Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Медіасистеми та технології
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)
(рівень вищої освіти)

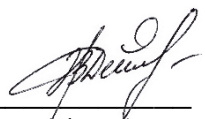
Дослідження процесу впровадження чат-боту
для покращення освітнього процесу
(тема)

Виконав:
здобувач 2 року навчання
групи ТЕМВм-24-1

Микита КАТКОВ
(власне ім'я, прізвище)

Спеціальність 186 Видавництво та поліграфія
(код і повна назва спеціальності)
Тип програми Освітньо-професійна
Освітня програма
Технології електронних мультимедійних видань

Керівник
доц. Павло КОЗУБ
(посада, власне ім'я, прізвище)

Допускається до захисту
Завідувач кафедри МСТ 
(підпис)
Жанна ДЕЙНЕКО
(власне ім'я, прізвище)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
Кафедра Медіасистеми та технології
Рівень вищої освіти другий (магістерський)
Спеціальність 186 Видавництво та поліграфія
Тип програми Освітньо-професійна
Освітня програма Технології електронних мультимедійних видань
(шифр і назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри МСТ _____
(підпис)
« 03 » листопада 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві *Каткову Микиті Сергійовичу*
(прізвище, ім'я, по батькові)

1. Тема роботи *Дослідження процесу впровадження чат-боту для покращення освітнього процесу*

затверджена наказом по університету від *03 листопада 2025 р. № 989 Ст*

2. Термін подання здобувачем роботи до екзаменаційної комісії *19 грудня 2025 р.*

3. Вихідні дані до роботи
Вимоги закладів вищої освіти до організації дистанційного навчання та елетронного документообігу робіт. Вихідні дані для оцінювання доцільності впровадження чат-бота.

4. Перелік питань, що потрібно опрацювати в роботі
Аналіз предметної області дистанційного навчання й проблем організації електронного документообігу робіт; Огляд та порівняльний аналіз існуючих LMS та підходів до використання чат-ботів у навчальному процесі; Формулювання припущень і критеріїв оцінювання доцільності застосування чат-бота; Розробка методики отримання та обробки експертних оцінок і результатів анкетування студентів; Постановка задачі; Визначення вимог, проєктування прототипу чат-бота; Формування висновків.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій
Вступ; Актуальність, мета, завдання та об'єкт дослідження; Особливості дистанційної та змішаної форми навчання; Аналіз LMS; Теоретичні основи та аналіз використання чат-ботів у навчанні; Формулювання припущень і критеріїв оцінювання; Архітектура та модель даних; Реалізація ключових сценаріїв чат-бота; Розробка методики; Постановка задач; Результати впровадження; Економічна частина; Висновки.

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім’я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	доц. Козуб П.А.		18.12.25
Економічна частина	доц. Потій О.О.		16.12.25

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Аналіз предметної області та рішень	03.11.25 – 05.11.25	виконано
2	Аналітичний огляд літератури та джерел	06.11.25 – 08.11.25	виконано
3	Аналіз дистанційної та змішаної форм навчання	09.11.25 – 12.11.25	виконано
4	Аналіз існуючих систем управління навчанням	13.11.25 – 16.11.25	виконано
5	Порівняльний аналіз використання чат-ботів	17.11.25 – 20.11.25	виконано
6	Обґрунтування припущень і визначення критеріїв оцінювання	21.11.25 – 24.11.25	виконано
7	Розробка методики оцінювання	25.11.25 – 28.11.25	виконано
8	Проведення оцінювання та обробка результатів	29.11.25 – 02.12.25	виконано
9	Проєктування архітектури, моделі даних та організації зберігання	03.12.25 – 05.12.25	виконано
10	Програмна реалізація прототипу, основних підсистем і сценаріїв взаємодії	06.12.25 – 07.12.25	виконано
11	Перевірка відповідності вимогам	08.12.25 – 10.12.25	виконано
12	Економічні розрахунки та оформлення	11.12.25 – 15.12.25	виконано

Дата видачі завдання 03 листопада 2025 р.

Здобувач _____
(підпис)

Керівник роботи _____ доц. Павло КОЗУБ
(підпис) (посада, власне ім’я, прізвище)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи містить 94 с., 13 табл., 32 рис., 2 дод., 30 джерел.

ЧАТ-БОТ У МЕСЕНДЖЕРІ, ДИСТАНЦІЙНЕ ТА ЗМІШАНЕ НАВЧАННЯ, АВТОМАТИЗАЦІЯ ОСВІТНЬОГО ПРОЦЕСУ, ЗДАЧА ТА ПЕРЕВІРКА НАВЧАЛЬНИХ РОБІТ, ЕЛЕКТРОННИЙ ДОКУМЕНТООБІГ, МОНІТОРИНГ УСПІШНОСТІ СТУДЕНТІВ, РОЛЬОВИЙ ДОСТУП, ЕКСПОРТ РЕЗУЛЬТАТІВ У ТАБЛИЧНИЙ ФОРМАТ (XLSX), ОЦІНЮВАННЯ ТА ЗВОРОТНИЙ ЗВ’ЯЗОК, ПРОТОТИПУВАННЯ ТА АПРОБАЦІЯ ПІДХОДУ.

Метою роботи є дослідження та апробація підходу до впровадження чат-бота у месенджері з підтримкою бот-інтерфейсів (із практичною орієнтацією на WhatsApp) шляхом розроблення і випробування прототипу, призначеного для автоматизації процесів подання, фіксації, перевірки та оцінювання навчальних робіт у закладі вищої освіти, а також формування рекомендацій щодо застосування й адаптації такого рішення в освітньому процесі.

Об’єктом дослідження є процес організації освітнього процесу у закладі вищої освіти в умовах дистанційної та змішаної форми навчання, зокрема електронний супровід взаємодії між студентами та викладачами під час подання, перевірки й обліку результатів навчальних робіт.

ABSTRACT

Explanatory note of qualified work has 94 p., 13 tabl., 32 fig., 2 app., 30 references.

CHAT-BOT IN MESSENGER, AUTOMATION OF THE LEARNING PROCESS, SUBMISSION AND VERIFICATION OF LEARNING WORK, ELECTRONIC DOCUMENTATION, MONITORING OF STUDENTS' SUCCESS, EXPORT OF RESULTS IN A SPREADSHEET FORMAT (XLSX), ROLE-BASED ACCESS.

The purpose of this work is to study and validate an approach to deploying a chatbot within a messenger that supports bot interfaces (with a practical focus on WhatsApp) by developing and testing a prototype intended to automate the submission, logging, review, and grading of academic assignments in a higher education institution, as well as to formulate recommendations for the use and adaptation of such a solution in the educational process.

The object of the research is the organization of the educational process in a higher education institution under distance and blended learning conditions, in particular the electronic support of interaction between students and instructors during the submission, review, and recording of assignment results.

ЗМІСТ

С.

ВСТУП 11
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ ДЛЯ АВТОМАТИЗАЦІЇ ЗДАЧІ НАВЧАЛЬНИХ РОБІТ 15
1.1 Особливості дистанційної та змішаної форми навчання у закладах вищої освіти 15
1.2 Аналіз існуючих систем управління навчанням (LMS) 18
1.3 Теоретичні основи використання чат-ботів у дистанційному навчанні 21
1.4 Порівняльний аналіз підходів до впровадження чат-ботів у дистанційному навчанні 23
1.5 Обґрунтування припущень щодо доцільності та ефективності впровадження чат-бота 24
2 РОЗРОБКА МЕТОДИКИ ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ ОРГАНІЗАЦІЇ ОСВІТНЬОГО ПРОЦЕСУ 26
2.1 Постановка задачі дослідження 26
2.2 Функціональні вимоги до системи 28
2.3 Нефункціональні вимоги до системи 30
3 ВИЗНАЧЕННЯ РЕКОМЕНДАЦІЙ ДЛЯ СТВОРЕННЯ ЧАТ-БОТУ 33
3.1 Методика оцінювання доцільності впровадження чат-бота 33
3.2 Обробка результатів експертного оцінювання викладачів 38
3.3 Обробка результатів анкетування студентів щодо доцільності використання чат-боту 45
3.4 Результати ефективності та доцільності чат-боту 49
4 СТВОРЕННЯ ЧАТ-БОТУ ДЛЯ ПОКРАЩЕННЯ ОСВІТНЬОГО ПРОЦЕСУ 51
4.1 Постановка задачі програмної реалізації та вимоги до системи 51
4.2 Архітектура програмного рішення та обґрунтування технологічного стеку 54

4.3 Проєктування моделі даних та організація зберігання інформації 59
4.4 Реалізація підсистеми доступу та керування користувачами 63
4.5 Реалізація основних процесів навчальної взаємодії 67
4.6 Тестування та перевірка відповідності поставленим задачам 74
5 ЕКОНОМІЧНА ЧАСТИНА 80
5.1 Характеристика науково-дослідного рішення 80
5.2 Етапи виконання НДР, їх трудомісткість та заробітна плата 81
5.3 Розрахунок одноразових витрат на розробку 83
ВИСНОВКИ 89
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ 92
ДОДАТОК А Приклади роботи чат-боту 95
ДОДАТОК Б Результати анкетування 98

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

API (Application Programming Interface) – набір правил, методів і протоколів, що забезпечують взаємодію між програмними компонентами (наприклад, між клієнтським інтерфейсом чат-бота та серверною частиною), а також доступ до функцій зовнішніх сервісів через запити.

БД (база даних) – структуроване сховище інформації, у якому зберігаються дані про користувачів, курси, завдання, подання, оцінки, статуси та службові журнали; забезпечує цілісність і відтворюваність навчових процесів.

СУБД (система керування базами даних) – програмний засіб для створення, зберігання, обробки та адміністрування баз даних; надає механізми запитів, транзакцій, індексації та контролю доступу.

SQL (Structured Query Language) – мова структурованих запитів, що використовується для отримання, додавання, оновлення та видалення даних у реляційних базах; застосовується під час формування вибірок для звітності та обробки станів подань.

SQLite (SQLite Database Engine) – вбудована реляційна СУБД, що зберігає дані у вигляді локального файлу; зручна для розгортання без окремого серверного компонента й придатна для навчових систем із помірним навантаженням.

LMS (Learning Management System) – система управління навчанням, що підтримує створення курсів, розміщення матеріалів, видачу завдань, збір робіт, оцінювання та формування журналу успішності (наприклад, Moodle, Google Classroom).

FSM (Finite State Machine) – модель керування станами діалогу, яка описує послідовність кроків взаємодії користувача з чат-ботом (введення даних, вибір дій, підтвердження операцій) і гарантує коректні переходи між етапами сценарію.

SMTP (Simple Mail Transfer Protocol) – протокол передавання електронної пошти; використовується для надсилання службових повідомлень, зокрема одноразових кодів підтвердження під час верифікації користувачів.

VPS (Virtual Private Server) – віртуальний приватний сервер, на якому може розгортатися серверна частина системи; забезпечує ізольовані ресурси, віддалений доступ і можливість налаштування середовища виконання.

UI (User Interface) – інтерфейс користувача; у контексті чат-бота включає структуру повідомлень, меню, кнопки, підказки та спосіб подання статусів, оцінок і результатів.

UX (User Experience) – користувацький досвід; характеризує зручність і зрозумілість взаємодії з системою, кількість кроків до результату, логіку сценаріїв, читабельність повідомлень і зниження помилок користувача.

ID (Identifier) – ідентифікатор; унікальне значення для однозначного розрізнення об’єктів (користувача, курсу, завдання, подання, файлу) у системі та базі даних.

PK (Primary Key) – первинний ключ; поле або набір полів у таблиці БД, що однозначно ідентифікує запис і використовується для зв’язування сутностей та забезпечення цілісності даних.

FK (Foreign Key) – зовнішній ключ; поле в таблиці БД, яке посилається на первинний ключ іншої таблиці та формує зв’язки між сутностями (наприклад, “завдання належить курсу”, “подання належить студенту”).

XLSX (Office Open XML Spreadsheet) – формат електронних таблиць, сумісний із Microsoft Excel та іншими табличними редакторами; застосовується для автоматичного формування відомостей і звітів за результатами перевірки робіт.

Excel (Microsoft Excel) – табличний процесор для перегляду, редагування й аналізу даних у форматах електронних таблиць; використовується для роботи з експортованими відомостями успішності.

ІТ (інформаційні технології) – сукупність методів і програмно-технічних засобів для збирання, зберігання, обробки та передавання інформації; у роботі – технологічна основа автоматизації навчальних процесів.

НДР (науково-дослідна робота) – комплекс дослідницьких і прикладних заходів, спрямованих на отримання обґрунтованих результатів і практичного рішення; включає аналіз, експеримент, проєктування та оцінювання ефективності.

ЄСВ (єдиний соціальний внесок) – обов’язковий платіж на загальнообов’язкове державне соціальне страхування; у роботі враховується під час розрахунку витрат у економічній частині.

WhatsApp (месенджер) – платформа обміну повідомленнями, що підтримує інтеграцію з чат-ботами через програмні інтерфейси; може використовуватися як канал доступу до навчальної системи для подання робіт, отримання результатів і сповіщень.

Viber (месенджер) – платформа обміну повідомленнями з підтримкою ботів та сервісних інтеграцій; може виступати альтернативним каналом взаємодії з навчальною системою за аналогічною логікою сценаріїв.

NureStudy Bot (інформаційна система у вигляді чат-бота) – розроблене в роботі рішення для автоматизації процесів здачі, перевірки та оцінювання навчальних робіт; забезпечує рольову взаємодію (студент/викладач), керування курсами й завданнями, облік подань і формування звітності у табличному форматі.

ВСТУП

У сучасних умовах, коли значна частина навчального процесу реалізується у дистанційній або змішаній формі, особливо гостро постає потреба в ефективних електронних засобах організації взаємодії між студентами та викладачами. Цифрові інструменти стають не додатковим, а обов’язковим елементом освітнього середовища.

Навчальний процес у закладах вищої освіти дедалі більше спирається на використання електронних ресурсів, систем управління навчанням, хмарних сховищ, електронної пошти та різноманітних онлайн-сервісів. Масове поширення мобільних пристроїв і месенджерів призвело до того, що значна частина повсякденної комунікації студентів і викладачів відбувається саме в цифровому середовищі, а не в аудиторіях [10]. У цих умовах особливої ваги набуває питання організації зручного, надійного й прозорого електронного супроводу навчальних робіт.

Традиційно здача лабораторних, розрахунково-графічних, модульних та інших студентських робіт здійснювалася в паперовому вигляді або за допомогою набору розрізнених інструментів: електронної пошти, особистих повідомлень у месенджерах, завантаження файлів у різні файлові сховища тощо. Такий підхід створює значне навантаження як на студентів, так і на викладачів. Студенти вимушені запам’ятовувати різні канали комунікації з кожним викладачем, стежити за дедлайнами, які часто озвучуються усно чи публікуються в різних чатах, а також зберігати локальні копії надісланих файлів, щоб за потреби повторно надати роботу. Викладачі, у свою чергу, витрачають багато часу на пошук робіт у поштових скриньках і чатах, ручне формування відомостей, ведення власних таблиць з оцінками й статусами перевірки.

Особливо відчутними ці проблеми стають саме в дистанційному та змішаному форматах навчання, коли практично всі етапи роботи з навчальними завданнями від видачі до захисту – відбуваються онлайн. За відсутності

централізованого механізму здачі та перевірки робіт викладачеві складно отримати цілісну картину успішності групи, швидко виявити студентів, які не здали завдання, або проконтролювати, чи враховані всі перездачі. Для закладу освіти ускладнюється збір агрегованої статистики, аналіз дотримання навчальних планів і контроль якості освітнього процесу [11]. У результаті значна частина часу, який міг би бути використаний на методичну роботу, консультації або розробку навчальних матеріалів, витрачається на рутинні операції.

Актуальність дослідження зумовлена необхідністю оптимізації інформаційних потоків у закладі вищої освіти та зменшення часових витрат учасників освітнього процесу, особливо в умовах широкого використання дистанційної та змішаної форми навчання. Сучасні студенти щоденно користуються месенджерами, зокрема такими як WhatsApp і Viber, які забезпечують швидкий обмін повідомленнями, підтримку взаємодії через чат-ботів та зручну роботу з файлами [12]. Логіка організації взаємодії (ролі користувачів, стани завдань, повідомлення, обмін файлами та сповіщення) є подібною для більшості платформ, тому підхід, що розглядається в роботі, може бути адаптований і перенесений на різні месенджери з підтримкою чат-ботів.

Наявні системи управління навчанням (LMS), такі як Moodle та інші, хоч і містять засоби для завантаження файлів і виставлення оцінок, часто вимагають розгорнутої інфраструктури, окремих веб-інтерфейсів і додаткового навчання користувачів [13]. У невеликих навчальних підрозділах або при створенні нових курсів їх упровадження може бути економічно невиправданим. Крім того, частина студентів все одно дублює комунікацію через месенджери, уточнюючи дедлайни чи статус перевірки, що підтримує фрагментацію інформації. Тому актуальним є створення “легкого” інструмента, інтегрованого безпосередньо у звичний канал комунікації, та визначення умов його впровадження в навчальний процес.

Основною проблемою, на вирішення якої спрямована кваліфікаційна робота, є відсутність єдиного та зручного підходу до організації повного циклу роботи з навчальними завданнями: від постановки завдання до здачі,

перевірки, оцінювання, можливих перездач і формування підсумкової звітності. На стороні студентів можна виокремити такі типові проблеми:

– відсутність централізованого переліку активних завдань, їх опису та дедлайнів;

– складність контролю власного прогресу за курсом, особливо при великій кількості предметів;

– невизначеність щодо того, чи отримав викладач надіслану роботу, чи не загубився файл серед інших повідомлень;

– відсутність єдиного місця, де можна переглянути історію надісланих робіт, отримані оцінки та коментарі.

Для викладачів характерні інші, але не менш суттєві проблеми:

– необхідність ручної обробки великої кількості повідомлень у різних каналах комунікації;

– дублювання операцій (перевірка, виставлення оцінки, перенесення оцінки у відомість або електронний журнал);

– складність відстеження, які студенти здали роботу вчасно, а хто має заборгованість;

– відсутність швидкого доступу до статистики успішності групи та її експортування.

Проблема, що розглядається в роботі, полягає не лише в технічному створенні чат-бота, а й у визначенні особливостей його впровадження в освітнє середовище: формуванні вимог, моделюванні сценаріїв взаємодії, забезпеченні зручності, безпеки та відтворюваності даних під час роботи з навчальними завданнями.

Об’єктом дослідження є організація освітнього процесу у закладі вищої освіти в умовах дистанційної та змішаної форми навчання.

Предметом дослідження є підходи, моделі та програмні механізми впровадження чат-бота у месенджерах із підтримкою бот-інтерфейсів для автоматизації процесів здачі, перевірки, оцінювання та обліку навчальних робіт.

Метою дослідження є процес впровадження чат-бота у месенджері з підтримкою бот-інтерфейсів (з практичною орієнтацією на WhatsApp), розроблення прототипу для перевірки запропонованого підходу та формування рекомендацій щодо його застосування й адаптації в навчальному процесі.

Досягнення поставленої мети передбачає аналіз існуючих підходів до організації електронної взаємодії в освіті, моделювання процесів здачі й перевірки робіт, визначення вимог і критеріїв оцінювання, проєктування архітектури та моделі даних, реалізацію прототипу чат-бота з використанням сучасних веб-технологій, а також експериментальну перевірку доцільності та ефективності запропонованого підходу.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ РІШЕНЬ ДЛЯ АВТОМАТИЗАЦІЇ ЗДАЧІ НАВЧАЛЬНИХ РОБІТ

1.1 Особливості дистанційної та змішаної форми навчання у закладах вищої освіти

Дистанційна та змішана форми навчання стали невід'ємною складовою освітнього процесу у закладах вищої освіти. Вони базуються на широкому використанні інформаційно-комунікаційних технологій, що забезпечують можливість взаємодії між викладачем і студентом без обов'язкової фізичної присутності в аудиторії [14]. Для таких форматів характерним є перенесення більшості елементів навчального процесу в електронне середовище: доведення навчальних матеріалів, постановка завдань, консультації, контроль знань та зберігання результатів навчальної діяльності.

Ключовою особливістю дистанційного навчання є розведення у часі та просторі діяльності викладача й студента. Значна частина взаємодій відбувається асинхронно: студент отримує доступ до матеріалів, самостійно опрацьовує їх, виконує завдання та надсилає результати для перевірки у зручний час у межах встановлених дедлайнів. Викладач, відповідно, перевіряє роботи та надає зворотний зв'язок у власному робочому графіку. Для організації таких процесів необхідні надійні канали електронної комунікації та інструменти, які дозволяють фіксувати кожен крок – від видачі завдання до виставлення оцінки.

Змішана форма навчання поєднує елементи традиційних очних занять з дистанційною взаємодією. Частина лекцій, практичних або лабораторних занять проводиться в аудиторіях, інша частина – за допомогою онлайн-платформ. При цьому значна кількість навчальних завдань, особливо лабораторних і індивідуальних робіт, все одно здається у віддаленому режимі через електронні сервіси. Для викладача це означає необхідність одночасно

керувати як очною активністю груп, так і онлайн-компонентами курсів, що суттєво ускладнює перевірку та облік.

Ще однією специфічною особливістю є зростання кількості каналів комунікації. У практиці закладів вищої освіти паралельно використовуються системи управління навчанням (LMS), електронна пошта, відеоконференції, хмарні сховища, а також чати груп у месенджерах (зокрема WhatsApp, Viber та інші платформи зі схожою логікою обміну повідомленнями й файлами). З одного боку, це створює гнучкі можливості для взаємодії, з іншого – призводить до фрагментації інформації [15]. Частина завдань може бути надіслана через LMS, частина – у вигляді файлів у чаті, окремі пояснення – у листах чи голосових повідомленнях. У результаті і студенти, і викладачі витрачають додатковий час на пошук потрібної інформації, уточнення дедлайнів та пошук уже надісланих робіт.

Для студентів у дистанційному й змішаному форматах зростає роль самоорганізації та самоконтролю. Вони мають планувати власний час, відстежувати дедлайни з кількох дисциплін, пам’ятати про варіанти завдань і вимоги різних викладачів. За відсутності централізованого інструменту, який би об’єднував інформацію про всі активні завдання та їх статуси, значно підвищується ризик пропуску термінів здачі або втрати файлів. Крім того, у студентів виникає потреба мати підтвердження факту здачі роботи й можливість переглянути історію взаємодії з викладачем щодо конкретного завдання.

Викладачі в умовах дистанційного та змішаного навчання стикаються з іншою групою викликів. Вони отримують значно більший обсяг електронних повідомлень, файлів і посилань, ніж у традиційному форматі. Особливо це відчутно під час сесії або при одночасному викладанні кількох курсів для різних груп. Для кожної роботи необхідно визначити студента, групу, дисципліну, тип завдання, варіант, дату здачі, а потім зберегти десь результати перевірки й оцінку. Якщо ці операції виконуються вручну, за допомогою електронних таблиць або особистих нотаток, зростає ймовірність помилок: пропуску окремих робіт, плутанини з варіантами, повторного запиту вже надісланих файлів тощо.

Важливим аспектом є й організація зворотного зв’язку. У дистанційному форматі студенти часто очікують отримати не лише оцінку, а й детальні коментарі викладача щодо своєї роботи, рекомендації з доопрацювання, пояснення помилок. Відповідно, система, що підтримує дистанційне або змішане навчання, має забезпечувати зручний механізм фіксації таких коментарів, можливість їх подальшого перегляду студентами, а також повторної здачі скоригованих робіт.

Ще однією суттєвою особливістю дистанційної та змішаної форми навчання є підвищені вимоги до надійності та безпеки інформаційних систем. Дані про навчальну діяльність студентів, їхні персональні дані, результати атестацій та екзаменів повинні зберігатися з дотриманням вимог законодавства та внутрішніх регламентів закладу освіти [16]. Використання випадкових хмарних сховищ, неавторизованих чатів або особистих акаунтів викладачів у різних сервісах ускладнює контроль доступу та може створювати ризики несанкціонованого розповсюдження інформації. Таким чином, дистанційна та змішана форми навчання у закладах вищої освіти характеризуються:

– перенесенням значної частини взаємодій у електронне середовище;

– асинхронністю виконання навчальних завдань та отримання зворотного зв’язку;

– множинністю каналів комунікації, що призводить до фрагментації інформації;

– зростанням вимог до самоорганізації студентів та обліку результатів діяльності викладачів;

– потребою в централізованих інструментах для зберігання, систематизації й аналізу даних про навчальні роботи;

– підвищеними вимогами до інформаційної безпеки та захисту персональних даних.

У сукупності ці особливості створюють об’єктивну потребу в спеціалізованих інформаційних системах, здатних забезпечити єдине

середовище для організації здачі, перевірки та обліку навчальних робіт. Одним із практичних підходів до реалізації такого середовища є розробка чат-бота в месенджері з підтримкою відповідних інтерфейсів взаємодії, інтегрованого в звичні для студентів та викладачів канали електронної комунікації, що й розглядається в подальших розділах даної роботи.

1.2 Аналіз існуючих систем управління навчанням (LMS)

Системи управління навчанням (Learning Management System, LMS) є базовим інструментом цифрової організації освітнього процесу в закладах вищої освіти. Вони забезпечують розміщення навчальних матеріалів, створення курсів, формулювання та приймання завдань, ведення електронного журналу, комунікацію між учасниками освітнього процесу та формування звітності. Серед найбільш поширених рішень можна виокремити Moodle та Google Classroom, які фактично стали де-факто стандартами для багатьох навчальних закладів.

Moodle є відкритою модульною платформою, яка надає розробникам і адміністраторам широкі можливості для налаштування структури курсів і логіки їх роботи. Кожен курс у Moodle має чітко визначену ієрархію елементів: теми (тематичні блоки), ресурси (файли, сторінки, посилання) та активності (завдання, тести, форуми, опитування тощо). Викладач може створювати різні типи завдань: роботи, що завантажуються у вигляді файлів; завдання, що виконуються безпосередньо в текстовому полі; тести з автоматичним оцінюванням; проєкти з груповою роботою. Для кожного завдання задаються дедлайни, критерії оцінювання, шкала балів, можливість повторної здачі, а також параметри відображення оцінок у журналі.

Характерною особливістю Moodle є система ролей і прав доступу. Викладач у межах свого курсу керує наповненням, створює завдання, переглядає здачі студентів, виставляє оцінки та коментарі. Студент отримує доступ до списку курсів, календаря подій, переліку активних завдань та

власного журналу оцінок. Типовий вигляд головної сторінки курсу в Moodle із тематичними блоками, переліком завдань і панеллю навігації (рис. 1.1).

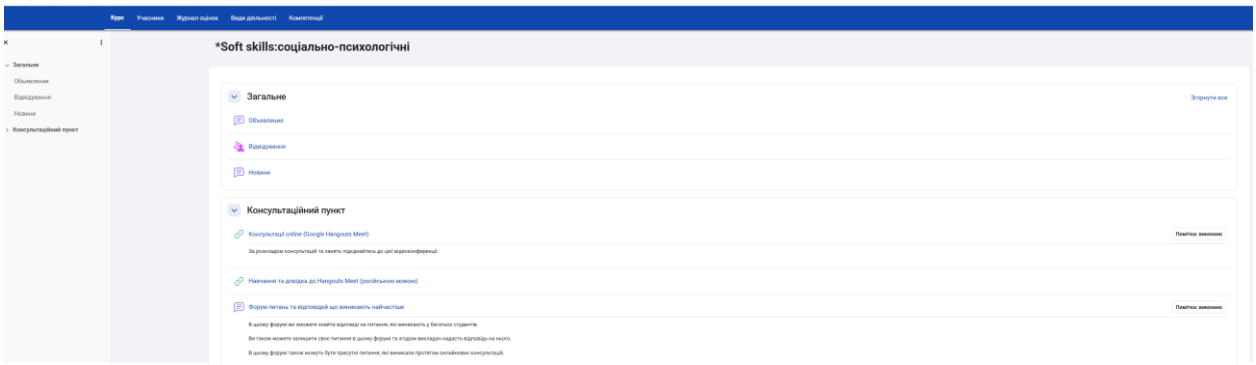


Рисунок 1.1 – Базова сторінка Moodle на прикладі dl.nure.ua

Важливим компонентом Moodle є електронний журнал, у якому зводяться результати всіх активностей курсу. Викладач може налаштувати вагу кожного завдання, групувати їх у категорії, експортувати оцінки у формат електронних таблиць, аналізувати прогрес окремих студентів і групи в цілому [13].

Суттєвою перевагою Moodle є можливість розширення функціональності за допомогою плагінів: модулів для інтеграції з відеоконференціями, системами перевірки на плагіат, зовнішніми сховищами файлів, аналітичними сервісами. Однак така гнучкість має і зворотний бік – потребу в кваліфікованому адмініструванні, регулярних оновленнях, налаштуванні резервного копіювання й моніторингу продуктивності сервера. Для кафедри або окремого викладача це означає залежність від центральної IT-служби університету та обмежені можливості оперативно змінювати конфігурацію під власні потреби.

Ще одним розповсюдженим рішенням є Google Classroom – хмарна система, тісно інтегрована з екосистемою Google Workspace (Google Drive, Google Docs, Google Meet тощо). Google Classroom орієнтований на максимально простий і зрозумілий інтерфейс. Викладач створює «клас», додає студентів за посиланням або кодом, формує стрічку оголошень і завдань [16]. Для кожного завдання можна вказати опис, прикріпити файли, шаблони

документів, посилання, а також задати термін здачі. Студенти бачать завдання у вигляді списку з датами, можуть відкривати прикріплені документи, виконувати їх безпосередньо в Google Docs або завантажувати власні файли. Приклад інтерфейсу створення завдання в Google Classroom наведено на рис. 1.2.

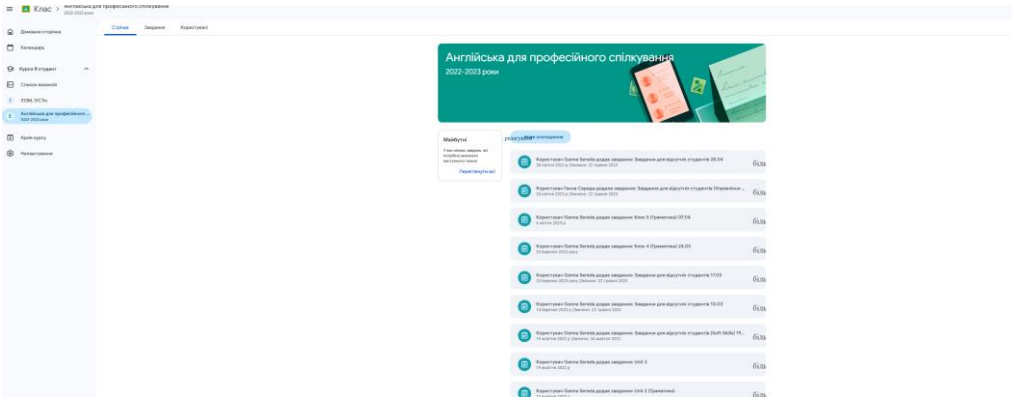


Рисунок 1.2 – Сторінка Google Classroom

Google Classroom надає базові можливості для оцінювання: викладач виставляє бали, може залишати коментарі до роботи, повертати завдання на доопрацювання, переглядати загальний журнал успішності. Журнал Classroom інтегрується з Google Таблицями, що полегшує експорт та подальший аналіз даних. Окремою перевагою є автоматичне створення папок у Google Drive для кожного завдання та студента, що спрощує організацію файлів. Разом із тим, така схема зберігання даних прив’язує навчальний процес до хмарної інфраструктури зовнішнього постачальника, а можливості тонкого налаштування структури курсів і прав доступу є значно скромнішими, ніж у Moodle [16].

Обидві системи підтримують базовий цикл роботи з навчальними завданнями: створення завдання, доведення його до студентів, здача роботи, перевірка, виставлення оцінки й формування підсумків. Вони також можуть надсилати сповіщення електронною поштою або через власні мобільні застосунки. Проте на практиці значна частина оперативної комунікації між студентами й викладачами все одно відбувається через месенджери (зокрема WhatsApp, Viber тощо). Студенти ставлять уточнювальні запитання,

надсилають скріншоти, узгоджують зміни термінів здачі, повідомляють про технічні проблеми саме в чатах, а не у вбудованих інструментах LMS.

З погляду окремої кафедри або викладача, який прагне швидко впорядкувати саме процес здачі й перевірки робіт, повноцінне впровадження Moodle чи Google Classroom може виявитися надмірним. Необхідність підтримувати сервер (у випадку Moodle) або підлаштовуватися під політику хмарного сервісу (у випадку Classroom), обмежені можливості кастомізації під специфічні потоки робіт, а також відсутність безпосередньої інтеграції з каналом повсякденної комунікації залишають незадоволеною потребу в «легкій» системі, що була б максимально наближена до звичного середовища спілкування студентів [17].

Таким чином, аналіз Moodle і Google Classroom показує, що, забезпечуючи широкий спектр функцій для організації навчального процесу, ці системи не вирішують повністю задачу зручної й централізованої роботи саме з процесом здачі, перевірки та обліку навчальних робіт у контексті активного використання месенджерів. Це обґрунтовує доцільність розробки спеціалізованого рішення у вигляді чат-бота в месенджері, який поєднує структурованість LMS із швидкістю й зручністю повсякденної комунікації.

1.3 Теоретичні основи використання чат-ботів у дистанційному навчанні

Свою популярність технологія чат-ботів у сфері освіти здобула завдяки поширенню дистанційного навчання та зростанню потреби в автоматизованій взаємодії між викладачем і студентом.

У багатьох джерелах наголошується на важливості створення ефективних інструментів, що забезпечують швидкий обмін інформацією, зворотний зв’язок та підтримку навчального процесу в умовах дистанційної або змішаної форми навчання [1]. Такі рішення сприяють підвищенню інтерактивності освітнього середовища, оптимізації комунікації між

учасниками навчального процесу та формуванню більш гнучких моделей організації освітньої діяльності.

Тему використання електронних засобів комунікації в освітньому середовищі висвітлено у матеріалах з електронного навчання [2], де підкреслюється, що інформаційне середовище є засобом ефективної взаємодії учасників освітнього процесу. Як приклад сучасного інструменту такої взаємодії в навчанні розглядаються чат-боти – цифрові асистенти, що забезпечують інтерактивний діалог зі здобувачами освіти та підтримують навчальний процес у віртуальному середовищі. Аналогічні підходи описуються в оглядах щодо використання чат-ботів, де їх розглядають як цифрових асистентів на основі ШІ, що працюють у форматі діалогу та можуть використовувати технології обробки природної мови для коректної відповіді на запити користувача. Такі рішення дають змогу швидше орієнтуватися в курсі, отримувати пояснення й підказки “у розмові”, а також підтримують самостійне навчання завдяки постійній доступності та миттєвим відповідям. Окремо відзначається потенціал персоналізації: чат-бот може враховувати різні критерії, пропонуючи релевантні матеріали. Крім навчальної підтримки, чат-боти можуть брати на себе частину організаційних дій — зокрема нагадування про завдання або появу нових матеріалів [3].

Згідно з методичними рекомендаціями Міністерства освіти і науки України [4], ефективна організація дистанційного навчання повинна передбачати інтеграцію автоматизованих сервісів комунікації, у тому числі чат-ботів, що допомагають у розподілі навчальних завдань, проведенні опитувань та зборі аналітики навчальної активності студентів.

Також наявно багато матеріалів, де розглянуто практичні аспекти та саме створення чат-ботів для сповіщення студентів про розклад занять і для автоматизованої перевірки поточної успішності [5].

Таким чином, аналіз літературних джерел свідчить, що чат-боти стають важливим елементом дистанційного освітнього процесу. Вони сприяють підвищенню ефективності комунікації, оперативності управління

навчальними подіями та залученню студентів до активної взаємодії з навчальним середовищем.

1.4 Порівняльний аналіз підходів до впровадження чат-ботів у дистанційному навчанні

Отже, проведений аналіз показує, що чат-бот у месенджері є доцільним рішенням для підтримки дистанційної освіти завдяки поєднанню мобільності, інтерактивності та оперативності комунікації. Інтеграція чат-бота з навчальними системами забезпечує постійний зворотний зв’язок між викладачем і студентом, сприяє автоматизації комунікацій і підвищенню ефективності навчального процесу. При цьому ключові принципи проєктування (рольова модель, стани завдань, журнал подій, сповіщення, обмін файлами) є типовими й можуть бути реалізовані в різних платформах із подібною логікою взаємодії.

Разом із тим, аналіз практичних підходів до впровадження чат-ботів у навчанні демонструє низку суттєвих недоліків, які обмежують ефективність таких рішень у реальному освітньому процесі. По-перше, у багатьох випадках чат-бот використовується переважно як канал сповіщень або довідковий інструмент, тоді як повний цикл роботи з навчальними завданнями (структуроване подання робіт, фіксація версій, контроль статусів, збереження історії перевірок, формування підсумкових відомостей) залишається поза межами системи або реалізований фрагментарно. По-друге, навіть за наявності інтеграції з LMS, відсутність єдиного “центра” даних і чіткої моделі процесів призводить до дублювання операцій: частина інформації зберігається в LMS, частина – у листуванні, частина – у локальних таблицях викладача, що знову породжує фрагментацію та знижує відтворюваність результатів. [6], [9]. По-третє, складність забезпечення коректного доступу та керованості сценаріїв у чат-інтерфейсі (ідентифікація, рольовий поділ функцій, запобігання помилковим діям, контроль повторних подань) часто залишається недостатньо

опрацьованою, що унеможливлює використання чат-бота як надійного інструмента обліку навчальних робіт у масштабі курсу або групи [7], [8].

Отже, попри доведену доцільність використання чат-ботів як каналу навчальної взаємодії, існуючі підходи не забезпечують повною мірою централізації та формалізації процесів здачі, перевірки та оцінювання навчальних робіт у єдиному середовищі. Це формує проблему, що потребує подальшого розв'язання: необхідність розроблення та обґрунтування підходу до впровадження чат-бота в месенджері з підтримкою бот-інтерфейсів, який забезпечує цілісний цикл роботи з навчальними завданнями, зменшує фрагментацію каналів, забезпечує прозорі статуси й історію взаємодії та створює дані, придатні для подальшого моніторингу й формування звітності.

1.5 Обґрунтування припущень щодо доцільності та ефективності впровадження чат-бота

Проведений аналіз існуючих підходів до організації здачі та перевірки навчальних робіт, а також виявлені проблеми фрагментації каналів комунікації та розрізненості інструментів, дають підстави сформулювати припущення щодо доцільності та потенційного ефекту впровадження спеціалізованого чат-бота в месенджері. Враховується поширена практика використання месенджерів як одного з основних каналів повсякденного зв'язку між студентами та викладачами, що робить перенесення частини навчальних операцій у звичне комунікаційне середовище організаційно прийнятним за умови відсутності додаткових бар'єрів для користувачів [19].

Перше припущення полягає в тому, що впровадження чат-бота для організації процесів здачі, перевірки та оцінювання навчальних робіт є доцільним і буде позитивно сприйняте основними користувачами. Очікується, що такий інструмент оцінюватиметься як актуальний і корисний, а використання звичного месенджера знизить поріг входу порівняно з окремими

платформами або набором розрізнених каналів, оскільки користувачам не потрібно освоювати новий інтерфейс і змінювати звичну модель комунікації.

Друге припущення полягає в тому, що застосування чат-бота забезпечить практичний ефект у вигляді підвищення ефективності організації роботи із завданнями. Під цим розуміється скорочення часу на рутинні операції (пошук актуальної інформації, ідентифікація подань, контроль статусів, підготовка відомостей), а також зменшення кількості типових організаційних помилок (дублювання подань, втрата файлів, складність відновлення історії взаємодії, пропуски дедлайнів через недостатню видимість термінів).

2 РОЗРОБКА МЕТОДИКИ ПРОВЕДЕННЯ ДОСЛІДЖЕННЯ ОРГАНІЗАЦІЇ ОСВІТНЬОГО ПРОЦЕСУ

Оскільки ключові проблеми проявляються на стику комунікації та обліку (передача завдань, приймання файлів, фіксація фактів здачі, надання зворотного зв’язку, перенесення результатів у відомості), доцільним є розгляд автоматизації саме цих рутинних операцій, не змінюючи інтелектуальну складову перевірки змісту робіт [7, 8].

Для реалізації автоматизації необхідно обрати придатну форму інформаційної системи, здатну працювати в звичному для користувачів каналі комунікації та забезпечувати керовані сценарії взаємодії для студента і викладача. Як практичний інструмент розглядається чат-бот у месенджері з підтримкою бот-інтерфейсів і програмних API, що дозволяє стандартизувати процеси подання та перевірки робіт, забезпечити однозначну прив’язку дій до користувачів і завдань, а також формувати звітність у табличному форматі. При цьому принципи побудови такого рішення (рольова модель, стани завдань і подань, журнал подій, сповіщення, обмін файлами) є типовими й можуть бути адаптовані до різних платформ, які підтримують чат-боти. У подальших підпунктах конкретизується постановка задачі автоматизації, визначаються критерії оцінювання доцільності та ефективності запропонованого підходу, а також задається основа для подальшої експериментальної перевірки результатів.

2.1 Постановка задачі дослідження

У вихідному стані процес здачі, перевірки та обліку навчальних робіт організовано фрагментарно і по-різному для різних дисциплін. Для студентів це проявляється в тому, що завдання, уточнення та дедлайни надходять з різних джерел (LMS, повідомлення в чатах, електронна пошта, усні оголошення), а подання робіт часто виконується «вручну» через надсилання

файлів у листах або месенджерах без єдиного стандарту. Для викладачів основна проблема полягає у великій кількості розрізнених повідомлень і вкладень, які потрібно знайти, ідентифікувати, зафіксувати факт здачі, надати відповідь студенту, а потім перенести результат у відомість або журнал. У таких умовах зростають витрати часу на адміністративні операції, підвищується ризик плутанини з файлами/версіями та ускладнюється отримання актуальної картини виконання завдань у групі [18].

Автоматизації підлягає організаційна частина взаємодії «студент – викладач», тобто керований документообіг навчальних робіт у цифровому середовищі. У межах задачі необхідно автоматизувати такі типові процеси:

– процес видачі завдань: публікація завдання з описом, матеріалами та дедлайном у єдиному каналі з доступом для відповідних студентів;

– процес подання роботи студентом: приймання файлів та фіксація факту здачі (дата/час, версія/спроба, коментар за потреби) з підтвердженням студенту;

– процес отримання і впорядкування подань для викладача: формування впорядкованої черги робіт за курсом/завданням/групою зі статусами, щоб викладач не «виловлював» файли в чатах або пошті;

– процес оцінювання і зворотного зв’язку: фіксація результату перевірки (оцінка, коментар, рішення «потребує доопрацювання/зараховано») та автоматичне інформування студента про результат;

– процес формування відомості: автоматичне зведення результатів у табличний вигляд і вивантаження відомості у форматі Excel (.xlsx) для подальшого використання в навчальній документації;

– процес контролю виконання: відображення поточного стану виконання завдань (хто здав/не здав, що на перевірці, що потребує доопрацювання) без ручного збирання інформації з різних каналів.

Оскільки оперативна взаємодія між студентами та викладачами на практиці здебільшого відбувається в месенджерах, доцільним є використання чат-бота як єдиного інтерфейсу для автоматизації ключових операцій: доведення

завдань і дедлайнів до студентів, приймання подань і файлів, фіксація факту здачі та версійності, організація перевірки, надання зворотного зв’язку, виставлення оцінок і формування підсумкової відомості (зокрема з можливістю експорту в табличний формат). Такий підхід зменшує частку неструктурованих каналів (особисті чати, листування без чіткої прив’язки до конкретного завдання), забезпечує відтворюваність операцій і створює основу для подальшої перевірки доцільності та ефективності рішення за визначеними критеріями. На наступних етапах ці критерії обґрунтовуються та перевіряються за результатами анкетування студентів і експертного оцінювання викладачів, а у разі підтвердження припущень виконується розробка прототипу програмного засобу як продукту для апробації обраного підходу.

2.2 Функціональні вимоги до системи

Функціональні вимоги визначають повний набір дій, які система повинна підтримувати в чат-інтерфейсі для основних учасників освітнього процесу – студентів і викладачів. У цьому контексті система розглядається як єдиний канал виконання типових операцій із навчальними завданнями, що дозволяє усунути фрагментацію взаємодії між поштою, різними чатами та окремими файлами, а також забезпечити відтворюваність процедур (реєстрація, подання, перевірка, оцінювання, формування результатів). З огляду на предметну область функції групуються за ключовими напрямами, що охоплюють повний життєвий цикл роботи з навчальними матеріалами та результатами:

- реєстрація та ідентифікація користувачів;
- розмежування ролей і прав доступу;
- доступ до курсів і керування підключенням до них;
- керування курсами (для викладача);
- створення та редагування завдань;
- перегляд завдань і матеріалів (для студента);
- подання робіт із додаванням файлів;

– повторне подання та збереження історії версій;

– перевірка робіт і робота з чергою подань (для викладача);

– виставлення оцінки, коментаря та зміна статусу подання;

– сповіщення про ключові події (завдання, дедлайни, результати);

– формування підсумкових результатів і експорт відомості в табличний формат (XLSX).

Базовою передумовою коректної роботи є реєстрація та ідентифікація користувачів із додатковою верифікацією (наприклад, через корпоративну електронну пошту) та подальше розмежування ролей і прав доступу. Це означає, що студент і викладач отримують різні набори можливостей у межах одного інтерфейсу, а система на рівні логіки доступу запобігає виконанню критичних операцій неавторизованими користувачами або користувачами з невідповідною роллю. Після успішної ідентифікації користувач має отримати доступ до курсів: студент – можливість переглядати перелік доступних дисциплін і приєднуватися до них, викладач – можливість створювати й адмініструвати власні курси, змінювати їх параметри та переводити в архівний стан без фізичного видалення даних, що є важливим для збереження історії та подальшої звітності.

Ключова частина функціоналу зосереджена на роботі із завданнями та поданнями. Система повинна забезпечувати створення та редагування завдань із необхідними атрибутами (назва, опис, тип роботи, дедлайн, максимальна оцінка, матеріали), а для студента – перегляд структурованого списку завдань із доступом до прикріплених ресурсів. Подання робіт має відбуватися у стандартизованому сценарії: додавання одного або кількох файлів, можливість додати короткий коментар, підтвердження здачі з автоматичною фіксацією часу та первинного статусу. Оскільки типовою є ситуація доопрацювання, система повинна підтримувати повторне подання із збереженням історії версій і коментарів, щоб і студент, і викладач бачили послідовність змін по конкретному завданню та могли відтворити хід взаємодії без пошуку в чатах або листах.

Для викладача необхідні інструменти перевірки, які мінімізують час на організаційні дії: формування черги подань, перехід до конкретного курсу й

завдання, швидкий доступ до вкладень, відображення метаданих здачі та наявності попередніх спроб. Результат перевірки має фіксуватися через виставлення оцінки, додавання коментаря та встановлення статусу подання (наприклад, «оцінено» або «потребує доопрацювання»), після чого студент автоматично отримує повідомлення з результатом. Окремо система повинна підтримувати сповіщення про події, які реально впливають на поведінку користувачів: поява нового завдання, наближення дедлайну, зміна статусу роботи, публікація оцінки чи коментаря. Такі повідомлення мають зменшувати кількість ручних уточнень і знижувати ризик пропуску термінів, але не бути надмірно нав’язливими.

Завершальним компонентом є формування підсумкових результатів і звітності. Система повинна надавати викладачеві узагальнену картину виконання завдань у межах курсу та підтримувати експорт відомості в табличний формат (XLSX) з уніфікованими полями (ідентифікаційні дані студента, група, курс/завдання, статус, оцінка, час останнього подання, за потреби кількість спроб). Автоматизація цього етапу усуває ручне переписування оцінок і статусів, забезпечує відтворюваність результатів та створює основу для подальшої кількісної оцінки доцільності й ефективності обраного підходу в експериментальній частині роботи.

2.3 Нефункціональні вимоги до системи

Окрім реалізації функцій для студентів і викладачів, система у форматі чат-бота має відповідати нефункціональним вимогам, які визначають якість її експлуатації в реальному навчальному процесі. Ключовими є надійність, безпека, зручність використання та масштабованість, оскільки саме ці характеристики забезпечують стабільну роботу сервісу при щоденному навантаженні, збереження даних без втрат і прогнозовану взаємодію користувачів із навчальними завданнями.

Вимога надійності означає стабільне функціонування чат-бота при одночасній роботі багатьох користувачів та коректну обробку типових подій: реєстрації, керування курсами й завданнями, подання робіт, перевірки, зміни статусів і формування відомостей. Система має гарантувати узгодженість даних: кожне подання повинно бути однозначно прив’язане до користувача й завдання, а зміни статусів – фіксуватися без “розривів” і суперечностей. У разі тимчасових збоїв мережі або недоступності зовнішніх сервісів (наприклад, поштового сервера) система має виконувати повторні спроби з контрольованими інтервалами, повертати зрозумілі повідомлення про помилки та не створювати дублювання операцій (зокрема повторного запису подання або повторної відправки результату). Обов’язковими є резервне копіювання даних і можливість відновлення після аварійних ситуацій, а також ведення журналу подій (логування) для подальшого аналізу інцидентів і відтворення історії дій у спірних випадках.

Вимога безпеки є критичною, оскільки система працює з персональними даними, навчальними матеріалами та результатами оцінювання. Доступ до функцій має бути прив’язаний до облікового запису користувача в месенджері та підтвердженої корпоративної електронної пошти, що знижує ризики підміни особи. Усі операції повинні виконуватися з урахуванням ролі: студент отримує доступ лише до власних подань і результатів, тоді як викладач – до курсів, завдань і робіт тих студентів, для яких він має повноваження. На серверному рівні необхідні валідація вхідних даних, перевірка прав доступу для кожної критичної операції, захист від некоректних запитів, а також безпечне зберігання конфігурації та секретів (токенів, паролів, ключів) без їх фіксації у відкритому вигляді. Додатково слід обмежувати надмірну активність для зменшення ризиків зловживань і спаму та мінімізувати обсяг чутливої інформації в повідомленнях. Робота системи має відповідати внутрішнім вимогам закладу освіти щодо захисту персональних даних.

Вимога зручності використання для чат-бота передбачає логічні й передбачувані діалоги, мінімальну кількість кроків для типових дій та

зрозумілі повідомлення на кожному етапі. Користувач має інтуїтивно розуміти, як виконати реєстрацію, обрати курс, знайти завдання, подати роботу або переглянути результат, без необхідності запам’ятовувати команди. Важливо забезпечити узгоджену термінологію, однакові формулювання статусів, чіткі інструкції (“оберіть курс”, “надішліть файл”, “підтвердіть подання”), а також коректну обробку помилок: система має пояснювати, що сталося, і підказувати наступний крок. Для викладача зручність означає швидкий доступ до черги робіт, можливість перегляду вкладень, фіксації оцінки/коментаря та зміни статусу без зайвих переходів.

Вимога масштабованості пов’язана зі збільшенням кількості курсів, завдань, подань і активних діалогів у процесі експлуатації. Архітектура має допускати розширення без кардинального перепроєктування: оптимізацію структури даних (нормалізація, індекси, коректні зв’язки), асинхронну обробку подій та можливість винесення окремих компонентів у самостійні модулі/сервіси за потреби. Система має зберігати прийнятний час відповіді при пікових навантаженнях (масові дедлайни, одночасні подання, активна перевірка), а також підтримувати зростання обсягу вкладень за рахунок розділення структурованих даних і файлового сховища. Окремо враховується переносимість і розгортання: параметри мають задаватися конфігурацією (зокрема через змінні середовища), щоб систему можна було запускати локально або на сервері/хостингу та адаптувати до різних умов експлуатації.

У сукупності перелічені нефункціональні вимоги задають рамки, у межах яких має бути спроєктоване та реалізоване рішення у форматі чат-бота: воно повинно бути стабільним, захищеним, зручним для щоденної роботи та здатним до розвитку й масштабування без втрати керованості освітніх процесів.

3 ВИЗНАЧЕННЯ РЕКОМЕНДАЦІЙ ДЛЯ СТВОРЕННЯ ЧАТ-БОТУ

3.1 Методика оцінювання доцільності впровадження чат-бота

У межах кваліфікаційної роботи проведено емпіричне дослідження, метою якого є кількісне обґрунтування доцільності впровадження чат-бота в месенджері як інструмента підтримки освітнього процесу. Оскільки значна частина очікуваних ефектів (зручність використання, актуальність, корисність взаємодії зі студентами, довіра до платформи, економія робочого часу) має якісний характер і не може бути повністю описана лише технічними метриками, для оцінювання обрано метод експертних оцінок.

Метод експертних оцінок – це процес формування кількісних суджень компетентними фахівцями щодо об’єкта дослідження за визначеними критеріями з подальшою формалізованою обробкою результатів (обчислення узагальнених показників та оцінювання узгодженості думок). Основний принцип методу полягає в тому, що індивідуальні та частково суб’єктивні оцінки експертів, отримані за єдиною процедурою, у підсумку дають відтворюваний узагальнений результат [21]. Процедуру експертного оцінювання реалізовано послідовно за такими етапами:

– до анкетування залучені викладачі, які безпосередньо працюють зі студентами та мають практичний досвід використання месенджерів/онлайн-інструментів у навчальній взаємодії.

– сформовано перелік критеріїв, які відображають ключові очікування від впровадження;

– проведено анкетування за допомогою Google Forms. Далі зображений загальний вигляд анкети (рис. 3.1) та блок експертного оцінювання критеріїв (рис. 3.2);

– виконано обчислення середніх оцінок, ваг критеріїв, коефіцієнта узгодженості (конкордації) та інтегрального показника ефективності;

– сформовано висновки щодо доцільності створення чат-боту.

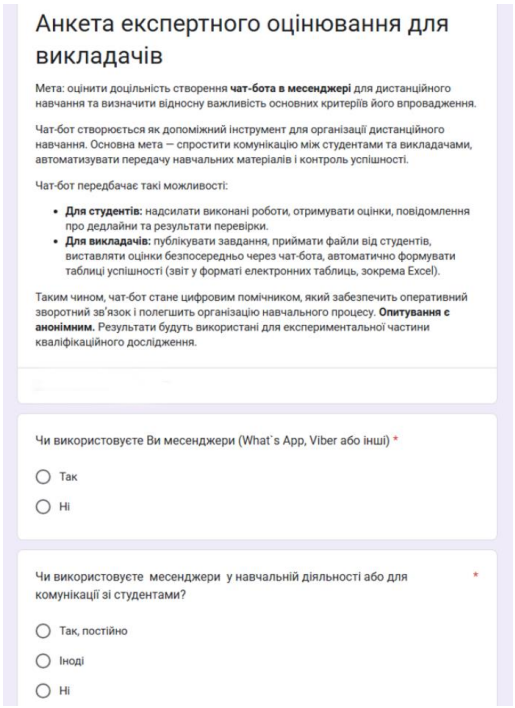


Рисунок 3.1 – Анкета експертного оцінювання

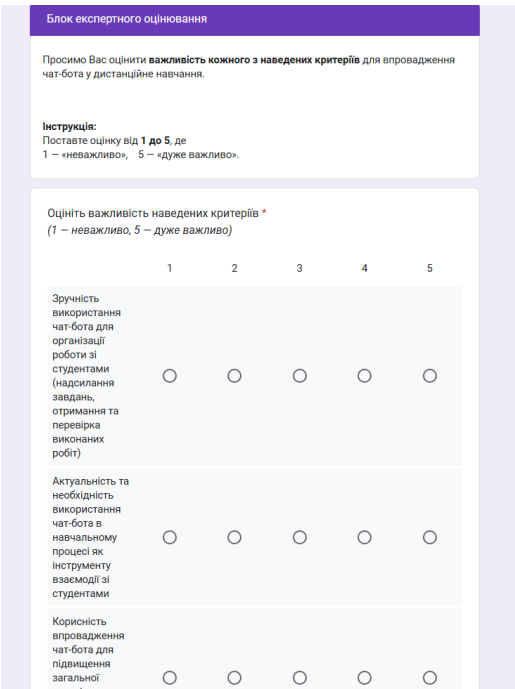


Рисунок 3.2 – Блок експертного оцінювання критеріїв

Окремо від експертного оцінювання викладачів проведено анкетування студентів як основної групи користувачів системи. Метою студентського

опитування є визначення доцільності впровадження чат-бота з позиції студентів, рівня готовності використовувати такий інструмент у навчальній діяльності, а також виявлення очікуваних переваг і потенційних бар'єрів застосування. Анкетування здійснювалося у форматі онлайн-опитування в Google Forms, загальна кількість респондентів становила 62 особи. Результати цього етапу наведено в підрозділі 3.3 у вигляді діаграм та узагальнених показників. Анкета для студентів зображена на рис. 3.3.

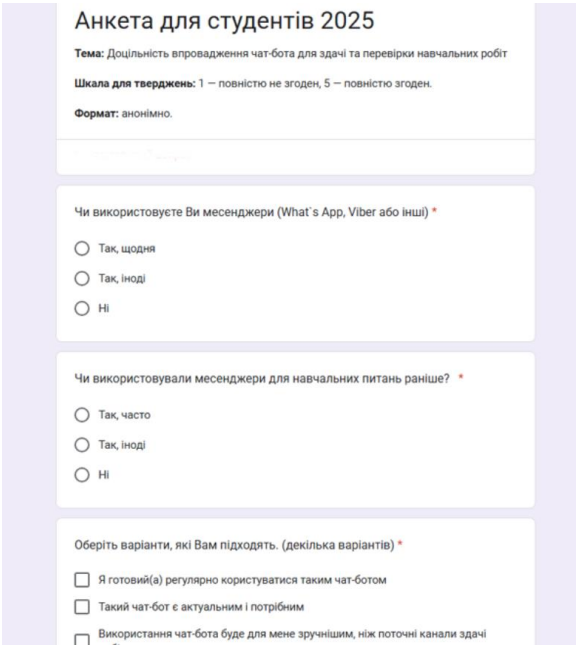


Рисунок 3.3 – Анкета для студентів

Анкета (рис. 3.1) містить вступні запитання, що характеризують досвід використання месенджерів. За результатами встановлено, що всі 6 респондентів використовують один з месенджерів, які підтримують чат-ботів, при цьому 4 застосовують його у навчальній діяльності постійно, а 2 – іноді.

Основна частина анкети (рис. 3.2) передбачала оцінювання впровадження чат-боту за п'ятьма критеріями за п'ятибальною шкалою, де 1 – мінімальна оцінка, а 5 – максимальна.

Критерії були наступними:

– зручність використання чат-бота для організації роботи (надсилання завдань, отримання та перевірка робіт);

– актуальність та необхідність застосування чат-бота в навчальному процесі;

– корисність впровадження для підвищення якості навчальної взаємодії зі студентами;

– надійність платформ, які підтримують чат-ботів та довіра до них;

– ефективність зі сторони економії робочого часу.

Індивідуальні оцінки експертів (рис. 3.4) зведено в табл. 3.1, після чого виконано розрахунки узагальнених показників та коефіцієнта узгодженості думок експертів. На цьому етапі первинні дані було структуровано за критеріями та експертами, визначено строкові суми і середні значення за кожним критерієм, що дозволило перейти від окремих суджень до узагальненого представлення результатів. Далі, з метою врахування відносної важливості кожного критерію, обчислено вагові коефіцієнти, які відображають внесок відповідного критерію у підсумкову оцінку.

	Чи використовуєте Ви месенджер	Чи використовуєте месенджери у навчал	Оцініть важливість наведених критеріїв (1 — неважливо, 5 — дуже важливо) [Зручн	Оцініть важливість наведених критеріїв (1 — неважливо, 5 — дуже важливо) [Актує	Оцініть важливість наведених крит (1 — неважливо, 5 — дуже важливо)
25	Так	Так, постійно	5	5	
25	Так	Так, постійно	5	4	
25	Так	Так, постійно	5	2	
25	Так	Іноді	4	4	
25	Так	Так, постійно	3	3	
25	Так	Іноді	5	5	

Рисунок 3.4 – Результати експертного оцінювання

Окрему увагу приділено перевірці узгодженості експертних суджень, оскільки наявність прийнятного рівня узгодженості є необхідною умовою коректності колективного висновку: за недостатньої узгодженості результати могли б бути випадковими або суперечливими, що знижувало б достовірність висновків. Для цього оцінки було додатково проаналізовано методами ранжування та застосовано коефіцієнт конкордації, який дає змогу кількісно визначити ступінь збігу оцінок між експертами та підтвердити надійність отриманих підсумків.

Таблиця 3.1 – Результати експертного оцінювання критеріїв

№	Критерій	E1	E2	E3	E4	E5	E6	Сума Σ
1	Зручність використання чат-ботів	5	5	5	4	4	4	27
2	Актуальність та необхідність	4	4	3	4	3	4	22
3	Корисність впровадження	4	4	3	5	3	4	23
4	Надійність платформ та довіра до них (What'sApp, Viber або інші месенджери з підтримкою чат-ботів)	5	5	5	4	4	5	28
5	Економія робочого часу викладача	5	4	5	5	4	5	28
Разом								128

Після узагальнення результатів експертного оцінювання (табл. 3.1) можна зробити висновок, що всі визначені критерії отримали високі оцінки: сума балів за кожним із них перебуває в діапазоні 22-28 за максимально можливих 30. Для наочності порівняння критеріїв за середніми значеннями наведено на рис. 3.5.

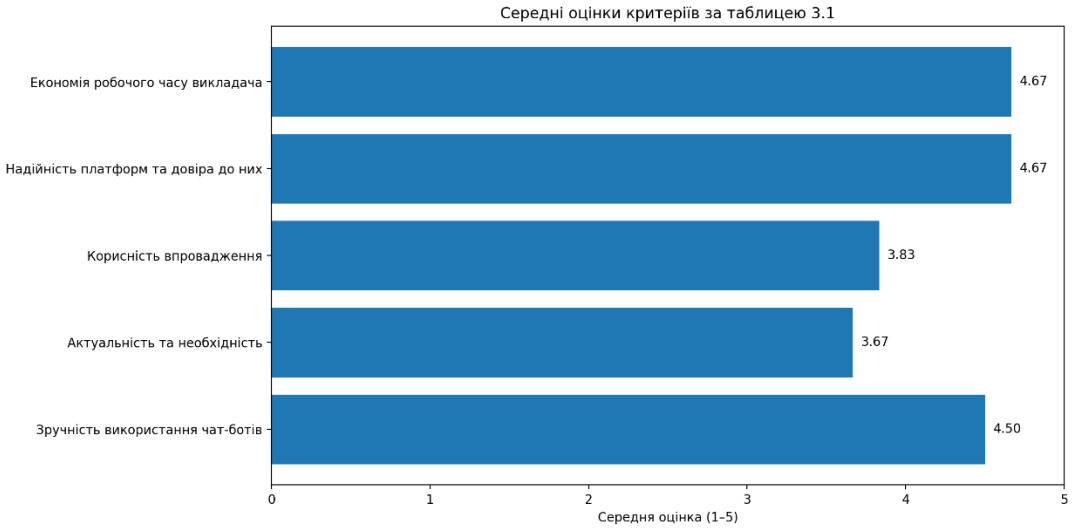


Рисунок 3.5 – Середні оцінки критеріїв за таблицею 3.1

Найвищі підсумкові оцінки зафіксовано за критеріями «Надійність платформ та довіра до них» і «Економія робочого часу викладача» (по 28 балів), що вказує на очікування практичного ефекту від упровадження та достатній рівень довіри до месенджерів як середовища взаємодії. Критерій «Зручність використання чат-бота» також отримав високу підтримку

(27 балів), що підтверджує прийнятність чат-інтерфейсу для виконання типових навчальних операцій. Нижчі, але загалом позитивні оцінки мають критерії «Актуальність та необхідність» (22 бали) і «Корисність упровадження» (23 бали), що може відображати відмінності у практиках організації навчальної роботи на різних дисциплінах і різний рівень потреби в автоматизації. Отримані дані надалі використовуються для обчислення середніх значень, визначення вагових коефіцієнтів критеріїв та перевірки узгодженості суджень експертів у наступних підпунктах.

3.2 Обробка результатів експертного оцінювання викладачів

Після проведення експертного опитування було отримано первинні оцінки. Метою обробки результатів є перехід від індивідуальних суджень експертів до інтегрованих показників, що дозволяють зробити обґрунтовані висновки за сформульованими припущеннями. Для цього послідовно виконуються такі етапи: обчислення середніх значень за кожним критерієм, визначення ваг критеріїв на основі сумарних оцінок, перевірка узгодженості думок експертів за допомогою коефіцієнта конкордації Кендалла та оцінювання статистичної значущості узгодженості.

Середнє значення для кожного критерію:

$$\bar{x}_i = \frac{\sum_{j-1}^{n} x_{ij}}{n}, \tag{3.1}$$

де $\bar{x}_i$ – середня оцінка i-го критерію;

x_{ij} – оцінка i-го критерію, виставлена j-м експертом;

$n -$ кількість експертів.

Загальна середня оцінка:

$$\bar{x}_{\text{заг}} = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}}{n \times m}, \tag{3.2}$$

де $\bar{x}_{заг}$ – загальна середня оцінка;

$m -$ кількість критеріїв.

Підставляючи у (3.1) значення $\bar{x}_i$ з табл. 3.1 та враховуючи кількість експертів $n = 6$, визначено середні оцінки для кожного критерію (округлено до двох знаків): для критерію $\bar{x}_1 = 4{,}50$, критерію $\bar{x}_2 = 3{,}67$, критерію $\bar{x}_3 == 3{,}83$, критерію $\bar{x}_4 = 4{,}67$, критерію $\bar{x}_5 = 4{,}67$. Отримані значення використовуються для подальших розрахунків узагальнених показників.

Далі, підставляючи у (3.2) розраховані середні оцінки $\bar{x}_i$ (або еквівалентно сумарне значення оцінок з табл. 3.1 $\sum x_{ij} = 128$) та враховуючи $m = 5$ критеріїв і $n = 6$ експертів, отримано загальну середню оцінку $\bar{x}_{заг} == 4{,}27$, що відображає узагальнене сприйняття оцінюваного підходу експертною групою. Вагу кожного критерію визначимо як частку суми його оцінок у загальній сумі всіх оцінок:

$$w_i = \frac{\sum_{j=1}^{n} x_{ij}}{\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}}, \tag{3.3}$$

де $w_i -$ вага i-го критерію;

x_{ij} – оцінка i-го критерію, виставлена j-м експертом;

$n -$ кількість експертів.

$m -$ кількість критеріїв.

Оскільки $\sum\sum x_{ij} = 128$, маємо:

$$w_1 = \frac{27}{128} = 0{,}2109,$$
$$w_2 = \frac{22}{128} = 0{,}1719,$$
$$w_3 = \frac{23}{128} = 0{,}1797,$$
$$w_4 = \frac{28}{128} = 0{,}2188,$$
$$w_5 = \frac{28}{128} = 0{,}2188.$$

Для оцінювання узгодженості суджень експертів використано коефіцієнт конкордації Кендалла W. Оскільки оцінювання проводилося за дискретною шкалою, можливі однакові бали за різними критеріями (наявність зв'язок), тому первинні оцінки в межах кожного експерта було перетворено на ранги: найвищому балу відповідає найкращий ранг (ранг 1), а за рівних оцінок критеріям присвоюється середній ранг. Такий перехід є необхідним, оскільки коефіцієнт W характеризує не абсолютні значення балів, а ступінь збігу впорядкування критеріїв між експертами (тобто наскільки подібним є їхнє ранжування). Під час обчислення коефіцієнта враховано поправку на зв'язки, що забезпечує коректність результату та запобігає його спотворенню у випадку однакових оцінок. Ранжовані значення та суми рангів за кожним критерієм систематизовано в табл. 3.2, а їх співвідношення наочно подано на рис. 3.6, що дозволяє візуально порівняти критерії за сумарним рангом перед подальшими розрахунками інтегрального показника оцінювання.

Суми рангів обчислюється та вноситься до таблиці 3.2 за формулою:

$$Ri = \sum_{j=1}^{n} r_{ij}, \tag{3.4}$$

де $R_i -$ сума рангів i-го критерію;

r_{ij} – ранг i-го критерію у j-го експерта;

$n -$ кількість експертів.

Середня сума рангів:

$$R^{-} = \frac{n(m+1)}{2}, \tag{3.5}$$

де $R^{-} -$ середня сума рангів;

m – кількість критеріїв;

$n -$ кількість експертів.

Таблиця 3.2 – Ранги критеріїв та суми рангів

№	Критерій	E1	E2	E3	E4	E5	E6	Σ рангів Ri
1	Зручність використання чат-ботів	2,0	1,5	2,0	4,0	2,0	4,0	15,5
2	Актуальність та необхідність	4,5	4,0	4,5	4,0	4,5	4,0	25,5
3	Корисність впровадження	4,5	4,0	4,5	1,5	4,5	4,0	23,0
4	Надійність платформ та довіра до них (What'sApp, Viber або інші месенджери з підтримкою чат-ботів)	2,0	1,5	2,0	4,0	2,0	1,5	13,0
5	Економія робочого часу викладача	2,0	4,0	2,0	1,5	2,0	1,5	13,0

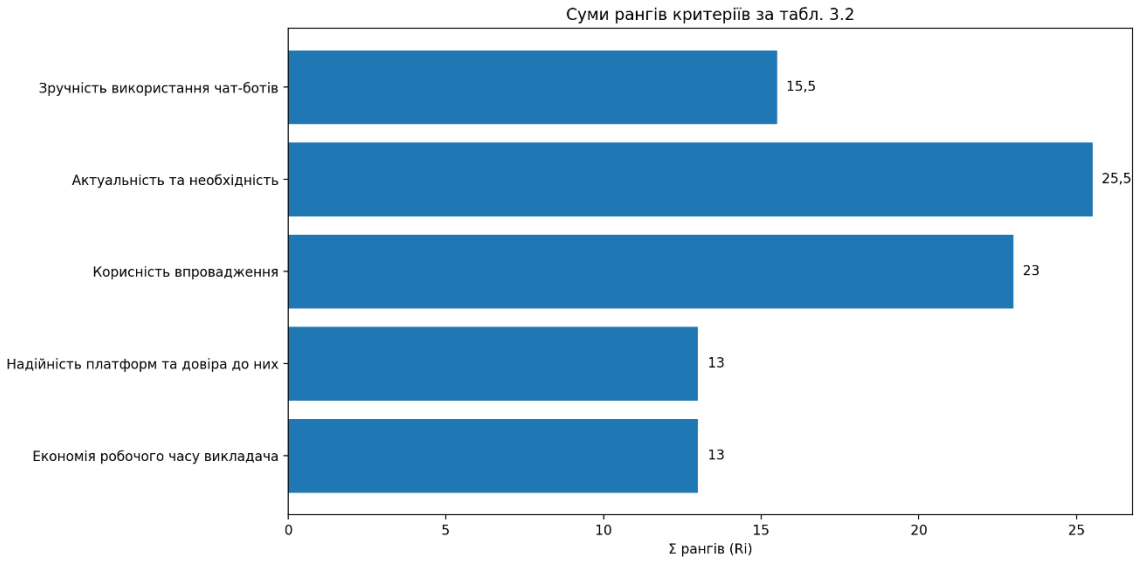


Рисунок 3.6 – Сума рангів за таблицею 3.2

Підставивши у (3.5) кількість експертів та кількість критеріїв, одержуємо середню суму рангів $R^- = 18$.

Для подальшого обчислення коефіцієнта конкордації необхідно оцінити, наскільки суми рангів за кожним критерієм відрізняються від їх середнього значення. З цією метою обчислимо суму квадратів відхилень:

$$S = \sum_{i=1}^{m}(R_i - R^-)^2, \qquad (3.6)$$

де S – сума квадратів відхилень;

R_i – сума рангів i-го критерію;

R^- – середня сума рангів.

Після підстановки до (3.6) значень сум рангів з табл. 3.2 та середньої суми рангів $R^{-} = 18$ обчислюємо квадрати відхилень для кожного критерію і підсумовуємо їх. У результаті отримано $S = 137{,}5$.

Далі використаємо поправку на зв'язані ранги за формулою:

$$T_j = \sum_k (t_k^3 - t_k), \tag{3.7}$$

де $T_j -$ поправка для j-го експерта;

$t_k -$ кількість елементів у k-й групі однакових оцінок.

Після застосування (3.7) встановлено, що в оцінках кожного експерта наявні дві групи зв'язаних рангів: одна група з трьох однакових оцінок ($t = 3$) та одна група з двох однакових оцінок ($t = 2$). Тому поправка для одного експерта обчислюється як сума внесків від обох груп та дорівнює 30. Оскільки в опитуванні брали участь 6 експертів, сумарна поправка за всіма експертами $\sum T = 180$.

Коефіцієнт конкордації визначаємо за формулою:

$$W = \frac{12S}{n^2(m^3 - m) - n\sum T}, \tag{3.8}$$

де $W -$ коефіцієнт конкордації;

$S -$ сума квадратів відхилень;

$n -$ кількість експертів;

$m -$ кількість критеріїв;

$\sum T -$ сумарна поправка на зв'язки.

Після визначення $S, n, m, \sum T$ підставляємо ці значення у формулу (3.8). Спочатку обчислюємо частину знаменника $n^2(m^3 - m) = 4320$. Далі враховуємо поправку на зв'язки $n\sum T = 1080$. Чисельник формули дорівнює $12S = 1650$.

Отже, коефіцієнт конкордації становить:

$$W = \frac{12 \times 137,5}{4320 - 1080} = 0,509.$$

Перевіримо статистичну значущість χ^2 за формулою:

$$\chi^2 = n(m-1)W, \quad (3.9)$$

де $\chi^2 -$ коефіцієнт конкордації;

$n -$ кількість експертів;

$m -$ кількість критеріїв;

$W -$ коефіцієнт конкордації.

Підставляємо у (3.9) значення n, m та W та отримуємо результат $12,22$.

Для $v = m - 1 = 4$ ступені свободи та α = 0,05 отримаємо $\chi^2_{0,05;4}$ = 9,49. Оскільки $12,22 > 9,49$, узгодженість думок експертів є статистично значущою.

Проведемо нормування середніх оцінок за формулою:

$$p_i = \frac{\bar{x}_i}{5}, \quad (3.10)$$

де $p_i -$ нормована оцінка i-го критерію;

$\bar{x}_i -$ середня оцінка i-го критерію;

$5 -$ кількість критеріїв.

$$p_1 = 0,9;\ p_2 = 0,733;\ p_3 = 0,767;\ p_4 = 0,933;\ p_5 = 0,933;$$

За допомогою формули розрахуємо інтегральний показник:

$$Q = \sum_{i=1}^{m} w_i p_i, \quad (3.11)$$

де $p_i -$ нормована оцінка i-го критерію;

$w_i -$ вага i-го критерію;

$Q -$ узагальнений показник.

Використовуючи знайдені вагові коефіцієнти критеріїв та нормовані значення середніх оцінок як зважену суму добутків для всіх критеріїв. Для забезпечення прозорості розрахунку проміжні результати подаємо у табличному вигляді. Отримана сума і є значенням інтегрального показника, що характеризує узагальнену оцінку доцільності/ефективності за обраними критеріями. Саме ці вихідні дані та результати проміжних обчислень занесено до табл. 3.3, яка використовується для подальшого переходу до 5-бальної шкали та формування підсумкового висновку.

Таблиця 3.3 – Розрахунок інтегрального показника

№	w_i	p_i	$w_i \times p_i$
1	0,2109	0,900	0,1898
2	0,1719	0,733	0,1260
3	0,1797	0,767	0,1378
4	0,2188	0,933	0,2042
5	0,2188	0,933	0,2042
		Разом Q:	0,8620

Перейдемо до 5-бальної шкали за формулою:

$$Q_5 = 5Q, \tag{3.12}$$

де $Q_5 -$ узагальнений показник за шкалою 1-5;

$Q -$ узагальнений показник за шкалою 0-1.

Підставивши отримане значення узагальненого показника Q у формулу переходу до 5-бальної шкали, одержуємо $Q_5 = 4{,}31$ бала.

3.3 Обробка результатів анкетування студентів щодо доцільності використання чат-боту

Для перевірки доцільності впровадження чат-бота в месенджері з погляду безпосередніх користувачів проведено анкетування студентів. На відміну від експертного оцінювання, де аналіз виконується на основі професійного досвіду викладачів і узгодженості їхніх суджень, студентське опитування спрямоване на виявлення практичних потреб, очікувань та рівня готовності користуватися запропонованим інструментом у повсякденній навчальній діяльності, оскільки саме студенти виконують і надсилають навчальні роботи.

Анкетування проводилося у дистанційному форматі за допомогою Google Forms. Повні результати студентського анкетування наведено у вигляді таблиці в додатку Б. У основному тексті подано лише узагальнені діаграми та ключові індикатори, необхідні для інтерпретації результатів. Респондентам пропонувалося відповісти на запитання, що характеризують їхній поточний досвід взаємодії з навчальними завданнями, канали комунікації з викладачами, наявні проблеми під час здачі робіт, а також ставлення до перенесення ключових операцій із завданнями у звичне середовище месенджера. Окремий блок питань був спрямований на оцінювання доцільності впровадження чат-бота та визначення функцій, які є найбільш очікуваними з боку студентів.

Зміст анкети сформовано таким чином, щоб отримати дані за кількома групами показників. По-перше, фіксувалися базові умови використання месенджерів як інструмента навчальної взаємодії, зокрема частота користування ними та фактична практика застосування для навчальних цілей. По-друге, оцінювалися типові труднощі, які виникають у студентів під час роботи із завданнями, наприклад пошук актуальної версії вимог, складність швидко знайти власну надіслану роботу, потреба повторно уточнювати статус перевірки, а також випадки, коли дедлайни пропускаються через недостатню видимість термінів. По-третє, визначався рівень зацікавленості в

запропонованому підході, що відображається через оцінку корисності чат-бота, його зручності та готовності студентів використовувати такий інструмент на постійній основі.

Результати опитування надалі інтерпретуються як індикатор доцільності впровадження чат-бота з позиції студентської аудиторії та використовуються для узгодження висновків із результатами експертного оцінювання у п. 3.2. Це дозволяє розглядати доцільність рішення комплексно, враховуючи як думку викладачів, так і очікування студентів, які є основними учасниками процесу здачі навчальних робіт.

За відповідями респондентів встановлено, що месенджери є масово поширеним серед студентів каналом комунікації. Зокрема, 87,1 % респондентів користуються месенджером щодня, а 12,9 % – користуються іноді. Відповідний розподіл наведено на рис. 3.7.

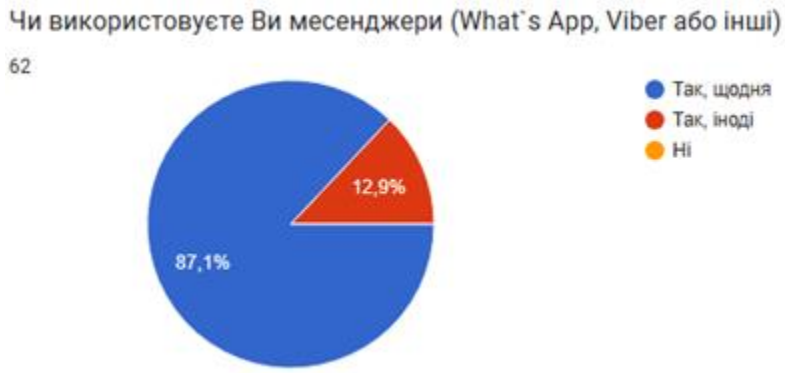


Рисунок 3.7 – Діаграма опитування студентів, перший блок

Досвід використання месенджера саме для навчальних питань також є високим: 74,2 % респондентів вказали, що застосовували месенджер для навчання часто, 22,6 % – іноді, і лише 3,2 % – не використовували його для навчальних питань. Розподіл відповідей наведено на рис. 3.8.

Отримані дані підтверджують наявність сформованої практики використання месенджерів у студентському середовищі, що створює передумови для інтеграції навчальних операцій у цей канал без додаткових витрат на освоєння нового інструмента.

Рисунок 3.8 – Діаграма опитування студентів, другий блок

Далі було проаналізовано ставлення студентів до ідеї впровадження чат-бота. Питання цього блоку передбачало можливість обрати кілька тверджень, тому сума відсотків за варіантами не дорівнює 100 %. Результати наведено на рис. 3.9.

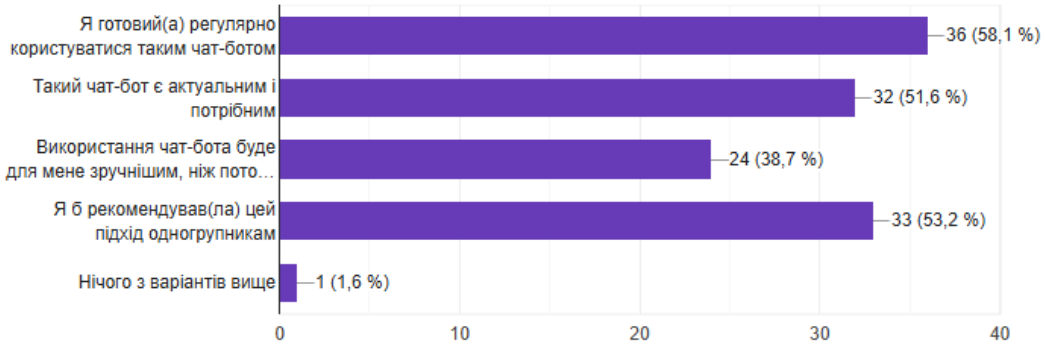


Рисунок 3.9 – Діаграма опитування студентів, третій блок

Найбільшою підтримкою користується твердження про готовність регулярно користуватися чат-ботом: таку позицію обрали 36 студентів, що становить 58,1 % від загальної кількості респондентів. Понад половина опитаних також вважає чат-бот актуальним і потрібним: 32 відповіді, або 51,6 %. Готовність рекомендувати підхід одногрупникам становить 53,2 % (33 відповіді). Водночас лише 38,7 % (24 відповіді) прямо зазначили, що такий чат-бот буде зручнішим за поточні канали, що вказує на наявність частини студентів, які не впевнені у перевазі нового підходу порівняно з чинною практикою.

Окремо проаналізовано фактори, що можуть перешкоджати використанню чат-бота в месенджері. Питання також допускало вибір кількох варіантів, тому сума відсотків перевищує 100 %. Результати наведено на рис. 3.10.

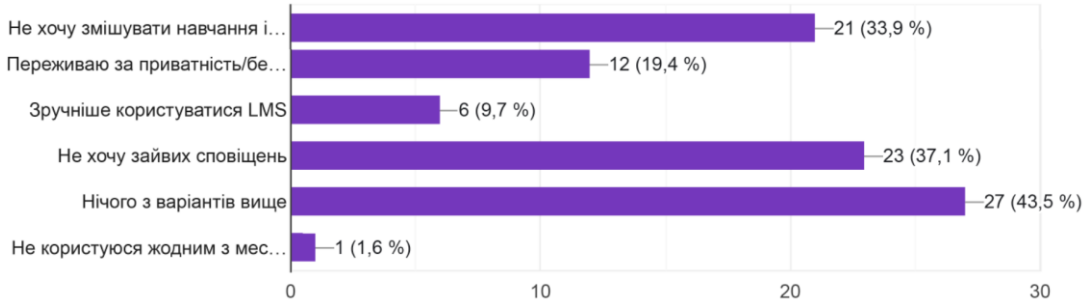


Рисунок 3.10 – Діаграма опитування студентів, четвертий блок

Найбільш поширеними потенційними бар'єрами є небажання отримувати зайві сповіщення – 37,1 % (23 відповіді) та небажання змішувати навчальні й особисті чати – 33,9 % (21 відповідь). Занепокоєння щодо приватності та безпеки зазначили 19,4 % (12 відповідей). При цьому значна частина респондентів – 43,5 % (27 відповідей) – обрала варіант, що жоден із перелічених факторів не є для них суттєвою перешкодою. Отже, для підвищення прийнятності рішення доцільно передбачити керування сповіщеннями, гнучкі налаштування частоти повідомлень, а також мінімізувати нав'язливі повідомлення у повсякденному використанні.

Додатково визначено функції, які студенти вважають найбільш важливими. Результати наведено на рис. 3.11.

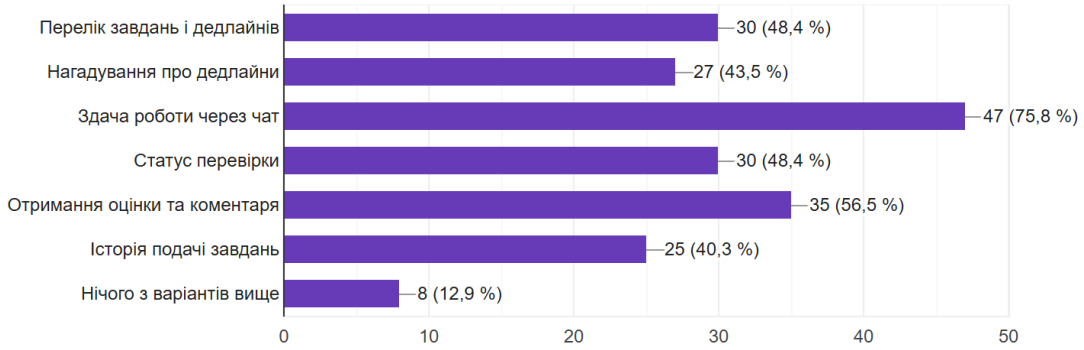


Рисунок 3.11 – Діаграма опитування студентів, п'ятий блок

Найвищий пріоритет має можливість здавати роботу через чат – її обрали 75,8 % (47 відповідей). Високу значущість мають також отримання оцінки та коментаря – 56,5 % (35 відповідей), перегляд переліку завдань і дедлайнів – 48,4 % (30 відповідей), відображення статусу перевірки – 48,4 % (30 відповідей), а також нагадування про дедлайни – 43,5 % (27 відповідей). Ці результати вказують, що студенти очікують від чат-бота насамперед інструментів, які скорочують кількість ручних уточнень і забезпечують швидкий доступ до актуального стану роботи.

Отримані результати демонструють високу поширеність месенджерів у студентському середовищі та наявність сформованого досвіду використання такого каналу для навчальних питань. Понад половина респондентів підтримує доцільність впровадження чат-бота в месенджері, готова користуватися ним регулярно та рекомендувати одногрупникам. Виявлені бар'єри мають переважно організаційний характер і можуть бути зменшені за рахунок налаштувань сповіщень та коректної побудови взаємодії. Таким чином, результати студентського анкетування підтверджують доцільність впровадження чат-бота з позиції студентської аудиторії та є підставою для подальшої інтерпретації разом із результатами експертного оцінювання у п. 3.2.

3.4 Результати ефективності та доцільності чат-боту

Доцільність підходу оцінювалася за двома джерелами даних: експертними оцінками викладачів за визначеними критеріями якості та прийнятності; анкетуванням студентів як основної групи користувачів. За результатами експертного оцінювання (табл. 3.1) найвищі середні значення отримали критерії, що відображають готовність використовувати рішення регулярно: «Надійність платформи» та «Економія робочого часу» (4,67), а також «Зручність використання чат-бота» (4,50).

Інтегральна оцінка, отримана як зважена сума нормованих значень критеріїв, становить $Q = 0{,}862$, що відповідає $Q_5 = 4{,}31$ бала за п'ятибальною

шкалою. Додатково підтверджено, що думки експертів є узгодженими: коефіцієнт конкордації Кендалла $W = 0{,}509$, а перевірка статистичної значущості показала $\chi^2 = 12{,}22 > \chi^2_{0{,}05;4} = 9{,}49$, тобто узгодженість є статистично значущою і колективний висновок не є випадковим.

Результати студентського анкетування (підрозділ 3.3) узгоджуються з експертними висновками та підтверджують прийнятність рішення з позиції користувачів. Встановлено, що месенджери є звичним каналом комунікації для більшості респондентів (87,1 % користуються щодня, 12,9 % – іноді), а також мають сформовану практику застосування для навчальних питань (74,2 % – часто, 22,6 % – іноді). Щодо безпосередньої готовності використовувати чат-бот: 58,1 % респондентів зазначили готовність регулярно користуватися таким інструментом, 51,6 % вважають його актуальним і потрібним, 53,2 % готові рекомендувати підхід одногрупникам. Таким чином, дані опитування студентів підтверджують, що запропонований формат взаємодії сприймається як прийнятний та корисний для повсякденних навчальних операцій.

Доцільність впровадження чат-бота на цьому етапі підтверджено результатами експертного оцінювання та даними опитування студентів, що свідчить про прийнятність запропонованого підходу для основних груп користувачів. Відповідно, наступним кроком є реалізація практичного макета, призначеного не як кінцевий продукт, а як інструмент для перевірки ефективності підходу в умовах, наближених до реального навчального процесу. Після розроблення та апробації макета планується оцінити ефективність за кількісними показниками, зокрема шляхом порівняння базового та нового варіантів організації процесу в четвертому розділі, а також виконати розрахунки економічної доцільності та потенційного ефекту масштабування, що детально розглядається в п’ятому розділі.

4 СТВОРЕННЯ ЧАТ-БОТУ ДЛЯ ПОКРАЩЕННЯ ОСВІТНЬОГО ПРОЦЕСУ

4.1 Постановка задачі програмної реалізації та вимоги до системи

У межах кваліфікаційної роботи реалізовано програмний засіб у вигляді чат-бота NureStudy Bot, призначеного для покращення освітнього процесу шляхом централізації операцій із навчальними роботами в єдиному інтерфейсі. Основною метою програмної реалізації є зменшення адміністративних витрат часу під час організації здачі та перевірки робіт, підвищення прозорості взаємодії «студент–викладач» і зниження кількості типових помилок, що виникають через використання розрізнених каналів комунікації (пошта, різні месенджери, файлообмінники тощо).

Важливою особливістю запропонованого підходу є платформна незалежність: система проєктується так, щоб її логіка не була жорстко прив’язана до конкретного месенджера. Чат-бот у цьому випадку розглядається як клієнтський канал взаємодії, тоді як основні процеси (керування курсами й завданнями, обробка подань, збереження даних, формування сповіщень та звітності) реалізуються на серверній стороні. Завдяки такому розділенню чат-бот може бути адаптований до будь-якої платформи, що підтримує бот-функціональність і роботу з повідомленнями та файлами, без зміни базових алгоритмів і структури даних (переважно потребує заміни або налаштування лише інтерфейсного модуля взаємодії) [23].

Об’єктом автоматизації є процеси видачі завдань, подання студентських робіт, їх перевірки, надання зворотного зв’язку та фіксації результатів. Предметом програмної реалізації виступають алгоритми та програмні механізми, що забезпечують керування цими процесами в асинхронному діалоговому середовищі месенджера з підтримкою чат-ботів, із використанням бази даних та організованого зберігання файлів. Така

структура забезпечує можливість подальшого перенесення рішення на інші платформи зі схожою логікою комунікації та сценаріями взаємодії.

Вимоги для студента:

– реєстрація в системі із підтвердженням (верифікацією) облікового запису через електронну пошту;

– перегляд доступних курсів і активних завдань із описом та дедлайнами;

– подання (здача) роботи з можливістю прикріплення файлів у межах встановлених обмежень, додавання текстового коментаря;

– повторне подання роботи (перездача) із фіксацією версійності та збереженням історії;

– перегляд переліку власних подань і їхніх статусів у режимі, наближеному до реального часу;

– отримання сповіщень про зміну статусу роботи, виставлення оцінки та коментарі викладача.

Вимоги для викладача:

– створення та керування курсами (опис, налаштування доступності);

– створення та керування завданнями (опис, вкладені файли, дедлайни, активність завдання);

– доступ до черги подань для перевірки та перегляд матеріалів студента;

– виставлення оцінки, додавання коментаря, повернення роботи на доопрацювання зі зміною статусу;

– формування відомостей/звітності за результатами оцінювання, у тому числі експорт у табличний формат (Excel) для подальшого використання.

Загальносистемні вимоги:

– підтримка чітко визначених статусів подань (подано/на

– перевірці/потрібно переробити/оцінено тощо) та їх збереження в базі даних;

– фіксація ключових подій процесу (дата/час подання, версія, результат перевірки, коментар) для відтворюваності та прозорості;

– забезпечення зручної навігації через інтерфейс із використанням кнопок та меню.

Нефункціональні вимоги до розробленої системи визначають загальні характеристики якості програмного засобу та умови його коректного функціонування в реальному середовищі експлуатації. Передусім система має забезпечувати надійність і коректність роботи, тобто стабільну обробку діалогових сценаріїв у середовищі месенджера з підтримкою чат-ботів та збереження даних без втрат, зокрема під час завантаження файлів і зміни статусів подань. Важливим є також критерій зручності використання: інтерфейс взаємодії повинен бути інтуїтивно зрозумілим і не вимагати додаткового навчання користувачів, а виконання типових операцій має здійснюватися з мінімальною кількістю кроків.

Окрему увагу приділено безпеці та контролю доступу, що передбачає обов’язкове розмежування прав між ролями студента та викладача, наявність механізмів верифікації користувачів, недопущення виконання викладацьких операцій студентами, а також захист від зловживань шляхом обмеження надмірної активності або спаму на рівні програмної логіки. З погляду продуктивності система повинна забезпечувати прийнятний час відповіді на основні операції, не створюючи помітних затримок у взаємодії, при цьому обмін файлами організовується з урахуванням технічних можливостей обраної платформи месенджера.

Не менш суттєвими є вимоги до супроводжуваності та розширюваності: структура програмного коду має бути модульною, щоб додавання нових сценаріїв, команд, форм звітності або статусів не потребувало повного перепроєктування системи [24]. Важливою нефункціональною характеристикою є також переносимість: прикладна логіка системи проєктується незалежно від конкретної платформи, а адаптація до іншого месенджера зводиться переважно до заміни модуля інтеграції (обробки повідомлень, файлів і сповіщень) без зміни моделі даних та основних алгоритмів. Нарешті, розгортання передбачає можливість запуску

програмного засобу як локально, так і на хостингу або VPS із винесенням налаштувань у конфігурацію через змінні середовища, що спрощує введення в експлуатацію та подальший супровід.

Зазначені нефункціональні вимоги визначили подальші рішення щодо архітектури та реалізації системи. У п. 4.2 розглядається архітектура та обґрунтування обраного технологічного стеку, у 4.3 – модель даних і організація зберігання інформації, у 4.4-4.5 – реалізація ключових підсистем і процесів, а в 4.6 – тестування.

4.2 Архітектура програмного рішення та обґрунтування технологічного стеку

Архітектура розробленого програмного засобу побудована відповідно до подієво-орієнтованої моделі взаємодії, характерної для чат-ботів у месенджерах із підтримкою бот-функціональності. У такій моделі кожне повідомлення користувача, команда або натискання інтерактивної кнопки розглядається як подія, що надходить через програмний інтерфейс платформи та маршрутизується до відповідного обробника [25]. Загальну структурну схему рішення та основні компоненти наведено на рис. 4.1.

У межах цієї схеми ядро чат-бота забезпечує приймання подій, диспетчеризацію обробки, підтримку багатокрокових діалогів, а також формування відповідей користувачеві у вигляді текстових повідомлень та інтерактивних елементів керування. Такий підхід є універсальним і не прив’язує систему до конкретної платформи: за потреби перенесення на інший месенджер змінюється переважно модуль інтеграції з API, тоді як внутрішня логіка обробників, модель даних і сценарії взаємодії залишаються незмінними.

Рисунок 4.1 – Архітектура системи та потік даних

Ключовим принципом архітектури є модульність: функціональні підсистеми виділено у самостійні модулі-обробники, кожен з яких відповідає за окремий процес. Зокрема, реалізацію логіки поділено за напрямами: реєстрація та верифікація користувачів; керування курсами; керування завданнями; подання студентських робіт; перевірка подань викладачем; облік оцінок та формування звітності [26]. Такий поділ зменшує зв’язаність компонентів, підвищує супроводжуваність коду та забезпечує можливість розвитку системи без необхідності повного перепроєктування. Структуру проєкту та розподіл модулів наведено на (рис. 4.2).

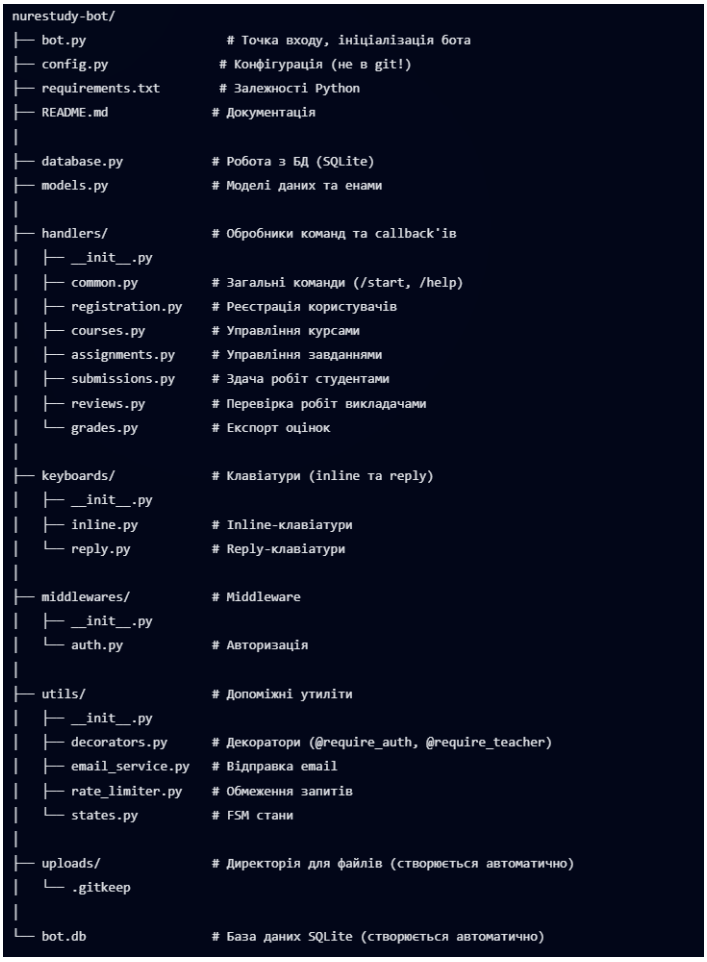


Рисунок 4.2 – Структура проєкту

Для коректної реалізації багатокрокових сценаріїв взаємодії, що складаються з послідовних етапів, у системі застосовано керування діалогами на основі скінченного автомата станів (FSM). Такий підхід використовується для процесів реєстрації та підтвердження користувача, створення курсів і завдань, а також подання студентських робіт із додаванням файлів і коментарів. FSM дозволяє однозначно визначати поточний етап взаємодії, контролювати допустимі дії та запобігати помилкам, які виникають у разі некоректної послідовності введення даних. У результаті діалоги стають відтворюваними, а поведінка системи – прогнозованою як для студента, так і для викладача.

Інтерфейс взаємодії в месенджері реалізовано з використанням кнопкових елементів керування, що підтримуються платформою чат-бота (меню швидкого доступу та контекстні інтерактивні кнопки). Кнопки меню застосовуються для відображення головного меню та швидкого доступу до

типових операцій залежно від ролі користувача, тоді як контекстні інтерактивні кнопки забезпечують навігацію в межах конкретної дії: вибір курсу, вибір завдання, перегляд подань, а також виконання операцій перевірки та оцінювання. Така організація інтерфейсу зменшує потребу у ручному введенні команд, знижує кількість помилок введення та прискорює виконання типових процедур.

Шар зберігання та обробки даних реалізовано на основі реляційної бази даних SQLite. Вибір SQLite обґрунтовується простотою розгортання і експлуатації, оскільки база даних функціонує як локальний файл, не потребує окремого серверного компонента та забезпечує достатню надійність і швидкодію для задач навчального процесу [27]. У БД зберігаються профілі користувачів, курси, завдання, подання студентів, результати перевірки та оцінювання, а також допоміжні метадані. Доступ до даних інкапсульовано у модулі бази даних, що забезпечує єдиний підхід до виконання запитів і підтримує цілісність інформації на рівні прикладної логіки. Деталізацію схеми даних доцільно навести окремо в підрозділі 4.3.

Окрім даних у БД, система передбачає організацію файлового сховища вкладень – файлів робіт і додаткових матеріалів. Реальні файли зберігаються у файловій системі, а у базі даних фіксуються їх метадані та зв’язки з відповідними поданнями. Такий підхід спрощує резервне копіювання, контроль обсягу даних і переносимість рішення при розгортанні в іншому середовищі.

Питання безпеки та коректності доступу реалізовано через рольову модель «студент/викладач» і механізми контролю доступу на рівні обробників. Додатково впроваджено верифікацію користувачів через підтвердження електронної пошти одноразовим кодом, що підвищує достовірність реєстрації та зменшує ризик створення некоректних облікових записів. Для зниження ризиків зловживань та надмірного навантаження застосовано обмеження частоти виконання дій (rate limiting), що є важливим для чат-інтерфейсів із потенційно великою кількістю повторних натискань кнопок або надсилання повідомлень.

Окремо слід відзначити організацію конфігурації застосунку: ключові параметри винесено у конфігураційний модуль config.py. Це забезпечує відокремлення налаштувань від програмної логіки та спрощує розгортання у різних середовищах без модифікації коду. До таких параметрів належать ключ доступу (токен) чат-бота, шлях до бази даних, директорія файлового сховища, SMTP-налаштування для надсилання коду підтвердження, а також службові обмеження, пов’язані з безпекою і захистом від зловживань. Оскільки інтеграційний шар винесений окремо, при перенесенні рішення на іншу платформу змінюються переважно параметри підключення та модуль взаємодії з API, тоді як модель даних і основні сценарії залишаються незмінними. Приклад структури конфігураційних параметрів наведено на рис. 4.3.

```
# ========================================
# ОСНОВНІ НАЛАШТУВАННЯ
# ========================================

# Telegram Bot Token
BOT_TOKEN = "1234567890:ABCdefGHIjklMNOpqrsTUVwxyz"

# ========================================
# БАЗА ДАНИХ
# ========================================

DATABASE_PATH = "bot.db"

# ========================================
# ФАЙЛОВЕ СХОВИЩЕ
# ========================================

UPLOAD_DIR = "./uploads"

# ========================================
# EMAIL СЕРВІС
# ========================================

# SMTP сервер
SMTP_SERVER = "smtp.gmail.com"
SMTP_PORT = 587

# Облікові дані
SMTP_USER = "your_email@gmail.com"
SMTP_PASSWORD = "your_app_password"  # Пароль додатка Google
FROM_EMAIL = "your_email@gmail.com"

# ========================================
# АВТЕНТИФІКАЦІЯ
# ========================================

# Секретний код для реєстрації викладачів
TEACHER_SECRET_CODE = "TEACHER2024"

# Дозволені домени email для студентів
ALLOWED_DOMAINS = ["nure.ua", "student.nure.ua"]

# ========================================
# RATE LIMITING (опціонально)
# ========================================

# Максимальна кількість повідомлень за хвилину
MAX_MESSAGES_PER_MINUTE = 20

# Максимальна кількість здач за годину
MAX_SUBMISSIONS_PER_HOUR = 10

# ========================================
# ІНШЕ
# ========================================

# Часовий пояс
TIMEZONE = "Europe/Kiev"

# Тайм-аут для верифікації email (хвилин)
EMAIL_VERIFICATION_TIMEOUT = 10
```

Рисунок 4.3 – Структура config.py

Підсумовуючи, розроблена система має подієво-орієнтовану модульну архітектуру, у якій логіка взаємодії з користувачем реалізована через обробники подій та механізм керування станами діалогів (FSM), а інтерфейс – через кнопкові елементи керування, що підтримуються платформою чат-бота (меню швидкого доступу та контекстні інтерактивні кнопки). Такий підхід не прив’язує реалізацію до конкретного месенджера і дозволяє адаптувати інтерфейсний шар до інших платформ із подієвою моделлю взаємодії без зміни основних сценаріїв.

Зберігання структурованих даних організовано на базі SQLite, тоді як вкладення робіт зберігаються у файловому сховищі з фіксацією метаданих у базі даних, що забезпечує прозорість та відтворюваність процесів. Винесення параметрів у конфігураційний модуль config.py підвищує переносимість і спрощує розгортання системи, а застосування рольового контролю доступу, верифікації користувачів та обмеження частоти дій забезпечує коректність і безпечність експлуатації. Далі в п. 4.3 наведено модель даних і організацію зберігання інформації, що деталізує структуру сутностей та зв’язків, на яких ґрунтуються ключові процеси системи.

4.3 Проєктування моделі даних та організація зберігання інформації

Інформаційна система чат-бота в месенджері потребує надійного збереження даних, які відображають навчальні процеси: реєстрацію користувачів, структуру курсів і завдань, факти подання робіт, результати перевірки та історію змін. З огляду на необхідність забезпечення цілісності даних, однозначного зв’язування подій із конкретними користувачами та можливості формування звітності, у роботі застосовано реляційну модель даних на основі SQLite. Структуру таблиць бази даних і призначення їх полів наведено в табл. 4.1-4.7.

Такий підхід дозволяє формалізувати основні сутності предметної області й забезпечити відтворюваність процесів, а також зберігає

переносимість рішення: модель даних і логіка обробки подій не залежать від конкретної платформи, тому під час інтеграції з іншим месенджером змінюється переважно інтерфейсний (інтеграційний) модуль, тоді як структура збереження та основні процеси залишаються незмінними.

Таблиця 4.1 – Users

Поле	Тип	Опис
Id	INTEGER PK	Унікальний ID
messenger_id	INTEGER UNIQUE	ID користувача
email	TEXT UNIQUE	Email користувача
username	TEXT	username
full_name	TEXT	Повне ім'я
group_name	TEXT	Група (для студентів)
role	TEXT	Роль
is_verified	BOOLEAN	Верифікація
created_at	TIMESTAMP	Дата реєстрації

Таблиця 4.2 – Courses

Поле	Тип	Опис
id	INTEGER PK	Унікальний ID
teacher_id	INTEGER FK	ID викладача
name	TEXT	Назва курсу
description	TEXT	Опис
is_active	BOOLEAN	Активність
created_at	TIMESTAMP	Дата створення

Таблиця 4.3 – Assignments

Поле	Тип	Опис
id	INTEGER PK	Унікальний ID
course_id	INTEGER FK	ID курсу
title	TEXT	Назва завдання
description	TEXT	Опис
deadline	TIMESTAMP	Дедлайн
is_active	BOOLEAN	Активність
created_at	TIMESTAMP	Дата створення

Таблиця 4.4 – Submissions

Поле	Тип	Опис
id	INTEGER PK	Унікальний ID
student_id	INTEGER FK	ID студента
assignment_id	INTEGER FK	ID завдання
version	INTEGER	Версія здачі
status	TEXT	Статус
comment	TEXT	Коментар
grade	TEXT	Оцінка
created_at	TIMESTAMP	Дата здачі
updated_at	TIMESTAMP	Дата оновлення

Таблиця 4.5 – Submissions_files

Поле	Тип	Опис
id	INTEGER PK	Унікальний ID
submission_id	INTEGER FK	ID здачі
file_name	TEXT	Ім'я файлу
file_path	TEXT	Шлях до файлу
messenger_file_id	TEXT	File ID
uploaded_at	TIMESTAMP	Дата завантаження

Таблиця 4.6 – Assignment_files

Поле	Тип	Опис
id	INTEGER PK	Унікальний ID
assignment_id	INTEGER FK	ID завдання (посилання на assignments)
file_name	TEXT	Ім'я файлу матеріалів до завдання
file_path	TEXT	Шлях до файлу у файловому сховищі
messenger_file_id	TEXT	File ID (для повторного завантаження)
uploaded_at	TIMESTAMP	Дата завантаження

Таблиця 4.7 – Reviews

Поле	Тип	Опис
id	INTEGER PK	Унікальний ID
submission_id	INTEGER FK	ID здачі (посилання на submissions)
teacher_id	INTEGER FK	ID викладача (посилання на users)
grade	TEXT	Оцінка за результатами перевірки
comment	TEXT	Коментар викладача до роботи
created_at	TIMESTAMP	Дата/час формування перевірки

Центральною сутністю є користувач системи, який описується таблицею users (табл. 4.1). У ній фіксуються ідентифікатор користувача в базі даних, унікальний ідентифікатор користувача в обраній платформі месенджера, електронна пошта та допоміжні атрибути профілю (ім’я, нікнейм, академічна група). Крім того, зберігається роль користувача, що визначає доступ до функцій системи та приймає значення «студент» або «викладач», а також ознака верифікації облікового запису для підвищення достовірності реєстрації.

Навчальна структура організована через таблицю courses (табл. 4.2), де кожний курс прив’язаний до викладача за зовнішнім ключем. У таблиці зберігаються назва, опис, ознака активності та час створення курсу. Використання ознаки активності дозволяє зберігати історичні дані без фізичного видалення записів, що є важливим для відтворюваності освітнього процесу та формування звітів.

Таблиця assignments (табл. 4.3) містить завдання, які належать конкретному курсу. Для кожного завдання зберігаються назва, опис, дедлайн, ознака активності та час створення. Зв’язок за ключем курсу забезпечує реалізацію ієрархії «курс – завдання», що використовується під час формування списків завдань у чат-інтерфейсі та під час контролю актуальності дедлайнів.

Факт здачі студентської роботи фіксується таблицею submissions (табл. 4.4), де кожний запис відповідає конкретному студенту та конкретному завданню. Для підтримки повторних здач передбачено поле версійності, яке відображає номер спроби, а також часові поля створення і оновлення, що дозволяє відстежувати динаміку подань. Стан подання зберігається у вигляді статусу з наперед визначеного набору значень, які відображають життєвий цикл роботи: «очікує перевірки», «на перевірці», «потребує виправлення», «оцінено». Додатково у таблиці зберігаються коментар студента (за потреби) та підсумкова оцінка, що забезпечує як відображення результатів у чаті, так і подальше формування відомостей.

Оскільки у навчальному процесі критично важливими є вкладення (файли звітів, архіви, зображення тощо), для їх обліку застосовано окремі

таблиці. Таблиця submission_files (табл. 4.5) зберігає перелік файлів, доданих до конкретного подання, включно з назвою, шляхом у файловому сховищі, ідентифікатором файлу на платформі месенджера (якщо він використовується для повторного доступу/завантаження) та часом завантаження. Аналогічно, таблиця assignment_files (табл. 4.6) описує файли, які прикріплюються до завдання як вихідні матеріали або додаткові ресурси. Винесення вкладень у окремі таблиці зв'язку дозволяє зберігати реальні файли у файловій системі, а в базі даних фіксувати їх метадані та прив'язку до навчальних сутностей, що є більш раціональним за зберігання великих бінарних об'єктів у самій БД.

Для фіксації результатів перевірки реалізовано таблицю reviews (табл. 4.7), яка містить записи про дії викладача щодо конкретного подання. У ній зберігається прив'язка до подання, прив'язка до викладача, оцінка та/або коментар, а також час створення запису. Наявність такої сутності забезпечує аудит процесу оцінювання та дозволяє зберігати історію зворотного зв'язку, що підвищує прозорість і відтворюваність перевірки робіт.

Підсумовуючи, застосована реляційна модель даних забезпечує однозначне зв'язування користувачів, курсів, завдань, подань і результатів перевірки. Поєднання SQLite для структурованих даних та файлового сховища для вкладень дозволяє одночасно підтримувати цілісність інформації й ефективно працювати з навчальними матеріалами [28]. Запропонована структура підтримує рольовий доступ, версійність здач і збереження історії перевірок, що відповідає вимогам прозорості та керованості процесів у межах системи.

4.4 Реалізація підсистеми доступу та керування користувачами

Підсистема доступу та керування користувачами забезпечує коректну ідентифікацію учасників освітнього процесу, верифікацію облікового запису та розмежування прав доступу відповідно до ролей «студент» і «викладач». Оскільки система функціонує у форматі чат-інтерфейсу месенджера (зокрема, WhatsApp) і взаємодія відбувається короткими діями та повідомленнями,

ключовими вимогами є керованість сценаріїв, відтворюваність процесу реєстрації та запобігання несанкціонованому виконанню критичних операцій. При цьому логіка доступу, верифікації та рольового розмежування є типовою для більшості платформ із підтримкою чат-ботів, що забезпечує можливість адаптації рішення до інших месенджерів без зміни базових принципів роботи.

Реєстрація користувача в системі ініціюється командою або натисканням кнопки «Реєстрація» та реалізована як послідовний діалог із фіксацією проміжних станів (FSM) [29]. На першому етапі користувач обирає роль у системі – «студент» або «викладач» (рис. 4.4). Вибір ролі визначає подальший маршрут реєстрації та набір доступних функцій після завершення процедури, що дозволяє одразу відокремити сценарії навчальної взаємодії від функцій керування курсами та перевірки робіт.

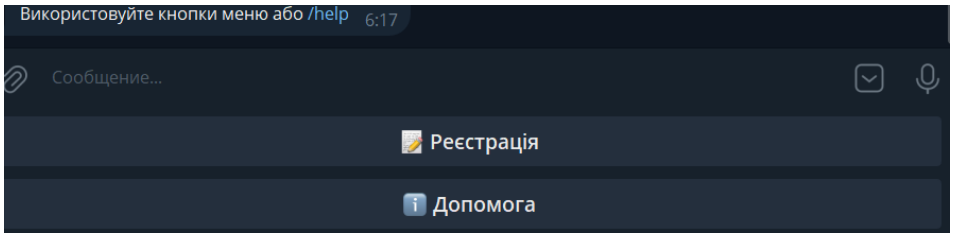


Рисунок 4.4 – Вибір ролі під час реєстрації

Наступним кроком є введення електронної адреси, яка використовується як додатковий ідентифікатор і канал верифікації. Після введення адреси система ініціює надсилання одноразового коду підтвердження на пошту, що підтверджує належність введеної адреси користувачеві та зменшує ризик реєстрації з помилковими або сторонніми даними. Дозволені домени електронної пошти можуть бути налаштовані; за замовчуванням з міркувань безпеки використовується домен nure.ua. Приклад листа з кодом підтвердження наведено на рис. 4.5.

Отриманий код користувач вводить у чаті бота, після чого система виконує перевірку правильності коду та приймає рішення щодо продовження реєстрації. Для підвищення безпеки та зменшення ризиків зловживань

застосовується обмеження частоти повторного надсилання коду, що запобігає масовим запитам і надмірному навантаженню на поштовий сервіс.

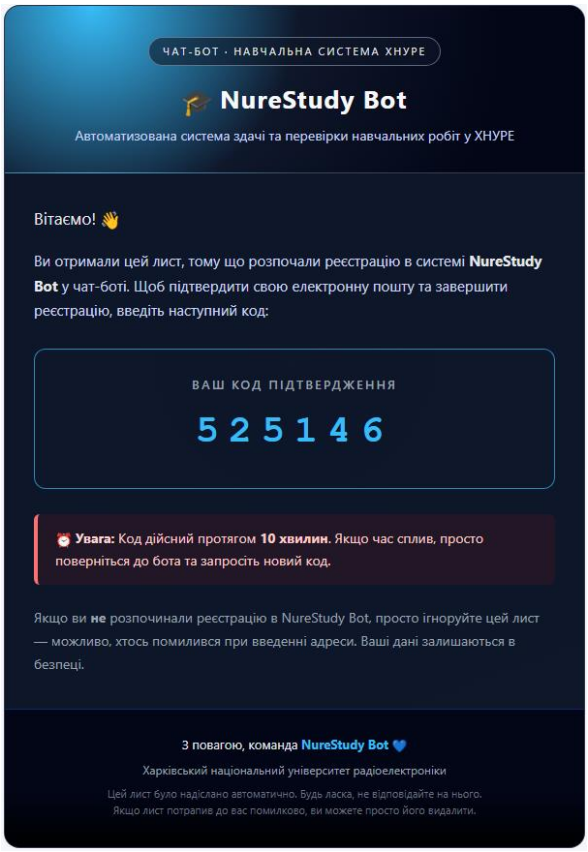


Рисунок 4.5 – Лист із кодом підтвердження на електронній пошті

Після успішної верифікації сценарій реєстрації продовжується відповідно до обраної ролі. Для ролі «студент» система запитує додаткові довідкові дані (зокрема академічну групу), після чого створює профіль і переводить користувача до основного функціоналу. Після завершення реєстрації студенту відображається головне меню з типовими діями (рис. 4.6), що забезпечує швидкий доступ до перегляду курсів, завдань та власних подань без необхідності ручного введення команд.

Для ролі «викладач» передбачено додатковий етап підтвердження права доступу до функцій керування навчальним процесом. Після верифікації електронної адреси користувач вводить службовий секретний код викладача, а також ідентифікаційні дані (ПІБ), які використовуються в системі для відображення та формування звітності. Після завершення реєстрації викладачу

відображається головне меню викладача (рис. 4.7), через яке реалізуються сценарії створення курсів, керування завданнями, перегляду черги подань на перевірку та формування/експортування результатів оцінювання.

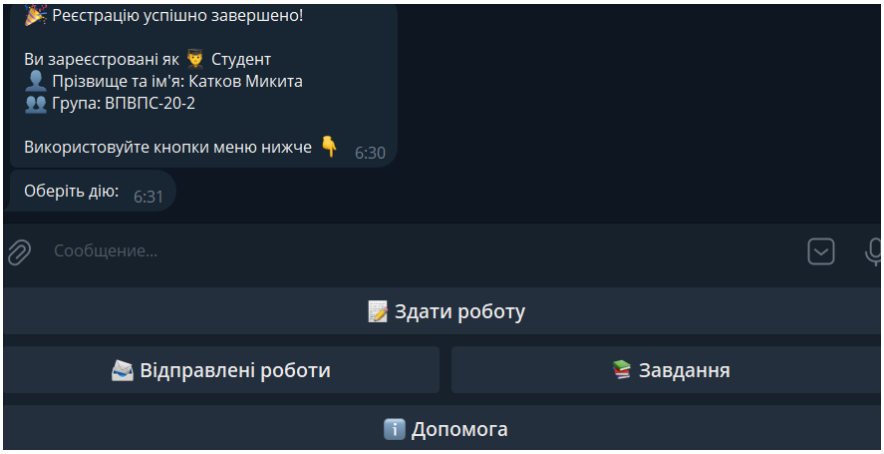


Рисунок 4.6 – Головне меню студента після завершення реєстрації

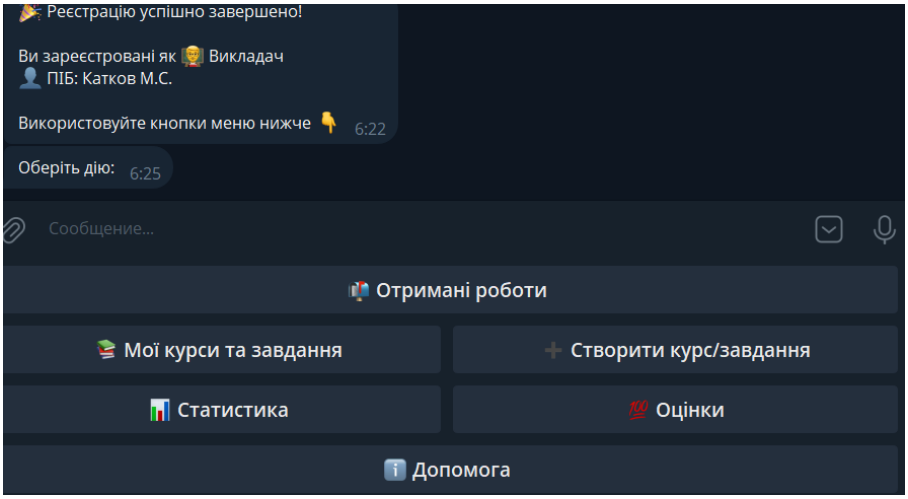


Рисунок 4.7 – Головне меню викладача після завершення реєстрації

Контроль доступу в системі реалізовано комплексно. По-перше, під час взаємодії користувача з ботом виконується прив’язка облікового запису користувача в месенджері до запису в базі даних за унікальним ідентифікатором платформи (user_id), що дозволяє визначити факт реєстрації та роль. По-друге, для команд і дій, які є критичними (створення курсів, перевірка та оцінювання), застосовуються перевірки прав доступу, які не допускають виконання викладацьких операцій студентами або неавторизованими користувачами. По-

третє, використовується обмеження частоти дій, що знижує ризики спаму та некоректних повторних натискань у чат-інтерфейсі.

Для уніфікації операцій із профілями користувачів та забезпечення єдиного підходу до доступу до даних застосовується централізований набір методів роботи з базою даних (створення користувача, отримання користувача за ідентифікатором платформи месенджера, підтвердження верифікації, оновлення профілю тощо). Фрагмент переліку основних API-методів модуля бази даних, які використовуються в межах підсистеми реєстрації та доступу, наведено на рис. 4.8.

```
# Отримати користувача за Telegram ID
user = db.get_user_by_telegram_id(telegram_id: int) -> dict

# Створити користувача
user_id = db.create_user(
    telegram_id: int,
    email: str,
    role: str,
    **kwargs
) -> int

# Верифікувати користувача
db.verify_user(user_id: int)

# Оновити дані користувача
db.update_user(user_id: int, **fields)

# Отримати завдання курсу
assignments = db.get_course_assignments(course_id: int) -> List[dict]

# Створити завдання
assignment_id = db.create_assignment(
    course_id: int,
    title: str,
    description: str = None,
    deadline: str = None
) -> int

# Отримати завдання
assignment = db.get_assignment(assignment_id: int) -> dict

# Отримати активні завдання
assignments = db.get_active_assignments() -> List[dict]
```

```
# Отримати курси викладача
courses = db.get_teacher_courses(teacher_id: int) -> List[dict]

# Створити курс
course_id = db.create_course(
    teacher_id: int,
    name: str,
    description: str = None
) -> int

# Отримати курс
course = db.get_course(course_id: int) -> dict

# Оновити курс
db.update_course(course_id: int, **fields)

# Видалити курс
db.delete_course(course_id: int)
        # Створити здачу
        submission_id = db.create_submission(
            student_id: int,
            assignment_id: int,
            comment: str = None
        ) -> int

        # Отримати здачі студента
        submissions = db.get_student_submissions(student_id: int) -> List[dict]

        # Отримати здачі на перевірку
        pending = db.get_teacher_pending_submissions(teacher_id: int)  > List[dict]

        # Оцінити роботу
        db.grade_submission(
            submission_id: int,
            grade: str,
            comment: str = None
        )

        # Повернути на доопрацювання
        db.request_submission_fix(
            submission_id: int,
            comment: str
```

```
# Реєстрація
class RegistrationStates(StatesGroup):
    choosing_role = State()
    waiting_email = State()
    waiting_code = State()
    waiting_teacher_secret = State()
    waiting_group = State()
    waiting_fullname = State()

# Курси
class CourseStates(StatesGroup):
    waiting_name = State()
    waiting_description = State()

# Завдання
class AssignmentStates(StatesGroup):
    waiting_course = State()
    waiting_title = State()
    waiting_description = State()
    waiting_files = State()
    waiting_deadline = State()

# Здача робіт
class SubmissionStates(StatesGroup):
    waiting_assignment = State()
    waiting_files = State()
    waiting_comment = State()

# Перевірка робіт
class ReviewStates(StatesGroup):
    waiting_grade = State()
    waiting_comment = State()
```

Рисунок 4.8 – Основні API методи Database

4.5 Реалізація основних процесів навчальної взаємодії

Після реалізації підсистеми доступу та керування користувачами ключовою частиною програмного засобу є підтримка основних процесів освітньої взаємодії, пов’язаних із організацією курсів, формуванням завдань, поданням робіт студентами та перевіркою результатів викладачем. Логіка системи побудована таким чином, щоб усі типові операції виконувалися у звичному чат-інтерфейсі месенджера (зокрема, WhatsApp або інших платформ

із підтримкою чат-ботів) із мінімальною кількістю кроків і без необхідності ручного введення складних команд. Для цього використовуються контекстні меню та кнопки, а багатокрокові сценарії реалізуються у формі керованих діалогів, що зменшує ймовірність помилок та забезпечує відтворюваність процедур [30].

На стороні викладача реалізовано процес створення та адміністрування курсів. Під час створення курсу система послідовно збирає необхідні атрибути, зокрема назву та опис, після чого формує запис у базі даних і переводить курс у активний стан. У межах одного інтерфейсу викладач може переглядати перелік власних курсів, редагувати назву та опис, а також деактивувати курс у разі завершення навчального циклу. Приклад діалогу створення курсу у чат-інтерфейсі наведено на рис. 4.9.

Рисунок 4.9 – Створення курсу викладачем

Наступним кроком є формування завдань у межах обраного курсу. Викладач створює завдання, задаючи назву та опис, а також (за потреби) встановлює термін виконання. У випадках, коли дедлайн не визначений, система підтримує сценарій створення завдання без терміну, що дозволяє використовувати інструмент для різних форматів навчальної роботи. Додатково передбачено можливість прикріплення файлів до завдання як вихідних матеріалів або додаткових ресурсів (методичні вказівки, шаблони звітів тощо). Створення завдання та прикріплення матеріалів демонструється на рис. 4.10. У результаті формується структурований набір активних завдань курсу, який надалі використовується студентами під час перегляду та здачі робіт.

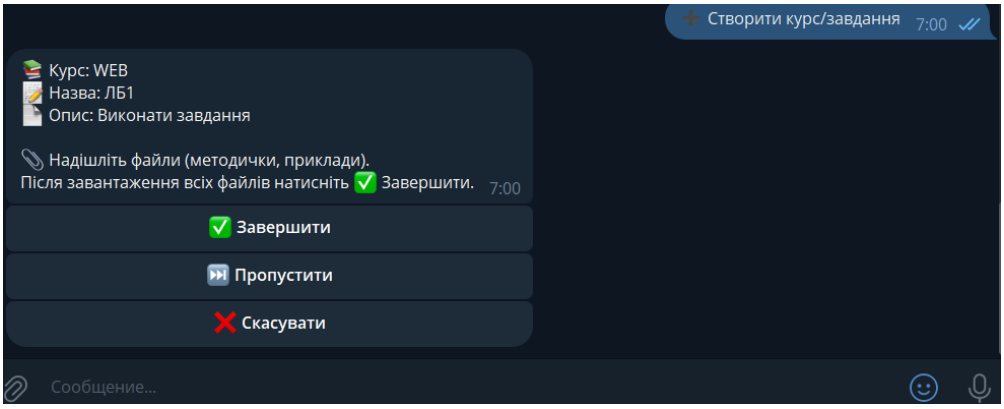


Рисунок 4.10 – Створення завдання та прикріплення матеріалів

З боку студента реалізовано процес ознайомлення з курсами та завданнями, а також подання виконаних робіт у стандартизованому форматі. Система формує перелік доступних курсів, після вибору курсу відображає актуальні завдання з основними характеристиками, включно з описом і термінами виконання. Для кожного завдання студент може переглянути прикріплені матеріали та, залежно від стану виконання, виконати одну з дій: здати роботу, переглянути власне подання або ініціювати повторну здачу у випадку повернення на доопрацювання. Приклад переходу «курси – завдання – дія для завдання» наведено на рис. 4.11.

Таким чином забезпечується прозоре відображення стану виконання завдань у межах курсу без необхідності пошуку повідомлень у загальному чаті або в сторонніх каналах комунікації.

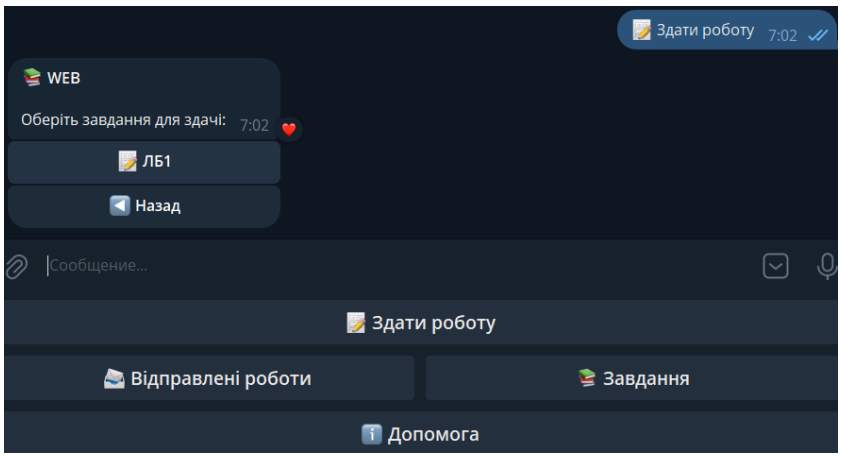


Рисунок 4.11 – Сценарій студента

Процес здачі роботи реалізовано як керований діалог, у межах якого студент може додавати вкладення у вигляді документів та зображень. Система підтримує накопичення кількох файлів у межах одного подання, а також коректно фіксує їхні метадані й прив’язку до конкретного завдання. Після завершення додавання вкладень студент підтверджує здачу та, за потреби, залишає коментар. Після підтвердження формується запис подання з початковим статусом «очікує перевірки», а студент отримує повідомлення про успішне завершення операції. Приклад діалогу здачі роботи з додаванням файлів і підтвердженням наведено на рис. 4.12.

Важливою особливістю є підтримка повторної здачі: якщо роботу повернено на доопрацювання, студент може створити нову версію подання, що дозволяє фіксувати прогрес і зберігати послідовність взаємодії.

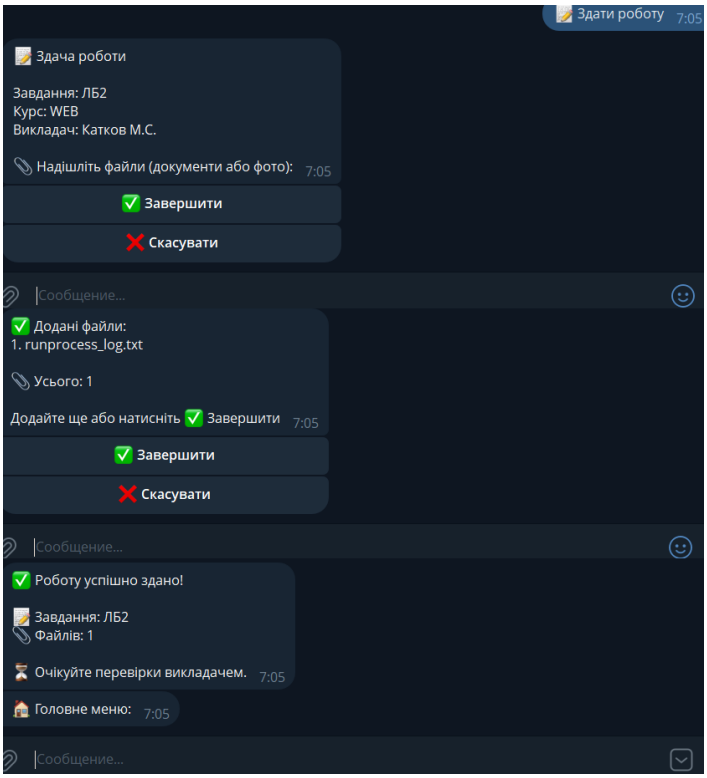


Рисунок 4.12 – Здача роботи студентом з файлами

Після створення подання автоматизується комунікаційний аспект процесу перевірки. Викладач отримує сповіщення про нову роботу на перевірку, що забезпечує оперативність реакції та зменшує ризик «втрати» подання серед

великої кількості повідомлень. Перевірка робіт реалізована через вхідну чергу, яка дозволяє послідовно переходити від вибору курсу до конкретної навчальної групи та далі – до списку студентів із наявними поданнями.

Така структура навігації зменшує часові витрати викладача на пошук потрібної роботи та одночасно підвищує керованість процесу оцінювання. Приклад переходу до перевірки та відкриття конкретного подання наведено на рис. 4.13.

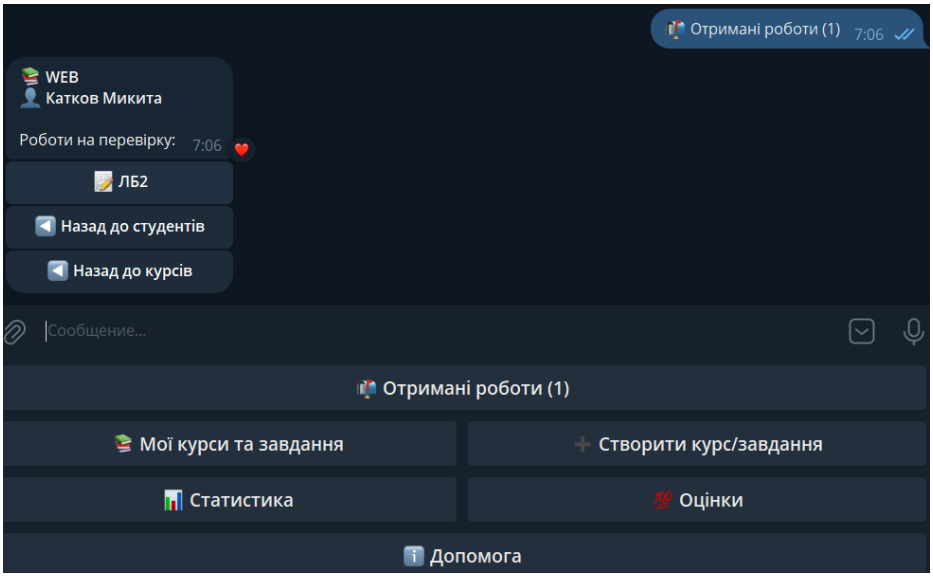


Рисунок 4.13 – Вхідна черга перевірки викладача

Оцінювання та формування зворотного зв’язку виконуються через контекстні дії, доступні для викладача під час перегляду подання. Система підтримує завантаження вкладень роботи, виставлення оцінки та додавання коментаря. За результатом перевірки викладач або фіксує остаточний результат із присвоєнням статусу «оцінено», або повертає роботу на доопрацювання зі статусом «потребує виправлення».

У кожному випадку студент автоматично отримує повідомлення з результатом (оцінкою та/або коментарем), що забезпечує прозорість і відтворюваність оцінювання. Приклад завершення перевірки наведено на рис. 4.14. Застосування фіксованих статусів із зрозумілими текстовими позначеннями дозволяє уніфікувати інтерпретацію стану роботи для обох сторін та спростити контроль виконання завдань у межах курсу.

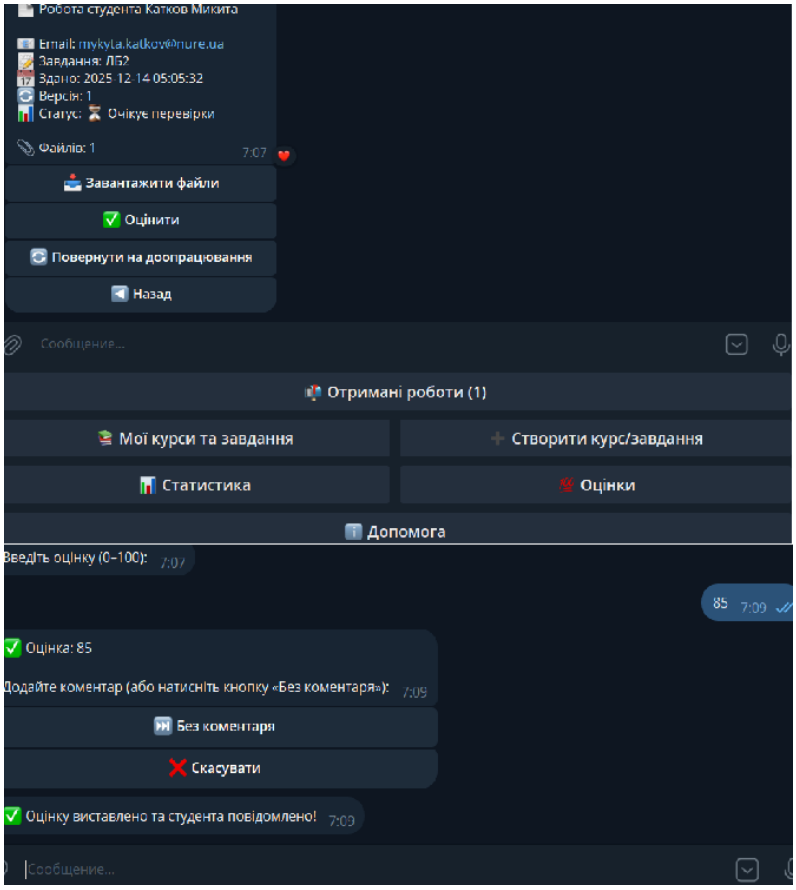


Рисунок 4.14 – Результат перевірки та повідомлення студента

Окремим елементом процесів, який завершує цикл «подання – перевірка – оцінка», є формування підсумкової звітності та вивантаження результатів у табличному форматі. Для зручності подальшого використання в навчальній документації реалізовано механізм експорту відомості в табличний файл формату електронних таблиць (зокрема, Excel), який автоматизує формування переліку студентів і їхніх результатів у межах обраного курсу та завдання.

Під час виконання цієї операції викладач у чат-інтерфейсі обирає курс і завдання, після чого система виконує вибірку даних із бази даних, агрегує інформацію про подання (актуальна версія, статус, оцінка, час подання) та формує файл .xlsx зі структурованою таблицею. Приклад запуску експорту та отримання сформованого файлу наведено на рис. 4.15, 4.16.

Згенерована відомість містить уніфікований набір полів, що забезпечує відтворюваність результатів і придатність документа до подальшого використання: ідентифікатор або ПІБ студента, академічну групу, назву курсу

та завдання, статус подання, підсумкову оцінку, дату/час останнього оновлення подання, а також (за потреби) короткий коментар викладача. Така форма подання дозволяє швидко отримати зріз поточного стану виконання завдань у групі, контролювати наявність робіт, що потребують доопрацювання або ще не перевірені, а також використовувати результати як основу для підсумкових відомостей і статистичного аналізу ефективності впровадження системи. Автоматизація експорту зменшує кількість ручних операцій (переписування оцінок, пошук робіт, узгодження статусів), а отже напряму підтримує вимоги щодо скорочення адміністративних витрат часу та підвищення прозорості оцінювання.

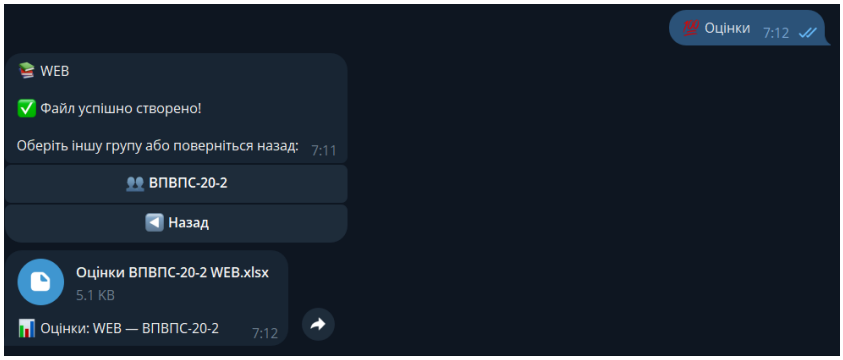
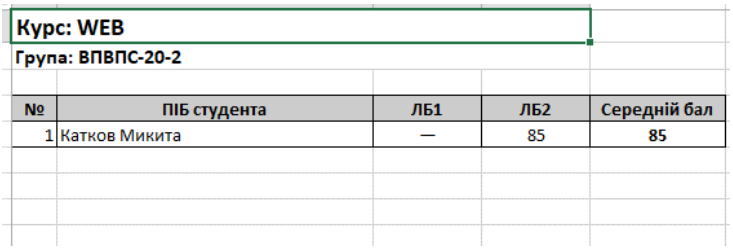


Рисунок 4.15 – Експорт відомості в Excel



Курс: WEB

Група: ВПВПС-20-2

№	ПІБ студента	ЛБ1	ЛБ2	Середній бал
1	Катков Микита	—	85	85

Рисунок 4.16 – Приклад отриманого файлу Excel

Отже, реалізовані процеси забезпечують повний цикл навчальної взаємодії в середовищі месенджера з підтримкою чат-ботів: від створення курсів і завдань до здачі, перевірки та інформування про результати. Така організація дозволяє централізувати ключові операції в одному каналі комунікації, зменшити кількість рутинних дій та забезпечити прозору

фіксацію подій, що є критично важливим для подальшого аналізу ефективності впровадження та формування звітності.

4.6 Тестування та перевірка відповідності поставленим задачам

Метою тестування розробленого чат-бота в месенджері є перевірка коректності реалізації функціональних сценаріїв, відповідності вимогам щодо доступу та безпеки, а також підтвердження стабільності роботи системи під час типових операцій навчальної взаємодії. Оскільки програмний засіб працює у форматі діалогового інтерфейсу, основну увагу зосереджено на функціональному тестуванні (black-box) з перевіркою завершеності сценаріїв та правильності переходів між станами, а також на перевірці обробки помилкових дій користувача.

Тестування проводилося шляхом поетапного програвання ключових сценаріїв для двох ролей – студента та викладача, із контролем того, що кожна дія призводить до очікуваного результату в чаті та коректно фіксується в базі даних. Для забезпечення відтворюваності перевірок використовувалися тестові облікові записи студента й викладача, тестовий курс і декілька завдань із різними параметрами (з дедлайном і без дедлайну), а також набори вкладень (документи й зображення) для перевірки механізму завантаження файлів.

Першою групою тестів перевірено коректність реєстрації та керування доступом. Підтверджено, що система коректно обробляє вибір ролі, введення електронної адреси та підтвердження одноразовим кодом, після чого відкриває відповідне меню користувача. Окремо перевірено негативні випадки: введення некоректної адреси, помилкового коду підтвердження, а також повторні спроби надсилання коду в межах короткого проміжку часу. У цих ситуаціях система не переходить до наступних кроків та повертає користувачеві повідомлення з поясненням помилки, не порушуючи цілісності облікового запису. Приклад негативного сценарію реєстрації наведено на рис. 4.17.

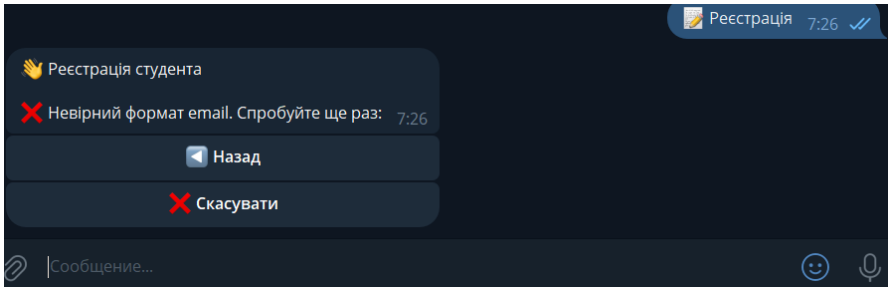


Рисунок 4.17 – Негативний сценарій реєстрації

Другою групою тестів перевірено рольове розмежування прав. Зокрема, підтверджено, що студент не має доступу до функцій створення курсів, додавання завдань, перегляду черги робіт на перевірку та виставлення оцінок. У разі спроби виконання недозволеної дії система коректно відхиляє запит, не змінюючи дані в БД, та інформує користувача про обмеження. Приклад повідомлення про недостатні права доступу наведено на (рис. 4.18). Для викладача, навпаки, підтверджено доступність повного набору операцій керування курсами, завданнями та перевіркою подань.

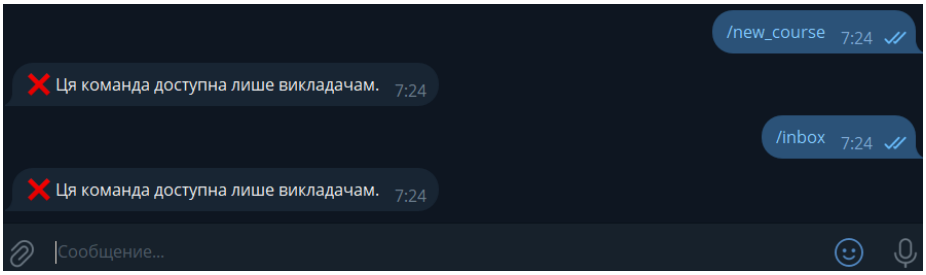


Рисунок 4.18 – Перевірка рольового доступу

Третьою групою тестів перевірено сценарії студентської взаємодії із завданнями та поданнями. Підтверджено, що студент може переглядати курси й завдання, додавати до подання декілька вкладень, завершувати здачу та отримувати підтвердження із фіксацією статусу «очікує перевірки». Окремо перевірено механізм повторної здачі у випадку повернення роботи на доопрацювання: студент може сформувати нову версію подання, а система зберігає послідовність подій та відображає актуальний статус. Приклад сценарію перездачі та зміни статусу наведено на (рис. 4.19).

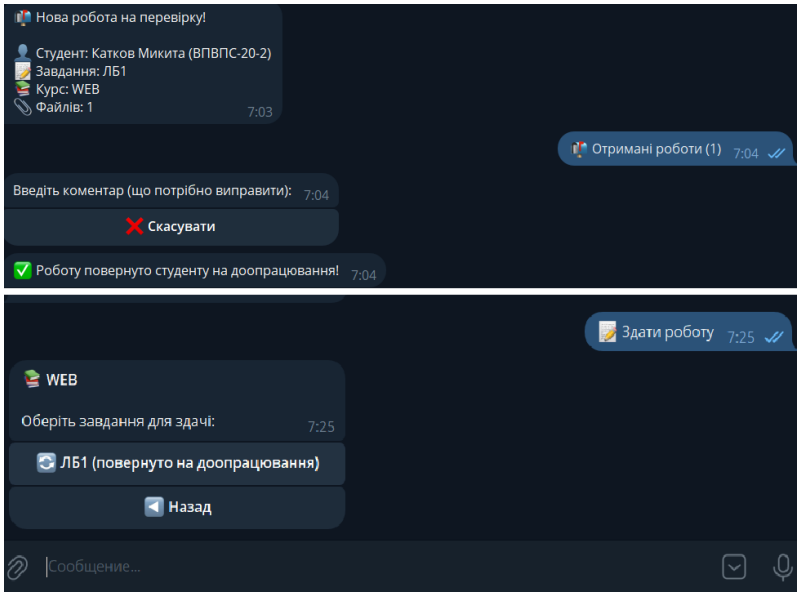


Рисунок 4.19 – Перездача та зміна статусу подання

Четвертою групою тестів перевірено процес перевірки робіт викладачем. Підтверджено коректність формування черги робіт, відкриття вкладень подання, виставлення оцінки та додавання коментаря. Також перевірено два результати перевірки: завершення оцінювання зі статусом «оцінено» та повернення роботи зі статусом «потребує виправлення». У кожному випадку студент отримує повідомлення з результатом (оцінка та/або коментар), що забезпечує прозорість і відтворюваність оцінювання.

Окремо перевірено формування звітності у форматі Excel. Підтверджено, що експорт відомості здійснюється для обраного курсу/завдання, а сформований файл містить структуровані поля (ідентифікаційні дані студента, група, статус, оцінка, часові мітки тощо) і коректно відкривається в табличному редакторі без втрати структури. Це завершує наскрізний цикл процесу та дозволяє використовувати результати оцінювання в подальшій навчальній документації.

Відповідність реалізованих можливостей поставленим завданням і вимогам підтверджено за результатами практичної перевірки ключових сценаріїв, відображених у попередніх підрозділах. Зокрема, у межах підсистеми доступу (п. 4.4) перевірено повний цикл реєстрації та ідентифікації

користувачів із вибором ролі та верифікацією електронної пошти (рис. 4.4-4.6), а також коректну обробку помилкових ситуацій під час реєстрації (рис. 4.17).

Додатково підтверджено розмежування ролей і прав доступу: студентські облікові записи не мають можливості виконувати викладацькі операції, а спроби доступу до таких функцій блокуються з відповідним повідомленням без змін у базі даних (рис. 4.18). Для уніфікації роботи з профілями користувачів перевірено коректність застосування централізованих методів доступу до даних у модулі бази даних (рис. 4.8), що забезпечує однакову логіку отримання/оновлення профілю та фіксації стану верифікації.

У межах реалізації основних процесів навчальної взаємодії (п. 4.5) підтверджено, що викладач має можливість створювати та адмініструвати курси (створення, перегляд, редагування, деактивація) у керованому діалозі (рис. 4.9), а також формувати завдання в межах курсу з підтримкою варіантів із дедлайном і без дедлайну та додаванням матеріалів (рис. 4.10). Зі сторони студента перевірено коректність навігації «курси – завдання – дія», відображення ключових параметрів завдань і поточних станів виконання (рис. 4.11). Окремо протестовано процес подання роботи як керований діалог із можливістю накопичення кількох файлів та підтвердженням здачі, після чого система фіксує подання в БД і повертає студенту повідомлення про успішне завершення операції (рис. 4.12). У випадку повернення роботи на доопрацювання підтверджено підтримку повторної здачі з коректною зміною статусів і збереженням послідовності подій, що продемонстровано у сценарії перездачі (рис. 4.19).

Для викладацької частини процесу додатково підтверджено сповіщення про нові подання та коректну навігацію до черги робіт на перевірку (рис. 4.13). Під час перевірки подань протестовано відкриття вкладень, виставлення оцінки, додавання коментаря та фіксацію результату зі статусами «оцінено» або «потребує виправлення», після чого студент автоматично отримує повідомлення з результатом (рис. 4.14). Завершальним елементом перевірки відповідності вимогам є формування підсумкової звітності: підтверджено експорт відомості у форматі Excel із уніфікованими полями (статуси, оцінки,

часові мітки тощо) та коректне отримання сформованого файла в чаті (рис. 4.15-4.16). Таким чином, практична перевірка показала, що реалізовані функції покривають повний цикл взаємодії «курс – завдання – подання – перевірка – оцінка – звітність» і відповідають вимогам, сформульованим у пп. 4.4-4.5, із підтвердженням коректної обробки помилкових дій і відсутності некоректних змін у даних при відхиленні операцій (рис. 4.17-4.18).

Узагальнення перевірки показало, що чат-бот NureStudy Bot забезпечує завершений цикл «курс – завдання – подання – перевірка – оцінка – звітність» у межах одного чат-інтерфейсу та демонструє відтворюваність результатів: однакові дії в однакових умовах приводять до однакових станів подань і однаково сформованих результатів (статуси, оцінки, експорт). Додатково підтверджено керованість діалогів (послідовні кроки, зрозумілі відповіді системи), а також коректність рольового доступу, що знижує ризик несанкціонованого виконання критичних операцій.

Рекомендації щодо подальшого вдосконалення та розширення функцій (як етап розвитку після апробації практичного макета):

– впровадити прив’язку академічної групи до конкретного групового чату (або кількох чатів) із автоматичним визначенням групи користувача за фактом участі в чаті та ролі;

– реалізувати автоматичне формування складу групи на основі учасників чату: викладач додає чат-бота до групового чату, після чого система формує перелік студентів (за messenger_id) і підтримує його актуальність;

– обмежити відображення курсів і завдань студенту лише тими курсами, які прив’язані до чатів, у яких студент присутній (зменшення інформаційного шуму та помилкового доступу до «чужих» курсів);

– додати механізм синхронізації складу групи (обробка вступу/виходу з чату, зміни ролей, контроль дублювань);

– розширити інструменти контролю якості даних і прозорості процесів: журнал подій для викладача (хто/коли/що здав, хто/коли оцінив), а також зручні фільтри черги подань (за групою, статусом, дедлайном);

– розглянути інтеграції з існуючими сервісами ЗВО (електронний журнал/розклад/LMS) для уникнення повторного введення даних і підвищення узгодженості інформації між системами.

З урахуванням наведеного, розділ підтверджує відповідність реалізованого макета поставленим завданням на рівні практичних сценаріїв використання та формує основу для наступного етапу – розширення та перевірка проєкту в умовах реального освітнього процесу.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Характеристика науково-дослідного рішення

Метою даного розділу є економічне обґрунтування витрат на проведення науково-дослідної роботи (НДР), в межах якої передбачається розробка та впровадження автоматизованої системи здачі та перевірки навчальних робіт "NureStudy Bot" через месенджери WhatsApp, Viber або інші з підтримкою чат-ботів. Ця система є сучасним рішенням для автоматизації рутинних процесів взаємодії між студентами та викладачами у вищих навчальних закладах.

Під час економічного обґрунтування буде здійснено:

– розрахунок трудомісткості та заробітної плати виконавцям;

– розрахунок одноразових витрат на розробку (кошторисної вартості НДР);

– оцінку результатів та визначення економічної ефективності НДР.

Розроблена система "NureStudy Bot" забезпечує:

– зручну та швидку авторизацію студентів та викладачів;

– функціонал для керування курсами, завданнями та оцінками;

– автоматизацію процесу здачі робіт (додавання файлів, коментарів, відстеження статусу);

– автоматизацію процесу перевірки робіт (перегляд здач, оцінювання, повернення на доопрацювання);

– централізоване зберігання даних у базі SQLite.

Реалізація НДР передбачає такі основні етапи:

– аналіз предметної області та технічне завдання: дослідження існуючих методів здачі/перевірки робіт та формування вимог до автоматизованої системи;

– розробка архітектури системи: визначення структури системи (Bot Core, Handlers, Middlewares) та вибір технологічного стеку (Python, SQLite);

– розробка ядра системи та основних модулів: реалізація логіки бота, обробників команд, FSM станів та клавіатур;

– створення механізмів взаємодії з БД: розробка функцій для збереження та отримання даних про користувачів, завдання та здачі;

– впровадження функціоналу для користувачів: реалізація інтерфейсу та логіки для студентів (здача робіт) та викладачів (перевірка, оцінювання, статистика);

– тестування та документування: проведення ручного та інтеграційного тестування, усунення помилок та підготовка документації.

5.2 Етапи виконання НДР, їх трудомісткість та заробітна плата

Під час виконання НДР, спрямованої на розробку "NureStudy Bot", було проведено аналіз потреб користувачів, досліджено можливості фреймворків та спроєктовано оптимальну архітектуру для автоматизованої системи здачі та перевірки навчальних робіт.

Умовно НДР можна розділити на три етапи: підготовчий, основний і заключний:

На підготовчому етапі було виконано всебічний підбір та детальний аналіз інформації, необхідної для формування вичерпних вимог до майбутньої системи. Результатом цієї роботи стало створення деталізованого технічного завдання, яке визначило функціональні рамки проєкту. Також було остаточно затверджено технологічний стек, який базувався на мові програмування Python, фреймворків, та системи керування базами даних SQLite.

Основний етап був присвячений безпосередній реалізації архітектури та функціоналу. Були розроблені ключові модулі ядра бота, включаючи Bot Core, Handlers та Middlewares, а також впроваджена логіка кінцевих автоматів (FSM) для ефективного управління станами користувачів. Паралельно створювалися та інтегрувалися надійні механізми взаємодії з базою даних, що забезпечило стійке зберігання всієї необхідної інформації про користувачів, навчальні курси та

результати здачі робіт [28-29]. Далі було реалізовано повний функціонал як для студентів, які отримали можливість зручно здавати роботи та відстежувати їхній статус, так і для викладачів, які отримали інструменти для перевірки завдань, їхнього оцінювання та ведення детальної статистики успішності.

Заключна частина передбачала фінальне забезпечення якості та готовності системи до експлуатації. Було проведено комплексне тестування всієї системи, що дозволило виявити та оперативно усунути всі виявлені помилки. Після цього була підготовлена детальна експлуатаційна документація, необхідна для подальшого обслуговування та супроводу проєкту. Завершилася робота складанням підсумкового звіту з роботи.

Для виконання роботи було залучено команду з чотирьох фахівців, що забезпечує необхідну якість та розподіл відповідальності на всіх етапах розробки. Середня заробітна плата робітників була сформована за версією сайту work.ua:

- системний архітектор, заробітна плата – 80 000,00 грн;
- backend-розробник (Python), заробітна плата – 65 000,00 грн;
- UX/UI аналітик, заробітна плата – 45 000,00 грн;
- тестувальник, заробітна плата – 50 000,00 грн.

Проведемо розрахунок трудовитрат та заробітної плати фахівців:

$$З_{ср.дн.} = \frac{З_{ср.міс.}}{n}, \quad (5.1)$$

де $З_{ср.дн.}$ – середньомісячна зарплата фахівця;

n – число робочих днів у місяці, (n = 22).

Підставимо дані до (5.1), отримаємо середньоденну заробітну плату для кожного з фахівців. Для системного архітектора вона становить 3636,36 грн/день, backend-розробника – 2954,55 грн/день, UX/UI аналітика – 2045,45 грн/день, тестувальника – 2272,73 грн/день.

Етапи виконання НДР, перелік і зміст робіт, трудомісткість їх виконання, заробітна плата виконавців робіт представлені в табл. 5.1.

Таблиця 5.1 – Розрахунок трудовитрат і заробітної плати фахівців

Перелік робіт	Кіль-кість вико-навців	Посада виконавця	Трудо-місткість, T_i(люд.-днів)	$З_{ср.дн.}$ (грн. /день)	Основна заробітна плата, $З_{осн.i}$(грн.)
1. Підготовчий етап					
1.1 Аналіз предметної області та формування вимог	1	UX/UI аналітик	4	2 045,45	8 181,80
1.2 Розробка архітектури системи та вибір технологічного стеку	1	Системний архітектор	4	3 636,36	14 545,44
2. Основний етап					
2.1 Розробка ядра системи (Bot Core, FSM, Middleware)	1	Backend-розробник	9	2 954,55	26 590,95
2.2 Створення механізмів взаємодії з БД (SQLite)	1	Системний архітектор	3	3 636,36	10 909,08
2.3 Впровадження функціоналу для користувачів	1	Backend-розробник	11	2 954,55	32 500,05
3. Заключний етап					
3.1 Тестування системи та складання тест-кейсів	1	Тестувальник	3	2 272,73	6818,19
3.2 Налагодження системи та усунення помилок	1	Backend-розробник	4	2 954,55	11 818,20
3.3 Аналіз результатів, складання звіту та документації	1	UX/UI аналітик	2	2 045,45	4 090,90
Усього			40		115 454,61

5.3 Розрахунок одноразових витрат на розробку

Калькуляція собівартості НДР розраховується відповідно до чинних нормативних актів України. До складу калькуляції витрат на виконання роботи входять такі статті:

– матеріальні витрати;

– витрати на оплату праці;

– єдиний соціальний внесок;

– амортизація основних засобів (вартість машинного часу);

– витрати на спожиту електроенергію;

– інші витрати.

Матеріальних витрат впродовж виконання проєкту не було. Витрати на оплату праці визначено з урахуванням кількості залучених працівників та тривалості виконання робіт. Загальна сума основної заробітної плати виконавців, відповідно до розрахунків, складає 115 454,61 грн. Додаткова заробітна плата складає 10 % від основної – 11545,46 грн.

Єдиний внесок на загальнодержавне соціальне страхування (ЄСВ) – це консолідований страховий внесок, збір якого здійснюється до системи загальнообов'язкового державного соціального страхування в обов'язковому порядку та на регулярній основі. Його метою є забезпечення захисту прав застрахованих осіб та членів їхніх сімей на отримання страхових виплат за діючими видами загальнообов'язкового державного соціального страхування.

Ставка єдиного соціального внеску (ЄСВ) становить 22 % від загального фонду оплати праці (суми основної та додаткової заробітної плати). Таким чином, розмір відрахувань на ЄСВ дорівнює 27 940,02 грн.

При виконанні НДР використовувалися 3 персональні комп'ютери вартістю 75 000,00 грн кожен. Оскільки дане устаткування є власністю організації-виконавця, необхідно розрахувати суму амортизаційних відрахувань за період виконання роботи.

Амортизація основних засобів розраховується за формулою:

$$AB = \sum_{k=1}^{L} \frac{BO_k}{TE_k} \times T, \tag{5.2}$$

де AB – сума амортизаційних відрахувань, нарахований під час проведення науково-дослідницької роботи;

BOk – вартість основних засобів k-го виду;

TEk – термін експлуатації основних засобів k-го виду, днів;

T – термін науково-дослідницької роботи, днів;

L – кількість видів обладнання.

Підставивши відомі значення до формули, отримаємо величину амортизаційних відрахувань – 8219,18 грн.

Витрати на використану обладнанням електроенергію:

$$B_e = M \times t \times T_{\text{кВт}}, \quad (5.3)$$

де M – потужність устаткування, тобто кількість енергії, споживаної за одиницю часу (кВт/година);

t – кількість годин використання устаткування за період проведення науково-дослідницької роботи;

$T_{кВт}$ – тариф, тобто вартість використання 1 кВт електроенергії.

Споживна потужність одного комп’ютера складає 0,4 кВт. Враховуючи, що для виконання роботи використовувались 3 комп'ютери протягом 40 робочих днів (по 8 годин щодня), загальний час роботи устаткування становить 960 годин. Підставивши значення до формули (5.3), отримаємо витрати на спожиту електроенергію у розмірі 1658,88 грн.

До інших статей витрат відносяться адміністративні витрати (водопостачання, водовідведення, освітлення, опалення), які прийнято у розмірі 20 % від витрат на основну оплату праці (основна та додаткова заробітна плата), та вартість оплати послуг зв’язку.

Вартість оплати послуг зв’язку (доступ до мережі Інтернет) становить 500,00 грн на місяць. За 40 днів виконання НДР витрати розраховуються пропорційно часу, та складають – 666,67 грн.

Результати розрахунку кошторису витрат на виконання НДР «Дослідження процесу впровадження чат-боту для покращення освітнього процесу» наведені в табл. 5.3.

Таблиця 5.3 – Кошторис витрат на розробку НДР

№ з/п	Стаття витрат	Сума, грн
1	Заробітна плата	115 454,61
2	Додаткова заробітна плата	11 545,46
3	Єдиний соціальний внесок (22 % від п.1)	27 940,02
4	Амортизація основних засобів	8 219,18
5	Витрати на спожиту електроенергію	1 658,88
6	Інші витрати, у тому числі:	
6.1	адміністративні витрати (20 % від п.1)	25 400,01
6.2	вартість послуг зв'язку	666,67
	Усього витрати на розробку (*Вр*)	190 884,83

5.4 Оцінка результатів науково-дослідної роботи

Оцінка результатів НДР – це визначення ефективності отриманих рішень порівняно з сучасним науково-технічним рівнем. Оскільки метою роботи є покращення освітнього процесу, основним результатом впровадження чат-боту є покращення характеристик системи (зручність, швидкість, прозорість) та економія часу викладачів і студентів.

Результат від впровадження НДР у даному випадку визначається як покращення j-тої характеристики процесу за формулою:

$$\Delta P_j = X_{\text{н}j} - X_{\text{б}j}, \tag{5.4}$$

де $X_{\text{б}j}$ – базове значення характеристики (традиційний метод здачі);

$X_{\text{н}j}$ – нове значення характеристики після впровадження чат-боту.

У якості досліджуваної характеристики обрано середній час адміністрування однієї лабораторної роботи (отримання файлу, фіксація здачі, зворотний зв'язок, виставлення оцінки). Час на саму перевірку (читання коду/звіту) вважаємо незмінним, оскільки це інтелектуальна праця.

Проведемо аналіз часових витрат.

Базовий варіант (через пошту): викладач витрачає час на пошук листа, завантаження архіву, ручне внесення оцінки в журнал, написання листа-відповіді. Середні витрати часу на адміністрування однієї роботи: $X_{\text{б}} = 8$ хв.

Новий варіант (чат-бот NureStudy): система автоматично сортує файли, формує чергу, сповіщає студента про оцінку. Викладач витрачає час лише на натискання кнопок інтерфейсу. Середні витрати: $X_н$ = 2 хв.

Результат (економія часу на одній роботі):

$$\Delta P_{\text{час}} = 8 - 2 = 6 \text{ хв.}$$

Розрахуємо сумарну економію часу для одного викладача, який веде 4 академічні групи (100 студентів) і приймає 5 лабораторних робіт та 5 практичних занять за семестр (20 за навчальний рік).

Загальна кількість робіт, що підлягають перевірці за рік:

$$N_{\text{робіт}} = 100 \text{ студ.} \times 20 \text{ робіт} = 2000 \text{ здач на рік.}$$

Загальна економія часу в годинах ($E_{\text{год}}$):

$$E_{\text{год}} = \frac{2000 \times 6}{60} = 200 \text{ годин.}$$

Це еквівалентно 25 робочим дням (при 8-годинному робочому дні), які викладач може використати для наукової або методичної роботи.

Результати порівняння ефективності процесів наведено в таблиці 5.4.

Таблиця 5.4 – Результат від впровадження НДР

Характеристика процесу	Традиційний метод	Метод з чат-ботом	Покращення
Час на обробку 1 роботи, хв.	8	2	6
Ризик втрати роботи, %	5 %	0 %	5 %
Швидкість сповіщення студента	Від годин до днів	Миттєво	У реальному часі
Зручність доступу до історії	Низька	Висока	Якісна зміна

Для оцінки ефективності витрат використаємо коефіцієнт $К_з$, який показує, наскільки кожна гривня витрат змінює характеристику процесу:

$$K_з = \frac{E}{B_p}, \tag{5.5}$$

де B_p – загальні витрати на розробку НДР (з табл. 5.3).

Підставимо наші дані до формули та отримаємо:

$$K_з = \frac{200 \text{ годин}}{190884{,}83 \text{ грн}} \approx 0{,}00105 \frac{\text{год}}{\text{грн}}.$$

При масштабуванні системи на рівні кафедри (наприклад, 20 викладачів), сукупний ефект складатиме 4000 годин економії на рік при тих самих одноразових витратах на розробку.

Додатковою перевагою розробленого рішення є платформна незалежність: чат-бот розглядається як інтерфейс доступу до функцій системи, тоді як основна логіка (обробка сценаріїв, робота з базою даних, формування звітності, керування статусами та сповіщеннями) реалізована у вигляді окремих модулів і може бути підключена до різних каналів комунікації. Завдяки цьому систему можна активувати та використовувати з будь-яким месенджером, що підтримує чат-ботів (зокрема, WhatsApp, Viber та іншими), адаптуючи лише інтеграційний модуль взаємодії з API конкретної платформи без зміни предметної моделі та ключових процесів.

Таким чином, результати розрахунків підтверджують, що використання розробленої системи дозволяє оптимізувати освітній процес за рахунок мінімізації витрат часу на адміністрування. Роботу можна вважати ефективною та економічно доцільною, що відповідає сучасному науково-технічному рівню.

ВИСНОВКИ

У кваліфікаційній роботі досліджено та апробовано підхід до впровадження чат-бота у месенджері з підтримкою бот-інтерфейсів (із практичною орієнтацією на WhatsApp) для автоматизації процесів подання, фіксації, перевірки та оцінювання навчальних робіт у закладі вищої освіти. На відміну від повноцінного промислового впровадження, отриманий результат розглядається як прототип (практичний макет), призначений для перевірки підходу та формування обґрунтованих рекомендацій щодо подальшого застосування й адаптації в освітньому процесі.

У ході виконання роботи проаналізовано особливості дистанційної та змішаної форми навчання та визначено причини фрагментації інструментів під час роботи з навчальними завданнями. Виконано огляд і порівняльний аналіз існуючих рішень (LMS та поширених способів взаємодії через месенджери), що дало змогу уточнити проблему та сформувати припущення щодо доцільності перенесення частини навчових операцій у чат-інтерфейс. Сформульовано припущення та визначено критерії оцінювання доцільності такого підходу, після чого розроблено методику отримання та обробки оцінок (експертне оцінювання й анкетування студентів) і підготовлено інструментарій збору даних.

На основі постановки задачі автоматизації конкретизовано перелік ключових процесів, що підлягають підтримці в чат-боті, та сформовано вимоги до рішення. Зокрема, уточнено склад основних сценаріїв для ролей студента і викладача, вимоги до рольового доступу, фіксації подань і результатів перевірки, журналювання подій, сповіщень, а також вимоги до якості функціонування (надійність, безпека, зручність використання та можливість розвитку). Відповідно до визначених вимог спроєктовано архітектуру прототипу та модель даних, що забезпечують централізоване збереження інформації про курси, завдання, подання, статуси й оцінки та

підтримують відтворюваність операцій упродовж повного циклу роботи із завданнями.

Реалізовано прототип чат-бота, який підтримує наскрізний цикл «курс – завдання – подання – перевірка – оцінювання – звітність» у діалоговому інтерфейсі. Проведено практичну перевірку відповідності реалізованих можливостей поставленим завданням і вимогам: підтверджено коректність реєстрації та ідентифікації користувачів, рольового розмежування прав, створення курсів і завдань, подання робіт із вкладеннями, повторного подання після повернення на доопрацювання, виставлення оцінки та коментарів викладачем, а також формування підсумкової відомості з експортом у табличний формат. Окремо перевірено обробку помилкових дій користувачів і відсутність некоректних змін у даних у разі відхилення дії системою.

Обраний підхід, за результатами експертного оцінювання та студентського анкетування, показав позитивне сприйняття і прийнятність використання чат-бота як інтерфейсу для типових навчальних операцій, а узгодженість експертних суджень підтверджує надійність отриманого узагальненого висновку. За результатами студентського анкетування встановлено, що месенджери є звичним каналом навчальної комунікації: 87,1 % респондентів користуються ними щодня, 74,2 % – часто застосовують їх для навчальних питань; при цьому 58,1 % висловили готовність регулярно користуватися чат-ботом, 51,6 % вважають підхід актуальним і потрібним, а 53,2 % готові рекомендувати його одногрупникам. Узагальнені результати експертного оцінювання також підтверджують доцільність використання чат-бота: інтегральний показник становить $Q = 0{,}862$, що відповідає $Q_5 = 4{,}31$ бала за п’ятибальною шкалою, а коефіцієнт конкордації Кендалла $W = 0{,}509$ свідчить про статистично значущу узгодженість суджень. Отримані результати дозволяють обґрунтованість обраного рішення та можливість подальшого вдосконалення проєкту.

Для цього сформовано рекомендації щодо подальшого розвитку рішення та підвищення практичної придатності: доцільним є розширення механізмів

прив’язки навчальних груп та курсів до конкретних чатів, автоматизоване отримання переліку студентів із групового чату, обмеження видимості курсів і завдань для студента лише тими навчальними просторами, до яких він фактично приєднаний, а також розвиток засобів аналітики, журналу подій та гнучкого експорту звітності.

Таким чином, мети роботи досягнуто: процес впровадження чат-бота досліджено, прототип розроблено та перевірено, а також підготовлено рекомендації для подальшої адаптації й практичного застосування в освітньому процесі. Проведена практична перевірка підтвердила, що прототип реалізує та покриває повний перелік необхідних функцій, сформований за результатами експертного оцінювання й опитування студентів (подача робіт, фіксація статусів, перевірка та оцінювання, сповіщення, формування звітності), і забезпечує централізований та прозорий електронний супровід навчальних робіт. Унаслідок цього зменшуються рутинні адміністративні витрати часу та ризики втрати або дублювання матеріалів, що в цілому підвищує керованість і ефективність організації навчального процесу.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. OptimaSchool. (2025). Як працює система зворотного зв'язку в дистанційній школі: коментарі, оцінки, консультації. https://optima.school/blog/ak-pracue-sistema-zvorotnogo-zvazku-v-distancijnij-skoli-komentari-ocinki-konsultacii/.

2. КППФК. (б. д.). Комп’ютерні технології в освітньому процесі. https://kppk.com.ua/ELLIB/ebook/Gorbenko/ICTEd/1/1.htm/.

3. Cluelabs. (2023). Інтеграція Чатбот-Рішень для Персоналізації Контенту в Електронному Навчанні. https://cluelabs.com/blog/інтеграція-чатбот-рішень-для-персона/.

4. Міністерство освіти і науки України. (2020). Методичні рекомендації щодо організації дистанційного навчання у закладах освіти. https://mon.gov.ua/npa/shodo-organizaciyi-distancijnogo-navchannya/.

5. SendPulse. (2024). Як створити чат-бот для сфери освіти. https://sendpulse.ua/blog/chatbot-for-education-sphere/.

6. UKRBOT. (б. д.). Чат боти: плюси і мінуси. https://ukr-bot.com/plyusi-ta-minusi-chat-botiv/.

7. Itweek. (2022). Чат-бот у Viber допоможе вчителям в онлайн-навчанні. https://itweek.com.ua/2022/09/08/chat-bot-u-viber-dopomozhe-vchytelyam-v-onlajn-navchanni/.

8. SendPulse. (2025). Рекомендації щодо створення ланцюжків чат-ботів. https://sendpulse.ua/knowledge-base/chatbot/flow-design/.

9. HBJ. (2025). Чат-боти на основі GPT: практичні кейси використання у бізнесі. https://journal.gen.tech/post/chat-boty-gpt-praktyka-biznes/.

10. Міністерство цифрової трансформації України. (2021). Цифровізація та розвиток IT – драйвер змін та освітніх інновацій в Україні. https://thedigital.gov.ua/news/education/tsifrovizatsiya-ta-rozvitok-it-drayver-zmin-ta-osvitnikh-innovatsiy-v-ukraini/.

11. Кабінет Міністрів України. (2021). Розпорядження від 3 березня 2021 р. № 167-р «Про схвалення Концепції розвитку цифрових компетентностей». https://zakon.rada.gov.ua/laws/show/167-2021-%D1%80#Text/.

12. Marketer. (2025). Дослідження мобільних застосунків: ТОП-20. https://marketer.ua/ua/top-20-most-popular-apps-of-the-year-in-general-and-separately-for-android-and-ios/.

13. Moodle. (n. d.). Moodle Documentation. About Moodle. https://docs.moodle.org/en/About_Moodle/.

14. На Урок. (2025). Дистанційне навчання: Як воно працює, чому це важливо і як його реалізувати. https://naurok.com.ua/distanciyne-navchannya-yak-vono-pracyue-chomu-ce-vazhlivo-i-yak-yogo-realizuvati-462916.html/.

15. На Урок. (2022). Як організувати ефективне онлайн-навчання та комунікацію з учнями: корисні поради та лайфхаки. https://naurok.com.ua/post/yak-organizuvati-efektivne-onlayn-navchannya-ta-komunikaciyu-z-uchnyami-korisni-poradi-ta-layfhaki/.

16. Google. (n. d.). Довідка Google Classroom. https://support.google.com/edu/classroom/?hl=uk#topic=10298088/.

17. EducateMe. (2025). Moodle vs Google Classroom: Comparison For Those Who Have Doubts. https://www.educate-me.co/blog/moodle-vs-google-classroom/.

18. Cluelabs. (2023). Чат-боти як Персоналізовані Допоміжні Засоби для Навчання: Можливості та Виклики. https://cluelabs.com/blog/чат-боти-як-персоналізовані-допоміжн/.

19. FlashNews. (2025). Штучний інтелект в освіті: як чат-боти змінюють школу і що робити далі. https://flashnews.com.ua/news/stucnii-intelekt-v-osviti-iak-cat-boti-zminiuiut-skolu-i-shho-robiti-dali/.

20. ResearchGate. (2024). Мобільне навчання в освітньому процесі. https://www.researchgate.net/publication/378980506_Mobilne_navcanna_v_osvitnomu_procesi/.

21. Часопис. (2011). Методика проведення та оцінки результатів експертних оцінок. https://umj.com.ua/uk/publikatsia-9571-metodika-provedennya-ta-ocinki-rezultativ-ekspertnix-ocinok-na-prikladi-vprovadzhennya-sistemi-monitoringu-zdorov-ya-naselennya-na-rivni-pervinnoi-mediko-sanitarnoi-dopomogi/.

22. НаУрок. (2025). Інструкція зі створення опитувань у Google – формах. https://naurok.com.ua/instrukciya-zi-stvorennya-opituvan-u-google---formah-451620.html/.

23. Gerabot. (2023). Як створити структуру чат-бота? https://gerabot.com/article/yak_stvoriti_strukturu_chatbota/.

24. Refactoring.Guru. (н.д.). Патерни проєктування. Структурні патерни. https://refactoring.guru/uk/design-patterns/structural-patterns/.

25. Aws. (2025). What is EDA (Event-Driven Architecture)? https://aws.amazon.com/what-is/eda/

26. Hillelblog. (2025). 15 принципів якісного коду: як писати зрозуміло, чисто й ефективно. https://blog.ithillel.ua/articles/principles-of-quality-code/.

27. Hillelblog. (2022) Введення в SQLite Python. https://blog.ithillel.ua/articles/introduction-to-sqlite-python/.

28. ServBay. (2025). Керування та використання бази даних SQLite 3. https://support.servbay.com/uk/database-management/getting-started/sqlite3-management-and-usage/.

29. Aiogram. (2025). Кінцевий автомат (FSM). https://docs.aiogram.dev/uk-ua/latest/dispatcher/finite_state_machine/index.html/.

30. Meta. (2025). About the WhatsApp Business Platform. https://developers.facebook.com/documentation/business-messaging/whatsapp/about-the-platform/.