

Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА


Галузь знань 12 – Інформаційні технології

Спеціальність 126 – Інформаційні системи та технології

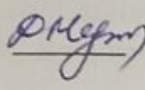
на тему: «Метод та форум-орієнтована інформаційна система з використання психіатричних препаратів»

КвРІСТ. 240178.24.01.02 ПЗ

Виконав: студент 2 курсу, група ІСТм-24-1

 Роман НЕДАШКІВСЬКИЙ
Підпис Ініціали, прізвище

Керівник: кандидат техн. наук, доцент
Науковий ступінь, вчене звання

 Дмитро МЕДЗАТИЙ
Підпис Ініціали, прізвище

До захисту допускаю:
Зав. кафедри КІІС,
PhD Ольга ПАВЛОВА



12 12 2025 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень МАГІСТР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

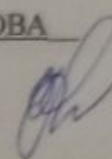
Спеціальність 126 ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ

Освітня програма ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА

“ 25 ” 08 2025 р.



ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА

Роману НЕДАШКІВСЬКОМУ

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Метод та форум-орієнтована інформаційна система з використання психіатричних препаратів

Керівник проекту (роботи) Дмитро МЕДЗАТИЙ, к.т.н., доцент.

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 25.08.2025 р. № 65

2. Строк подання студентом проекту (роботи) на кафедру 01.12.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) _____

Огляд відомих методів та рішень

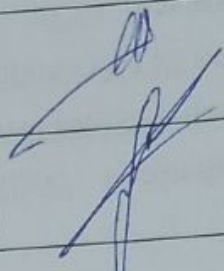
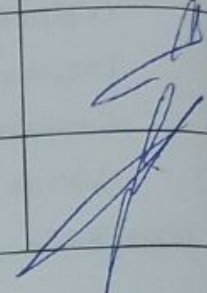
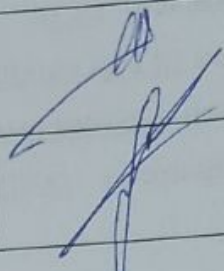
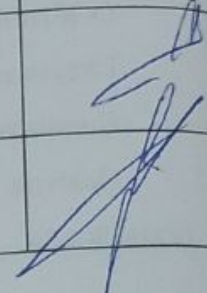
Архітектура та технологічне обґрунтування інформаційної системи p-psuforum

Метод та алгоритм структурно-аналітичної обробки даних інформаційної системи

Форум-орієнтована інформаційна система з використання психіатричних препаратів

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) _____

6. Консультанти розділів кваліфікаційної роботи магістра

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Сергій ЛИСЕНКО, професор кафедри КПС		
Антиплагіат	Андрій НІЧЕПОРУК, доцент кафедри КПС		

7. Дата видачі завдання « 01 » 09 2025р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи з керівником	01.09.2025	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	15.09.2025	виконано
3	Робота над розділом 1 – огляд відомих методів та рішень	01.10.2025	виконано
4	Робота над розділом 2 – архітектура та технологічне обґрунтування інформаційної системи p-psyforum	15.10.2025	виконано
5	Робота над наковою публікацією	15.10.2025	виконано
6	Робота над розділом 3 – метод та алгоритм структурно-аналітичної обробки даних інформаційної системи	01.11.2025	виконано
7	Робота над розділом 4 – форум-орієнтована інформаційна системи з використання психіатричних препаратів	15.11.2025	виконано
8	Оформлення пояснювальної записки згідно вимог	01.12.2025	виконано
9	Попередній захист ВКР	02.12.2025	виконано
10	Захист ВКР на засіданні ЕК	19.12.2025	

Студент

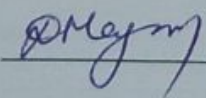


Роман НЕДАШКІВСЬКИЙ

Підпис

Ініціали, прізвище

Керівник роботи



Дмитро МЕДЗАТИЙ

Підпис

Ініціали, прізвище

РЕФЕРАТ

Тема кваліфікаційної роботи магістра: Метод та форум-орієнтована інформаційна система з використання психіатричних препаратів

Автор роботи: Роман НЕДАШКІВСЬКИЙ

Керівник роботи: Медзатий Д.М., кандидат технічних наук, доцент, доцент кафедри комп'ютерної інженерії та інформаційних систем.

Пояснювальна записка: 70 с., 23 рис., 7 табл., 2 дод., 66 джерел.

Перелік ключових слів: форум-орієнтована інформаційна система, психіатричні препарати, агрегація даних, структурний облік, алгоритм транзакції, цілісність інформації, обмін досвідом, база знань.

Об'єктом дослідження є процес управління інформацією та обміну досвідом у межах спеціалізованої інформаційної системи.

Предметом дослідження є метод структурно-статистичної агрегації даних та форум-орієнтована інформаційна система з використання психіатричних препаратів.

Метою кваліфікаційної роботи магістра є підвищення ефективності обміну знаннями щодо психіатричних препаратів шляхом створення форум-орієнтованої інформаційної системи, яка забезпечує можливість структурованого обміну досвідом, моніторингу активності користувачів та підтримки актуальної бази знань.

Для розв'язання поставлених задач використовувалися методи за допомогою яких можна розв'язати поставлені задачі та використати основні положення системного аналізу (ієрархічності, декомпозиції та ін.), загальної теорії систем, теорії моделювання процесів. При проведенні моделювання процесу обміну досвідом та при розробленні методу управління даними використано теоретико-множинні підходи, методи концептуального моделювання, евристичні оцінки, принципи побудови баз знань та формування логічного висновку.

Наукова новизна отриманих результатів: Набув подальшого розвитку метод структурно-статистичної агрегації та трансформації даних, який, на відміну від

відомих підходів, інтегрований безпосередньо в транзакційний цикл форум-орієнтованої системи. Це забезпечує автоматичний, цілісний та актуальний облік персональної активності користувачів (статистичні метрики), необхідний для верифікації контенту.

Набула подальшого розвитку інформаційна технологія забезпечення цілісності та структурного обміну знаннями, орієнтована на створення спеціалізованих форум-орієнтованих систем. Вона відрізняється використанням алгоритму життєвого циклу користувацької транзакції для управління контентом та статистики, що підвищує надійність системи.

На основі проведених досліджень розроблена архітектура та компоненти програмного забезпечення форум-орієнтованої інформаційної системи. Архітектура включає трирівневу модель (клієнтська частина, серверний API, реляційна СУБД) та модулі, що забезпечують реалізацію розробленого методу агрегації даних, управління контентом та контроль цілісності облікових даних.

Практична значимість отриманих результатів полягає у реалізації форум-орієнтованої інформаційної системи, яка є ефективним інструментом для фахівців, пацієнтів та дослідників. Система забезпечує структурований обмін досвідом, ведення аналітики активності користувачів та контроль цілісності облікових даних. Ключовими перевагами такої системи стає: оперативне оновлення інформації; підтримка цілісності статистичного обліку за допомогою розробленого методу; підвищення прозорості та надійності обміну інформацією. Створена система є корисним для користувачів інструментом для підтримки актуальної бази знань та забезпечення ефективної взаємодії у сфері використання психіатричних препаратів.

ЗМІСТ

КАЛЕНДАРНИЙ ПЛАН	Error! Bookmark not defined.
СКРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ.....	5
ВСТУП.....	6
1 ОГЛЯД ВІДОМИХ МЕТОДІВ ТА РІШЕНЬ	8
1.1 Аналіз предметної області і виявлення наявних проблем і завдань	8
1.2 Порівняльний аналіз переваг та недоліків існуючих рішень	9
1.3 Підходи до вирішення задачі за темою дослідження.....	19
1.4 Постановка задачі	20
2 АРХІТЕКТУРА ТА ТЕХНОЛОГІЧНЕ ОБҐРУНТУВАННЯ	
ІНФОРМАЦІЙНОЇ СИСТЕМИ P-PSYFORUM.....	22
2.1 Формалізація та вибір базових компонентів форумно-орієнтованої інформаційної системи P-PsyForum.....	22
2.2 Концептуальне проектування інформаційної системи. Обґрунтування компонентів клієнтської частини	31
2.3 Висновки	37
3 МЕТОД ТА АЛГОРИТМ СТРУКТУРНО-АНАЛІТИЧНОЇ ОБРОБКИ	
ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ	39
3.1 Метод моніторингу інформаційної чистоти та оцінки довіри контенту	39
3.2 Алгоритм життєвого циклу користувачької транзакції та обліку активності 46	46
3.3 Висновки	51
4 ФОРУМ-ОРІЄНТОВАНА ІНФОРМАЦІЙНА СИСТЕМА З	
ВИКОРИСТАННЯ ПСИХІАТРИЧНИХ ПРЕПАРАТІВ.....	53
4.1 Діаграма архітектури та основні компоненти інформаційної системи	53

4.2 Програмне забезпечення форуму та інтерфейс системи.....	61
4.3 Ключовий інтерактивний функціонал та апробація методу структурно-статистичної агрегації.....	68
4.4 Висновки.....	75
ВИСНОВКИ.....	77
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	79
ДОДАТОК А.....	85
ДОДАТОК Б.....	87
ДОДАТОК В.....	90

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ІС - інформаційна система

БД - база даних

СУБД - система керування базами даних

ВСТУП

У сучасній психіатричній та медичній практиці одним із найважливіших аспектів є вибір та використання лікарських засобів, інформація про які потребує глибоких знань та врахування індивідуальних характеристик пацієнтів.

Незважаючи на наявність великої кількості довідкових джерел та наукових публікацій, інформація про психіатричні препарати часто може не відповідати дійсності та й не завжди доступна для швидкого використання. Крім того, існує проблема недостатньої інтеграції між медичною спільнотою та пацієнтами, що обмежує обмін досвідом, знижує ефективність лікування та створює ризик поширення неперевіреної інформації.

Із цього слідує, відсутність єдиного, структурованого та надійного інформаційного середовища для інтеграції клінічного, наукового та практичного досвіду є суттєвою проблемою. Аналіз та розробка сучасних засобів для забезпечення структурованого обміну знаннями та моніторингу активності користувачів має вирішальне значення для підвищення якості консультативної допомоги та забезпечення надійності поширюваної інформації.

Актуальність роботи полягає у необхідності створення інформаційної технології, що дозволяє структурувати обмін досвідом та підтримувати актуальну базу знань щодо психіатричних препаратів, одночасно контролюючи цілісність облікових даних користувачів. Відповідно, дослідження спрямоване на розробку методу та програмної реалізації форум-орієнтованої інформаційної системи, що вирішує ці завдання.

Метою кваліфікаційної роботи є підвищення ефективності обміну знаннями щодо психіатричних препаратів шляхом створення форум-орієнтованої інформаційної системи, яка забезпечує можливість структурованого обміну досвідом, моніторингу активності користувачів та підтримки актуальної бази знань.

Поставлена мета досягається розв'язанням таких основних задач:

1) необхідно розробити метод структурно-статистичної агрегації та трансформації даних, орієнтований на забезпечення цілісного та актуального обліку активності користувачів системи;

2) метод повинен бути формалізований у вигляді алгоритму життєвого циклу користувацької транзакції, що описує послідовність дій від ініціації запиту до динамічного оновлення статистики;

3) необхідно розробити архітектуру та здійснити програмну реалізацію форум-орієнтованої інформаційної системи, яка забезпечує функціональність обміну даними та реалізацію розробленого методу.

Об'єктом дослідження є процес управління інформацією та обміну досвідом у межах спеціалізованої інформаційної системи.

Предметом дослідження є метод структурно-статистичної агрегації даних та форум-орієнтована інформаційна система з використання психіатричних препаратів.

Практична цінність отриманих результатів полягає у реалізації форум-орієнтованої інформаційної системи. В результаті виконаного наукового дослідження розроблена інформаційна система є ефективним інструментом для фахівців та користувачів.

У даній роботі викладено вимоги до методології розробки інформаційної системи та представлено її програмну реалізацію. Для розв'язання поставлених задач використовуються основні положення системного аналізу (ієрархічності, декомпозиції та ін.), загальної теорії систем, теорії моделювання процесів.

За темою кваліфікаційної роботи магістра опублікована одна стаття у фаховому виданні і одна стаття у Scopus. Недашківський Р.В. Форум-орієнтована інформаційна система з використання психіатричних препаратів. Філософські виміри техніки: *Збірник тез IV Міжнародної наукової конференції молодих учених та здобувачів вищої освіти*, 26 листопада 2025 р. С. 100–101.

1 ОГЛЯД ВІДОМИХ МЕТОДІВ ТА РІШЕНЬ

1.1 Аналіз предметної області і виявлення наявних проблем і завдань

Психіатрична практика є однією із найважливіших сфер де правильний вибір та використання психіатричних препаратів, які потребують глибоких знань та враховують індивідуальні характеристики пацієнта [1]. Не дивлячись на наявність великої кількості джерел та наукових публікацій, інформація про лікарські засоби часто розпорошується, представлена в різних розмірах і не завжди доступна для швидкого використання в клінічній активності [2].

Саме тому, важливими проблемами для цього поля є фрагментація даних, обмежений доступ до актуальних знань сфери, дійсно великий ризик поширення дезінформації та відсутність практичного та вручного інформаційного середовища для інтеграції клінічної, наукової та практичної інформації.

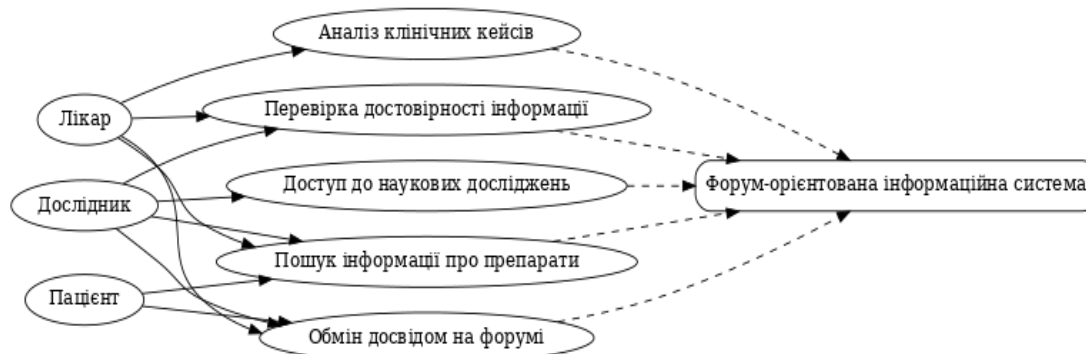


Рисунок 1.1 – Структура роботи системи

Представлена UML-діаграма прецедентів відобразила основних учасників у системі - лікаря, пацієнта та дослідника, а також сценарії їхньої можливої взаємодії з форум-орієнтованою інформаційною системою. Лікар отримує можливість здійснити пошук інформації про психіатричні препарати, брати участь в обговореннях на самому форумі, перевіряти достовірність отриманих даних та аналізувати реальні клінічні кейси [3].

Пацієнт отримує доступ до бази знань, може ділитися своїм досвідом лікування і брати участь у тематичних дискусіях. Дослідник, використовує систему для пошуку важливих для нього даних, участі в дискусіях і доступу до різноманітних результатів наукових досліджень [4-5]. Діаграма показала різні рівні доступу та функціональність для кожної категорії із користувачів, що дає розуміння про універсальність та практичну значимість запропонованої системи.

1.2 Порівняльний аналіз переваг та недоліків існуючих рішень

Розвиток інформаційних технологій у сфері психічного здоров'я призвів до появи великої кількості різних інформаційних систем та онлайн-платформ, спрямованих на підтримку медичної практики та обмін знаннями між користувачами цих різноманітних систем [7-8]. Саме у галузі психіатрії особливе значення мають ті системи, що дозволяють акумулювати дані про потрібні психіатричні засоби, забезпечувати швидкий доступ до довідкової інформації про препарати та навіть створювати середовище для комунікації між лікарями, пацієнтами та дослідниками.

Аналіз різних платформ дає можливість не лише визначити їхні сильні сторони, але ще й виявити їх слабкі місця та обмеження, які треба врахувати при проектуванні нової інформаційної системи [9]. Саме тому, порівняльний аналіз існуючих систем є важливим для початку дослідження, що дозволить сформулювати дійсно потрібні пріоритети до майбутнього рішення та уникнути повторення недоліків наявних розробок.

Розуміючи важливість даної теми, створення саме форум-орієнтованої інформаційної системи для психіатричних препаратів передбачає інтеграцію двох потрібних компонентів: структурованих даних про лікарські засоби та механізмів спільного обміну знаннями між користувачами.

Саме цей підхід дозволяє забезпечити не тільки лиш доступ до актуальної та перевіреної фармакологічної інформації, але й створити гідну платформу, де користувачі зможуть обговорювати практичні аспекти застосування препаратів,

обмінюватися своїм досвідом щодо можливих побічних ефектів та терапевтичної ефективності, а ще отримувати рекомендації від фахівців [10].

Саме такий комплексний підхід створює умови для формування безпечного, зручного та наукового середовища, що стане важливою основою для підтримки різних клінічних рішень, навчання медичних фахівців та проведення наукових досліджень у сфері психіатрії [11-12].

Drugs.com є однією із найбільших та найпопулярніших міжнародних онлайн-платформ, що спеціалізуються на наданні довідкової інформації про лікарські препарати. Система орієнтується на широку аудиторію: як на медичних працівників, так і на пацієнтів, які прагнуть отримати достовірні відомості про призначені їм препарати.

Головною перевагою Drugs.com є величезна база даних, яка охоплює тисячі препаратів, включаючи як торгові, так і міжнародні непатентовані назви. Кожна картка препарату має детальний опис складу, показання до застосування, можливі побічні ефекти, можливу взаємодію з іншими лікарськими засобами, рекомендації щодо дозування та умови зберігання препаратів [13-14].

При цьому, маючи свої переваги, Drugs.com має й певні недоліки. Платформа є чисто довідковим ресурсом і не передбачає інтерактивної взаємодії між користувачами, відсутні механізми форумного обговорення чи обміну практичним досвідом [15].

Для психіатричної сфери це є суттєвим обмеженням, адже обговорення клінічних випадків та індивідуальної реакції пацієнтів на препарати має дійсно важливе значення. Drugs.com можна та треба розглядати як високоякісне джерело довідкової інформації, але воно не здатне забезпечити функції комунікації та колективного аналізу.

Drugs.com вирізняється своїм різноманіттям та багатим функціоналом, який робить його корисним ресурсом у своїй інформаційній сфері про лікарські засоби.

Платформа пропонує користувачам не тільки стандартний пошук препаратів за назвою чи діючою речовиною, а ще й спеціальні інструменти, такі як «Drug Interactions Checker» для перевірки взаємодії між кількома ліками, «Pill Identifier»

для визначення препарату за зовнішнім виглядом, а ще базу даних про побічні ефекти та рекомендації щодо їхнього контролю.

Платформа регулярно оновлюється та містить не тільки корисну інформацію від різних фармацевтичних компаній та державних регуляторів, так а ще й відгуки пацієнтів, що створює повноцінне бачення про ефективність різних ліків у реальних умовах [16-17].

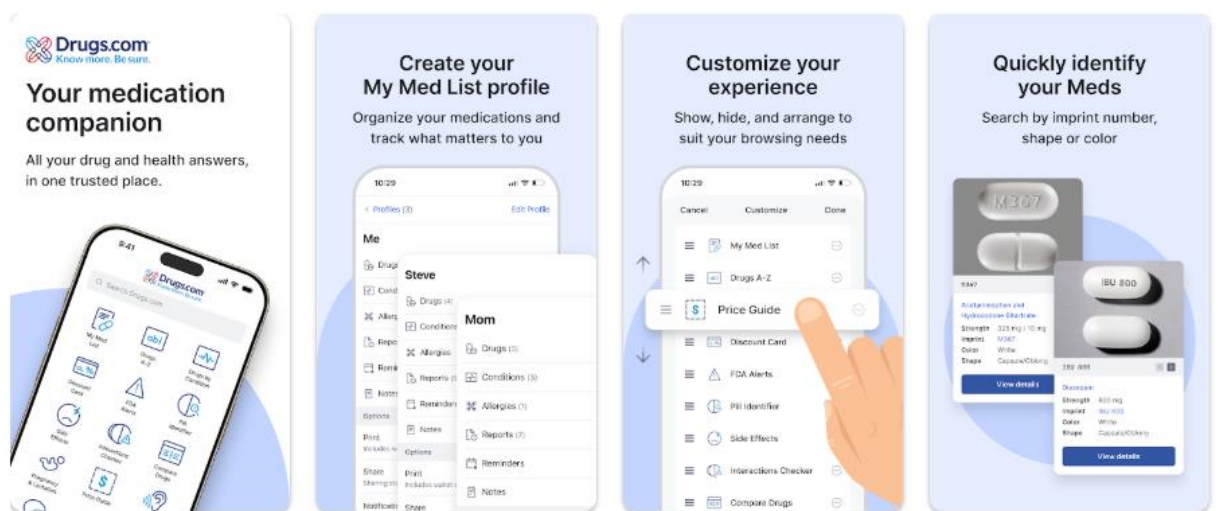


Рисунок 2.2 – Різноманітний функціонал Drugs.com

Drugs.com, не дивлячись на те що має свій високий рівень авторитетності та багатий функціонал, має й обмеження, які звужують його практичне використання у специфічній сфері психіатрії.

Перша проблема полягає у тому, що ресурс орієнтований передусім на загальну фармакологію, тому психотропні препарати хоч і представлені достатньо широко, але без глибокої спеціалізації на клінічних аспектах їх застосування у психіатрії.

Друга проблема впливає у системі бо вона не передбачає інтеграції з електронними медичними записами та клінічними системами підтримки прийняття рішень, що знижує її цінність для практикуючих лікарів у їхній повсякденній роботі [18-19].

WebMD - ще одна з найбільших і найвідоміших у всьому світі онлайн-платформ, яка спеціалізується на наданні медичної інформації широкому колу користувачів. Сайт пропонує доступ до різних медичних довідників, описів захворювань, симптом-чекерів, статей, блогів і новин у сфері охорони здоров'я.

Особливістю саме цієї платформи є її орієнтація на звичайних користувачів, а не тільки на лікарів. Інформація подається у зрозумілому для людей форматі, часто супроводжується ілюстраціями, порадами з профілактики та поясненнями медичних термінів.

WebMD містить інтегровані сервіси для пошуку лікарів, онлайн-консультації, а ще ж персоналізовані рекомендації для здорового способу життя. Важливою фішкою цієї системи є поєднання науково обґрунтованої інформації з доступною для пересічного користувача мовою подання, саме це робить її зручною для масового використання.

Однією з ключових переваг WebMD є зручна та добре структурована пошукова система для пошуку лікарських засобів, яка сильно полегшує доступ користувачів до необхідної для них інформації. Сервіс дозволяє знаходити препарати за їхньою назвою, діючою речовиною та навіть за характерними ознаками, такими як форма випуску чи виробник [20]. Пошукова система інтегрована з довідковими матеріалами та статтями, що дає змогу користувачам одразу отримати контекст та корисні поради щодо правильного використання медикаментів.

Це робить WebMD не просто статичним каталогом, а інтелектуальним асистентом, який допомагає користувачам орієнтуватися у великих масивах медичних даних.

Проте, незважаючи на високу точність довідкової інформації, платформа орієнтована переважно на стандартизовані клінічні описи, що залишає простір для пошуку рішень, які б поєднували офіційні дані з практичним досвідом пацієнтів у форматі інтерактивних обговорень

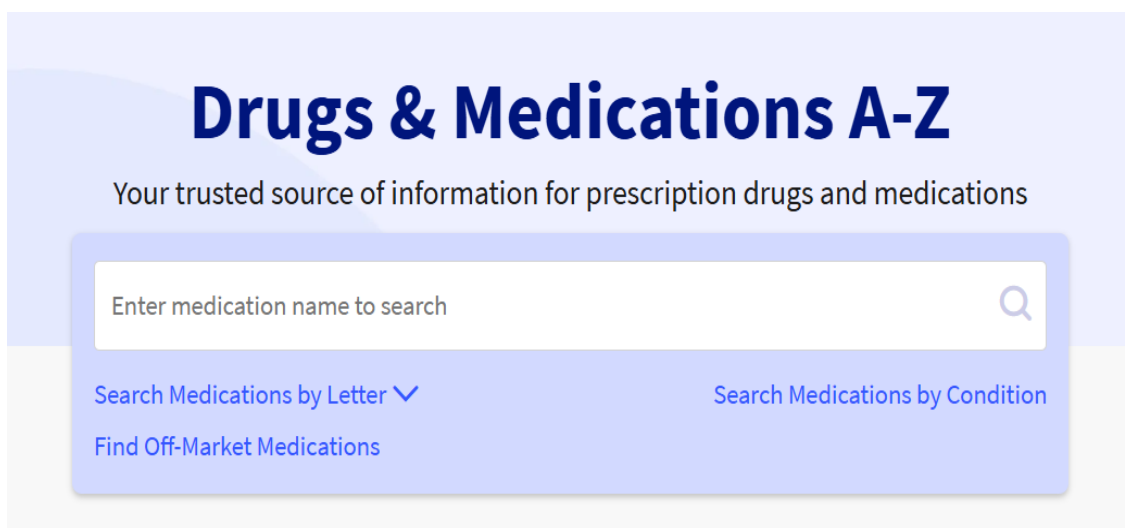


Рисунок 2.3 – Зручна пошукова система для пошуку потрібних препаратів

Не зважаючи на очевидні переваги та провідне місце серед медичних платформ, WebMD має й низку обмежень, які треба враховувати під час порівняльного аналізу. Найбільш поширеною критикою є надмірна популяризація контенту: з метою зробити інформацію доступною для широкої аудиторії, ресурс подає матеріали у спрощеному вигляді, що сильно знижує їхню наукову глибину [21]. Це створює певний баланс між «доступністю» та «точністю», що може бути неприйнятним для багатьох медичних фахівців.

Наукова спільнота ще відзначає, що WebMD орієнтований здебільшого на англomовну аудиторію, і хоча платформа має міжнародне визнання, мовна доступність лишається обмеженою [22]. WebMD можна назвати дійсно хорошим інформаційним центром у сфері цифрової медицини, що значною мірою вплинув на розвиток електронних медичних сервісів, проте використання цього ресурсу потребує критичного мислення користувача, співставлення з іншими науковими джерелами та обов'язкового підтвердження медичними фахівцями.

PatientsLikeMe також всесвітньо відома онлайн-платформа, орієнтована на створення спільноти пацієнтів, які обмінюються своїм досвідом лікування, застосування лікарських препаратів та подолання хронічних і рідкісних захворювань. Система заснована у США та за час свого існування об'єднала сотні тисяч користувачів із різних країн світу.

Основна ідея платформи в тому, щоб пацієнти мали змогу самостійно вести електронні щоденники стану здоров'я, де фіксують інформацію про діагноз, призначені ліки, побічні ефекти, результати лікування та їх загальне самопочуття[23].

Така система дозволяє формувати більш глибокі та предметні дискусії, оскільки учасники бачать контекст, у якому інші користувачі приймають препарати, і можуть порівнювати свій досвід. Вибір теми для обговорення на PatientsLikeMe зазвичай має саме прикладний характер: від обговорення нових лікарських засобів і клінічних досліджень до аналізу ефективності конкретних методів лікування у повсякденних умовах.

Особливістю платформи є можливість збору й агрегації даних користувачів, які в подальшому використовуються для наукових досліджень і статистики, що робить обговорення не лише корисними для учасників, а й корисним для широкої медичної спільноти.

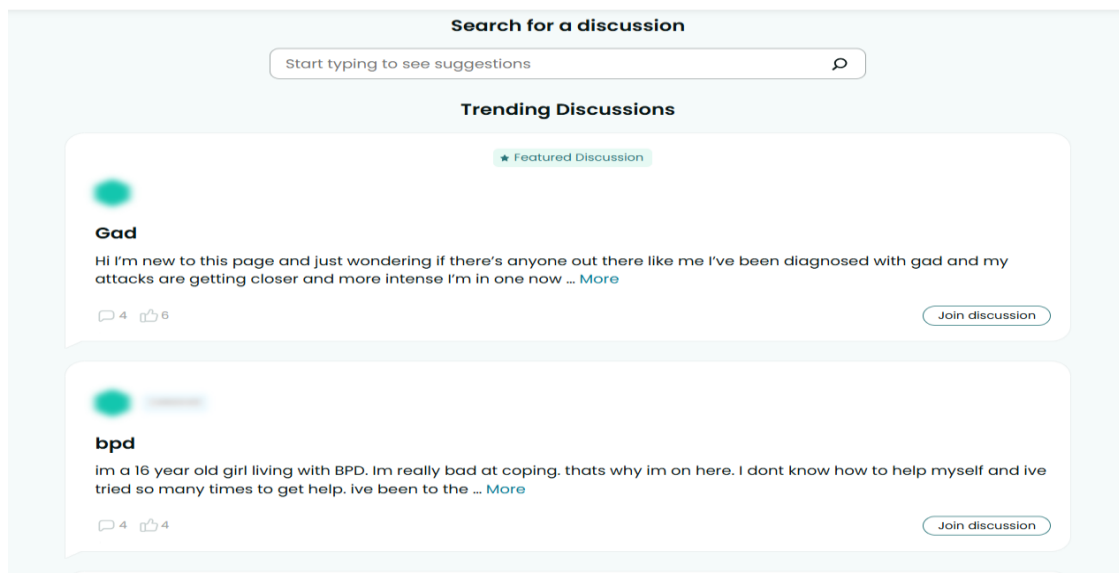


Рисунок 2.4 – Список тем для обговорення створених користувачами

PatientsLikeMe має як свої переваги, так і низку незручних обмежень. До переваг можна віднести створення цінного масиву емпіричних даних, який можна використовувати як пацієнтам, так і дослідникам та фармацевтичним компаніям

для вивчення ефективності препаратів у реальних умовах, поза рамками клінічних випробувань. [24].

Проблемними залишаються питання надійності даних, оскільки інформація подається самими пацієнтами без медичної перевірки, а також конфіденційності - адже чутливі медичні відомості збираються й зберігаються у великій базі даних, що вимагає від платформи високих стандартів захисту. Більш того, комерційна модель платформи передбачає співпрацю з фармацевтичними компаніями, що може породжувати сумніви у повній незалежності аналізу даних.

HealthBoards найстаріша та найбільш відома серед вищезазначених інтернет-форумів, орієнтованих на обговорення медичних питань і досвіду пацієнтів, який був заснований ще на початку 2000-х років і з того часу зібрав багатомільйонну аудиторію [25]. Його структура побудована у вигляді сотень підфорумів, де кожен розділ відповідає певному напрямку медицини або конкретному захворюванню: від кардіології, ендокринології, неврології та імунології до загальної фармакології, побічних ефектів ліків і підтримки пацієнтів із хронічними захворюваннями.

Всі учасники форуму можуть ділитися власним досвідом прийому медикаментів, запитувати поради у людей із подібними симптомами чи діагнозами, а ще ж отримувати підтримку від спільноти. Це створює унікальну можливість не лише для пацієнтів, але й для дослідників та різних медичних фахівців, які можуть відслідковувати реальні відгуки, формувати уявлення про ефективність лікування у повсякденному житті та виявляти проблемні питання у застосуванні тих чи інших препаратів. Тому HealthBoards і стає важливим комунікативним майданчиком.

[Drug List](#) > **Haloperidol**

Discussions that mention Haloperidol.

View posts only related to

GENERAL HEALTH BOARD

[Hi, what is up with this lady?](#)

... causes are: typical or atypical antipsychotics such as haloperidol (Haldol) and droperidol, olanzapine (Zyprexa) SSRIs...

[Please Please Please Let Me Eat Grapefruit!!! :\)](#)

... erythromycin felodipine fentanyl finasteride haloperidol indinavir lercanidipine lidocaine lovastatin ...

DRUG INTERACTIONS / SIDE EFFECTS BOARD

[prednisone and hiccup type spasms](#)

... chlorpromazine (Thorazine), nifedipine (Procardia), haloperidol (Haldol), phenytoin (Dilantin), metoclopramide (Reglan...

PAIN MANAGEMENT BOARD

[Tramadol](#)

Рисунок 1.5 – Вибір теми для обговорення створених користувачами для пошуку медичних препаратів

Завдяки величезній кількості активних учасників форуму, відповіді зазвичай з'являються швидко, а накопичений архів робить платформу цінним ресурсом для майбутніх користувачів. Але слід зазначити і суттєві недоліки платформи [26].

Це підкреслює важливу закономірність: подібні форуми можуть слугувати як і дійсно чудовим інструментом для додаткового ознайомлення з досвідом різних пацієнтів і для психологічної підтримки від них, але не можуть і не повинні замінювати професійну медичну допомогу.

HealthBoards поєднує у своїй системі багато гідних переваг для колективного обміну знаннями з ризиками, властивими для будь-якої неофіційної медичної інтернет-спільноті, і ще може розглядатися як допоміжний, але не головний інструмент у сфері психологічного здоров'я. Платформа змушує користувачів самостійно фільтрувати обсяги інформації, що підвищує ймовірність помилкової самодіагностики або нераціонального вибору терапії.

Patient.info авторитетна англomовна платформ, яка поєднує в собі функціонал інформаційного медичного порталу та інтерактивного форуму для пацієнтів. Її головна ціль - надання перевіреної та зрозумілої для користувачів інформації про захворювання, симптоми, діагностику та сучасні методи лікування.

Patient Communities

Welcome!

Email

Never shown to the public

Username

Password

I confirm that I am over 18 years old.

I confirm that I am over 18 years old.

I am human

By registering, you agree to the [privacy policy](#) and [terms of service](#).

Sign Up

Log in with Discourse ID

Sign in with Google

Log in with LinkedIn

Sign in with Apple

Рисунок 1.6 – Меню реєстрації користувача

Patient.info вдало поєднує у своїй системі різноманітні переваги офіційного медичного ресурсу та живого форумного спілкування, виступаючи хорошим та надійним прикладом того, як інформаційна система може ефективно працювати одночасно на рівні доказової медицини та на рівні взаємної підтримки пацієнтів [27].

Платформа структурує свій контент таким чином, щоб професійні статті від лікарів були інтегровані безпосередньо в обговорення, створюючи для цього надійний контекст для користувацьких відгуків.

Таблиця 1.1 – Порівняння можливостей Інформаційних систем

Критерій / Система	Drugs.com	WebMD	PatientsLike Me	HealthBoards	Patient.info
Доступність	+	+	-	+	+
Якість інформації	-	+	+	+	+
Можливість комунікації	+	+	-	-	+
Актуальність даних	-	+	+	+	+
Зручність пошуку	-	+	+	+	+
Безпека та верифікація	-	+	+	+	+
Сильна сторона	Соц. аспект	Баланс досвіду і фактів	Науковість	Освіта й доступність	Поєднання переваг
Слабка сторона	Недостовірність	Може бути обмежена	Відсутність комунікації	Відсутність практики	Складність у реалізації

Аналіз можливостей інформаційних систем показав, що жодна з існуючих систем не є ідеальною. Drugs.com забезпечує широкий доступ і комунікацію, але страждає від низької достовірності та відсутності модерації. WebMD частково виправляє цю проблему, проте залишаються вузькоспеціалізованими та не завжди охоплює широку аудиторію. PatientsLikeMe та HealthBoards гарантують високу

якість і структурованість інформації, але позбавлені інтерактивності та соціальної взаємодії.

Саме тому потрібна комбінована система, яка поєднує науково перевірену базу знань із форумом для обговорень, виступає найбільш перспективним варіантом. Вона може об'єднати лікарів, пацієнтів і дослідників у межах одного ресурсу, Даючи комплексний і збалансований підхід до вивчення та використання психіатричних препаратів [28].

1.3 Підходи до вирішення задачі за темою дослідження

Підходи до вирішення задачі створення форум-орієнтованої інформаційної системи у сфері психіатричних препаратів передбачають у собі комплексне поєднання технологічних, інформаційних та організаційних рішень. Основна ідея полягає у розробці системи, що забезпечуватиме не лише зберігання і пошук даних про лікарські засоби, але й інтерактивну взаємодію між різними категоріями користувачів - лікарями, пацієнтами, дослідниками та представниками фармацевтичних компаній.

У частині форумної взаємодії потрібним підходом є організація тематичних гілок і модерації, що дозволить підтримувати порядок і захищати користувачів від поширення небезпечних порад. Також важливою задачею є захист персональних даних і забезпечення анонімності для користувачів, які обговорюють чутливі проблеми психіатричного характеру [29].

Для повної реалізації системи потрібно застосувати комплексний підхід, що охоплює декілька ключових напрямів проєктування. Функціональний підхід у собі передбачає логічне розділення системи на окремі модулі, такі як база даних медичних препаратів, форум для комунікації, аналітичний модуль та інтерфейс користувача.

Інформаційний аспект має полягати у структурованому зберіганні перевірених даних із чітким розмежуванням за критеріями достовірності та суб'єктивності.

Технологічна складова базується на використанні сучасних вебфреймворків, тоді як аналітичний підхід потребує впровадження алгоритмів обробки природної мови (NLP) для глибокого аналізу пацієнтських відгуків. Особливу увагу приділяють безпековому підходу, що спрямований на безумовне забезпечення анонімності користувачів та надійний захист їхніх персональних даних.

Можна сказати що підходи до вирішення задачі повинні враховувати не лише технічну, а й соціальну складову. Оскільки система орієнтована на роботу з психіатричними препаратами, вона повинна забезпечувати високий рівень достовірності поданої інформації та відрізняти перевірені медичні дані від особистих відгуків пацієнтів. Це дозволить користувачам легко орієнтуватися в потоці інформації й уникати можливих хибних інтерпретацій. Важливим підходом буде створення системи модерації та експертної перевірки, яка допоможе знизити ризик поширення небезпечних рекомендацій.

Також необхідно приділити увагу анонімності та етичним аспектам, бо багато пацієнтів, що обговорюють психіатричні теми, можуть відчувати певний соціальний тиск і потребують захищеного простору для спілкування.

1.4 Постановка задачі

Основна постановка задач у межах дослідження полягає в тому, щоб створити форум-орієнтовану інформаційну систему, яка дозволить об'єднати перевірені фармакологічні дані з інструментами для обміну досвідом між користувачами. Вирішення цього завдання передбачає кілька важливих етапів.

Спочатку потрібно визначити вимоги до системи, що забезпечить баланс між достовірністю даних та інтерактивністю комунікації.

Далі слід розробити концептуальну модель, яка включатиме базу даних із чітко структурованими відомостями про препарати, а також модуль форуму для користувацьких дискусій.

Та слід розробити концептуальну модель, яка включатиме базу даних із чітко структурованими відомостями про препарати, а також модуль форуму для користувацьких дискусій.

Окремим завданням буде впровадження механізмів модерації та анонімності, що дозволить уникнути поширення неправдивих або небезпечних рекомендацій від недобросовісних користувачів[30].

Важливою частиною проекту є й створення зручного інтерфейсу користувача, який буде зрозумілим і лікарям, і пацієнтам. Завершальним етапом постановки задач є апробація системи та оцінка її практичної користі, що дає змогу підтвердити ефективність інтеграції бази знань та форуму в єдине середовище.

Комплексне вирішення цих завдань дозволить сформувати сучасну та зручну платформу, здатну задовольнити потреби медичної спільноти та сприяти підвищенню якості лікування у сфері такій важливій та потрібній медичній сфері як психіатрії.

Для досягнення поставленої мети проекту потрібно вирішити наступні завдання: Проаналізувати існуючі інформаційні системи та онлайн-платформи, що функціонують у сфері медицини та фармакології.

Дати визначення ключовим недолікам наявних рішень і сформувати технічні вимоги до нової системи. Створити концептуальну модель інформаційної системи, адаптовану для специфіки роботи з даними про психіатричні препарати.

Спроекувати та реалізувати базу даних із можливістю зберігання структурованої інформації про психіатричні препарати.

Розробити інтерактивну платформу для обговорення про препарати та аналізу клінічних випадків. Додати туди ефективні механізми модерації контенту, верифікації фахових даних та захисту персональної інформації користувачів.

Створити адаптивний інтерфейс користувача, що відповідає потребам лікарів та пацієнтів, та провести апробацію розробленої системи та оцінити її ефективність у порівнянні з існуючими аналогічними платформами.

2 АРХІТЕКТУРА ТА ТЕХНОЛОГІЧНЕ ОБҐРУНТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ P-PSYFORUM

2.1 Формалізація та вибір базових компонентів форумно-орієнтованої інформаційної системи P-PsyForum

Спочатку потрібно визначитись з формалізацією та обґрунтуванням архітектурних рішень та технологічного стеку, необхідних для забезпечення надійної розробки, експлуатації та подальшого масштабування форумно-орієнтованої інформаційної системи P-PsyForum. Вихідні вимоги до стеку зосереджені на поєднанні транзакційної узгодженості, модульності коду, високої продуктивності REST API та готовності до горизонтального масштабування.

Для забезпечення відповідності сучасним стандартам веб-доступності та крос-браузерної сумісності, семантична розмітка HTML5 виступає не лише як каркас, а ще й як засіб побудови чіткої інформаційної ієрархії, це потрібно для форумних платформ з великим обсягом текстових даних.

Використання новітніх можливостей CSS3, таких як Flexbox та Grid Layout, дозволяє реалізувати складні адаптивні макети, які коректно відображають статистичні блоки та стрічки повідомлень на пристроях з різною роздільною здатністю екрана. Це створює надійний технологічний фундамент для впровадження динамічних елементів React.js, оскільки чітко структурований DOM-дерево спрощує маніпуляції зі станом сторінки та прискорює рендеринг інтерфейсу.

Крім базової стилізації, архітектурний підхід передбачає модульність CSS-правил, що запобігає конфліктам стилів при розширенні функціоналу системи. Використання новітніх методологій розробки дозволяє відокремити візуальне оформлення від бізнес-логіки, полегшуючи процес підтримки коду та його подальшу модернізацію.

Саме таке поєднання перевірених часом стандартів розмітки з інноваційними методами оформлення надає не лише естетичну привабливість системи P-

PsyForum, а ще й її високу технічну стабільність, це дає всі необхідні умови для коректної візуалізації результатів, які генерують метод структурно-статистичної агрегації.

Клієнтська реалізація інформаційної системи побудована на гібридному підході, вона дозволяє оптимізувати продуктивність та забезпечити необхідну гнучкість розробки. Процеси маркування та візуальної стилізації будуються на використанні стандарту HTML5 для формування семантичної структури сторінок та CSS3 для оформлення інтерфейсу [31].

Щоб забезпечити дотримання принципу DRY (Don't Repeat Yourself) та забезпечення високого рівня модульності стилів застосовується препроцесор SASS (SCSS). Препроцесор дозволяє організувати архітектуру стилів через використання змінних, міксинів та механізмів вкладеності, що спрощує подальшу підтримку коду. Динамічна логіка клієнтської частини поєднує використання стандартного Vanilla JavaScript для виконання простих операцій з DOM та фреймворку React.js для розробки складних компонентних інтерфейсів.

Забезпечення візуальної цілісності та адаптивності всього інтерфейсу в межах SASS-архітектури впроваджено систему глобальних дизайн-токенів, які стандартизують колірну палітру, типографіку та сітку компонентів.

Використання міксинів для медіа-запитів дозволяє реалізувати гнучкий адаптивний дизайн (Responsive Web Design), це гарантує коректне відображення аналітичних метрик та форумних гілок на пристроях із різною роздільною здатністю екрана.

Крім структурної організації, препроцесорна обробка стилів інтегрується в загальний конвеєр збірки проекту, воно забезпечує автоматичну мінімізацію CSS-файлів та підвищує загальну продуктивність рендерингу сторінок на стороні клієнта.

Застосування React.js дає можливості для легкої імплементації SPA-функціоналу (Single Page Application), зокрема в модулях керування статистикою та особистому кабінеті користувача, оскільки такий підхід мінімізує обчислювальні

витрати («оверхед») у статичних секціях та максимізує швидкість взаємодії у динамічних блоках.

Серверна частина системи, яка відповідає за реалізацію бізнес-логіки та обслуговування API-запитів, функціонує в середовищі виконання Node.js із використанням фреймворку Express.js.

Обрання даного технологічного рішення породжене можливістю реалізації повностекового (full-stack) підходу, що забезпечить уніфікацію мови програмування на клієнтському та серверному рівнях, спрощуючи їх підтримку та розвиток усієї системи.

Використання подієво-орієнтованої та неблокуючої моделі введення-виведення Node.js робить можливим для серверної частини ефективно обробляти численні одночасні з'єднання, без чого не зможе обійтися форум-орієнтована система із високою інтенсивністю оновлення контенту.

У рамках Express.js середовища реалізовано конвеєрну обробку запитів через механізм проміжного програмного забезпечення (middleware), це дає змогу ізолювати функціонал логуювання, парсингу вхідних JSON-структур та первинної фільтрації трафіку від основної логіки обробки даних.

Подібний підхід не лише підвищує стійкість системи до навантажень, а й забезпечує масштабованість архітектури, дозволяючи інтегрувати додаткові захисні фільтри або сервіси аналітики без необхідності рефакторингу ключових компонентів бізнес-логіки.

Архітектурно серверний модуль спроектовано за багат шаровою моделлю, яка включає контролерний шар для маршрутизації REST API та валідації запитів, сервісний шар для імплементації основної бізнес-логіки й методу агрегації, а також шар доступу до даних (DAO). Для забезпечення надійної ідентифікації та безпеки користувачів інтегровано механізми JWT-токенів (JSON Web Tokens), які є сучасним стандартом безсесійної аутентифікації в розподілених веб-системах.

Зберігання та управління даними (Data Layer) - для основного сховища даних обрано реляційну СУБД MySQL [32]. Критичні переваги: реляційна модель MySQL гарантує сувору ACID-відповідність (Atomicity, Consistency, Isolation,

Durability), є важливим для транзакційної цілісності обліку користувацької активності, зокрема при виконанні методу структурно-статистичної агрегації.

MySQL ідеально підходить для зберігання структурованих даних (профілі користувачів, пости, відповіді, довідкова інформація по препаратах) та виконання складних аналітичних і статистичних запитів.

Допоміжні сервіси та середовище розробки. Для забезпечення повноцінної експлуатації передбачена інтеграція з допоміжними сервісами (кешування, бекап/реплікація бази). Як основний інструмент розробки вибрано Visual Studio Code, що надасть ефективні засоби для відлагодження та супроводу JavaScript-проектів.

Процес розробки та супроводу системи базується на принципах безперервної інтеграції, де Visual Studio Code виступає центральним вузлом завдяки підтримці розширень для статичного аналізу коду (ESLint) та інструментів контролю версій (Git). Це дозволяє мінімізувати кількість синтаксичних та логічних помилок ще на етапі написання скриптів.

Паралельно із цим, для стабільної роботи продуктивного середовища використовуються механізми автоматизованого створення резервних копій MySQL, це дає можливість збереження бази постів та персональної статистики у разі технічних збоїв.

Додатково можна відзначити що, архітектура передбачає можливість впровадження In-memory сховищ (наприклад, Redis) для кешування найбільш часто запитуваних даних, що сильно знижує латентність системи при генерації статистичних звітів.

Аргументований вибір технологій, що поєднує узгодженість мови, стандартизовані фреймворки та надійну СУБД, формує міцну технологічну основу, яка є водночас гнучкою для розвитку та оптимізованою для розгортання.

Обрана архітектура та її компоненти, що забезпечують модульність та чіткий поділ відповідальності, відображені в Рисунку 2.1 - діаграмі взаємодії архітектурних компонентів інформаційної системи P-PsyForum.

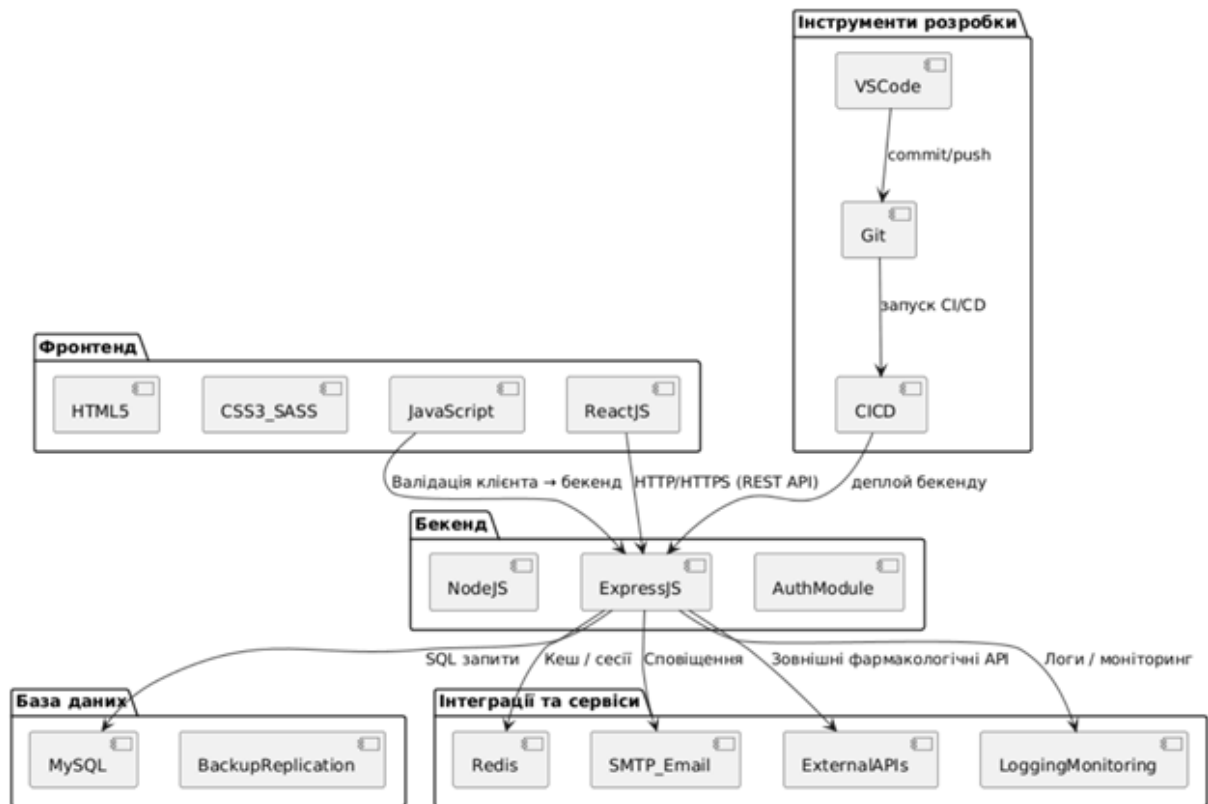


Рисунок 2.1 - Діаграма взаємодії архітектурних компонентів інформаційної системи

Забезпечуючи максимальну ефективність розробницького циклу та високої якості програмного коду, основним середовищем розробки (IDE/текстовим редактором) для проекту P-PsyForum вибрано Visual Studio Code (VS Code).

Вибір VS Code базується на його архітектурних перевагах та на функціональних можливостях, що відповідає потребам до сучасних full-stack JavaScript-застосувань [33].

Інтеграція середовища Visual Studio Code (VS Code) у процес розробки необхідна через низку його архітектурних переваг, серед яких ключове місце посідає оптимізація ресурсоспоживання та висока загальна продуктивність.

Дане середовище характеризується високою швидкістю виконання операцій та мінімальним навантаженням на апаратні ресурси системи, дозволяючи підтримувати стабільну продуктивність розробника при роботі з розгалуженими та складними кодовими базами, властивими для багатомодульних веб-систем.

Необхідною особливістю є нативна підтримка базового технологічного стеку проєкту, включаючи HTML5, CSS3, JavaScript та Node.js. Глибока інтеграція з препроцесорами, зокрема SASS/SCSS, дозволяє впроваджувати сучасні методології стилізації з використанням інструментів інтелектуального підсвічування синтаксису, автоматичного доповнення коду та миттєвої валідації структури файлів.

Ергономіка розробницького процесу значно підвищується завдяки концепції уніфікованого робочого простору, це реалізується через вбудований термінал (Integrated Terminal) [34]. Воно дозволяє виконувати повний спектр команд життєвого циклу програмного продукту - від менеджменту залежностей до компіляції клієнтських асетів та ініціалізації серверних процесів - без необхідності перемикання контексту між різними прикладними програмами.

Для глибшої інтеграції процесів розробки в межах VS Code застосовується механізм конфігурації багатопотокових завдань (Tasks), дозволяючи синхронізувати одночасний запуск клієнтського dev-сервера React та серверного процесу Node.js. Використання інтегрованого дебагера забезпечує можливість встановлення точок зупинки (breakpoints) у вихідному коді, дозволяючи проводити інспекцію стану об'єктів та потоків даних у реальному часі.

Це дійсно потрібно для налагодження складних ланцюгів асинхронних запитів та перевірки логіки методу агрегації. Крім цього, підтримка інтелектуального доповнення коду (IntelliSense) на базі аналізу типів даних суттєво пришвидшує написання SQL-запитів та взаємодію з API-ендпоінтами, забезпечуючи високу точність розробки при збереженні швидкості імплементації нових функцій.

Функціональні можливості середовища розширюються за рахунок розвиненої екосистеми додаткових модулів (Extensions), які забезпечують безшовну інтеграцію з системами контролю версій (Git), інструментами автоматичного форматування та лінтингу коду (Prettier, ESLint), а також засобами локального тестування.

Такий функціональний інструментарій дає необхідні умови для підготовки коду до етапів безперервної інтеграції та розгортання (CI/CD), гарантуючи високу якість програмної реалізації системи P-PsyForum.

Серверна частина всієї інформаційної системи P-PsyForum, розроблена на платформі Node.js із використанням фреймворку Express.js, функціонально забезпечує обробку вхідних запитів, контроль доступу, виконання бізнес-логіки та взаємодію із рівнем збереження даних.

Для досягнення високої модульності та масштабованості кодової бази архітектура сервера базується на основі принципу поділу відповідальності (Separation of Concerns) через багат шарову структуру.

Першим рівнем цієї структури є контролерний шар (Router/Controller), що виступає точкою входу для зовнішніх запитів. Він забезпечує прийом HTTP-запитів, первинну валідацію вхідних параметрів та делегування операцій до сервісного шару.

Цей шар відповідає за функціонування REST-ендпоінтів для виконання CRUD-операцій над ключовими сутностями системи, такими як теми, дописи та профілі користувачів, а також за адміністрування механізмів автентифікації на базі JWT-токенів.

Центральне місце в архітектурі займає сервісний шар (Business Logic Layer), який містить ядро бізнес-логіки системи. У рамках цього шару імплементовано метод структурно-статистичної агрегації, алгоритми перевірки прав доступу, логіку модерації та процеси верифікації фармакологічних довідкових даних.

Важливою особливістю є те, що цей шар інкапсулює алгоритми обробки, забезпечуючи їх незалежність від транспортних протоколів або специфіки сховища. Взаємодія безпосередньо із базою даних реалізована через шар доступу до даних (DAO/Repository Layer), який забезпечує необхідний рівень абстракції, інкапсулює SQL-операції та здійснює трансляцію результатів запитів до MySQL у доменні моделі.

Вибір реляційної СУБД MySQL зумовлений необхідністю забезпечення транзакційної цілісності та відповідності ACID-властивостям, воно має важливе

значення для стабільної роботи форумно-орієнтованої системи та достовірності статистики. На відміну від документно-орієнтованих рішень, реляційна модель MySQL демонструє високу ефективність при виконанні агрегаційних завдань, зокрема складних JOIN-запитів для генерації аналітичних звітів та відображення історії активності [35].

Для глибшого розуміння архітектурної переваги обраного рішення варто зазначити, що використання MySQL дозволяє впровадити суворі обмеження цілісності на рівні схеми даних, зокрема за допомогою зовнішніх ключів (Foreign keys) та каскадних операцій. Це запобігає появі «сирітських» записів (наприклад, коментарів без батьківського поста) та гарантує логічну єдність інформаційного простору форуму навіть при інтенсивному конкурентному доступі.

Впровадження індексів типу b-tree на важливих полях, таких як ідентифікатори користувачів та часові мітки публікацій, значно оптимізує швидкість виконання операцій читання, що треба для забезпечення високої продуктивності SPA-інтерфейсу при завантаженні великих гілок обговорень.

Додатковою перевагою є можливість використання тригерів та збережених процедур для автоматизації рутинних завдань обробки даних безпосередньо в середовищі СУБД. Вона дозволяє делегувати частину обчислювального навантаження з прикладного рівня Node.js на рівень збереження, забезпечуючи максимальну швидкість реакції системи на дії користувача.

Синергетичний ефект від поєднання хорошого двигуна InnoDB та оптимізованих запитів дозволяє підтримувати консистентність методу агрегації в режимі реального часу, створюючи надійне підґрунтя для подальшого масштабування системи та інтеграції додаткових аналітичних модулів.

СУБД забезпечує надійну реалізацію транзакційних сценаріїв, дозволяючи виконувати атомарні операції, такі як синхронне створення теми та оновлення індексів активності. MySQL також виступає оптимальним середовищем для зберігання структурованих даних, включаючи облікові записи, ієрархію форумного контенту та логі аудиту. Використання цих механізмів для побудови

матеріалізованих уявлень та реплікації додатково гарантує прискорення вибірки даних і високу відмовостійкість всієї інформаційної системи.

Обраний Node.js для серверної частини сайту повністю виправданий, оскільки його асинхронна, подієво-орієнтована архітектура здатна ефективно обслуговувати велику кількість одночасних з'єднань, це типова ситуація для веб-форумів. Переваги та компроміси, пов'язані з використанням Node.js, відображені у порівняльному аналізі:

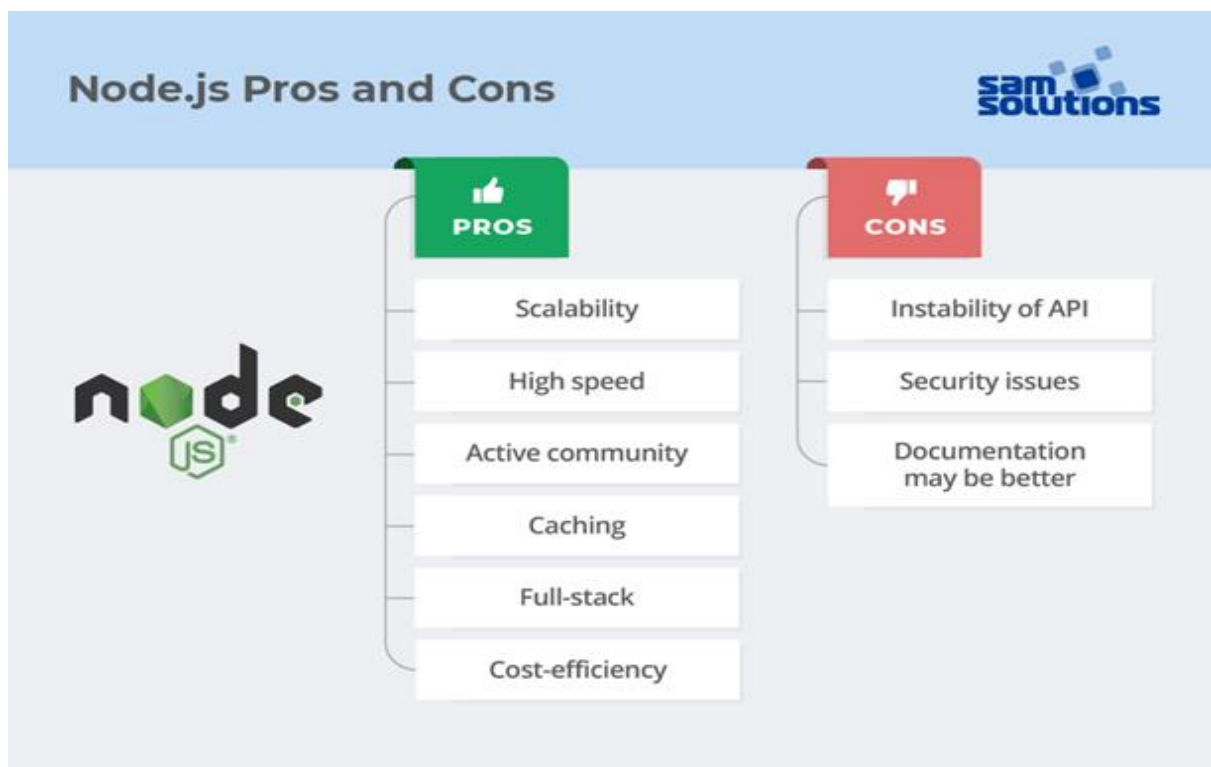


Рисунок 2.2 - Плюси та мінуси технології [Node.js](#)

Node.js створює оптимальне середовище для реалізації методу структурно-статистичної агрегації та ефективного обслуговування динамічних користувацьких інтерфейсів на React.js. Аргументований вибір інструментарію та архітектурних шарів закладає міцну основу для подальшої програмної імплементації, деталі якої будуть викладені у наступних розділах роботи.

2.2 Концептуальне проектування інформаційної системи. Обґрунтування компонентів клієнтської частини

Концептуальне проектування полягає у відображенні функціональних вимог системи на конкретні програмні та апаратні компоненти, які працюватимуть в рамках обраної трирівневої архітектури.

Проектування системи базується на фундаментальних принципах модульності та багат шарової архітектури, дозволяючи забезпечити ефективну інкапсуляцію бізнес-логіки та її ізоляцію від транспортного рівня та рівня збереження даних. Логічна модель інформаційної системи охоплює декілька взаємопов'язаних структурних рівнів, першим з яких є рівень представлення (Presentation Layer).

Даний рівень відповідає за візуалізацію даних, управління інтерактивною взаємодією з користувачем та реалізацію вимог до ергономіки інтерфейсу. Саме тут здійснюється динамічна побудова елементів управління та графічна інтерпретація інформаційних потоків, зокрема візуалізація статистичних метрик, сформованих на рівні додатка.

Центральним компонентом системи виступає рівень додатка (Application/Service Layer), який виконує роль ядра і забезпечує реалізацію основної бізнес-логіки, контроль доступу та процеси аналітичної обробки інформації. На цьому рівні імплементовано метод структурно-статистичної агрегації, функціонал якого полягає в атомарному оновленні метаданих користувача після кожної транзакції, такої як публікація допису або відповіді. Крім цього, рівень додатка координує процеси обробки запитів до бази даних, застосовує визначені правила валідації та здійснює комплексні авторизаційні перевірки.

Фундаментом системи є рівень даних (Data Layer), призначений для перманентного зберігання інформації, гарантування її цілісності та транзакційної узгодженості. У межах цього рівня моделюються всі ключові сутності системи, включаючи деталізовані профілі користувачів, ієрархічну структуру форумних повідомлень, журнали активності та довідкову інформацію щодо фармакологічних

препаратів. Також на рівні даних зберігаються агреговані статистичні показники, що забезпечує швидкий доступ до аналітичної інформації без необхідності повторних ресурсомістких обчислень [36-37].

Такий структурний поділ мінімізує зв'язаність компонентів та спрощує подальший розвиток системи. Модель проектування також включає визначення потоків даних між цими рівнями, механізми обробки асинхронних операцій та протоколи захисту інформації (зокрема, обробку авторизаційних токенів).

Структурне та функціональне розміщення програмних компонентів, що реалізують ці рівні взаємодії в системі P-PsyForum, відображено у Рисунку 2.3 - Діаграмі проектування інформаційної системи.

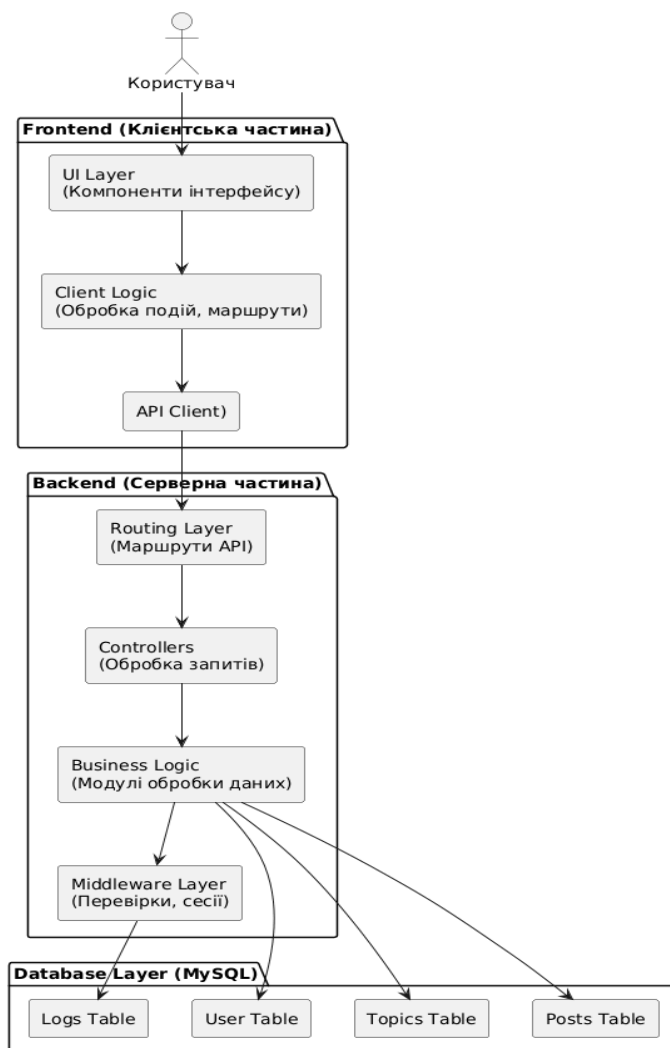


Рисунок 2.3 - Діаграма проектування інформаційної системи

Реалізація клієнтської частини системи P-PsyForum вимагає інструментарію, здатного забезпечити високу продуктивність, модульність та ефективне управління станом для таких складних динамічних модулів, як персональна статистика користувача. Після аналізу архітектурних вимог, для рівня представлення обраний фреймворк [React.js](#).

Вибір фреймворка React.js ґрунтується на парадигмі компонентного програмування, яка дозволяє декомпонувати інтерфейс користувача на незалежні, інкапсульовані та повторно використовувані UI-компоненти. Саме це значно підвищує супровідність коду та спрощує розробку складних функціональних блоків.

Використання декларативного підходу до опису інтерфейсів дозволяє правильно визначити стан усієї системи у будь-який момент часу, тому що це має суттєве значення для форумних платформ із великою кількістю інтерактивних елементів.

За допомогою односпрямованому потоку даних забезпечується передбачуваність роботи застосунку та спрощується процес відстеження помилок у складній логіці взаємодії користувача з базою психіатричних препаратів.

Основна перевага React.js полягає у механізмі Віртуального DOM (Virtual DOM), який оптимізує процеси рендерингу, мінімізуючи безпосередні маніпуляції з реальним DOM. Це забезпечує ефективне та швидке оновлення інтерфейсу лише у змінених вузлах, які оперують динамічними даними (наприклад, лічильниками активності у реальному часі) [38].

React.js забезпечує високу гнучкість завдяки своїй природі як бібліотеки, а не повноцінного фреймворку, дозволяючи інтегрувати найкращі рішення для маршрутизації та управління глобальним станом.

Це дає перевагу перед іншими популярними фреймворками у сфері кривої навчання, розміру спільноти та екосистемі доступних інструментів, що підтверджує його доцільність для довгострокового розвитку проєкту.

Таблиця 2.1 – Порівняння популярних фреймворків

Фреймворк	Продуктивність	Простота навчання	Масштабованість	Гнучкість	Спільнота та підтримка
React.js	+	+	+	+	+
Vue.js	+	+	+	+	+
Angular	+	–	+	–	+
Svelte	+	+	–	+	–
Next.js	+	–	+	+	+
Nuxt.js	+	–	+	+	+

Аналіз цих платформ демонструє, що кожна з них має свої власні унікальні переваги, які можуть бути корисними залежно від типу застосунку. React.js вирізняється гнучкістю та величезною спільнотою, що робить його універсальним рішенням для форумів, соціальних платформ і динамічних інтерфейсів.

Для рівня Сховища Даних інформаційної системи P-PsyForum обрано реляційну СУБД MySQL. Вибір реляційної моделі (RDBMS) поставив через необхідність суворої відповідності ACID-властивостям (Atomicity, Consistency, Isolation, Durability).

Ці властивості гарантують, що операції, пов'язані з обліком активності користувача та оновленням статистичних показників (зокрема, при виконанні методу структурно-статистичної агрегації), виконуються атомарно і не призводять до порушення цілісності даних.

Строга схема MySQL забезпечує надійні зв'язки між основними сутностями системи: користувачами, темами, постами, відповідями та довідковою фармакологічною інформацією.

Система MySQL додатково забезпечує ефективну підтримку складних SQL-запитів, включаючи операції об'єднання (JOIN) та агрегаційні функції, такі як COUNT та SUM. Використання цих інструментів дає можливість реалізувати

механізм динамічного розрахунку звань або рангів користувачів, що базується на об'єктивному аналізі їхньої сукупної активності в межах платформи.

На відміну від нереляційних (NoSQL) СУБД, які переважно фокусуються на гнучкості схеми та горизонтальному масштабуванні, MySQL пропонує незамінну транзакційну гарантію, що є пріоритетом для медико-соціальної інформаційної платформи.

Детальний порівняльний аналіз, що обґрунтовує переваги MySQL над іншими класами СУБД, враховуючи необхідність підтримки цілісного статистичного аналізу, представлено у Таблиці 2.2 - Порівняння архітектурних характеристик реляційних та нереляційних СУБД.

Таблиця 2.2 – Порівняння різних СУБД між собою

СУБД	Реляційна структура	Висока продуктивність	Гнучкість масштабування	Підтримка складних запитів	Простота використання	Підходить для форумів
MySQL	+	+	+	+	+	+
PostgreSQL	+	+/-	+	+	-	+
MongoDB	-	+	+	-	+	-
SQLite	+	-	-	+/-	+	-
Oracle DB	+	+	+	+	-	+
Microsoft SQL Server	+	+	+	+	-	+

В кінці можна зазначити те що, проведений аналіз, варто сказати, вибір MySQL як основної СУБД для форумно-орієнтованої інформаційної системи є найбільш раціональним та технічно правильним рішенням. Порівняння з іншими системами управління базами даних показало, що MySQL забезпечує оптимальний баланс між продуктивністю, стабільністю, простотою адміністрування та гнучкістю розширення, що особливо важливо для системи, яка передбачає активну взаємодію користувачів, зростання кількості повідомлень, тем та коментарів [39].

На етапі концептуального проектування важливим також є вибір мови програмування та середовища виконання для серверної частини (Backend), оскільки саме цей рівень забезпечує координацію між клієнтськими запитами та базою даних. Сучасний ринок пропонує декілька домінуючих технологій, кожна з яких має свої особливості обробки потоків даних та управління пам'яттю. Для системи P-PsyForum ключовими критеріями вибору є швидкість розробки, можливість обробки асинхронних подій (що характерно для форумів з активним оновленням) та наявність розвиненої екосистеми бібліотек.

Детальний аналіз серверних мов та платформ, що дозволив визначити Node.js як найбільш релевантне рішення для реалізації бізнес-логіки та методу агрегації, наведено у Таблиці 2.3.

Таблиця 2.3 – Порівняння серверних мов програмування один з одним

Технологія / Критерій	Node.js (JavaScript)	Python (Django/FastAPI)	PHP (Laravel)	Java (Spring)	Ruby on Rails
Швидкість розробки	+	+	+	-	+
Асинхронність (I/O)	+	+/-	-	+	-
Full-stack уніфікація	+	-	-	-	-

Кінець таблиці 2.3

Екосистема пакетів	+ (NPM)	+ (PyPI)	+ (Composer)	+ (Maven)	+ (Gems)
Продуктивність під навантаженням	+	-	+/-	+	-
Масштабованість	+	+	+/-	+	+/-

Як демонструє Таблиця 2.3, Node.js вигідно вирізняється концепцією уніфікації мови (JavaScript на обох рівнях системи), що значно прискорює ітераційну розробку та спрощує підтримку коду за рахунок спільного використання об'єктних моделей. Модель неблокуючого введення-виведення дозволяє підтримувати тисячі паралельних з'єднань, це потрібно для форумної платформи, де користувачі постійно генерують трафік через публікації та перегляди статистики.

Додатковим аспектом концептуального проектування є організація внутрішньої взаємодії модулів. На відміну від монолітних структур, обрана архітектура на базі Node.js дозволяє імплементувати сервіс-орієнтований підхід. Це означає, що логіка методу структурно-статистичної агрегації виділяється в окремий сервісний шар, який не залежить від протоколів передачі даних.

Впровадження об'єктно-реляційного відображення (ORM) або використання підготовлених SQL-запитів у середовищі Node.js дозволяє нівелювати ризики впровадження стороннього коду (SQL-ін'єкцій), що разом із властивостями ACID обраної СУБД MySQL створює багат шаровий захист інформаційних активів системи.

2.3 Висновки

Програмна реалізація інформаційної системи P-PsyForum базується на трирівневій архітектурі, що дає забезпечення та чіткий поділ відповідальності та

функціональну ізоляцію компонентів. Було обґрунтовано вибір Full-Stack JavaScript-парадигми, де ключову роль на серверній частині відіграє середовище виконання Node.js у комбінації з фреймворком Express.js.

Це рішення зумовлене потребою в асинхронному, подієво-орієнтованому вводу/виводу, що є оптимальним для ефективної обробки великої кількості одночасних з'єднань, характерних для веб-форумів, забезпечуючи високу пропускну здатність (throughput) та мінімальну затримку (latency).

Серверна структура реалізована за шаровою моделлю, що включає контролерний шар (обробка REST API та JWT-аутентифікація), сервісний шар (впровадження ключової бізнес-логіки та методу структурно-статистичної агрегації) та шар доступу до даних (DAO). Така архітектура гарантує модульність, спрощує супровід та тестування системи.

Концептуальне проектування, забезпечило інтеграцію найбільш ефективних технологічних рішень для рівня Представлення та рівня Даних. Для клієнтської частини обґрунтовано використання можливостей фреймворка React.js,

Архітектура React.js, заснована на компонентному моделюванні та механізмі Віртуального DOM (Virtual DOM), мінімізує ресурсномісткі операції з DOM, що дає змогу для високопродуктивної візуалізації статистичних даних у реальному часі на зміні статистики користувача.

Фундаментальний вибір Реляційної СУБД MySQL для рівня сховища даних прямо пов'язаний з необхідністю забезпечення ACID-властивостей та транзакційної цілісності. Ця вимога є беззаперечною для надійного та точного обліку активності користувачів і коректної роботи методу агрегації.

Вибрана комбінація технологій створює надійну основу для подальшої імплементації, що здатна ефективно керувати як комунікаційними, так і науково-аналітичними завданнями системи P-PsyForum. Завдяки чіткому розподілу обов'язків між фронтенд-компонентами та бекенд-сервісами, система зберігає високу швидкість обробки запитів навіть під час виконання складних процедур агрегації даних.

3 МЕТОД ТА АЛГОРИТМ СТРУКТУРНО-АНАЛІТИЧНОЇ ОБРОБКИ ДАНИХ ІНФОРМАЦІЙНОЇ СИСТЕМИ

3.1 Метод моніторингу інформаційної чистоти та оцінки довіри контенту

Метод структурної організації та обробки контенту в інформаційній системі P-PsyForum являє собою послідовність кроків, спрямовану на забезпечення повного циклу життєдіяльності інформації: від її створення користувачем до збереження у базі даних, подальшої агрегації та відображення у вигляді персональної статистики.

Цей процес базується на використанні реляційної моделі даних MySQL та модульності серверної логіки Express.js. Процедура ініціалізації та введення даних передбачає фіксацію користувацьких дій, таких як реєстрація, авторизація та створення контенту, що збираються через форми, обробляються клієнтським інтерфейсом React.js та передаються на сервер.

На етапі первинної обробки та валідації здійснюється перевірка коректності введених значень на стороні сервера для захисту бази даних. Наступна систематизація та класифікація дозволяють ідентифікувати типи контенту, присвоюючи сутностям необхідні метадані для організації логічних зв'язків.

Важливим для всієї системи є створення та агрегація статистичних метрик, що забезпечує автоматичний розрахунок і постійне оновлення показників активності користувача. Надійне зберігання цих даних, включаючи облікові записи та історію обговорень, реалізується у реляційній базі даних із забезпеченням цілісності зв'язків [40]. Паралельно на рівні Express.js виконується контроль доступу та історичний облік, що дозволяє проводити ретроспективний аналіз активності.

Візуалізація отриманих звітів здійснюється через динамічні компоненти React.js, надаючи користувачеві зручний інтерфейс для управління профілем та перегляду власної статистики. Цілісність системи підтримується логікою

синхронного видалення пов'язаних даних, що гарантує актуальність динамічної статистики.

Технічна реалізація довідкового каталогу та забезпечення зовнішньої крос-референсної підтримки базуються на використанні HTML/CSS для створення структурованої навігації до офіційних фармацевтичних ресурсів. В основі архітектури даних лежить принцип реляційної асоціації, реалізований через механізм зовнішніх ключів у MySQL, що унеможливорює появу неузгоджених записів. Для уникнення дублювання інформації застосовується принцип мінімізації надмірності через нормалізацію бази даних.

Швидкість відгуку системи при високих навантаженнях забезпечується методом асинхронної обробки запитів Node.js, тоді як автоматизована контрольна валідація на сервері запобігає некоректному збереженню інформації. Нарешті, метод ієрархічної стильової організації SASS та метод асинхронної візуалізації через AJAX-запити в React.js гарантують цілісність UI/UX дизайну та динамічність взаємодії користувача з інтерфейсом системи [41].

Реалізація форум-орієнтованої інформаційної системи P-PsyForum, яка інтегрує методи структурної організації контенту та статистичного обліку, дозволяє створити потужний інструмент для систематизації досвіду використання різноманітних психіатричних препаратів.

Для поглиблення аналізу треба підкреслити, як саме синергія обраних технологій трансформується у функціональні переваги для кінцевого користувача. Вибір компонентно-орієнтованого підходу React.js дозволяє побудувати інтерфейс, який адаптується до специфічних потреб пацієнтів та спеціалістів, забезпечуючи високу читабельність тексту та зручність навігації у складних гілках обговорень.

Для медико-соціальної платформи, де точність сприйняття інформації про дозування чи побічні ефекти препаратів це має першочергове значення. На рівні серверної архітектури Express.js виконує роль гнучкого регулятора, що забезпечує безперебійну роботу методу структурно-статистичної агрегації навіть при різких сплесках активності. Використання асинхронних обробників подій дозволяє

системі одночасно підтримувати сотні активних сесій, не втрачаючи при цьому жодного пакету даних, що стосується користувацького досвіду.

Це створює надійний фундамент для формування «бази знань», де кожен пост стає частиною верифікованої статистики. Таким чином, технологічний стек не просто підтримує роботу сайту, а виступає гарантом того, що зібрана інформація буде коректно оброблена, збережена та представлена у вигляді зручних аналітичних зрізів.

Особливе місце в екосистемі P-PsyForum займає СУБД MySQL, яка завдяки своїй реляційній природі дозволяє будувати складні звіти про динаміку обговорень певних груп препаратів. Це дає змогу виявляти закономірності у відгуках користувачів, які можуть бути корисними для подальшого статистичного аналізу. Впровадження індексів та оптимізованих запитів гарантує, що пошук необхідної фармакологічної інформації або перегляд історії активності відбуватиметься за частки секунди.

За результатом поєднання цих технологій утворюється цілісне інформаційне середовище, яке забезпечує не лише комунікацію, а й глибоку систематизацію медичного досвіду на принципах достовірності та транзакційної цілісності.

Це сприяє підвищенню інформаційної обізнаності людей та покращенню якості обміну досвідом між користувачами форуму у чутливій медичній предметній області. Використання технологій, таких як React.js, Express.js та MySQL, дало створити дійсно потрібну та корисну екосистему, що підтримує спільноту користувачів.

Створення та подальший розвиток інформаційної системи P-PsyForum зумовлені низкою важливих соціальних та практичних чинників. У контексті збереження психічного здоров'я та прийняття інформованих рішень, система виступає інструментом протидії дезінформації та ризикам самолікування.

Надаючи платформу для обміну досвідом, вона сприяє більш обґрунтованому підходу користувачів до процесу лікування, який здійснюється у координації з фахівцями. Важливою функцією системи є запобігання інформаційним ризикам: завдяки структурній організації та механізмам валідації даних стає можливим

своєчасне виявлення та блокування недостовірних або потенційно небезпечних відомостей, що мінімізує вплив некваліфікованих порад у публічному просторі.

Якість обміну досвідом покращується через систематизацію тематичних розділів та можливість верифікації активності авторів за допомогою статистичних метрик, що забезпечує високий рівень достовірності контенту [42]. Окрім комунікативної функції, система має значний аналітичний потенціал. Завдяки механізмам історичного обліку даних з'являється можливість відстежувати довгострокові тенденції у застосуванні препаратів, побічні ефекти та зміни у схемах лікування, що становить наукову цінність для подальших досліджень.

Разом із цим, інструменти прогнозування інформаційних трендів дозволяють оперативно ідентифікувати зростання інтересу до певних медичних проблем або нових терапевтичних методів.

Етичний та регуляторний аспекти реалізуються через логіку контролю доступу та модераційні алгоритми, які гарантують дотримання стандартів спілкування у надчутливій сфері психіатрії. Довідковий характер системи підкреслюється наявністю верифікованих посилань на аптечні агрегатори, що забезпечує прозору навігацію до офіційних ресурсів.

Одночасно із цим, збір персональної статистики стимулює користувачів до відповідальної та конструктивної участі у житті спільноти, створюючи безпечне анонімне середовище для психологічної підтримки.

Технічний базис системи, заснований на реляційному управлінні даними MySQL та серверному контролі Express.js, гарантує захист інформаційної цілісності та запобігає несанкціонованим змінам контенту. У перспективі такий структурований формат накопичення даних може стати цінним ресурсом для практикуючих лікарів та дослідників, дозволяючи вивчати реальний клінічний досвід пацієнтів поза межами контрольованих досліджень, що значно підвищує рівень обізнаності та медичної освіти суспільства [43].

P-PsyForum є дійсно хорошим інструментом для забезпечення структурованого та надійного обміну інформацією між користувачами у сфері

психіатричних препаратів, що сприяє інформованості, підтримці спільноти та підвищенню якості життя її учасників.

Ефективність розробленої інформаційної системи P-PsyForum визначається її здатністю вирішувати виявлені проблеми предметної області - фрагментацію даних, ризик поширення неперевіреної та шкідливої інформації та відсутність єдиного структурованого середовища для обміну досвідом.

Мета полягає у розробці та впровадженні форуму орієнтованої інформаційної системи з використанням сучасного технологічного стеку, яка забезпечує структурну організацію, систематизацію та статистичний облік користувацького контенту у сфері психіатричних препаратів, створюючи справді надійну платформу для обміну особистим досвідом та отримання довідкової інформації від інших користувачів.

Об'єкти проектування розробленої інформаційної системи P-PsyForum охоплюють сукупність сутностей та процесів, що підлягають безпосередньому управлінню, обробці та аналітичній оцінці. Першочерговим об'єктом є контент, що генерується користувачами (UGC), який включає тексти опублікованих тем, повідомлень та коментарів, виступаючи основним джерелом інформації в системі.

Паралельно до цього із об'єктами проектування виступають профілі та активність користувачів, що охоплюють реєстраційні дані та всі зафіксовані дії в межах платформи. Особливе значення мають статистичні та агреговані метрики, які відображають розрахункові показники активності та зв'язки між сутностями. Крім того, до об'єктів проектування відносяться зовнішні крос-референсні джерела, інтегровані в інтерфейс для довідкової підтримки, а також системні процеси, що забезпечують валідацію даних, контроль цілісності бази даних та сесійне управління [44-45].

Враховуючи ключову роль персональної статистики як доказу функціональності системи, було розроблено метод структурно-статистичної агрегації та трансформації даних. Цей метод формалізує процес перетворення простих користувацьких транзакцій на аналітичні метрики, демонструючи інтелектуальну обробку інформації.

На початковому етапі планування та формалізації визначаються критерії активності, що підлягають обліку, здійснюється структурне проектування статистичних сутностей у MySQL та формулюються агрегуючі SQL-запити для збору даних у реальному часі.

Процес збору даних та ініціації обліку на базі Express.js передбачає перехоплення транзакцій після успішного створення контенту, екстракцію метаданих автора та запуск процедури оновлення статистичних лічильників. транзакційна агрегація та контроль цілісності в MySQL забезпечують автоматичне інкрементування відповідних полів бази даних лише за умови успішного збереження контенту в основній таблиці, що гарантує реляційну цілісність системи.

Заключний етап методу включає візуалізацію статистики та її динамічну презентацію за допомогою React.js. Сервер генерує статистичний звіт у форматі JSON через API, після чого компоненти клієнтської частини трансформують отримані дані у візуальні елементи, такі як графіки та таблиці.

За допомогою реактивної природи фреймворку забезпечується миттєве оновлення статистики на сторінці користувача без необхідності її перезавантаження, що підкреслює високу ефективність та інтерактивність розробленої інформаційної системи.

Метод структурно-статистичної агрегації та трансформації даних розроблений так, щоб ідеально складатися з кількох послідовних етапів - від планування метрик, перехоплення користувацької транзакції, забезпечення цілісності запису в MySQL до агрегації та динамічної візуалізації результатів на React.js.

Таблиця 3.1 наочно демонструє, яким саме чином технологічний стек системи (React.js, Express.js, MySQL) розподіляє ключові завдання між собою та забезпечує безперервний життєвий цикл користувацької транзакції, перетворюючи сирі дані на необхідні аналітичні показники.

Таблиця 3.1 - Зіставлення етапів методу структурно-статистичної агрегації та технологій

Етап Методу	Ключове Завдання	Відповідальна Технологія	Очікуваний Результат / Продукт
1. Планування та Формалізація	Визначення метрик та структури даних.	MySQL	Створення полів \$posts_count\$, \$comments_count\$ у таблиці \$Users\$.
2. Збір та Ініціація Обліку	Перехоплення користувацької транзакції та валідація.	Express.js	Верифікований \$HTTP POST\$-запит та виділений \$user_id\$.
3. Транзакційна Агрегація та Контроль	Безпосереднє оновлення лічильників у базі даних.	MySQL	Інкрементування статистичного поля (\$posts_count\$) та підтвердження цілісності зв'язку.
4. Візуалізація та Динамічна Презентація	Формування та відображення кінцевого аналітичного звіту.	React.js	Динамічне оновлення профілю користувача та відображення актуальної статистики.

Така формалізація дозволяє Інформаційній системи P-PsyForum оперативно трансформувати сирі користувацькі дані в потрібні для системи аналітичні метрики (такі як наприклад, персональна статистика користувача), оцінювати рівень

активності користувачів та забезпечувати високу структурну організованість контенту.

Впровадження цього потрібного для всієї системи методу гарантує дійсно ефективний збір даних про активність користувачів форуму, їхню взаємодію з всією системою та контентом що на ній знаходиться, моніторинг цілісності даних у реальному часі, а також забезпечує високу реактивність інтерфейсу при відображенні персональної аналітики та звітів [46].

3.2 Алгоритм життєвого циклу користувацької транзакції та обліку активності

Функціонування форум-орієнтованої інформаційної системи базується на чіткому алгоритмі, який описує повний цикл роботи з даними, починаючи від ініціації користувачем (наприклад, написання посту) до кінцевої візуалізації результатів (оновлення статистики). Цей алгоритм забезпечує цілісність даних та коректність статистичного обліку.

Після завершення етапу ініціації, управління передається на серверний рівень, де вступає в дію механізм проміжної обробки. Тут алгоритм передбачає обов'язкову стадію синтаксичного аналізу вхідних пакетів даних.

Валідація на боці Express.js дозволяє відсіяти некоректні запити (наприклад, порожні повідомлення або спроби несанкціонованого доступу) ще до моменту звернення до бази даних. Це має важливу цінність для збереження ресурсів системи, оскільки подальший метод структурно-статистичної агрегації потребує гарантовано чистих вхідних параметрів для коректного інкрементування лічильників.

Центральним компонентом алгоритму є забезпечення атомарності операції. Коли сервер підтверджує успішне збереження тексту посту в MySQL, алгоритм автоматично активує тригерний механізм оновлення. Замість того, щоб щоразу перераховувати всі повідомлення користувача в базі даних (що було б

неефективним при великих обсягах даних), метод структурно-статистичної агрегації виконує інкрементальне оновлення конкретного поля в таблиці.

Це дозволяє системі підтримувати актуальний стан статистики в режимі реального часу, забезпечуючи миттєвий зворотний зв'язок для користувача через React-інтерфейс, який динамічно підтягує оновлені значення без повного перезавантаження сторінки.

Така послідовність кроків гарантує, що кожна дія користувача проходить через суворий фільтр бізнес-логіки, перетворюючись на достовірну статистичну одиницю.

Це створює замкнений цикл обробки інформації, де кожен елемент контенту стає частиною великої аналітичної бази, що служить важливим аспектом для форум-орієнтованої системи, спрямованої на моніторинг активності учасників.

Алгоритм життєвого циклу користувацької транзакції в системі P-PsyForum являє собою цілісний процес обробки інформації, що дає можливість на трансформацію вводу користувача у структуровані дані та аналітичні метрики. Початковим етапом є ініціація, збір та передача первинних даних, де користувач взаємодіє з інтерфейсом React.js для створення контенту. Клієнтська частина автоматично формує запит на сервер, доповнюючи його необхідними метаданими, такими як ідентифікатор автора, часові мітки та авторизаційні дані сесії.

Наступний етап включає серверну валідацію та контроль доступу засобами логіки Express.js. На цьому рівні вхідний запит проходить критичну перевірку на коректність формату даних та наявність прав доступу. У разі виявлення невідповідностей процес переривається, що запобігає пошкодженню інформаційної структури. При успішному проходженні перевірок реалізується транзакційне зберігання контенту у відповідних таблицях MySQL. Використання механізмів реляційної цілісності та зовнішніх ключів гарантує нерозривний зв'язок між користувачем та створеними ним сутностями.

Для глибшого наукового базування самого процесу варто також деталізувати механізм переходу від збереження даних до їх аналітичної обробки сервером. Це

дозволяє продемонструвати, як саме забезпечується висока швидкість відгуку системи при збереженні точності показників.

Необхідним аспектом серверної фази є використання патерну «Observer» або механізму «Hooks», саме це й дозволяє відокремити процес запису основного контенту від допоміжних обчислень. Коли транзакція в MySQL отримує статус завершеної, система ініціює фоновий процес, який виконує метод структурно-статистичної агрегації.

Це мінімізує час очікування для користувача, оскільки основний HTTP-запит може бути завершений одразу після успішного запису тексту поста, тоді як оновлення складних аналітичних зрізів відбувається паралельно. Такий підхід дозволяє уникнути блокування основного потоку виконання та забезпечує плавність роботи інтерфейсу навіть при великій кількості одночасних публікацій.

На фінальній стадії візуалізації особлива увага приділяється реактивному оновленню стану додатка (State Management). Отримавши JSON-відповідь від сервера з новим значенням лічильників, React.js виконує процедуру порівняння віртуального DOM, оновлюючи лише ті фрагменти сторінки, які відображають статистику активності та звання користувача.

Для забезпечення стабільності цього процесу на клієнтському рівні використовується архітектурний підхід односпрямованого потоку даних. Це означає, що будь-яка зміна статистики, ініційована методом структурно-статистичної агрегації, проходить через єдину точку входу в стані додатка, що виключає появу візуальних артефактів або застарілих даних на екрані. Завдяки використанню високорівневих хуків (Hooks), таких як `useEffect` та `useState`, система автоматично синхронізує локальний стан браузера з останніми змінами в базі даних MySQL відразу після отримання підтвердження від API.

Такий механізм дозволяє підтримувати високу щільність інформації на сторінці форуму без шкоди для продуктивності, оскільки React.js мінімізує кількість операцій із реальним деревом об'єктів браузера, фокусуючись лише на змінених аналітичних показниках.

Це реалізує принцип динамічної презентації, де користувач отримує візуальне підтвердження своєї дії в межах декількох мілісекунд. Таким чином, метод структурно-статистичної агрегації перетворюється з абстрактної математичної моделі на функціональний інструмент стимулювання соціальної взаємодії всередині платформи, створюючи цілісне середовище для обміну досвідом.

Відразу після успішної фіксації даних у сховищі активується фаза агрегації та актуалізації статистичних метрик. Вона передбачає пряме інкрементування лічильників у профілі користувача, що забезпечує миттєве відображення результатів роботи методу агрегації. Завершується цикл стадіями сповіщення, візуалізації та формування звітів. Сервер надсилає клієнту підтвердження виконання операції, після чого компоненти React.js динамічно оновлюють інтерфейс та статистичні показники без перезавантаження сторінки.

Створений алгоритм життєвого циклу транзакції дозволяє здійснювати структурний збір даних, забезпечувати надійне зберігання інформації, автоматично вести статистичний облік активності користувачів та забезпечувати високу швидкість взаємодії в реальному часі.

Завдяки автоматизованій обробці кожної взаємодії, система трансформує розрізнені коментарі у структуровану базу знань, де кожен користувач має можливість миттєво оцінити актуальність інформації та авторитетність порад інших учасників.

Це і перетворює P-PsyForum з простого інструменту комунікації на інтелектуальне середовище цифрової підтримки, яке мінімізує інформаційний шум і допомагає користувачам фокусуватися на найважливішому - пошуку ефективних шляхів до одужання на основі перевіреного колективного досвіду

Основні кроки алгоритму, що включають у себе перевірку даних, зберігання та фіналізацію, зображені у вигляді схеми на Рисунку 3.1.



Рисунок 3.1 - Схема роботи алгоритму життєвого циклу користувацької транзакції

Така графічна модель візуалізує усю логіку управління користувацькими даними, починаючи від ініціації запиту клієнтом і завершуючи динамічним оновленням інтерфейсу або обробкою помилки.

Візуалізація алгоритму є необхідністю для демонстрації архітектурної послідовності дій, які забезпечують реалізацію всіх описаних методів контролю цілісності та обліку активності.

Аналіз представленої схеми дозволяє виділити важливі точки синхронізації між асинхронною природою клієнтських запитів та детермінованою логікою серверної обробки. Кожен вузол алгоритму функціонує як окремий фільтр цілісності: якщо на етапі валідації виявляється невідповідність структури даних (наприклад, некоректний тип контенту або спроба маніпуляції ідентифікаторами), система автоматично повертає керування до блоку обробки помилок.

Це не лише захищає базу даних від сміттєвих записів, а й надає користувачеві миттєвий зворотний зв'язок через React-компоненти сповіщень, що є важливим аспектом ергономіки інтерфейсу.

Особливе значення в алгоритмі має розгалуження після успішного збереження транзакції. Паралельно з відправкою підтвердження клієнту (HTTP 201 Created), внутрішній механізм ініціює метод структурно-статистичної агрегації. Такий розподіл завдань дозволяє відокремити важку обчислювальну логіку від процесу взаємодії з користувачем.

Оскільки метод структурно-статистичної агрегації оперує безпосередньо реляційними зв'язками в MySQL, алгоритм гарантує, що інкрементовані лічильники активності (пости, коментарі, рівень репутації) завжди відповідають фактичній кількості об'єктів у системі. Це виключає ризик розсинхронізації даних, який часто виникає в системах без суворого транзакційного контролю.

Завершальна стадія алгоритму, що відповідає за візуалізацію, демонструє переваги реактивного підходу. Отримавши пакет оновлених аналітичних даних, клієнтський додаток виконує ре-рендеринг лише необхідних вузлів інтерфейсу.

Це створює ефект «живої системи», де користувач бачить зміни у своєму профілі або загальних рейтингах платформи без очікування повного оновлення сторінки.

Тим самим, алгоритм життєвого циклу користувацької транзакції не просто описує технічний шлях даних, а виступає програмним каркасом, який забезпечує динамічність, безпеку та аналітичну точність інформаційної системи P-PsyForum.

3.3 Висновки

У третьому розділі кваліфікаційної роботи детально розглянуто та розроблено методологічну базу та алгоритмічне забезпечення для функціонування Форум-Орієнтованої Інформаційної Системи P-PsyForum.

Що включає цілий комплекс різних методів структурної організації та статистичного обліку, які забезпечують категоризацію контенту, та методи забезпечення цілісності та сесійної безпеки, які захищають дані користувачів за допомогою хешування та контролю доступу (Express.js / MySQL) [47].

Важливим елементом, який забезпечує коректну функціональність всієї системи, є розроблений метод структурно-статистичної агрегації та трансформації даних. Такий метод формалізує процес перетворення сирих користувацьких транзакцій на цінні аналітичні метрики (статистику активності), які необхідні для демонстрації надійності та функціональності платформи.

Для повного опису логіки роботи інформаційної системи було розроблено Алгоритм життєвого циклу користувацької транзакції. Цей алгоритм описує послідовність дій від ініціації користувачем (створення посту на форумі) до кінцевої візуалізації результатів, включаючи валідацію, зберігання даних та паралельне оновлення статистики.

В основі із запропонованого алгоритму лежить подійно-орієнтована модель, яка дозволяє ізолювати потрібні процеси запису в базу даних від важких операцій аналітичної обробки.

За допомогою впровадженню проміжних шарів валідації (middleware) на рівні сервера, система здійснює попередню перевірку вхідних токенів та структури об'єктів ще до моменту їх потрапляння до ядра системи. Це мінімізує навантаження на інфраструктуру та запобігає появі небажаних помилок при одночасних запитах від багатьох учасників спільноти.

Після чого сам алгоритм забезпечує структурний збір, надійне зберігання інформації користувачів та навіть автоматичний облік активності, що є найважливішим для взаємодії з ним у реальному часі.

Реалізований підхід перетворює кожен дію користувача на структурований елемент бази знань, забезпечуючи пацієнтам відчуття безперервної підтримки та впевненості у точності представлених даних, що є ключовим фактором у процесі інформаційної допомоги при виборі та корекції стратегії лікування.

4 ФОРУМ-ОРІЄНТОВАНА ІНФОРМАЦІЙНА СИСТЕМА 3 ВИКОРИСТАННЯ ПСИХІАТРИЧНИХ ПРЕПАРАТІВ

4.1 Діаграма архітектури та основні компоненти інформаційної системи

Інформаційна система P-PsyForum розроблена як форум-орієнтована веб-платформа з чітко визначеною трирівневою архітектурою (Triple-Tier Architecture). Цей підхід забезпечує високу масштабованість, розділення відповідальності між компонентами та надійність.

Для глибшого розкриття архітектурного потенціалу системи необхідно деталізувати логіку взаємодії між цими рівнями, оскільки саме на стику технологій забезпечується стабільність форумного середовища.

Проміжним ланцюгом, що забезпечує цілісність системи, виступає рівень бізнес-логіки (Application Tier), побудований на базі Node.js. Цей рівень не просто транслює дані, а виконує роль інтелектуального фільтра та координатора.

На відміну від традиційних монолітних систем, така архітектура дозволяє паралельно обробляти запити на читання контенту та складні операції запису, що ініціюють метод структурно-статистичної агрегації.

Завдяки використанню REST API як стандарту комунікації, клієнтська частина залишається повністю незалежною від внутрішньої структури бази даних, що дозволяє проводити оновлення інтерфейсу без зупинки роботи всього сервісу.

Важливою перевагою обраної трирівневої моделі є можливість впровадження спеціалізованих механізмів обробки помилок на кожному етапі. Наприклад, якщо рівень сховища даних (Data Tier) тимчасово недоступний, серверний рівень може коректно обробити цю ситуацію та надіслати відповідне сповіщення на клієнтську частину, запобігаючи «падінню» всього додатка.

Такий підхід гарантує високу відмовостійкість системи P-PsyForum, що є критичним для платформ, де користувачі шукають оперативну підтримку та інформацію про препарати.

Основні архітектурні рівні системи P-PsyForum побудовані за принципом класичної трирівневої моделі, що забезпечує чіткий поділ функціональних обов'язків між компонентами. Клієнтський рівень (Frontend) представлений сучасним односторінковим додатком (SPA), реалізованим за допомогою бібліотеки React.js.

Цей рівень повністю відповідає за формування користувацького інтерфейсу, обробку вхідних форм та забезпечення високої реактивності системи. Саме тут відбувається фінальна візуалізація статистичних метрик, отриманих у результаті роботи методу структурно-статистичної агрегації, що дозволяє користувачу взаємодіяти з даними без зайвих затримок.

Для забезпечення високої продуктивності клієнтського рівня використовується концепція компонентно-орієнтованого проектування, де кожен елемент інтерфейсу (наприклад, картка користувача або блок аналітики) є незалежною одиницею коду.

Це дозволяє реалізувати механізм «лінивого завантаження» (Lazy loading), коли важливим є контент форуму рендериться першочергово, а допоміжні статистичні віджети підвантажуються асинхронно.

Завдяки такому підходу мінімізується початковий час завантаження сторінки, що безпосередньо впливає на показники утримання користувачів. Окрім того, клієнтський рівень бере на себе частину обчислювальної логіки щодо сортування та фільтрації отриманих від API даних, що знижує навантаження на центральний процесор сервера та забезпечує миттєву реакцію інтерфейсу на дії клієнта.

Взаємодія між клієнтським та серверним рівнями базується на принципах а stateless-архітектури, де кожен запит містить у собі всю необхідну інформацію для його успішної ідентифікації та обробки. Серверний рівень, функціонуючи як оркестратор бізнес-логіки, використовує систему проміжних обробників (Middleware) для забезпечення наскрізної безпеки.

Це все дозволяє відокремити технічні задачі, такі як парсинг вхідних JSON-структур або стиснення відповідей (Gzip), від основної логіки, яку реалізує метод

структурно-статистичної агрегації. Така модульність серверного рівня гарантує, що будь-яке розширення функціоналу системи, наприклад, додавання нових аналітичних модулів або інтеграція зовнішніх сервісів, не вплине на стабільність існуючих маршрутів та механізмів обробки даних.

Серверний рівень (Backend / API), розроблений на базі технологічного стеку Node.js із використанням фреймворку Express.js, виступає проміжним шаром системи. Він забезпечує стабільну маршрутизацію та обробку HTTP-запитів, виконує сувору валідацію даних і контроль доступу через механізми автентифікації та авторизації.

Важливою функцією цього рівня є саме програмна імплементація алгоритму життєвого циклу користувачької транзакції, зокрема ініціація процесів оновлення статистики безпосередньо після підтвердження збереження контенту.

Фундаментом всієї системи є рівень даних (Database), реалізований на базі реляційної СУБД MySQL. Даний рівень призначений для надійного та структурованого зберігання всієї інформації проекту, включаючи облікові записи користувачів, ієрархію форумних дописів та лічильники активності.

Використання реляційної моделі є засадничим для підтримки цілісності методу агрегації, оскільки дозволяє підтримувати суворі зв'язки між сутностями та гарантувати транзакційну узгодженість даних у масштабах всієї інформаційної системи [48].

Ця архітектура дозволяє системі ефективно обробляти транзакції, забезпечувати безпеку даних та надавати користувачеві швидкий, інтерактивний інтерфейс.

Загальну логіку взаємодії між базою даних, серверною частиною та інтерфейсом користувача наочно представлено у структурі архітектури форум-орієнтованої інформаційної системи на рисунку 4.1.

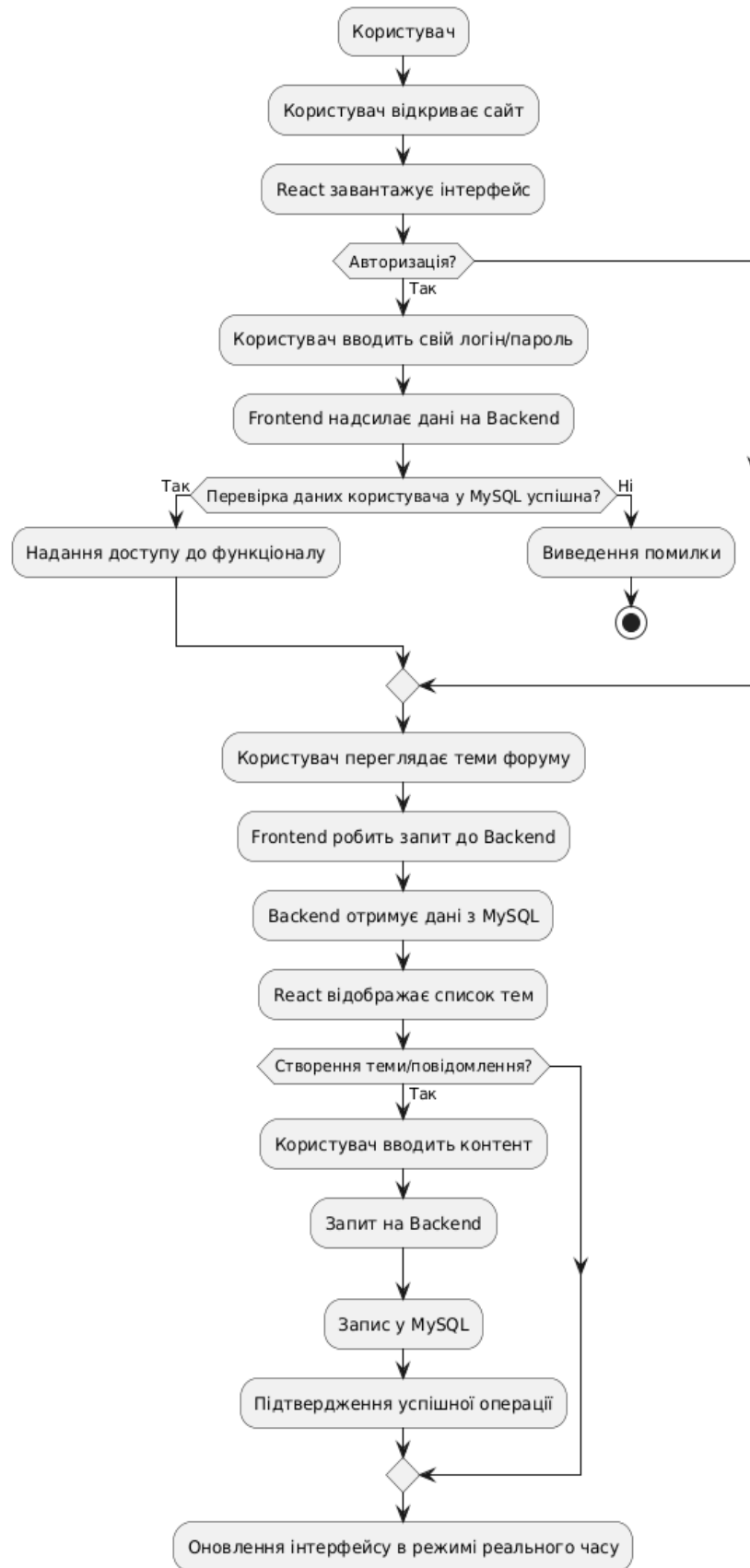


Рисунок 4.1 - Архітектура форум-орієнтованої інформаційної системи з використання психіатричних препаратів

Головна мета форум-орієнтованої інформаційної системи з використання психіатричних препаратів – базується саме забезпеченні структурованого обміну досвідом між користувачами форуму, надійного збору та ефективної агрегації даних щодо використання лікарських засобів.

Така система дозволяє користувачам (включаючи фахівців, пацієнтів та дослідників) отримувати та обговорювати інформацію про препарати в єдиному середовищі, своєчасно отримувати актуальні знання та взаємодіяти між собою для підвищення інформованості.

Переваги такої системи: цілісність статистичного обліку активності користувачів (завдяки розробленому методу агрегації), зменшення ризику поширення неперевіреної, шкідливої інформації завдяки модерації та валідації, а також підтримка актуальної бази знань. Ця система є важливим інструментом для подолання інформаційної фрагментації та підвищення прихильності до терапії.

Логічна структура інформаційної системи P-PsyForum побудована на взаємодії ключових архітектурних модулів, які забезпечують виконання функціональних вимог та практичну реалізацію наукової новизни дослідження.

Особлива увага при проектуванні приділена суворому поділу відповідальності: клієнтський модуль фокусується на процесах відображення та інтерактивної взаємодії, тоді як серверний модуль інкапсулює критичну бізнес-логіку, включаючи виконання методу структурно-статистичної агрегації [49].

Система включає декілька засадничих архітектурних модулів, першим з яких є модуль аутентифікації та безпеки. Він відповідає за ідентифікацію користувачів, безпечно збереження облікових даних через хешування паролів та управління правами доступу.

Для підтримки стабільності сесій застосовано технологію JWT (JSON Web Token), що дозволяє забезпечити безперервність авторизації при мінімальному навантаженні на серверні ресурси.

Паралельно функціонує модуль керування контентом (CMS), який обробляє всі транзакції, пов'язані з життєвим циклом постів та коментарів. Цей модуль

виступає точкою інтеграції для алгоритму користувацької транзакції, гарантуючи коректну валідацію та збереження інформації.

Центральне місце в структурі займає модуль агрегації та моніторингу активності, який безпосередньо реалізує методом структурно-статистичної агрегації. Даний модуль активується модулем CMS відразу після успішної фіксації нового контенту, забезпечуючи атомарне оновлення лічильників активності у базі даних [50].

Для забезпечення високої надійності процесу агрегації, даний модуль функціонує в ізольованому контексті, що дає йому можливість уникнути блокування основного потоку обробки запитів.

Коли модуль CMS подає сигнал про завершення запису для нового об'єкта, метод структурно-статистичної агрегації собою ініціює каскадне оновлення статистичних профілів. Цей процес супроводжується важливою внутрішньою перевіркою консистентності, наприклад: система зіставляє фактичну кількість записів у реляційних таблицях даних із агрегованими значеннями в профілі користувача.

У разі виявлення можливих розбіжностей у системі алгоритм сам автоматично виконує процедуру ресинхронізації, що повністю розкриває математичну точність аналітичних даних. Через це, такий підхід робить систему стійкою до потенційних збоїв у мережевому з'єднанні або можливих короткочасних перевантажень бази даних, зберігаючи цілісність соціального капіталу користувачів.

Особлива увага при проектуванні API модуля була приділена стандартизації форматів обміну даними (JSON), що дозволяє легко інтегрувати нові типи активності без необхідності глобального рефакторингу. Маршрутизація запитів побудована за принципом RESTful-архітектури, де кожна дія має свій унікальний ендпоінт та чітко визначений набір параметрів.

Це забезпечує високу прозорість розробки форуму: фронтенд-команда може доволі чітко відстежувати, які саме дані передаються та збираються для методу структурно-статистичної агрегації, а розробники серверної частини проєкту мають

можливість оптимізувати внутрішні SQL-запити незалежно від візуального відображення.

Взаємодія усіх компонентів системи один з одним реалізується через важливий API модуль, який визначає чіткий інтерфейс та набір можливих маршрутів для обміну даними між клієнтським та серверним рівнями, саме це й забезпечує логічну завершеність та керованість системи.

Ця взаємодія базується на чітко визначеній структурі кінцевих точок, які відповідають за кожен аспект функціонування платформи.

Основні маршрути API-модуля, що забезпечують авторизацію, створення контенту та роботу зі статистикою, систематизовано та представлено у таблиці 4.1.

Таблиця 4.1 - Основні маршрути API модуля

Етап Методу	Ключове Завдання	Відповідальна Технологія	Очікуваний Результат / Продукт
1. Планування та Формалізація	Визначення метрик та структури даних.	MySQL	Створення полів \$posts_count\$, \$comments_count\$ у таблиці \$Users\$.
2. Збір та Ініціація Обліку	Перехоплення користувацької транзакції та валідація.	Express.js	Верифікований \$HTTP POST\$-запит та виділений \$user_id\$.
3. Транзакційна Агрегація та Контроль	Безпосереднє оновлення лічильників у базі даних.	MySQL	Інкрементування статистичного поля (\$posts_count\$) та підтвердження цілісності зв'язку.

Кінець таблиці 4.1

4. Візуалізація та Динамічна Презентація	Формування та відображення кінцевого аналітичного звіту.	React.js	Динамічне оновлення профілю користувача та відображення актуальної статистики.
--	--	----------	--

Представлена структура API-маршрутів підтверджує чітке розділення відповідальності між компонентами системи. Кожен маршрут спроектований таким чином, щоб забезпечити як стандартну CRUD-функціональність (створення, читання, оновлення, видалення) для контенту, так і виконання важливих для сайту операцій системи, пов'язаних з інтеграцією методу агрегації.

Зокрема, маршрути створення нового контенту (POST /api/posts та /api/posts/:id/reply) є точками ініціації розробленого алгоритму, що гарантує цілісність обліку активності користувачів.

Для наочної демонстрації структури даних, що лежить в основі функціонування форуму та реалізації методу структурно-статистичної агрегації, розроблено спрощену схему реляційних сутностей зображену на Рисунку 4.2. Ця схема відображає основні таблиці бази даних MySQL та їхні ключові атрибути.

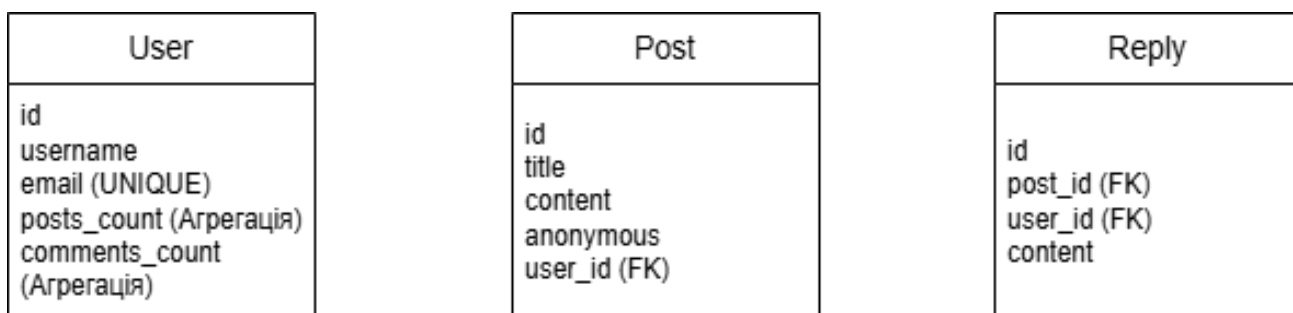


Рисунок 4.2 - Діаграма класів форум-орієнтованої інформаційної системи з використання психіатричних препаратів

Діаграма класів інформаційної системи (Рисунок 4.1) відображає архітектуру рівня даних системи P-PsyForum, який реалізовано за допомогою СУБД MySQL. Вона чітко ілюструє три основні сутності, що формують ядро форуму: User (облікові записи користувачів), Post (теми, створені користувачами) та Reply (коментарі). Ця структура відповідає за весь контент які створюють користувачі та для реєстрації нових користувачів та авторизації існуючих.

Для забезпечення функціоналу системи, оскільки вона не лише зберігає контент, але й містить поля (`posts_count`, `comments_count`) у таблиці User, які безпосередньо використовуються для реалізації методу структурно-статистичної агрегації. Візуалізація підтверджує, що архітектура даних оптимізована як для зберігання структурованої інформації, так і для ефективного, цілісного обліку активності користувачів [51].

4.2 Програмне забезпечення форуму та інтерфейс системи

Розроблена Форум-орієнтована інформаційна система являє собою повнофункціональний веб-додаток, реалізований відповідно до визначеної тривірневої архітектури. Візуальна частина системи була спроектована з акцентом на ергономічність, інтуїтивну навігацію та забезпечення обміну досвідом між користувачами, що відповідає меті предметної області.

Для глибшого аналізу практичної реалізації системи необхідно розглянути, як саме архітектурні рішення трансформуються у функціональні елементи інтерфейсу. Ергономічність системи базується на принципах адаптивного дизайну, що реалізується через гнучку сітку компонентів React.js та препроцесорні стилі SASS.

Це дозволяє інтерфейсу динамічно підлаштовуватися під технічні параметри пристрою користувача, зберігаючи при цьому доступність до важливих елементів управління, таких як форми пошуку препаратів та навігаційні панелі форумних гілок. Особлива увага приділяється ієрархії візуальних акцентів: найбільш затребувана інформація (статистика активності та нові повідомлення)

розміщується у фокусних зонах, що мінімізує когнітивне навантаження на відвідувача форуму.

Одним із основних аспектів ергономіки є впровадження механізмів миттєвого зворотного зв'язку (Feedback loops). Кожна дія користувача, оброблена через метод структурно-статистичної агрегації, супроводжується візуальною зміною стану інтерфейсу - від оновлення індикаторів прогресу до появи повідомлень про зміну статусу. Це створює відчуття безперервної взаємодії та високої швидкодії системи, попри складність фонових процесів обробки даних у MySQL.

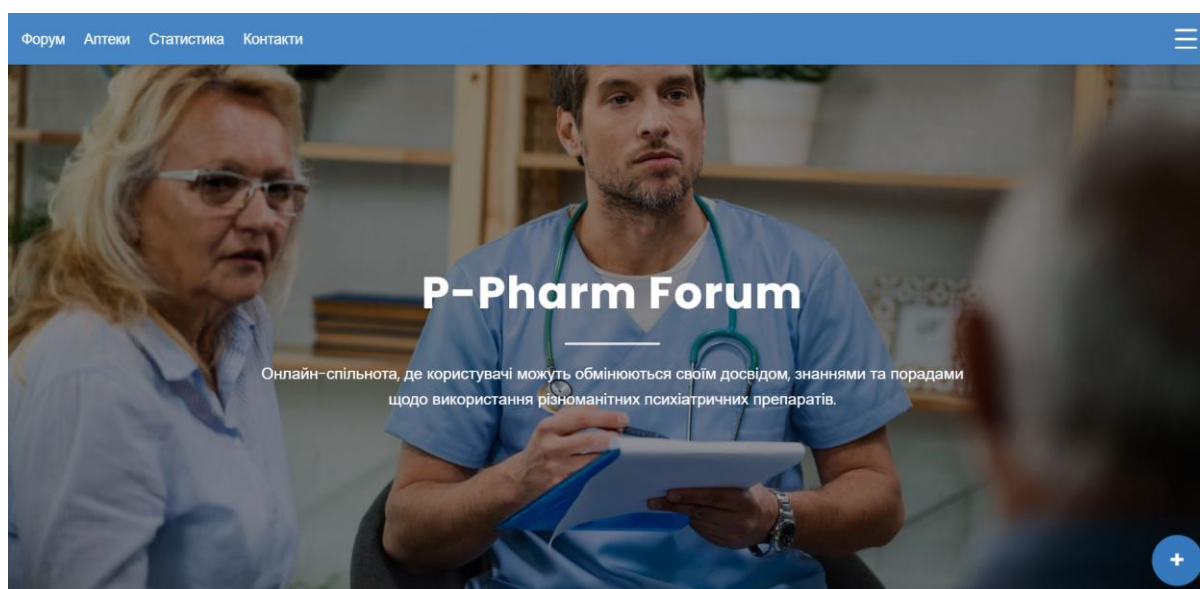


Рисунок 4.3 - Титульна сторінка форуму

Рисунок 4.3 вітає користувача титульною сторінкою, яка слугує основним інформаційним та навігаційним центром системи. У верхній частині інтерфейсу розташовано функціональний хедер із ключовими навігаційними елементами: «Форум», що забезпечує прямий доступ до обміну досвідом та створення контенту; «Аптеки», що надає користувачам довідкову інформацію про місцезнаходження або наявність препаратів; та «Статистика», яка є візуалізацією результатів роботи розробленого методу структурно-статистичної агрегації у системі.

Нижче під навігаційною панеллю розміщені блоки швидкого перегляду актуальної інформації, які завантажуються асинхронно для запобігання блокуванню основного потоку виконання програми.

Реалізація такого підходу дозволила зберегти високу швидкість відгуку інтерфейсу навіть за умов низької пропускної здатності мережі, це важливо задля стабільної роботи візуальних компонентів при отриманні великих масивів даних із серверної частини системи.



Рисунок 4.4 - Інформаційний блок про основний функціонал форуму

На додаток до ергономічної навігаційної панелі, центральна область титульної сторінки містить невеликий, але інформативний блок, який знайомить користувача з ключовими можливостями платформи.

Цей блок деталізує три основні напрямки, доступні в системі: можливість спілкування на Форумі для взаємної підтримки, функціонал зручного збору та

відображення статистики активності, та довідковий модуль з посиланнями на українські аптечні мережі для практичної допомоги [52].

Такий розподіл елементів допомагає користувачеві відразу зрозуміти структуру ресурсу, не блукаючи по вкладках меню. По суті, головна сторінка працює як диспетчер, що спрямовує запити до потрібної частини бази даних, забезпечуючи швидкий відгук системи навіть при першому візиті.

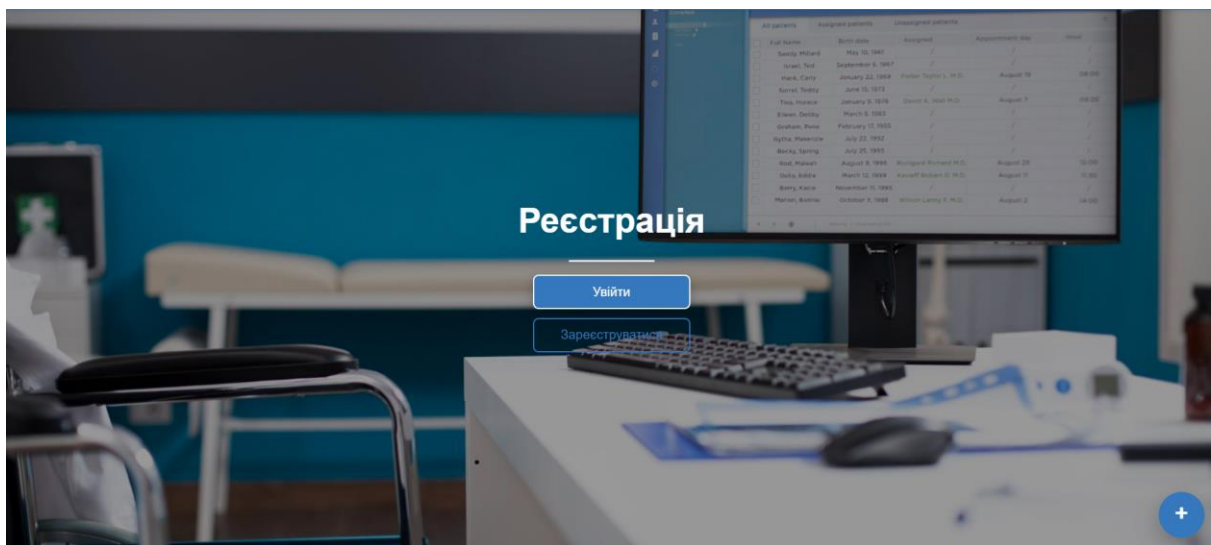


Рисунок 4.5 - Меню реєстрації та авторизації користувача

Інтеграція користувача в систему розпочинається з блоку аутентифікації, який візуалізований за допомогою кнопок «Зареєструватися» та «Увійти». Ці елементи інтерфейсу є прямою реалізацією функціональних вимог, що відповідають за реєстрацію нових користувачів та їх авторизований вхід.

Доступ до більшості інтерактивних функцій системи - таких як створення нових тем, коментування та перегляд персоналізованої статистики, що є результатом методу агрегації - вимагає попередньої аутентифікації.

Сам цей блок і потрібен для активації модуля аутентифікації та безпеки, забезпечуючи контроль доступу та цілісність даних користувачів у інформаційній системі.

РЕЄСТРАЦІЯ

Ім'я та Прізвище

Дата народження

Email

Пароль

Зареєструватися

Рисунок 4.6 - Меню для реєстрації нового користувача на форумі

Процес реєстрації нового користувача є більш розширеним. Окрім обов'язкових полів електронної пошти (e-mail) та пароля, які проходять хешування перед збереженням, користувачу також необхідно надати ім'я та прізвище а ще треба вписати й дату народження.

Збір цих додаткових атрибутів (згідно з метаданими з файлу `ppharm_db_users.sql` - `username`, `bday`) необхідний для персоніфікації профілю та, потенційно, для подальшого сегментування користувачів у межах аналітичних досліджень, зберігаючи при цьому найвищий рівень конфіденційності.

ВХІД

Email

Пароль

Запам'ятати мене

Увійти

Рисунок 4.7 - Меню для авторизації існуючого користувача форуму

Форма входу вимагає від користувача введення валідної електронної пошти (e-mail) та його особистого пароля. Ці дані використовуються модулем

аутифікації для верифікації облікових даних. Для підвищення зручності користування інтегровано опціональний елемент контролю (чекбокс) для запам'ятовування сесії. Успішна верифікація ініціює видачу захищеного JWT-токена, який забезпечує подальший доступ до приватної функціональності системи [53-54].

Після введення даних система автоматично перевіряє їх на відповідність формату, щоб не навантажувати сервер завідомо некоректними запитами. Якщо пароль і пошта збігаються з записами в базі даних, клієнт отримує токен, який зберігається в браузері та автоматично додається до кожного наступного звернення до API.

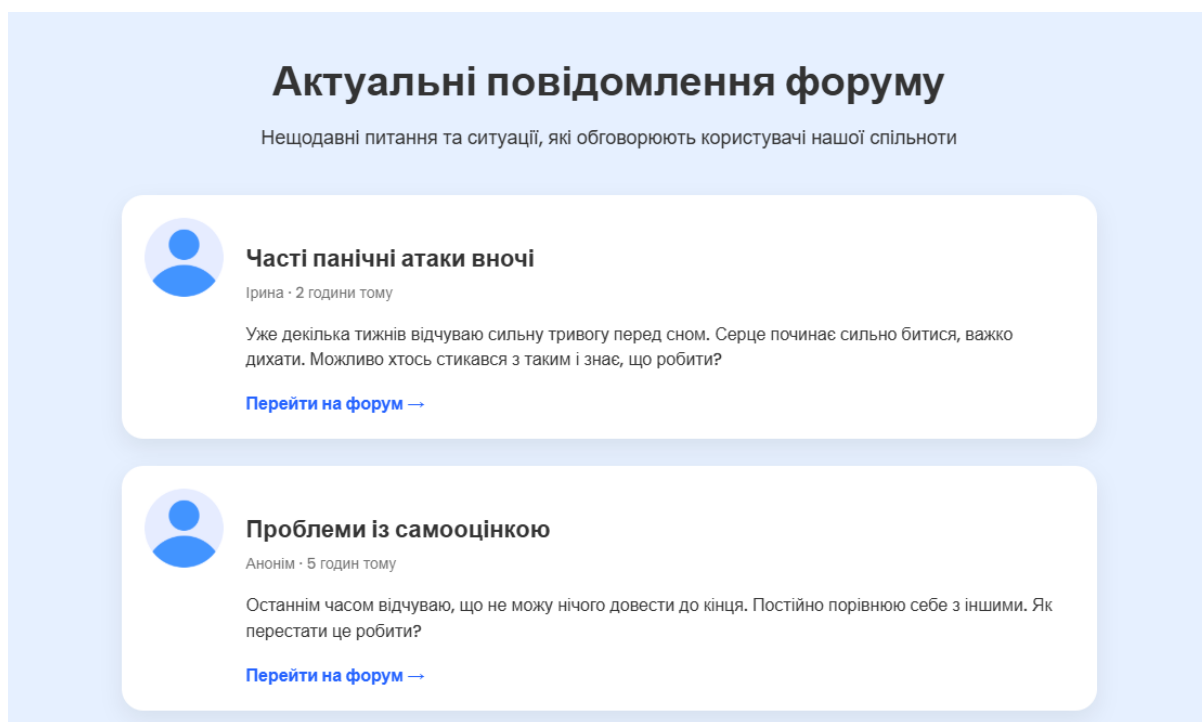


Рисунок 4.8 - Актуальні повідомлення форуму виведені на титульну сторінку

Кінцевою частиною титульної сторінки є блок «Актуальні повідомлення форуму». Цей динамічний компонент служить для попереднього перегляду контенту, відображаючи три найсвіжіші теми, створені користувачами. Кожне повідомлення представлено стисло, що дозволяє відвідувачу швидко оцінити поточну активність та тематику обговорень.

Цей блок виконує функцію «міст-перехідника»: він не лише інформує про останні події, але й містить клікабельний елемент, що спрямовує користувача безпосередньо на сторінку «Форум». Таке дизайнерське рішення забезпечує безперешкодний перехід від пасивного перегляду інформації до активної взаємодії та участі у спільноті.

Цей віджет завжди автоматично оновлюється щоразу, коли хтось із учасників публікує новий пост, тому головна сторінка завжди виглядає «живою» для пошукових систем і постійних відвідувачів.

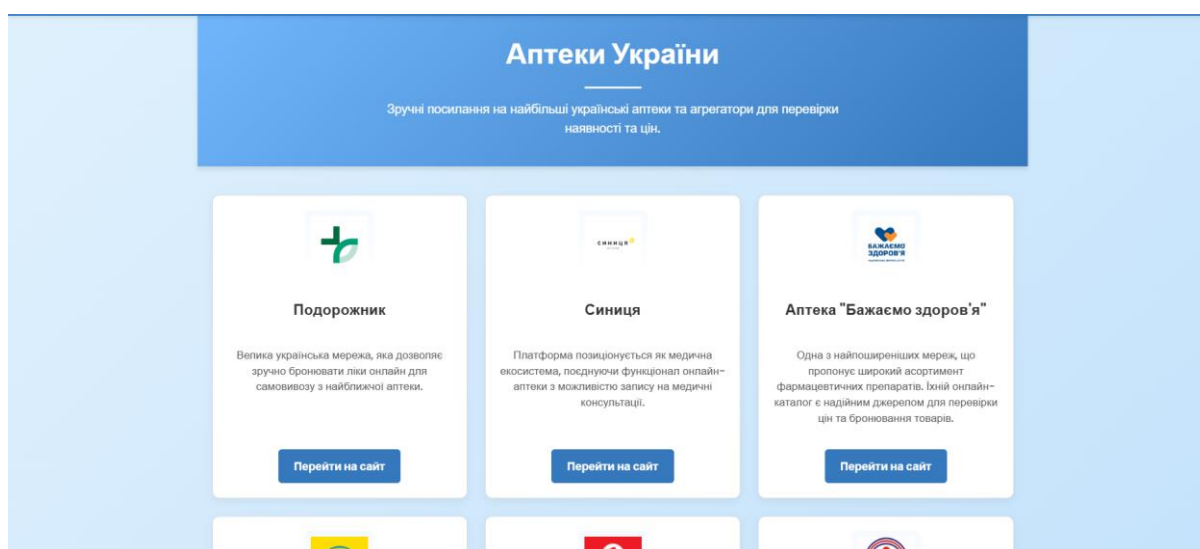


Рисунок 4.9 - Секція з переходами на перевірені та популярні аптеки України

Секція «Аптеки України» реалізує довідкову функцію, інтегровану у систему відповідно до загальної концепції підтримки користувачів. Цей модуль являє собою структурований каталог, що містить інформацію про найпопулярніші аптечні мережі України. Кожна позиція каталогу включає офіційний логотип, короткий опис мережі, а також пряме посилання на її офіційний веб-сайт, де користувачі можуть оперативно здійснити пошук та замовлення необхідних препаратів.

Адміністрування цього модуля є достатньо гнучким: він може бути динамічно доповнений або урізаний розробником (адміністратором) відповідно до актуальних потреб та змін на фармацевтичному ринку. Система забезпечує

інформаційну ланку між віртуальною підтримкою та практичними медичними потребами користувачів форуму [55].

Для забезпечення високої адаптивності адміністрування, архітектура модуля побудована на принципах декларативного опису даних, де нові категорії препаратів або параметри моніторингу додаються через зміну конфігураційних об'єктів без втручання в основне ядро системи.

Це дозволяє адміністратору оперативно реагувати на появу нових терапевтичних протоколів, інтегруючи їх у загальну структуру форуму та налаштовуючи специфічні ваги для методу структурно-статистичної агрегації в межах нових розділів. Окрім гнучкості налаштувань, серверна частина забезпечує автоматичне логування всіх адміністративних дій, що формує надійний аудит-слід та дозволяє відстежувати вплив змін у структурі контенту на динаміку користувацької активності.

Візуальна презентація частини функціоналу форуму охопила: ергономіку титульної сторінки з її ключовими навігаційними елементами, механізми авторизації та реєстрації, динамічний прев'ю-блок актуальних повідомлень, що стимулює перехід до форуму, а також довідковий модуль «Аптеки України».

Усі представлені фрагменти інтерфейсу підтверджують, що програмна реалізація системи відповідає визначеним функціональним вимогам, забезпечуючи високу зручність використання та готовність до подальшої інтеграції ключового наукового функціоналу - методу структурно-статистичної агрегації.

4.3 Ключовий інтерактивний функціонал та апробація методу структурно-статистичної агрегації

Ключові інтерактивні функції системи не лише забезпечують комунікацію, а й слугують майданчиком для апробації методу структурно-статистичної агрегації. Саме цей компонент є ядром всієї інформаційної системи. Інтерфейс сторінок форуму наочно показує, як працюють алгоритми керування даними та як вони обробляються «під капотом» [56].

Окрему увагу приділено модулю персональної статистики: він підтверджує виконання основних вимог, демонструючи, що метрики активності оновлюються в реальному часі відразу після кожної транзакції в базі даних. Це візуально підтверджує цілісність системи та коректність роботи агрегаційних конвеєрів. В результаті ми отримуємо систему, яка самостійно підтримує актуальність даних без втручання адміністратора.

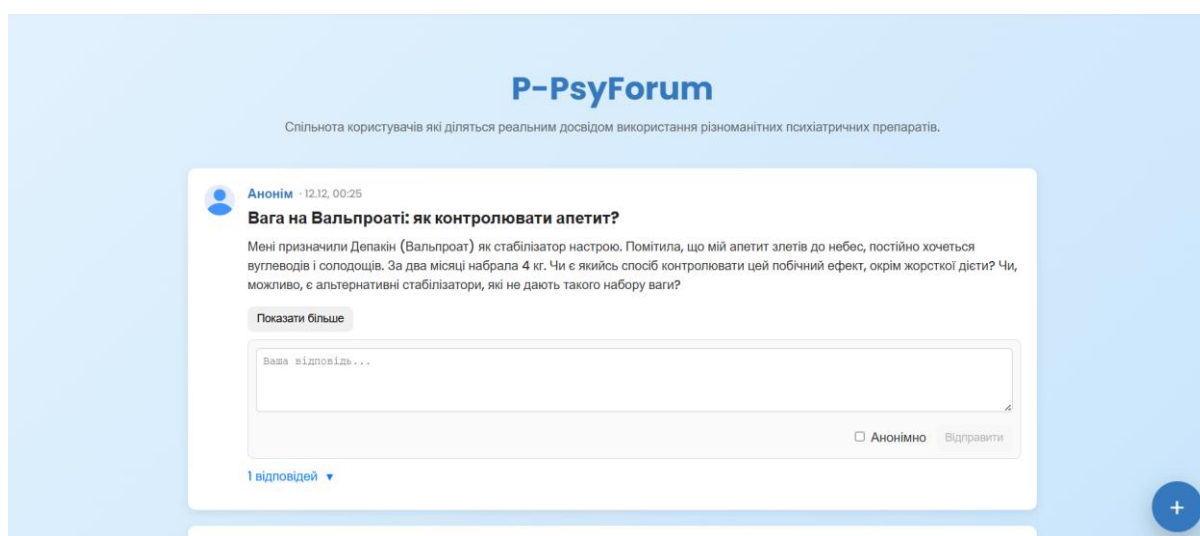


Рисунок 4.10 - Інтерфейс форуму та повідомлення користувачів

Форум є центральною інтерактивною частиною системи, де реалізовано вимоги до керування контентом. На цій сторінці, що зображено на рисунку 4.10 відображається повний лістинг усіх повідомлень, залишених користувачами протягом усього часу функціонування платформи.

Для забезпечення високої інтерактивності форумного модуля, механізм відображення повідомлень базується на принципі динамічної пагінації та віртуалізації списків. Це дозволяє системі стабільно працювати навіть при наявності тисяч публікацій, завантажуючи лише ті об'єкти, які знаходяться у видимій зоні екрана користувача.

Кожен елемент списку є автономним React-компонентом, що має власний життєвий цикл та стан. Така архітектура дає змогу реалізувати функціонал

«миттєвого оновлення»: коли один користувач залишає коментар, метод структурно-статистичної агрегації оновлює лічильник відповідей на сервері, а клієнтська частина через API-модуль актуалізує цифровий індикатор під постом без переривання процесу читання для інших учасників.

Особлива увага в інтерфейсі приділена візуальному розмежуванню типів контенту та ролей авторів. Завдяки інтеграції з базою даних MySQL, система в реальному часі зіставляє ідентифікатор автора з його поточним рангом, що дозволяє виводити поруч із іменем специфічні маркери авторитетності (наприклад, колірне виділення нікнейму для експертів). Це створює багаторівневу інформаційну структуру, де користувач може миттєво оцінити значущість повідомлення на основі візуальних підказок.

Реалізація функції «Анонімно» на рівні бази даних передбачає маскування персональних ідентифікаторів у публічному доступі, проте внутрішній алгоритм продовжує враховувати ці публікації для методу структурно-статистичної агрегації, забезпечуючи баланс між приватністю користувача та повнотою аналітичних даних системи.

Кожне повідомлення містить достатню інформацію для ознайомлення: заголовок, ім'я автора (або позначку "Анонімно"), час публікації, а також загальну кількість відповідей. Нижче кожного посту розташовані елементи для подальшої взаємодії: поле для написання коментаря, чекбокс «Анонімно» для збереження конфіденційності, та можливість перегляду відповідей, залишених іншими користувачами [57-58].

Для ініціювання нової теми в нижньому правому куті розміщена кнопка з іконкою «плюс», яка забезпечує швидкий доступ до форми створення нового повідомлення.

Така організація елементів дозволяє користувачеві брати участь у дискусії без зайвих переходів між сторінками, завдяки чому значно підвищується швидкість взаємодії. Використання плаваючої кнопки (Floating Action Button) для створення нових тем дає користувачеві сталий доступ до ключового функціоналу незалежно від довжини стрічки повідомлень.

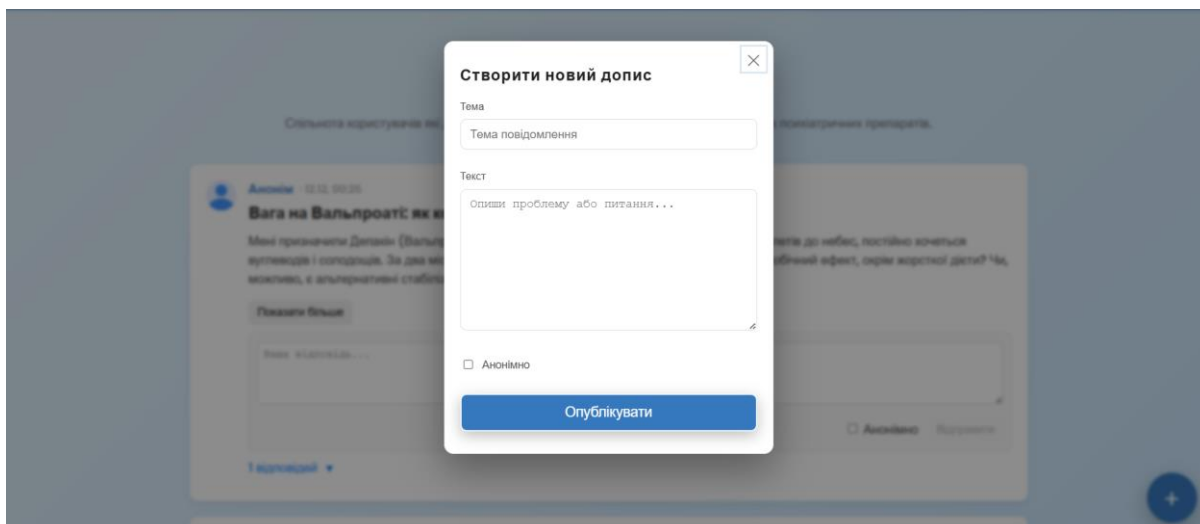


Рисунок 4.11 - Вікно з можливістю написання повідомлення на форумі

Доступ до форми створення нового допису ініціюється натисканням на динамічну кнопку «+», розташовану у нижньому правому куті інтерфейсу Форуму. Вікно «Створити новий допис» який показано на рисунку створений для можливої реалізації вимоги з керування контентом.

Форма вимагає введення Теми повідомлення та основного Тексту повідомлення. Для збереження анонімності інтегровано чекбокс «Анонімно», що забезпечує конфіденційність автора. Після заповнення полів та натискання «Опублікувати», інформація відправляється на сервер.

У цей момент модуль керування контентом виконує життєвий цикл транзакції: зберігає новий пост у базу даних, а потім автоматично ініціює метод структурно-статистичної агрегації для оновлення лічильника `posts_count` у профілі користувача [59].

Успішно створене повідомлення негайно з'являється у загальному лістингу Форуму, та буде доступне для комунікації з ним серед інших користувачів, це забезпечує безперервність інформаційного обміну та дозволяє спільноті в режимі реального часу накопичувати актуальні дані, формуючи динамічне середовище для підтримки та аналізу досвіду використання препаратів.

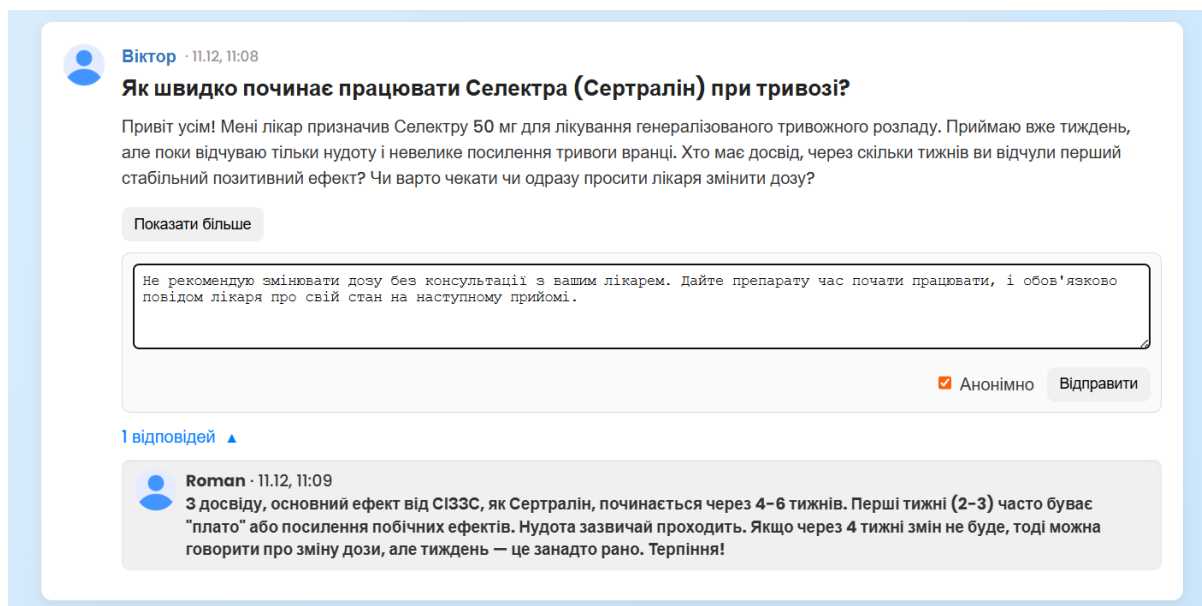


Рисунок 4.12 - Взаємодія користувача з повідомленнями інших користувачів

Секція відповідей забезпечує реалізацію інтерактивної вимоги про Додавання коментарів, формуючи гілку обговорення під основним дописом.

Кількість наявних відповідей відображається на клікабельній кнопці, що забезпечує швидкий доступ до розширеної інформації та підтверджує оперативний облік контенту. Сама секція відображає коментарі у хронологічному порядку, а саме - від найновіших відповідей до найстаріших, що сприяє актуальності інформації для користувача [60].

Функціонал цієї секції вельми таки схожий на механізм створення нового допису: користувач має поле для введення тексту відповіді, можливість зберегти анонімність за допомогою чекбоксу, а також кнопку «Відправити».

Після публікації коментаря, відбувається важлива транзакція: коментар зберігається, і автоматично ініціюється метод агрегації, який оновлює лічильник `comments_count` у профілі автора відповіді. Це підтверджує цілісність даних та безперебійну роботу системи обліку активності. Кожен внесок користувача ментально відображається в його персональній статистиці, створюючи прозору систему соціального підтвердження та мотивації учасників спільноти до активного обміну досвідом.

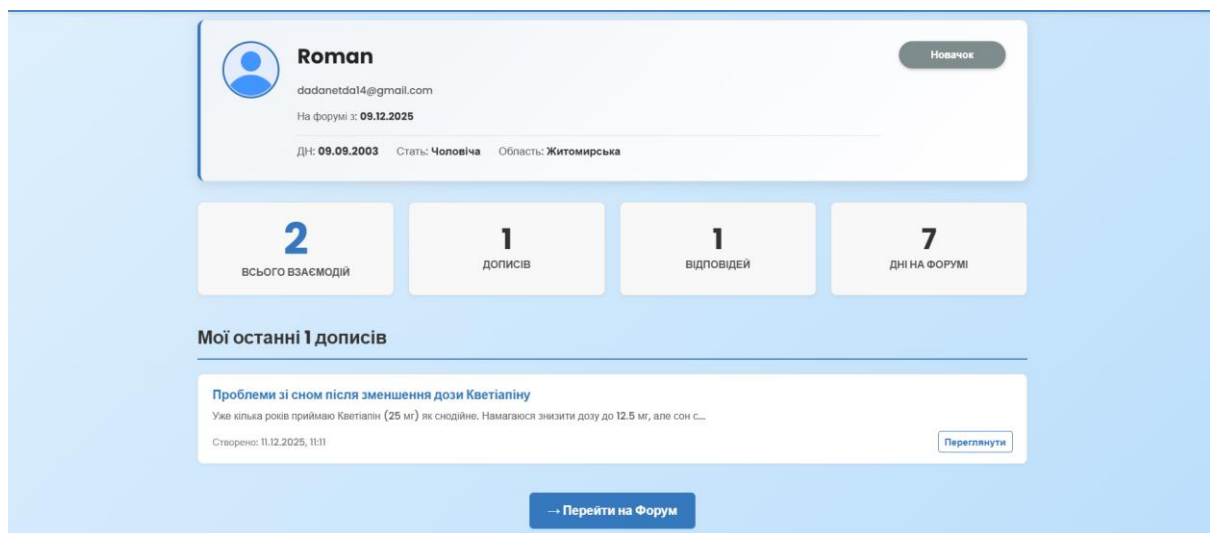


Рисунок 4.13 - Динамічно оновлювана статистика користувача форуму

Інтерфейс персональної статистики яка динамічно оновлюється в реальному часі (Рисунок 4.13) є візуальним втіленням вимоги про Оновлення статистики та Відображення статистики, демонструючи результат роботи методу структурно-статистичної агрегації. Статистику розділено на функціональні модулі, забезпечуючи користувачу комплексний огляд його активності.

Перша частина, профільна картка, відображає основні дані користувача та його поточне звання. Система автоматично присвоює звання на основі двох динамічних критеріїв: загальної кількості дописів та тривалості перебування на форумі (`created_at`). Ця механіка мотивує користувачів до активної та довгострокової участі у спільноті.

Для надання користувачу актуального огляду його активності, статистика додатково інтегрує механізм вибірки останніх дописів. Цей функціонал реалізовано через окремий запит до серверного API, який використовує функцію SQL-запиту з фільтрацією за ідентифікатором користувача (`user_id = ?`) [61]. Запит не лише повертає необхідні метадані кожного посту (заголовок, вміст, час створення), але й обмежує результат, використовуючи оператор `LIMIT` (у даному випадку, максимум 15 останніх постів).

Логіка міжмодульної взаємодії та доступу до функціональних можливостей системи P-PsyForum регламентується через чітко визначену структуру API-

маршрутів. Модуль аутентифікації забезпечує ідентифікацію користувачів та видачу персональних JWT-токенів через відповідний POST-запит, що є необхідною умовою для подальшої безпечної роботи.

Для забезпечення високої відмовостійкості та прозорості обміну даними, кожен API-маршрут спроектований з урахуванням принципів ідемпотентності, що запобігає дублюванню операцій агрегації при випадкових повторних запитах від клієнта.

Взаємодія між модулем керування контентом та модулем обліку активності реалізована через внутрішню подієву модель (Event-driven interaction): як тільки транзакція в MySQL отримує статус успішно завершеної, генерується внутрішня подія, яка стає тригером для методу структурно-статистичної агрегації.

Це дозволяє розділити часову шкалу запису даних та їх аналітичної обробки, забезпечуючи миттєву відповідь користувачеві (Response time) при збереженні гарантованої консистентності статистики у фоновому режимі.

Спеціалізований інтерфейс модуля обліку активності використовує агрегаційні конвеєри запитів до бази даних, що дозволяють формувати аналітичні звіти «на льоту» без створення надмірного навантаження на систему [62]. Завдяки використанню JWT-токенів, кожен запит до статистичних маршрутів супроводжується автоматичною перевіркою прав доступу (Scopes), саме це гарантує конфіденційність медико-соціальних даних та запобігає несанкціонованому отриманню чужих аналітичних профілів [].

Така уніфікована архітектура API-маршрутів створює надійний програмний інтерфейс, де кожен метод - від авторизації до фінальної агрегації - діє як частина єдиного, злагодженого механізму, орієнтованого на достовірність та безпеку інформації у системі P-PsyForum.

Модуль керування контентом охоплює маршрути для створення нових постів та відповідей, виконання яких автоматично ініціює роботу методу структурно-статистичної агрегації для актуалізації даних. Паралельно із цим, модуль обліку активності надає спеціалізований інтерфейс для отримання персональної статистики користувача у форматі структурованого аналітичного звіту. Такий

підхід дозволяє забезпечити уніфіковану обробку даних та безперервний моніторинг діяльності суб'єктів у межах інформаційного середовища форуму [64-66].

4.4 Висновки

Архітектурна основа системи побудована на стійкій тривірневій архітектурі, що забезпечує чітке розділення відповідальності між компонентами. Як технологічна база був обраний стек, що включає MySQL для цілісного зберігання даних, бекенд на Node.js з Express.js для обробки бізнес-логіки та API, та фронтенд на сучасному JavaScript з використанням бібліотеки React.js для динамічних елементів.

Такий підхід забезпечує не лише надійність та масштабованість, але й високий рівень безпеки, що підтверджується інтеграцією механізмів bcrypt.js для хешування паролів та JWT-токенів для авторизації.

Розроблений зрозумілий користувацький інтерфейс забезпечує високий рівень ергономіки та інтуїтивності. Було успішно реалізовано основні вимоги до доступу до системи: форми реєстрації та авторизації гарантують безпечну ідентифікацію користувачів та збір профілюючих метаданих.

Структура головної сторінки ефективно орієнтує користувача, акцентуючи увагу на ключових функціональних модулях. Допоміжні секції, як-от динамічний блок актуальних повідомлень та довідковий модуль «Аптеки України», виконують роль стимулювання активності та надання практичної інформаційної підтримки.

Ядро функціоналу та метод агрегації реалізовано у ключових комунікаційних модулях форуму що підтверджує функціональність системи у сфері керування контентом. Користувачам надана можливість створювати нові теми та відповідати на них з підтримкою анонімності.

Важливо те що, процес створення контенту інтегровано з науковою складовою: кожна транзакція публікації допису або відповіді автоматично ініціюється методом структурно-статистичної агрегації.

Візуалізація статистики та React.js: Результати агрегації візуалізовані на персональній статистиці користувача, що є прямою демонстрацією виконання вимог. Інтерфейс користувацької статистики реалізовано на React.js, що дозволяє динамічно оновлювати стан даних.

Система відображає ключові лічильники (пости, відповіді, дні на форумі) та автоматично присвоює звання на основі агрегованих метрик. Це підтверджує цілісність обліку активності та безперебійну роботу методу, який забезпечує атомарне оновлення лічильників після кожної дії користувача.

Важливо додати, наявність секції «Мої останні дописи» з використанням оптимізованих SQL-запитів демонструє ефективну роботу з великими обсягами даних. Таким чином, програмна реалізація успішно створила функціональне середовище, яке готове до подальших етапів апробації.

Впроваджений та використаний стек технологій забезпечив високу швидкість обробки запитів та безшовну синхронізацію між серверною логікою та клієнтським інтерфейсом, саме це робить взаємодію з платформою максимально плавною для користувача.

Особливу увагу в реалізації було приділено масштабованості, структурі коду та оптимізації бази даних. Завдяки цьому система буде витримувати значне зростання кількості учасників без втрати продуктивності.

P-PsyForum постає не просто як прототип, а як цілісна інформаційна інфраструктура з надійним захистом даних, продуманою ергономікою та інтегрованими модулями підтримки, що у цілому створює якісно новий стандарт спеціалізованих вебплатформ для звичайних користувачів та фахівців у галузі психіатрії.

ВИСНОВКИ

У роботі за результатами виконаних теоретичних та практичних досліджень розроблено та імплементовано форумно-орієнтовану інформаційну систему P-PsyForum, призначену для формування єдиного інтегрованого середовища для обміну досвідом між користувачами форуму та аналізу структурованої активності цих же користувачів у сфері психіатричних препаратів.

Ядром системи є важливий для системи метод структурно-статистичної агрегації, який забезпечує атомарний збір та оновлення метаданих користувачів, необхідних для формування об'єктивного профілю їхньої поведінки.

У першому розділі було проведено детальний аналіз предметної області та ідентифіковано ключові проблеми, що стосуються фрагментації інформації про лікарські засоби, обмеженого доступу до актуальних знань та високого ризику поширення помилкової інформації. На підставі цих проблем було формалізовано функціональні вимоги до майбутньої інформаційної системи та запропоновано її структуру, спрямовану на створення єдиного інформаційного середовища для лікарів, пацієнтів та дослідників.

У другому розділі виконано архітектурне та технологічне обґрунтування системи. Було доведено доцільність використання саме трирівневої архітектури на базі Full-Stack JavaScript-парадигми. Для забезпечення транзакційної цілісності та підтримки складних аналітичних запитів обрано реляційну СУБД MySQL (ACID).

На рівні додатку застосовано Node.js/Express.js для високопродуктивної обробки API-запитів, а на рівні представлення - фреймворк React.js для створення динамічного, компонентного інтерфейсу та ефективного керування станом (Virtual DOM).

У третьому розділі було детально розроблено та формалізовано метод структурно-статистичної агрегації. Основна суть методу полягає у забезпеченні атомарного обліку кожної дії користувача (публікації посту чи відповіді) з одночасним оновленням його персональних статистичних лічильників та індексів. Це дозволяє системі підтримувати актуальний та об'єктивний статистичний

портрет користувача, який слугує основою для подальшого профілювання та аналізу. Було також розроблено деталізовану логічну модель бази даних, що відображає зв'язки між сутностями, необхідними для агрегації.

У четвертому розділі здійснено програмну імплементацію та апробацію розробленого рішення. Було успішно реалізовано ключові функціональні модулі, включаючи безпечні механізми авторизації/аутентифікації на базі JWT-токенів та повноцінний функціонал форуму (створення тем користувачами форуму та відповідей на них).

Важливим результатом є реалізація персонального статистики, який візуально підтверджує працездатність методу агрегації. Динамічні лічильники (пости, відповіді, дні на форумі) та система автоматичного присвоєння звань на основі агрегованих даних демонструють успішне виконання усіх функціональних вимог системи.

Набула подальшого розвитку інформаційна технологія обробки даних у форумно-орієнтованих системах шляхом інтеграції методу структурно-статистичної агрегації, який дозволяє формувати об'єктивні, транзакційно узгоджені метадані користувачів у реальному часі, що стає основою для їх подальшої поведінкової класифікації та аналізу.

Впровадження результатів роботи дозволили створити функціонально завершену, безпечну та масштабовану платформу, що ефективно вирішує проблему фрагментації медико-соціальної інформації та створює прозоре середовище для обміну досвідом між користувачами форуму. Система повністю готова до експлуатації та подальшого розширення функціоналу, зокрема, у напрямку автоматизованого аналізу контенту.

За темою кваліфікаційної роботи магістра опублікована одна стаття у фаховому науковому виданні: Недашківський Р.В. Форум-орієнтована інформаційна система з використання психіатричних препаратів. Філософські виміри техніки: Збірник тез IV Міжнародної наукової конференції молодих учених та здобувачів вищої освіти, 26 листопада 2025 р. С. 99–101.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Torous J., Firth J., Huckvale K. Digital mental health and COVID-19: Using technology to accelerate access and equity. *NPJ Digital Medicine*. 2021. Vol. 4. Article 65.
2. Insel T. Digital mental health care: five lessons from Act 1 and a preview of Acts 2–5. *npj Digital Medicine*. 2023. Vol. 6. Article 12.
3. WHO. mhGAP Intervention Guide for mental, neurological and substance use disorders in non-specialized health settings. *World Health Organization*. 2016.
4. Антонова О. В. Цифровізація сфери охорони здоров'я як чинник трансформації сучасної медицини. *Наукові записки НаУКМА. Соціологічні науки*. 2021. Т. 4. С. 12–19.
5. Naslund J.A., Aschbrenner K.A., Marsch L.A., Bartels S.J. The future of mental health care: peer-to-peer support and social media. *Epidemiology and Psychiatric Sciences*. 2016. Vol. 25. No. 2. pp. 113–122.
6. Lora A., et al. Information for mental health systems: an instrument for assessment of inputs, processes and outputs. *International Journal of Mental Health Systems*. 2016. Vol. 10. Article 54.
7. Berardi C., Antonini M., Jordan Z., et al. Barriers and facilitators to the implementation of digital technologies in mental health systems: a qualitative systematic review. *BMC Health Services Research*. 2024. Vol. 24. Article 98.
8. Луньов В. Є. Цифрові технології в охороні психічного здоров'я: світовий досвід та перспективи впровадження. *Психологічне консультування і психотерапія*. 2021. Вип. 15. С. 22–30.
9. Banks A., Porcello E. *Learning React: Modern Patterns for Developing React Apps*. O'Reilly Media. 2020. 2nd ed.
10. Cantelon M., Harter M., Holowaychuk T., Rajlich N. *Node.js in Action*. Manning Publications. 2017.

11. Коломійцев О. В., Кучук Г. А. Метод синтезу структури бази даних для розподілених інформаційних систем. *Збірник наукових праць Харківського національного університету Повітряних Сил*. 2020. № 4. С. 102–109.
12. Verborgh R. REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces. O'Reilly Media, 2021. 180 p.
13. Tilkov S., Vinoski S. Node.js: Using JavaScript to Build High-Performance Network Programs. *IEEE Internet Computing*. 2010. Vol. 14. No. 6. pp. 80–83.
14. Електронна система охорони здоров'я (eHealth). Архітектура та технічні стандарти взаємодії інформаційних систем. 2024. URL: <https://ehealth.gov.ua/tehnichni-dokumenty/> (дата звернення: 09.09.2025).
15. Mikowski M., Powell J. Single Page Web Applications: JavaScript end-to-end. Manning. 2013.
16. Subramanian V. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node. Apress. 2019.
17. Meta Platforms Inc. Thinking in React: Component-Based UI Development. 2024. URL: <https://react.dev/learn/thinking-in-react> (дата звернення: 19.09.2025).
18. Brown E. Web Development with Node and Express: Leveraging the JavaScript Stack. O'Reilly Media. 2019.
19. Google Developers. Web Vitals: Essential metrics for a healthy site and real-time performance monitoring. 2024. URL: <https://web.dev/articles/vitals> (дата звернення: 21.09.2025).
20. New Relic Inc. Full-Stack Monitoring: Real-time visibility into application performance and infrastructure. 2024. URL: <https://newrelic.com/resources/ebooks/guide-to-full-stack-monitoring> (дата звернення: 22.09.2025).
21. Oracle Corporation. MySQL Database Design and Data Modeling Guide: Handling Hierarchical Data. 2024. URL: <https://dev.mysql.com/doc/refman/8.0/en/hierarchical-data-queries.html> (дата звернення: 26.09.2025).

22. Li W. Real-time data processing in modern web applications. *International Journal of Computer Science*, 2023. Vol. 45. pp. 210–222.
23. Pries A., Lakhe R. Methods of structural data analysis and aggregation in relational environments. CRC Press. 2014.
24. Пасічник В. В., Резніченко В. А. Організація баз даних та знань. К.: ВХВ, 2011.
25. Greenhalgh T., et al. Beyond adoption: a new framework for evaluating digital health. *Journal of Medical Internet Research*, 2021. Vol. 23. e367.
26. Скрипник Ю. О. Психологічна підтримка в мережі: аналіз форумних платформ. *Психологія і суспільство*, 2023. № 2. С. 112–120.
27. Kleppmann M. Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems. O'Reilly Media. 2017.
28. Masdari M., Khoshnevis A. A survey and classification of the workload forecasting methods in cloud computing. *Cluster Computing*. 2022. Vol. 25. P. 2399–2424.
29. Auth0 by Okta. JSON Web Token (JWT) Introduction and Best Practices for Single Page Applications. 2024. URL: <https://auth0.com/docs/secure/tokens/json-web-tokens> (дата звернення: 07.10.2025).
30. Melnikov D. Modern Authentication Protocols: JWT and OAuth 2.0. Wiley. 2021.
31. Довбиш А. С. Методи забезпечення безпеки даних у розподілених інформаційних системах. *Кібербезпека: освіта, наука, техніка*. 2019. № 2. С. 58–64.
32. Lodderstedt T., McGloin M., Hunt P. OAuth 2.0 *Threat Model and Security Considerations*. RFC 6819. 2013.
33. OWASP Foundation. Session Management Decision Graph and Security Risks Analysis. 2024. URL: https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html (дата звернення: 09.10.2025).

34. Rice S.M., et al. Online and social networking interventions for the treatment of depression in young people: a systematic review. *Journal of Medical Internet Research*. 2014. Vol. 16. No. 9. e215.
35. Mayo Clinic Staff. Support groups: Make connections, get help and find community for mental health support. 2024. URL: <https://www.mayoclinic.org/healthy-lifestyle/stress-management/in-depth/support-groups/art-20044655> (дата звернення: 09.10.2025).
36. Mozilla Web Security Group. Guidelines for Securing Web Applications and Session Cookies. 2024. URL: https://infosec.mozilla.org/guidelines/web_security (дата звернення: 10.10.2025).
37. National Institute of Mental Health (NIMH). Technology and the Future of Mental Health Treatment. 2024. URL: <https://www.nimh.nih.gov/health/topics/technology-and-the-future-of-mental-health-treatment> (дата звернення: 10.10.2025).
38. Eysenbach G. The role of the Internet in patient-centred care and health education. *Health Education Research*. 2012. Vol. 15. No. 2. pp. 171–187.
39. Pew Research Center. Social Media and Health Information: Trust, Engagement and Patient Community Dynamics. 2024. URL: <https://www.pewresearch.org/internet/2024/02/01/how-americans-use-social-media-for-health-information/> (дата звернення: 19.10.2025).
40. Privacy International. The Importance of Anonymity and Pseudonymity for Online Mental Health Support and Privacy Rights. 2024. URL: <https://privacyinternational.org/long-read/digital-mental-health-balancing-anonymity-and-security> (дата звернення: 21.10.2025).
41. Yang, H., Cao, J., Wan J. A Large-Scale Image Repository for Automated Pavement Distress Analysis. *Sci Data*. 2025. С. 1-12.
42. Olisah C. C. Minimizing Separability: A Comparative Analysis of Illumination Compensation Techniques in Face Recognition. *I.J. Information Technology and Computer Science*. 2017. №5. С. 40–51.

43. De Choudhury M., De S. Mental health discourse on reddit: Self-disclosure, subjectivity, and anonymity. *Proceedings of the International AAAI Conference on Web and Social Media*. 2014. Vol. 8. No. 1. pp. 71–80.
44. React Dev. Managing State in React. 2024. URL: <https://react.dev/learn/managing-state> (дата звернення: 29.10.2025).
45. Node.js Foundation. Node.js v20.x Documentation. 2023. URL: <https://nodejs.org/docs/latest-v20.x/api/> (дата звернення: 03.11.2025).
46. Гнатюк С. О. Принципи побудови адаптивних інтерфейсів медичних інформаційних систем. *Інформаційні технології в медицині*. 2022. № 5. С. 12–18.
47. Дорошенко А. Ю. Паралельні обчислення та хмарні сервіси в сучасних інформаційних системах. *Комп'ютерні системи та мережі*, 2022. № 14. С. 82–89.
48. American Psychological Association (APA). Guidelines for Ethical Practice in Telehealth and Online Support Groups: Privacy and Anonymity Factors. 2024. URL: <https://www.apa.org/practice/guidelines/telepsychology> (дата звернення: 17.11.2025).
49. W3C Web Accessibility Initiative (WAI). Designing Interactive Web Interfaces for Accessibility in Healthcare Systems. 2024. URL: <https://www.w3.org/WAI/design-develop/> (дата звернення: 21.11.2025).
50. Wang W. Cross-Technology Communications for Heterogeneous IoT Devices. *IEEE Transactions on Wireless Communications*, 2021. Vol. 19. P. 796–806.
51. Nielsen Norman Group. User Experience in Healthcare: Design Guidelines for Patient-Centered Digital Systems. 2024. URL: <https://www.nngroup.com/articles/ux-medical-devices/> (дата звернення: 24.11.2025).
52. Provost F., Fawcett T. Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. O'Reilly Media. 2013.
53. Мельник Р. А. Методи статистичної обробки користувацького контенту в інформаційних системах. *Вісник ЛНУ*. 2017. № 14. С. 55–61.
54. Kesselheim A.S., et al. Pharmaceutical policy in the United States in 2016: Many changes, one constant. *Health Affairs*. 2016. Vol. 35. No. 10. pp. 1795–1803.
55. Гірник М. А. Алгоритми ранжування повідомлень на спеціалізованих форумах. *Штучний інтелект*, 2023. Вип. 2. С. 55–63.

56. Powell J., et al. The Internet as a vehicle for health information: Patterns of usage and challenges. *Health Informatics Journal*. 2013. Vol. 19. No. 1. pp. 3–15.
57. Pressman R. *Software Engineering: A Practitioner's Approach*. McGraw-Hill. 2014. 8th ed
58. Martin R.C. *Clean Architecture: A Craftsman's Guide to Software Structure and Design*. Prentice Hall. 2017.
59. Atlassian Corporation. *The Agile Coach: Modern Software Development Methodologies and DevOps Integration*. 2024. URL: <https://www.atlassian.com/agile> (дата звернення: 29.11.2025).
60. Greenhalgh T., et al. Beyond adoption: a new framework for theorizing and evaluating nonadoption, abandonment, and challenges to scale-up. *Journal of Medical Internet Research*. 2017. Vol. 19. No. 11. e367.
61. Microsoft Corporation. *Cloud Design Patterns and Distributed Systems Administration Guide*. 2024. URL: <https://learn.microsoft.com/en-us/azure/architecture/patterns/> (дата звернення: 29.11.2025).
62. Bass L., Clements P., Kazman R. *Software Architecture in Practice*. 4th ed. Addison-Wesley Professional, 2021. 464 p.
63. U.S. Department of Health & Human Services. *Health Information Privacy: Security Standards for the Protection of Electronic Protected Health Information*. 2024. URL: <https://www.hhs.gov/hipaa/for-professionals/security/index.html> (дата звернення: 30.11.2025).
64. Fowler M. *Refactoring: Improving the Design of Existing Code*. 2nd ed. Addison-Wesley, 2020. 448 p.
65. Node.js Foundation. *Building Microservices with Node.js and Express: Scalability and Containerization Patterns*. 2024. URL: <https://nodejs.org/en/learn/asynchronous-work/event-loop-timers-and-nexttick> (дата звернення: 30.11.2025).
66. Andress J. *The Basics of Information Security: Understanding the Fundamentals of InfoSec*. Syngress. 2019.

ДОДАТОК А (обов'язковий)

ЛІСТИНГ ПРОГРАМНОГО КОДУ СЕРВЕРНОГО МОДУЛЯ ОБРОБКИ ДАНИХ ТА МЕТОДІВ СТАТИСТИЧНОЇ АГРЕГАЦІЇ

Модуль «Застосування методу структурно-статистичної агрегації для моніторингу користувацької активності та динамічного формування профілів учасників форуму».

```
const express = require('express');
const mysql = require('mysql2/promise');
const bcrypt = require('bcryptjs');
const jwt = require('jsonwebtoken');
require('dotenv').config();

const app = express();
app.use(express.json());

const pool = mysql.createPool({
  host: process.env.DB_HOST,
  user: process.env.DB_USER,
  password: process.env.DB_PASS,
  database: process.env.DB_NAME,
  waitForConnections: true,
  connectionLimit: 10
});

function authenticateToken(req, res, next) {
  const authHeader = req.headers['authorization'];
  const token = authHeader && authHeader.split(' ')[1];
  if (token == null) return res.status(401).json({ message:
'Unauthorized' });

  jwt.verify(token, process.env.JWT_SECRET, (err, user) => {
    if (err) return res.status(403).json({ message: 'Forbidden'
});
    req.user = user;
    next();
  });
}

app.post('/auth/register', async (req, res) => {
  const { username, email, password } = req.body;
  const password_hash = await bcrypt.hash(password, 10);
  const [result] = await pool.query(
    'INSERT INTO users (username, email, password_hash) VALUES
(?, ?, ?)',
    [username, email, password_hash]
  );
});
```

```

    const token = jwt.sign({ id: result.insertId },
process.env.JWT_SECRET);
    res.json({ token });
  });

app.post('/api/posts', authenticateToken, async (req, res) => {
  const { title, content, anonymous } = req.body;
  const [result] = await pool.query(
    'INSERT INTO posts (user_id, title, content, anonymous)
VALUES (?, ?, ?, ?)',
    [req.user.id, title, content, anonymous ? 1 : 0]
  );
  res.json({ id: result.insertId });
});

app.get('/api/user/stats', authenticateToken, async (req, res) => {
  try {
    const userId = req.user.id;
    const [[profileRow]] = await pool.query(
      'SELECT username, email, created_at, bday FROM users
WHERE id = ?',
      [userId]);

    const [[postCountRow]] = await pool.query(
      'SELECT COUNT(*) as post_count FROM posts WHERE user_id
= ?',
      [userId]);

    const [[replyCountRow]] = await pool.query(
      'SELECT COUNT(*) as reply_count FROM replies WHERE
user_id = ?',
      [userId]);

    const [userPosts] = await pool.query(
      `SELECT id, title, content, created_at, anonymous
FROM posts WHERE user_id = ?
ORDER BY created_at DESC LIMIT 15`,
      [userId]);

    res.json({
      profile: profileRow,
      postCount: postCountRow.post_count,
      replyCount: replyCountRow.reply_count,
      userPosts: userPosts
    });
  } catch (err) {
    res.status(500).json({ message: 'Error fetching user
statistics' });
  }
});

app.listen(process.env.PORT || 3000);

```

ДОДАТОК Б
(обов'язковий)

КОПІЯ ОПУБЛІКОВАНОЇ НАУКОВОЇ СТАТТІ

- 1) Недашківський Р.В. Форум-орієнтована інформаційна система з використання психіатричних препаратів. Філософські виміри техніки: *Збірник тез IV Міжнародної наукової конференції молодих учених та здобувачів вищої освіти*, 26 листопада 2025 р. С. 100–101

*IV Міжнародна конференція молодих учених та здобувачів вищої освіти
«Філософські виміри техніки»*

досвідом, слугує містком між медичними фактами та реальним життям пацієнтів, підвищуючи прихильність до терапії. Незважаючи на перевагу у психосоціалній підтримці, ключове обмеження системи полягає у необхідності жорсткої модерації та валідації інформації через ризик поширення необ'єктивного або небезпечного досвіду.

***Ключові слова:** форум, інформаційна система, обмін досвідом, валідація та безпека інформації*

Roman Nedashkivsky

*Scientific supervisor: Dmytro Medzaty, Ph.D., Assoc. Prof.
Khmelnitskyi National University, Ukraine*

FORUM-ORIENTED INFORMATION SYSTEM ON THE USE OF PSYCHIATRIC DRUGS

В сучасному суспільстві проблеми ментального здоров'я стають все більш поширеними, а доступ до достовірної та зрозумілої інформації про лікування є критично важливим [1]. Психіатричні препарати часто оточені міфами, стигмою та упередженнями, що може призводити до нерозуміння їхньої дії, небажання приймати призначене лікування або навіть самолікування. Потреба у форум-орієнтованій інформаційній системі з використання психіатричних препаратів виникає з кількох ключових аспектів. По-перше, медикаментозне лікування психічних розладів є складною темою, яка вимагає від пацієнтів та їхніх близьких глибокого розуміння особливостей дії, можливих побічних ефектів, схем прийому та взаємодії з іншими речовинами. По-друге, незважаючи на існування офіційних медичних джерел, пацієнти часто шукають досвід «з перших рук» – інформацію від тих, хто вже приймав або приймає певний препарат. Саме тут форум-орієнтований підхід набуває особливої значущості, оскільки він створює безпечний простір для обміну персональним досвідом, обговорення ефективності та побічних реакцій, які можуть відрізнятися у різних людей. Така система може стати містком між сухою медичною інформацією та реальним життям пацієнта, допомагаючи знизити рівень тривоги перед початком лікування, а також підвищити прихильність до терапії завдяки підтримці спільноти. Крім того, наявність централізованої, але багаторівневої (від медичної до досвіду пацієнтів) інформації може слугувати додатковим інструментом для фахівців, які можуть бачити загальні тенденції обговорень та типові проблеми, з якими стикаються їхні пацієнти, що в кінцевому підсумку сприяє підвищенню якості психіатричної допомоги в цілому.

Створення форум-орієнтованої інформаційної системи з використання психіатричних препаратів вимагає інтеграції декількох ключових модулів, кожен з яких реалізує важливі функції, працюючи разом у єдиному, цілісному інформаційному середовищі. Наріжним каменем системи є модуль бази знань та довідника препаратів, чия основна функція – надання достовірної, структурованої медичної інформації. Цей модуль містить детальні профілі препаратів, що охоплюють активну речовину, механізми дії, показання, дозування та, що критично важливо, потенційні побічні ефекти та взаємодії. Додаткова функція цього модуля – забезпечення глосарію термінів для спрощення складної медичної лексики для широкої аудиторії.

«Форум-орієнтованість» системи визначає модуль форуму та спільноти. Його ключова функція – створення безпечного простору для соціальної взаємодії та обміну досвідом. Це реалізується через організацію тематичних розділів (форумів) за категоріями препаратів або проблемами, що дозволяє користувачам ефективно здійснювати обмін досвідом та отримувати підтримку від однодумців. Для підтримки порядку та безпеки, невід'ємною функцією цього модуля є модерація, яка жорстко контролює вміст, запобігаючи поширенню небезпечних медичних порад та забезпечуючи підтримуючу атмосферу.

*IV Міжнародна конференція молодих учених та здобувачів вищої освіти
«Філософські виміри техніки»*

Ефективний доступ до масиву даних забезпечує модуль інтерактивного пошуку та фільтрації. Його функції спрямовані на оперативне знаходження релевантної інформації – від розширеного пошуку за назвами ліків чи симптомами до фільтрації відгуків користувачів за діагнозом, віком або тривалістю лікування. Це дозволяє користувачам швидко відсортувати персональний досвід, роблячи його більш значущим. Додаткова важлива функція – порівняння препаратів, що надає візуальні інструменти для оцінки відмінностей між кількома варіантами лікування.

Критичну важливість має модуль безпеки та валідації інформації. Його основні функції – забезпечення надійності та стичності. Впровадження застережень про медичні поради на кожній сторінці є обов'язковим, підкреслюючи, що система не замінює консультації лікаря. Верифікація інформації гарантує, що дані у базі знань відповідають офіційним джерелам, тоді як забезпечення конфіденційності та анонімності заохочує користувачів до відкритого обговорення чутливих тем. Доповнює цю структуру модуль інформаційного сповіщення, який виконує функцію інформування про оновлення безпеки (наприклад, відкликання препаратів) та дозволяє користувачам налаштувати спостереження за обговореннями для отримання своєчасних новин та відповідей.

Форум-орієнтована інформаційна система з використання психіатричних препаратів має низку суттєвих переваг у порівнянні з традиційними джерелами, такими як виключно офіційні медичні сайти або закриті наукові бази даних, але водночас має й певні обмеження. Головна перевага полягає у людиноцентричному підході та соціальній складовій – на відміну від «сухих» медичних текстів, які лише констатують факти, ця система пропонує унікальну функцію обміну реальним досвідом через форум. Це створює відчуття спільноти та підтримки, що є вкрай важливим для пацієнтів із ментальними розладами, оскільки знижує рівень стигматизації та ізоляції. Пацієнти можуть знайти відповіді на питання про суб'єктивні відчуття, які рідко описуються в інструкціях, наприклад, як саме препарат впливає на сон, апетит чи концентрацію у повсякденному житті, що є значною перевагою перед монотонними клінічними описами в офіційних ресурсах. Крім того, така система може бути динамічнішою у відображенні нових тенденцій у використанні препаратів порівняно з часом оновлення офіційних джерел.

Водночас, ключове обмеження форум-орієнтованої системи – достовірність інформації. На відміну від перевірених наукових публікацій чи офіційних державних реєстрів ліків, контент на форумі є переважно досвідом, що може бути необ'єктивним, спотвореним або навіть небезпечним. Існує високий ризик поширення помилкових уявлень про схеми лікування, дозування або взаємодію, особливо якщо система не має жорсткої та кваліфікованої медичної модерації, що є значним недоліком порівняно з професійними медичними порталами. Також ця система не є інструментом для персоналізованого лікування; вона надає загальну інформацію та досвід, який не може враховувати індивідуальні біологічні та клінічні особливості кожного користувача, що вимагає постійного наголосу на необхідності консультації з лікарем. Таким чином, хоча система є інструментом для психосоціальної підтримки та отримання «живої» інформації, її дані мають сприйматися з високою дозою критичності.

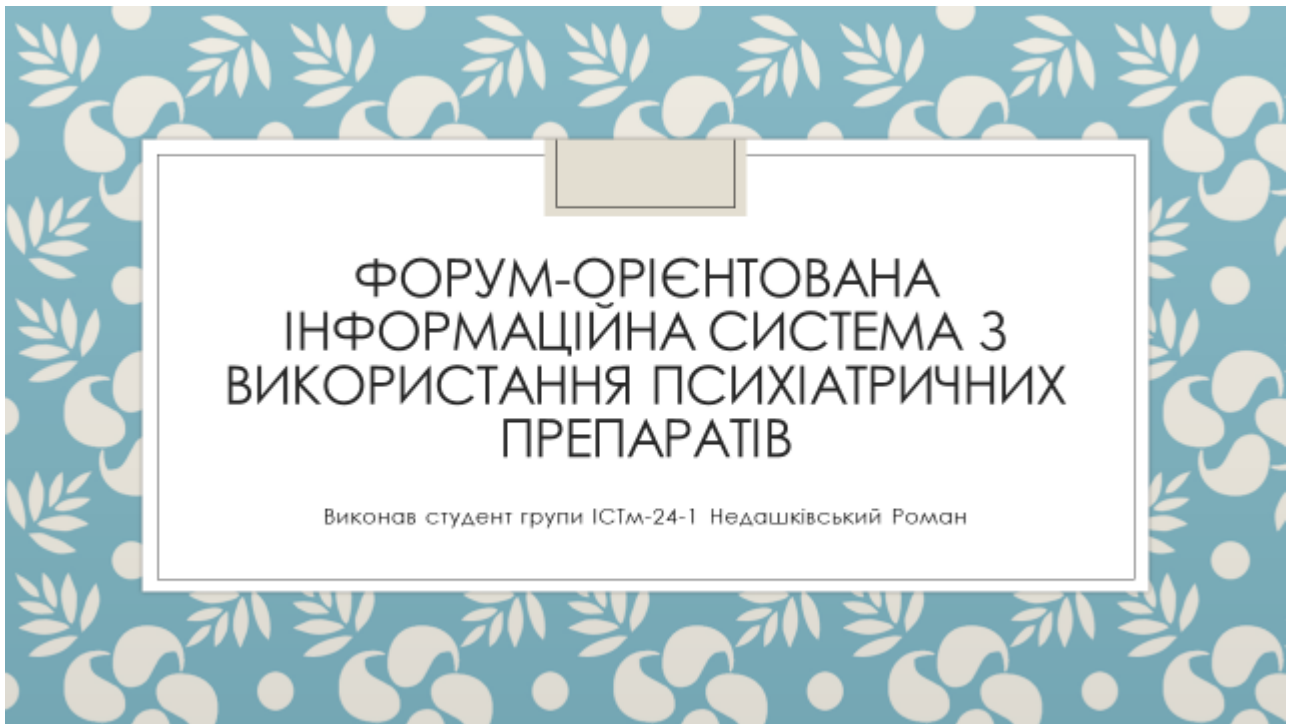
Джерела та література

1. Espinoza García A. D. Early detection of mental health problems in university students. *Revista de la Facultad de Medicina Humana*. 2024. Vol. 24, no. 1.
2. Internalized Stigma and Adherence to Treatment among Outpatients with Mental Illness / F. Kazemi et al. *Iranian Journal of Psychiatry and Behavioral Sciences*. 2024.

ДОДАТОК В

(обов'язковий)

ПРЕЗЕНТАЦІЙНІ МАТЕРІАЛИ



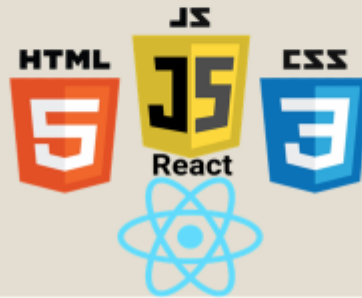
Мета кваліфікаційної роботи

- Метою кваліфікаційної роботи є підвищення ефективності обміну знаннями щодо психіатричних препаратів шляхом створення форум-орієнтованої інформаційної системи, яка забезпечує можливість структурованого обміну досвідом, моніторингу активності користувачів та підтримки актуальної бази знань.



Front-end частина форуму

- Весь проект побудований на HTML/CSS/JS, але для забезпечення динамічності був обран React. Він відповідає за інтерактивне управління станом інтерфейсу та автоматичне оновлення статистики користувача в режимі реального часу. Завдяки використанню цього фреймворку та компонентного підходу, система миттєво реагує на дії користувача (публікації, відгуки, запити), перераховуючи показники активності без перезавантаження сторінки.



Back-end частина форуму

- Back-end частина системи реалізована на платформі Node.js та з використанням Express.js, це забезпечує проєкту високу швидкість обробки асинхронних запитів. Node.js потрібно для забезпечення високої швидкості обробки асинхронних запитів та підтримки стабільної роботи системи під навантаженням завдяки своїй неблокуючій архітектурі. Express.js виступає у ролі інтелектуального посередника (middleware), який здійснює маршрутизацію, валідацію даних та контроль доступу за допомогою JWT-токенів.



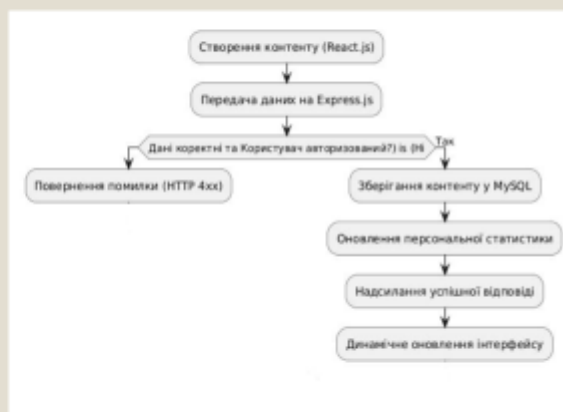
СУБД MySQL

- MySQL потрібен для безпечного зберігання облікових даних користувачів (їх логінів, імен та зашифрованих паролів), а також для систематизації та архівації всіх повідомлень на форумі. Вибрана саме MySQL тому що відома своєю високою швидкістю обробки стандартних запитів на читання та запис, що дозволяє миттєво завантажувати історію листування та забезпечує її стабільну роботу системи навіть при великій кількості одночасних обговорень користувачів.



Метод структурно-статистичної агрегації даних

- Метод структурно-статистичної агрегації даних реалізує повний цикл обробки контенту: від перехоплення транзакції в Express.js до її фіксації в MySQL та миттєвої візуалізації в React.js. Метод автоматизує перетворення дій користувача на аналітичні метрики (лічильники активності, динаміка постів), забезпечуючи реляційну цілісність бази даних та оновлення інтерфейсу в реальному часі без перезавантаження сторінок.



Авторизація користувача форуму

ВХІД

Email
you@example.com

Пароль
Ваш пароль

Запам'ятати мене

Увійти

◦ Реєстрація та вхід потрібні для створення персонального профілю користувача, до якого прив'язується вся його подальша статистика. Система безпечно зберігає логіни та паролі в MySQL, а Express.js перевіряє права доступу, щоб ніхто сторонній не міг зайти в чужий акаунт. Це дозволяє кожному користувачу мати власну історію обговорень, бачити свій рівень активності на форумі та зберігати анонімність при спілкуванні на чутливі медичні теми

РЕЄСТРАЦІЯ

Ім'я та Прізвище
Ім'я Прізвище

Дата народження
ДДММРРРР

Email
you@example.com

Пароль
Придумайте пароль

Зареєструватися

Написання повідомлення на форумі

◦ Після відправки тексту користувачем, основне навантаження лягає на MySQL, воно забезпечує надійне збереження повідомлення у реляційній таблиці та автоматично підтримує зв'язок із профілем автора через механізм зовнішніх ключів. Саме на рівні СУБД реалізується контроль цілісності: система дає гарантію, що статистика активності оновиться лише за умови успішного запису самого повідомлення.

Створити новий допис

Тема
Тема повідомлення

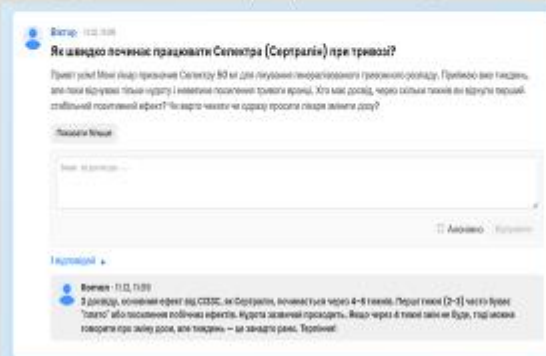
Текст
Опишіть проблему або питання...

Анонімно

Опублікувати

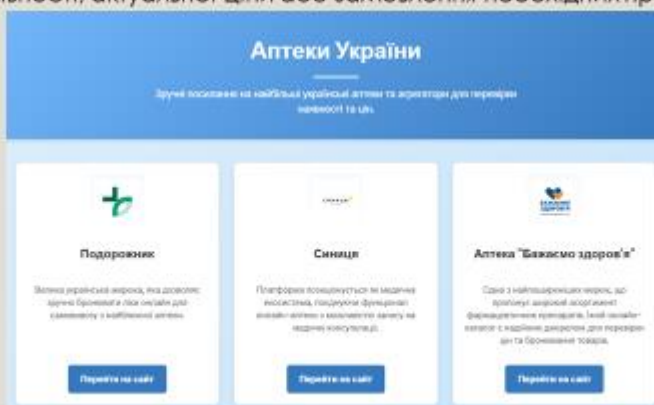
Повідомлення та відповіді на форумі

- На сайті всі обговорення відображаються у зручному хронологічному порядку. Завдяки налаштуванням MySQL найновіші повідомлення та коментарі завжди з'являються зверху, щоб користувачі бачили найсвіжіші поради. Під кожним постом можна одразу написати свою відповідь, яка миттєво прикріплюється до теми за допомогою бази даних.



Аптеки України

- Важливим елементом системи є довідковий каталог із прямими посиланнями на найбільші аптечні мережі України. Це дозволяє користувачеві не лише обговорити досвід лікування на форумі, а й одразу перейти до офіційних ресурсів для перевірки наявності, актуальної ціни або замовлення необхідних препаратів.



Персональна статистика користувача

- Персональний кабінет користувача є закритим сегментом системи, де за допомогою React.js реалізується візуальна презентація результатів методу структурно-статистичної агрегації. У цьому сегменті користувач бачить власні аналітичні метрики. Свою кількість дописів, відповідей, загальну кількість взаємодій та час перебування у спільноті. Ці дані не є статичними — вони динамічно підтягуються з MySQL через API-запити та миттєво оновлюються після кожної нової дії.



Висновок

- Інформаційна система P-PsyForum - це не просто сайт для спілкування, а продумане середовище, де технології працюють на користь пацієнта. Завдяки розробленому методу структурно-статистичної агрегації, кожен голос у спільноті стає частиною великої бази знань, яка допомагає людям орієнтуватися у складному світі психіатричних препаратів.
- Простими словами, цей проект допомагає людям з психічними розладами не заплутатися у лікуванні та знайти підтримку. Система автоматично збирає досвід сотень пацієнтів у зрозумілу статистику, щоб можна було одразу побачити, які поради перевірені часом, а які ліки можна замовити в аптеці поруч. Саме це і робить платформу потрібним для людей навігатором, де реальний досвід учасників перетворюється на зрозумілу допомогу для кожного, хто шукає шлях до одужання.



ДЯКУЮ ЗА УВАГУ!

Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Роман НЕДАШКІВСЬКИЙ

Співавтор:

Назва: Метод та форум-орієнтована інформаційна система з використання психіатричних препаратів

Експерт: Дмитро МЕДЗАТИЙ

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1: 4.8%

Коефіцієнт подібності 2: 1.3%

Мікропробіли: 2

Заміна букв: 2

Інтервали: 0

Білі знаки: 1

Дата створення звіту: 2025-12-21 15:01:49.0

Після аналізу Звіту подібності констатую наступне:

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.

Обґрунтування:

2025-12-21

Дата



Доцент Андрій Нічепорук

експерт

Anti-Plagiarism (UA) v-15.284 Educational

The maximum coincidence with one document 2.0%

Dictionaries check: en_US, ru_RU, ua_UA. Errors in the documents: 12%

ID: 268107 Title: МКР Метод та форум-орієнтована інформаційна система з використання психіатричних препаратів Added in a DB: 2025-12-21 Authors: Роман НЕДАШКІВСЬКИЙ Heads: Дмитро МЕДЗАТИЙ Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	116802	736	4914 (4%)	56 (8%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Роман НЕДАШКІВСЬКИЙ

Тема: Метод та форум-орієнтована інформаційна система з використання психіатричних препаратів

Спеціальність: 126 «Інформаційні системи та технології»

Обсяг дипломної роботи:

Кількість сторінок записки _____

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи є розроблення методу та форум-орієнтованої інформаційної системи з використання психіатричних препаратів, де люди описують свій досвід дії препарату, з чим комбінують, які дози приймають і з якими побічними реакціями мають справу.

2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню.

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому розділі проведений огляд відомих методів та рішень. У другому розділі виконано проектування форум-орієнтованої інформаційної системи з використання психіатричних препаратів. У третьому розділі розроблено метод та алгоритм функціонування форум-орієнтованої інформаційної системи з використання психіатричних препаратів. У четвертому розділі описано реалізацію форум-орієнтованої інформаційної системи з використання психіатричних препаратів.

4. Позитивні сторони роботи:

5. Негативні сторони роботи: мало уваги приділено формалізації методу

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно з діючими стандартами оформлення документації.

7. Відгук про роботу в цілому: Робота виконана на середньому науково-технічному рівні.

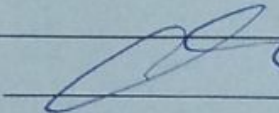
8. Інші зауваження: _____

9. Оцінка дипломної роботи: добре/С (80).

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи)

Олександр Барнак, з/в. керівн. КМУ

“ ” _____ 2025 р.

 (підпис)

Зав. кафедри КПС
д-р. філософії Ользі ПАВЛОВІЙ

Роман НЕДАШКІВСЬКИЙ

ПБ здобувача вищої освіти

ФІТ, 2 курсу, групи ІСТм-24-1

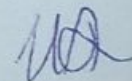
ЗАЯВА

З правилами чинного Положення про систему забезпечення академічної доброчесності у Хмельницькому національному університеті, згідно з яким виявлення академічного плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту і застосування заходів академічної відповідальності, ознайомлений (а). Про використання спеціалізованих програмних засобів (СПЗ) StrikePlagiarism та Anti-Plagiarism для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність академічного плагіату оповіщений (а). Надаю університету право на передачу моєї роботи для обробки та збереження в базах даних СПЗ і використання роботи для виявлення академічного плагіату в інших роботах, які перевіряються СПЗ.

Також надаю свою згоду на обробку й збереження університетом моєї роботи в Інституційному репозитарії Хмельницького національного університету.

Робота надається для перевірки в електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

10 грудня 2025 року



РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи Метод та форум-орієнтована інформаційна система з використання психіатричних препаратів
 Автор Роман НЕДАШКІВСЬКИЙ
 Освітня програма Інформаційні системи та технології
 Рівень вищої освіти другий (магістерський)
 Спеціальність 126 Інформаційні системи та технології
 Науковий керівник: к.т.н., доцент Дмитро МЕДЗАТИЙ

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмним засобом(ами) комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	відповідає
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 2) окремі виявлені збіги є загальноживаними фразами або виразами, про що свідчить посилання системи на збіг з джерелами на один фрагмент речення;
- 3) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.
- 4) значна частина знайденого плагіату відноситься до списку використаних джерел

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості StrikePlagiarism, складає 4,82% і адресується до 28 першоджерела; та системою Anti-Plagiarism складає 2%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

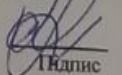
15.12.2025

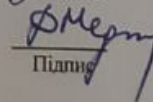
Завідувач кафедри

Гарант освітньої програми

Керівник кваліфікаційної роботи


Підпис


Підпис


Підпис

Ольга ПАВЛОВА
Ім'я, ПРІЗВИЩЕ

Ольга ПАВЛОВА
Ім'я, ПРІЗВИЩЕ

Дмитро МЕДЗАТИЙ
Ім'я, ПРІЗВИЩЕ