

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проєктами

Кафедра програмного забезпечення автоматизованих систем

Спеціальність 121 «Інженерія програмного забезпечення»

Освітня програма «Інженерія програмного забезпечення»

«Допущений до захисту»

Завідувач кафедри



— проф. Приходько С.Б.

«___» _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти «бакалавр»

на тему: Розробка чат-боту для конференції ITMAS

Виконав: студент групи 4151



Коростельов Я.О.

(підпис, ПІБ)

Керівник роботи:

Доцент, к.т.н., доцент

(посада, науковий ступень вчене звання)



Пухалевич А.В.

(підпис, ПІБ)

Миколаїв — 2025 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проєктами
 Кафедра програмного забезпечення автоматизованих систем
 Спеціальність 121 «Інженерія програмного забезпечення»
 Освітня програма «Інженерія програмного забезпечення»

 **«ЗАТВЕРДЖУЮ»**
 Гарант освітньої програми
 _____ доц. Макарова Л.М.
 (підпис)
 « 08 » 04 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти «бакалавр»

Студенту Коростельову Ярославу Олеговичу
 (Прізвище, ім'я, по батькові)

1. Тема роботи: Розробка чат-боту для конференції ITMAS

Керівник роботи Пухалевич Андрій Володимирович
 Затверджені наказом ректора №153 від 08.04.2025 р.

2. Термін подання роботи: 02.06.2025 р.

3. Вихідні дані по роботі: _____

4. Перелік питань, що належать до розробки (найменування розділів) _____

Титульний аркуш; Завдання на кваліфікаційну роботу; Анотація (українською та англійською мовами); Зміст; Перелік умовних позначень, символів, одиниць та термінів (при необхідності); Вступ; Аналіз практичної задачі інженерії програмного забезпечення; Проєкт програмного забезпечення; Результати розробки програмного забезпечення; Розділ з охорони праці; Висновки; Список використаних джерел; Додатки (технічне завдання, текст програми, опис програми, інструкція користувача, програма та методика випробувань програмного забезпечення).

5. Перелік презентаційних матеріалів 1. Титульний слайд; 2. Мета та завдання кваліфікаційної роботи; 3. Аналіз вимог до програмного забезпечення; 4. Архітектура програмного забезпечення; 5. Діаграма варіантів використання; 6. Діаграма діяльності; 7. Діаграма класів; 8. Діаграма послідовності; 9. Логічна модель даних; 10. Фізична модель даних; 11. Результати розробки; 12. Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
4	Гурець Н.В., ст. викладач		

7. Дата видачі завдання 08.04.2025 р.

КАЛЕНДАРНИЙ ПЛАН

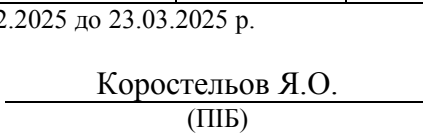
Номер	Назва етапів роботи	Терміни виконання	Примітка
1.	Підготовка розділу ВСТУП	31.03.2025	ПП*
2.	Аналіз практичної задачі інженерії ПЗ	07.04.2025	ПП*
3.	Аналіз вимог до ПЗ	14.04.2025	ПП*
4.	Розробка архітектури ПЗ	21.04.2025	
5.	Розробка проекту ПЗ	05.05.2025	
6.	Реалізація та тестування ПЗ	12.05.2025	
7.	Підготовка розділу Результати розробки ПЗ	16.05.2025	
8.	Підготовка розділу з охорони праці	19.05.2025	ПП*
9.	Підготовка розділу ВИСНОВКИ	23.05.2025	
10.	Оформлення списку використаних джерел та додатків	26.05.2025	
11.	Подання на кафедру ПЗАС тексту остаточного варіанту роботи, підписаного її керівником, у роздрукованому та електронному форматі разом із заявами щодо самостійності виконання роботи та ідентичності друкованої та електронної версій роботи (Додатки 1 і 2 «Порядку здійснення заходів з перевірки робіт на наявність текстових збігів/ідентичності/схожості із використанням програмно-технічних засобів», який введений в дію наказом ректора НУК за №20 від 20.01.2020 р.)	02.06.2025	не пізніше, ніж за 14 діб до захисту (згідно п.4.1 зазначеного Порядку
12.	Підготовка презентації та доповіді	06.06.2025	
13.	Попередній захист роботи на засіданні кафедри ПЗАС	09.06.2025	
14.	Подання на кафедру ПЗАС електронних версій наступних документів у форматі pdf: кваліфікаційної роботи; файлу-опису кваліфікаційної роботи (згідно Додатку до наказу ректора НУК за №287-уч від 19.05.2020 р.); презентації доповіді, а також Авторського договору з додатком	16.06.2025	

* - за результатами переддипломної практики (ПП), яка була з 03.02.2025 до 23.03.2025 р.

Студент


(підпис)

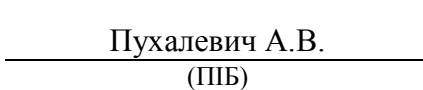
Коростельов Я.О.


(ПІБ)

Керівник роботи


(підпис)

Пухалевич А.В.


(ПІБ)

АНОТАЦІЯ

Кваліфікаційна робота присвячена вирішенню практичної задачі інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розробці чат-боту для конференції ITMAS.

Було проаналізовано практичну задачу інженерії програмного забезпечення з розробки чат-боту для конференції ITMAS, розглянуто існуючі аналоги та виконано постановку задачі на розробку чат-боту для конференції ITMAS. Здійснено аналіз вимог до чат-боту для конференції ITMAS. Розроблено проєкт чат-боту для конференції ITMAS.

Кваліфікаційна робота викладена на 89 сторінках друкованого тексту та містить 22 рисунки, 6 таблиць, 5 додатків та список використаних джерел з 21 найменування.

ABSTRACT

The qualification work is dedicated to solving a practical software engineering problem characterized by complexity and uncertainty of conditions, namely the development of a chatbot for the ITMAS conference.

The practical software engineering problem of developing a chatbot for the ITMAS conference was analyzed, existing analogues were considered and the task for developing a chatbot for the ITMAS conference was formulated. The requirements for the chatbot for the ITMAS conference were analyzed. A chatbot project for the ITMAS conference was developed.

The qualification work is presented on 89 pages of printed text and contains 22 figures, 6 tables, 5 appendices and a list of sources used with 21 names.

ЗМІСТ

ВСТУП	7
1 АНАЛІЗ ПРАКТИЧНОЇ ЗАДАЧІ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS	9
1.1 Аналіз практичної задачі інженерії програмного забезпечення з розробки чат-боту для конференції ITMAS	9
1.2 Аналіз існуючого програмного забезпечення для чат-ботів	12
1.3 Постановка задачі на розробку чат-боту для конференції ITMAS	17
2 РОЗРОБКА ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS	20
2.1 Аналіз вимог до чат-боту для конференції ITMAS	20
2.1.1 Побудова діаграми варіантів використання чат-боту для конференції ITMAS.....	20
2.1.2 Специфікації варіантів використання чат-боту для конференції ITMAS.....	22
2.2 Розробка архітектури чат-боту для конференції ITMAS	26
2.3 Проєкт чат-боту для конференції ITMAS.....	27
2.3.1 Ескізний проєкт чат-боту для конференції ITMAS	27
2.3.2 Розробка інтерфейсу чат-боту для конференції ITMAS	30
2.4 Технічний проєкт чат-боту для конференції ITMAS	31
2.4.1 Діаграма класів	32
2.4.2 Діаграми послідовностей.....	36
2.4.3 Робочий проєкт чат-боту для конференції ITMAS.....	37
3 РЕЗУЛЬТАТИ РОЗРОБКИ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS.....	49
4 ОХОРОНА ПРАЦІ	51
ВИСНОВКИ.....	58
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	59
ДОДАТОК А – ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS	61

	6
ДОДАТОК Б – ТЕКСТ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS	68
ДОДАТОК В – ІНСТРУКЦІЯ КОРИСТУВАЧА ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS	73
ДОДАТОК Г – ОПИС ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS	81
ДОДАТОК Д – ПРОГРАМА ТА МЕТОДИКА ВИПРОБУВАНЬ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS	86

ВСТУП

Кваліфікаційна робота присвячена розв'язанню практичної задачі інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розробці чат-боту для конференції ITMAS [1]. Комплексність та невизначеність умов цієї задачі обумовлена великою кількістю науковців та студентів, які приймають участь в конференції та великим обсягом функціональних аспектів, таких як: реєстрація, розсилання запрошень, інформування про недоліки оформлення тез, затвердження та інформування про оплату, автоматичне інформування клієнтів про наступні події. Все це повинно буде реалізовано при розробці чат-боту для конференції.

Доцільність чат-боту для конференції ITMAS полягає у відсутності власного чат-боту, що значно б спростило роботу команди, яка забезпечує проведення конференції ITMAS, також повністю задовольняло б вимогам організаційного комітету конференції, оскільки зараз дії інформування виконуються через поштові повідомлення. Для пришвидшення інформування учасників конференції ITMAS доцільно використовувати додаткові канали, такі як телеграм. При цьому використання існуючих аналогів, наприклад, від компаній «Botsband» та «Lampa», не є доцільним через комерційну складову та перевантажений функціонал. Таким чином розробка чат-боту для конференції ITMAS є доцільною, оскільки це автоматизує реєстрацію, додавання, редагування, видалення та супроводження матеріалів конференції, а також дозволить автоматизувати процес обслуговування учасників конференції шляхом пришвидшення оформлення участі в конференції.

Метою кваліфікаційної роботи є розробка чат-боту для конференції ITMAS.

Для досягнення поставленої мети слід вирішити наступні завдання:

- 1) виконати аналіз практичної задачі інженерії програмного забезпечення з розробки чат-боту для конференції ITMAS;
- 2) проаналізувати існуючі аналоги програмного забезпечення та обґрунтувати доцільність розробки власного програмного забезпечення;
- 3) виконати постановку задачі на розробку чат-боту для конференції ITMAS;
- 4) виконати аналіз вимог до чат-боту для конференції ITMAS;
- 5) розробити архітектуру чат-боту для конференції ITMAS;
- 6) розробити проєкт чат-боту для конференції ITMAS;
- 7) виконати реалізацію чат-боту для конференції ITMAS;
- 8) провести тестування та випробування чат-боту для конференції ITMAS.

Об'єктом кваліфікаційної роботи є процес розробки чат-боту для конференції ITMAS.

1 АНАЛІЗ ПРАКТИЧНОЇ ЗАДАЧІ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ПОСТАНОВКА ЗАДАЧІ НА РОЗРОБКУ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS

1.1 Аналіз практичної задачі інженерії програмного забезпечення з розробки чат-боту для конференції ITMAS

Розробка чат-боту для конференції ITMAS є практичною задачею інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов.

Актуальність задачі визначається тим, що все більше підприємств, згідно звіту Gartner Research [2] використовують чат-боти через їх доступність 24/7 та здатність оптимізовувати рутинні процеси.

Згідно аналізу SendPulse [3] під час війни значно виріс рівень використання інструментів штучного інтелекту, а саме чат-ботів, які збирають інформацію заздалегідь визначеними сценаріями, також надають професіоналами інструкції щодо поведінки у складних ситуаціях. Так чат-ботом «Республіканський штаб солідарності» тільки за перші півтора місяці війни скористались більше 2 тисяч людей [2]. Чат-бот Shelter [3] допомагає в пошуку тимчасового прихистку у Вінницькій, Львівській, Івано-Франківській та Чернівецькій областях.

Ще один чат-бот - KARTAMAPA - спрямований на підтримку вимушених переселенців. Його головна функція - допомогти користувачам швидко знаходити найближчі об'єкти критичної інфраструктури (такі як лікарні, укриття, пункти гуманітарної допомоги тощо) за допомогою Google Maps. Розробники прагнули не лише полегшити адаптацію переселенців у нових населених пунктах, а й забезпечити швидкий доступ до важливої інформації для всіх мешканців, особливо в умовах надзвичайних ситуацій та воєнного стану.

Проведення онлайн-конференцій в умовах воєнного стану, коли можливі відключення електроенергії, потребують автоматизованого помічника, який зможе відповідати на питання, робити розсилки, інформувати учасників, незалежно від робочого часу (при організації безперервного живлення, його робота можлива 24/7).

Інтернет конференція ITMAS проводиться кожен рік на базі кафедри Програмного забезпечення автоматизованих систем Національного університету кораблебудування імені адмірала Макарова [1]. Інтернет конференція «Інформаційні технології: моделі, алгоритми, системи» проводиться кожен рік, починаючи з 2019 року. Основним напрямком конференції є публікація новітніх технологій та досліджень у сфері інформаційних технологій. Починаючи з 2025 року метою проведення конференції також є залучення та співпраця з європейськими ІТ науковцями. Залучення та супровід нових учасників конференції вимагає збільшення штату організаційного комітету інтернет-конференції, а також призводить до збільшення людино-годин на обробку матеріалів конференції. Отже, зі зростанням вимог та через складність процесу реєстрації, інформування, консультування учасників конференції та відсутність автоматизованого процесу перевірки виконання умов оформлення матеріалів, ускладнюється процес проведення конференції.

Для підвищення ефективності супроводу учасників конференції треба розробити інтелектуальний чат-бот. Чат-бот автоматизує процес консультування учасників конференції, буде виконувати реєстрацію учасників, нагадувати про важливі дати, виконувати оповіщення про процес обробки матеріалів.

Головна сторінка інтернет-конференції представлена на рисунку 1.1.

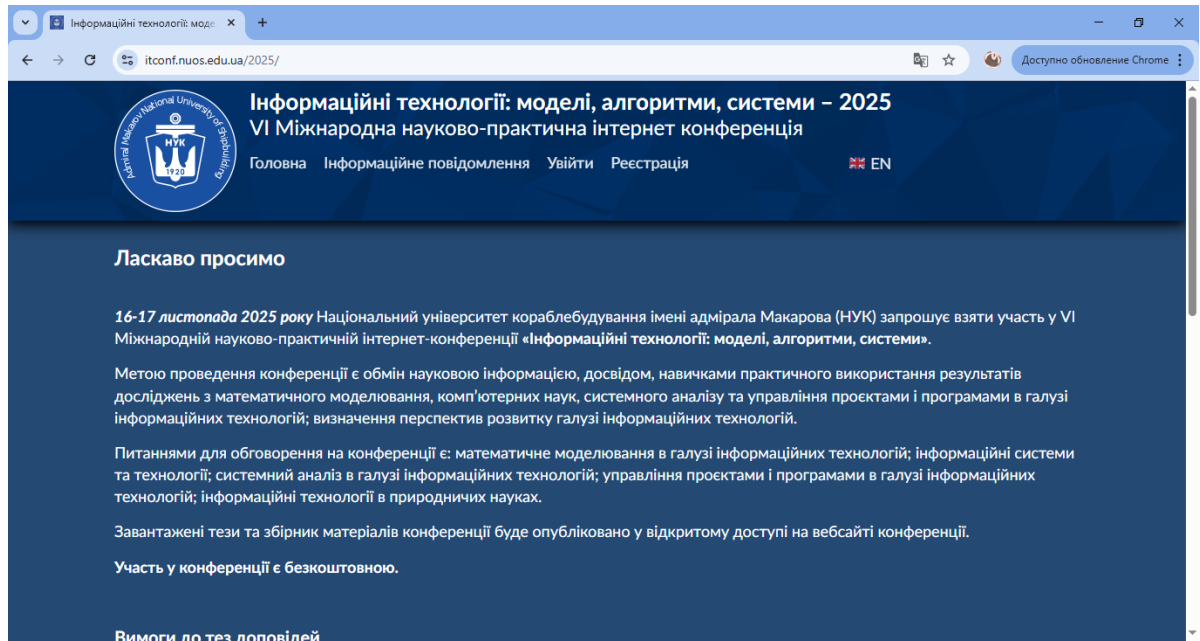


Рисунок 1.1 – Титульна сторінка сайту конференції ITMAS

Після того, як оргкомітет затвердить основні концепції конференції: тему, секції, головні дати та тайменги, вимоги до оформлення матеріалів. Координатор та секретар конференції розсилають інформаційне повідомлення про тему та напрямки роботи конференції, готують форму реєстрації, надсилають запрошення учасникам, що приймали участь у попередніх конференціях, також поширюється інформація про проведення конференції в соціальних мережах, офіційних сторінках університету. Дуже багато часу займає спілкування з учасниками конференції, відповіді на однотипні питання, щодо формату конференції, оформлення та інше. Перевірка оформлення матеріалів згідно вимог, відповідність до тематики секції та напрямку виконується редколегією, при невідповідності та наявності недоліків матеріали надсилаються на доопрацювання, секретар та редколегія конференції постійно спілкуються з учасниками, надсилають типові повідомлення та відповідають на досить типові питання, саме цей момент і вимагає автоматизації.

Після проведення конференції робота оргкомітету не завершується. Після події оргкомітет займається підсумками: обробляє результати, надсилає матеріали учасникам, збирає відгуки, аналізує статистику. Це дозволяє оцінити актуальність заходу, зробити висновки для проведення майбутніх конференцій. Також складається офіційний звіт або аналітичний

огляд конференції, саме завдяки такому підходу проведення інтернет-конференції ITMAS, вона стає справжньою платформою для обміну знаннями, досвідом і новими ідеями.

1.2 Аналіз існуючого програмного забезпечення для чат-ботів

У 2025 році чат-боти вже давно перестали бути простою новинкою - сьогодні це повноцінні цифрові помічники, які працюють безперервно, підтримують користувачів і активно залучають потенційних клієнтів [4] та учасників конференцій. Завдяки інтеграції штучного інтелекту, технологіям обробки природної мови (NLP) та можливості реагувати в режимі реального часу [4], чат-боти утримують відвідувачів, надаючи чіткі й оперативні відповіді на їхні запити.

Сучасні AI-боти аналізують поведінку користувача, спираються на великі бази знань і постійно вдосконалюються завдяки алгоритмам машинного навчання [4]. Вдало обране чат-бот-рішення значно покращує взаємодію з аудиторією сайту: забезпечує персоналізований підхід, підвищує швидкість комунікації та покращує загальний досвід обслуговування [4].

Сучасні чат-боти еволюціонують у високотехнологічні системи аутентифікації, які поєднують штучний інтелект, машинне навчання та обробку природної мови (NLP) [4]. Вони не лише аналізують контекст ситуації, а й запам'ятовують індивідуальні особливості користувача, розпізнають його емоційний стан і на основі цього надають персоналізовані рекомендації чи відповіді [4]. Завдяки цьому такі боти можуть ефективно обробляти складні запити, які не під силу традиційним системам з фіксованими продукційними правилами, крім цього чат-боти створені на основі правил важко розширювати, вони мають обмежений функціонал. AI-бот (з використанням машинного навчання та NLP) з кожною новою взаємодією удосконалюється, глибше розуміє наміри користувача й забезпечує максимально природну та адаптивну комунікацію.

Завдяки Large Language Model (LLM), чат-боти тепер можуть розуміти запитання та твердження набагато детальніше. Вони розглядають весь контекст того, що вводить користувач, а не просто вибирають одне чи два ключових слова. Це імітує розмову, подібну до людської, де користувачі відчувають себе справді почутими та зрозумілими. Сучасні технології та фреймворки дозволяють створювати чат-боти, що можуть навчатися на основі взаємодій, тобто якщо користувач неодноразово ставить запитання про певний продукт чи послугу, чат-бот адаптується, щоб наступного разу відповідати точніше. Це постійне вдосконалення може допомогти надавати якісне обслуговування учасників конференції цілодобово.

Сучасні чат-боти зі штучним інтелектом часто працюють у хмарній інфраструктурі, тобто операції виконується на віддалених серверах. Коли відвідувач вебсайту або телеграм-каналу ставить запитання, текст через API надсилається на сервери, де штучний інтелект чат-бота обробляє запит у режимі реального часу. Чат-бот формулює відповідь, надсилає її згідно responses і відображає користувачеві. AI відповідає за кодування та навчання складних моделей, безпеку та обробку даних.

Завдяки LLM чат-бот може адаптувати відповіді до кожного користувача. Наприклад, якщо відвідувач запитує про секції конференції, бот може запам'ятати ці деталі та пізніше надсилати промоматеріали наступної конференції або пов'язаних заходів, що будуть проводитися навчальним закладом, майже як людина, яка пам'ятає контекст повідомлення в чаті. Такий «персональний підхід» може дати учасникам заходу відчуття цінності та призвести до залучення більшої аудиторії.

Так фірма B2B досягла 5-кратного зростання бази користувачів після успішного запуску інтелектуального чат-боту [5]. Замість того, щоб наймати цілий парк нових співробітників, вони навчили свого чат-бота займатися адаптацією, поширеними запитаннями та вирішенням проблем із продуктом.

Чат-боти зі штучним інтелектом також скорочують час очікування, дають ґрунтовніші відповіді на запитання та звільняють оргкомітет для складних або високопріоритетних справ. Ефективно керуючи рутинними

взаємодіями, розробка зможе ефективно оптимізувати весь процес супроводу учасників конференції, потенційно заощаджуючи час та зусилля.

На сьогоднішній день існує декілька компаній, які займаються розробкою чат-ботів на комерційній основі.

Чат-бот від компанії «BotsBand» [6] для конференції WomanConf автоматизує відповіді на типові запитання, бот здатен миттєво надати ключову інформацію: розклад подій, місце проведення, зміни в програмі, посилання на реєстрацію або карту з точками інтересу. Це особливо важливо на масштабних конференціях, фестивалях, спортивних турнірах, де швидкий доступ до актуальної інформації - запорука ефективного перебування на заході.

Знімок екрана сторінки компанії «BotsBand» наведено на рисунку 1.2.

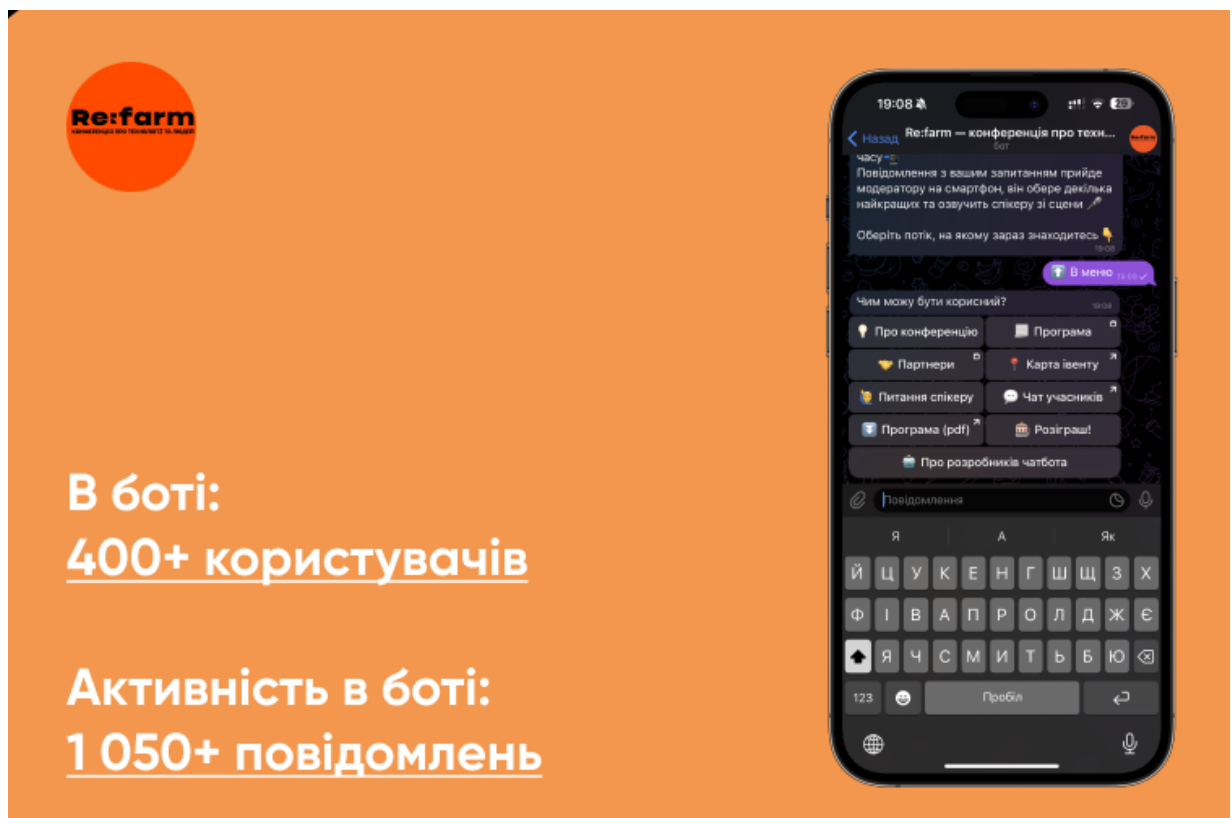


Рисунок 1.2 – Чат-бот для конференції від компанії «BotsBand».

«BotsBand» також надає можливість інтеграції з Google Maps, учасник події може одразу побачити, як дістатися до обраної локації, де розташовані укриття, санітарні пункти або зони відпочинку [6]. Ця функція особливо актуальна в умовах воєнного стану.

Компанія «SMART Chatbot» протягом багатьох років розробляє рішення для автоматизації комунікації з учасниками заходів [7]. Компанія «SMART Chatbot» створює адаптовані бот-асистенти для онлайн та офлайн заходів з функціями: відповіді на типові питання користувачів, швидкий пошук по словах та фразах, багатомовність, самостійне налаштування адміністративної частини боту клієнтом, підключення оператора контакт-центру до боту, додавання необмеженої кількості баз знань, інтеграція з ERP, CRM, CMS, обробка грошових транзакцій [7].

Знімок екрана сторінки компанії «SMART Chatbot» наведено на рисунку 1.3.

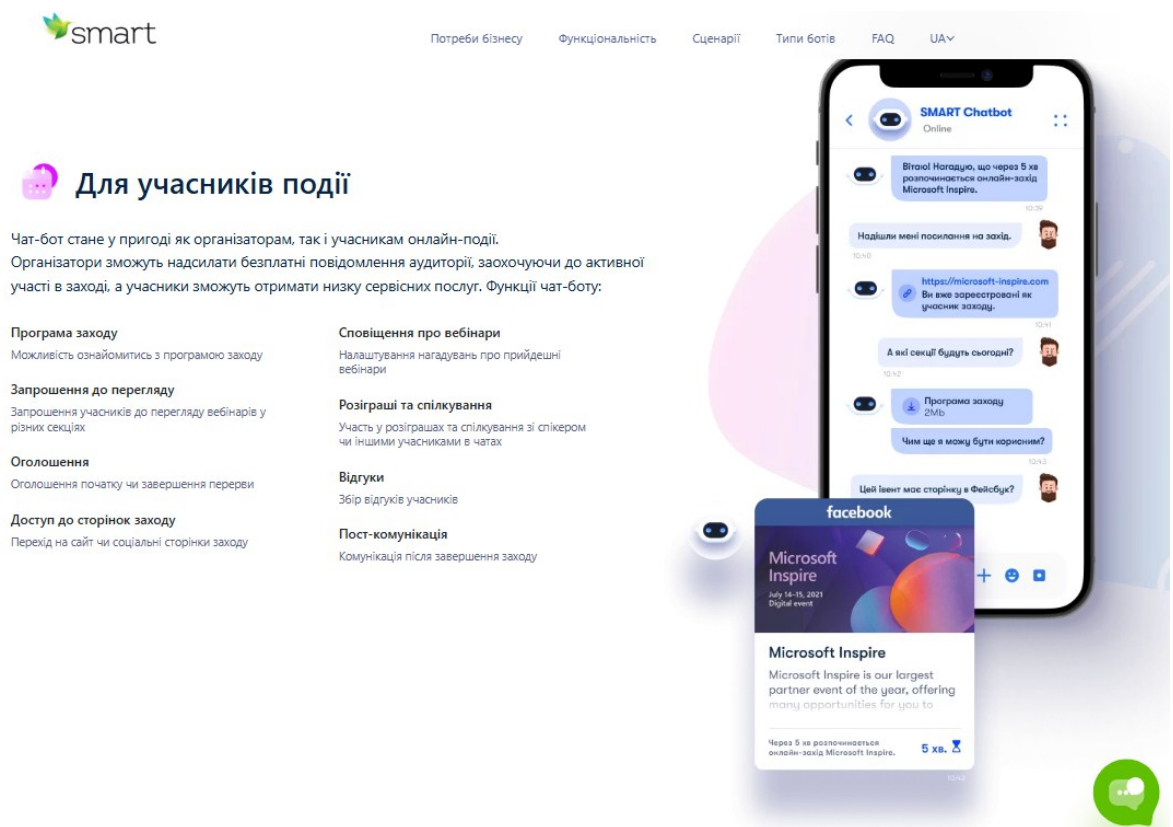


Рисунок 1.3 – Чат-бот для заходу від компанії «SMART Chatbot»

Продукти компанії «SMART Chatbot» підтримують омніканальність: Telegram, Viber, WhatsApp, Facebook Messenger, Instagram. Це означає, що учасники можуть звертатися у зручному месенджері, а організатори - централізовано отримувати й обробляти всі повідомлення, крім цього додатково платформа пропонує Smart Easy Bot - простий у налаштуванні конструктор ботів [7]. Він дозволяє швидко створити сценарії для інформування, реєстрації, збору зворотного зв'язку або навігації по події.

Компанія «Lampra» пропонує створення чат-ботів: практичних, адаптивних та доступних інструментів для бізнесу та організаторів подій [8]. Компанія презентує чат-ботів як універсальних цифрових помічників, які працюють в месенджерах (Telegram, Facebook Messenger) та на вебсайтах. Їх можна застосовувати в різних сферах: e-commerce, освіті, обслуговуванні, банківській сфері та, зокрема, на конференціях. Саме в межах заходів чат-бот Lampra може виконувати функції реєстрації, інформування учасників, пошуку локацій, а також збирання зворотного зв'язку [8].

Знімок екрана зі сторінки компанії «Lampra» наведено на рисунку 1.4.

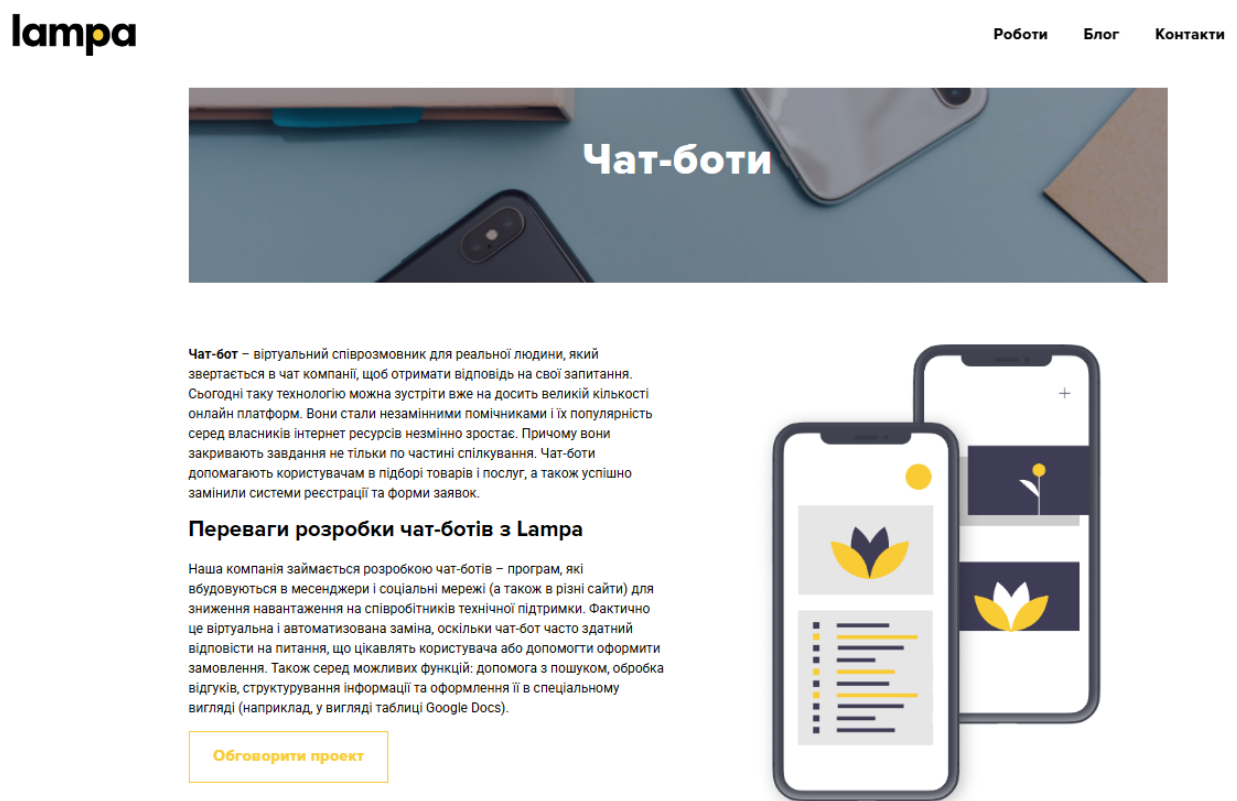


Рисунок 1.4 – Знімок екрана зі сторінки компанії «Lampra»

Робота зі створення чат-боту починається з детального аналізу потреб клієнта, розробляється технічне завдання, відбувається інтеграція рішення з CRM чи іншими системами, а потім проводиться тестування перед запуском. Такий підхід дозволяє створювати індивідуальні боти «під ключ», навіть для специфічних подій з великою аудиторією [8]. Команда компанії «Lampra» підкреслює важливість швидкої розробки, зручного інтерфейсу.

Проаналізувавши вже існуючі подібні програмні рішення зі створення чат-ботів для проведення конференцій, можна виділити наступні недоліки вже існуючих рішень:

- вже існуюче програмне забезпечення має важкий та перевантажений функціонал, який не має потреби в контексті безкоштовної інтернет-конференції;
- через функціональну перевантаженість готові рішення використовують багато запитів до серверів, що призводить до зниження реакції системи;
- деякі сервіси не призначені саме для вирішення поставленої задачі;
- сервіси не надають потрібного обсягу інформації, яка необхідна учасникам інтернет-конференції, деяка інформація представлена недостатньо, деяка взагалі непотрібна;
- головним недоліком вже існуючих рішень є досить велика вартість обслуговування.

1.3 Постановка задачі на розробку чат-боту для конференції ITMAS

Згідно проведеного аналізу практичної задачі інженерії програмного забезпечення було визначено, що для поліпшення роботи команди, яка забезпечує проведення конференції ITMAS, існує необхідність розробки чат-боту для інтернет-конференції. Розробка чат-боту для конференції ITMAS є необхідною, оскільки це автоматизує реєстрацію, додавання, редагування, видалення та супроводження матеріалів конференції, а також дозволить автоматизувати процес обслуговування учасників конференції шляхом пришвидшення оформлення участі в конференції.

Постановка задачі кваліфікаційної роботи: необхідно розробити чат-бот для конференції ITMAS.

Розробка повинна задовольняти наступним функціональним вимогам:

- інтерфейс чат-боту повинен бути інтегровано в Telegram;
- чат-бот повинен виконувати сценарій привітання;

- чат-бот повинен мати навігаційне меню, що дозволяє поділити його роботу за певними сценаріями відповідно до обраних користувачем пунктів;
- чат-бот повинен розсилати інформаційний лист про проведення інтернет-конференції за базою даних учасників конференцій минулих років;
- чат-бот повинен вміти відповідати на питання стосовно тематики проведення конференції, стосовно секцій конференції, мати відповіді та приклади матеріалів стосовно секцій конференції;
- чат-бот повинен вміти відповідати на питання стосовно важливих дат конференції ITMAS;
- чат-бот повинен вміти реєструвати учасників конференції;
- чат-бот повинен надавати консультацію та вміти завантажувати матеріали до депозитарію конференції, повинна бути можливість відмінити завантаження, повторити завантаження;
- чат-бот повинен вміти надавати відповіді учасникам конференції стосовно статусу їх матеріалів;
- чат-бот повинен проводити статистику користувачів та їхніх матеріалів;
- чат-бот повинен вміти давати відповіді стосовно оформлення матеріалів (наприклад, оформлення малюнків, таблиць та інше);
- чат-бот повинен обробляти відгуки стосовно формату проведення конференції.

Вимоги до організації вхідних та вихідних даних.

Для організації вхідних та вихідних даних чат-бот для конференції ITMAS повинен мати такі функції:

1) Реєстрація учасників конференції:

Вхідні дані: введені дані користувача (адреса електронної пошти, пароль).

Вихідні дані: успішна реєстрація учасника конференції в системі.

2) Авторизація учасників конференції:

Вхідні дані: введені облікові дані учасника конференції (адреса електронної пошти, пароль).

Вихідні дані: успішна авторизація учасника конференції в системі.

3) Запит на перегляд інформації про важливі дати події:

Вхідні дані: назва події.

Вихідні дані: успішний перегляд учасником конференції інформації про подію.

4) Запит на отримання інформації за назвою секції:

Вхідні дані: ідентифікатор назви секції.

Вихідні дані: успішне виконання пошуку учасником конференції за характеристиками.

5) Запит на отримання інформації за датою:

Вхідні дані: ідентифікатор події за датою.

Вихідні дані: успішне виконання пошуку учасником конференції за датою.

6) Завантаження тез:

Вхідні дані: файл тез.

Вихідні дані: ідентифікатор тез.

7) Видалення тез:

Вхідні дані: ідентифікатор тез.

Вихідні дані: успішне видалення тез.

8) Отримання додаткової інформації від користувачів, тези яких будуть публікуватись:

Вхідні дані: ідентифікатор користувача, повне ім'я, номер телефону та адреса користувача, місце роботи, посада, вчене звання, назва публікації.

Вихідні дані: успішне додавання інформації до системи.

9) Вихід з облікового запису:

Вхідні дані: ідентифікатор учасника конференції.

Вихідні дані: успішний вихід з облікового запису.

Деталізація постановки завдання наведена в технічному завданні, що описане у Додатку А.

2 РОЗРОБКА ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS

2.1 Аналіз вимог до чат-боту для конференції ITMAS

Під час аналізу вимог до програмного забезпечення було розроблено діаграму варіантів використання, яка відображає сценарії взаємодії користувачів з чат-ботом. Крім того, було розроблено та надано специфікацію варіантів використання, що містить докладний опис кожного сценарію. Ця специфікація включає в себе інформацію про дії користувачів та їх взаємодію з ботом. Вона допомагає визначити точні вимоги до функціональності, що дозволяє краще зрозуміти, як повинен працювати бот в реальних умовах.

2.1.1 Побудова діаграми варіантів використання чат-боту для конференції ITMAS

Після аналізу функціональних вимог до програмного забезпечення було виявлено актор «Користувач» та актор «Бот». Користувач – це особа, що використовує чат-бот в межах розробленого функціоналу. Бот – це програмне забезпечення, що імітує поведінку людини, шляхом ведення природньої розмови та надання інформаційних послуг.

Виділимо основні варіанти використання:

- переглянути інформацію стосовно конференції;
- переглянути інформацію про важливі дати
- переглянути вимоги до оформлення матеріалів;
- отримати інформацію про напрямки роботи конференції;
- здійснити реєстрацію учасників конференції;
- здійснити подання тез.

На рисунку 2.1 зображено діаграму варіантів використання чат-боту.

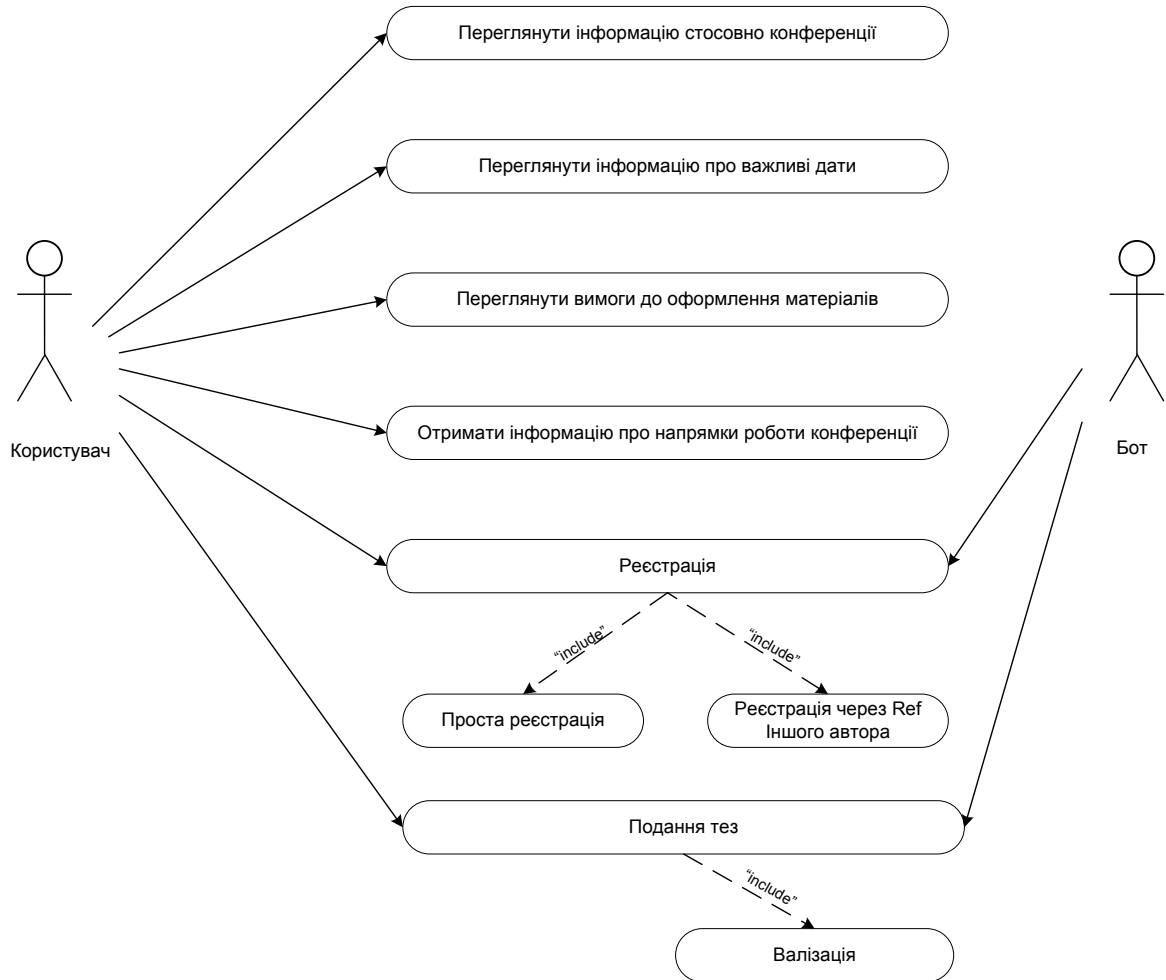


Рисунок 2.1 – Use-case діаграма чат-боту для конференції ITMAS

Опис виявлених варіантів використання чат-боту наведено в таблиці 2.1.

Таблиця 2.1 – Опис виявлених прецедентів програмного забезпечення

Дійова особа	Назва варіанту використання	Опис варіанту використання
1	2	3
Користувач	Перегляд інформації про конференцію	Отримання загальної інформації про мету, формат і особливості конференції, включно з датами, місцем проведення та контактами.
Користувач	Ознайомлення з напрямками конференції	Перегляд переліку наукових напрямів, до яких користувач може подати свою тезу або заявку.
Користувач	Вивчення вимог до оформлення тез	Доступ до інструкцій щодо структури, формату та обсягу матеріалів для подачі тез.

Продовження таблиці 2.1

1	2	3
Користувач	Перегляд важливих дат	Ознайомлення з дедлайнами подачі матеріалів, реєстрації та датами проведення заходів.
Користувач	Реєстрація учасника	Заповнення форми з особистими даними для участі в конференції.
Користувач	Альтернативна реєстрація	Використання запрошення або посилання від іншого автора для реєстрації в системі.
Користувач	Подання тез	Надсилання файлу з тезами через чат-бот або форму, з автоматичним підтвердженням отримання.
Користувач	Валідація поданих матеріалів	Отримання повідомлення про успішне прийняття матеріалів або необхідність виправити недоліки.

Була виконана реалізація варіантів використання у вигляді use-case діаграми для подальшого проєктування чат-боту для інтернет-конференції.

2.1.2 Специфікації варіантів використання чат-боту для конференції ITMAS

Конкретизуємо варіанти використання.

Специфікація варіанту використання «Реєстрація на конференцію».

Назва прецеденту: Реєстрація на конференцію

Опис прецеденту: Користувач заповнює реєстраційну форму через чат-бот, дані зберігаються, і він отримує підтвердження участі.

Основна дійова особа: Користувач (учасник, що бажає взяти участь у конференції)

Передумови: Користувач має доступ до чат-бота, готовий надати особисті дані та зацікавлений в участі.

Базовий потік: Користувач обирає пункт «Реєстрація» → вводить дані → бот перевіряє й зберігає → надсилає підтвердження з ID.

Альтернативний потік:

- Якщо користувач ввів неповні або некоректні дані:

- Бот надсилає повідомлення про помилку з поясненням, що саме не так (наприклад, некоректний email).
- Запитує повторне введення потрібного поля.
- Після коректного вводу продовжує реєстрацію та надсилає підтвердження.

Специфікація варіанту використання «Перегляд важливих дат»:

Назва прецеденту: Перегляд важливих дат

Опис прецеденту: Користувач отримує доступ до ключових термінів, пов'язаних із конференцією (дедлайни подання тез, реєстрації, проведення заходу тощо).

Основна дійова особа: Користувач (учасник конференції)

Передумови: Користувач має доступ до чат-бота; чат-бот оновлений і має дані про актуальні дати.

Базовий потік: Користувач обирає в боті пункт «Важливі дати», після чого бот відображає дати з подіями у вигляді списку, користувач може переглянути деталі кожної дати.

Альтернативний потік: (дати ще не опубліковані):

- Користувач обирає пункт «Важливі дати».
- Бот повідомляє: «На жаль, наразі календар подій ще не опублікований. Слідкуйте за оновленнями».
- Пропонує підписатися на оновлення або залишити email для автоматичного сповіщення.

Специфікація варіанту використання «Подання тез».

Назва прецеденту: Подання тез.

Опис прецеденту: Користувач надсилає тези своєї доповіді для участі в конференції через чат-бот, отримує підтвердження та очікує на результат перевірки.

Основна дійова особа: Користувач (учасник конференції)

Передумови: Користувач успішно зареєстрований у системі; має підготовлений файл із тезами згідно з вимогами.

Базовий потік:

- Користувач у чат-боті обирає пункт «Подати тези».
- Бот запитує «завантажити файл та додати короткий опис» (наприклад, напрям або ключові слова).
- Користувач надсилає файл і потрібну інформацію.
- Бот перевіряє чи зареєстрований користувач, якщо так надає підтвердження.
- Якщо користувач не зареєстрований, бот переходить до сценарію реєстрація користувача.
- Тези зберігаються у базі даних і надходять членам оргкомітету на розгляд.

Специфікація варіанту використання «Валідація поданих матеріалів»:

Назва прецеденту: Валідація поданих матеріалів

Опис прецеденту: Бот перевіряє формат файлу, та надсилає файл редколегії, для перевірки отриманих тез на відповідність технічним та змістовим вимогам, після чого змінює їхній статус та повідомляє користувача.

Основна дійова особа: Координатор (який перевіряє), користувач (отримує результат)

Передумови: Матеріали подані користувачем і збережені в базі даних, координатор має доступ до інтерфейсу перевірки.

Базовий потік:

- Координатор відкриває список поданих тез, де переглядає кожну заявку, при сумнівних питаннях координатор надсилає тези до редколегії, після перевірки додає коментар або схвалює
- бот оновлює статус,
- бот надсилає повідомлення користувачу.

Альтернативний потік (у разі необхідності доопрацювання):

- Координатор відкриває подані тези.
- Виявляє порушення вимог (некоректне форматування, відсутні дані тощо).
- Координатор додає коментар із вказаними помилками.
- Бот встановлює статус «Потребує доопрацювання».
- Бот надсилає користувачу повідомлення з поясненням і проханням виправити недоліки.

Специфікація варіанту використання «Отримати інформацію про напрямки роботи конференції»:

Назва прецеденту: Отримати інформацію про секції роботи конференції

Опис прецеденту: Користувач звертається до чат-бота, щоб дізнатися, які тематичні напрямки охоплює конференція, з метою подачі тез або вибору секції для участі.

Основна дійова особа: Користувач (потенційний учасник)

Передумови: Чат-бот доступний і містить актуальний перелік напрямків, а користувач має доступ до відповідної функції.

Базовий потік:

- Користувач обирає пункт «Секції»
- бот надсилає перелік напрямків роботи інтернет-конференції
- користувач знайомиться з описами, за потреби уточнює або переходить до подання тез.

Альтернативний потік (з уточненням напрямку):

- Користувач обирає пункт «Секції» .
- Бот надсилає перелік напрямків з короткими описами.
- Користувач обирає певний напрям (наприклад, «Інформаційні технології»).
- Бот надає розширену інформацію: тематика, ключові питання, рекомендовані секції.
- Користувач може одразу перейти до подачі тез за обраним напрямом.

2.2 Розробка архітектури чат-боту для конференції ITMAS

Архітектура чат-боту для конференції ITMAS представлена на рисунку 2.2.

Діаграма компонентів дає детальну характеристику архітектури чат-боту для конференції ITMAS разом з її компонентами, модулями та взаємодіями між ними.

Чат-бот для конференції ITMAS складається з наступних частин:

- Клієнтська частина, через яку користувач надсилає повідомлення — це стандартний застосунок Telegram.
- Сервіс Telegram, що приймає повідомлення від користувача та перенаправляє їх до Rasa через Webhook.
- Головний модуль бот-ядра Rasa Core + NLU (логічний компонент, що відповідає за класифікацію намірів, який вбудований у NLU).
 - Блок знань бота, що містить шаблонні відповіді на часті питання.
 - Модуль Tracker Store, що зберігає поточний стан діалогу.
 - Модуль Action Server, окремий сервер кастомної логіки.
 - Модуль DatabaseManager, що здійснює роботу з базою SQLite: формує запити, перетворює дані на відповіді.

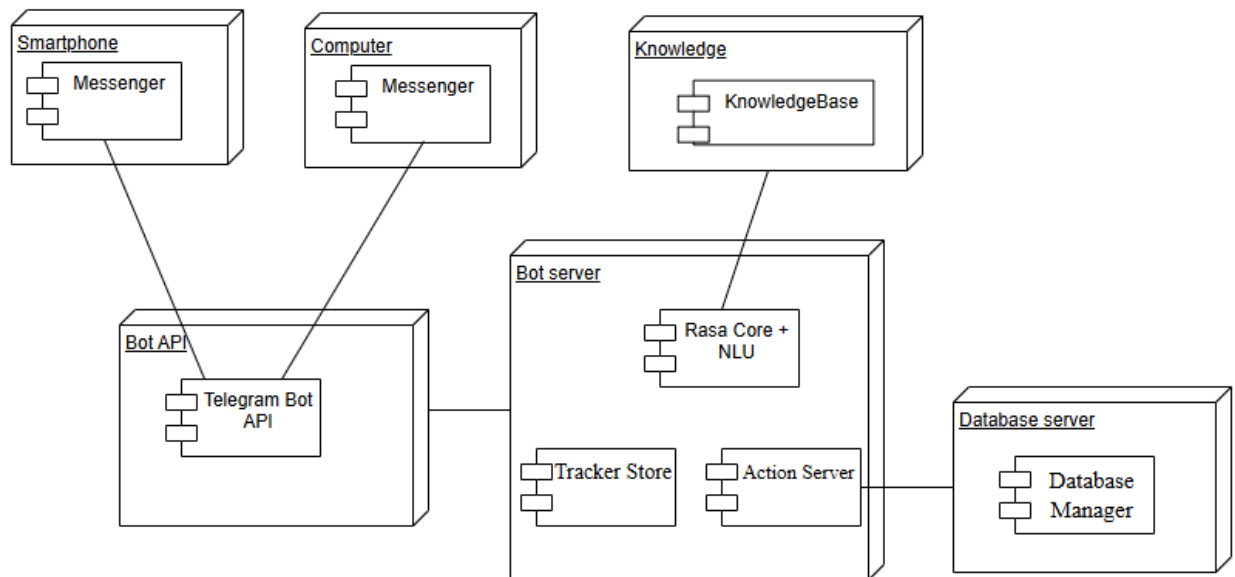


Рисунок 2.2 – Діаграма компонентів

2.3 Проєкт чат-боту для конференції ITMAS

2.3.1 Ескізний проєкт чат-боту для конференції ITMAS

Важливим етапом розробки програмного забезпечення, зокрема чат-бота для інтернет-конференції ITMAS, є формування ескізного проєкту. Саме на цьому етапі визначається загальна концепція системи, її структура та логіка роботи. Ескізний проєкт дозволяє розробникові краще усвідомити цілі й очікування замовника, а також сформуванати цілісне бачення майбутнього функціоналу чат-бота.

Завдяки розробці ескізного проєкту можна заздалегідь виявити потенційні проблеми реалізації, обрати найбільш ефективні підходи до розв’язання поставлених задач і забезпечити чітке планування розробки.

Для опису логіки функціонування чат-бота конференції ITMAS будемо використовувати діаграми діяльності, які допомагають змодельовати послідовність дій користувача та боту.

Нижче представлена діаграма діяльності, що описують ключові сценарії взаємодії користувача з чат-ботом конференції ITMAS.

Діаграма діяльності наведена на рисунку 2.3 та характеризується таким описом:

Діаграма діяльності демонструє базовий сценарій взаємодії незареєстрованого користувача з чат-ботом. На початку взаємодії бот надсилає привітальне повідомлення та запитує, чи користувач уже зареєстрований. Після спілкування чат-бота з користувачем, запускається сценарій «Реєстрація користувача», у ході реєстрації користувач вводить свої дані (ПІБ, email, установа), а бот перевіряє коректність введеного. Після успішної реєстрації користувач отримує підтвердження з унікальним ID. Далі бот намагається виявити наміри користувача, якщо користувач бажає отримати інформацію про подання матеріалів на конференцію ITMAS, тоді запускається сценарій «Виведення інформації про конференцію». Основною метою даного модуля є ознайомлення користувача з вимогами щодо оформлення тез, строками їх подачі та подальшими діями, на діаграмі цей сценарій не деталізований.

Якщо бот виявив намір «Подання тез», він відображає вимоги до оформлення файлу (формат, обсяг, структура), після чого користувач завантажує файл у вказаному форматі. Матеріали потрапляють до «кошика завантаження», звідки передаються на валідацію.

На етапі валідації перевіряється технічна відповідність файлу встановленим вимогам. Якщо помилок немає - матеріали зберігаються в базі даних, і користувач отримує підтвердження. У разі невідповідності бот інформує про проблему та пропонує повторно завантажити файл.

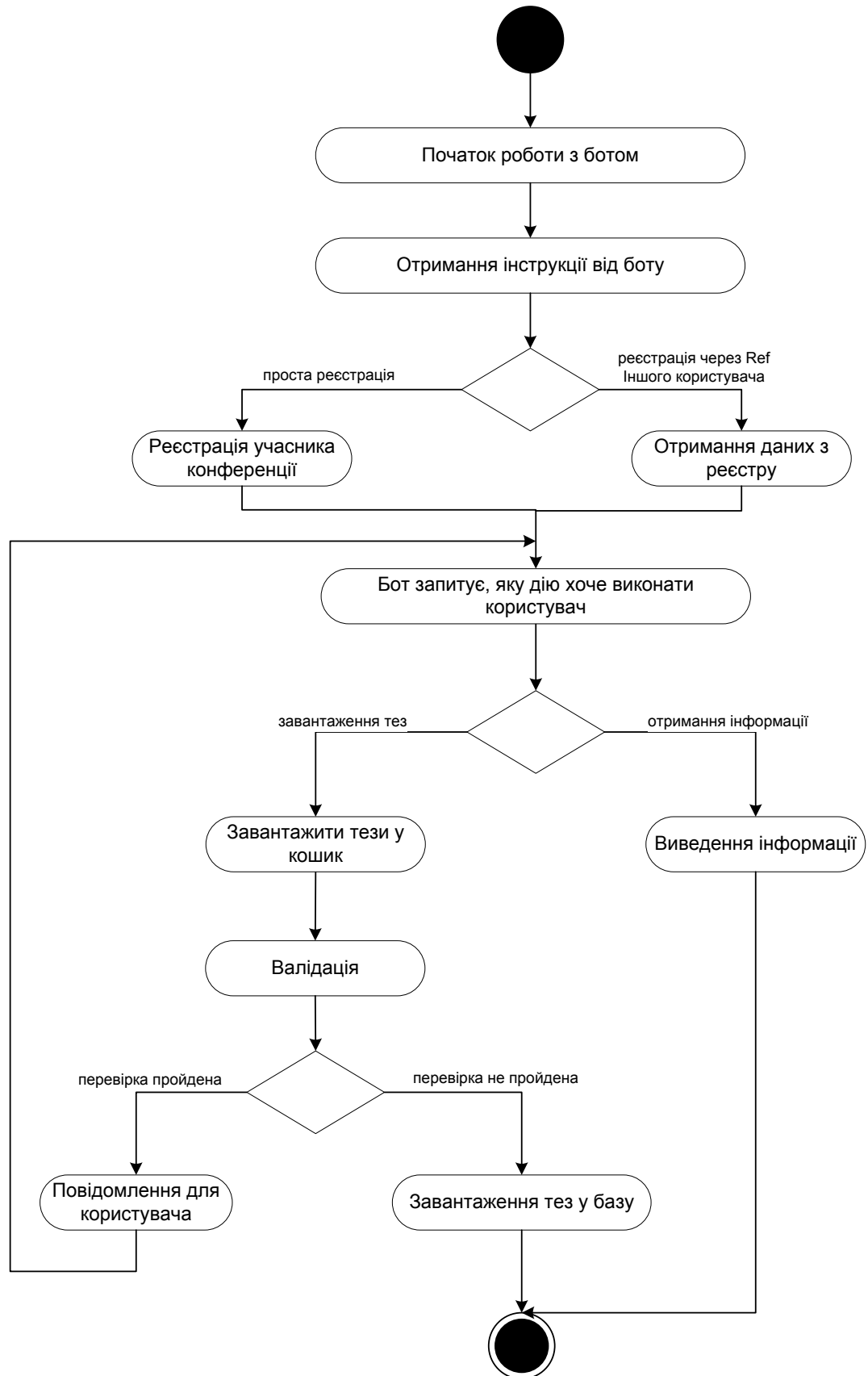


Рисунок 2.3 – Діаграма діяльності

Таким чином, діаграма описує модель взаємодії учасника конференції з чат-ботом, забезпечуючи ефективну подачу, перевірку й зворотний зв'язок.

2.3.2 Розробка інтерфейсу чат-боту для конференції ITMAS

У даному розділі кваліфікаційної роботи представлені ескізи роботи бота. На початку роботи з ботом, він буде відправляти вітальне повідомлення, через певний час, якщо не буде дії від користувача пункти за якими користувач може обрати певну дію. Вітальне повідомлення зображено на рисунку 2.4.

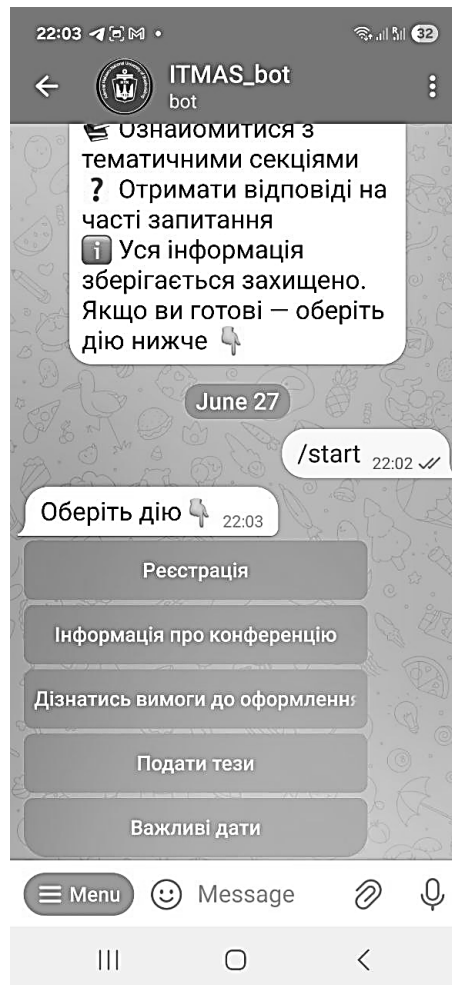


Рисунок 2.4 – Початок роботи з чат-ботом

Надання інформації стосовно напрямків роботи інтернет-конференції ITMAS буде представлено у вигляді файлів, які користувач може отримати натиснувши на пункти меню. Приклад ескізу представлено на рисунку 2.5.

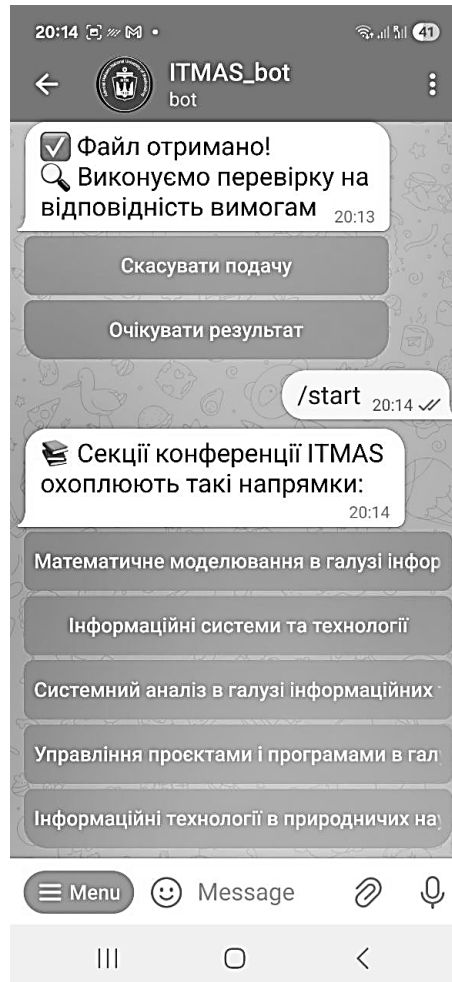


Рисунок 2.5 – Надання інформації стосовно секцій

Таким чином, щоб зменшити навантаження на користувача діалог з ботом буде вестись за трьома сценаріями: через пункти меню, через текстові повідомлення користувача та активні кнопки після повідомлення бота.

2.4 Технічний проєкт чат-боту для конференції ITMAS

Після завершення аналізу ескізного проєкту, було розроблено технічний проєкт. Для цього було побудовано дві основні моделі, які відображають різні аспекти функціонування чат-боту. Зокрема, створено статичну модель, що відображає основні класи, їхні властивості, зв'язки та ієрархію у вигляді діаграми класів. Також сформовано динамічну модель у вигляді діаграми послідовності, що демонструє послідовність взаємодії між об'єктами під час виконання варіанту використання "Завантажити тези".

Такий підхід дозволяє глибше зрозуміти як структуру, так і поведінку чат-боту в контексті конкретних сценаріїв.

2.4.1 Діаграма класів

Для більш детальної ілюстрації структури розроблюваного чат-боту для конференції ITMAS було розроблено діаграму класів.

Статична модель, зображена на рисунку 2.6, представлена діаграмою класів.

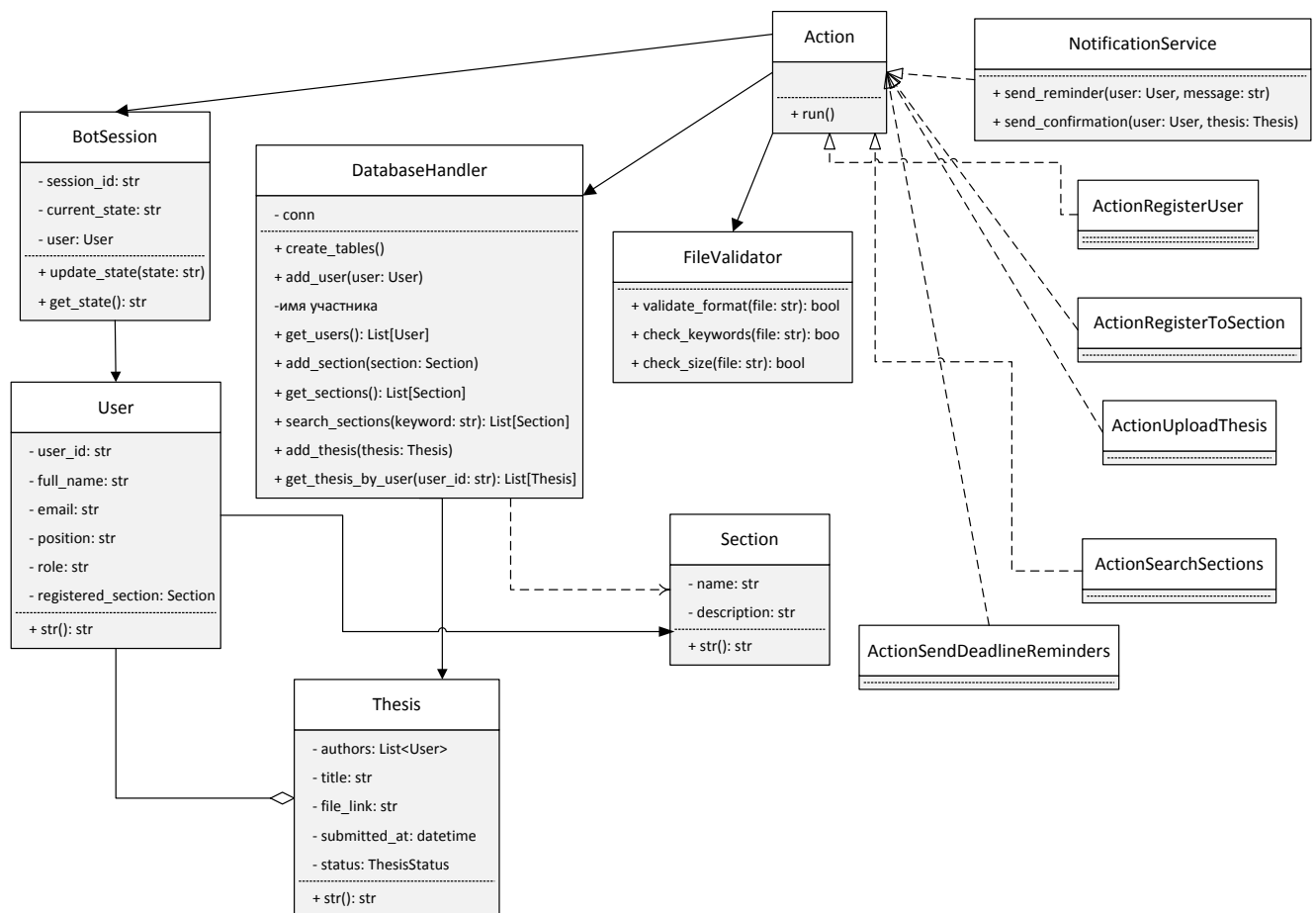


Рисунок 2.6 – Діаграма класів чат-боту для конференції ITMAS

Специфікації класів

Специфікації класів чат-боту для конференції ITMAS наведені нижче.

Назва класу: «User»

Відповідальність: Представляє користувача конференції, що може надсилати тези, реєструватися в секціях тощо.

Атрибути:

- full_name - повне ім'я (ПІБ)
- email - контактна електронна адреса
- position - наукове чи професійне звання/посада
- registered_section - секція, в якій користувач планує прийняти участь

Нижче представлені функції класу:

- __str__() - повертає текстове представлення користувача (наприклад, ім'я та email)

Назва класу: Section

Відповідальність: Представляє окремий тематичний напрямок роботи конференції

Атрибути:

- name - назва секції (наприклад, «Штучний інтелект»)
- description - текстовий опис тематики секції

Нижче представлені функції класу:

- __str__() - текстове представлення (наприклад, назва секції)

Назва класу: Thesis

Відповідальність: Представляє подані тези користувача для участі у конференції

Атрибути:

- authors: List[User] - список авторів
- file - шлях або посилання на файл із тезами (PDF)
- submitted_at - дата та час подання

Нижче представлені функції класу:

- __str__() - назва або перші рядки тез із іменами авторів

Назва класу: DatabaseHandler

Відповідальність: Забезпечує взаємодію з базою даних: створення, збереження та отримання записів

Атрибути:

- conn - з'єднання з базою даних (об'єкт SQLite/ORM)

Нижче представлені функції класу:

- create_tables() - створює всі необхідні таблиці в базі
- add_user() - додає користувача в таблицю
- add_users() - додає список користувачів
- get_sections() - повертає список усіх секцій
- search_sections() - фільтрує секції за ключовим словом
- get_thesis() - повертає інформацію про конкретну тезу
- get_thesis_by_user() - витягує всі тези, подані користувачем

Назва класу: BotSession

Відповідальність: Відстежує стан діалогу користувача з ботом і зберігає контекст

Атрибути:

- current_state - поточний крок/стан діалогу (наприклад, "очікує файл")
- user - прив'язаний користувач

Нижче представлені функції класу:

- update_state() - оновлює поточний стан
- get_state() - повертає поточний стан діалогу

Назва класу: FileValidator

Відповідальність: Перевіряє файл на відповідність технічним вимогам

Атрибути: (відсутні, всі методи статичні або сервісні)

Нижче представлені функції класу:

- validate_format() - перевіряє розширення (чи PDF)
- check_keywords() - шукає обов'язкові ключові слова в тезах
- check_size() - перевіряє розмір або обсяг документа

Назва класу: NotificationService

Відповідальність: Відправляє електронні листи, нагадування та підтвердження користувачам

Атрибути: відсутні

Нижче представлені функції класу:

- send_reminder() - надсилає нагадування (наприклад, про дедлайн)
- send_confirmation() - повідомляє про успішне подання тез

Назва інтерфейсу: Action

Відповідальність: Узагальнення дій на базі фреймворка Rasa

Атрибути: відсутні

Нижче представлені функції класу:

- run() - абстрактний метод, що виконує дію, коли бот його викликає

Назва класу: ActionRegisterUser

Відповідальність: Реєструє користувача за даними, отриманими з діалогу

Атрибути: db_handler - доступ до бази для збереження користувача

Нижче представлені функції класу:

- run() - метод, що отримує ім'я, email, посаду, роль з діалогу; створює об'єкт User; викликає db_handler.add_user(); надсилає повідомлення-підтвердження

Назва класу: ActionRegisterToSection

Відповідальність: Записує користувача до вибраної секції

Атрибути: db_handler- для оновлення поля користувача

Нижче представлені функції класу:

- run() - метод, що читає section_id із повідомлення; знаходить відповідну секцію через db_handler; оновлює запис користувача; підтверджує дію повідомленням

2.4.2 Діаграми послідовностей

На основі побудованої статичної моделі та діаграми варіантів використання розроблено динамічну модель чат-бота. Діаграма послідовності для варіанту використання «Завантажити тези» зображена на рисунку 2.7.

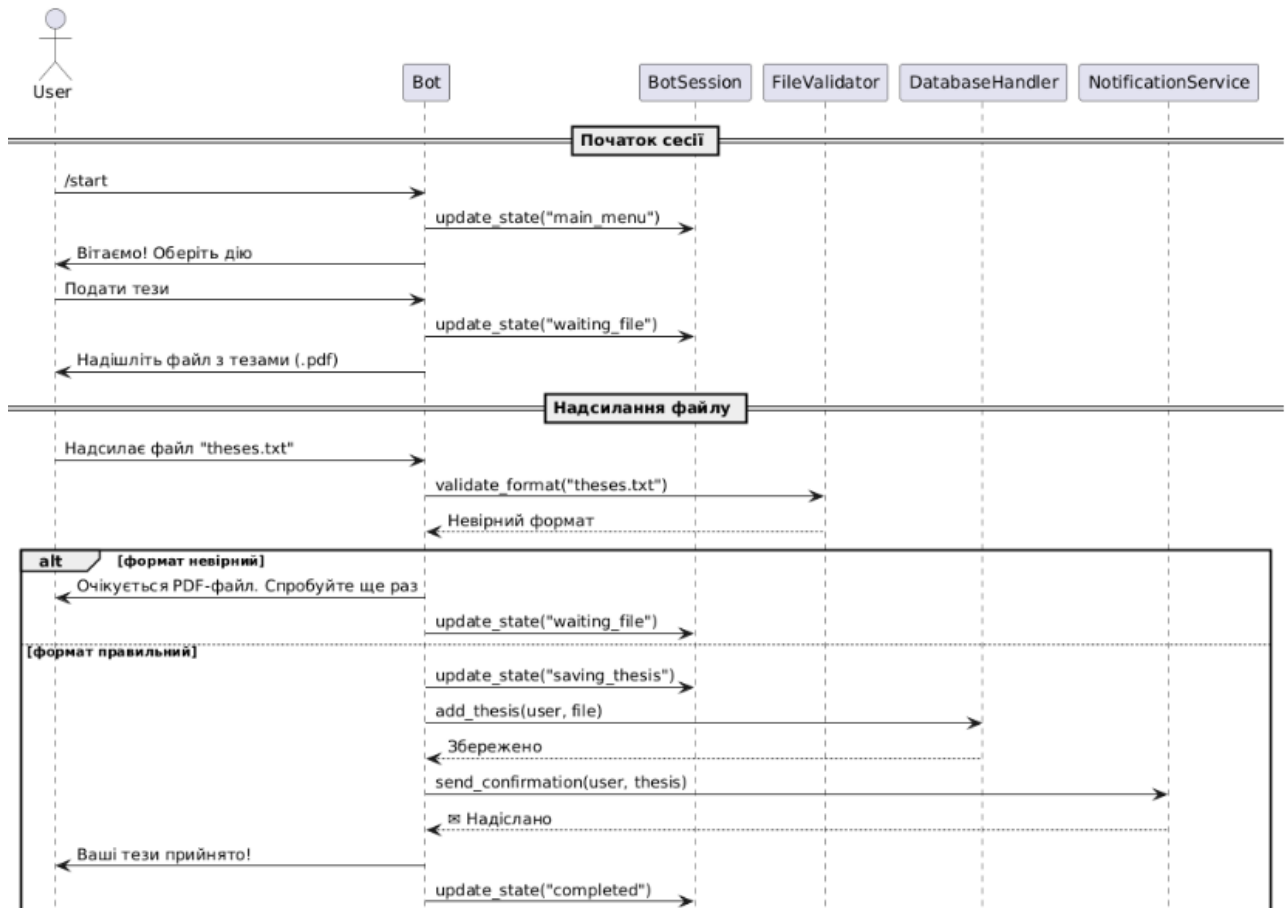


Рисунок 2.7 – Діаграма послідовності для варіанту використання
«Завантажити тези»

На діаграмі відображено процес взаємодії користувача з чат-ботом під час завантаження тез конференції. Користувач запускає бота в Telegram, обирає пункт "Завантажити тези", після чого бот чекає від користувача фізичний файл або посилання на файл. Далі виконується перевірка, чи є формат файлу валідним. Якщо так, тези зберігаються, і бот повідомляє про успішне завантаження. Якщо ні, користувач отримує повідомлення про відмову. Увесь процес демонструє, як бот керує логікою перевірки та обробки даних з використанням фреймворку Rasa.

Логічна модель даних

Після розробки діаграми класів, специфікацій та діаграми послідовності постає потреба в глибшому аналізі взаємозв'язків між сутностями. Щоб сформувати цілісне уявлення про структуру та взаємодію моделей даних чат-бота для конференції ITMAS, необхідно створити логічну модель даних із визначенням основних сутностей і їхніх атрибутів.

Логічна модель даних чат-боту для конференції ITMAS надана на рисунку 2.8.

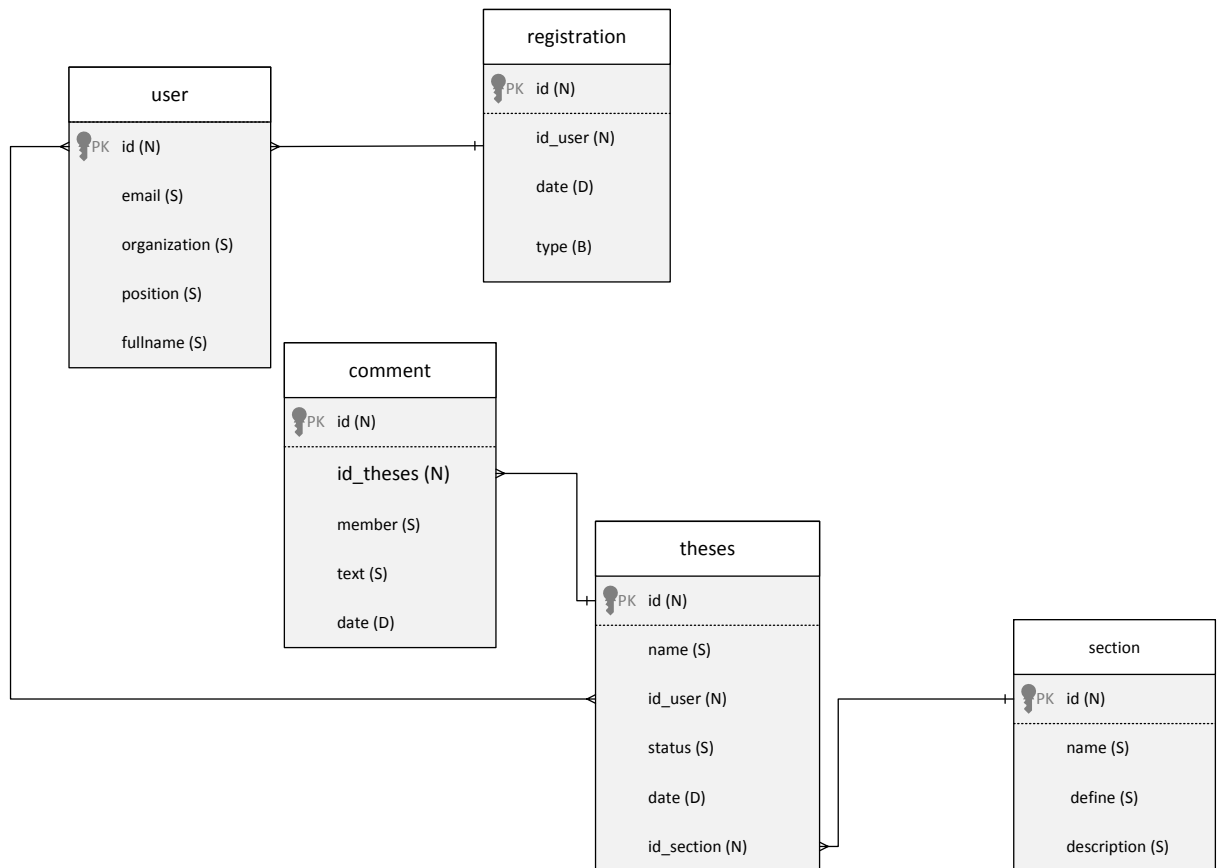


Рисунок 2.8 – Логічна модель даних чат-боту для конференції ITMAS

2.4.3 Робочий проєкт чат-боту для конференції ITMAS

Обґрунтування вибору мови програмування та системи керування базами даних

У сучасному цифровому світі чат-боти стали ефективним інструментом автоматизації взаємодії з користувачами, особливо під час проведення онлайн-заходів, таких як інтернет-конференції. Для розробки чат-бота

важливо обрати правильне поєднання технологій. Для проєкту було обрано мову програмування Python, фреймворк Rasa та базу даних SQLite.

Python є однією з найпопулярніших мов програмування у світі. Її переваги полягають у простоті синтаксису, широкій підтримці бібліотек для штучного інтелекту (AI) та обробки природної мови (NLP), а також активній спільноті розробників [9]. Завдяки зрозумілому синтаксису Python дозволяє швидко реалізувати функціональність без потреби у великому об'єму коду, а інтеграція з фреймворками Natural Language Processing, такими як бібліотеки spaCy, NLTK та TensorFlow, що створені спеціально під Python, є досить легкою [10].

Rasa - це open-source фреймворк, орієнтований на створення інтелектуальних діалогових агентів. Його функціональність дозволяє не лише розпізнавати наміри, але й керувати діалогами з урахуванням контексту.

Rasa NLU та Rasa Core, де перший компонент відповідає за аналіз запитів, а другий - за керування діалогом на основі сценаріїв [11]. Також фреймворк досить гнучкий при розгортанні, та на відміну від Dialogflow, Rasa може бути розгорнута локально, що забезпечує більший контроль над персональними даними користувачів [12]. Досить привабливою є можливість кастомізації, оскільки відкритий код дає змогу змінювати або розширювати поведінку бота відповідно до специфіки заходу. Можливість інтеграції з Telegram, Slack, вебсайтами — забезпечує універсальність у використанні ботів на різних платформах [13]. Rasa надає повноцінну платформу для створення розумного помічника, який може допомагати учасникам конференції з реєстрацією, завантаженням тез, отриманням інформації тощо.

Для зберігання даних про учасників конференції, запити, відповіді та історію взаємодій обрано SQLite, оскільки ця реляційна СУБД інтегрується напряму у застосунок без необхідності налаштування окремого серверу. Вбудовану базу даних SQLite зручно використовувати для проєктів середнього масштабу [14]. Сумісність з Python — стандартна бібліотека sqlite3 забезпечує легку інтеграцію та швидкий доступ до бази даних [15].

Для зберігання структурованих даних у рамках чат-бота інтернет-конференції доцільним є використання SQLite3 — реляційної вбудованої бази даних, яка не потребує окремого сервера. SQLite3 є однією з найпоширеніших у світі СУБД, особливо у випадках, коли потрібна легкість, мобільність і швидкість. Вся база даних зберігається в одному .sqlite або .db файлі на диску, що значно спрощує розгортання та резервне копіювання [16]. SQLite3 вже включена до стандартної бібліотеки Python (sqlite3), що дозволяє працювати з нею без встановлення додаткових бібліотек [17].

SQLite підтримує більшість стандартних SQL-операцій: SELECT, JOIN, GROUP BY, ORDER BY, транзакції тощо. Також забезпечує атомарність, узгодженість, ізолюваність і довговічність транзакцій, що важливо навіть для простих чат-ботів, які мають взаємодіяти з базою одночасно з декількох сесій [18].

Ідеально підходить для хостингу чат-бота на віртуальних серверах, Raspberry Pi або інших платформах з обмеженими обчислювальними можливостями, отже можна легко створити таблиці для збереження інформації про користувачів (ID, імена, email), реєстрації учасників.

Оскільки інтернет-конференції часто мають обмежений час проведення й не потребують складної багатокористувацької транзакційної обробки, SQLite є достатнім і раціональним вибором. Завдяки своїй легкості, LiteSQL споживає мінімум ресурсів, що важливо при хостингу чат-бота на хмарних або обмежених серверах.

Фізична модель даних

Фізична модель даних чат-боту для конференції ITMAS представлена в таблицях 2.2-2.6. Структура таблиць побудованих для чат-боту для конференції ITMAS містить в собі назви атрибутів, типи даних та ключі доступу до відповідних кортежів.

Таблиця 2.2 – Структура таблиці user

Назва	Тип даних	Ключ
id	integer	primary key
Email	varchar	
Organization	varchar	
Position	varchar	
Fullname	varchar	

Таблиця 2.3 – Структура таблиці registration

Назва	Тип даних	Ключ
Id	integer	primary key
id_user	integer	foreign key
Date	date	
Type	boolean	

Таблиця 2.4 – Структура таблиці comment

Назва	Тип даних	Ключ
id	integer	primary key
id_theses	integer	foreign key
member	varchar	
text	varchar	
date	date	

Таблиця 2.5 – Структура таблиці theses

Назва	Тип даних	Ключ
Id	integer	primary key
Name	varchar	
id_user	integer	foreign key
Status	varchar	
Date	date	
id_section	integer	foreign key

Таблиця 2.6 – Структура таблиці section

Назва	Тип даних	Ключ
id	integer	primary key
name	varchar	
define	varchar	
description	varchar	

Тестування чат-боту для конференції ITMAS

Метою тестування кожного зі сценаріїв використання чат-бота для конференції ITMAS є перевірка коректності функціонування та надійності роботи програмного забезпечення. Нижче подано опис процесу тестування для кожного окремого варіанту.

1) Реєстрація на конференцію:

Вхідні дані: Ім'я: "Іван Петренко", Email: "ivan@example.com", Посада: "доцент", Роль: "учасник".

Тест: Користувач проходить процес реєстрації через чат-бота, вводячи дані у відповідь на запити бота.

Результат: Успішна реєстрація користувача, дані зберігаються в базі, бот надсилає підтвердження з ID.

1.1) Реєстрація – альтернативний потік:

Вхідні дані: Email: "ivan@example.com" (вже зареєстрований), інші поля будь-які.

Результат: Бот повідомляє про помилку, що користувач вже існує. Пропонує авторизуватися.

2) Перегляд важливих дат:

Вхідні дані: Команда користувача: «Важливі дати».

Тест: Користувач обирає відповідний пункт у боті.

Результат: Бот надсилає перелік дат: дедлайн подання тез, дата конференції, реєстраційний термін.

2.1) Перегляд важливих дат – альтернативний потік:

Вхідні дані: Команда «Важливі дати», але календар ще не опублікований.

Результат: Бот повідомляє: «Наразі календар подій не опубліковано». Пропонує підписатися на оновлення.

3) Подання тез:

Вхідні дані: Користувач надіслав PDF-файл "theses.pdf" + ключові слова: "Інформаційні системи та технології".

Тест: Користувач вибирає «Подати тези», надсилає файл та опис.

Результат: Бот перевіряє реєстрацію, зберігає файл, надсилає підтвердження про прийом тез.

3.1) Подання тез – альтернативний потік:

Вхідні дані: Користувач не зареєстрований, надсилає файл.

Результат: Бот перенаправляє на сценарій «Реєстрація», після чого дозволяє подати тези.

4) Валідація поданих матеріалів:

Вхідні дані: Координатор відкриває заявку, натискає "Підтвердити".

Тест: Перевірка поданих тез у системі координатором.

Результат: Статус тез змінюється на "Прийнято". Користувачу надходить повідомлення: "Ваші тези підтверджено".

4.1) Валідація – альтернативний потік:

Вхідні дані: Файл містить джерела оформлені не за вимогами.

Результат: Координатор додає коментар, бот встановлює статус "Потребує доопрацювання" та надсилає користувачу запит на виправлення.

5) Отримати інформацію про секції конференції:

Вхідні дані: Користувач обирає пункт «Секції».

Тест: Бот надсилає перелік доступних напрямків роботи конференції із короткими описами.

Результат: Користувач отримує перелік секцій, може перейти до подання тез.

5.1) Отримання інформації – альтернативний потік:

Вхідні дані: Користувач обирає секцію «Інформаційні технології».

Результат: Бот надсилає розширену інформацію: тематика, ключові слова, рекомендації щодо тез, та пропонує подати матеріали.

Випробування програмного забезпечення

Програма та методика випробувань чат-боту для інтернет-конференції наведено в Додатку В. На основі інструкцій, що наведені в цьому додатку, проведемо випробування розробленого чат-боту.

Після запуску чат-бота користувачу відображається привітальне повідомлення та коротка інструкція щодо його використання. Приклад привітального повідомлення наведено на рисунку 2.9.

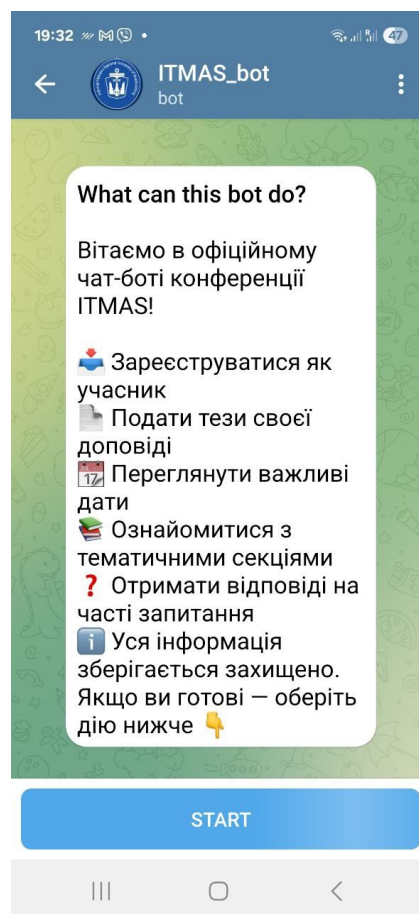


Рисунок 2.9 – Вітальне повідомлення

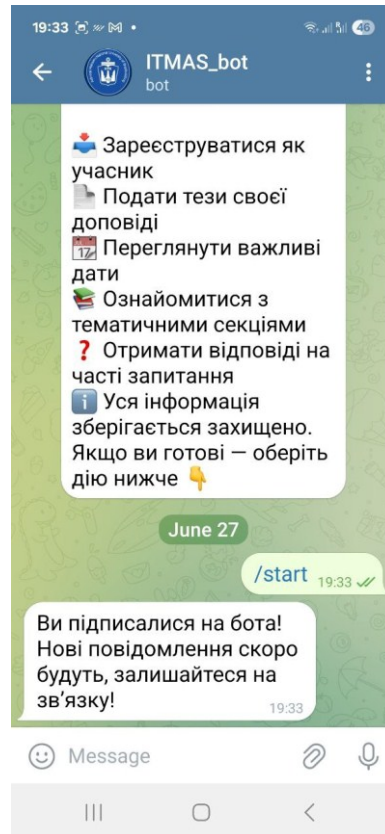


Рисунок 2.10 – Початок роботи з чат-ботом

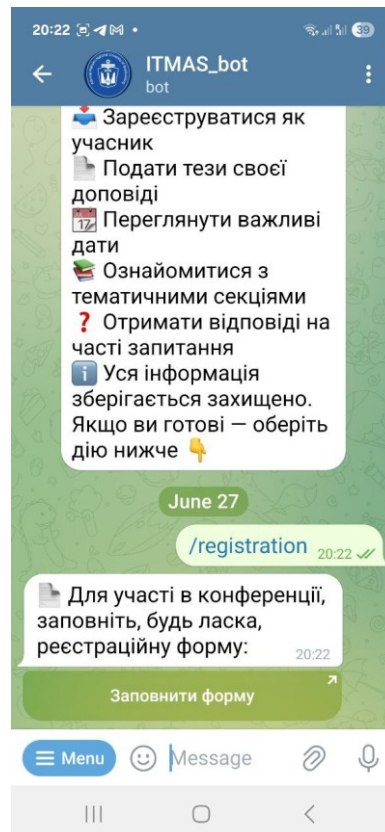


Рисунок 2.11 – Реєстрація на конференцію за допомогою чат-бота



Рисунок 2.12 – Відповідь чат-бота на заповнення форми

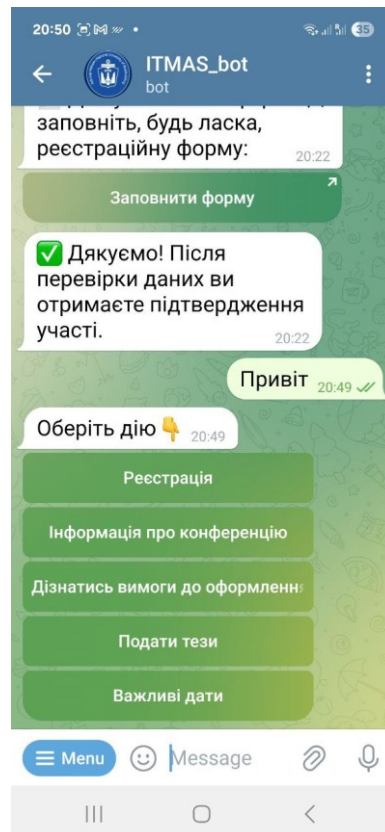


Рисунок 2.13 – Реакція бота при привітанні

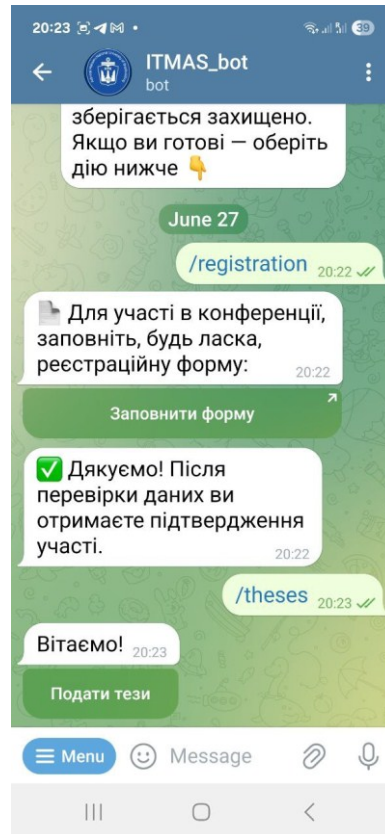


Рисунок 2.14 – Подання тез

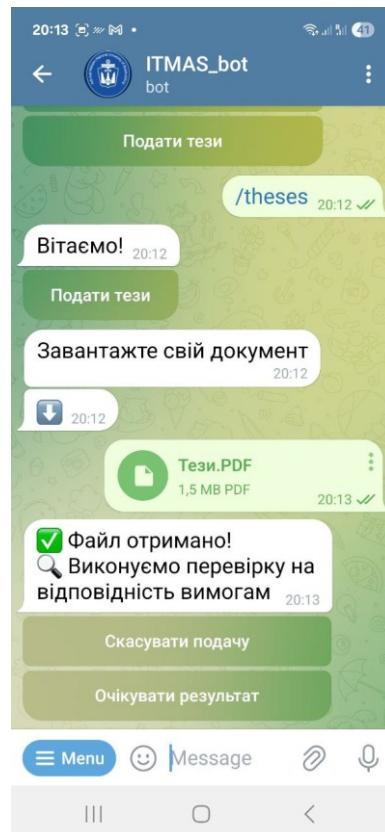


Рисунок 2.15 – Подання тез при позитивному сценарії

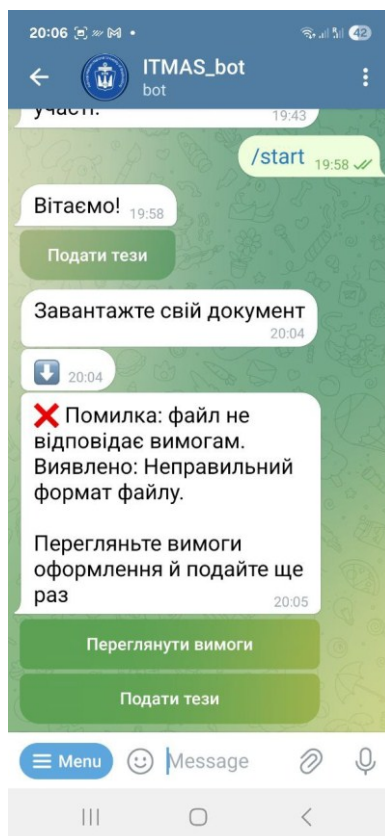


Рисунок 2.16 – Подання тез при негативному сценарії

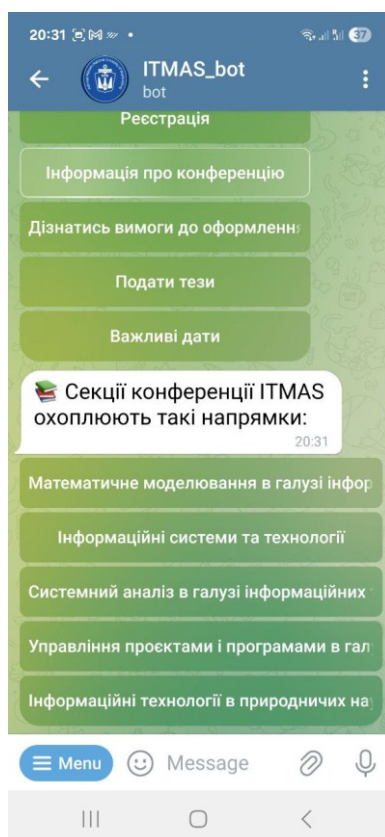


Рисунок 2.17 – Реакція бота при виборі пункту меню «Напрямки роботи конференції»



Рисунок 2.18 – Інформування учасника конференції про напрямок роботи

Таким чином було проведено випробування розробленого чат-боту за інструкціями, що наведено у Додатку Д – «Програма та методика випробувань чат-боту для конференції ITMAS».

3 РЕЗУЛЬТАТИ РОЗРОБКИ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS

В ході виконання кваліфікаційної роботи було розроблено чат-бот для конференції ITMAS. Для реалізації чат-боту було використано інтегроване середовище розробки (IDE) PyCharm 2025.1, мова програмування Python, компоненти фреймворку Rasa: обробка природньої мови за допомогою компоненту Rasa NLU, для управління діалогом (визначення наступної дії на основі контексту розмови) використовувався компонент Rasa Core, для інтеграції з базами даних Rasa SDK, для збереження даних користувачів використовувалась SQLite.

Зовнішній вигляд старту роботи чат-боту для конференції ITMAS проілюстровано на рисунку 3.1.

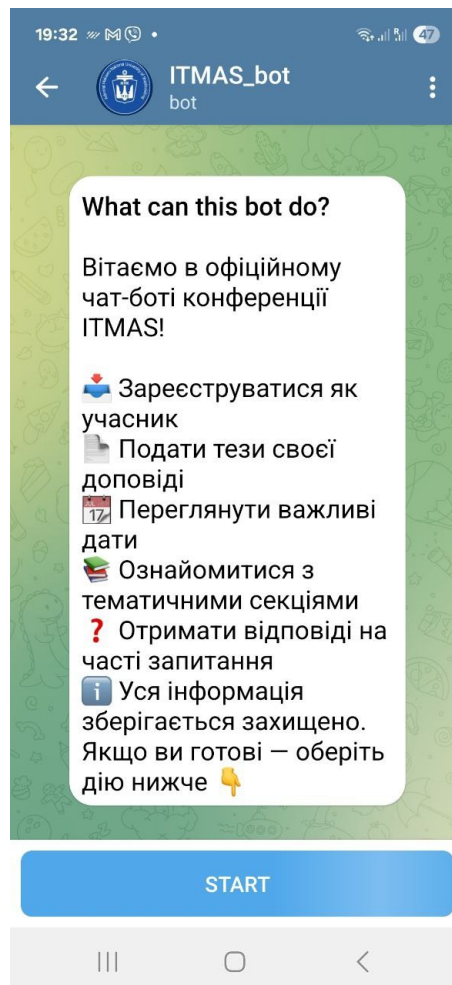


Рисунок 3.1 – Старт роботи з чат-ботом для конференції ITMAS

Технічне завдання на розробку чат-боту для конференції ITMAS викладене у Додатку А.

Частина коду розробленого проекту чат-боту для конференції ITMAS представлена у Додатку Б.

Інструкція користувача програмного забезпечення «ITMAS_bot» наведена у Додатку В.

У Додатку Г викладено опис чат-боту для конференції ITMAS.

У Додатку Д надана програма та методика випробувань, згідно якої було проведено відповідне випробування чат-боту для конференції ITMAS, на основі якого можна зробити висновок про повну працездатність роботи чат-боту для конференції ITMAS.

Для взаємодії з ботом для конференції ITMAS користувач може використовувати мобільний застосунок Telegram, або використовувати вебверсію додатку при роботі з ПК.

4 ОХОРОНА ПРАЦІ

Згідно з Законом України «Про охорону праці» [19]: охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності [19].

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам законодавства [19].

Головною метою охорони праці є створення на кожному робочому місці безпечних умов праці, безпечної експлуатації обладнання, зменшення або повна нейтралізація дії шкідливих і небезпечних виробничих факторів на організм людини і, як наслідок, зниження виробничого травматизму та професійних захворювань [20].

Розглянемо основні принципи державної політики в галузі охорони праці [20]:

- пріоритет життя і здоров'я працюючих, відповідальність роботодавця за створення безпечних і здорових умов праці [20];
- комплексне вирішення завдань охорони праці на основі створення національної програми поліпшення безпеки, гігієни праці та виробничого середовища [20];
- соціальний захист працівників, повне відшкодування шкоди потерпілим від нещасних випадків, профзахворювань [20];
- встановлення єдиних вимог з охорони праці для всіх підприємств та суб'єктів підприємницької діяльності незалежно від форм власності [20];
- організація навчання населення, професійної підготовки і підвищення кваліфікації працюючих з питань охорони праці [20];

Фінансування охорони праці здійснюється власником. Працівник не несе жодних витрат на заходи з охорони праці. Для фінансування заходів з охорони праці на підприємствах, у галузях і на державному рівні створюються фонди охорони праці [20].

Одним з найважливіших завдань охорони праці є забезпечення таких умов праці, які б вилучали можливість дії на працюючих різного роду небезпечних і шкідливих виробничих факторів. Згідно зі статтею 153 Кодексу законів про працю, власник підприємства зобов'язаний забезпечити належне технічне обладнання всіх робочих місць і створювати на них умови праці відповідно до нормативних актів з охорони праці [20].

Умови праці - це сукупність факторів виробничого середовища і виробничого процесу, які впливають на здоров'я і працездатність людини під час її професійної діяльності [20].

Працездатність - здатність людини до праці, яка визначається рівнем її фізичних і психофізіологічних можливостей, а також станом здоров'я та професійною підготовкою [20].

Безпека праці - стан умов праці, за якого відсутній виробничий травматизм [20]. Безпека умов праці - стан умов праці, за яких вплив на працюючого небезпечних і шкідливих виробничих факторів взагалі відсутній або дія шкідливих виробничих факторів не перевищує граничне допустимих рівнів [20].

Безпека виробничого процесу - здатність виробничого процесу відповідати вимогам безпеки праці під час його проведення в умовах, встановлених нормативно-технічною документацією [20].

Безпека виробничого устаткування - здатність устаткування зберігати безпечний стан при виконанні заданих функцій у певних умовах протягом встановленого часу [20].

Структура управління питаннями охорони праці на підприємстві

На підприємстві з кількістю працюючих 50 і більше осіб роботодавець створює службу охорони праці відповідно до типового положення, що затверджується центральним органом виконавчої влади, що забезпечує формування державної політики у сфері охорони праці [19].

На підприємстві з кількістю працюючих менше 20 осіб для виконання функцій служби охорони праці можуть залучатися сторонні спеціалісти на договірних засадах, які мають відповідну підготовку [19].

Служба охорони праці підпорядковується безпосередньо роботодавцю.

Керівники та спеціалісти служби охорони праці за своєю посадою і заробітною платою прирівнюються до керівників і спеціалістів основних виробничо-технічних служб [19].

Спеціалісти служби охорони праці у разі виявлення порушень охорони праці мають право [19]:

- видавати керівникам структурних підрозділів підприємства обов'язкові для виконання приписи щодо усунення наявних недоліків, одержувати від них необхідні відомості, документацію і пояснення з питань охорони праці [19];

- вимагати відсторонення від роботи осіб, які не пройшли передбачених законодавством медичного огляду, навчання, інструктажу, перевірки знань і не мають допуску до відповідних робіт або не виконують вимог нормативно-правових актів з охорони праці [19];

- зупиняти роботу виробництва, дільниці, машин, механізмів, устаткування та інших засобів виробництва у разі порушень, які створюють загрозу життю або здоров'ю працюючих [19];

- надсилати роботодавцю подання про притягнення до відповідальності працівників, які порушують вимоги щодо охорони праці [19].

Зобов'язання, права та відповідальність посадових осіб за виконання покладених на них функцій щодо питань охорони праці розглядаються в посадових інструкціях, форма яких розроблена Держнаглядохоронпраці. Згідно зі ст. 13 Закону України «Про охорону праці» роботодавець зобов'язаний створити в кожному структурному підрозділі та на робочому

місці умови праці відповідно до вимог нормативних актів, а також забезпечити додержання прав працівників, гарантованих законодавством про охорону праці [20]. З цією метою він забезпечує функціонування системи управління охороною праці, для чого [20]:

- створює відповідні служби та призначає посадових осіб, які забезпечують функціонування системи охорони праці [20];
- розробляє за участю сторін колективного договору й реалізує комплексні заходи для досягнення встановлених нормативів з охорони праці, впроваджує прогресивні досягнення з охорони праці [20];
- забезпечує усунення причин, що призводять до нещасних випадків, професійних захворювань, і виконання профілактичних заходів, визначених комісіями за підсумками розслідування цих причин [20];
- організовує проведення аудиту охорони праці, лабораторних досліджень умов праці, атестації робочих місць на відповідність нормативним актам про охорону праці, оцінку технічного стану виробничого обладнання та устаткування [20];
- розробляє і затверджує положення, інструкції, інші нормативні акти про охорону праці, що діють у межах підприємства [20];
- здійснює постійний контроль за додержанням працівниками технологічних процесів, правил та вимог щодо охорони праці [20];
- організовує пропаганду безпечних методів праці й співробітництво з працівниками у галузі охорони праці [20].

У випадку відсутності в нормативних актах про охорону праці вимог, які необхідно виконати для забезпечення безпечних і нешкідливих умов праці на певних роботах, власник зобов'язаний вжити заходів, що забезпечать безпеку працівників [20].

Вимоги до виробничих приміщень для експлуатації ПК

У цьому розділі наведена характеристика приміщень, де експлуатуються ПК [21]. Наведені їх параметри.

Об'ємно-планувальні рішення будівель та приміщень для роботи з ПК мають відповідати вимогам ДСанПіН 3.3.2.007-98 [21]. Розміщення робочих місць з ПК у підвальних приміщеннях, на цокольних поверхах заборонено [21].

Площа на одне робоче місце становить не менше ніж $6,0 \text{ м}^2$, а об'єм не менше ніж $20,0 \text{ м}^3$ [21]. Приміщення для роботи з ПК повинні мати природне та штучне освітлення відповідно до СНиП II-4-79 [21].

Природне освітлення має здійснюватись через світлові прорізи, орієнтовані переважно на північ чи північний схід, і забезпечувати коефіцієнт природної освітленості (КПО) не нижче, ніж 1,5% [21].

Виробничі приміщення повинні обладнуватись шафами для зберігання документів, магнітних дисків, полицями, стелажми, тумбами тощо, з урахуванням вимог до площі приміщень [21].

У приміщеннях з ПК слід щоденно робити вологе прибирання [21].

Приміщення з ПК мають бути оснащені аптечками першої медичної допомоги [21]. При приміщеннях з ПК мають бути обладнані побутові приміщення для відпочинку під час роботи, кімната психологічного розвантаження [21]. В кімнаті психологічного розвантаження слід передбачити встановлення пристроїв для приготування й роздачі тонізуючих напоїв, а також місця для занять фізичною культурою [21].

Гігієнічні вимоги до параметрів виробничого середовища включають вимоги до параметрів мікроклімату, освітлення, шуму і вібрації, рівнів електромагнітного та іонізуючого випромінювання [21].

Розробка заходів щодо зменшення дії шкідливих факторів при роботі за персональним комп'ютером

Для того, щоб зменшити вплив шкідливих факторів на моральний та фізичний стан працівника, що часто працює з персональним комп'ютером, необхідно дотримуватись деяких правил [21]:

- щогодини робити десятихвилинну перерву на відпочинок для очей від будь-яких подразників: моніторів, телевізорів, дрібних текстів тощо [21];
- ні в якому разі не використовувати для роботи телевізор замість монітора [21];
- за можливості одразу прибирати будь-які плями або осілий пил з екрана монітора [21];
- щогодини провітрювати приміщення в якому розташовуються комп'ютери [21];
- щодня виконувати вологе прибирання у робочому приміщенні, де встановлені комп'ютери [21];
- необхідно постійно слідкувати за поставою: ноги повинні твердо стояти на підлозі чи на спеціальній підставці; стегна мають бути розташовані під прямим кутом до тулуба, а гомілки – під прямим кутом до стегон; сидіти потрібно прямо або з невеликою амплітудою нахилу вперед; відстань від очей до екрана монітора має бути не менше 55-60 см; розташувати центр екрана таким чином, щоб він знаходився напроти рівня очей або трохи нижче [21];
- задля збереження очей необхідно моргати кожні 3-5 секунд та щодня виконувати спеціальну гімнастику для очей [21];
- при чітко вираженому відчутті втоми чи сонливості потрібно зробити невелику перерву та дати очам відпочити [21].

Дотримуючись правил, що зменшують вплив негативних чинників на самопочуття важливо також знати декілька рекомендованих умов для роботи за комп'ютером [21]:

- мати твердий стілець, край якого не повинен тиснути на судини під колінами [21];
- триматися від монітора на відстані 50-70 сантиметрів [21];
- робити перерву 15-20 хвилин під час роботи сидячи кожні 1-2 години [21];
- використовувати лише якісні та безпечні для очей джерела освітлення приміщень [21].

ВИСНОВКИ

В кваліфікаційній роботі було розв'язано практичну задачу інженерії програмного забезпечення, що характеризується комплексністю та невизначеністю умов, а саме розроблено чат-бот для конференції ITMAS.

В процесі розв'язання вказаної задачі було виконано аналіз практичної задачі інженерії програмного забезпечення з розробки чат-боту для конференції ITMAS. Було проаналізовано існуючі аналоги програмного забезпечення та обґрунтовано доцільність розробки власного програмного забезпечення. Виконано постановку задачі на розробку чат-боту для конференції ITMAS. Виконано аналіз вимог до чат-боту для конференції ITMAS. Розроблено архітектуру та проєкт чат-боту для конференції ITMAS.

Виконано реалізацію, тестування та випробування чат-боту для конференції ITMAS.

Також було розглянуто питання з охорони праці.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Інформаційні технології: моделі, алгоритми, системи – 2025. VI Міжнародна науково-практична інтернет конференція. URL: <https://itconf.nuos.edu.ua/2025/> (дата звернення: 21.05.2025).
2. 2025 Gartner® Voice of the Customer for Digital Commerce. URL: <https://commercetools.com/resources/analyst-report/gartner-voice-of-the-customer-for-digital-commerce-ar> (дата звернення: 24.04.2025)
3. Як чат-боти на базі SendPulse допомагають під час війни. URL: <https://sendpulse.ua/blog/how-chatbots-help-in-times-of-war> (дата звернення: 24.04.2025)
4. Найкращі чат-боти для сайту 2025. URL: <https://digital-expert.online/ua/chat-bot-builder> (дата звернення: 25.05.2025)
5. Why Your Business Needs a Custom AI Chatbot in 2025. URL: <https://kanhasoft.com/blog/why-your-business-needs-a-custom-ai-chatbot-in-2025/> (дата звернення: 15.05.2025)
6. Telegram-бот Bruno для конференцій. URL: <https://botsband.online/chatbot-events> (дата звернення: 15.05.2025)
7. SMART Chatbot. Дозвольте боту допомогти вашому бізнесу. URL: https://chatbot.smart-it.com/ua/?utm_source=chatgpt.com (дата звернення: 15.05.2025)
8. Lampa. URL: <https://lampalampa.net/ua/chat-boti/> (дата звернення: 16.05.2025)
9. Python Software Foundation. Welcome to Python.org. URL: <https://www.python.org/> (дата звернення: 27.05.2025)
10. SpaCy Documentation. Industrial-strength Natural Language Processing. URL: <https://spacy.io/> (дата звернення: 27.05.2025)
11. Rasa. *Open source conversational AI*. URL: <https://rasa.com/> (дата звернення: 27.05.2025)

- 12.Rasa Docs. *Rasa Open Source Documentation*. URL: <https://rasa.com/docs/rasa/> (дата звернення: 27.05.2025)
- 13.Rasa Integrations. *Messaging and Channel Integrations*. URL: <https://rasa.com/docs/rasa/connectors/> (дата звернення: 27.05.2025)
- 14.SQLite. *About SQLite*. URL: <https://sqlite.org/about.html> (дата звернення: 27.05.2025)
- 15.Python Docs. *sqlite3 — DB-API 2.0 interface for SQLite databases*. URL: <https://docs.python.org/3/library/sqlite3.html> (дата звернення: 27.05.2025)
- 16.Oetiker T. (2020). *LiteSQL ORM for Embedded and Lightweight Database Solutions*. URL: <https://sourceforge.net/projects/litesql/> (дата звернення: 27.05.2025)
- 17.Hipp D. R. *The Architecture of SQLite*. URL: <https://sqlite.org/arch.html> (дата звернення: 27.05.2025)
- 18.Gagnon R. *Using SQLite in your Applications*. Journal of Software Development. 2020. 14(2), P 44-51.
- 19.Про охорону праці: Закон України від 21.11.2002 р. №229-IV. Дата оновлення: 01.10.2023. URL: <https://zakon.rada.gov.ua/laws/main/2694-12#Text>. (дата звернення: 18.03.2025).
20. Винокурова Л. Е. Основи охорони праці: Навчальний посібник для професійно-технічних навчальних закладів/Л. Е. Винокурова, М. В. Васильчук, М. В. Гаман. - К.: Факт. 2005.
21. Желібо Є. П. Основи ОХОРОНИ ПРАЦІ./ М. П. Гандзюк, Є. П. Желібо, М. О. Халімовський.- Київ «Каравела». 2023.- 384 с.

ДОДАТОК А – ТЕХНІЧНЕ ЗАВДАННЯ НА РОЗРОБКУ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS

1 Вступ

1.1 Назва програмного забезпечення

Назва програмного забезпечення, що розробляється – «Телеграм-бот для конференції ITMAS»

1.2 Сфера застосування

Сферою застосування є месенджер Telegram.

2 Підстава для розробки програмного забезпечення

Розробка програмного забезпечення виконується на підставі завдання на кваліфікаційну роботу, виданого кафедрою ПЗАС НУК імені адмірала Макарова.

3 Призначення розробки

3.1 Функціональне призначення

Функціональним призначенням чат-боту для конференції ITMAS є:

- збір інформації стосовно особистої інформації учасників конференції;
- надання інформації стосовно важливих дат проведення конференції;
- збір та надання інформації стосовно матеріалів конференції;
- надання інформації стосовно оформлення тез конференції.

3.2 Експлуатаційне призначення

Експлуатаційним призначенням чат-боту для конференції є полегшення інформування учасників конференції ITMAS стосовно оформлення матеріалів конференції, інформування про важливі дати та покращення зручності надання інформації користувачу.

4 Технічні вимоги до програми

4.1 Вимоги до функціональних характеристик

4.1.1 Вимоги до переліку виконуваних функцій

- інтерфейс чат-боту повинен бути інтегровано в Telegram;
- чат-бот повинен виконувати сценарій привітання;
- чат-бот повинен мати навігаційне меню, що дозволяє поділити його роботу за певними сценаріями відповідно до обраних користувачем пунктів;
- чат-бот повинен розсилати інформаційний лист про проведення інтернет-конференції за базою даних учасників конференцій минулих років;
- чат-бот повинен вміти відповідати на питання стосовно тематики проведення конференції, стосовно секцій конференції, мати відповіді та приклади матеріалів стосовно секцій конференції;
- чат-бот повинен вміти відповідати на питання стосовно важливих дат конференції ITMAS;
- чат-бот повинен вміти реєструвати учасників конференції;
- чат-бот повинен надавати консультацію та вміти завантажувати матеріали до депозитарію конференції, повинна бути можливість відмінити завантаження, повторити завантаження;
- чат-бот повинен вміти надавати відповіді учасникам конференції стосовно статусу їх матеріалів;
- чат-бот повинен проводити статистику користувачів та їхніх матеріалів;

- чат-бот повинен вміти давати відповіді стосовно оформлення матеріалів (наприклад, оформлення малюнків, таблиць та інше);
- чат-бот повинен обробляти відгуки стосовно формату проведення конференції.

4.1.2 Вимоги до організації вхідних та вихідних даних

Дані зберігаються у файлі формату json.

Вимоги до організації вхідних та вихідних даних.

Для організації вхідних та вихідних даних чат-бот для конференції ITMAS повинен мати такі функції:

1) Реєстрація учасників конференції:

Вхідні дані: введені дані користувача (адреса електронної пошти, пароль).

Вихідні дані: успішна реєстрація учасника конференції в системі.

2) Авторизація учасників конференції:

Вхідні дані: введені облікові дані учасника конференції (адреса електронної пошти, пароль).

Вихідні дані: успішна авторизація учасника конференції в системі.

3) Запит на перегляд інформації про важливі дати події:

Вхідні дані: назва події.

Вихідні дані: успішний перегляд учасником конференції інформації про подію.

4) Запит на отримання інформації за назвою секції:

Вхідні дані: ідентифікатор назви секції.

Вихідні дані: успішне виконання пошуку учасником конференції за характеристиками.

5) Запит на отримання інформації за датою:

Вхідні дані: ідентифікатор події за датою.

Вихідні дані: успішне виконання пошуку учасником конференції за датою.

6) Завантаження тез:

Вхідні дані: файл тез.

Вихідні дані: ідентифікатор тез.

7) Видалення тез:

Вхідні дані: ідентифікатор тез.

Вихідні дані: успішне видалення тез.

8) Отримання додаткової інформації від користувачів, тези яких будуть публікуватись:

Вхідні дані: ідентифікатор користувача, повне ім'я, номер телефону та адреса користувача, місце роботи, посада, вчене звання, назва публікації.

Вихідні дані: успішне додавання інформації до системи.

9) Вихід з облікового запису:

Вхідні дані: ідентифікатор учасника конференції.

Вихідні дані: успішний вихід з облікового запису.

4.2 Вимоги до надійності

Програмне забезпечення повинно працювати безперебійно за умови безперервної роботи сервера, стабільного функціонування месенджера Telegram і надійного підключення до Інтернету. Дані регулярно оновлюються через певний інтервал шляхом перезапису відповідного файлу, в якому вони зберігаються до наступного оновлення. Такий підхід зменшує ризик втрати інформації.

У разі введення некоректної команди користувачем, бот повторно поставить запитання. Для полегшення взаємодії та зменшення навантаження на користувача передбачено сценарій роботи з використанням інтерактивного меню.

4.3 Умови експлуатації

4.3.1 Кліматичні умови експлуатації

Кліматичні умови експлуатації програмного забезпечення мають відповідати умовам експлуатації технічних засобів, що використовуються для роботи з програмним забезпеченням.

4.3.2 Вимоги до чисельності та кваліфікації користувачів

Користувачі програмного забезпечення повинні мати базові навички вміння роботи за комп'ютером або смартфоном, мінімальні навички роботи з месенджером Telegram.

4.4 Вимоги до складу і параметрів технічних засобів

4.4.1 Вимоги до користувацького обладнання:

Комп'ютер повинен мати наступну мінімальну комплектацію:

- CPU: Intel Pentium 4 або новіше із підтримкою SSE3;
- RAM: 8 GB;
- HDD: 1 GB вільного місця

Мінімальна комплектація смартфона:

- Версія Android 14 / iOS 16
- ОЗУ 8 GB
- 1 GB вільної пам'яті

4.4.2 Вимоги до серверного обладнання:

Рекомендованими вимогами до серверного обладнання є:

- серверний процесор Intel Xeon або AMD EPYC;
- ОЗУ 16 GB;

– жорсткий диск: 1 ТВ.

4.5 Вимоги до інформаційної та програмної сумісності

Клієнт повинен мати в наявності наступні програмні засоби:

використовувати операційну систему Windows 10 або іншу операційну систему. Також рекомендовано встановити десктопну версію месенджеру Telegram.

4.6 Вимоги до маркування та упаковки

Вимоги до маркування та упаковки відсутні.

4.7 Вимоги до транспортування та зберігання

Вимоги до транспортування та зберігання відсутні.

4.8 Спеціальні вимоги

Спеціальні вимоги відсутні.

5 Вимоги до програмної документації

Склад програмної документації повинен включати в себе:

- технічне завдання;
- текст програми;
- керівництво користувача;
- опис програми;
- програму і методику випробувань.

6 Техніко-економічні показники

У кваліфікаційній роботі техніко-економічні показники не розраховуються.

7 Стадії та етапи розробки

Етапи розробки програмного забезпечення наведено в завданні до кваліфікаційної роботи.

8 Порядок контролю і прийомки

Контроль здійснюється керівником кваліфікаційної роботи на кожному етапі розробки програмного забезпечення. Приймання здійснюється за програмою та методикою випробувань. Випробування проводиться в зазначений термін.

ДОДАТОК Б – ТЕКСТ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS

Текст actions.py

```
import json
from typing import Any, Dict, List, Text
from rasa_sdk import Action, Tracker
from rasa_sdk.executor import CollectingDispatcher
from classes import User, Thesis, Section

def load_users_from_json(filepath: str) -> Dict[str, User]:
    with open(filepath, 'r', encoding='utf-8') as f:
        data = json.load(f)
    return {user_id: User(user_id, info["full_name"], info["role"])
            for user_id, info in data.items()}

def load_sections_from_json(filepath: str) -> Dict[str, Section]:
    with open(filepath, 'r', encoding='utf-8') as f:
        data = json.load(f)
    return {section_id: Section(section_id, name) for section_id, name
            in data.items()}

def load_dates_from_json(filepath: str) -> Dict[str, str]:
    with open(filepath, 'r', encoding='utf-8') as f:
        data = json.load(f)
    return data

USERS = load_users_from_json("data/authors.json")
SECTIONS = load_sections_from_json("data/sections.json")
DATES = load_dates_from_json("data/dates.json")
THESES = []

class ActionShowSectionsWithButtons(Action):
    def name(self) -> Text:
        return "action_show_sections_buttons"

    def run(self, dispatcher: CollectingDispatcher,
            tracker: Tracker,
            domain: Dict[Text, Any]) -> List[Dict[Text, Any]]:
        buttons = []
        for section_id, section in SECTIONS.items():
            buttons.append({
                "title": section.name,
                "payload":
f'/register_to_section>{"section_id":"{section_id}"}}'
```

```

    })
    dispatcher.utter_message(
        text="Оберіть секцію для реєстрації:",
        buttons=buttons
    )
    return []

class ActionRegisterToSection(Action):
    def name(self) -> Text:
        return "action_register_to_section"

    def run(self, dispatcher: CollectingDispatcher,
            tracker: Tracker,
            domain: Dict[Text, Any]) -> List[Dict[Text, Any]]:
        user_id = tracker.sender_id
        section_id = tracker.get_slot("section_id")

        if user_id not in USERS:
            dispatcher.utter_message(text="Вас не знайдено в базі авторів.")
            return []

        if section_id not in SECTIONS:
            available = ", ".join([f"{k} ({v.name})" for k, v in SECTIONS.items()])
            dispatcher.utter_message(text=f"Такої секції не існує. Доступні: {available}")
            return []

        user = USERS[user_id]
        section = SECTIONS[section_id]

        if user.registered_section:
            dispatcher.utter_message(text=f"Ви вже зареєстровані у секції: {user.registered_section.name}.")
            return []

        user.registered_section = section
        dispatcher.utter_message(text=f"Ви успішно зареєстровані в секції: {section.name}.")
        return []

class ActionUploadThesis(Action):
    def name(self) -> Text:
        return "action_upload_thesis"

    def run(self, dispatcher: CollectingDispatcher,
            tracker: Tracker,
            domain: Dict[Text, Any]) -> List[Dict[Text, Any]]:
        user_id = tracker.sender_id
        thesis_title = tracker.get_slot("thesis_title")

```

```

file_link = tracker.get_slot("file_link")

if user_id not in USERS:
    dispatcher.utter_message(text=" Вас не знайдено в базі
авторів.")
    return []

user = USERS[user_id]

if user.role.lower() != "спікер":
    dispatcher.utter_message(text="Тези можуть надсилати лише
спікери.")
    return []

thesis = Thesis(user, thesis_title, file_link)
THESES.append(thesis)

dispatcher.utter_message(text=f" {thesis}")
return []

```

```

class ActionShowImportantDates(Action):

```

```

    def name(self) -> Text:
        return "action_show_important_dates"

    def run(self, dispatcher: CollectingDispatcher,
            tracker: Tracker,
            domain: Dict[Text, Any]) -> List[Dict[Text, Any]]:
        if not DATES:
            dispatcher.utter_message(text="Інформація про важливі дати
наразі відсутня.")
            return []

        sorted_dates = sorted(DATES.items())
        messages = ["Важливі дати конференції:"]
        for date_str, description in sorted_dates:
            messages.append(f"📅 {date_str}: {description}")

        dispatcher.utter_message(text="\n".join(messages))
        return []

```

Текст classes.py

```

from datetime import datetime

class User:
    def __init__(self, user_id, full_name, role="учасник"):
        self.user_id = user_id
        self.full_name = full_name
        self.role = role
        self.registered_section = None

class Section:
    def __init__(self, section_id, name):
        self.section_id = section_id
        self.name = name

class Thesis:
    def __init__(self, author: User, title: str, file_link: str):
        self.author = author
        self.title = title
        self.file_link = file_link
        self.submitted_at = datetime.now()

    def __str__(self):
        return f"Тези «{self.title}» надіслані автором {self.author.full_name}."

```

Текст conference_queries.sql

```

SELECT * FROM users WHERE registered_section = 'section_001';

```

```

SELECT u.user_id, u.full_name, COUNT(DISTINCT s.section_id) AS
section_count

```

```

FROM users u
JOIN thesis_authors ta ON u.user_id = ta.user_id
JOIN theses t ON ta.thesis_id = t.thesis_id
JOIN users u2 ON u2.user_id = ta.user_id AND u2.registered_section IS
NOT NULL

```

```

JOIN sections s ON s.section_id = u2.registered_section
GROUP BY u.user_id
HAVING section_count >= 2;

```

```

SELECT DISTINCT s.section_id, s.name
FROM sections s
JOIN users u ON s.section_id = u.registered_section;

```

```
SELECT * FROM theses WHERE submitted_at > '2025-06-01T00:00:00';
```

```
SELECT registered_section, COUNT(*) AS user_count
FROM users
GROUP BY registered_section;
```

```
SELECT u.user_id, u.full_name
FROM users u
JOIN thesis_authors ta ON u.user_id = ta.user_id
WHERE ta.thesis_id = 1;
```

```
SELECT t.*
FROM theses t
JOIN thesis_authors ta ON t.thesis_id = ta.thesis_id
JOIN users u ON u.user_id = ta.user_id
WHERE u.email = 'name@example.com';
```

```
SELECT *
FROM users u
WHERE role = 'спікер'
AND NOT EXISTS (
    SELECT 1 FROM thesis_authors ta WHERE ta.user_id = u.user_id
);
```

```
SELECT * FROM sections
WHERE name LIKE '%технології%' OR description LIKE '%інформаційні
технології %';
```

```
SELECT u.user_id, u.full_name, COUNT(ta.thesis_id) AS thesis_count
FROM users u
JOIN thesis_authors ta ON u.user_id = ta.user_id
GROUP BY u.user_id
HAVING thesis_count > 1;
```

```
SELECT * FROM users WHERE registered_section IS NULL;
```

```
SELECT s.section_id, s.name, COUNT(t.thesis_id) AS thesis_count
FROM sections s
JOIN users u ON s.section_id = u.registered_section
JOIN thesis_authors ta ON u.user_id = ta.user_id
JOIN theses t ON ta.thesis_id = t.thesis_id
GROUP BY s.section_id;
```

ДОДАТОК В – ІНСТРУКЦІЯ КОРИСТУВАЧА ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS

Після запуску чат-бота користувачу відображається привітальне повідомлення та коротка інструкція щодо його використання. Приклад привітального повідомлення наведено на рисунку В.1.

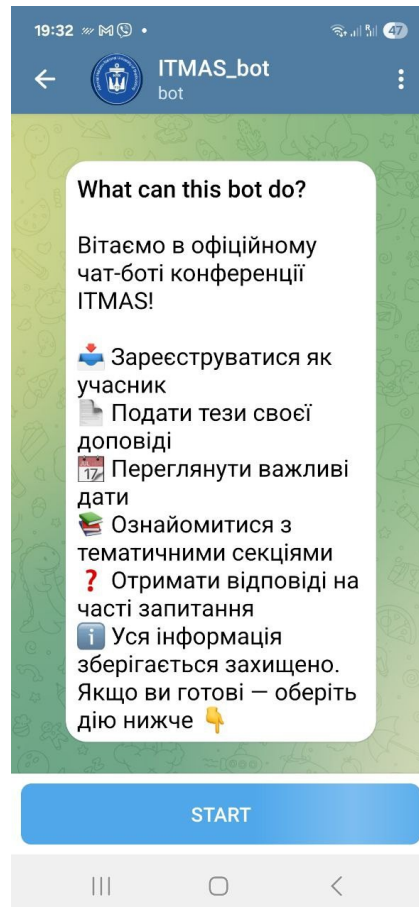


Рисунок Д.1 – Старт роботи з чат-ботом

Для того щоб почати взаємодію з ботом користувач спочатку повинен натиснути кнопку «Start», після цього користувач отримує доступ до основних функцій помічника «ITMAS_bot» конференції ITMAS, дивись рис. Д.2. Після запуску чат-бот надсилає структуроване привітання з описом доступних дій:

- Зареєструватися як учасник
- Подати тези своєї доповіді
- Переглянути важливі дати
- Ознайомитися з тематичними секціями

- Отримати відповіді на часті запитання

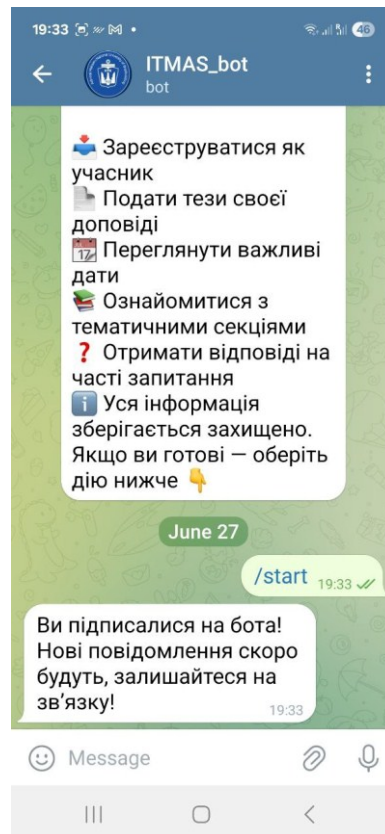


Рисунок Д.2 – Початок роботи з чат-ботом

Щоб зареєструватися на конференцію ITMAS через Telegram-бота, користувач повинен спершу знайти бот у месенджері за назвою ITMAS_bot і запустити його командою /start або кнопкою «Start». Після привітання бота користувач натискає кнопку «Заповнити форму», розташовану внизу екрана, або вводить відповідну команду, дивись рис. Д.3. Бот одразу надає активне посилання на реєстраційну форму, розміщену на сайті конференції.

Користувач переходить за посиланням, де відкривається форма, яку необхідно заповнити, вказавши свої дані: прізвище, ім'я, по батькові, адресу електронної пошти, навчальний заклад або організацію, посаду, науковий ступінь та роль на конференції. Після заповнення всіх обов'язкових полів користувач натискає кнопку «Зареєструватися». У разі успішного подання форми, бот надсилає підтвердження — як на email, так і, в Telegram-боті у вигляді повідомлення з подякою та нагадуванням, що підтвердження також буде надіслано на електронну пошту.

Користувач також може повернутися до головного меню, що знаходиться внизу екрану та продовжити взаємодію з ботом, подати тези або ознайомитися з тематичними напрямками конференції.



Рисунок Д.3 – Реєстрація на конференцію за допомогою чат-бота

Після заповнення й надсилання форми користувач отримує повідомлення в боті з підтвердженням того, що його дані надіслані організаторам для перевірки. Зокрема, бот зазначає, що після перевірки даних буде надіслано підтвердження участі, дивись рис. Д.4.

Це означає, що реєстрація технічно відбулася успішно, але статус "учасника" активується лише після автоматичної перевірки валідності електронної пошти, та перевірки даних координаторами конференції. Таким чином, користувач може очікувати підтвердження на електронну пошту або подальші інструкції в Telegram і може переходити до наступних кроків.



Рисунок Д.4 – Відповідь чат-бота на заповнення форми

Якщо користувач, не вказує свої наміри, та самостійно не вибирає дію через меню, бот знову пропонує меню з кількома кнопками для подальших дій. Серед них — «Реєстрація», «Інформація про конференцію», «Дізнатись вимоги до оформлення», «Подати тези» та «Важливі дати» (рисунок Д.5).

Це спонукає учасника перейти до наступних кроків участі — наприклад, подання тез або ознайомлення з програмою заходу. Бот підтримує логіку супроводу та зберігає зручну структуру взаємодії з учасником конференції.

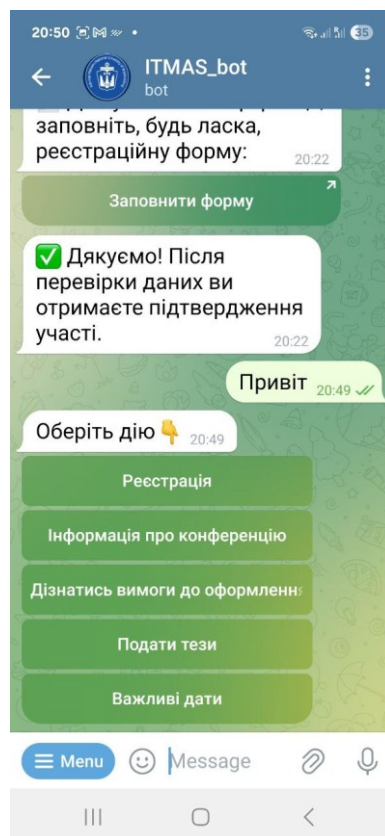


Рисунок Д.5 – Реакція бота при привітанні

Користувач, пройшовши попередню реєстрацію, повертається до головного меню, де обирає пункт «Подати тези». У відповідь бот надсилає супровідний текст, який підтверджує готовність прийняти матеріали, та активне посилання на інтерфейс завантаження (рисунок Д.6). Уся логіка побудована так, аби зробити процес подання інтуїтивним і зручним, після підтвердження реєстрації користувач може подавати наукові матеріали, не залишаючи середовища Telegram. Це значно спрощує навігацію й забезпечує постійну підтримку учасника протягом усього шляху участі в конференції ITMAS.

Користувач, який вже зареєстрований, може надіслати документ у форматі PDF, це визначено умовами оформлення матеріалів. Як тільки файл отримано, бот підтверджує завантаження та повідомляє, що розпочато перевірку відповідності технічним вимогам. У цей момент користувач має можливість скасувати подання або очікувати результат перевірки (рисунок Д.7). Цей процес дозволяє здійснити завантаження наукових

матеріалів безпосередньо в месенджері, що значно спрощує взаємодію й не потребує переходу на додаткові платформи.

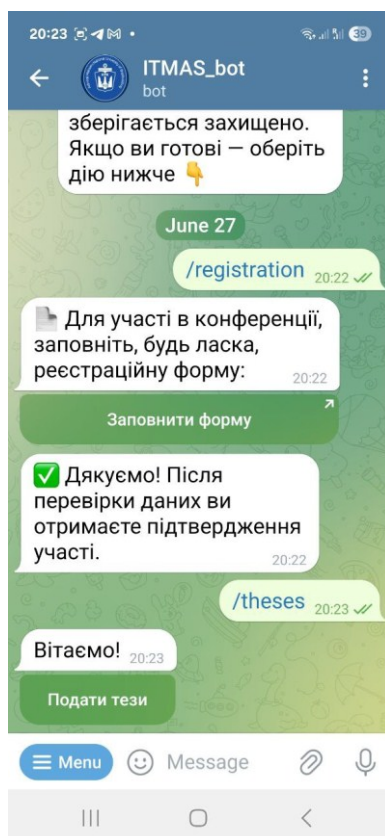


Рисунок Д.6 – Подання тез

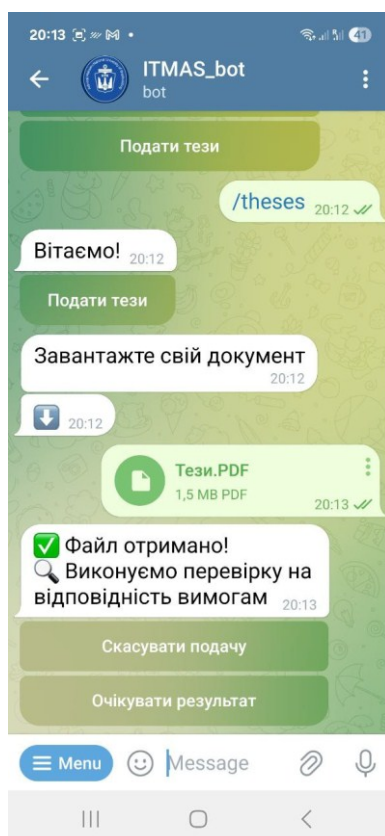


Рисунок Д.7 – Подання тез при позитивному сценарії

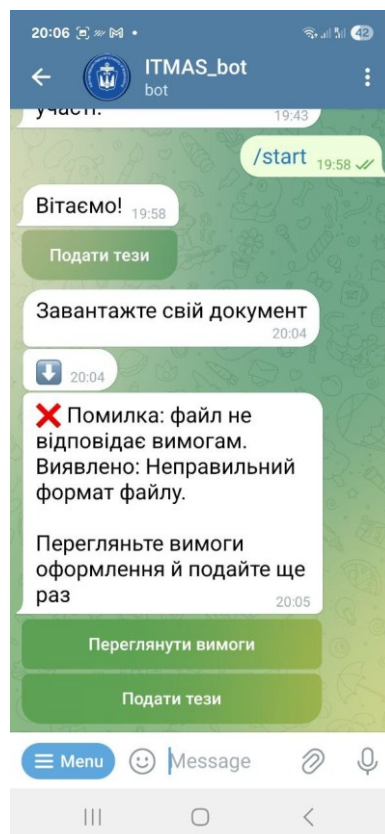


Рисунок Д.8 – Подання тез при негативному сценарії

Після вибору пункту «Секції конференції», бот надсилає повідомлення з переліком напрямків, які охоплює конференція ITMAS. Кожна секція подана у вигляді окремої кнопки для подальшого вибору. Серед напрямків користувач може обрати, зокрема: «Математичне моделювання в галузі IT», «Інформаційні системи та технології», «Системний аналіз в інформаційній сфері», «Управління проектами і програмами», «IT у природничих науках» (рисунок Д. 9).

Якщо учасник бажає отримати інформацію про конкретний напрямок роботи конференції ITMAS (рисунок Д.10), йому потрібно натиснути лише на активні кнопки під повідомленням боту, або написати текстове повідомлення, за яким бот зможе розпізнати намір користувача.

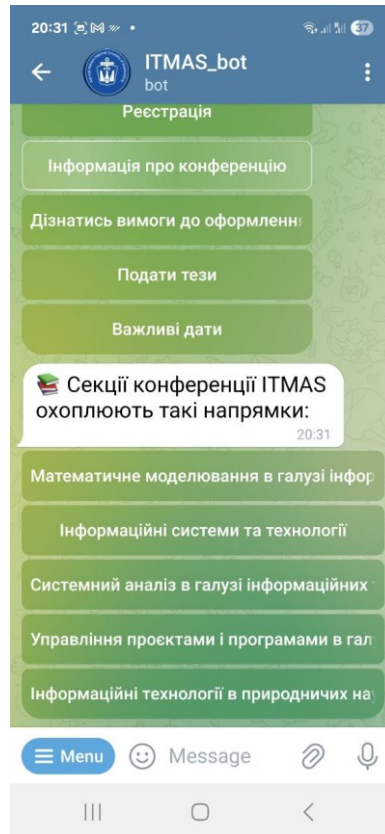


Рисунок Д.9 – Реакція бота при виборі пункту меню «Секції»



Рисунок Д.10 – Інформування учасника конференції про напрямки роботи

ДОДАТОК Г – ОПИС ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS

В цьому додатку виконано опис чат-боту для конференції ITMAS.

Найменування програми: Чат-бот для конференції ITMAS.

Позначення програми: ITMAS bot.

1 Функціональне призначення

1.1 Класи розв'язуваних завдань та призначення програми

Призначенням розроблюваного програмного забезпечення є автоматизація взаємодії між організаторами конференції ITMAS та її учасниками. Чат-бот забезпечує доступ до інформації про конференцію, секції, виконує реєстрацію учасників, завантаження тез, здійснює відповіді на часті запитання, забезпечує зворотний зв'язок 24/7.

1.2 Функціональні обмеження на застосування

Потребує доступу до Інтернету та встановленого додатку Telegram при використанні на смартфоні

2 Опис логічної структури

2.1 Опис структури

Чат-бот для конференції ITMAS складається з наступних частин:

- Клієнтська частина, через яку користувач надсилає повідомлення — це стандартний застосунок Telegram.
- Сервіс Telegram, що приймає повідомлення від користувача та перенаправляє їх до Rasa через Webhook.
- Головний модуль бот-ядра Rasa Core + NLU (логічний компонент, що відповідає за класифікацію намірів, який вбудований у NLU).
- Блок знань бота, що містить шаблонні відповіді на часті питання.
- Модуль Tracker Store, що зберігає поточний стан діалогу.
- Модуль Action Server, окремий сервер кастомної логіки.

- Модуль DatabaseManager, що здійснює роботу з базою SQLite: формує запити, перетворює дані на відповіді.

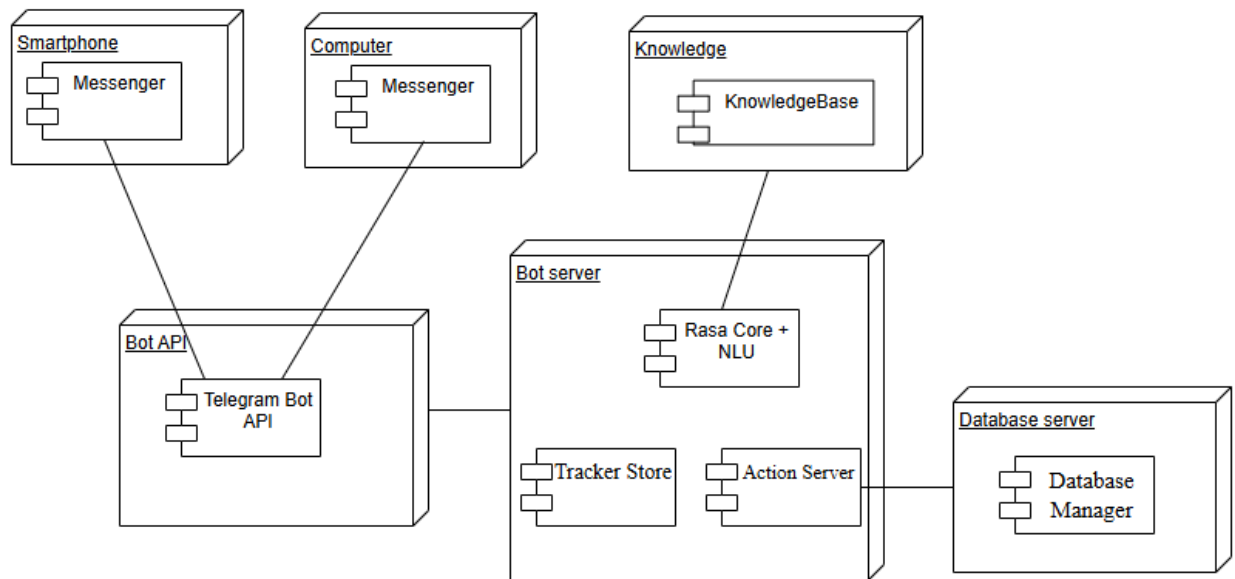


Рисунок Г.1 – Схема взаємодії

2.2 Мови програмування

Для розробки чат-боту для конференції ITMAS було обрано мову програмування Python 3.13, фреймворк Rasa та базу даних SQLite. В якості середовища розробки програм було використано IDE PyCharm.

2.3 Вхідні та вихідні дані

Для таблиці "user" вхідними даними є ідентифікатор, адреса електронної пошти, організація, посада, ПППБ.

Для таблиці "registration" вхідними даними є: ідентифікатор, назва, дата, тип.

Для таблиці "comment" вхідними даними є ідентифікатор та ідентифікатор тез, член організаційного комітету, дата, опис .

Для таблиці "theses" вхідними даними є ідентифікатор, ідентифікатор учасника, назва, дата завантаження, секція.

Для таблиці "section" вхідними даними є ідентифікатор та назва секції, визначення.

Вихідними даними є інформація про результати виконання ботом операцій на реєстрацію учасника, перегляд списку секцій конференції, надання опису секцій конференції, завантаження тез, надсилання повідомлень учасникам конференції.

3 Склад і функції

3.1 Склад програмного забезпечення

Програмне забезпечення складається з таких модулів:

- TelegramConnector: забезпечує взаємодію з Telegram Bot API, здійснює прийом повідомлень, пересилання запитів у ядро, доставка відповідей користувачу.
- DialogManager (Rasa Core): управляє потоком діалогу, зберігає історію взаємодії, обирає наступну дію (відповідь, виклик дії, форма
- ActionHandler: виконує кастомну логіку, звернення до бази, побудова динамічної відповіді, обробка фідбеку.
- Модуль Tracker Store, що зберігає поточний стан діалогу.
- Модуль Action Server, окремий сервер кастомної логіки.
- Модуль DatabaseManager, що здійснює роботу з базою SQLite: формує запити, перетворює дані на відповіді.

3.2 Функції програмного забезпечення

Розроблене в ході кваліфікаційної роботи програмне забезпечення має містити наступні функції:

- інтеграція з Telegram;
- виконувати сценарій привітання;
- вміти здійснювати розсилку інформаційного повідомлення;
- вміти здійснювати реєстрацію учасника конференції;
- вміти відповідати на питання стосовно тематики проведення конференції,
- вміти відповідати на питання стосовно секцій конференції, мати відповіді та приклади матеріалів стосовно секцій конференції;

- вміти відповідати на питання стосовно важливих дат конференції ITMAS;
- повинен надавати консультацію та вміти завантажувати матеріали до депозитарію конференції, повинна бути можливість відмінити завантаження, повторити завантаження;
- вміти надавати відповіді учасникам конференції стосовно статусу їх матеріалів;
- вміти давати відповіді стосовно оформлення матеріалів.

4 Умови застосування

4.1 Вимоги до складу та параметрів технічних засобів

Сервер має наступні мінімальні технічні та системні параметри:

- серверний процесор Intel Xeon Silver або Xeon Platinum;
- оперативна пам'ять: 32 гігабайти;
- жорсткий диск: SSD з підтримкою інтерфейсу NVMe обсягом 1 TB.

Клієнтська частина повинна мати наступні мінімальні технічні та системні параметри:

- процесор: Intel Pentium 4 або новіше із підтримкою SSE3;
- оперативна пам'ять: 8 ГБ;
- 8 ГБ вільної пам'яті

4.2 Вимоги до програмної сумісності

Для повноцінного функціонування чат-боту на комп'ютері необхідно використовувати операційну систему Windows версії не нижче 11. Також рекомендовано встановити десктопну версію месенджеру Telegram.

4.3 Спосіб виклику програми

Виклик програмного забезпечення відбувається шляхом надсилання повідомлення користувачем у Telegram-чат з ботом, що активує обробку даних через webhook-запит до сервера Rasa.

Вхідною точкою в програмне забезпечення є функція `start_bot()` у файлі `main.py`, яка ініціалізує `webhook`-зв'язок між Telegram Bot API та ядром чат-бота Rasa.

Метод `start_bot()` викликається для підключення до Telegram, реєстрації `webhook`-адреси й запуску циклу обробки вхідних повідомлень. Він використовується для забезпечення постійного з'єднання між користувачем і ботом, що дозволяє обробляти запити в реальному часі.

ДОДАТОК Д – ПРОГРАМА ТА МЕТОДИКА ВИПРОБУВАНЬ ЧАТ-БОТУ ДЛЯ КОНФЕРЕНЦІЇ ITMAS

1 Об'єкт випробувань

Об'єктом випробувань є чат-бот для конференції ITMAS.

2 Мета випробувань

Метою проведення випробувань є перевірка можливостей програмного забезпечення на відповідність вимогам технічного завдання.

3 Вимоги до програми

При проведенні випробувань необхідно перевірити, що розроблене програмне забезпечення реалізує наступні функції:

- інтеграція з Telegram;
- сценарій привітання;
- розсилка інформаційного повідомлення;
- реєстрація учасника конференції;
- інформування стосовно тематики проведення конференції;
- надсилати приклади матеріалів стосовно секцій конференції;
- інформування стосовно важливих дат конференції ITMAS;
- завантаження матеріали до депозитарію конференції;
- відміна завантаження матеріалів до депозитарію конференції;
- відповідати учасникам конференції стосовно статусу їх матеріалів;
- відповідати стосовно оформлення матеріалів.

4 Вимоги до програмної документації

До складу програмної документації, що надається на випробування входить:

- технічне завдання;
- текст програми;

- інструкція користувача;
- опис програми;
- програма і методика випробувань.

5 Засоби та порядок випробувань

5.1 Засоби випробувань

До складу технічних засобів, які мають використовуватися при випробуваннях належать:

- процесор: Intel Core i5 із підтримкою SSE3;
- оперативна пам'ять: 8 ГБ;
- пристрої вводу-виводу: маніпулятор типу "миша", клавіатура,

До складу програмних засобів, які мають використовуватися при випробуваннях належать:

- операційна система Windows 10;
- месенджер Telegram.

5.2 Порядок проведення випробувань

- 1) Перевірка комплектності програмної документації;
- 2) Перевірка функції інтеграції з Telegram;
- 3) Перевірка функції сценарію привітання;
- 4) Перевірка функції розсилки інформаційного повідомлення;
- 5) Перевірка функції реєстрації учасника конференції;
- 6) Перевірка функції інформування стосовно тематики проведення конференції,
- 7) Перевірка функції вміння надсилати приклади матеріалів стосовно секцій конференції;
- 8) Перевірка функції інформування стосовно важливих дат конференції ITMAS;
- 9) Перевірка функції завантаження матеріалів до депозитарію конференції;
- 10) Перевірка функції відміни завантаження матеріалів до депозитарію конференції;

- 11) Перевірка функції вміння відповідати учасникам конференції стосовно статусу їх матеріалів;
- 12) Перевірка функції вміння відповідати стосовно оформлення матеріалів.

6 Методи випробувань

Відповідно до порядку проведення випробувань було складено такі методи випробування чат-боту для конференції ITMAS:

1) Випробування функції реєстрація на конференцію:

Вхідні дані: Ім'я: "Іван Петренко", Email: "ivan@example.com", Посада: "доцент", Роль: "учасник".

Тест: Користувач проходить процес реєстрації через чат-бота, вводячи дані у відповідь на запити бота.

Очікуваний результат: Успішна реєстрація користувача, дані зберігаються в базі, бот надсилає підтвердження з ID.

1.1) Реєстрація – альтернативний потік:

Вхідні дані: Email: "ivan@example.com" (вже зареєстрований).

Результат: Бот повідомляє про помилку, що користувач вже існує. Пропонує авторизуватися.

2) Випробування функції перегляд важливих дат:

Вхідні дані: Команда користувача: «Важливі дати».

Тест: Користувач обирає відповідний пункт у боті.

Очікуваний результат: Бот надсилає перелік дат: дедлайн подання тез, дата конференції, реєстраційний термін.

2.1) Перегляд важливих дат – альтернативний потік:

Вхідні дані: Команда «Важливі дати», але календар ще не опублікований.

Результат: Бот повідомляє: «Наразі календар подій не опубліковано». Пропонує підписатися на оновлення.

3) Випробування функції подання тез:

Вхідні дані: Користувач надіслав PDF-файл "theses.pdf" + ключові слова: "Інформаційні системи та технології".

Тест: Користувач вибирає «Подати тези», надсилає файл та опис.

Результат: Бот успішно перевіряє реєстрацію, зберігає файл, надсилає підтвердження про прийом тез.

3.1) Подання тез – альтернативний потік:

Вхідні дані: Користувач не зареєстрований, надсилає файл.

Очікуваний результат: Бот успішно перенаправляє на сценарій «Реєстрація», після чого дозволяє подати тези.

4) Випробування функції валідація поданих матеріалів:

Вхідні дані: Координатор відкриває заявку, натискає "Підтвердити".

Тест: Перевірка поданих тез у системі координатором.

Очікуваний результат: Статус тез успішно змінюється на "Прийнято". Користувачу надходить повідомлення: "Ваші тези підтверджено".

4.1) Валідація – альтернативний потік:

Вхідні дані: Файл містить джерела оформлені не за вимогами.

Очікуваний результат: Координатор успішно додає коментар, бот встановлює статус "Потребує доопрацювання" та надсилає користувачу запит на виправлення.

5) Випробування функції отримати інформацію про секції конференції:

Вхідні дані: Користувач обирає пункт «Секції».

Тест: Бот надсилає перелік доступних напрямків роботи конференції із короткими описами.

Очікуваний результат: Користувач успішно отримує перелік секцій, може перейти до подання тез.

5.1) Отримання інформації – альтернативний потік:

Вхідні дані: Користувач обирає секцію «Інформаційні технології».

Результат: Бот надсилає розширену інформацію: тематика, ключові слова, рекомендації щодо тез, та пропонує подати матеріали.