avg_mark)
      AND Contest_Fact.Id_category_user = cf.Id_category_user) AS
lessThanAvg,
      (SELECT COUNT(cf.Avg_mark)
      FROM Contest_Fact cf
      WHERE @avgMark < (cf.Avg_mark)
      AND Contest_Fact.Id_category_user = cf.Id_category_user) AS
biggerThanAvg,
      COUNT(Contest_Fact.Avg_mark) AS
allMarks,

```

```

AVG(Contest_Fact.Avg_mark) AS
avgEducMark,
@avgMark
avgMark,
CASE
WHEN @avgMark > AVG(Contest_Fact.Avg_mark)
THEN 'LOW'
ELSE 'HIGH'
END AS
markClassification,
ROUND(CAST(CAST((SELECT COUNT(cf.Avg_mark)
FROM Contest_Fact cf
WHERE @avgMark > cf.Avg_mark
AND Contest_Fact.Id_category_user = cf.Id_category_user) AS FLOAT)
/ CAST(COUNT(Contest_Fact.Avg_mark) AS FLOAT) AS FLOAT) * 100, 2) AS
lessProbability,
ROUND(CAST(CAST((SELECT COUNT(cf.Avg_mark)
FROM Contest_Fact cf
WHERE @avgMark < (cf.Avg_mark)
AND Contest_Fact.Id_category_user = cf.Id_category_user) AS FLOAT)
/ CAST(COUNT(Contest_Fact.Avg_mark) AS FLOAT) AS FLOAT) * 100, 2) AS
biggerProbability
FROM Contest_Fact
JOIN Category_user_Dim ON Contest_Fact.Id_category_user =
Category_user_Dim.Id_category_user
GROUP BY Category_user_Dim.Name_category, Contest_Fact.Id_category_user

```

	Name_category	lessThanAvg	biggerThanAvg	allMarks	avgEducMark	avgMark	markClassification	lessProbability	biggerProbability
1	Відвідувач	9	28	37	8.18918918918919	7.79310344827586	HIGH	24,32	75,68
2	Жюрі	1	40	41	8.09756097560976	7.79310344827586	HIGH	2,44	97,56
3	Користувач	29	9	38	7.07894736842105	7.79310344827586	LOW	76,32	23,68

Рис. 51 Класифікація відповідно до категорії оцінюванів

Аналізуючи отримані дані можна виділити наступні правила:

Якщо тип користувача «Відвідувач», то тип оцінки високий ("High") з імовірністю 75,68%.

– Якщо тип користувача «Жюрі», то тип оцінки високий ("High") з імовірністю 97,56%.

Якщо тип користувача «Користувач», то тип оцінки низький ("Low") з імовірністю 23,68%.

– Якщо напрямок освіти «Вища професійна», то тип оцінки низький ("Low") з імовірністю 63,64%.

– Якщо напрямок освіти «Вища технічна», то тип оцінки високий ("High") з імовірністю 68%.

– Якщо напрямок освіти «Вища гуманітарна», то тип оцінки низький ("Low") з імовірністю 63,64%.

Якщо напрямок освіти «Повна середня», то тип оцінки високий ("High") з імовірністю 68%.

Можна зробити висновок, що на рівень оцінки робіт більшою мірою впливає користувач, який оцінює, а не освіта, яку має автор роботи хоча як можна помітити все ж вона впливає на рівень оцінки, яка отримана.

4.2.2 Використання методу наївного Байєса

Наївні методи Байєса — це набір контрольованих алгоритмів навчання, заснованих на застосуванні теореми Байєса з «наївним» припущенням про умовну незалежність між кожною парою ознак, заданою значенням змінної класу. Теорема Байєса встановлює наступний зв'язок із заданою змінною класу та залежним вектором ознак через x_1, x_2 :

$$P(y | x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots, x_n | y)}{P(x_1, \dots, x_n)}$$

Використовуючи наївне припущення про умовну незалежність

$$P(x_i | y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i | y),$$

для всіх i це співвідношення спрощено до

$$P(y | x_1, \dots, x_n) = \frac{P(y) \prod_{i=1}^n P(x_i | y)}{P(x_1, \dots, x_n)} \quad [10]$$

Типи простого класифікатора Байєса

Багаточлен наївного Байєса:

Це здебільшого використовується для проблеми класифікації документів, тобто, чи належить документ до категорії спорту, політики, технологій тощо.

Функції прогнози, які використовує класифікатор, — це частота слів, присутніх у документі.

Наївний Байєс-Бернуллі:

Це схоже на мультиноміальний наївний метод Бесса, але предикторами є булеві змінні. Параметри, які ми використовуємо для прогнозування змінної класу, приймають лише значення «так» або «ні», наприклад, зустрічається слово в тексті чи ні.

Гауссовий наївний Байес:

Коли предиктори приймають безперервні значення і не є дискретними, ми припускаємо, що ці значення взято з розподілу Гауса [11].

Реалізація алгоритму

На основі розгорнутого кубу створюємо нову структуру інтелектуального аналізу даних. Обираємо алгоритм Байеса (рис. 52-53).

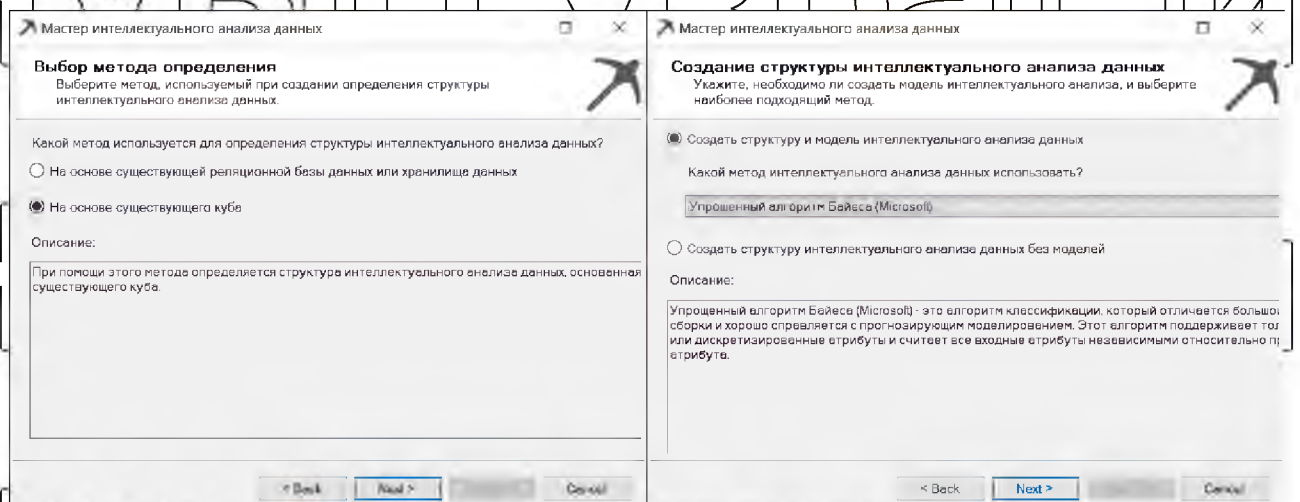


Рисунок. 52-53 Створення структури аналізу на основі алгоритму Байєса

Обираємо вимір для аналізу (рис. 54-55).

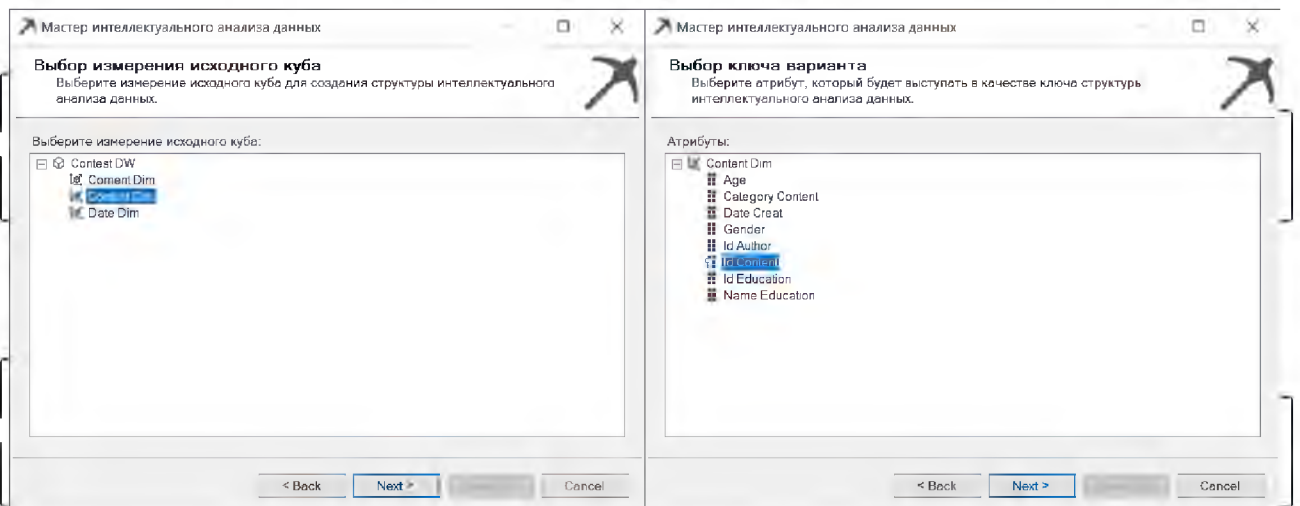


Рис. 54-55 Вибір виміру для аналізу

Обираємо стовбці для аналізу та прогнозування (рис. 56-57).

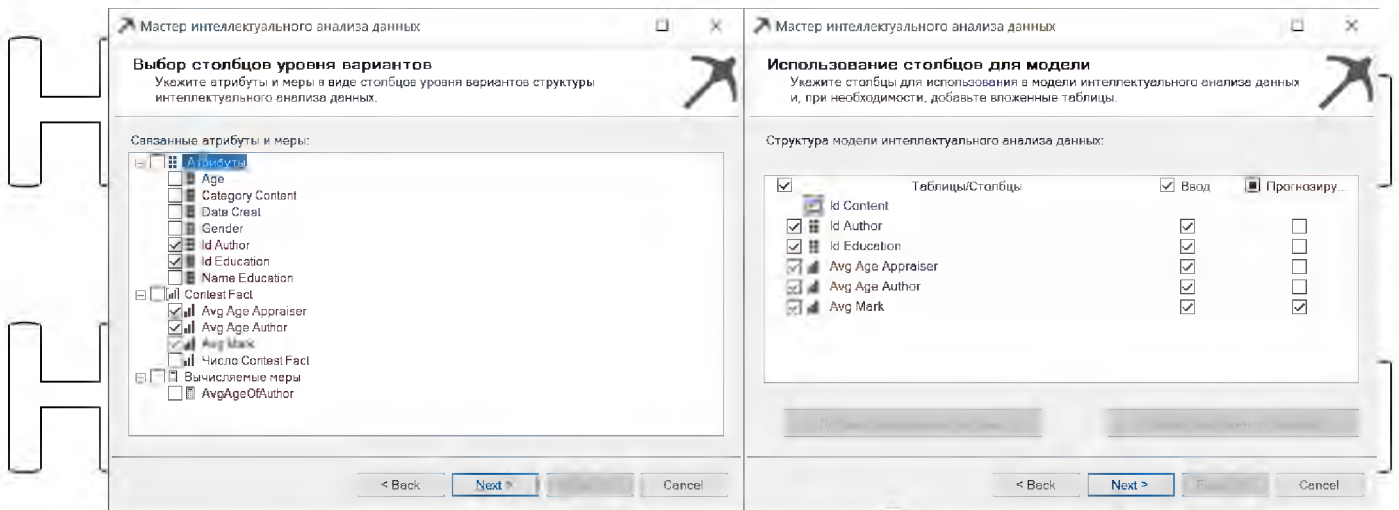


Рис. 56-57 Вибір стовпців для аналізу та прогнозування

На основі створеної структури для аналізу буде сформовано мережу залежностей (рис. 58).

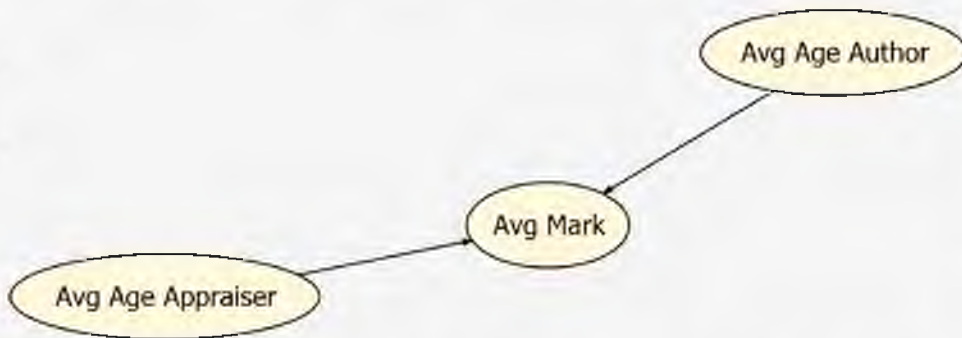


Рис. 58 Розгорнута мережа залежностей

На основі побудованої мережі можна зробити висновок, що на середню оцінку безпосередньо впливають вік автора та людей, які оцінюють роботи.

Також можна розглянути більш детальний опис за кожним прогнозованим значенням. Нижче на рис. 59 наведено детальний опис профілю атрибуту «Avg Mark».

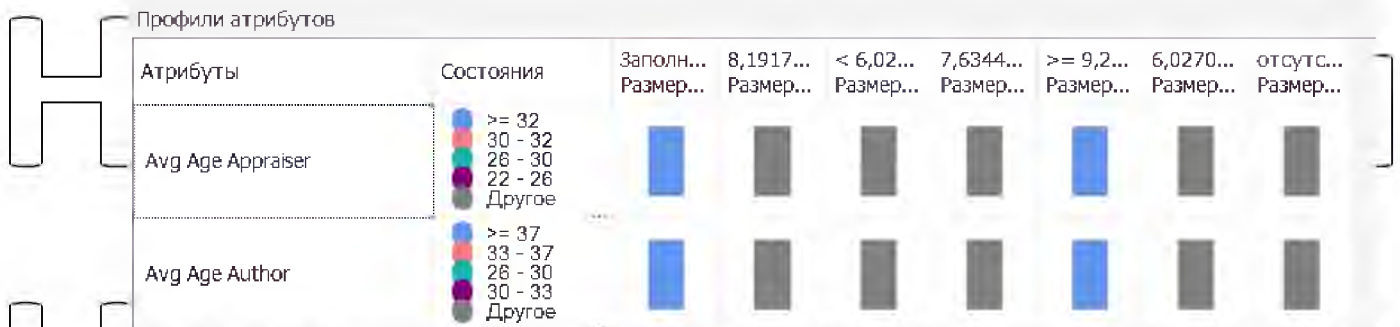


Рис. 59 Профіль атрибуту «Avg Mark»

4.3 Дослідження використання методу асоціативних правил

Інтелектуальний аналіз правил асоціації знаходить цікаві асоціації та зв'язки між великими наборами елементів даних. Це правило показує, як часто набір елементів зустрічається в транзакції [12].

Типи вивчення правил асоціації

Вивчення правила асоціації можна розділити на три алгоритми:

Апріорний алгоритм: використовує часті набори даних для створення правил асоціації. Він призначений для роботи з базами даних, які містять транзакції. Цей алгоритм використовує пошук у ширину та хеш-дерево для ефективного обчислення набору елементів.

Він в основному використовується для аналізу ринкового кошика та допомагає зрозуміти продукти, які можна купити разом. Його також можна використовувати в галузі охорони здоров'я для виявлення реакції пацієнтів на ліки.

Алгоритм Eclat: розшифровується як перетворення класу еквівалентності. Цей алгоритм використовує метод пошуку в глибину для пошуку частих наборів елементів у базі даних транзакцій. Він виконує швидше, ніж алгоритм Apriori.

Алгоритм зростання F-P: розшифровується як Frequent Pattern і є вдосконаленою версією алгоритму Apriori. Він представляє базу даних у формі деревовидної структури, яка відома як шаблон частоти або дерево. Метою цього дерева частот є виділення найбільш частих шаблонів.

Застосування вивчення правил асоціації

Він має різні застосування в машинному навчанні та аналізі даних. Нижче наведено деякі популярні програми вивчення правил асоціації.

- **Аналіз ринкового кошика:** це один із популярних прикладів і застосувань аналізу правил асоціації. Ця методика зазвичай використовується великими роздрібними торговцями для визначення зв'язку між товарами.
- **Медична діагностика:** за допомогою правил асоціації пацієнтів можна легко вилікувати, оскільки це допомагає визначити ймовірність хвороби певної хвороби.
- **Білкова послідовність:** правила асоціації допомагають визначити синтез штучних білків.
- Він також використовується для розробки каталогу та аналізу підерів збитків і багатьох інших програм [13].

Реалізація алгоритму

На основі розгорнутого кубу створюємо нову структуру інтелектуального аналізу даних, використовуючи алгоритм пошуку взаємозв'язнів (рис. 60-61).

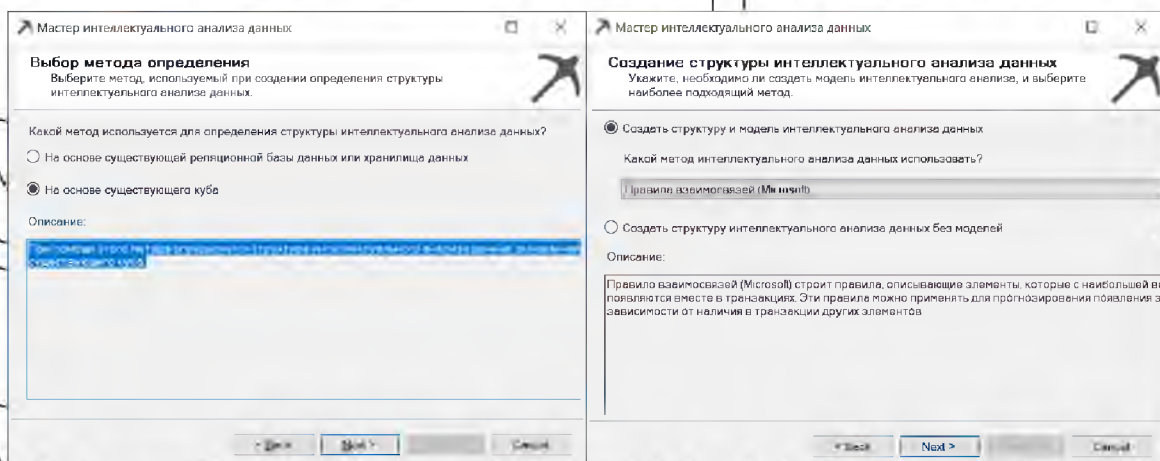


Рис. 60-61 Вибір методу інтелектуального аналізу

Обираємо вимір та ключ на основі якого буде проводитись аналіз (рис. 62-

63).

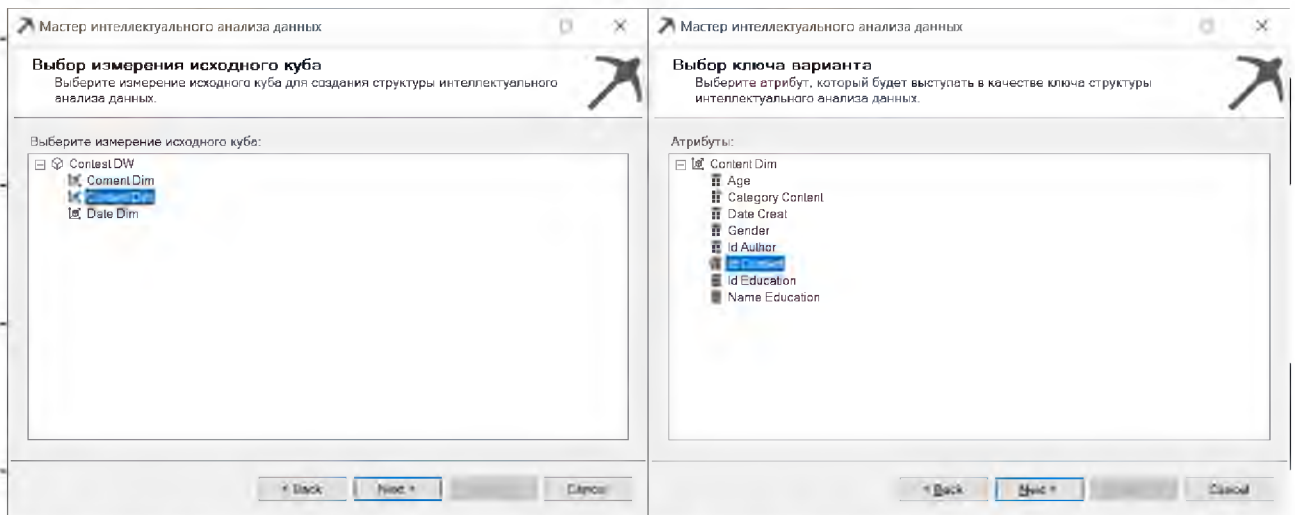


Рис. 62-63 Вибір виміру та ключа, на основі якого проводиться аналіз

А також, обираємо поля для введення та прогнозування (рис. 64-65).

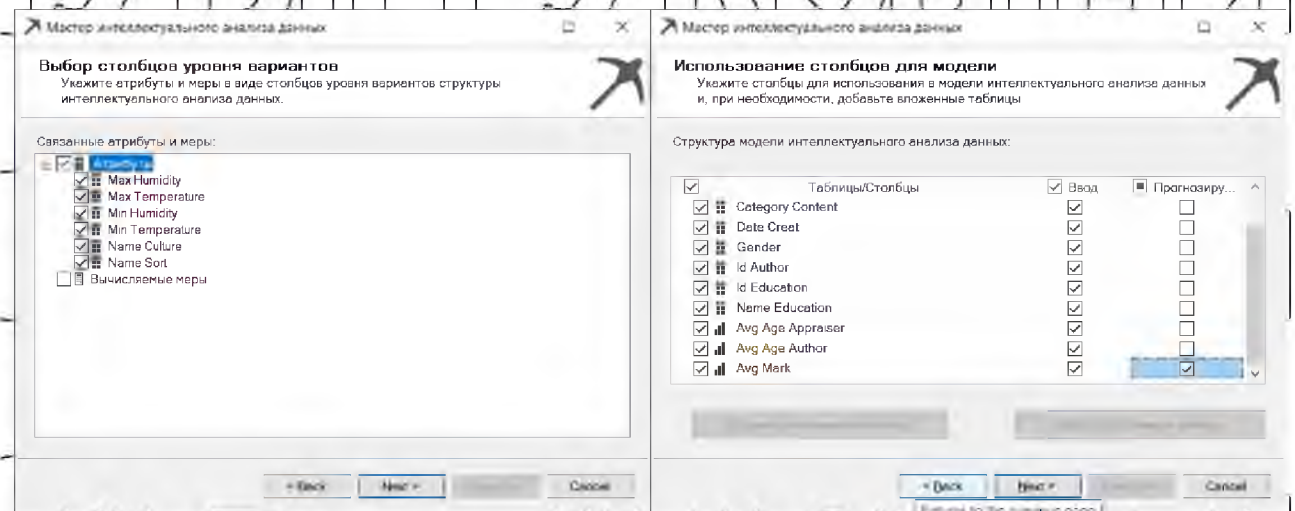


Рис. 64-65 Вибір полів для введення та прогнозування

На основі розгорнутої структури було побудовано мережу залежностей середньої оцінки від заданих параметрів (рис. 66).

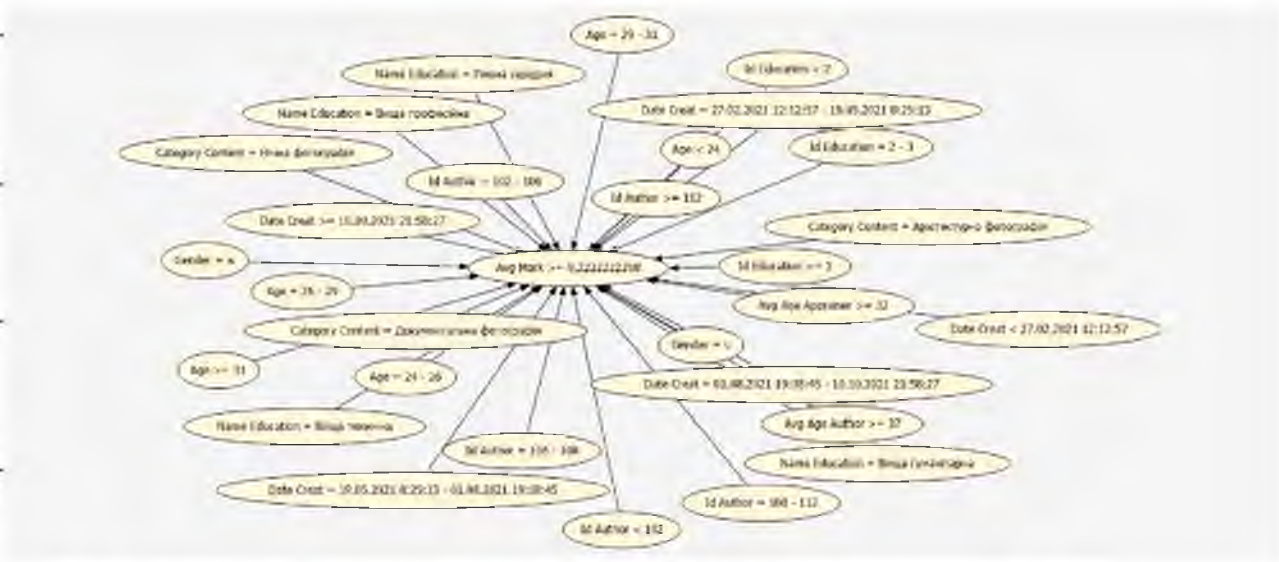


Рис. 66 Мережа залежностей

Також були побудовані правила з вказаною імовірністю та важливістю. Деякі з них наведені на таблиці 1.

Таблиця 1

Сформовані правила на основі знайдених взаємозалежностей

Вероятність	Важливість	Правило
1,000	0,296	Avg Age Appraiser >= 37 -> Avg Mark >= 9,2222222208
1,000	0,296	Avg Age Appraiser >= 30 -> Avg Mark >= 9,2222222208
1,000	0,296	Avg Age Author >= 37, Avg Age Appraiser >= 32 -> Avg Mark >= 9,2222222208
1,000	-0,171	Age < 24, Date Creat >= 29.19.2021 21:58:27 -> Avg Mark >= 9,2222222208
1,000	-0,171	Age < 24, Category Content = Агротехнічна фотографія -> Avg Mark >= 9,2222222208
1,000	-0,171	Age < 24, Date Creat = 27.02.2021 12:12:57 - 19.05.2021 8:25:13 -> Avg Mark >= 9,2222222208
1,000	-0,171	Age < 24, Date Creat = 01.08.2021 19:08:45 - 10.08.2021 21:58:27 -> Avg Mark >= 9,2222222208
1,000	-0,171	Age < 24, Category Content = Ніва фотографія -> Avg Mark >= 9,2222222208
1,000	-0,171	Name Education = Півня серпень, Date Creat = 19.05.2021 8:25:13 - 01.08.2021 19:08:45 -> Avg Mark >= 9,2222222208
1,000	-0,171	Name Education = Півня серпень, Date Creat >= 10.10.2021 21:58:27 -> Avg Mark >= 9,2222222208
1,000	-0,171	Age = 26 - 29, Date Creat = 19.05.2021 8:25:13 - 01.08.2021 19:08:45 -> Avg Mark >= 9,2222222208
1,000	-0,171	Age = 26 - 29, Date Creat >= 10.08.2021 21:58:27 -> Avg Mark >= 9,2222222208
1,000	-0,171	M Author = 106 - 108, Date Creat = 19.05.2021 8:25:13 - 01.08.2021 19:08:45 -> Avg Mark >= 9,2222222208
1,000	-0,171	Date Creat >= 10.10.2021 21:58:27, Age = 29 - 31 -> Avg Mark >= 9,2222222208
1,000	-0,171	Date Creat >= 10.10.2021 21:58:27, Category Content = Агротехнічна фотографія -> Avg Mark >= 9,2222222208
1,000	0,171	Date Creat >= 10.10.2021 21:58:27, Gender = ж -> Avg Mark >= 9,2222222208
1,000	0,171	Date Creat < 27.02.2021 12:12:57, M Author = 100 - 102 -> Avg Mark >= 9,2222222208
1,000	-0,120	Age < 24, Date Creat = 19.05.2021 8:25:13 - 01.08.2021 19:08:45 -> Avg Mark >= 9,2222222208
1,000	-0,120	Age < 24, M Author >= 112 -> Avg Mark >= 9,2222222208
1,000	-0,120	Age < 24, Gender = ж -> Avg Mark >= 9,2222222208
1,000	-0,120	Age < 24, Name Education = Ніва фотографія -> Avg Mark >= 9,2222222208
1,000	-0,120	Age < 24, M Education >= 3 -> Avg Mark >= 9,2222222208
1,000	-0,120	Name Education = Півня серпень, Category Content = Агротехнічна фотографія -> Avg Mark >= 9,2222222208
1,000	-0,120	Age = 28 - 29, Category Content = Агротехнічна фотографія -> Avg Mark >= 9,2222222208
1,000	-0,120	M Author = 106 - 108, Date Creat = 27.02.2021 12:12:57 - 19.05.2021 8:25:13 -> Avg Mark >= 9,2222222208
1,000	-0,120	M Author < 102, Date Creat >= 19.10.2021 21:58:27 -> Avg Mark >= 9,2222222208
1,000	-0,120	M Author < 102, Date Creat = 27.02.2021 12:12:57 - 19.05.2021 8:25:13 -> Avg Mark >= 9,2222222208

4.4 Дослідження використання алгоритмів кластеризації

Кластеризація — це неконтрольований алгоритм на основі машинного навчання, який об'єднує групу точок даних у кластери, щоб об'єкти належали до однієї групи.

Кластеризація допомагає розбити дані на кілька підмножин. Кожна з цих підмножин містить дані, схожі одна на одну, і ці підмножини називаються кластерами. Телер, коли дані нашої клієнтської бази розділені на кластери, ми можемо прийняти обґрунтоване рішення про те,

Що таке кластер?

- Кластер — це підмножина подібних об'єктів
- Підмножина об'єктів, така що відстань між будь-яким із двох об'єктів у кластері менша за відстань між будь-яким об'єктом у кластері та будь-яким об'єктом, який не знаходиться всередині нього.
- Зв'язана область багатовимірного простору з порівняно високою щільністю об'єктів. Це, на нашу думку, найкраще підходить для цього продукту.

Що таке кластеризація в інтелектуальному аналізі даних?

- Кластеризація — це метод перетворення групи абстрактних об'єктів у класи подібних об'єктів.
- Кластеризація — це метод поділу набору даних або об'єктів на набір значущих підкласів, які називаються кластерами.
- Він допомагає користувачам зрозуміти структуру або природне групування в наборі даних і використовується або як окремий інструмент для кращого розуміння розподілу даних, або як етап попередньої обробки для інших алгоритмів [14]

Реалізація алгоритму

На основі розгорнутого кубу створюємо нову структуру інтелектуального аналізу даних, використовуючи алгоритм кластерного аналізу. Обираємо вимір, на основі якого буде проводитись аналіз (рис. 67-68)

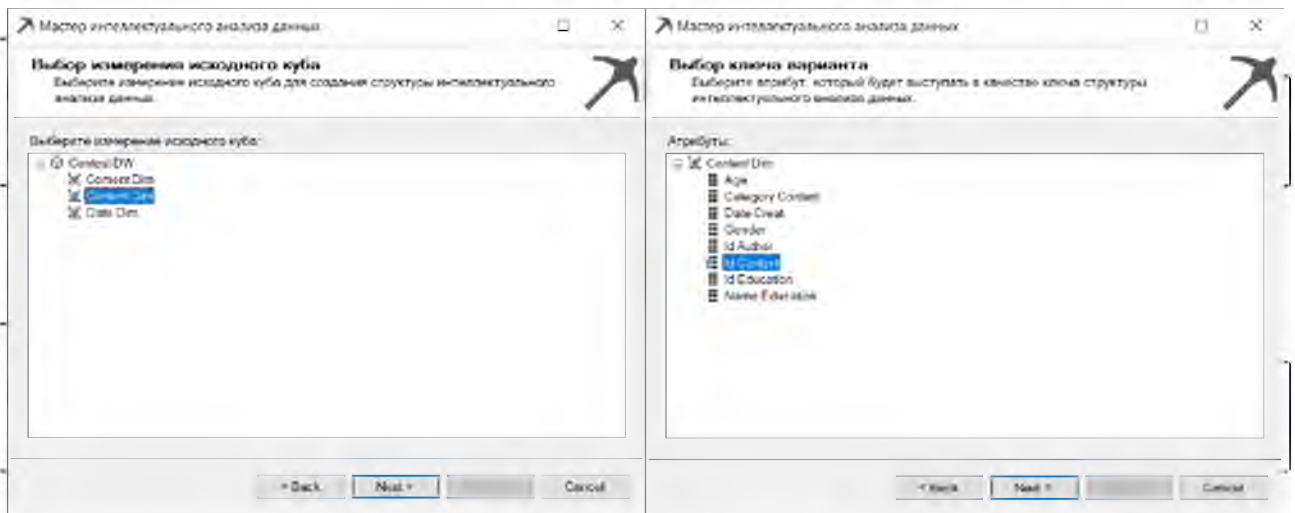


Рис. 67-68 Вибір виміру

Наступним етапом визначаємо поля та фіксуємо типи їх значень (рис. 69-70).

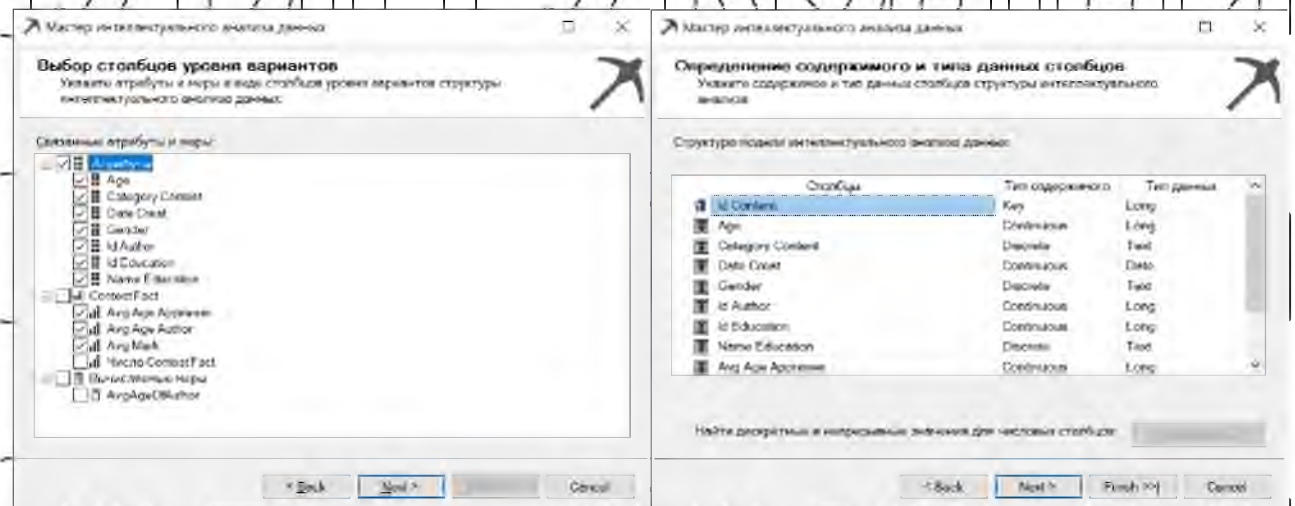


Рис. 69-70 Визначення полів

На основі розгорнутої структури було отримано модель, наведену на рис. 71.

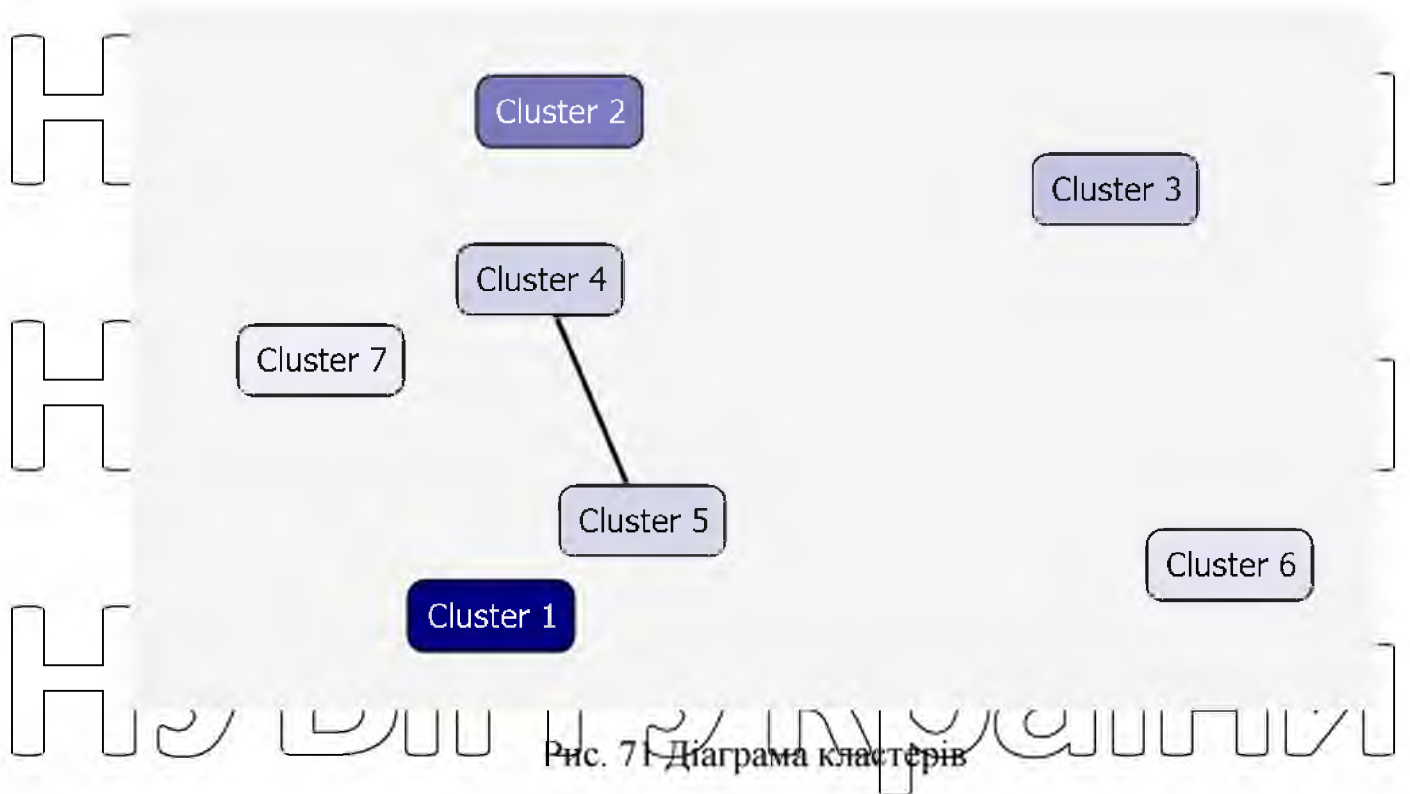


Рис. 71 Діаграма кластерів

Аналізуючи діаграму (див. рис. 71) можна зробити висновок, що найбільше даних зосереджено у кластері 1 ("Cluster 1"). Далі йдуть кластер 2 ("Cluster 2"), після якого кластери 3, 4 та 5.

На рис. 72 наведена детальна характеристика усіх кластерів у графічному вигляді.

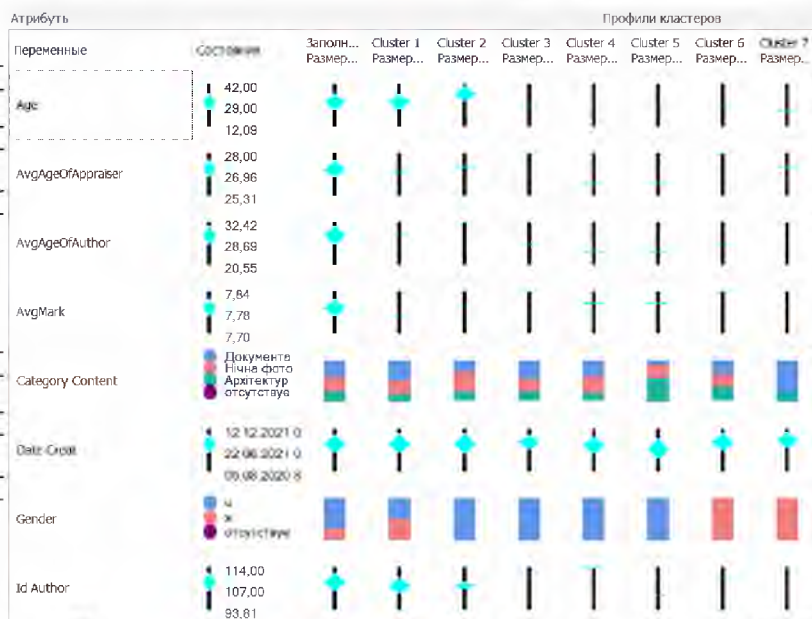


Рис. 72 Профілі кластерів

Для кращого розуміння розглянемо та нижче проаналізовано кожен кластер.

Кластер 1

Змінні	Значення	Імовірність
Name Education	Вища професійна	100,000 %
Gender	ч	50,847 %
Category Content	Документальна фотографія	49,153 %
Gender	ж	49,153 %
AvgAgeOfAuthor	26,9 - 28,7	47,508 %
Id Education	0 - 1	47,467 %
AvgAgeOfAuthor	28,7 - 30,5	45,738 %
Id Author	94 - 103	34,757 %
AvgAgeOfAppraiser	25,3 - 26,6	33,680 %
Category Content	Нічна фотографія	32,203 %
Id Author	104 - 106	28,786 %
Age	26 - 29	27,555 %
Age	30 - 32	26,555 %
Date Creat	05.08.2020 14:02:36 - 11.04.2021 3:43:07	24,661 %
Date Creat	22.06.2021 6:00:00 - 02.09.2021 8:16:52	24,546 %
Age	13 - 25	24,482 %
Date Creat	11.04.2021 3:43:07 - 22.06.2021 6:00:00	24,165 %
Id Education	2	23,226 %
Id Author	107 - 109	22,490 %
Age	33 - 42	20,841 %
Date Creat	02.09.2021 8:16:52 - 12.12.2021 0:00:00	20,315 %
AvgAgeOfAppraiser	27,3 - 28,0	18,808 %
Category Content	Архітектурна фотографія	18,644 %
AvgAgeOfAppraiser	26,6 - 27,0	14,677 %
AvgAgeOfAppraiser	27,0 - 27,3	14,075 %
Id Author	110 - 114	12,434 %
Id Education	3 - 2	10,361 %
AvgAgeOfAuthor	20,6 - 26,9	3,573 %
AvgAgeOfAuthor	30,5 - 32,4	3,173 %
Id Education	3 - 4	2,946 %
AvgMark	7,7 - 7,8	2,636 %
AvgMark	7,8	1,482 %

Кластер 1: Дані кластера містять роботи з категорії «архітектурна фотографія» (18,644 %), «документальна фотографія» (49,153 %), «нічна фотографія» (32,203 %), зроблені автором жінкою або чоловіком з вищою гуманітарною освітою (100 %), віком 26,9 - 28,7 років (47,508 %) і середньою оцінкою 7,7 - 7,8 (2,636 %).

Кластер 2

Змінні	Значення	Імовірність
Name Education	Вища гуманітарна	100,000 %

Gender	ч	100,000 %
Id/Author	104 - 106	62,463 %
Age	33 - 42	55,031 %
Category Content	Нічна фотографія	48,387 %
AvgAgeOfAuthor	30,5 - 32,4	47,131 %
Id Education	3 - 4	39,259 %
Category Content	Документальна фотографія	29,032 %
Date Creat	05.08.2020 14:02:36 - 11.04.2021 3:43:07	26,401 %
AvgAgeOfAppraiser	25,3 - 26,6	26,357 %
Id Education	3 - 2	26,301 %
Date Creat	11.04.2021 3:48:07 - 22.06.2021 6:00:00	23,903 %
Date Creat	22.06.2021 6:00:00 - 02.09.2021 8:16:52	23,698 %
Category Content	Архітектурна фотографія	22,581 %
AvgAgeOfAppraiser	27,3 - 28,0	22,518 %
Age	30 - 32	21,343 %
Date Creat	02.09.2021 8:16:52 - 12.12.2021 0:00:00	19,586 %
Id/Author	94 - 103	18,737 %
Id/Author	107 - 109	18,405 %
AvgAgeOfAppraiser	27,0 - 27,3	14,709 %
AvgAgeOfAppraiser	26,6 - 27,0	13,887 %
Id Education	2	13,697 %
Age	26 - 29	10,347 %
Id Education	0 - 1	4,742 %
Age	13 - 25	3,788 %
AvgAgeOfAuthor	28,7 - 30,5	2,859 %
AvgMark	7,7 - 7,8	2,636 %
AvgMark	7,8	1,482 %

Кластер 2: Дані кластера містять роботи з категорії «архітектурна фотографія» (22,581 %), «документальна фотографія» (29,032 %), «нічна фотографія» (48,387 %), зроблені автором чоловіком з вищою гуманітарною освітою (100 %), віком 30,5 - 32,4 років (47,131 %), де середній вік людей що оцінювали 25,3 - 26,6 років (26,357 %).

Кластер 3

Змінні	Значення	Імовірність
Name Education	Повна середня	100,000 %
Gender	ч	100,000 %
Id/Author	110 - 114	93,029 %
Age	26 - 29	76,255 %
AvgAgeOfAuthor	20,6 - 26,9	70,650 %
Category Content	Документальна фотографія	46,154 %
Id Education	3 - 4	37,030 %
Date Creat	22.06.2021 6:00:00 - 02.09.2021 8:16:52	33,080 %
Category Content	Нічна фотографія	30,769 %

AvgAgeOfAuthor	26,9 - 28,7	28,464 %
Date Creat	11.04.2021 3:43:07 - 22.06.2021 6:00:00	26,281 %
AvgAgeOfAppraiser	27,3 - 28,0	24,650 %
Date Creat	02.09.2021 8:16:52 - 12.12.2021 0:00:00	24,365 %
Age	13 - 25	23,642 %
Category Content	Архітектурна фотографія	23,077 %
Date Creat	05.08.2020 14:02:36 - 11.04.2021 3:43:07	12,585 %
AvgAgeOfAppraiser	27,0 - 27,3	10,302 %
Id Education	3 - 2	10,055 %
AvgAgeOfAppraiser	25,3 - 26,6	7,617 %
AvgAgeOfAppraiser	26,6 - 27,0	7,068 %
Id Author	107 - 109	6,836 %
AvgMark	7,7 - 7,8	2,620 %

Кластер 3: Дані кластера містять роботи з категорії «архітектурна фотографія» (23,077 %), «документальна фотографія» (46,154 %), «нічна фотографія» (30,769 %), зроблені автором чоловіком з повною середньою освітою (100 %), віком 20,6 - 26,9 років (70,650 %), де середній вік людей що оцінювали 27,3 - 28,0 років (24,650 %) з оцінкою роботи 7,7 – 7,8 (2,620 %).

Кластер 4

Змінні	Значення	Імовірність
Gender	ч	100,000 %
Name Education	Вища технічна	100,000 %
AvgAgeOfAuthor	20,6 - 26,9	99,499 %
Age	13 - 25	80,984 %
Id Author	110 - 114	50,000 %
AvgAgeOfAppraiser	25,3 - 26,6	47,725 %
Category Content	Документальна фотографія	40,000 %
Category Content	Нічна фотографія	40,000 %
Date Creat	11.04.2021 3:43:07 - 22.06.2021 6:00:00	28,832 %
Id Education	2	28,699 %
Date Creat	05.08.2020 14:02:36 - 11.04.2021 3:43:07	28,209 %
Id Education	3 - 2	26,567 %
Date Creat	22.06.2021 6:00:00 - 02.09.2021 8:16:52	25,338 %
Id Education	0 - 1	23,248 %
Category Content	Архітектурна фотографія	20,000 %
Id Education	3 - 4	16,936 %
Date Creat	02.09.2021 8:16:52 - 12.12.2021 0:00:00	15,194 %
AvgAgeOfAppraiser	26,6 - 27,0	10,892 %
AvgMark	7,7 - 7,8	2,620 %
AvgAgeOfAppraiser	27,0 - 27,3	7,663 %

Кластер 4: Дані кластера містять роботи з категорії «архітектурна фотографія» (20,000 %), «документальна фотографія» (40,000 %), «нічна фотографія» (40,000 %), зроблені автором чоловіком з вищою технічною освітою (100 %), віком 20,6 - 26,9 років (99,499 %), де середній вік людей що оцінювали 25,3 - 26,6 років (47,725 %) з оцінкою роботи 7,7 – 7,8 (2,620 %).

Кластер 5

Змінні	Значення	Імовірність
Name Education	Вища технічна	100,000 %
Gender	ч	100,000 %
Id Author	94 - 103	99,503 %
AvgAgeOfAuthor	20,6 - 26,9	99,499 %
Age	13 - 25	61,080 %
Category Content	Архітектурна фотографія	55,556 %
AvgAgeOfAppraiser	25,3 - 26,6	47,725 %
Date Creat	05.08.2020 14:02:36 - 11.04.2021 3:43:07	44,647 %
Age	26 - 29	38,920 %
Category Content	Нічна фотографія	33,333 %
Id Education	2	28,699 %
Id Education	3 - 2	26,567 %
Date Creat	11.04.2021 3:43:07 - 22.06.2021 6:00:00	23,597 %
Id Education	0 - 1	23,248 %
Date Creat	22.06.2021 6:00:00 - 02.09.2021 8:16:52	17,392 %
Id Education	3 - 4	16,936 %
Category Content	Документальна фотографія	11,111 %
AvgAgeOfAppraiser	26,6 - 27,0	10,892 %
Date Creat	02.09.2021 8:16:52 - 12.12.2021 0:00:00	10,648 %
AvgAgeOfAppraiser	27,0 - 27,3	

Кластер 5: Дані кластера містять роботи з категорії «архітектурна фотографія» (55,556 %), «документальна фотографія» (11,111 %), «нічна фотографія» (33,333%), зроблені автором чоловіком з вищою технічною освітою (100 %), віком 20,6 - 26,9 років (99,499 %), де середній вік людей що оцінювали 25,3 - 26,6 років (47,725 %).

Кластер 6

Змінні	Значення	Імовірність
Gender	ж	100,000 %
Name Education	Повна середня	100,000 %
Age	26 - 29	93,835 %

AvgAgeOfAuthor	20,6 - 26,9	70,650 %
Id Author	110 - 114	68,746 %
Id Education	3 - 4	37,030 %
Category Content	Нічна фотографія	33,333 %
Category Content	Архітектурна фотографія	33,333 %
Category Content	Документальна фотографія	33,333 %
Id Author	107 - 109	31,226 %
AvgAgeOfAuthor	26,9 - 28,7	28,464 %
Date Creat	22.06.2021 6:00:00 - 02.09.2021 8:16:52	26,805 %
AvgAgeOfAppraiser	27,3 - 28,0	24,650 %
Date Creat	11.04.2021 3:43:07 - 22.06.2021 6:00:00	23,706 %
Date Creat	02.09.2021 8:16:52 - 12.12.2021 0:00:00	23,533 %
Date Creat	05.08.2020 14:02:36 - 11.04.2021 3:43:07	18,950 %
AvgAgeOfAppraiser	27,0 - 27,3	10,302 %
Id Education	3 - 2	10,055 %
AvgAgeOfAppraiser	25,3 - 26,6	7,617 %

Кластер 6. Дані кластера містять роботи з категорії «архітектурна фотографія» (33,333 %), «документальна фотографія» (33,333 %), «нічна фотографія» (33,333 %), зроблені автором жінкою з повною середньою освітою (100 %), віком 20,6 - 26,9 років (70,650 %), де середній вік людей що оцінювали 27,3 - 28,0 років (24,650 %).

Кластер 7

Змінні	Значення	Імовірність
Gender	ж	100,000 %
Name Education	Вища гуманітарна	100,000 %
Age	13 - 25	98,869 %
Id Author	110 - 114	84,110 %
Category Content	Документальна фотографія	75,000 %
AvgAgeOfAuthor	30,5 - 32,4	47,131 %
Id Education	3 - 4	39,259 %
Date Creat	02.09.2021 8:16:52 - 12.12.2021 0:00:00	29,500 %
Date Creat	22.06.2021 6:00:00 - 02.09.2021 8:16:52	29,226 %
AvgAgeOfAppraiser	25,3 - 26,6	26,357 %
Id Education	3 - 2	26,301 %
Category Content	Архітектурна фотографія	25,000 %
AvgAgeOfAppraiser	27,3 - 28,0	22,518 %
Date Creat	11.04.2021 3:43:07 - 22.06.2021 6:00:00	21,034 %
AvgAgeOfAppraiser	27,0 - 27,3	14,709 %
AvgAgeOfAppraiser	26,6 - 27,0	13,887 %
Id Education	2	13,697 %
Date Creat		

Кластер 7: Дані кластера містять роботи з категорії «архітектурна фотографія» (25,000 %), «документальна фотографія» (75,000 %), зроблені автором жінкою з вищою гуманітарною освітою (100 %), віком 30,5 - 32,4 років (41,131 %), де середній вік людей що оцінювали 25,3 - 26,6 років (26,357 %).

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

ВИСНОВКИ

Завданням магістерської роботи було створення системи підтримки прийняття рішень з організації фотовиставок, яка допомогла б зробити процес оцінки всіх робіт більш швидшим та організованим.

Мета роботи була повністю реалізована. Була змодельована нечітка експертна система з вхідними та вихідними даними, а також правилами поведінки. На її основі в інформаційну систему проведення фотоконкурсів була впроваджена система підтримки.

В ході дослідження системи підтримки прийняття рішень були використані наступні методи:

- Метод задач класифікації. Можна зробити висновок, що на рівень оцінки робіт більшою мірою впливає користувач, який оцінює, а не освіта, яку має автор роботи хоча як можна помітити все ж вона впливає на рівень оцінки, яка отримана.
- Метод наївного Байеса. На основі побудованої мережі можна зробити висновок, що на середню оцінку безпосередньо впливають вік автора та людей, які оцінюють роботи.
- Метод алгоритмів кластеризації. Результатом розгортання структури є система кластерів, які в собі зберігають інформацію про категорії зображень, гендер, вік та освіту автора, середній вік оцінюючих, а також саму оцінку.

У процесі написання магістерської роботи було досліджено предметну область, сформовані вимоги для створення системи підтримки прийняття рішень, проведено дослідження декількома методами та побудовані відповідні алгоритми роботи.

Результатом даної роботи є готова система підтримки прийняття рішень з організації фотовиставок та дослідження, зроблені завдяки даним, взятим із системи.

Розроблена система допомагає, на основі оцінок, переглядів та коментарів користувачів, обрати найкращі роботи для подальшої взаємодії із журі, а

використані методи дозволили зробити певні висновки, які дали змогу проаналізувати фактори, які впливають на результат оцінки.

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Застосування UML (частина 2). Діаграма послідовності - sequence diagram. Державний Університет Телекомунікацій : веб-сайт. URL: <https://dut.edu.ua/ua/news-1-626-7897-zastosuvannya-uml-chastina-2-diagrama-poslidovnosti---sequence-diagram-kafedra-kompyuternih-nauk-ta-informacijnih-tehnologiy> (дата звернення: 01.11.2022).
2. Дудзяний І. М. Об'єктно-орієнтоване моделювання програмних систем : навчальний посібник. Львів : Видавничий центр ЛНУ імені Івана Франка, 2007. 108 с. URL: https://is.tntu.edu.ua/data/elibrarv/3/obp_uml.pdf (дата звернення: 01.11.2022).
3. Основні переваги субд MySQL. StudFiles : веб-сайт. URL: <https://studfile.net/preview/5607354/page:3/> (дата звернення: 01.11.2022).
4. Методы и модели анализа данных: OLAP и Data Mining / Барсегян А.А., Куприянов М. С., Степаненко В. В., Холод И. И. СПб : БУВ-Петербург, 2004. 336 с.
5. Overview of Service Manger OLAP cubes for advanced analytics. Microsoft : веб-сайт. URL: <https://learn.microsoft.com/uk-ua/system-center/scsm/olap-cubes-overview?view=sc-sm-2022> (дата звернення: 02.11.2022).
6. Data Warehouse Concepts. Amazon : веб-сайт. URL: https://aws.amazon.com/data-warehouse/?nc1=h_ls (дата звернення: 02.11.2022).
7. Сторінка даних та OLAP технології. Система електронного забезпечення навчання ЗНУ : веб-сайт. URL: https://moodle.znu.edu.ua/pluginfile.php?file=/4816&4/mod_resource/content/2/%20/%d0%9b%d0%b5%d0%ba%d1%86%d1%96%d1%8f%202.pdf (дата звернення: 02.11.2022).
8. Баклан І. В. Експертні системи. Курс лекцій : Навчальний посібник. Київ : Національна академія управління, 2012. 132 с. URL:

http://baklaniv.at.ua/MSAI/ekspertni_sistemi-kurs_lekcij.2012.pdf (дата звернення: 02.11. 2022).

9. Learn-One-Rule Algorithm. Geeksforgeeks : веб-сайт. URL: <https://www.geeksforgeeks.org/learn-one-rule-algorithm/> (дата звернення: 02.11. 2022).

10. Naive Bayes. Scikit-learn : веб-сайт. URL: https://scikit-learn.org/stable/modules/naive_bayes.html (дата звернення: 02.11. 2022).

11. Naive Bayes Classifier. Towards data science : веб-сайт. URL: <https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c> (дата звернення: 02.11. 2022).

12. Association Rule. Geeksforgeeks : веб-сайт. URL: <https://www.geeksforgeeks.org/association-rule/> (дата звернення: 02.11. 2022).

13. Association Rule Learning. Java T Point : веб-сайт. URL: <https://www.javatpoint.com/association-rule-learning> (дата звернення: 02.11. 2022).

14. Clustering in Data Mining. Java T Point : веб-сайт. URL: <https://www.javatpoint.com/data-mining-cluster-analysis> (дата звернення: 02.11. 2022).

НУБІП у країні

НУБІП у країні

НУБІП у країні

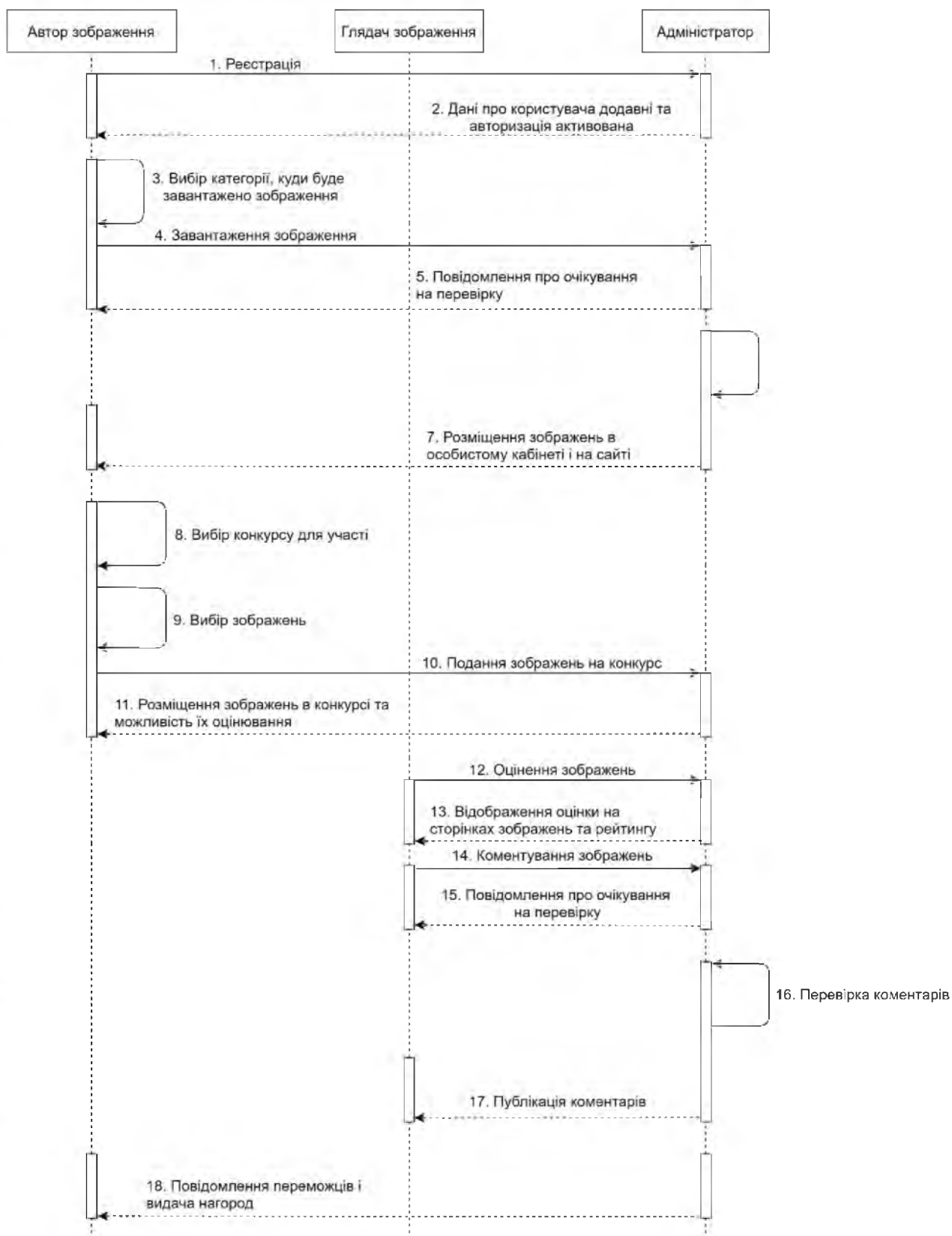
ДІАГРАМА ПОСЛІДОВНОСТІ

НУБІП у країні

НУБІП у країні

НУБІП у країні

НУБІП у країні



Діаграма послідовності

НУБІП УкРАЇНИ ДОДАТОК Б

НУБІП УкРАЇНИ

НУБІП УкРАЇНИ

ДІАГРАМА АКТИВНОСТІ

НУБІП УкРАЇНИ

НУБІП УкРАЇНИ

Сторінок - 1

НУБІП УкРАЇНИ

НУБІП УкРАЇНИ

НУБІП УМІТІНІ

НУБІ

ІНІ

НУБІ

ІНІ

НУБІ

ІНІ

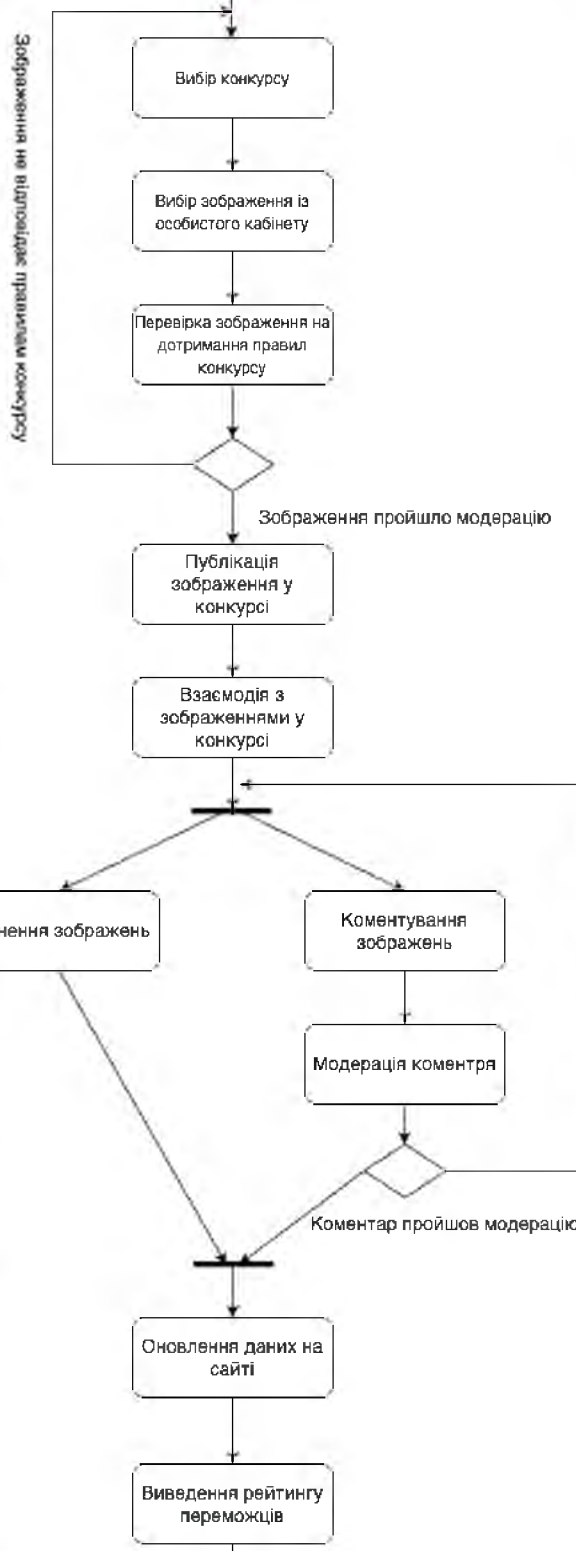
НУБІ

ІНІ

НУБІ

ІНІ

НУБІ



Зображення не відповідає правилам конкурсу

Зображення пройшло модерацію

Коментар не пройшов модерацію

Коментар пройшов модерацію

Діаграма активності

НУБІП України

НУБІП України

НУБІП України

СТВОРЕННЯ БАЗИ ДАНИХ

НУБІП України

НУБІП України

Сторінок - 4

НУБІП України

НУБІП України

Створення таблиць

```
CREATE TABLE IF NOT EXISTS `categories` (
  `id` int(1) NOT NULL AUTO_INCREMENT,
  `name` varchar(25) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;
```

```
CREATE TABLE IF NOT EXISTS `contest` (
```

```
  `id` int(1) NOT NULL AUTO_INCREMENT,
  `active` int(1) NOT NULL,
  `title` varchar(100) NOT NULL,
  `description` varchar(500) NOT NULL,
  `end` datetime NOT NULL,
```

```
  PRIMARY KEY (`id`),
  KEY `active` (`active`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;
```

```
CREATE TABLE IF NOT EXISTS `content` (
```

```
  `id` int(1) NOT NULL AUTO_INCREMENT,
  `iduser` int(1) NOT NULL,
```

```
  `photo` varchar(33) NOT NULL DEFAULT "",
```

```
  `date` timestamp NOT NULL DEFAULT current_timestamp(),
```

```
  `rating` float NOT NULL DEFAULT 0,
```

```
  `nr_rating` int(1) NOT NULL DEFAULT 0,
```

```
  `description` varchar(100) NOT NULL DEFAULT "",
```

```
  `views` int(1) NOT NULL DEFAULT 0,
```

```
  `category` int(1) NOT NULL DEFAULT 0,
```

```
  `approved` enum('0','1') NOT NULL DEFAULT '0',
```

```
  PRIMARY KEY (`id`),
```

```

KEY `iduser` (`iduser`),
KEY `category` (`category`),
KEY `rating` (`rating`),

```

```

KEY `views` (`views`),

```

```

KEY `approved` (`approved`)

```

```

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;

```

```

CREATE TABLE IF NOT EXISTS `ratings` (

```

```

`id` int(1) NOT NULL AUTO_INCREMENT,

```

```

`iduser` int(1) NOT NULL DEFAULT 0,

```

```

`rate` tinyint(1) NOT NULL DEFAULT 0,

```

```

`date` timestamp NOT NULL DEFAULT current_timestamp(),

```

```

`photo_id` int(1) NOT NULL,

```

```

PRIMARY KEY (`id`),

```

```

KEY `photo_id` (`photo_id`),

```

```

KEY `iduser` (`iduser`)

```

```

) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;

```

```

CREATE TABLE IF NOT EXISTS `users` (

```

```

`id` int(1) NOT NULL AUTO_INCREMENT,

```

```

`email` varchar(50) NOT NULL,

```

```

`name` varchar(30) NOT NULL,

```

```

`password` varchar(128) NOT NULL DEFAULT "",

```

```

`user` varchar(30) NOT NULL DEFAULT "",

```

```

`registered` timestamp NOT NULL DEFAULT current_timestamp(),

```

```

`slogan` varchar(50) NOT NULL DEFAULT "",

```

```

`profile_picture` varchar(50) NOT NULL DEFAULT "",

```

```

PRIMARY KEY (`id`),

```

```

KEY `email` (`email`),

```

```
KEY `password` (`password`),
```

```
KEY `user` (`user`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;
```

```
CREATE TABLE IF NOT EXISTS `comments` (
```

```
`id` int(1) NOT NULL AUTO_INCREMENT,
```

```
`photo_id` int(1) NOT NULL DEFAULT 0,
```

```
`user_id` int(1) NOT NULL DEFAULT 0,
```

```
`comment` varchar(250) NOT NULL DEFAULT "",
```

```
`date` timestamp NOT NULL DEFAULT current_timestamp(),
```

```
PRIMARY KEY (`id`),
```

```
KEY `photo_id` (`photo_id`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;
```

```
CREATE TABLE IF NOT EXISTS `settings` (
```

```
`id` int(1) NOT NULL AUTO_INCREMENT,
```

```
`set_key` varchar(50) NOT NULL,
```

```
`set_value` varchar(150) NOT NULL,
```

```
PRIMARY KEY (`id`),
```

```
UNIQUE KEY `set_key` (`set_key`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COLLATE=utf8_general_ci;
```

Внесення умовно-постійної інформації

Інформація про категорії

```
INSERT INTO `categories` (`id`, `name`) VALUES
```

```
(2, 'Архітектура'),
```

```
(3, 'Аніме'),
```

```
(4, 'Подорожі'),
```

```
(5, 'Гумор'),
```


НУБІП у країні

(6, 'Садівництво'),

(7, 'Дизайн'),

(8, 'Мистецтво'),

(9, 'Тварини'),

НУБІП у країні

(10, 'Розваги'),

(11, 'Спорт'),

(12, 'Стріт-арт'),

(13, 'Транспорт'),

НУБІП у країні

(14, 'Освіта'),

(15, 'Татуювання'),

(16, 'Іжа');

Інформація про налаштування

НУБІП у країні

INSERT INTO `settings` (`id`, `set_key`, `set_value`) VALUES

(1, 'site_url', 'http://picturest.skymod.com.ua/'),

(2, 'site_logo', 'Picturest'),

(3, 'admin_user', 'admin'),

(4, 'admin_pass', '".hash('sha512','password123')."'),

НУБІП у країні

(5, 'max_uploadsize', '20'),

(6, 'max_files', '5');

НУБІП у країні

НУБІП у країні