

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**

15.03 — КМР. 1939 –“С” 2022.12.30. 04 ПЗ

**ЗАБОЛОТЬКА ДМИТРА ВОЛОДИМИРОВИЧА**

2023 р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ  
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

**Факультет інформаційних технологій**

УДК 004.4:004.738

**«ПОГОДЖЕНО»**

**«ДОПУСКАЄТЬСЯ ДО**

Декан факультету

**ЗАХИСТУ»**

інформаційних технологій

Завідувач кафедри комп'ютерних наук

Глазунова О.Г., д.п.н., професор

Голуб Б.Л., к.т.н., доцент

\_\_\_\_\_ 202\_ р.

\_\_\_\_\_ 202\_ р.

**МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**

на тему Аналітична система для соціальної мережі

Спеціальність 121 - Інженерія програмного забезпечення

(код і назва)

Освітня програма Програмне забезпечення інформаційних систем

(назва)

Орієнтація освітньої програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

**Гарант освітньої програми**

К.Т.Н., доцент

(науковий ступінь та вчене звання)

(підпис)

(ПІБ)

**Керівник магістерської кваліфікаційної роботи**

ст. викладач

(науковий ступінь та вчене звання)

(підпис)

Міловідов Ю.О.

(ПІБ)

**Виконав**

(підпис)

Заболотько Д.В.

(ПІБ студента)

КИЇВ-2023

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ  
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ**

Факультет (ННІ) \_\_\_\_\_ інформаційних технологій \_\_\_\_\_

**ЗАТВЕРДЖУЮ**

Завідувач кафедри \_\_\_\_\_

\_\_\_\_\_ комп'ютерних наук \_\_\_\_\_

\_\_\_\_\_ К.Т.Н., доцент \_\_\_\_\_ Голуб Б. Л.

(науковий ступінь, вчене звання) (підпис) (ПІБ)

“ \_\_\_\_\_ ” \_\_\_\_\_ року

**З А В Д А Н Н Я**

**ДО ВИКОНАННЯ МАГІСТЕРСЬКОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ СТУДЕНТУ**

\_\_\_\_\_ Заболотьку Дмитру Володимировичу \_\_\_\_\_

(прізвище, ім'я, по батькові)

Спеціальність \_\_\_\_\_ 121 - Інженерія програмного забезпечення \_\_\_\_\_

(код і назва)

Освітня програма \_\_\_\_\_ Програмне забезпечення інформаційних систем \_\_\_\_\_

(назва)

Орієнтація освітньої програми \_\_\_\_\_ освітньо-професійна \_\_\_\_\_

(освітньо-професійна або освітньо-наукова)

Тема магістерської кваліфікаційної роботи \_\_\_\_\_ Аналітична система для соціальної мережі \_\_\_\_\_

затверджена наказом ректора НУБіП України від “ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_\_\_ р. № \_\_\_\_\_

Термін подання завершеної роботи на кафедру \_\_\_\_\_

(рік, місяць, число)

Вихідні дані до магістерської кваліфікаційної роботи \_\_\_\_\_

1. Дані із соцмережі \_\_\_\_\_

2. <https://dou.ua/forums/tags/tech/?from=fortech> \_\_\_\_\_

Перелік питань, що підлягають дослідженню:

1. Публікації користувачів в розробленій соціальній мережі \_\_\_\_\_

2. Моделювання та розробка соцмережі та аналітичного модуля \_\_\_\_\_

3. Використання OLAP та Data Mining технологій для аналізу публікацій \_\_\_\_\_

Перелік графічного матеріалу (за потреби) \_\_\_\_\_

Дата видачі завдання “ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_\_\_ р.

Керівник магістерської кваліфікаційної роботи \_\_\_\_\_ Міловідов Ю.О. \_\_\_\_\_

(підпис)

(прізвище та ініціали)

Завдання прийняв до виконання \_\_\_\_\_ Заболотько Д.В. \_\_\_\_\_

(підпис)

(прізвище та ініціали студента)

## ЗМІСТ

Перелік умовних позначень .....	4
Вступ.....	5
1 Системний аналіз предметної області.....	7
1.1 Опис предметної області .....	7
1.2 Аналіз існуючих рішень .....	8
1.3 Постановка завдань дослідження.....	11
2 Моделювання системи.....	13
2.1 Об'єктно-орієнтоване моделювання.....	13
2.2 Опис операцій технологічного процесу обробки даних .....	16
3 Розробка системи .....	18
3.1 Інформаційне забезпечення системи .....	18
3.2 Програмне забезпечення системи .....	25
4 Результати.....	39
4.1 Інтерфейс соцмережі.....	39
4.2 Результати аналітичного модуля.....	44
Висновки .....	57
Список використаних джерел.....	59

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

CSS – Cascading Style Sheets

DM – Data Mining

ERD – Essence Relation Diagramming

JS – JavaScript

JWT – JSON Web Token

OLAP – Online analytical processing

SADT – Structured Analysis and Design Technique

SASS – Syntactically Awesome Style Sheets

SSAS – SQL Server Analysis Services

UML – Unified Modeling Language

БД – база даних

ІТ – інформаційні технології

СУБД – система управління базою даних

## ВСТУП

Соціальні мережі стали феноменом 21 століття, збільшивши розмаїття соціальної взаємодії та створивши додаткові можливості для комерції в різних сферах життя. За незначних витрат – девайс та доступ до мережі Інтернет, соціальні мережі надають багато способів представлення власних думок, досягнень, знань, послуг чи товарів на широкий огляд. Такий спосіб комунікації вимагає значних ресурсів у власників соцмереж – потужні сервера, призначені обробляти величезний потік запитів, зберігати масиви різної інформації та оперативно представляти результати. Щоб забезпечувати такі, на перший погляд прості, але неймовірно масштабні процеси важливо безперервно вдосконалювати системи в цілому та запроваджувати все нові функції для збереження конкурентоспроможності.

Для розвитку та вдосконалення соціальних мереж застосовуються різноманітні технології. Соцмережі отримують дуже великі об'єми різних даних, отже представляють собою перспективне джерело актуальних даних. Наприклад, створення та реалізація інтелектуальних алгоритмів підбору контенту спростила пошук публікацій для стрічки новин. Дані алгоритми засновані на аналізі зазначених вподобань, переглянутого контенту та пошукових запитів конкретного користувача. Але з популярністю соцмереж, продовжують рости масиви накопичених даних, що можуть бути використані для аналізу, а отже туди потрапляє все більше неякісних даних. Тому, з часом, з'явився попит на спеціалізовані мережі, що могли б забезпечувати схожий функціонал, але містити менше розмаїття інформації, щоб забезпечило більшу чистоту даних.

Актуальність дослідження полягає у визначенні технологій, що користуються попитом в соцмережі сфері ІТ, спрощуючи вибір напрямків навчання початківцям та допомагаючи визначити інструменти для роботи спеціалістам.

Об'єктом дослідження виступають дописи користувачів в розробленій соціальній мережі.

Предмет – аналітична система дописів для визначення та прогнозування популярності технологій ІТ.

Мета дослідження полягає в застосуванні технологій OLAP та Data Mining (DM) для аналізу контенту в розроблюваній соціальній мережі.

Завдання дослідження:

- реалізувати систему аналізу контенту соціальної мережі використовуючи технології OLAP та DM;
- проаналізувати отримані результати від системи та зробити висновки щодо ефективності застосування даних технологій в сфері соціальних мереж.

Методи дослідження: OLAP та DM.

Наукова новизна дослідження полягає в застосування технологій React.js, Redux, SASS для розробки та застосування OLAP та DM для дослідження даних в сфері соціальних мереж професійного спрямування для прогнозування популярності різних напрямків в ІТ.

Пояснювальна записка містить чотири основні розділи, висновки та перелік використаних джерел. Перший розділ містить системний аналіз, що описує особливості предметної області та поставлені задачі програмного забезпечення. Наступний розділ демонструє процес створення об'єктно-орієнтованої моделей. Третій розділ представляє два головні етапи розробки – інформаційне та програмне забезпечення системи. В четвертому розділі наведені результати роботи інтелектуальних структур та аналіз ефективності їх використання.

# 1 СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Опис предметної області

Соціальна мережа [1] – це онлайн-платформа, яка дозволяє користувачам спілкуватися, ділитися інформацією, створювати та дивитися контент, додавати друзів тощо. Ось деякі з можливостей соціальної мережі:

- Спілкування: користувачі можуть спілкуватися між собою за допомогою текстових повідомлень, відеодзвінків або голосових повідомлень.
- Пошук та додавання друзів: користувачі можуть знайти та додати друзів на платформі, щоб бути в курсі їхніх активності та дізнаватися новини про них.
- Публікація контенту: користувачі можуть створювати та ділитися зображеннями, відео, аудіо, текстовими повідомленнями та іншими типами контенту на своїх сторінках або в спільнотах.
- Створення спільнот: користувачі можуть створювати спільноти на платформі, щоб обговорювати спільні інтереси та ділитися контентом.
- Розміщення реклами: соціальні мережі надають можливість рекламувати товари та послуги в різних форматах на їхніх платформах, щоб привернути увагу користувачів.
- Використання чат-ботів: деякі соціальні мережі дозволяють користувачам взаємодіяти з автоматизованими чат-ботами, які можуть надавати різноманітну інформацію та відповідати на запитання.
- Відстеження активності: користувачі можуть відстежувати активність своїх друзів та стежити за новинами від певних брендів, компаній або персоналій.

Так як в соцмережах, як правило, відсутні обмеження на кількість, наповненість та формат публікації контенту, виникає перенасичення даними, що



можуть не представляти значимості з точки зору аналізу. Саме для забезпечення схожих потреб було створенні та запроваджені, так звані, професійні соцмережі.

Професійні соцмережі мають свої особливості в порівнянні з загальними. В таких системах зберігається звичні поняття – користувач, публікація, підписка, діалог і т. п. Але відмінність полягає в можливості розширити інформацію про користувача, наприклад вказати теперішню посаду, місце та період роботи, вказати додаткові відомості про власні навички чи щось подібне. Такими особливостями користується і не тільки система, підбираючи певні публікації чи відкриті вакансії, а й інші користувачі, що бажають, наприклад найняти персонал з певними знаннями чи досвідом.

## **1.2 Аналіз існуючих рішень**

Програмне забезпечення веб версії соціальної мережі може бути застосовано в багатьох різних галузях. Деякі з них включають:

- Соціальна комунікація: основна мета соціальних мереж полягає в тому, щоб допомогти людям спілкуватися, обмінюватися інформацією та взаємодіяти один з одним. Програмне забезпечення веб версії соціальної мережі допомагає забезпечити цю функцію шляхом надання користувачам зручного інтерфейсу для створення та управління своїми профілями, пошуку та додавання друзів, обміну повідомленнями тощо.
- Реклама та маркетинг: соціальні мережі є важливим каналом для реклами та маркетингу. Програмне забезпечення веб версії соціальної мережі може бути використане для розміщення рекламних оголошень, залучення цільової аудиторії та проведення аналізу результатів рекламної кампанії.
- Освіта та навчання: соціальні мережі можуть бути використані в навчальних цілях, де вони можуть допомогти студентам та викладачам обмінюватися інформацією, викладати та вчитися новому матеріалу, обговорювати питання тощо. Програмне забезпечення веб версії

соціальної мережі може бути використане для створення таких навчальних платформ.

- Бізнес та економіка: соціальні мережі можуть допомогти бізнесу залучити нових клієнтів та збільшити продажі. Програмне забезпечення веб версії соціальної мережі може бути використано для створення комунікаційної плат.

Популярними представниками професійних соцмереж є:

- LinkedIn;
- Stackoverflow;
- Facebook.

LinkedIn [2] – соцмережа, орієнтована на бізнес та працевлаштування. Основною особливістю даної системи полягає в наявності двох основних типів користувачів – працівник та компанія. Кардинальна різниця в цих ролях полягає в наданні різного функціоналу. Наприклад користувач-працівник може прикріпити власне резюме, вказати навички та досвід безпосередньо в системі і саме на основі цих даних алгоритми почнуть підбирати відповідні вакансії та публікації, що могли б зацікавити. Щодо користувача-компанії, надавши всі необхідні дані про себе та створивши потрібні вакансії, алгоритми швидко підберуть кандидатів з відповідною кваліфікацією. Саме такі розширені можливості різних типів користувачів дозволяють ефективно використовувати пошукові алгоритми системи та швидко знаходити потрібну інформацію.

Переваги даної соцмережі [3] полягають:

- наявні власні веб-сайт та мобільний додаток;
- розширені характеристики профілю користувача;
- помітні різні статуси користувачів;
- забезпечення комунікації між користувачами.

Недоліками системи можна назвати:

- неконтрольована можливість встановлення зв'язків з іншими учасниками;
- підтвердження навичок іншого користувача;

- постійна розсилка листів на електронну пошту.

Stackoverflow [4] – веб-сайт для програмістів, що заснований на принципі запитання-відповідь. Мережа повністю зав'язана на знаннях людей, що, як правило, пропонують готове рішення проблеми або радять корисні посилання, де можна знайти відповідь.

Переваги Stackoverflow [5]:

- можливість оцінювання відповіді чи рішення іншого користувача;
- досить велика та різноманітна база запитань, на які вже давно відповіли провідні користувачі-спеціалісти;
- відповіді представляють собою готове рішення заданого питання.

До недоліків можна віднести:

- запитання мають бути чітко сформовані та стосуватись саме програмування;
- запропоновані відповіді, незважаючи на високий рейтинг, часто можуть мати розбіжності з офіційною документацією.

Facebook [6] – дуже популярна та відома соцмережа з одним із найбільшим числом користувачів. Мережа була створена для швидкого доступу до каталогів студентів з особистою інформацією. Система виявилась дуже успішна і швидко почала набирати популярності.

Переваги соцмережі [7] полягають в:

- наявні власні веб-сайт та мобільний додаток;
- можливість доєднатись до групи інших користувачів за інтересами.

Недоліки Facebook:

- потрібно встановлювати окремий додаток для спілкування;
- перенасичення публікаціями великої кількості користувачів.

### 1.3 Постановка завдань дослідження

Загалом дослідження складається з двох основних завдань – створення системи аналізу та проведення аналізу отриманих результатів. Питання на які має дати відповіді представлене дослідження такі:

1. Які напрямки в ІТ зараз більш популярні?
2. Чи існує зв'язок між посадою користувача та темою його публікацій?

**1.3.1 Вимоги до соцмережі.** Програма повинна забезпечувати можливість виконання наступних функцій:

- реєстрація нових облікових записів;
- заповнення інформації про користувача;
- перегляд новин на сайті;
- підписка на інших користувачів;
- написання власних постів в профілі;
- чати.

Вимоги до надійності системи:

- передбачити авторизацію різних користувачів в системі;
- передбачити блокування некоректних дій користувача при роботі з системою.

Вимоги до складу і параметрів технічних засобів:

- система повинна працювати в будь-якому веб-браузері;
- мінімальна конфігурація для пристроїв взаємодії повинна відповідати мінімальним запитам для користування веб-браузерами.

Вимоги для інформаційної та програмної сумісності:

- Система повинна працювати на версіях веб-браузерів молодше 3 років.

Вимоги до інтерфейсу користувача:

- Інтерфейс повинен бути зрозумілим для користувачів системи та виконаним в приємній кольоровій гаммі.
- Повинен містити такі пункти, як:
  - реєстрація/авторизація;

- заповнення інформації<sup>4</sup>
- пошук користувачів;
- перегляд новин;
- перегляд власного запису.

## 2 МОДЕЛЮВАННЯ СИСТЕМИ

### 2.1 Об'єктно-орієнтоване моделювання

Unified Modeling Language (UML) [8] – уніфікована мова моделювання, яка являє собою мову візуального моделювання, яка призначена для візуалізації, специфікації, конструювання й документування систем. Головна перевага застосування UML є стандартизований підхід до опису та створенню діаграм, що дозволяє використовувати її для демонстрації складних бізнес процесів та взаємозв'язків програмних модулів.

Будь-яка мова складається зі словника і правил комбінування слів для отримання осмислених конструкцій. Такою є і UML. Відмінною його рисою є те, що словник мови утворює графічні елементи. Кожному графічному символу відповідає конкретна семантика, тому модель, створена одним розробником, може однозначно бути зрозуміла іншим, а також програмним засобом, який інтерпретує UML.

Речі, які простіше виразити графічно, виражаються мовою UML, а ті, що легше виразити у вигляді тексту виражаються мовою програмування.

Слід підкреслити, що UML – це саме мова, а не метод. В цій мові не повідомляється про те, що потрібно робити першим, а що останнім. Для UML діаграм існують три основні типи візуальних позначень:

- зв'язки;
- текст;
- графічні символи.

**2.1.1 Діаграма прецедентів.** Діаграма прецедентів [9] – послідовність дій, ініційована актором(людина або системою), яка взаємодіє з інформаційною системою або іншими акторами і отримує в результаті деяке повідомлення від неї.

Діаграма прецедентів призначена для представлення сценаріїв взаємодії між акторами і прецедентами та опису функціональних аспектів системи.

Основним призначенням діаграми прецедентів є опис функціональності і поведінки, що дозволяє замовнику, користувачеві і розробнику спільно обговорювати проєктовану систему.

Основними компонентами діаграми, що представлена на рисунку 1 є:

- Актор — діюча особа (Користувач, аналітик, модератор).
- Прецедент — випадок використання, дія (Реєстрація, авторизація, Збір даних з бази даних і т. д.).
- Граничні межі системи — охоплюють усі випадки використання у системі.

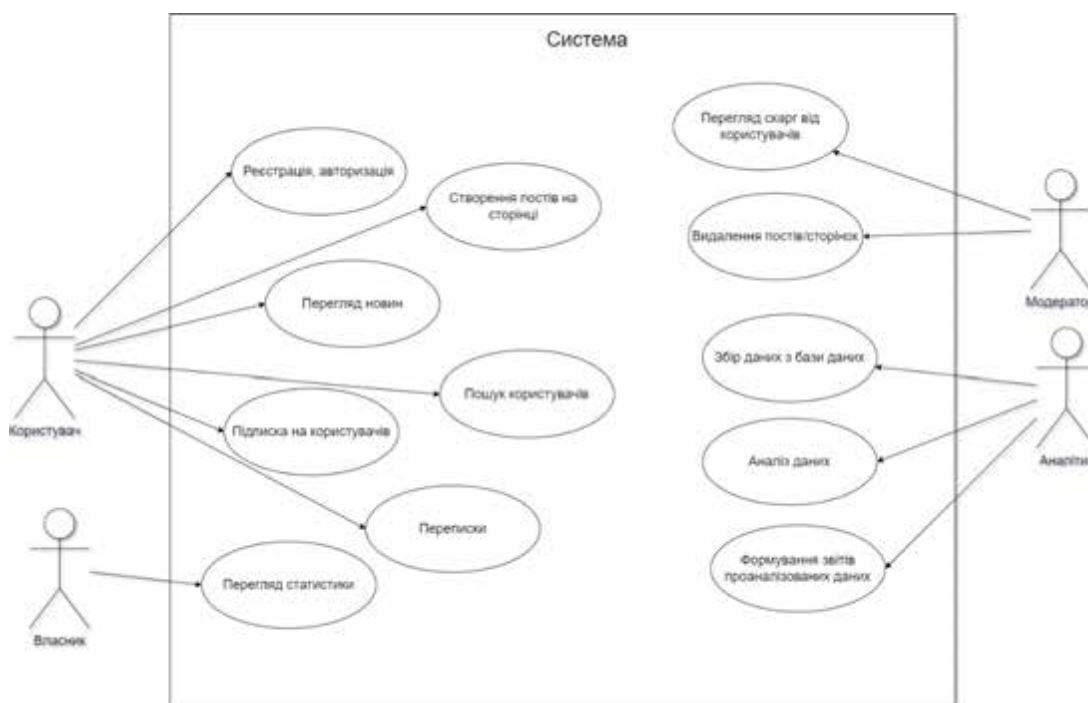


Рис. 2.1 Діаграма прецедентів соцмережі

Діаграма прецедентів демонструє, що система має надавати досить багато функцій, в залежності від користувача. Отже для системи важливу роль відіграє тип користувача, адже згідно цього надаються різні області доступу.

**2.1.2 Діаграма станів.** Діаграма станів — діаграма, що визначає зміну станів об'єкту у часі, одна з діаграм моделювання поведінки в UML [10]. Подає об'єкт як автомат з теорії автоматів зі стандартизованими умовними позначеннями.

Елементами діаграми є:

- Коло, що позначає початковий стан.
- Коло з маленьким колом усередині, що позначає кінцевий стан (якщо є).
- Округлений прямокутник, що позначає окремий стан. Верхівка прямокутника містить назву стану, в середині може бути горизонтальна лінія, під якою записуються активності, що відбуваються в даному стані.
- Стрілка, що позначає перехід. Назва події (якщо є), що викликає перехід, відзначається над/під стрілкою. Вартовий вираз може бути доданий перед «/» і укладений у квадратні дужки (назва\_події), він означає, що перехід відбувається лише за умови істинності виразу. Якщо при переході відбувається якась активність, то воно додається після «/» (назва події).
- Товста горизонтальна лінія, яка є точкою об'єднання або розгалуження переходів.

На рисунку 2.2 представлена діаграма станів для процесу «Авторизація». Вона демонструє головні стани системи під час авторизації користувача через браузер в соцмережу.

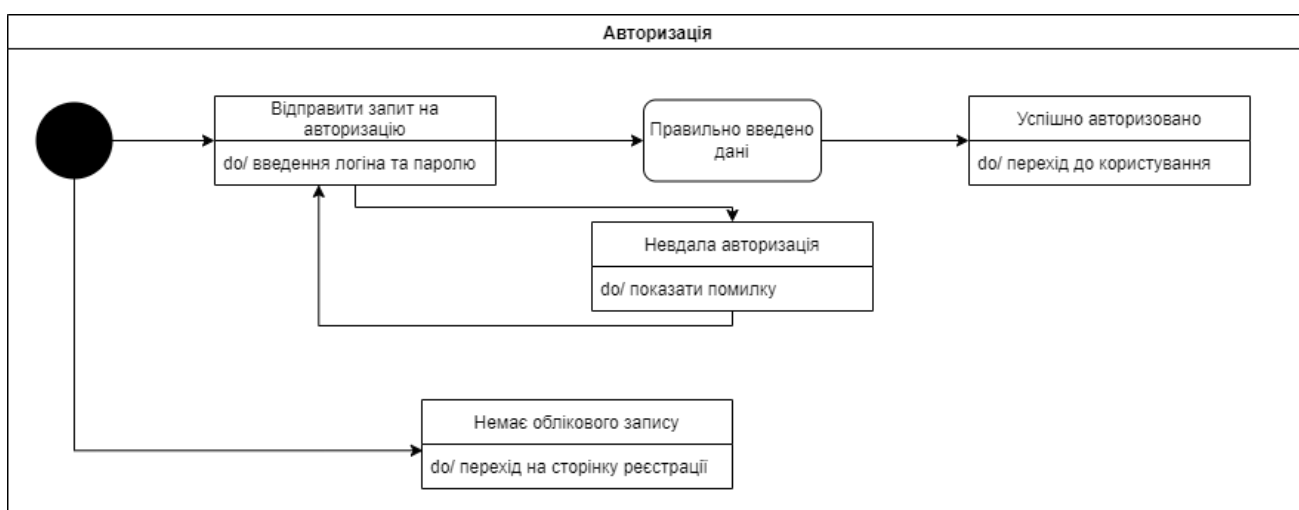


Рис. 2.2 Діаграма станів для Авторизації



На рисунку 2.3 представлена діаграма станів сторінки профіль. Вони має дві основні вітки – зміна статусу та публікація нового допису.

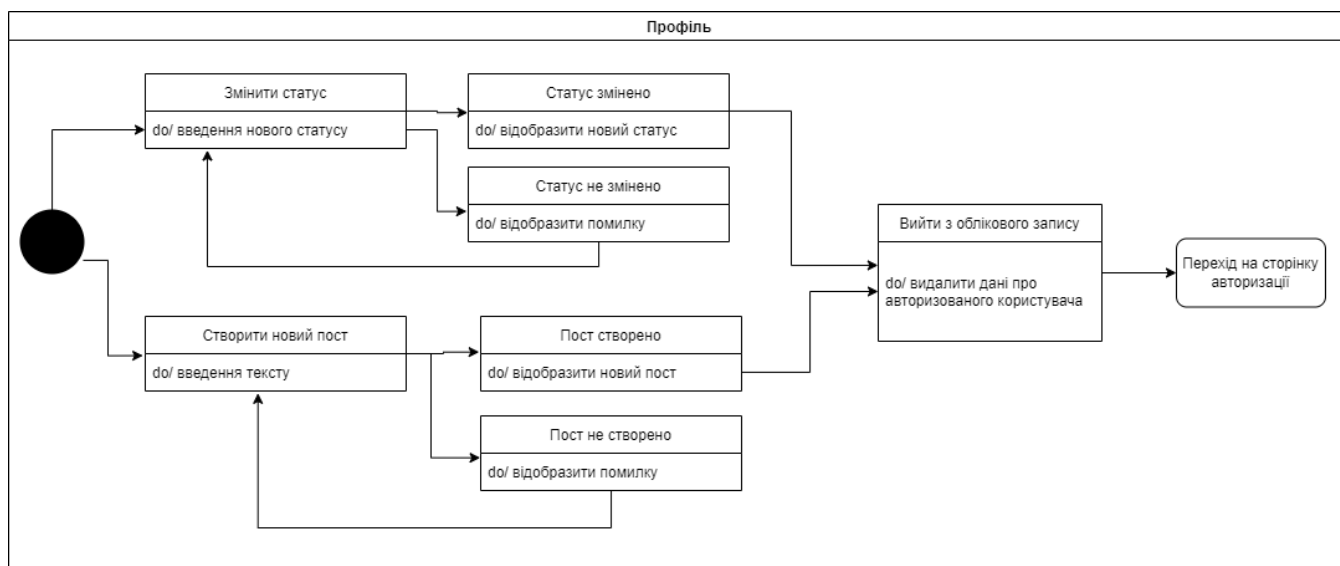


Рис. 2.3 Діаграма станів під час роботи на сторінці Профіль

## 2.2 Опис операцій технологічного процесу обробки даних

Технологічний процес обробки даних в соціальній мережі може включати наступні операції:

- **Збір та зберігання даних.** Для роботи соціальної мережі необхідно збирати та зберігати дані користувачів, такі як особисті дані, друзі, повідомлення, пости, зображення та відео. Ці дані зазвичай зберігаються в базі даних.
- **Обробка даних.** Дані можуть бути оброблені для забезпечення швидкодії та ефективності роботи соціальної мережі. Це може включати індексацію даних, щоб забезпечити швидкий пошук, а також оптимізацію запитів до бази даних.
- **Аналіз даних.** Дані можуть бути проаналізовані, щоб дати корисну інформацію про користувачів та їх поведінку на сайті. Наприклад, можна аналізувати, які типи контенту найбільш популярні серед користувачів, які групи користувачів взаємодіють між собою тощо.

- **Забезпечення безпеки.** Також важливо забезпечити безпеку даних користувачів, щоб забезпечити їх конфіденційність та запобігти злому. Це може включати захист від хакерів, захист від вірусів та інших шкідливих програм.
- **Відображення даних.** Дані можуть бути відображені на веб-сторінках соціальної мережі, щоб користувачі могли взаємодіяти з ними. Це може включати відображення профілів користувачів, повідомлень, постів та іншого контенту.

## 3 РОЗРОБКА СИСТЕМИ

### 3.1 Інформаційне забезпечення системи

**3.1.1 Логічна модель даних.** Логічна модель [11] покликана в графічному вигляді представити основні дані, що беруть участь у процесах, що підлягають автоматизації. Таке представлення надає можливість оцінити складність розробки джерела даних, його тип та технології його розробки.

На рисунку 3.1 представлена логічна схема даних соцмережі. Вона містить схематичний вигляд зв'язків між сутностями та інформацією, що вони зберігають, а саме:

- Post – сутність зберігає дані про публікації користувачів та відповідну супутню інформацію, таку як текст, дата публікації, фото, кількість лайків та т.п..
- User – сутність містить дані про користувача соцмережі та має зв'язок з іншими сутностями. Поля сутності мають зберігати важливі дані про акаунт користувача, а саме ім'я, електронна пошта, фото, посилання на акаунти з інших соцмереж та т.п..
- Dialogs – сутність має містити інформацію про чати користувачів та мати поля, призначені записувати код користувача, останнє повідомлення, останнє оновлення та т.п..
- Messages – сутність представляє собою повідомлення, яким обмінюється користувач.
- Followers – сутність містить дані про користувачів, що слідкують за користувачем.
- Following – сутність має інформацію про те, яких користувачів відстежує користувач.

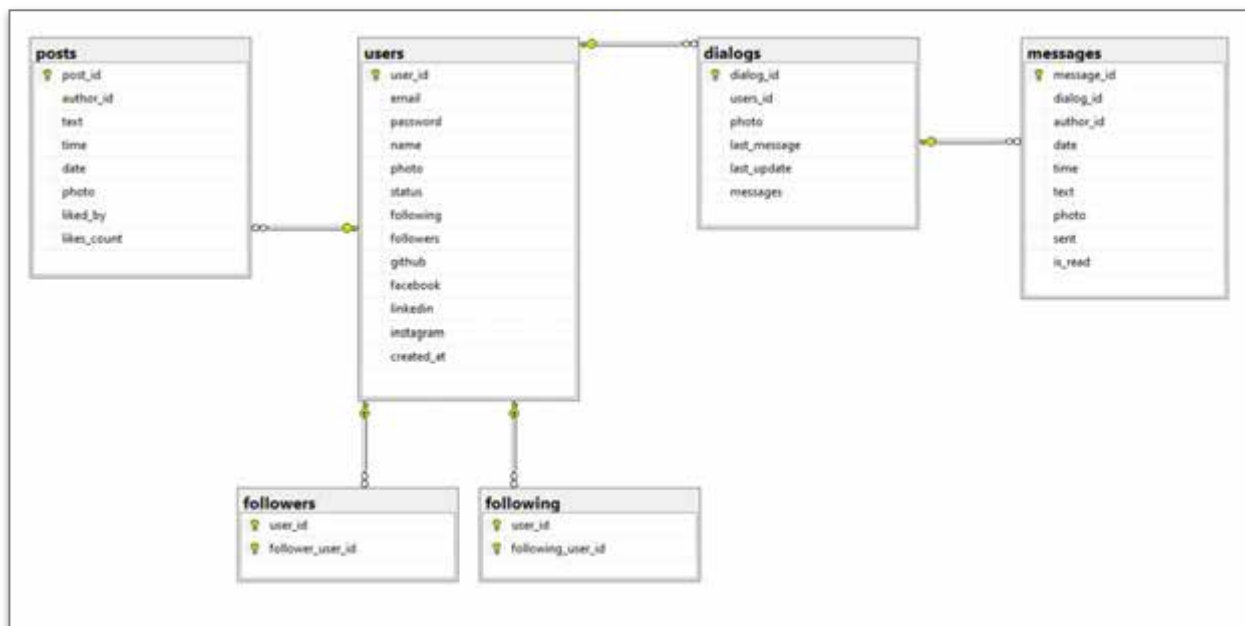


Рис. 3.1 Логічна схема соцмережі

**3.1.2 Вибір середовища проектування БД.** База даних, яка використана при розробці – нереляційна MongoDB [12]. MongoDB - це документ-орієнтована система управління базами даних (СУБД), що зберігає дані у вигляді документів у форматі BSON (Binary JSON). MongoDB розроблена з урахуванням масштабованості та високої доступності. Вона дозволяє зберігати великі об'єми даних, швидко їх зчитувати та записувати, та легко масштабувати.

Однією з головних особливостей MongoDB [13] є те, що вона дозволяє зберігати дані у вигляді документів зі структурою даних, подібно до формату JSON. Кожен документ може містити будь-яку кількість ключів та значень, які можуть бути вкладеними один в одного. Це дає можливість зберігати дані без використання схеми бази даних, що дозволяє швидко змінювати структуру даних.

MongoDB також має розширені можливості для масштабування та реплікації даних. Дані можуть бути розподілені на кілька серверів, що дозволяє забезпечити високу доступність та швидкість роботи з даними. MongoDB також підтримує реплікацію даних, що дозволяє створювати резервні копії даних та забезпечувати відновлення у випадку відмови сервера.

MongoDB є дуже популярною СУБД серед розробників веб-додатків та мобільних додатків, оскільки вона надає потужні можливості для зберігання та керування даними, що забезпечує ефективну роботу додатків з великим обсягом даних.

**3.1.3 Реалізація фізичної моделі даних.** На основі аналізу логічної моделі даних та середовищ проектування БД було створено оперативну БД в MongoDB. Схема бази наведена на рисунку 3.2, її структура дозволяє зберігати всі необхідні дані.

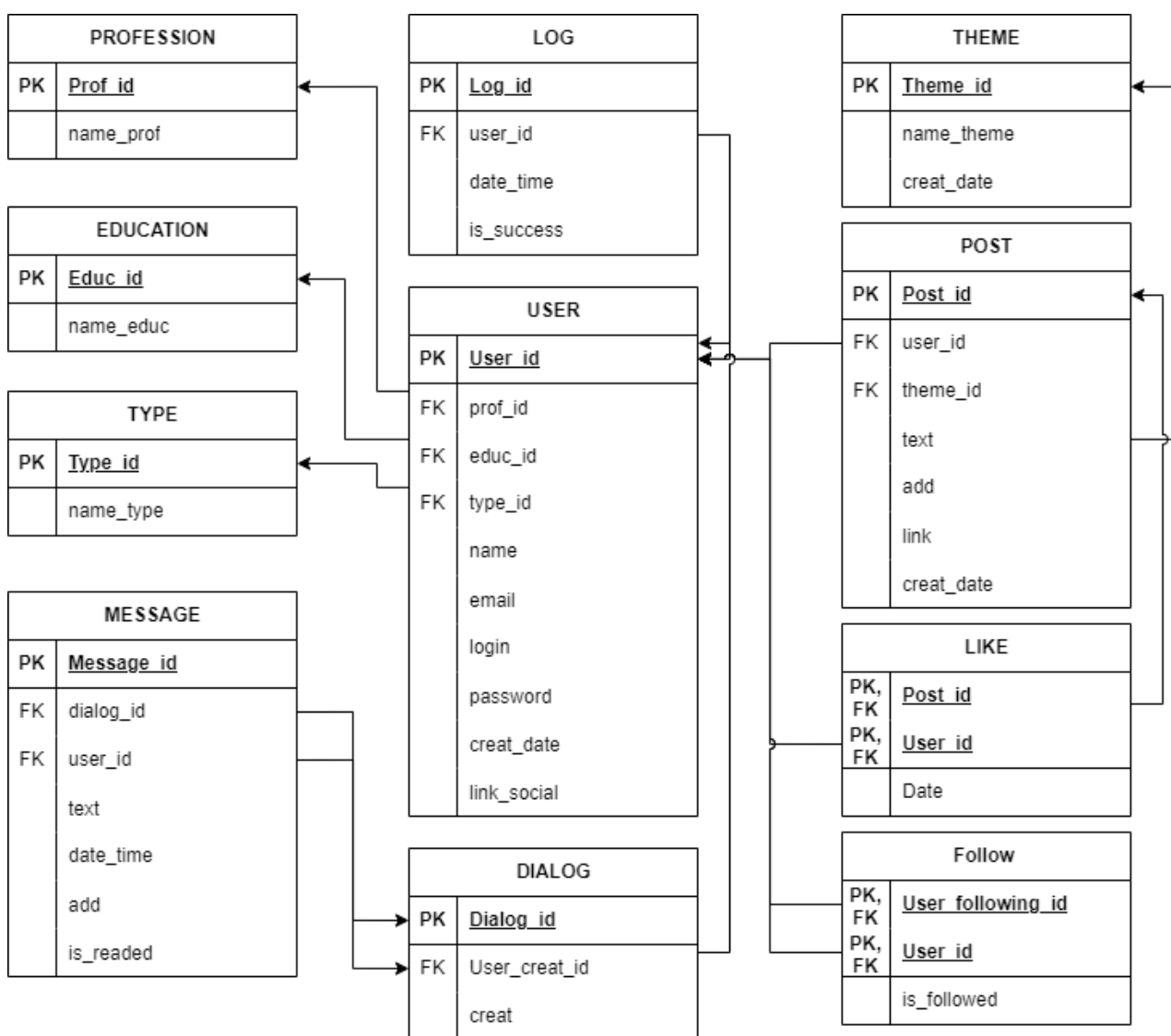


Рис. 3.2 Схема оперативної бази даних

Розглянемо її структуру більш детально. Вона складається з чотирьох, так званих, таблиць-довідників та семи оперативних таблиць:

- Profession – таблиця-довідник, що зберігає загальні найменування позицій, на яких працюють користувачі соціальної мережі.
- Education – таблиця-довідник, що містить загальні найменування типів освіти.
- Type – таблиця-довідник, що зберігає назву типу користувача соцмережі.
- Theme – таблиця-довідник, що зберігає назви технологій, про які написано в пості користувача. Особливість даної сутності полягає в наявності поля призначеного зберігати дату першозгадки. Це надає можливість відстежувати тим популярності технології в залежності від кількості згадок з початку її появи в соцмережі як об'єкта посту.
- User – сутність, що містить основні дані про користувача соцмережі. Вона має такі поля:
  - User\_id – поле виступає первинним ключем;
  - Prof\_id – виступає зовнішнім ключем, що пов'язаний з Profession, зберігаючи код позиції користувача;
  - Educ\_id – виступає зовнішнім ключем Education та містить код освіти користувача;
  - Type\_id – виступає зовнішнім ключем Type та містить код типу користувача, що визначає права доступу користувача;
  - Name – поле зберігає імена користувачів соцмережі;
  - Email – містить електронну адресу;
  - Login – зберігає логін, який використовується для авторизації;
  - Password – містить пароль доступу до аккаунта користувача;
  - Create\_date – містить дату створення аккаунта в соцмережі;
  - Link\_social – зберігає посилання на аккаунти користувача в інших соцмережах.
- Log – сутність призначена зберігати дані про сесії користувачів та має такі поля:
  - Log\_id – поле виступає первинним ключем;

- User\_id – поле виступає зовнішнім ключем User та зберігає дані про користувача, що почав сесію;
- Date\_time – поле, типу дата-час, містить дані про момент відкриття сесії;
- Is\_success – поле, що призначене зберігати відомості про успішність спроби авторизації та створення сесії.
- Dialog – сутність зберігає дані про створені діалоги користувачів та має такі поля:
  - Dialog\_id – поле виступає первинним ключем та зберігає унікальний код діалога користувачів;
  - User\_creat\_id – поле є зовнішнім ключем від User та зберігає код користувача, що створив даний діалог;
  - Creat – поле зберігає дату створення діалогу.
- Message – сутність містить дані про повідомлення, що надсилає користувач в соцмережі, та має такі поля:
  - Message\_id – поле виступає первинним ключем;
  - dialog\_id – поле виступає зовнішнім ключем Dialog та містить дані про діалог, в який надіслано повідомлення;
  - user\_id – поле виступає зовнішнім ключем User та зберігає код користувача, який надіслав повідомлення;
  - text – поле містить текст повідомлення;
  - date\_time – поле зберігає дату та час надісланого повідомлення;
  - add – поле зберігає файл, що прикріплений до повідомлення;
  - is\_readed – поле містить відомості про те, чи є повідомлення прочитаним.
- Post – сутність зберігає інформацію про контент, що створюють користувачі та має такі поля:
  - Post\_id – поле виступає первинним ключем;
  - User\_id – поле виступає зовнішнім ключем User та зберіє код користувача, що є автором посту;

- Theme\_id – поле виступає зовнішнім ключем Theme та містить тему контенту;
- Text – поле містить саме текстове наповнення посту;
- Add – зберігає доданий до посту файл;
- Link – поле містить посилання на джерела, що може додати автор до посту;
- Create\_date – поле містить дату та час публікації посту.
- Like – сутність, що зберігає дані про вподобаний пост користувачем та має такі поля:
  - Post\_id – поле виступає складеним ключем, адже виступає первинним ключем для даної сутності, а також є зовнішнім ключем з Post;
  - User\_id – поле виступає складеним ключем, що ідентифікує користувача з сутності User, що вподобав пост;
  - Date – поле зберігає дату та час вподобання.
- Follow – сутність зберігає інформацію про підписки користувачів та має такі поля:
  - User\_following\_id – поле представляє собою складений ключ, що ідентифікує код користувача з сутності User, котрий стежить за іншим користувачем;
  - User\_id – поле представляє собою складений ключ, що ідентифікує код користувача з сутності User, за яким стежить інший користувач;
  - Is\_followed – поле зберігає інформацію про те, чи підписаний користувач.

**3.1.4 Проектування сховища даних.** Сховище даних [14] – спеціалізована база даних, що має такі особливості:

- предметність – сховище містить інформацію, що стосується конкретного об'єкту бізнес процесу;
- узагальнення – сховище отримує дані з різних джерел інформації;



- чистота – вхідні дані проходять фільтрацію, що дозволяє прибрати дублі, доповнити чи узагальнити записи;
- незмінність – наповнення сховища даних відбувається досить рідко, після накопичення вагомого об’єму даних та їх обробки;
- залежність від часу – визначною характеристикою сховища даних є часовий вимір, що дозволяє відслідковувати динаміку розвитку об’єкта а також прогнозувати її.

Технологічно сховища даних тісно пов'язані із засобами оперативної аналітичної обробки даних (OLAP-технологіями), що дозволяють аналітикам, керівникам і керівникам вищої ланки вивчати великі обсяги взаємопов'язаних даних за допомогою швидкого інтерактивного відображення інформації на різних рівнях деталізації.

На рисунку 3.4 представлено спроектоване сховище даних для соціальної мережі.

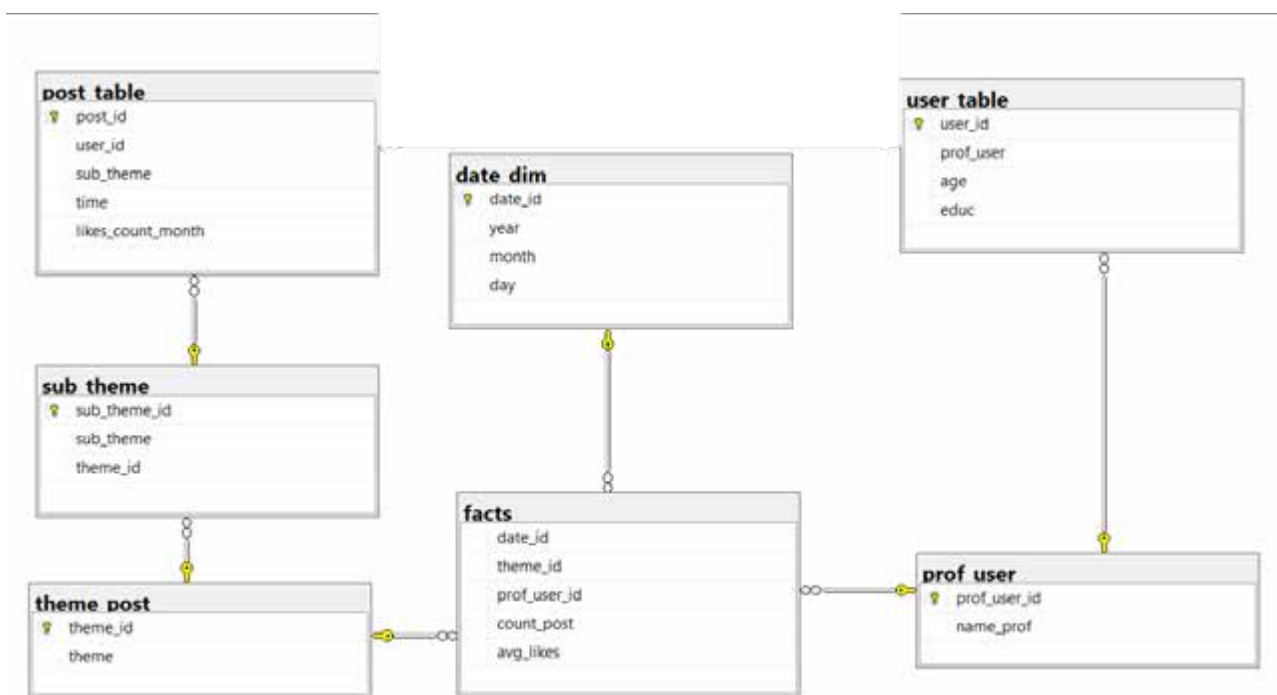


Рис. 3.3 Сховище даних

Загальна структура сховища даних складається з таблиць-вимірів та таблиці-фактів. Спроектоване сховище даних має шість вимірів:

- Theme\_post – таблиця-вимір, що зберігає теми публікацій користувачів соцмережі.
- Sub\_theme – таблиця-вимір містить назви технологій, що стосуються публікації.
- Post\_table – таблиця-вимір зберігає властивості публікації: автор, час, кількість вподобань.
- Date\_dim – таблиця-вимір, що містить дати проведення аналізу.
- User\_table – таблиця-вимір надає дані про користувачів, що публікували контент.
- Prof\_user – таблиця-вимір зберігає назви професій користувачів.

Сховище даних має одну таблицю-факт – Facts, що призначена аналізувати дані в розрізі вимірів.

## 3.2 Програмне забезпечення системи

**3.2.1 Архітектура програмного забезпечення.** На основі наведеного вище аналізу предметної області та спроектованими БД та СД було обрано звичну для веб-сайтів архітектуру системи – клієнт-серверну. Дана архітектура [15] має певні переваги:

- Простота – не потребує завантаження потужних додатків;
- Доступність – в більшості випадків потрібен лише доступ до інтернету, робоча станція та браузер;
- Легкість – певна частина виконання операцій відбувається саме на серверній частині, а отже не вимагає особливих ресурсів.

На рисунку 3.4 представлена топологія соцмережі з її основними вузлами:

- робоча станція – девайс з доступом до Інтернет мережі (ПК користувача);
- вузли з даними – сервер, що містить потрібні компоненти для роботи системи.

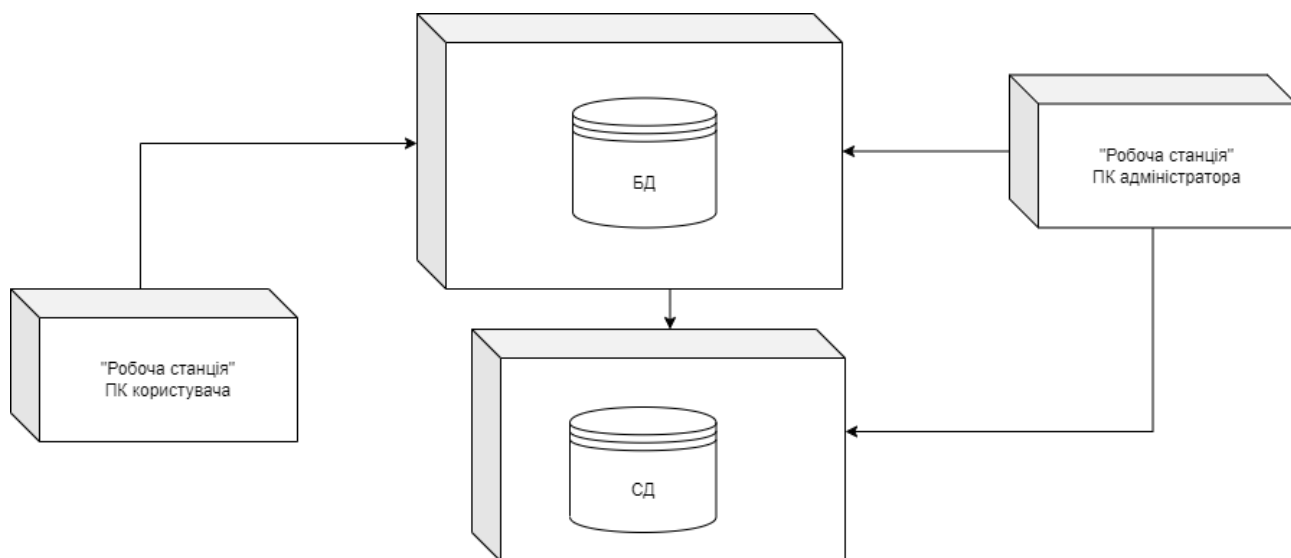


Рис. 3.4 Топологія системи

**3.2.2 Обрані інструменти та програмні засоби для реалізації програмного забезпечення системи.** Враховуючі всі поставлені вимоги та визначену архітектуру системи було обрано такі інструменти розробки:

- React.js [16] – простий фреймворк для створення та розгортання веб-додатків.
- Redux [17] – бібліотека, що використовується React для керування станом системи.
- SASS (Syntactically Awesome Style Sheets) [18] – інструмент, що дозволяє спростити роботу з CSS.

*3.2.2.1 React.js.* React JS [19] (або просто React) – це відкрита JavaScript бібліотека, що має багато варіантів застосування, в тому числі для створення інтерфейсів веб-додатків. Фреймворк був створений компанією Facebook ще у 2013, але загальнодоступним став лише в 2015 році. Він реалізований на декларативному підході розробки інтерфейсу – розробник зосереджений на статичному інтерфейсі, а не на його зміні. Дана концепція розробки робить React простим та ефективним інструментом. React також використовує віртуальний DOM (Document Object Model), що дозволяє оптимізувати швидкість рендерингу компонентів та реагувати на зміни стану додатку.

Однією з головних концепцій React є компоненти – незалежні модулі системи, що надають змогу вносити зміни окремо, не зачіпаючи всю систему цілком. Такий підхід має значну перевагу – зменшення необхідної пам'яті та функціональних одиниць зарахунок повторного використання одного і того ж компонента.

Загалом, React – це потужний інструмент, який дозволяє створювати складні та динамічні інтерфейси веб-додатків швидко та ефективно. Він підходить як для початківців у розробці веб-додатків, так і для досвідчених розробників, що шукають інструмент з високою продуктивністю та гнучкістю.

**3.2.2.2 Redux.** Redux [20] є відкритою бібліотекою для керування станом додатків у JavaScript, що дозволяє зберігати та оновлювати стан додатків в одному місці. Redux побудований на базі парадигми Flux, яка є архітектурним підходом до створення веб-додатків, що дозволяє керувати станом додатку та забезпечує однорідний потік даних.

Одним з головних принципів Redux є зберігання всього стану додатку в одному "Store" (магазині), який є незмінним та доступним з будь-якого місця додатку. Зміни стану відбуваються за допомогою дій (actions), які ініціюють зміни даних та розсилають повідомлення про зміни по всьому додатку..

**3.2.2.3 SASS.** SASS (Syntactically Awesome Style Sheets) [21] – це мова передпроцесора, яка розширює функціональність звичайних CSS-стилів. SASS дозволяє використовувати змінні, вкладені селектори, побудову міксінів та багато інших корисних функцій, що значно спрощують написання CSS-стилів.

Передпроцесор SASS дозволяє розбити стилі на менші частини та використовувати їх багаторазово. Наприклад, ви можете створити міксіні для визначення затінення та використовувати його в багатьох місцях, замість того, щоб копіювати той самий CSS-код в кожному селекторі.

Однією з найбільш корисних функцій SASS є змінні. Змінні дозволяють вам зберігати значення, такі як кольори та розміри, у змінній та використовувати їх багаторазово в CSS-стилях. Якщо ви хочете змінити значення змінної, вам

потрібно буде змінити лише її значення в одному місці, а не у кожному селекторі, де вона використовується.

Іншою корисною функцією SASS є вкладені селектори. Замість того, щоб використовувати багато класів та ID-елементів, ви можете вкладати селектори один в одного, що робить ваш CSS-код більш зрозумілим та організованим.

Узагальнюючи, SASS дозволяє розширити можливості CSS та зробити його більш зрозумілим та організованим за допомогою змінних, вкладених селекторів, міксинів та багатьох інших корисних функцій.

**3.2.3 Технології OLAP та Data mining.** OLAP [22]– технологія, що надає можливість проводити аналіз даних з різних точок зору. Якщо використовувати оперативну базу даних для проведення досліджень то для цього потрібні будуть значно більші ресурси, ніж якщо це виконувати на сховищах даних. До переваг OLAP можна віднести:

- універсальність – технологія може бути застосована до різних типів об'єктів;
- узгодженість – технологія застосовується на сховищах даних, що же містить очищені та несуперечливі дані;
- швидкість – швидкий аналіз даних у досить різноманітних розрізах;
- ефективність – технологія дозволяє використовувати отримані результати для бізнес-моделювання, інтелектуального аналізу чи проведені звітності;\
- зрозумілість – технологія дозволяє швидко візуалізувати отримані результати для легкого читання.

Data mining [22] – технологія покликана за допомогою інтелектуального аналізу знаходити приховані залежності у вигляді значущих особливостей, кореляцій, тенденцій та шаблонів. Концепція даної технології полягає в тому, що вона висуває власні гіпотези залежності даних. Такий підхід має значні переваги – виявлення невідомих закономірностей, що несуть практичну значимість, а отже можливість збільшення комерційних вигод, шляхом впливу на ці залежності.

Основна особливість Data Mining – це поєднання широкого математичного інструментарію (від класичного статистичного аналізу до нових кібернетичних методів) і останніх досягнень у сфері інформаційних технологій. В технології Data Mining гармонійно об'єдналися строго формалізовані методи і методи неформального аналізу, тобто кількісний і якісний аналіз даних.

До методів і алгоритмів Data Mining відносяться наступні: штучні нейронні мережі, дерева рішень, символні правила, методи найближчого сусіда і k-найближчого сусіда, метод опорних векторів, байесові мережі, лінійна регресія, кореляційно-регресійний аналіз; ієрархічні методи кластерного аналізу, неієрархічні методи кластерного аналізу, у тому числі алгоритми k-середніх і k-медіани; методи пошуку асоціативних правил, у тому числі алгоритм Apriori; метод обмеженого перебору, еволюційне програмування і генетичні алгоритми, різноманітні методи візуалізації даних і безліч інших методів.

**3.2.4 Розгортання OLAP-куба.** Куб-OLAP [23]– особлива структура даних, що може бути заснована на сховищі даних, та призначена для роботи з великими об'ємами даних швидко та просто. Його створили за допомогою служби SSAS.

SQL Server Analysis Services (SSAS) [24]– служба, інтегрована в середовище Visual Studio, що використовується для роботи з технологіями OLAP та Data mining. Її функціонал призначений для роботи з кубами OLAP, що створюються на основі сховищ даних.

Робота над розгортанням куба OLAP за допомогою SSAS починається з підключення до проекту сховища даних. Даний процес, представлений на рисунку 3.5, реалізується за допомогою модуля Data Source Wizard.

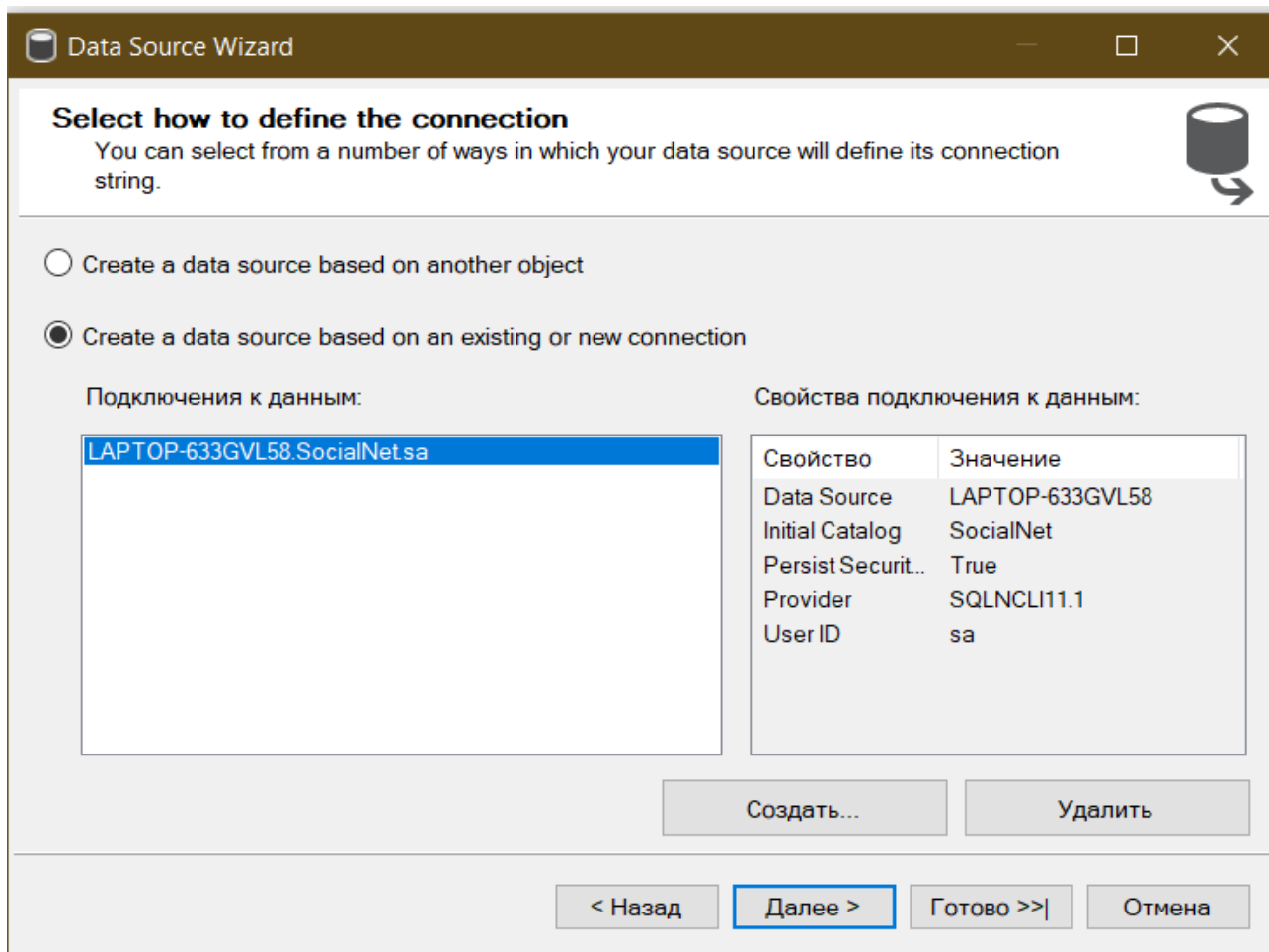


Рис. 3.5 Підключення сховища SocialNet

Далі в проєкті потрібно створити уявлення джерела даних, адже підключення до джерела даних забезпечує просте з'єднання з реляційною базою даних, а уявлення джерела даних потрібне для додавання зв'язків, створення розрахунків та встановлення логічних ключів. Загалом, уявлення джерела демонструє як сховище даних в SSAS буде співставлене з реляційною схемою та надає шар абстрагування поверх реляційної бази даних.

Наступним етапом в розгортання куба OLAP є створення його вимірів [25] – наборів впорядкованих ієрархічних рівнів в кубі, що використовується як основа для аналізу. Загалом куб має три основних виміри, що представлені на рисунках 3.6-3.7.

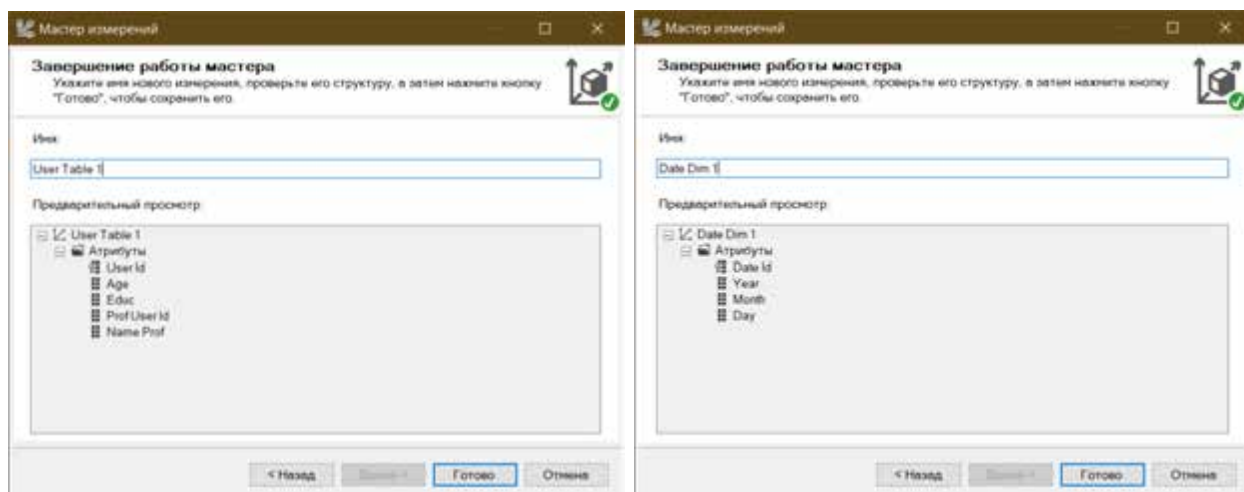


Рис. 3.6 Вимір Користувач та Дата

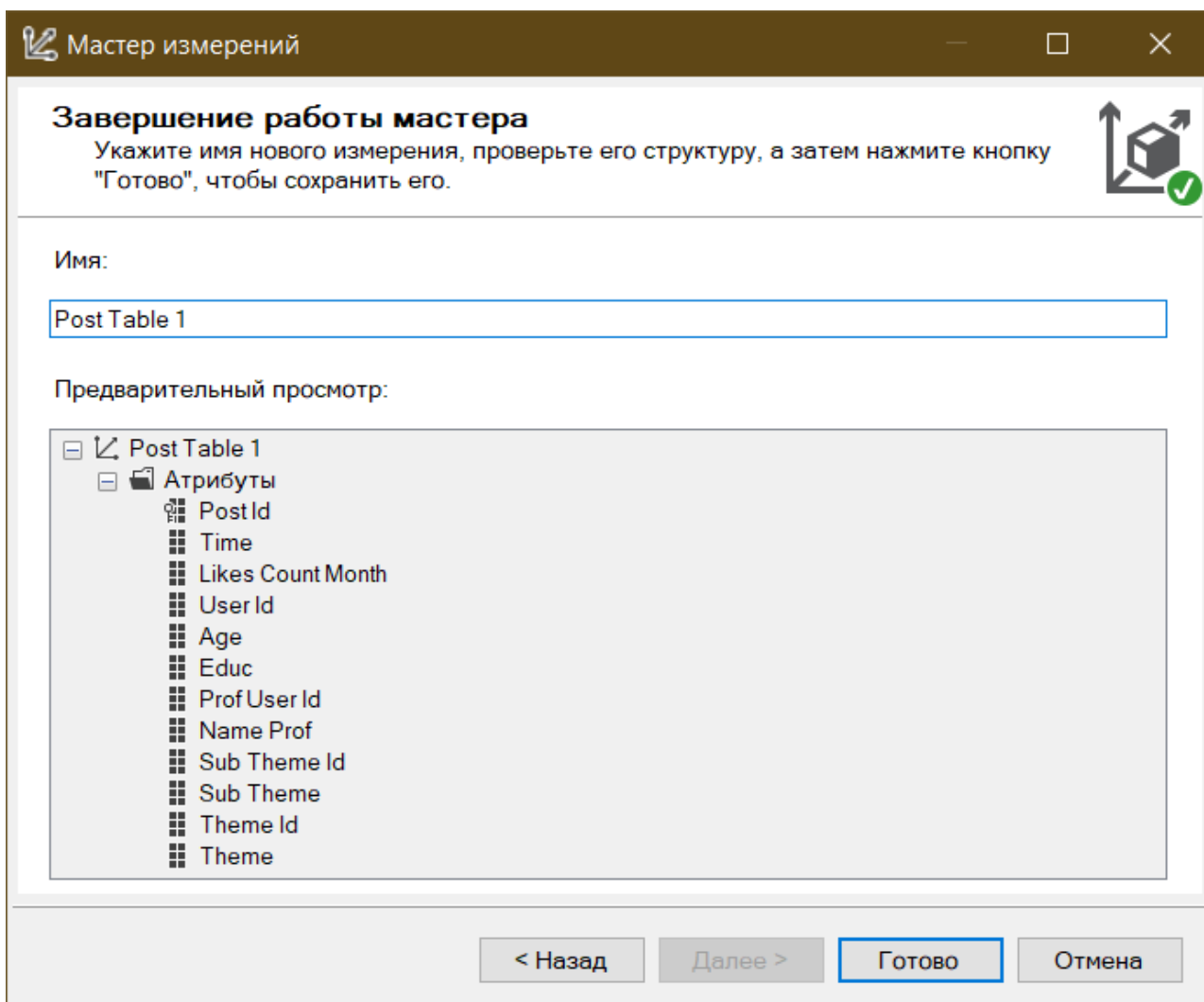


Рис. 3.7 Вимір Пост, що пов'язаний з іншими вимірами



Далі починається досить важливий процес – створення структури OLAP куба, основного елементу аналізу. Даний процес реалізується за допомогою модуля Cube Wizard, що представлено на рисунку 3.8. Зараз дуже важливо правильно обрати групу мір для куба та визначити його виміри. Міра – набір значень, що оснований на стовпці в таблиці фактів куба, які зазвичай є числовими значеннями. Міри – це центральні значення куба, попередньо оброблені, агрегатні й проаналізовані.

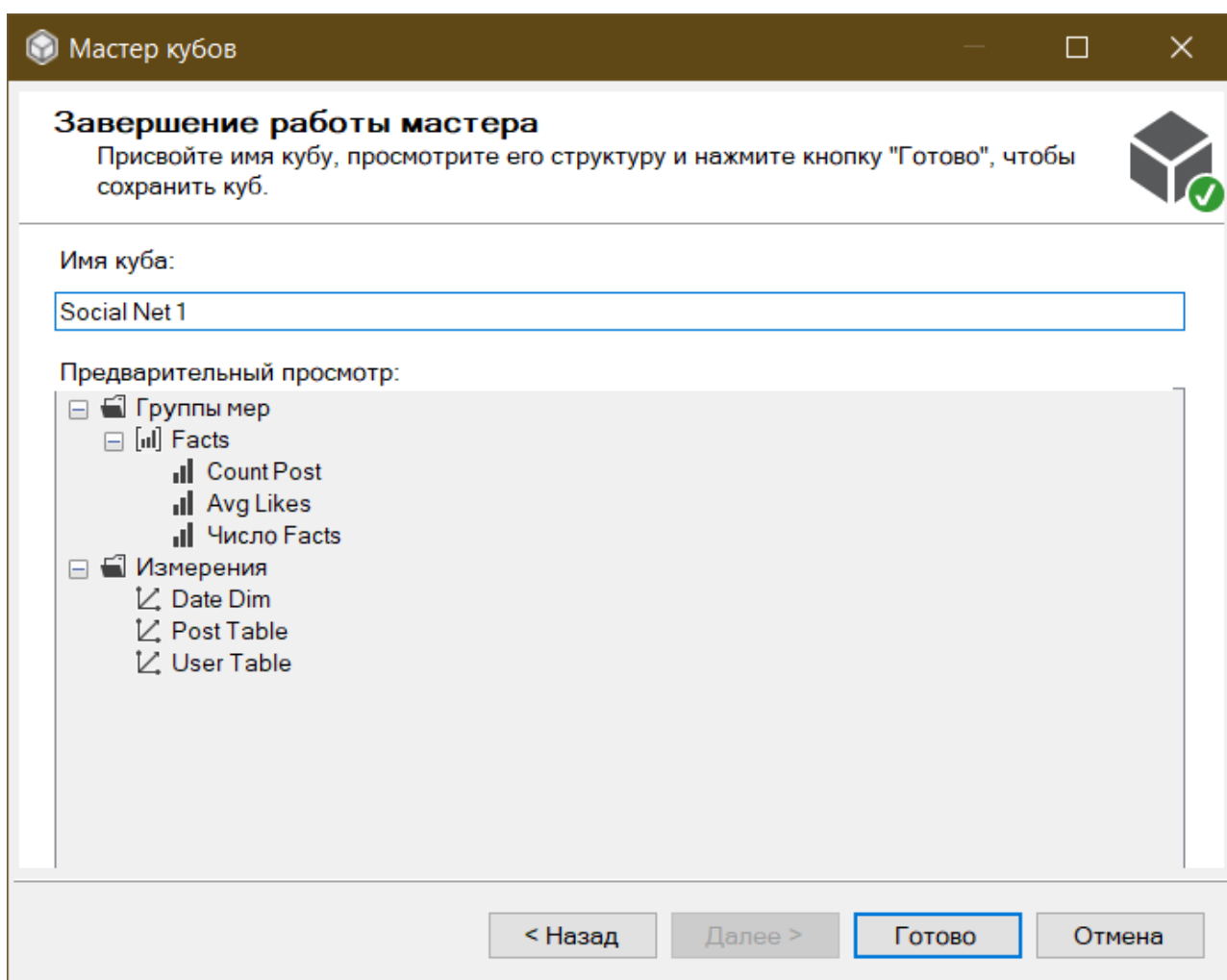


Рис. 3.8 Структура куба

В результаті виконання всіх операцій було створено куб Social\_Net що зображений на рисунку 3.9.

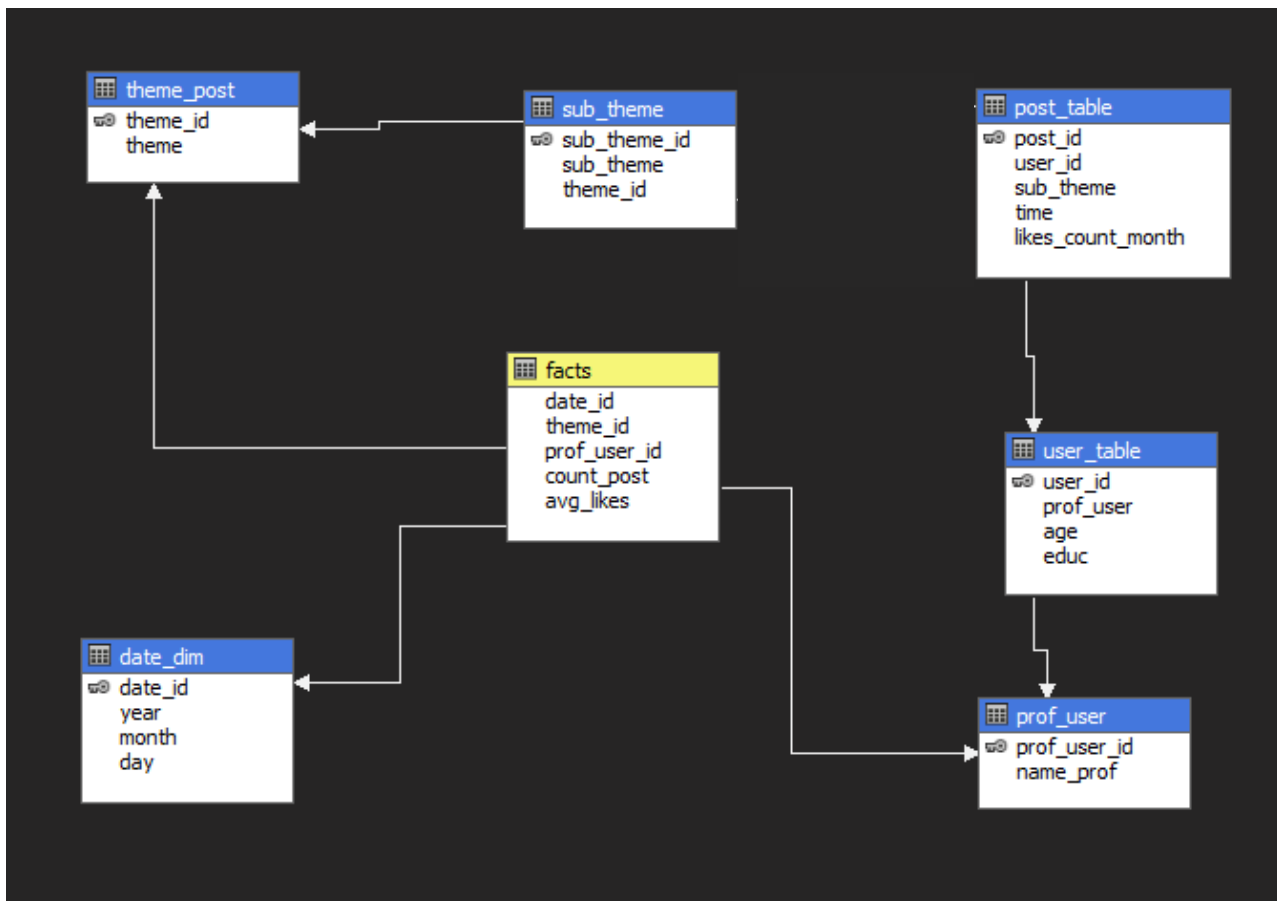


Рис. 3.9 Куб Social\_Net

### 3.2.5 Розробка аналітичного модуля системи.

**3.2.5.1 Decision Trees Algorithm.** Алгоритм Microsoft Decision Trees [26] — це гібридний алгоритм, який включає різні методи для створення дерева та підтримує кілька аналітичних завдань, зокрема регресію, класифікацію та асоціацію. Алгоритм Microsoft Decision Trees підтримує моделювання як дискретних, так і безперервних атрибутів. Алгоритм заснований на байєсовській підході вивчення моделі взаємодії шляхом отримання наближених апостеріорних розподілів для моделей.

На основі розробленого раніше Social\_Net куба потрібно розгорнути нову структуру – зріз куба, що містить необхідні дані для аналізу. Для цього необхідно обрати виміри та міри куба, а також ключові параметри для цієї структури. Цей процес представлено на рисунку 3.10.

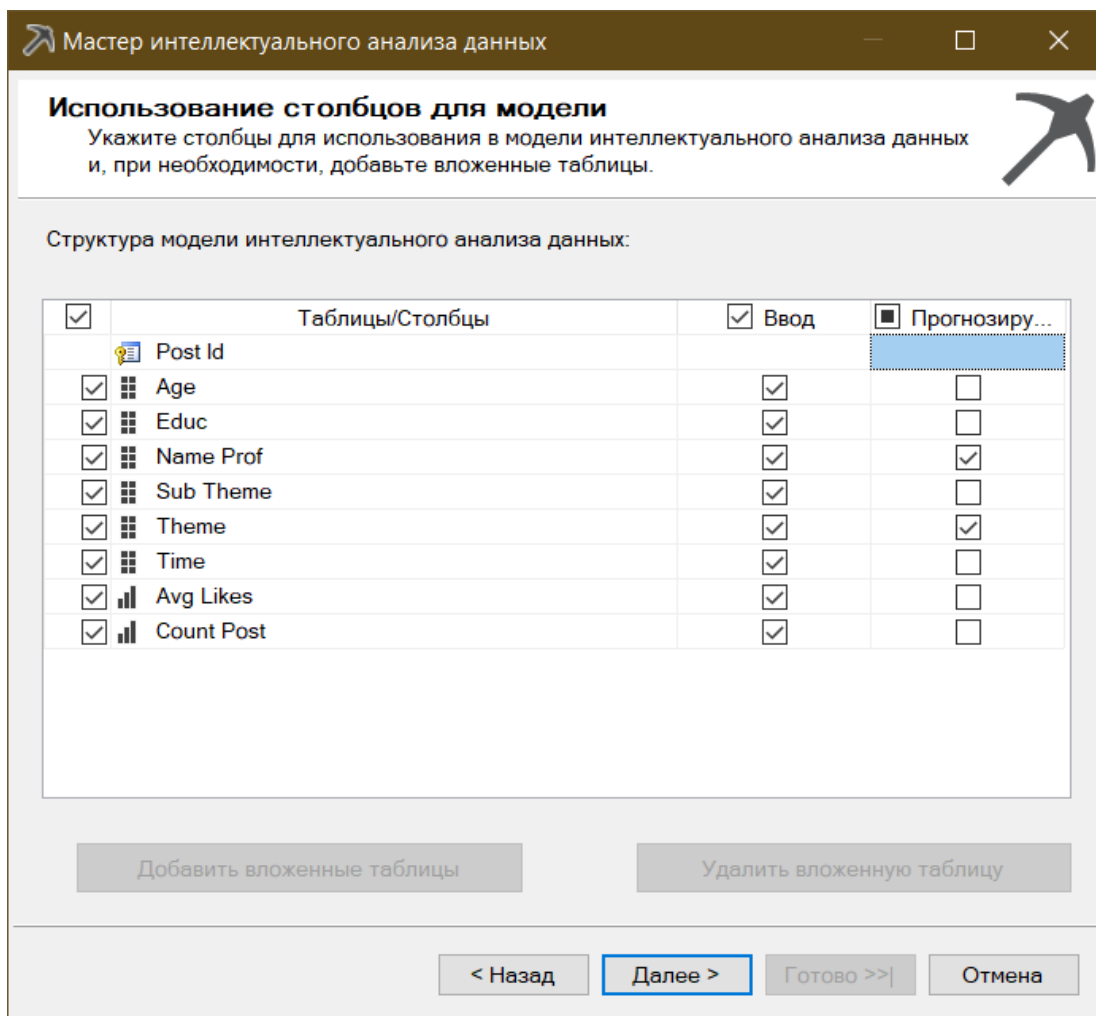


Рис. 3.10 Вибір прогнозованих полів для структури DecTree

Для дослідження були обрані поля з вимірів Post та User, а також міри. В якості прогнозованих значень було обрано кілька пунктів – Name Prof(назва професії) та Theme (тема).

**3.2.5.2 Naive Bayes Algorithm.** Алгоритм Microsoft Naive Bayes [27] — це набір алгоритмів класифікації на основі теореми Байєса . Це не один алгоритм, а сімейство алгоритмів, де всі вони мають загальний принцип, тобто кожна пара ознак, що класифікуються, не залежить одна від одної. Слово naive в назві Naive Bayes походить від того факту, що алгоритм використовує байєсівські методи, але не враховує залежностей, які можуть існувати.

Цей алгоритм менш інтенсивний у обчисленні, ніж інші алгоритми Microsoft, і тому корисний для швидкого генерування моделей видобутку для виявлення зв'язків між вхідними стовпцями та передбачуваними стовпцями.

Структура даних для алгоритм Naive Bayes має таку ж схему, що і для алгоритма Decision Trees (рис. 3.10).

**3.2.5.3 Association Algorithm.** Алгоритм асоціативних правил Microsoft [28] — це метод аналізу даних, який використовується для виявлення цікавих взаємозв'язків або асоціацій між різними елементами в наборі даних. Основна мета алгоритму асоціативних правил - знайти і вивести правила асоціацій між різними елементами даних.

Основними кроками алгоритму асоціативних правил є:

- Підготовка даних: Спершу необхідно підготувати дані, наприклад, створити таблицю, де рядки представляють окремі транзакції, а стовпці - різні товари або елементи, які можуть бути придбані або використані в цих транзакціях. Кожен елемент в таблиці може бути позначений як присутній (1) або відсутній (0) в конкретній транзакції.
- Побудова множин асоціативних правил: Алгоритм асоціативних правил шукає комбінації товарів, які часто купують разом. Для цього він обчислює метрики, такі як підтримка, достовірність і підтримка. Основні терміни, пов'язані з цими метриками:
  - Підтримка (Support): Це частота, з якою конкретний набір товарів (або правило) зустрічається в наборі даних. Висока підтримка вказує на популярність асоціації.
  - Достовірність (Confidence): Ця метрика вказує на ймовірність того, що, коли клієнт купує один набір товарів, він також купить інший товар (або товари) з асоційованим правилом.
- Відбір правил із заданими параметрами: Після обчислення метрик підтримки і достовірності можна встановити певні порогові значення для цих метрик. Правила, які задовольняють ці критерії, вважаються цікавими і корисними для подальшого використання.
- Інтерпретація і застосування: Знайдені асоціативні правила можна інтерпретувати і використовувати в різних сферах, наприклад, для рекомендацій, крос-продажів, організації товарів у магазинах і багатьох

інших застосуваннях.

Деякі з відомих алгоритмів асоціативних правил включають в себе "Apriori" і "FP-growth". Вони використовуються для ефективного пошуку асоціативних правил у великих наборах даних.

Алгоритм асоціативних правил є корисним інструментом для виявлення прихованих зв'язків між елементами даних і може бути використаний в різних областях для покращення прийняття рішень і надання рекомендацій.

Особливість в структурі інтелектуального аналізу призначеного для алгоритму асоціативних правил полягає в типі оброблюваних даних – вони мають бути дискретні, як представлено на рисунку 3.11.

Определение содержимого и типа данных столбцов  
Укажите содержимое и тип данных столбцов структуры интеллектуального анализа

Структура модели интеллектуального анализа данных:

Столбцы	Тип содержимого	Тип данных
Post Id	Key	Long
Age	Discretized	Long
Educ	Discrete	Text
Likes Count Month	Discretized	Long
Name Prof	Discrete	Text
Sub Theme	Discrete	Text
Theme	Discrete	Text
Time	Discretized	Date

Найти дискретные и непрерывные значения для числовых столбцов:

< Назад **Далее >** Готово >>| Отмена

Рис. 3.11 Типы даних для алгоритму асоціативних правил структури 1Rule

**3.2.5.4 Clustering Algorithm.** Алгоритм кластеризації Microsoft [29] — це алгоритм сегментації або кластеризації, який перебирає випадки в наборі даних, щоб згрупувати їх у кластери, які містять подібні характеристики. Ці угруповання корисні для вивчення даних, виявлення аномалій у даних і створення прогнозів.

Алгоритм кластеризації визначає зв'язки в наборі даних, які інколи не можливо логічно отримати шляхом випадкового спостереження.

Для проведення кластерного аналізу скористаємось засобами SSAS. За допомогою майстра інтелектуального аналізу, на основі створеного кубу, створюється структура, що застосовує алгоритм кластерного аналізу. Вона представлена на рисунку 3.12.

Мастер интеллектуального анализа данных

**Определение содержимого и типа данных столбцов**  
Укажите содержимое и тип данных столбцов структуры интеллектуального анализа

Структура модели интеллектуального анализа данных:

Столбцы	Тип содержимого	Тип данных
Post Id	Key	Long
Age	Continuous	Long
Educ	Discrete	Text
Likes Count Month	Continuous	Long
Name Prof	Discrete	Text
Sub Theme	Discrete	Text
Theme	Discrete	Text
Time	Continuous	Date
Avg Likes	Continuous	Long
Count Post	Continuous	Long

Найти дискретные и непрерывные значения для числовых столбцов:

< Назад **Далее >** Готово >>| Отмена

Рис. 3.12 Структура моделі аналізу Cluster

Для проведеного аналізу було використано Expectation Maximization (EM). У кластеризації EM алгоритм ітераційно уточнює початкову модель кластера, щоб відповідати даним, і визначає ймовірність існування точки даних у кластері. Алгоритм завершує процес, коли ймовірнісна модель відповідає даним. Функція, що використовується для визначення відповідності, — це логарифмічна вірогідність даних, заданих для моделі.

Результати методу кластеризації EM є імовірнісними. Це означає, що кожна точка даних належить до всіх кластерів, але кожне призначення точки даних кластеру має різну ймовірність. Оскільки метод дозволяє перекривати кластери, сума елементів у всіх кластерах може перевищувати загальну кількість елементів у навчальному наборі. У результатах моделі майнінгу оцінки, які вказують на підтримку, коригуються з урахуванням цього.

## 4 РЕЗУЛЬТАТИ

### 4.1 Інтерфейс соцмережі

Користувач завжди починає роботу з системою через інтерфейс, призначений спростити та забезпечити зв'язок з сервером. Важливо підтримувати його в актуальному стані, зберігаючи звичні поняття, але надаючи їм додатковий функціонал. Розглянемо структуру та функціонал розробленої соціальної мережі більш детально.

Перше, що зустрічає користувач, при підключенні до сайту – сторінка Авторизації, з можливістю перейти на сторінку Реєстрації, що представлено на рисунку 4.1. Авторизація працює за алгоритмом JWT Token Authorization. JWT Token Authorization – це метод авторизації на веб-сайтах або додатках, який використовує JSON Web Token (JWT) для передачі та перевірки інформації про авторизованого користувача. JWT – це стандарт відкритого обміну токенів, який визначає компактний та самодостатній спосіб передачі інформації між сторонами у вигляді JSON об'єктів.

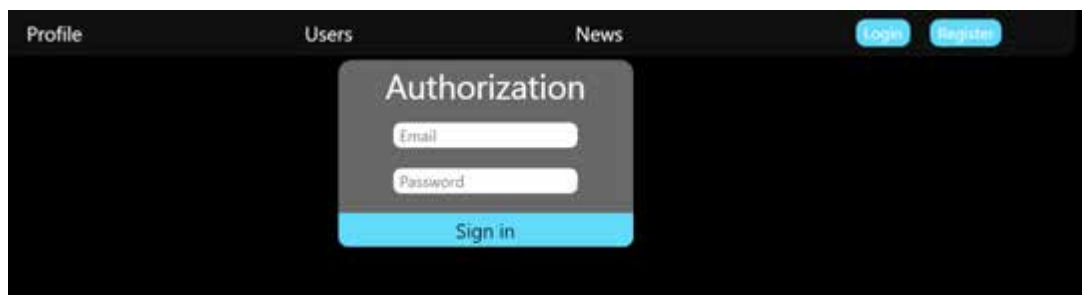


Рис. 4.1 авторизація користувача

При використанні JWT Token Authorization, після успішної аутентифікації користувача (зазвичай за допомогою логіна та пароля), сервер генерує JWT токен та надсилає його користувачеві. Клієнт зберігає цей токен та включає його у кожен запит, що робить до сервера. Сервер перевіряє токен та, якщо він дійсний, дає користувачеві доступ до запитуваних ресурсів.



JWT токен складається з трьох частин: заголовку, корисної навантаження та підпису. Заголовок містить тип токenu та алгоритм шифрування, який використовується для генерації підпису. Корисна навантаження містить інформацію про користувача та додаткові дані, які визначає розробник додатку. Підпис генерується на основі заголовку, корисної навантаження та секретного ключа, який зберігається на сервері.

JWT Token Authorization є безпечним методом авторизації, оскільки токен зберігається на клієнті та передається зашифрованим, що унеможлиблює його підробку або зміну. Використання JWT дозволяє зменшити навантаження на сервер, оскільки не потрібно зберігати стан сесії на сервері, а також дозволяє збільшити масштабованість додатку.

Після авторизації відбувається перехід на сторінку пошуку користувачів, що представлено на рисунку 4.2. Вона демонструє:

- меню сторінок;
- коротку інформацію профіля авторизованого користувача;
- коротку інформацію про профілі інших користувачів з можливістю відразу підписатись на них.

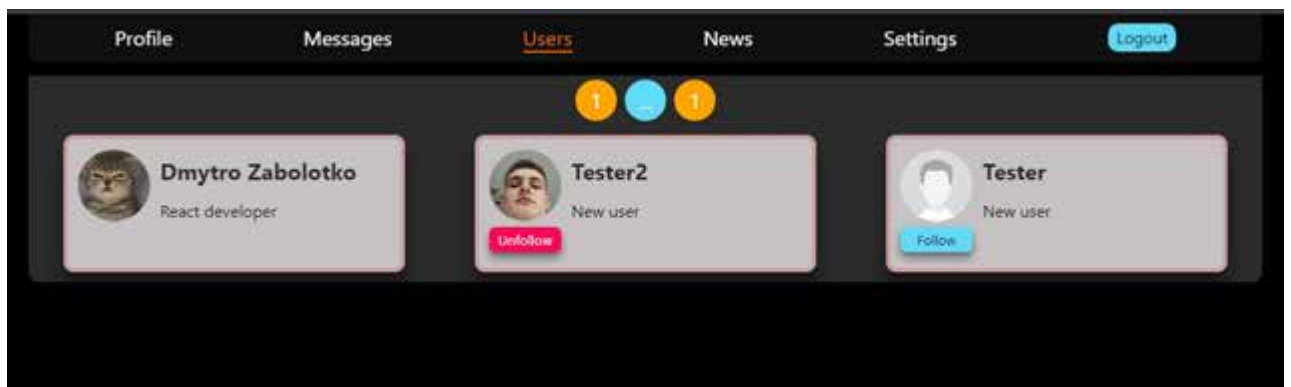


Рис. 4.2 сторінка пошуку користувачів

Щоб ознайомитись з профілем авторизованого користувача потрібно перейти на сторінку Профіль за допомогою меню. Загальний вигляд сторінки представлено на рисунку 4.3.

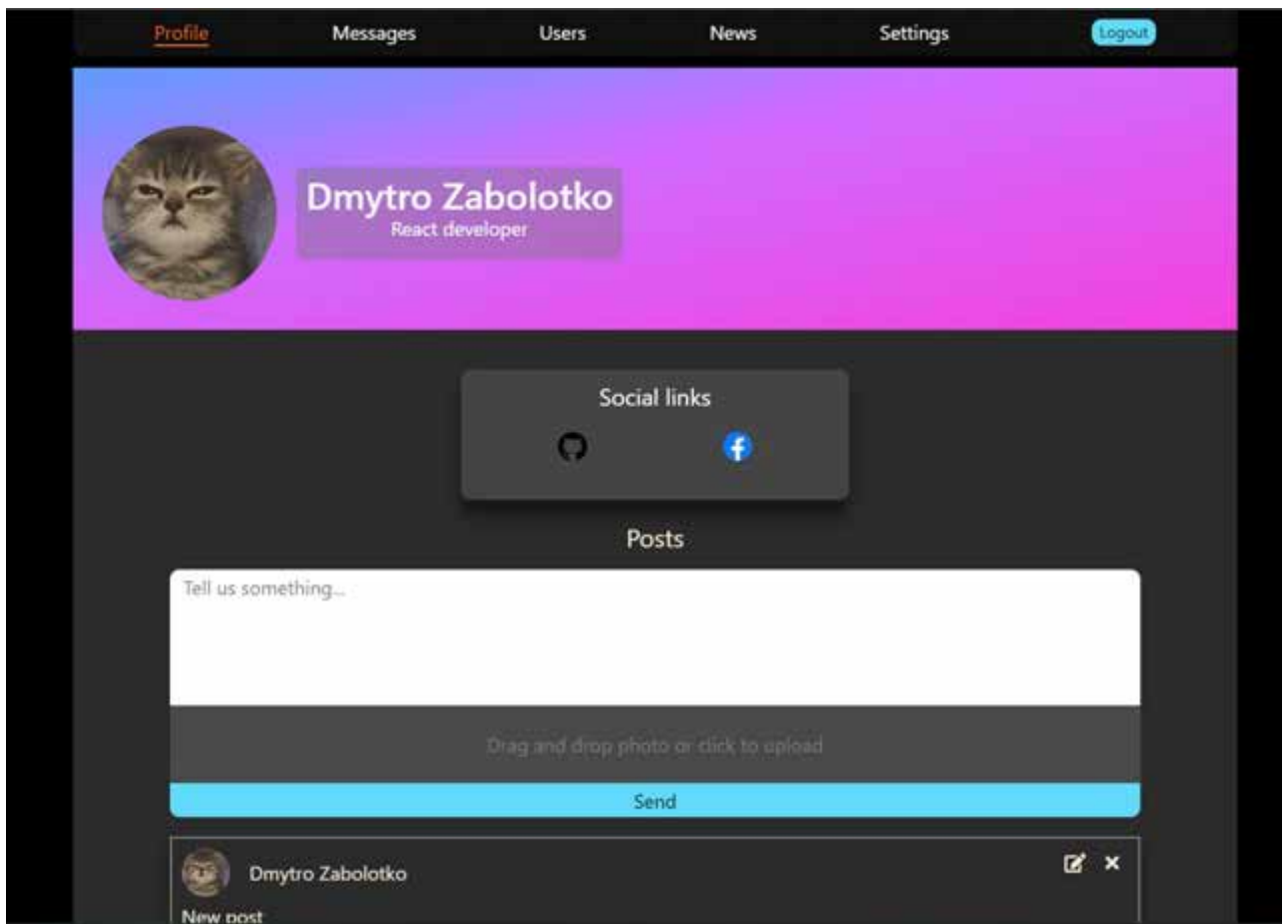


Рис. 4.3 Сторінка Профіль

Сторінка Профіль містить інформацію про авторизованого користувача:

- фотографія користувача;
- ім'я користувача та його посада;
- посилання на інші соціальні мережі спеціаліста;
- область для створення нової публікації;
- опубліковані пости.

Приклад вигляду опублікованого посту представлено на рисунку 4.4.

Структура допису досить звична:

- ім'я та фотографія автора допису;
- наповнення публікації – основний текст допису;
- фотографія, прикріплена до публікації;
- кількість вподобань;
- дата та час публікації.

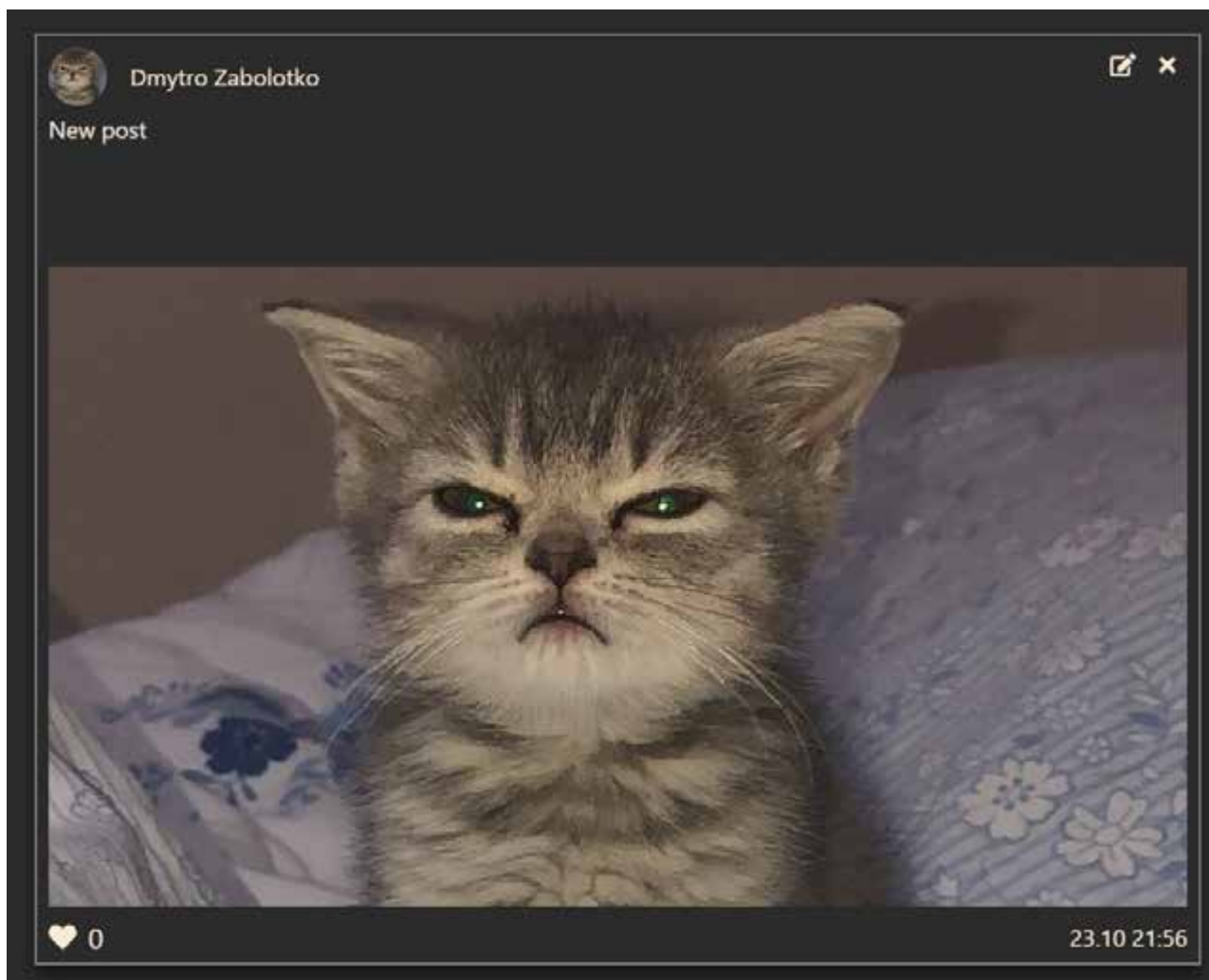


Рис. 4.4 Допис користувача

На сайті реалізовано функціонал для комунікації між користувачами у звичному форматі – діалог. Щоб скористатись цим, потрібно перейти на сторінку Повідомлення, представлену на рисунку 4.5. Зліва знаходиться область в яку виводиться список діалогів, що були створенні користувачем. Основна частина сторінки – область обраного діалогу. Вона містить самі повідомлення, від останнього надісланого, а також текстове поле для введення наступного повідомлення.

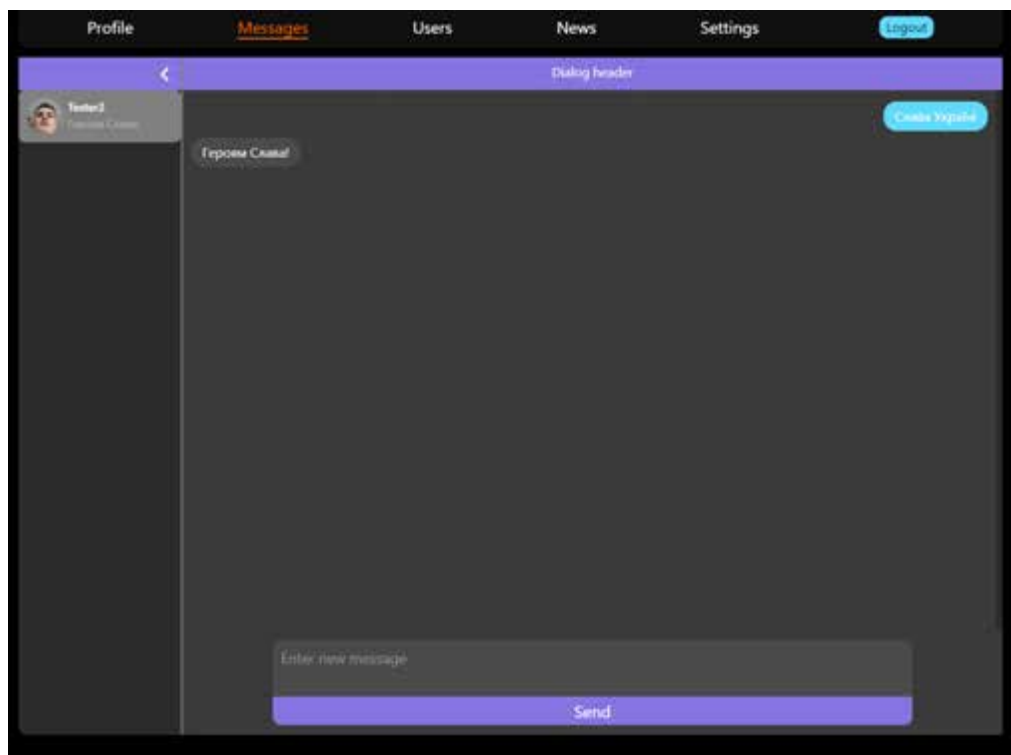


Рис. 4.5 Переписка користувачів

Ще однією сторінкою, доступною на панелі меню – Налаштування, представлена на рисунку 4.6. Вона надає можливість доповнювати інформацію про профіль авторизованого користувача чи вносити зміни.

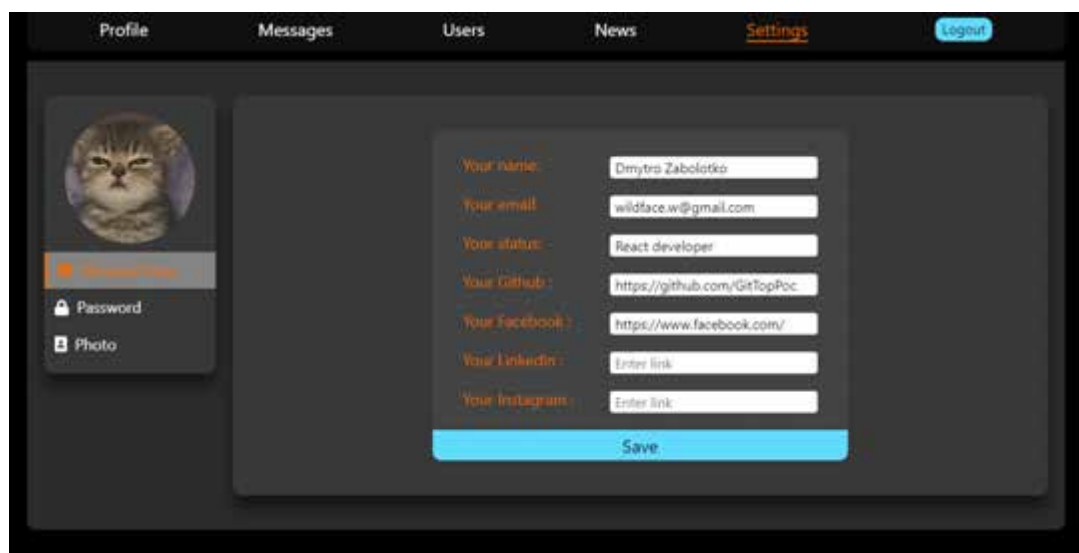


Рис. 4.6 таблиця з заявками на дзвінок

## 4.2 Результати аналітичного модуля

**4.2.1 Decision Trees Algorithm.** Результатом розгортання структури DecTree є візуалізоване дерево рішень, що представлено на рисунку 4.7-4.8.

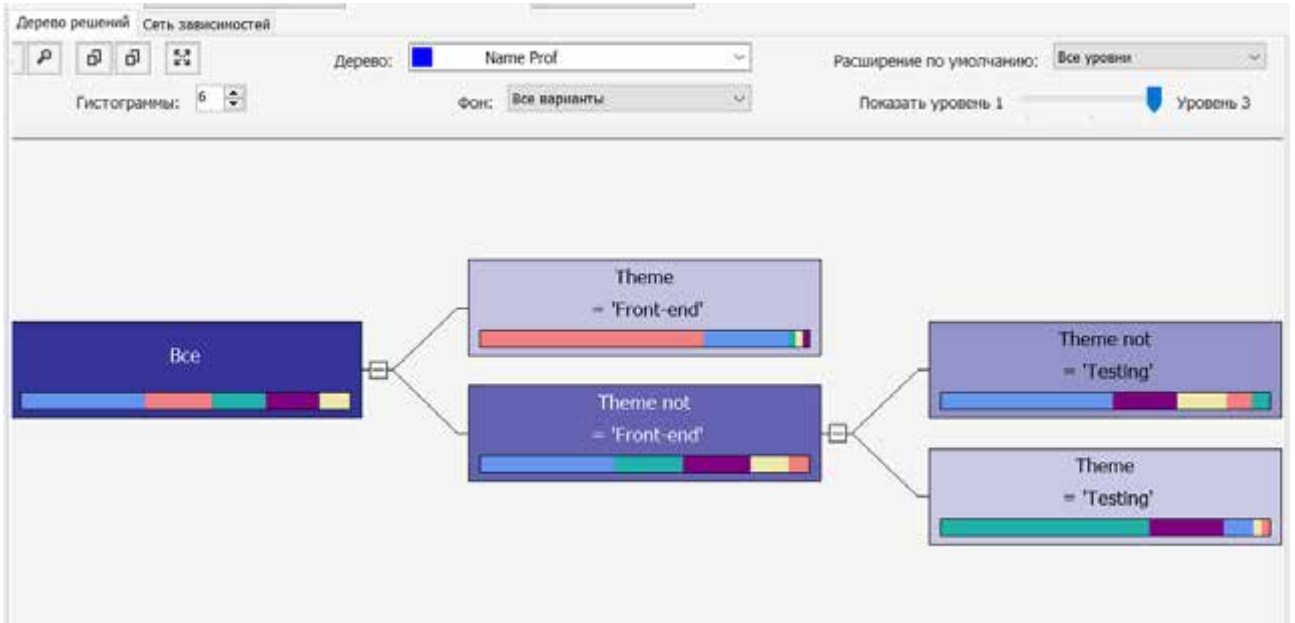


Рис. 4.7 Дерево рішення для прогнозування Name\_Prof

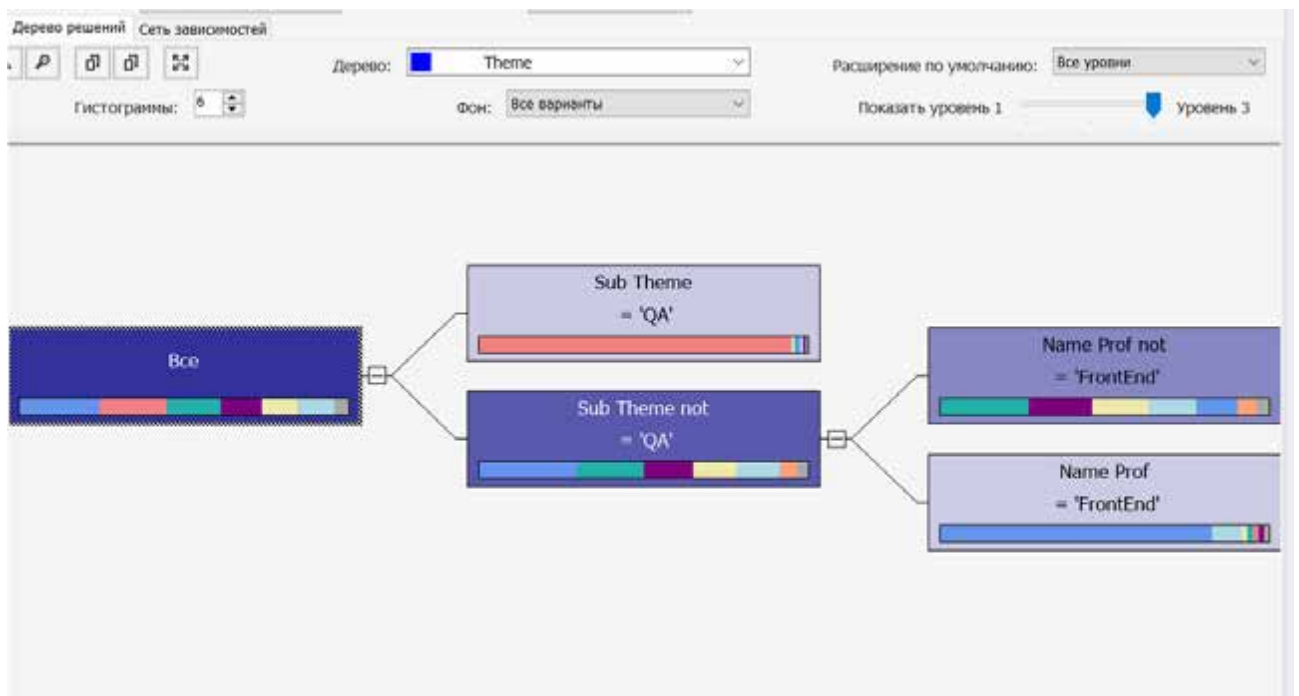


Рис. 4.8 Дерево рішення для прогнозування Theme

Спочатку проаналізуємо Name\_Prof (рис. 4.7). Дерево має три рівні – загальний, Theme є/не Front-end, Theme є/не Testing.

Загальний рівень містить значення всіх даних, представлено на рисунку 4.9. Наведений розподіл демонструє характер усіх даних куба. Найбільші показники у професії Engineering – багато компаній використовують загальний термін для назв посад їх спеціалістів. Схожі параметри мають і такі спеціальності як Java Developer, FrontEnd та Tester, отже вони мають схожі рівнозначні позиції популярності в соцмережі.

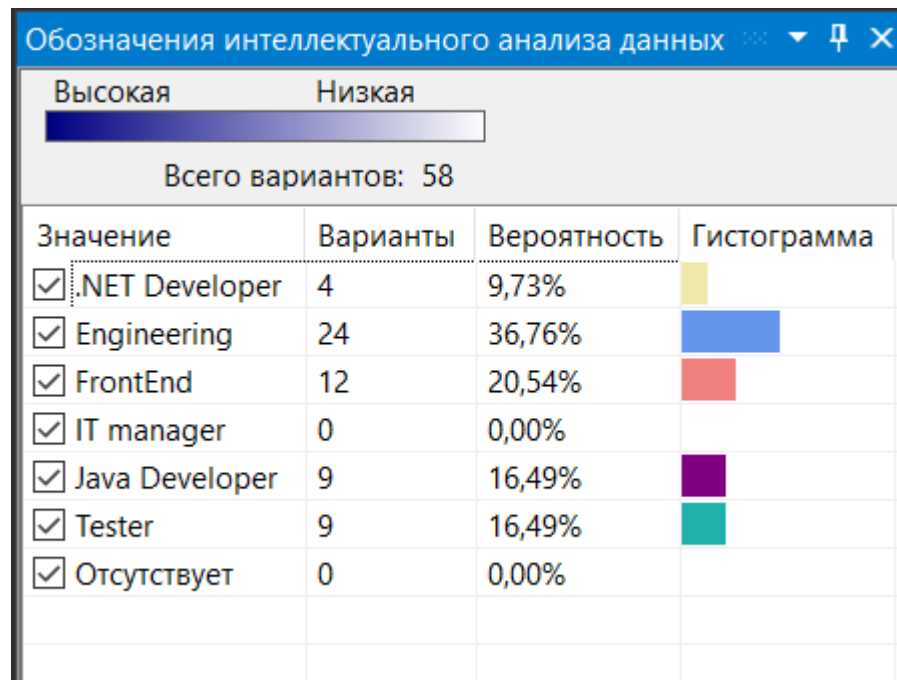


Рис. 4.9 Значення загального рівня

Далі ознайомимось з даними вузла Theme є Front-end, представлено на рисунку 4.10. До даного вузла належить 15 записів. Отже на тему Front-end пишуть публікації користувачі, що займають такі посади як FrontEnd (67,06%) чи Engineering (25,88%). Ймовірність, що інші користувачі можуть опублікувати контент на дану тему складає 2,35%.

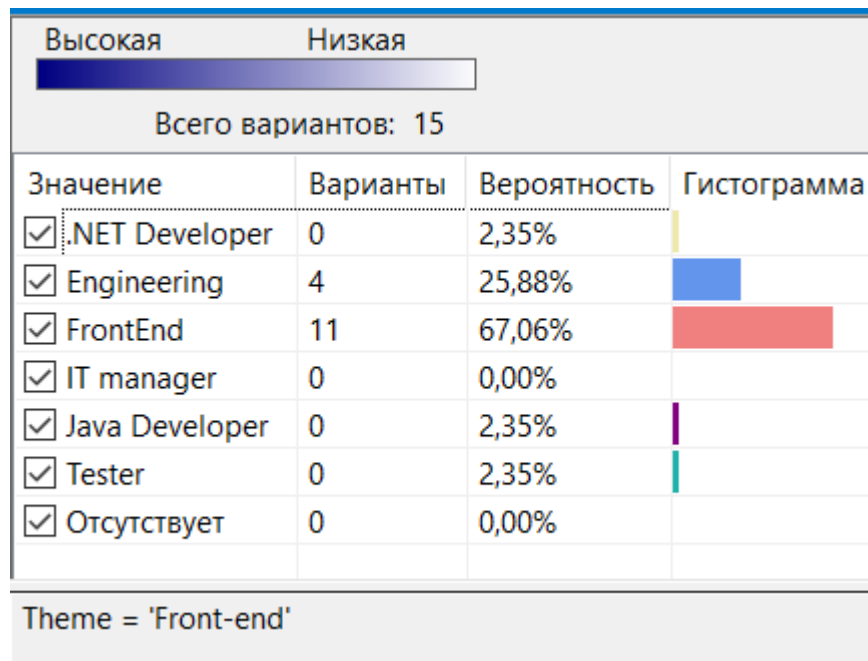


Рис. 4.10 Ймовірності публікації на тему Front-end різними типами користувачів

Наступний рівень Theme є/не Testing, значення його обох віток представлено на рисунку 4.11.

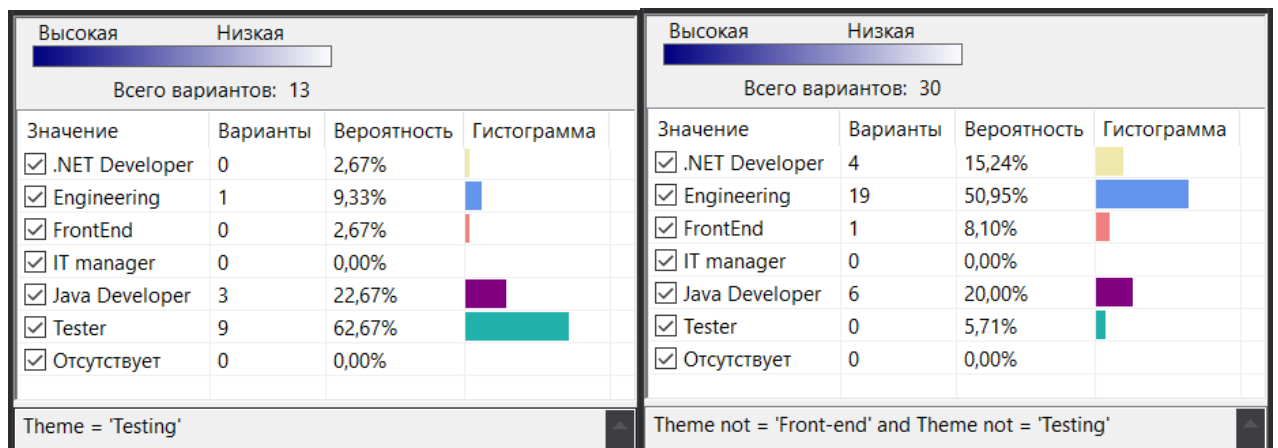


Рис. 4.11 Вітки ймовірності третього рівня

Перший скріншот містить ймовірності користувачів, що напишуть на тему Testing. Найбільша вірогідність звичайно належить користувачам, що обіймають посади Tester – 62,67%, але даною темою цікавляться і Java Developer – 22,67%.

Ймовірність, що інші користувачі можуть опублікувати контент на дану тему складає 2,67%.

На другому скріншоті представлений вузол Theme not = 'Front-end' and Theme not = 'Testing', що має вірогідності публікації контенту на інші теми, що не стосуються Front-end та Testing. Отже користувачі-Engineering найвірогідніше (50.95%) публікують на інші теми, спеціалісти інших галузей публікують не так часто на різні теми.

Тепер розглянемо дерево рішення для прогнозування Theme (рис. 4.8). Дерево має також три рівня, що діляться за підтемами. Така залежність природня та зрозуміла, отже нових знань щодо прогнозування теми публікацій дана структура не несе.

**4.2.2 Naive Bayes Algorithm.** Результатом розгортання структури NaivBaise є мережа залежностей представлена на рисунку 4.12.

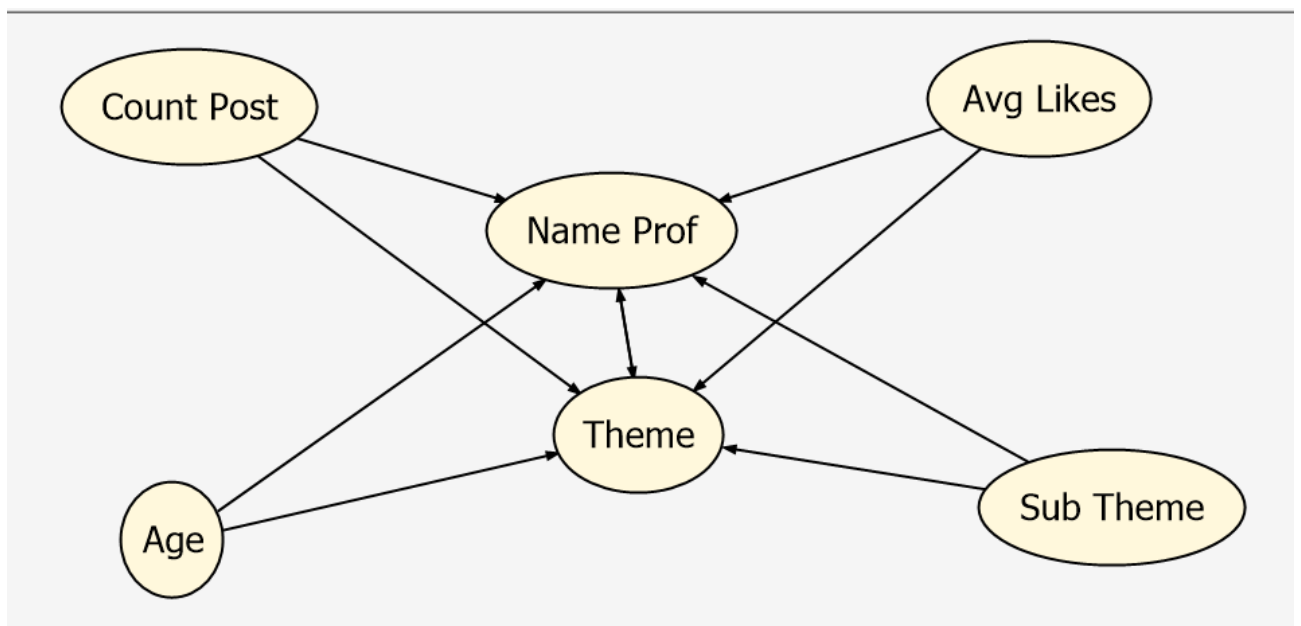


Рис. 4.12 Мережа залежностей

Вище представлено в центрі мережі знаходяться прогнозовані значення та їх взаємозв'язки з іншими даними. Щоб розібратись в мережі залежностей необхідно переглянути їх профілі атрибутів. Вони представлені на рисунку 4.13.



Профили атрибутов							
Атрибуты ↑	Состояния	Tester Размер: 9	.NET Developer Размер: 3	FrontEnd Размер: 11	Java Developer Размер: 9	Engineering Размер: 25	отсутс... Размер...
Age	<ul style="list-style-type: none"> <li>28 - 34</li> <li>&lt; 28</li> <li>34 - 40</li> <li>40 - 43</li> <li>Другое</li> </ul>						
Avg Likes	<ul style="list-style-type: none"> <li>&gt;= 4782</li> <li>2142 - 3225</li> <li>1412 - 2142</li> <li>&lt; 1412</li> <li>Другое</li> </ul>						
Count Post	<ul style="list-style-type: none"> <li>&gt;= 3</li> <li>&lt; 3</li> <li>Отсутствует</li> </ul>						
Sub Theme	<ul style="list-style-type: none"> <li>QA</li> <li>Python</li> <li>Java</li> <li>Azure</li> <li>Другое</li> </ul>						
Theme	<ul style="list-style-type: none"> <li>Front-end</li> <li>Testing</li> <li>Python</li> <li>Java</li> <li>Другое</li> </ul>						

Рис. 4.13 Профіль атрибуту Name\_prof

Проаналізуємо значення кожного атрибуту більш детально:

- Tester
  - вік спеціалістів в діапазоні 28-34 з ймовірністю 88,9%;
  - публікації спеціалістів збирають більше 4500 вподобань за місяць загалом;
  - кількість публікацій за місяць складає більше двох;
  - публікації на підтему QA.
- .NET Developer
  - вік спеціалістів немає явної залежності;
  - публікації спеціалістів збирають більше 4500 лайків за місяць;
  - кількість публікацій за місяць складає більше двох;
  - публікації на підтему Azure(66,7%) та iOS(33,3%).
- FrontEnd
  - вік спеціалістів в діапазоні 34-40 з ймовірністю 81,9%;
  - публікації спеціалістів збирають більше 4500 вподобань за місяць загалом;
  - кількість публікацій за місяць складає більше двох;

- публікації на підтему HTTP/CSS (45,5%), JS(18,2%) чи інші.
- Java Developer
  - вік спеціалістів в діапазоні 28-34 з ймовірністю 88,9%;
  - публікації спеціалістів збирають більше 4500 вподобань за місяць загалом;
  - кількість публікацій за місяць складає більше двох;
  - публікації на підтему QA(33,3%), Java(44,4) чи інші.
- Engineering
  - вік спеціалістів немає явної залежності;
  - публікації спеціалістів збирають більше 4500 вподобань за місяць загалом;
  - кількість публікацій за місяць складає більше двох;
  - публікації на різні теми, що немає явної залежності.

На рисунку 4.14 представлені профілі атрибутів для прогнозування теми публікації.

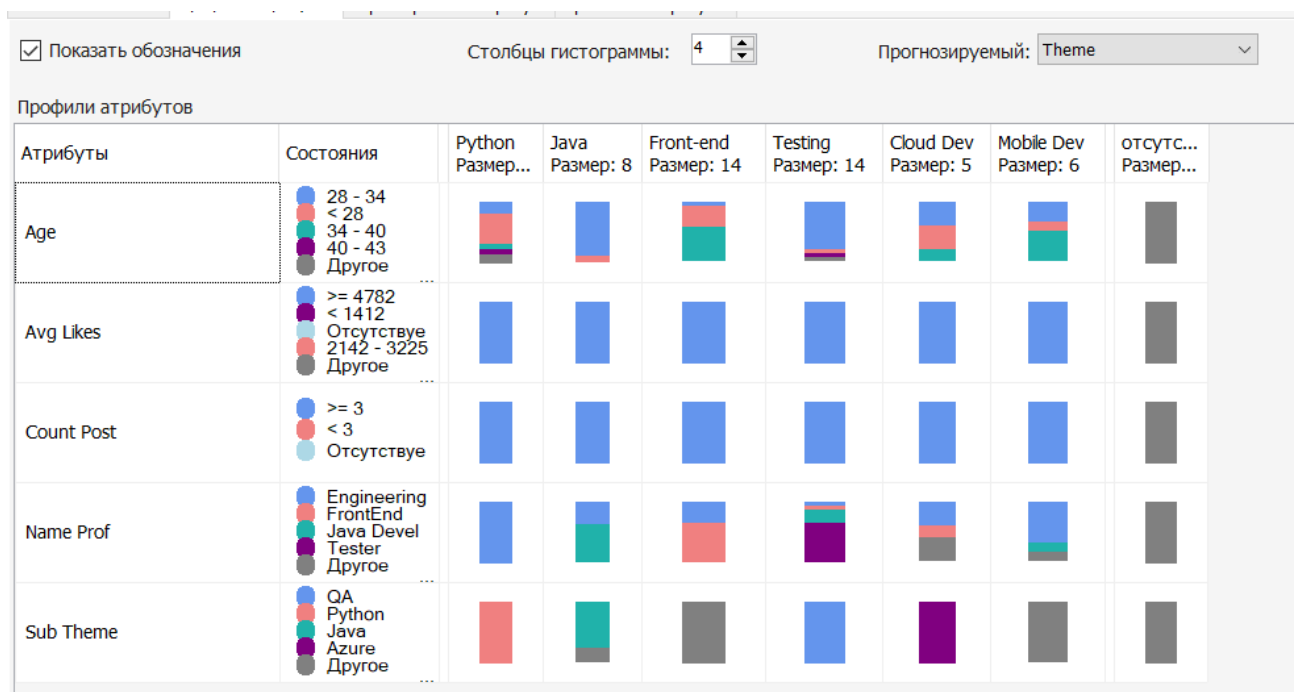


Рис. 4.14 Профілі атрибутів Theme

Розглянемо атрибути детальніше:

- Python
  - вік спеціалістів, що публікують на дану тему, до 28 складає 50%;
  - спеціалісти – Engineering.
- Java
  - вік спеціалістів, що публікують на дану тему, 28-34 складає 87,5%;
  - спеціалісти – Engineering (37,8%) та Java Developer(62,5%).
- Front-end
  - вік спеціалістів, що публікують на дану тему, 34-40 складає 57,1%;
  - спеціалісти – Engineering (35,7%) та FrontEnd(64,3%).
- Testing
  - вік спеціалістів, що публікують на дану тему, 28-34 складає 78,6%;
  - спеціалісти – Tester (64,3%) та Java Developer(21,4%).
- Cloud Dev
  - вік спеціалістів, що публікують на дану тему, до 34 складає 80%;
  - спеціалісти – Engineering (40%) та .NET Developer(40%).
- Mobile Dev
  - вік спеціалістів, що публікують на дану тему, 34-40 складає 50%;
  - спеціалісти – Engineering (66,7%).

**4.2.3 Association Algorithm.** Результатом розгортання структури 1Rule є перелік правил, представлений на рисунку 4.15, що мають вірогідність та важливість – чим вище значення, тим більш ймовірність підтвердження правила.

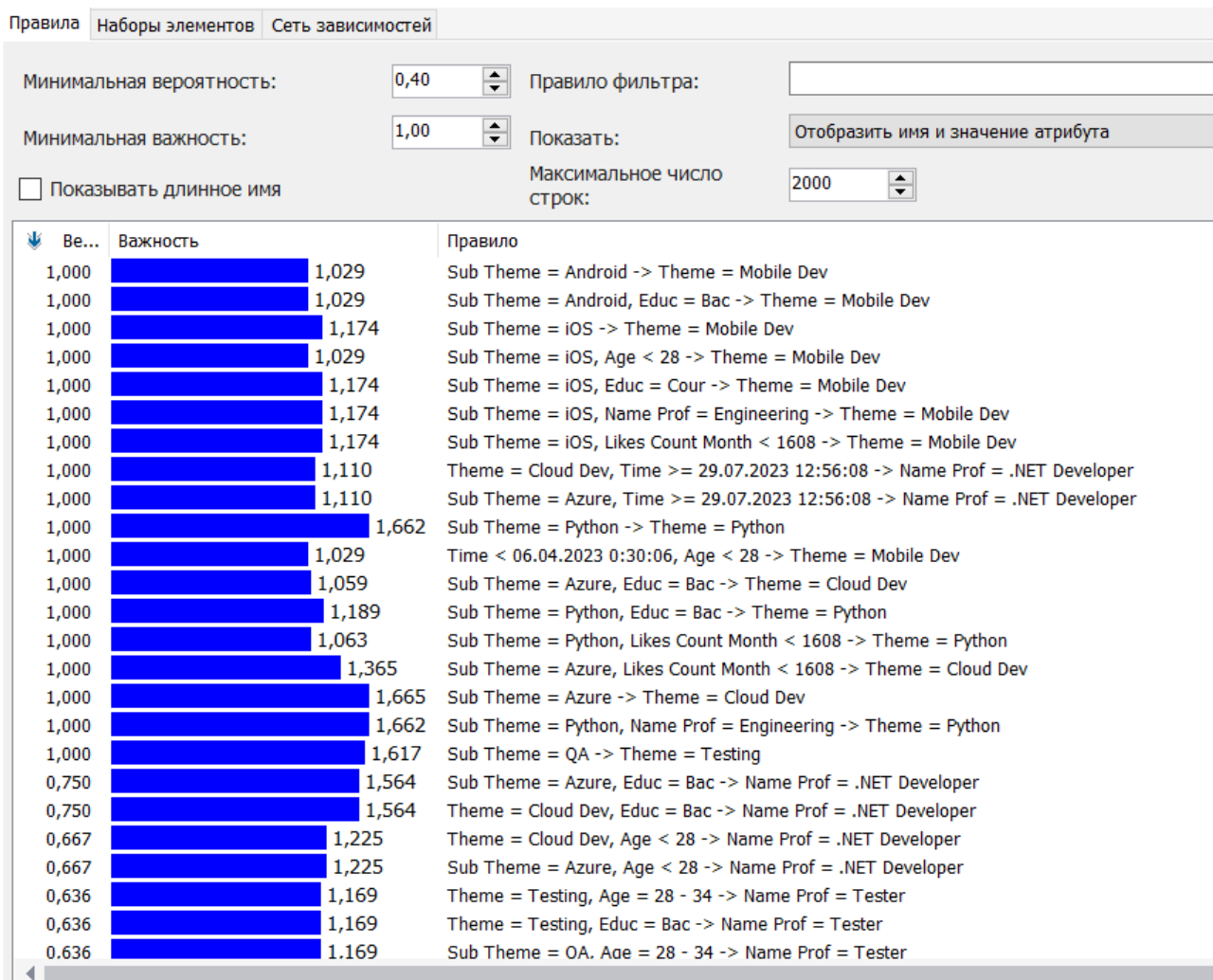


Рис. 4.15 Асоціативні правила структури 1Rule

Узагальнюючи отримані правила, їх значущості та вірогідності, можна зробити такі висновки:

- Більша частина правил стосується тем-підтем дописів, що є досить природньо та мають досить високі значення показників.
- Були підтверджені результати структури DecTree, що заснована на алгоритмі дерева рішень, а саме стійкі залежності теми-підтеми публікації та професій користувача.
- Виявлені правила, що стосуються віку спеціаліста, їх професій та теми допису, але зі незначними показниками.

**4.2.4 Clustering Algorithm.** Після обробки структури Cluster отримуємо модель даних представлену на рисунку 4.16.

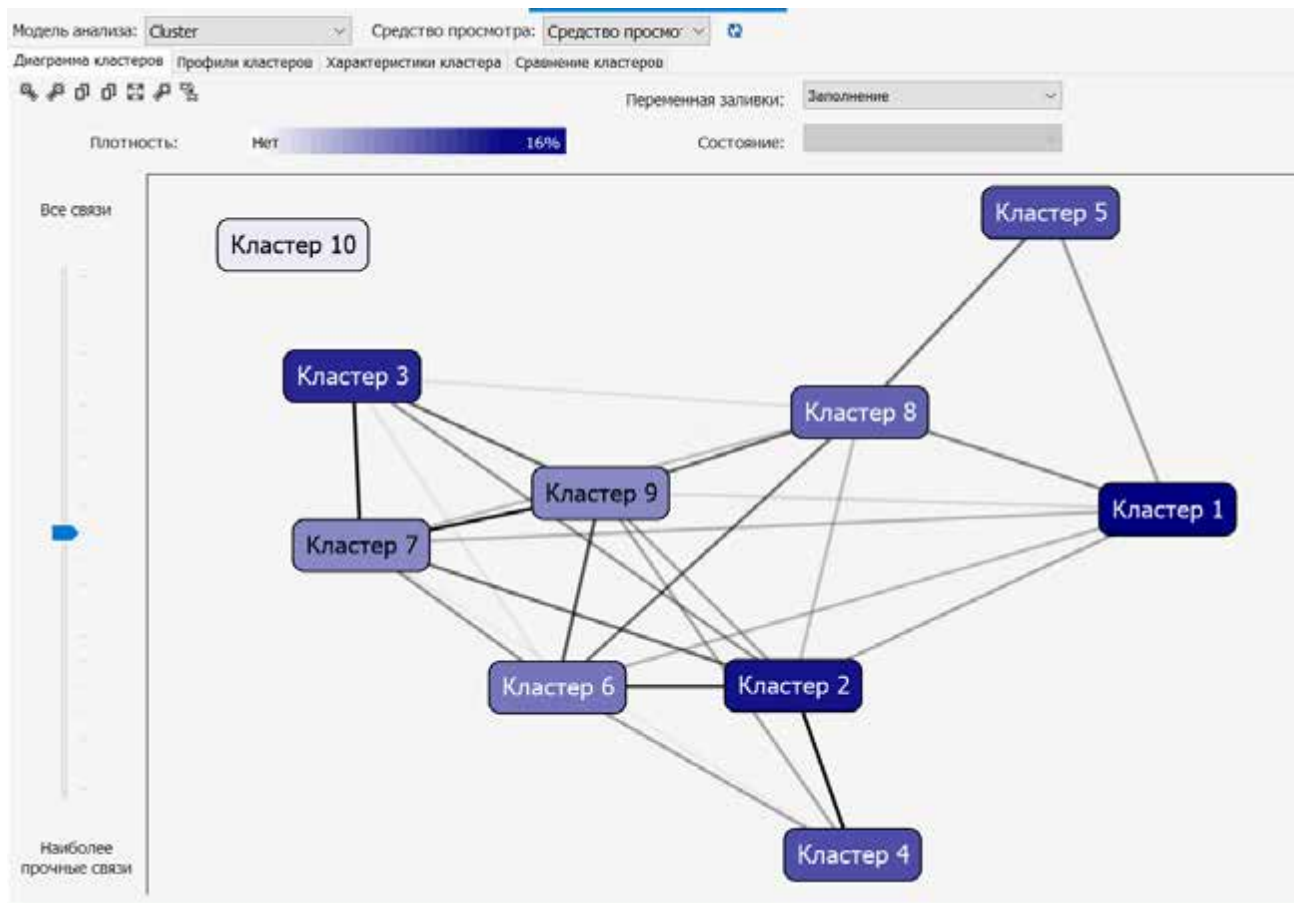


Рис. 4.16 Кластеры інтелектуальної структури Cluster

З представленої моделі можна побачити, що найбільше даних міститься саме в кластерах 1, 2 та 3, але варто проаналізувати всі кластери, що зрозуміти дані цілком. На рисунку 4.17 представлено профілі кластерів та їх характеристики.

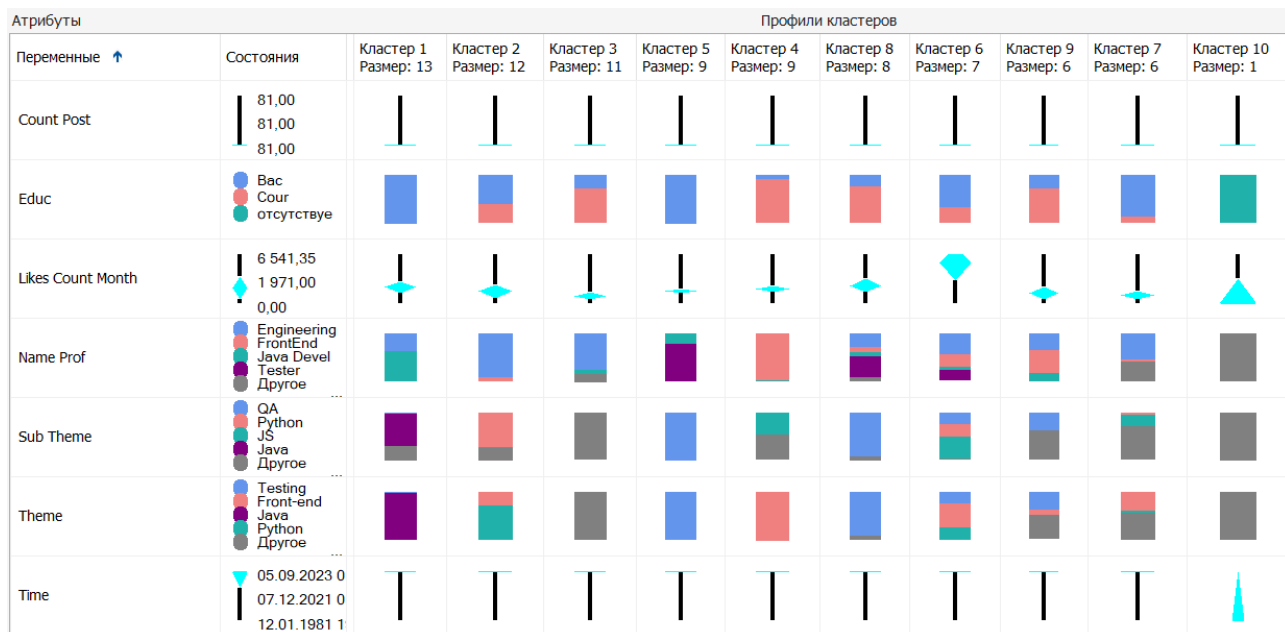


Рис. 4.17 Профілі кластерів

Опис кожного кластера дозволяє зрозуміти які дані загалом містяться в системі, їх кореляцію та особливості. Розглянемо кластери більш детально:

- Кластер 1: автори публікацій Java Developer (62,45%) чи Engineering (37,5%) з освітою – бакалавр, віком 27-30 (62,65%), дописи на тему Java(97,5%) набирають 1971 – 2998 (45,54%) вподобань;
- Кластер 2: автори публікацій Engineering (90,57%) з освітою – бакалавр(60,6%), віком 27-30 (49,63%), дописи на тему Python (70,48%) набирають 944 – 1971 (42,54%) вподобань;
- Кластер 3: автори публікацій чи Engineering (76,5%) з освітою – курси (70%), дописи на тему Mobile Dev (91,99%) набирають 944 – 1971 (50,3%) вподобань;
- Кластер 4: автори публікацій FrontEnd (96,6%) з освітою – курси (89,44%), віком 31-34 (93,27%), дописи на тему FrontEnd набирають 944 – 1971 (60,7%) вподобань;
- Кластер 5: автори публікацій Tester (77,97%) з освітою – бакалавр, віком 31-34 (75,86%), дописи на теми Testing набирають 944 – 1971 (90,45%) вподобань;

- Кластер 6: автори публікацій Engineering (43,45%) з освітою – бакалавр (66,14%), віком 27-30 (62,65%), дописи на тему FrontEnd (49,19%) набирають 2999 – 6541 (54,36%) вподобань;
- Кластер 7: автори публікацій Engineering (54,41%) з освітою – бакалавр (92,74%), віком 27-30 (52,78%), дописи на тему Cloud Dev (60,47%) набирають 944 – 1970 (54,3%) вподобань;
- Кластер 8: автори публікацій Tester (42,9%) з освітою – курси (78,56%), віком 27-30 (61,34%), дописи на тему Testing (95,74%) набирають 1971 – 2998 (41,04%) вподобань;
- Кластер 9: автори публікацій FrontEnd (60,12%) з освітою – курси (78,54%), віком 38-40 (75,36%), дописи на тему Testing (40,21%) набирають 944 – 1970 (56,12%) вподобань;
- Кластер 10: містить дані з відсутніми деякими полями.

Прогнозування даною структурою досить складний та заплутай процес, адже алгоритм більше підходить для аналізу існуючих даних в розрізі їх загальних типів та характеристик.

### 4.3 Аналіз ефективності використання алгоритмів

Структури інтелектуального аналізу виявили чимало неочевидних особливостей даних системи. Тепер варто проаналізувати ефективність цих алгоритмів для прогнозування технологій, що користуються популярністю серед спеціалістів, визначаючи як прогнозоване поле – назва посади (Name\_prof). Для оцінки в SSAS існує така функціонал [30], що дозволяє представити ефективність структури інтелектуального аналізу у вигляді діаграми точності прогнозу, порівнюючи з ідеальними умовами. Чим ближче лінія графіку структури до лінії ідеальної моделі тим ефективніший алгоритм цієї структури для прогнозування на основі даних систем. Для кращого читання діаграми варто зазначити:

- Вісь X – представляє відсоток перевіреного набору даних, що використовується для порівняння прогнозів.
- Вісь Y – представляє відсоток передбачених значень.
- Оцінка – використовується для порівняння з іншими моделями.

- Вірогідність прогнозу - показує оцінку ймовірності, необхідну для кожного прогнозу, щоб охопити показану цільову сукупність.

На рисунках 4.18-4.19 представлені діаграми точності прогнозу для кожної структури.

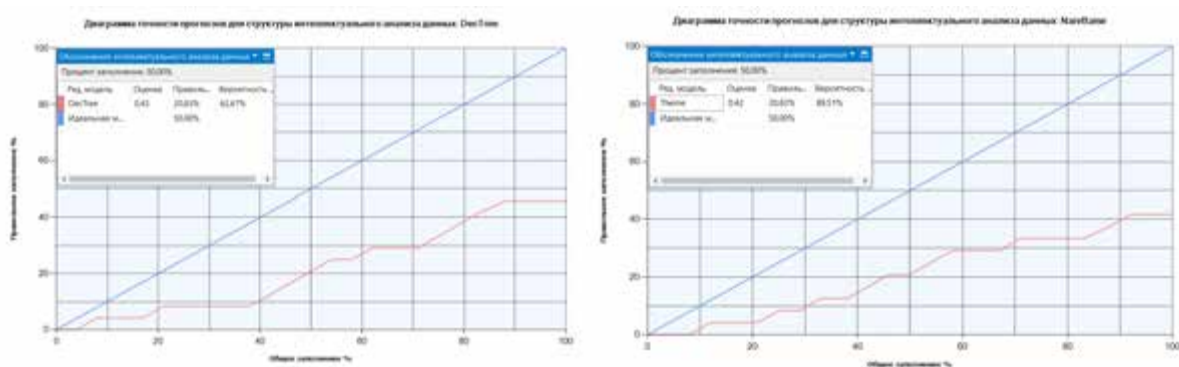


Рис. 4.18 Діаграми точності прогнозування DecTree та NaivBaise

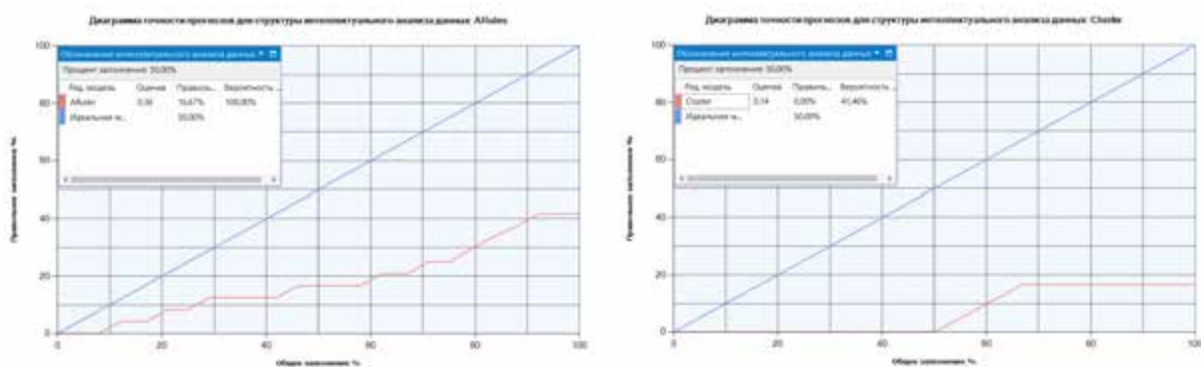


Рис. 4.19 Діаграми точності прогнозування 1Rule та Cluster

Узагальнені дані всіх показників діаграм наведено в таблиці 1. Деякі показники структур мають досить високі значення, але важливо зрозуміти, що ефективність застосованого алгоритму залежить саме від комплексних оцінок.



## Показники структур

Модель	Оцінка	Вірогідність прогнозування
DecTree	0.43	62.67%
NaiveBaise	0.42	89.51%
1Rule	0.36	100%
Cluster	0.14	41.46%

Виходячи з наведених показників в таблиці 1 можна зробити такі висновки:

- DecTree – одна з найефективніших структур інтелектуального аналізу серед інших, хоч значення оцінки не високе, вірогідність прогнозування охоплює більше половини даних системи.
- NaiveBaise – має одні з найкращих показників ефективності прогнозування, адже має майже такий самий показник оцінки, як і DecTree, але охоплює значно більшу кількість даних системи – 89.51%.
- 1Rule – структура інтелектуального аналізу має найвищий показник вірогідності прогнозування – 100%, але оцінка серед інших складає 0.36, отже поступається іншим.
- Cluster – малоефективна структура для прогнозування даних системи, адже має досить низькі показники.

Загалом можна сказати, що дослідження даних професійної соцмережі наведеними алгоритмами має свої переваги, що дозволяють дізнатись шукані відповіді.

## ВИСНОВКИ

Робота полягала в розробці веб-сайту для соціальної мережі професійного спрямування, а також створення окремого аналітичного блоку призначеного для отриманих даних з метою отримання нових знань.

В ході роботи були використані такі технології та інструменти: React.js, Redux, SASS, MS SQL Server, HTML5, CSS, JS, OLAP та Data Mining. Представлено опис аналогів розроблюваної системи та аналіз їх переваг та недоліків.

На основі приведеного системного аналізу предметної області та визначеного переліку всіх вимог до системи було проведено об'єктно-орієнтоване моделювання та створено опис операцій обробки даних в системі.

Наступним етапом при розробці було створення інформаційного та програмного забезпечення. До інформаційного забезпечення ввійшли: оперативна база даних для сайту соцмережі та сховище даних для аналітичного блоку. Щодо програмного забезпечення, то до нього ввійшли програмні модулі для сайту, OLAP куб та структури інтелектуального аналізу призначені для аналітичного блоку.

Завершення роботи представлено у вигляді опису інтерфейсу веб-сайта соцмережі, результати аналітичного блоку та аналіз отриманих знань.

В ході дослідження було знайдено відповідь на поставлені питання: через виявлення тісного зв'язку між посадою користувача та темами його дописів, визначити значні показники популярності в окремих напрямках не вдалося, але часто згадуваною темою в різних спеціалістах є тема тестування програмного забезпечення.

Були виявлені такі знання:

- Алгоритм дерева рішень виявив, що Engineering має найбільшу популярність, адже є загальним поняттям, а схожі параметри мають такі спеціальності як Java Developer, FrontEnd та Tester, отже вони мають схожі рівнозначні позиції популярності в соцмережі.

- Алгоритм наївного Баєса виявив зв'язки спеціалістів з їх віком, позицією та темою допису.
- Алгоритм асоціативних правил підтвердив отримані знання інших алгоритмів.

Дана робота спрощує дослідження даних в сфері соціальних мереж професійного спрямування та демонструє ефективність використання технологій OLAP та DM для цього процесу. Подальший розвиток системи надасть більше корисних знань, що зможуть певні переваги.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ЩО ТАКЕ СОЦІАЛЬНІ МЕРЕЖІ? [Електронний ресурс] URL: <https://futurenow.com.ua/shho-take-sotsialni-merezhi-vydy-klasyfikatsiya-bezpeka/>
2. LinkedIn [Електронний ресурс] URL: <https://ua.linkedin.com/>
3. Linked In – це ... [Електронний ресурс] URL: <https://cleverstaff.net/blog/uk/linkedin-tse/>
4. Stackoverflow [Електронний ресурс] URL: <https://stackoverflow.com/>
5. What is Stack Overflow?. [Електронний ресурс] URL: <https://www.zdnet.com/education/computers-tech/what-is-stack-overflow/>
6. Facebook [Електронний ресурс] URL: <https://uk-ua.facebook.com/>
7. What is Facebook? [Електронний ресурс] URL: <https://www.techtarget.com/whatis/definition/Facebook>
8. UML: огляд основних типів діаграм [Електронний ресурс] URL: <https://habr.com/ru/articles/738428/>
9. UML-діаграма прецедентів [Електронний ресурс] URL: <https://gazette.com.ua/rizne/uml-diyagrama-pretседentiv.html>
10. Що таке діаграма стану? [Електронний ресурс] URL: <https://uk.theastrologypage.com/state-diagram>
11. Що таке логічна модель? [Електронний ресурс] URL: [https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema6/tema6\\_2](https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema6/tema6_2)
12. MongoDB [Електронний ресурс] URL: <https://www.mongodb.com/>
13. MongoDB Tutorial [Електронний ресурс] URL: <https://www.w3schools.com/mongodb/>
14. СХОВИЩА ДАНИХ. [Електронний ресурс] URL: [https://stud.com.ua/121123/informatika/shovischa\\_daniv](https://stud.com.ua/121123/informatika/shovischa_daniv)
15. Розуміння Клієнт-Серверної Архітектури на прикладах. [Електронний ресурс] URL: <https://dou.ua/forums/topic/44636/>
16. React.js. [Електронний ресурс] URL: <https://legacy.reactjs.org/>
17. Redux [Електронний ресурс] URL: <https://redux.js.org/>

18. SASS [Электронный ресурс] URL: <https://sass-lang.com/>
19. The Best Guide to Know What Is React [Электронный ресурс] URL: <https://simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs>
20. What Is Redux? [Электронный ресурс] URL: <https://medium.com/swlh/what-is-redux-b16b42b33820>
21. SASS [Электронный ресурс] URL: <https://sass-lang.com/documentation/>
22. Барсегян А. А., Куприянов М. С., Степаненко В. В., Холод И. И. Методы и модели анализа данных: OLAP и Data Mining. — СПб.: БХВ-Петербург, 2004. — 336 с
23. Общие сведения о кубах OLAP в Service Manager для расширенной аналитики. [Электронный ресурс] URL: <https://docs.microsoft.com/ru-ru/system-center/scsm/olap-cubes-overview?view=sc-sm-2019>
24. Multidimensional Model Databases (SSAS) [Электронный ресурс] URL: <https://learn.microsoft.com/en-us/analysis-services/multidimensional-models/multidimensional-model-databases-ssas?view=asallproducts-allversions>
25. Огляд онлайнної аналітичної обробки (OLAP) [Электронный ресурс] URL: <http://surl.li/majux>
26. Microsoft Decision Trees Algorithm. [Электронный ресурс] URL: <https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-decision-trees-algorithm?view=asallproducts-allversions>
27. Microsoft Naive Bayes Algorithm. [Электронный ресурс] URL: <https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-naive-bayes-algorithm?view=asallproducts-allversions>
28. Microsoft Association Algorithm. [Электронный ресурс] URL: <https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-association-algorithm?view=asallproducts-allversions>
29. Microsoft Clustering Algorithm. [Электронный ресурс] URL: <https://docs.microsoft.com/en-us/analysis-services/data-mining/microsoft-clustering-algorithm?view=asallproducts-allversions>

30. Lift Chart (Analysis Services - Data Mining) [Электронный ресурс]

URL: <https://learn.microsoft.com/ru-ru/analysis-services/data-mining/lift-chart-analysis-services-data-mining?view=asallproducts-allversions>