

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ
БОТА-КОНСУЛЬТАНТА ДЛЯ ІНФОРМУВАННЯ
ПРАЦІВНИКІВ МЕТАЛУРГІЙНОГО ПІДПРИЄМСТВА

RESEARCH AND SOFTWARE IMPLEMENTATION
OF A CONSULTANT BOT FOR INFORMING EMPLOYEES
OF A METALLURGICAL ENTERPRISE

(назва теми)

Виконав(ла): студент(ка) 2 курсу, групи КНТ-213м

Спеціальності 122 Комп'ютерні науки

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Системи штучного інтелекту

ПОКРЕПОВ О.В.

(ПРИЗВИЩЕ та ініціали)

Керівник ШИТІКОВА О.В.

(ПРИЗВИЩЕ та ініціали)

Рецензент КОЦУР М.І

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Факультет ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти магістр
 Спеціальність 122 Комп'ютерні науки
 (код і найменування)
 Освітня програма (спеціалізація) Системи штучного інтелекту
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.

Сергій Субботін

« » 2024 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

ПОКРЕПОВА Олексія Володимировича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна реалізація бота-консультанта для інформування працівників металургійного підприємства
Research and software implementation of a consultant bot for informing employees of a metallurgical enterprise

керівник проєкту (роботи) к.т.н., ШИТИКОВА Олена Вікторівна,
 (науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “07” жовтня 2024 року № 398

2. Строк подання студентом проєкту (роботи) 01 грудня 2024 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз та дослідження особливостей ринку чат-ботів.

2. Вибір середовища розробки. 3. Проєктування вебзастосунку 4. Розробка телеграм-бота 5. Економіко-організаційна частина

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

Слайди презентації

(Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
90 с., 31 рис., 29 табл., 3 дод., 47 джерел..

ЧАТ-БОТ, БОТ-КОНСУЛЬТАНТ, МЕСЕНДЖЕР, ТЕЛЕГРАМ,
ТЕЛЕГРАМ-БОТ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ.

Об'єкт дослідження – розробка інформаційних сервісів для сфери промисловості.

Предмет дослідження – методи та засоби розробки бота-консультанта.

Мета дипломної роботи – розробка інформаційного чат-бота в месенджері, який є надійним джерелом довідкової інформації для працівників металургійного підприємства з метою підвищення їх обізнаності про технологічні процеси, заходи безпеки та новини підприємства.

Завдання дипломної роботи:

- дослідити особливості технологій чат-ботів;
- проаналізувати можливості застосування чат-ботів у металургійній сфері;
- проаналізувати популярні месенджери, їх переваги та недоліки;
- обрати технології для розробки чат-бота;
- розробити бота-консультанта для інформаційної обізнаності працівників металургійного підприємства.

Матеріали та технічні засоби: прийоми об'єктно-орієнтованого проектування, об'єктно-орієнтована мова програмування PHP, система керування реляційними базами даних MySQL, середовище розробки JetBrains PhpStorm, персональний комп'ютер під управлінням ОС Microsoft Windows 7.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 90 pages, 31 figures, 29 tables, 3 appendixes, 47 sources.

CHAT-BOT, BOT-CONSULTANT, MESSENGER, TELEGRAM, TELEGRAM-BOT, SOFTWARE.

Object of Research – Development of information services for the industrial sector.

Subject of Research – Methods and tools for developing a consulting bot.

Purpose of the Thesis – To develop an informational chatbot in the Telegram messenger that serves as a reliable source of reference information for employees of the metallurgical enterprise, aimed at increasing their awareness of technological processes, safety measures, and company news.

Tasks of the Thesis:

- investigate the features of chatbot technologies;
- analyze the possibilities of using chatbots in the metallurgical sector;
- analyze popular messengers, their advantages, and disadvantages;
- develop a methodology for selecting technologies for chatbot development;
- develop a consulting bot for the informational awareness of employees of the metallurgical enterprise.

Materials and technical means: techniques of object-oriented design, object-oriented programming language PHP, relational database management system MySQL, development environment JetBrains PhpStorm, personal computer running Microsoft Windows 7.

Вступ.....	9
1 Аналіз та дослідження особливостей ринку чат-ботів.....	11
1.1 Визначення і поняття чат-бота. Коротка історія появи та поширення.....	11
1.2 Типи чат-ботів. Функціонал чат-бота як інструменту комунікації	14
1.2.1 Бот як інструмент технічної підтримки для користувачів	15
1.2.2 Використання бота для віддаленого керування	16
1.2.3 Боти, що використовуються для транзакцій	17
1.2.4 Інформаційні чат-боти.....	18
1.3. Огляд месенджерів для використання чат-бота	20
1.4 Висновки за розділом 1	28
2 Вибір середовища розробки	29
2.1 Аналіз існуючих середовищ для розробки чат-ботів	29
2.2 Вибір засобів розробки вебзастосунку.....	33
2.3 Вибір сховища даних.....	36
2.4 Висновки за розділом 2	38
3 Проєктування вебзастосунку	40
3.1 Архітектура застосунка	40
3.2 Взаємодія компонентів застосунку	41
3.3 Діаграма варіантів використання	43
3.4 Діаграма класів вебзастосунку	44
3.3 Висновки за розділом 3	46
4 Розробка телеграм-бота	48
4.1 Реєстрація бота в Телеграм	48
4.2 Структура та реалізація бота.....	54
4.3 Тестування вебзастосунку.....	60
4.4 Опис роботи вебзастосунку	61
4.5 Висновки за розділом 4	62
5 Економіко-організаційна частина	63
5.1 Розрахунок основної заробітної плати	63
5.2 Розрахунок основної заробітної плати	68
5.3 Розрахунок додаткової заробітної плати	68
5.4 Єдиний соціальний внесок.....	69

5.5 Визначення витрат.....	69
5.6 Висновки за розділом 5	71
Висновки.....	72
Перелік джерел посилення	73
ДОДАТОК А Технічне завдання.....	77
ДОДАТОК Б Текст програми	80
ДОДАТОК В Слайди презентації.....	84

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

API	– Application Programming Interface;
IDE	– Integrated Development Environment
БД	– база даних;
МП	– мова програмування;
ООП	– об’єктно-орієнтоване програмування;
ОС	– операційна система;
ПЗ	– програмне забезпечення;
ПК	– персональний комп’ютер.

ВСТУП

У світі сучасних технологій чат-боти вже давно перестали бути просто інструментом для розваг чи звичних запитань. Вони перетворилися на фундаментальну складову комунікаційних стратегій у найрізноманітніших галузях, зокрема в промисловості. Ці програмні рішення, що здатні вести діалог із користувачами як за допомогою тексту, так і голосу, забезпечують не просто швидкий, а часто миттєвий обмін інформацією. В епоху, де швидкість і точність мають вирішальне значення, роль чат-ботів важко переоцінити: вони оптимізують процеси, підвищують продуктивність і стають невід'ємним елементом організаційного ландшафту.

Особливо актуальним є використання чат-ботів у металургійній сфері, де потреба в оперативному доступі до інформації стоїть на передньому плані. Це безпосередньо впливає не лише на ефективність виробництва, а й на безпеку працівників. Від надання базових даних, таких як графіки змін, до інформування про нові стандарти безпеки чи політики підприємства — спектр функцій чат-ботів вражає своєю багатогранністю. Наприклад, працівник, який має термінове питання щодо технічних регламентів, може отримати відповідь за лічені секунди, що значно зменшує час простою і підвищує ефективність внутрішніх комунікацій [1].

Впровадження чат-ботів здатне революціонізувати корпоративну взаємодію. Вони автоматизують рутинні запити, зменшуючи навантаження на персонал, який раніше витрачав значну частину робочого часу на відповіді співробітникам. Це, своєю чергою, дає можливість спеціалістам зосередитися на складніших завданнях, таких як технічні поломки чи стратегічне планування. Наприклад, металургійні підприємства з їхніми високими вимогами до безпеки та точності роботи отримують інструмент, що забезпечує постійний потік інформації та знижує ризики, пов'язані з людським фактором [2].

Що робить чат-боти ще більш привабливими для металургійної галузі? Їхня здатність функціонувати без необхідності в складному програмному

забезпеченні. Завдяки інтеграції з популярними платформами працівники отримують простий і доступний інструмент, що не потребує додаткових ресурсів для встановлення чи навчання. Такі боти можуть не лише надавати оперативну інформацію про виробничі процеси, але й організовувати навчальні заходи, збирати зворотний зв'язок чи навіть допомагати відстежувати стан здоров'я персоналу. Усе це сприяє формуванню інтерактивного робочого середовища, де кожен працівник відчувається почутим і значущим [1].

У цьому контексті розробка інформаційного чат-бота для металургійного підприємства стає не просто бажаною, а необхідною. Основною метою такого рішення є створення інструмента, здатного ефективно поєднувати функції інформаційного довідника, платформи для внутрішньої комунікації та засобу для зниження навантаження на персонал. Використовуючи месенджер як базу, цей чат-бот зможе стати надійним помічником для всіх працівників підприємства, забезпечуючи не лише доступ до актуальних даних, але й підтримуючи постійний зв'язок із командою.

Таким чином, чат-боти вже сьогодні демонструють колосальний потенціал для трансформації не лише комунікацій, а й організаційних процесів у галузі металургії. Їхня здатність адаптуватися до змін і потреб робить їх незамінним інструментом сучасного підприємства.

1 АНАЛІЗ ТА ДОСЛІДЖЕННЯ ОСОБЛИВОСТЕЙ РИНКУ ЧАТ-БОТІВ

1.1 Визначення і поняття чат-бота. Коротка історія появи та поширення

Чат-боти, або ж інтернет-боти, не є простою новинкою сучасності. Вони стали справжнім феноменом у світі цифрових технологій, виконуючи автоматизовані дії через інтерфейси, якими користуються звичайні люди. Слова "чат-бот" або "віртуальний співрозмовник" увійшли до лексикону ще в 1994 році, коли Майкл Молдинг створив програму під назвою Verbot (Verbal Robot) Джулія. Цей етап став лише початком, адже чат-боти здатні на значно більше. Вони імітують людську мовну поведінку, виступаючи у ролі співрозмовників у популярних месенджерах, і взаємодіють з платформами через API, виконуючи складну бізнес-логіку, що зберігається на віддалених серверах розробників. Від прогнозів погоди до повідомлень про важливі події – сфера їхніх можливостей безмежна.

Проте цікаво зауважити, що чат-боти не є винаходом тільки 21-го століття. Протилежно розповсюдженій думці, ідеї створення машинного інтелекту почали формуватися набагато раніше, в середині минулого століття. Ще в 1940-х роках, завдяки роботі англійського математика Алана Тюрінга, було закладено теоретичну основу для розвитку машинного мислення. Тюрінг, творець знаменитого "Тесту Тюрінга", що став не тільки перевіркою здатності машини мислити, як людина, але й основою для розвитку сучасної САРТСНА, поклав фундамент для еволюції того, що ми тепер називаємо штучним інтелектом .

1966 рік ознаменувався створенням першого чат-бота ELIZA в Массачусетському технологічному інституті під керівництвом Джозефа Вейценбаума. Простота цього бота, який імітував спілкування з психотерапевтом через ключові слова, не завадила йому стати віхою в історії. Він продемонстрував величезний потенціал обробки природної мови (NLP), що стало основою для подальших досягнень у цій сфері.

Але прогрес не зупинився. На початку 1970-х, в межах проєкту PARRY, розробленого Кеннетом Колбі, чат-боти змогли зробити значний крок уперед. PARRY мав здатність імітувати пацієнта з параноїдальною шизофренією, так віртуозно, що навіть досвідчені психіатри не могли відрізнити його від справжньої людини. Цей бот став першою віхою на шляху створення ботів з більш складною логікою, здатних імітувати не лише прості відповіді, але й багатопланові поведінкові моделі.

1980-ті роки принесли ще одну революцію у світ чат-ботів. Ролло Карпентер створив Jabberwacky – бот, який ставить за мету не просто генерувати відповіді, а й адаптуватися до діалогів у реальному часі. Він міг зберігати контекст попередніх розмов, інтегруючи гумор і неформальне спілкування, що виводило взаємодію з ботом на новий рівень. В результаті цього експерименту чат-боти почали набувати особливостей, притаманних живим співрозмовникам: вони стали менш механістичними, а взаємодія з ними більш природною.

Наступний етап розвитку чат-ботів привів до створення Dr. Sbaitso в 1992 році, програми, яка стала справжнім нововведенням для MS-DOS. Це був не просто чат-бот, а радник, що став відомим завдяки своєму синтезованому голосу. Хоча він і звучав штучно, здатність до граматично правильного мовлення вражала і відкривала нові горизонти для подальшого розвитку подібних технологій [3]-[9].

Сьогодні чат-боти стали невід'ємною частиною нашого цифрового середовища, проникаючи в усі сфери життя. Вони активно використовуються від електронної комерції до освіти, виконуючи функції технічної підтримки, консультантів, персональних асистентів. Розвиток цих технологій йде шаленими темпами завдяки інтеграції штучного інтелекту та машинного навчання, що дозволяє чат-ботам не тільки відповідати на запити, але й краще розуміти потреби користувачів і більш ефективно взаємодіяти з ними.

В 21 столітті чат-боти зазнали значного прогресу, головним чином завдяки розробкам у галузі штучного інтелекту (ШІ) та машинного навчання. Раніше ці боти могли лише імітувати діалоги на основі жорстко визначених

сценаріїв, але сучасні чат-боти використовують потужні алгоритми обробки природної мови (NLP), що дозволяє їм не лише генерувати відповіді, а й розуміти та адаптувати мову до різноманітних ситуацій. Платформи, такі як IBM Watson та Google Dialogflow, стали основними інструментами для створення інтерактивних чат-ботів, здатних адаптуватися до потреб користувачів і бізнесів, відкриваючи величезні можливості для автоматизації й оптимізації процесів.

Чат-боти пережили значні зміни з 2016 року, коли їх популярність зростає завдяки інтеграції з месенджерами, такими як Facebook Messenger, Slack та Телеграм. Раніше ці боти функціонували переважно на вебсайтах або в окремих додатках, але сьогодні користувачі можуть взаємодіяти з чат-ботами прямо через улюблені канали спілкування, що значно спрощує процес отримання інформації або виконання завдань. Наприклад, завдяки підтримці Facebook Messenger бізнеси отримали можливість автоматизувати спілкування з клієнтами, швидко відповідаючи на типові запитання і допомагаючи з оформленням замовлень. Крім того, Телеграм дозволяє створювати власних ботів за допомогою простих інструментів, що робить процес розробки доступним навіть для початківців.

З точки зору бізнесу, чат-боти стали ефективними інструментами для маркетингу, обслуговування клієнтів та автоматизації рутинних процесів. Завдяки таким інструментам, як IBM Watson і Google Dialogflow, компанії можуть швидко впроваджувати чат-ботів у свої робочі процеси, автоматизуючи обробку замовлень, бронювання, відповідь на типові питання та навіть продаж. Ці технології також допомагають аналізувати поведінку користувачів, збираючи дані для вдосконалення обслуговування та маркетингових стратегій [10]-[15].

Чат-боти набули особливої популярності й в освітній сфері. Сьогодні вони виконують роль асистентів, допомагаючи студентам знаходити необхідні навчальні матеріали, реєструватися на курси або навіть проходити тестування. Завдяки розвитку технологій NLP, чат-боти можуть надавати рекомендації та пояснення, допомагаючи студентам розв'язувати проблеми та краще розуміти складні теми. Це робить навчальний процес більш інтерактивним і доступним

для широкої аудиторії, а також сприяє індивідуалізації підходу, адаптуючи вміст і методи подачі матеріалу до потреб кожного студента.

Однак із широким впровадженням чат-ботів виникають й нові виклики, зокрема в контексті конфіденційності та безпеки даних. З кожним роком чат-боти отримують доступ до більшого обсягу особистої інформації, що викликає занепокоєння щодо можливих загроз для безпеки даних. Не менш важливим є етичний аспект їх використання, адже існують побоювання, що боти можуть бути застосовані для маніпуляції поведінкою користувачів, збору інформації без їхньої згоди чи навіть розповсюдження дезінформації.

Проте технології не стоять на місці і продовжують вдосконалюватися, відкриваючи нові можливості для застосування чат-ботів у різних сферах. Інтеграція чат-ботів з новітніми технологіями, такими як розширена реальність (AR) та Інтернет речей (IoT), може ще більше підвищити їх функціональність, дозволяючи взаємодіяти не тільки з користувачами, а й з фізичними об'єктами, створюючи нові форми автоматизації та обслуговування [16]-[19].

1.2 Типи чат-ботів. Функціонал чат-бота як інструменту комунікації

На сьогоднішній день не існує єдиної універсальної класифікації чат-ботів, але на основі досвіду використання різних продуктів можна виділити чотири основні типи, кожен з яких виконує специфічні завдання, хоча часто функціонал одного бота може перекривати кілька категорій. Чат-боти швидко здобули популярність у різних галузях і стали невід'ємною частиною сучасних комунікаційних процесів, виконуючи різноманітні функції в залежності від потреб користувачів і бізнесів.

Одним із основних типів є інформаційні боти, які забезпечують користувачів швидким доступом до різноманітних даних. Вони спеціалізуються на наданні відповідей на поширені запитання або інформуванні про останні новини. Ідея полягає в тому, щоб користувачі могли без зайвих зусиль отримати необхідну інформацію, не звертаючись до оператора чи не витрачаючи час на

пошук в інтернеті. Це значно економить час і ресурси, адже автоматизоване відповідання на типові запити дозволяє швидко задовольнити потреби без потреби у людському втручанні.

Такі боти використовуються в багатьох сферах – від новинних агентств до компаній, що надають послуги, адже здатність швидко отримати важливу інформацію без зайвих процедур чи затримок є важливою для користувачів, які потребують оперативних рішень.

1.2.1 Бот як інструмент технічної підтримки для користувачів

Чат-боти стали невід'ємною складовою сучасного бізнес-середовища, особливо в умовах швидкого розвитку електронної комерції в Україні, де технології автоматизації набувають все більшого значення. Одним з найяскравіших прикладів успішного впровадження чат-ботів є компанія Rozetka — найбільший інтернет-магазин в Україні, який активно використовує ці інструменти для покращення обслуговування клієнтів. Завдяки цьому компанія змогла значно підвищити рівень задоволення користувачів, надаючи оперативні та точні відповіді на запити, що в свою чергу дозволило знизити навантаження на службу підтримки та підвищити ефективність роботи.

Чат-бот Rozetka не лише швидко знаходить потрібні товари, але й надає вичерпну інформацію щодо доставки або відповідає на типові запитання без необхідності очікувати зв'язку з оператором. Це значно економить час клієнтів, дозволяючи їм швидше отримувати потрібну інформацію та ухвалювати рішення. Крім того, автоматизація процесу взаємодії дозволяє компанії зменшити навантаження на операторів, особливо під час пікових навантажень, коли кількість запитів значно зростає. Це дає змогу операторам зосередитися на вирішенні складніших питань, що потребують особистої уваги.

Удосконалення взаємодії за допомогою чат-ботів також сприяє підвищенню рівня задоволеності клієнтів. Адже швидкість та точність відповідей безпосередньо впливають на лояльність споживачів. Чат-боти здатні

зберігати історію попередніх запитів і на основі цього персоналізувати взаємодію, пропонуючи клієнтам товари, що відповідають їхнім попереднім покупкам або інтересам. Це збільшує ймовірність повторних покупок, що є важливим чинником для бізнесу, що прагне до стабільного розвитку та зростання.

Різні дослідження свідчать, що бізнеси, які впроваджують чат-ботів, здатні значно знизити витрати на обслуговування клієнтів, при цьому покращуючи якість послуг та зберігаючи конкурентоспроможність на ринку. Чат-боти дозволяють компаніям впроваджувати інноваційні технології, що відповідають вимогам сучасного споживача, забезпечуючи більш високий рівень ефективності та зручності для клієнтів [20]-[24].

1.2.2 Використання бота для віддаленого керування

Чат-боти стали популярним інструментом для інтеграції з різними системами, виступаючи зручним інтерфейсом для їх управління. Вони активно використовуються для взаємодії з інфраструктурою розумного дому, отримання сповіщень від систем моніторингу, а також для керування рекламними кампаніями і надання інформаційної підтримки на запит користувача. Сучасні рішення, як-от розумний дім, часто забезпечують API, через які чат-боти можуть взаємодіяти із контролерами. Така інтеграція дає змогу користувачам управляти своїм розумним домом без необхідності встановлювати окремі додатки. Це особливо зручно для популярних рішень на базі Xiaomi та інших відомих брендів, оскільки дозволяє скоротити час розробки і замінити створення окремих додатків для різних платформ чат-ботами, які працюють в межах популярних месенджерів.

Чат-боти також здобули популярність серед DevOps інженерів, які використовують їх для отримання сповіщень про позаштатні ситуації, моніторингу стану систем і дистанційного виконання команд. Завдяки готовим інтеграціям із більшістю сучасних сервісів, чат-боти стали простим і ефективним

рішенням, яке не потребує великих затрат часу чи ресурсів. Вони дозволяють забезпечити безпечний та безкоштовний спосіб створення інтерфейсу для керування системами й обміну повідомленнями.

Не можна не згадати і про роль чат-ботів у сфері технічної підтримки. Вони широко використовуються для прийому запитів користувачів, забезпечуючи оперативні відповіді через зручний інтерфейс. Таке рішення є ідеальним для так званої клієнтської частини ботів-асистентів, оскільки дозволяє швидко та ефективно взаємодіяти з клієнтами, надаючи їм необхідну допомогу та актуальну інформацію в режимі реального часу. Це не лише значно підвищує ефективність роботи фахівців з підтримки, але й покращує загальний досвід взаємодії користувачів із сервісами.

1.2.3 Боти, що використовуються для транзакцій

Транзакційні чат-боти є спеціалізованими програмами, призначеними для автоматизації фінансових та інших важливих операцій, таких як купівля-продаж товарів, перекази, управління замовленнями, а також інші види транзакцій з грошовими коштами. Вони знайшли своє застосування у багатьох різних галузях, включаючи банківську справу, роздрібну торгівлю, логістику, страхування і сферу послуг, де основним завданням є забезпечення швидкого та зручного обслуговування клієнтів без необхідності звертатися до живих операторів.

Ключова перевага транзакційних чат-ботів полягає в тому, що вони здатні здійснювати операції в реальному часі, використовуючи лише інтерфейс популярного месенджера або спеціального додатку. Наприклад, банківський чат-бот може запропонувати користувачам перевірити баланс, виконати переказ коштів, оплатити рахунки або навіть подати запит на кредит. Такі інструменти значно оптимізують фінансові процеси, підвищуючи зручність і швидкість, що особливо важливо для користувачів, котрі цінують простоту та мобільність у своїх фінансових операціях.

1.2.4 Інформаційні чат-боти

Інформаційні чат-боти виконують важливу роль у комунікації з користувачами, пропонуючи новини, розважальний контент або тематичну інформацію, адаптовану під інтереси аудиторії. Ці боти популярні завдяки інтеграції у месенджери, що підвищує ефективність взаємодії. Зокрема, користувачі можуть отримувати новинні оновлення, веселі зображення або анекдоти, як це було колись у сервісах на кшталт ICQ. Розміщення інформаційного бота безпосередньо в особистому просторі месенджера привертає більше уваги до контенту, ніж традиційні новинні сайти або додатки, що потребують окремого встановлення.

За технічною реалізацією інформаційні боти бувають трьох типів:

- а) сценарні – орієнтовані на передбачувані запити, що полегшує їхнє налаштування за допомогою конструкторів;
- б) інтелектуальні – використовують технології обробки природної мови (NLP), що дозволяє гнучко реагувати на запити, однак такі боти потребують складнішої розробки;
- в) гібридні – поєднують сценарний підхід із елементами інтелектуальної обробки, дозволяючи певну варіативність у введенні запитів. Виконують розробку методів проєктування інтерфейсу органайзера, експериментальна оцінка технічної ефективності.

Сучасні інформаційні чат-боти активно використовуються у сфері маркетингу та підтримки користувачів. Вони здатні обробляти численні запити та підтримувати інтерес аудиторії без потреби в додаткових людських ресурсах. Це робить їх важливим інструментом у бізнесі, особливо коли йдеться про розважальний контент або швидке надання інформації про товари чи послуги.

Структуру та приклад роботи чат-бота зображено на рисунку 1.1.

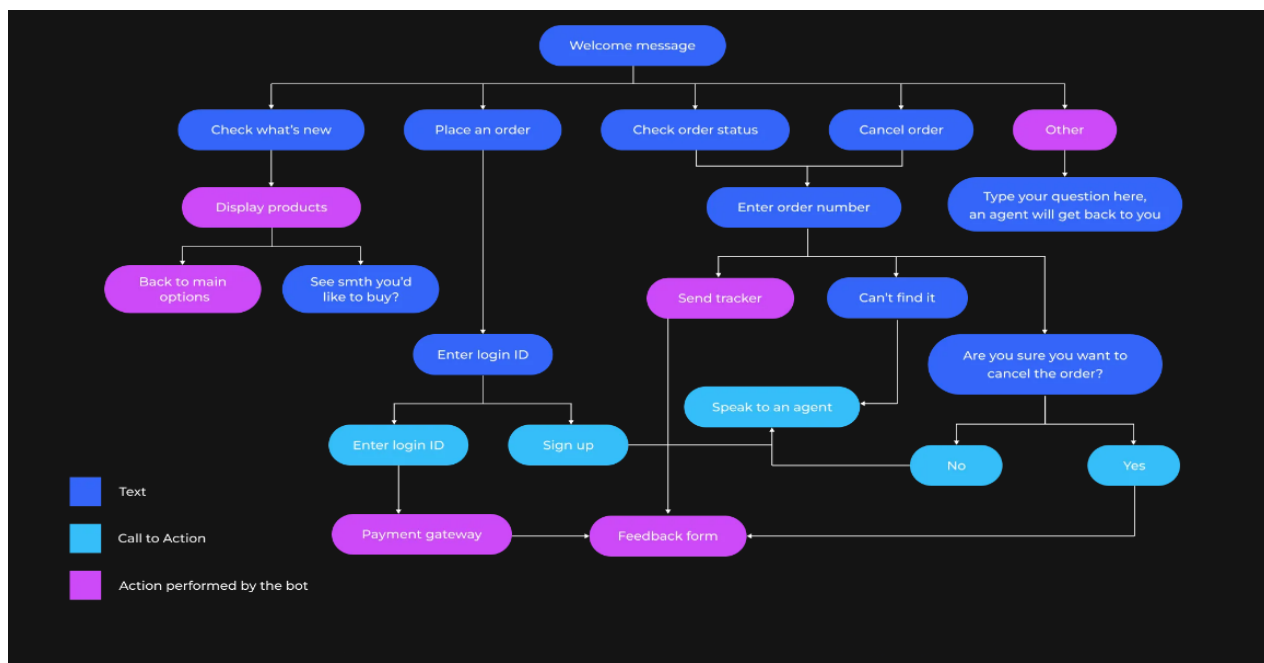


Рисунок 1.1 – Структура роботи чат-ботів [26].

Архітектура чат-бота складається з інтерфейсу користувача, серверної частини та бази даних. Інтерфейс забезпечує взаємодію, сервер обробляє запити, а база даних зберігає інформацію.

Опис галузі застосування ботів наведено у таблиці 1.1.

Таблиця 1.1 – Галузі застосування чат-ботів

Тип бота	Галузі застосування
Бот як асистент або фахівець технічної підтримки	Технічна підтримка, навчання та наставництво, збір зворотного зв'язку

Продовження таблиці 1.1

Тип бота	Галузі застосування
Бот як інтерфейс для віддаленого управління	Системи розумного будинку, індустрія IoT, управління бізнес-процесами
Транзакційні чат-боти	Фінансові послуги, електронна комерція, замовлення послуг
Інформаційні чат-боти	Новинні агрегатори, підтримка користувачів

1.3. Огляд месенджерів для використання чат-бота

Розвиток технологій розпочався ще в 80-х роках, і з того часу вони стали невід'ємною частиною повсякденного життя людей. Одним з важливих досягнень стало впровадження персональних комп'ютерів (ПК), які революціонізували спосіб комунікації. У 1996 році була випущена програма ICQ, що здобула популярність під неофіційною назвою «Аська». Ця безкоштовна платформа дозволяла обмінюватися повідомленнями через інтернет без обмежень за обсягом, що значно полегшило спілкування між людьми.

Зростання популярності таких сервісів привело до появи нових додатків, розроблених незалежними програмістами. Наприклад, Jabber став одним із варіантів для організації комунікації в корпоративному середовищі та домашніх мережах, пропонуючи альтернативу вже відомим програмам.

На початку 2000-х років, з розвитком мобільних технологій, на ринку з'явилися нові операційні системи, такі як Android і iOS. Ці платформи створили нові можливості для розробників, які почали адаптувати програми під мобільні пристрої. ICQ та інші подібні сервіси перейшли на мобільні платформи, забезпечуючи користувачам можливість завжди бути на зв'язку.

Такі програми, що забезпечують обмін повідомленнями, стали називати «месенджерами», відповідно до англійського терміна «message». Раніше використовувані платформи вимагали реєстрації через ім'я користувача та пароль. Але з ростом популярності мобільних телефонів з'явилася нова форма аутентифікації — через номер телефону, що значно спростило процес входу.

Станом на 2023 рік, на більшості сучасних смартфонів встановлені один або декілька месенджерів. До них належать популярні додатки, такі як Ватсап (заснований у 2009 році), Вайбер (2010), Facebook Messenger (2011), Snapchat (2011) і Телеграм (2013). Ці програми стали важливою частиною нашого повсякденного спілкування, надаючи зручні способи обміну інформацією та підтримки контактів.

Процент щоденних користувачів у найпопулярніших месенджерах зображено на рисунку 1.2:

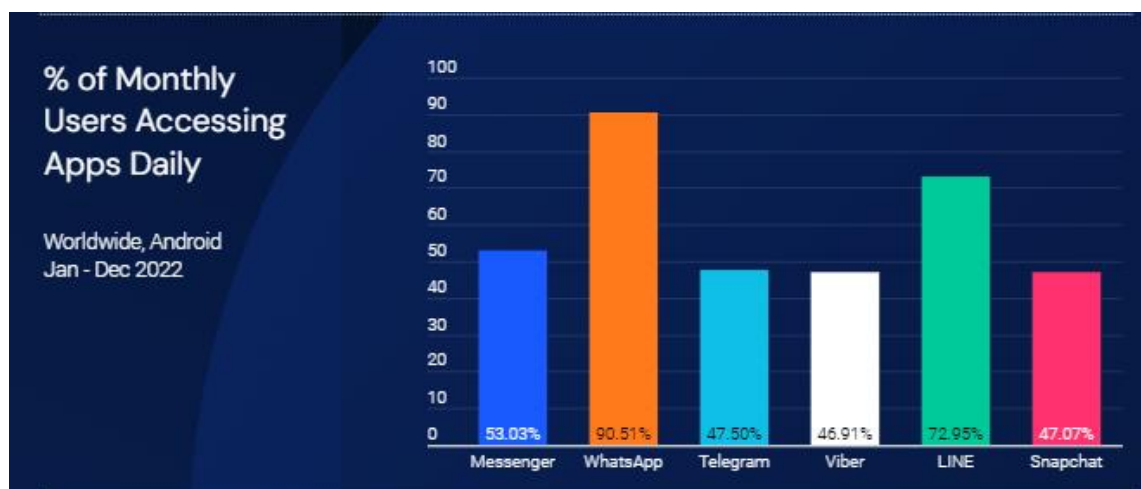


Рисунок 1.2 – Процент щоденних користувачів [27].

Процент щомісячних користувачів зображено на рисунку 1.3 [27]:

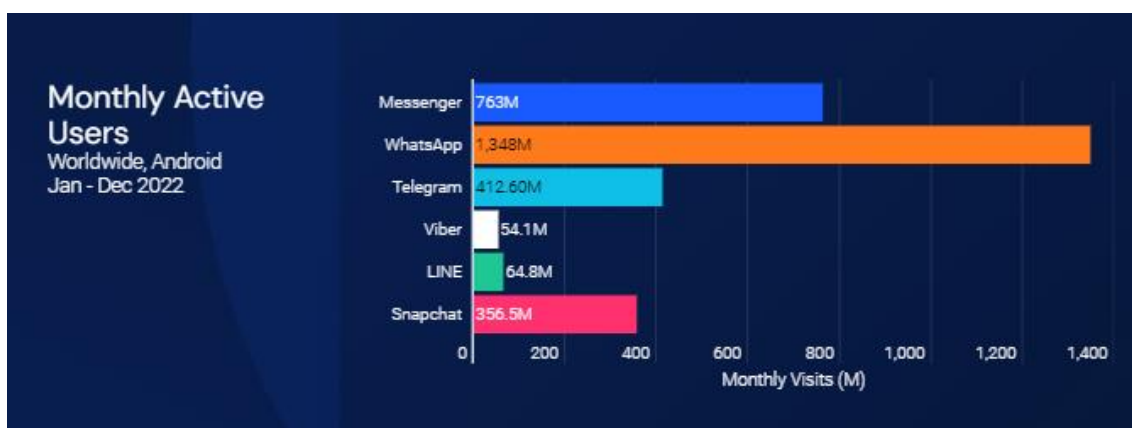


Рисунок 1.3 – Процент щомісячних користувачів [27].

Основні дані про найпоширеніші додатки для обміну повідомленнями свідчать, що близько 90% власників смартфонів у світі використовують принаймні один месенджер. Багато користувачів віддають перевагу спілкуванню через повідомлення, коли йдеться про взаємодію з компаніями. Зазначено, що месенджери тепер перевершують соціальні мережі за популярністю в отриманні новин. Додаток, пов'язаний із Facebook, не займає перші місця за популярністю лише у десяти країнах. Snapchat утримує статус найулюбленішого месенджера серед підлітків у США. Ватсап має 340 мільйонів активних користувачів в Індії, що є найвищим показником серед країн. Що стосується типів файлів, голосові повідомлення займають друге місце після фотографій у Facebook Messenger.

За загальною статистикою, у 2023 році у світі налічувалося приблизно 2,7 мільярда користувачів мобільних месенджерів. В активності месенджерів щомісяця спостерігається на 20% більше активних користувачів порівняно з соціальними мережами. Кожну хвилину відправляється понад 41 мільйон повідомлень. 36% споживачів, які користуються месенджерами, відзначають, що використовують принаймні два з них. Водночас 55% віддають перевагу обміну повідомленнями замість дзвінків до компаній. Близько 30% користувачів чат-ботів на основних платформах використовують їх для спілкування з представниками бізнесу. Провідні месенджери, такі як Facebook Messenger, Скайп і Телеграм, підтримують інтеграцію чат-ботів у свої системи. За даними опитувань у США, приблизно 30% користувачів чат-ботів повідомили про їх

використання для взаємодії з бізнесом, причому понад половина з тих, хто цікавиться цією можливістю, хочуть спілкуватися з місцевими підприємствами. Групові чати та обмін мультимедіа стали двома найчастіше використовуваними функціями в месенджерах. Важливо відзначити, що у 2018 році додатки для обміну повідомленнями почали перевершувати соціальні мережі за кількістю людей, які отримують новини.

У 2024 році Ватсап залишився беззаперечним лідером серед месенджерів, з більш ніж 2 мільярдами активних користувачів. Ця платформа забезпечує простий обмін текстовими повідомленнями, голосовими та відео дзвінками, а також підтримує групові чати. Ватсап щоденно обробляє понад 100 мільярдів повідомлень, що підкреслює його популярність у всьому світі. Користувачі цінують Ватсап за простоту та зручність, а також за регулярні оновлення, які додають нові функції, такі як підключення до чат-ботів для бізнесу.

WeChat - це китайський супермесенджер, який має приблизно 1.3 мільярда користувачів. Він не лише дозволяє обмінюватися повідомленнями, але й надає послуги соціальних мереж, мобільних платежів, ігор, новин та бізнес-сервісів, що робить його незамінним інструментом у повсякденному житті китайців. Ця платформа стала важливою частиною китайської економіки, дозволяючи користувачам здійснювати покупки, бронювати квитки та навіть оплачувати комунальні послуги через додаток. Важливим аспектом WeChat є його інтеграція з багатьма сервісами, що підвищує зручність користування.

Telegram продовжує демонструвати значне зростання популярності, досягнувши 800 мільйонів користувачів. Він відомий своїми сильними функціями безпеки, такими як шифрування повідомлень і можливість самознищення повідомлень. Telegram також підтримує великі групи до 200,000 учасників і має розвинену платформу ботів, що робить його привабливим для користувачів, які шукають функціональність і безпеку. Популярність Telegram зросла, особливо серед активістів і журналістів, які цінують анонімність та безпеку.

Facebook Messenger, із понад 1 мільярдом користувачів, пропонує широкий спектр функцій, зокрема можливість створення віртуальних кімнат для відео дзвінків. Messenger постійно вдосконалюється, зокрема у сфері безпеки, пропонуючи шифрування для особистих бесід. Крім того, платформа інтегрується з Facebook, що дозволяє користувачам легко спілкуватися з друзями і знайомими. Однією з останніх функцій є можливість надсилати гроші через Messenger, що робить його більш функціональним.

Вайбер, який також має понад 800 мільйонів активних користувачів, відрізняється наявністю безкоштовних голосових і відеодзвінків, а також великою колекцією стікерів. Останні оновлення Вайбер зосереджені на покращенні взаємодії з користувачами, вводячи функції, такі як зникаючі повідомлення та нові інтерактивні стікери. Вайбер також активно просувається в бізнес-сегменті, пропонуючи рішення для комунікацій між підприємствами та їхніми клієнтами.

Серед інших популярних месенджерів виділяються Signal та Snapchat. Signal, що забезпечує високий рівень конфіденційності, набрав популярність завдяки своїм функціям шифрування. Snapchat, зокрема, популярний серед молоді, завдяки своїм інтерактивним функціям, таким як зниклі повідомлення та різноманітні фільтри для фото та відео.

Таким чином, ці месенджери не лише забезпечують основні функції спілкування, але й постійно вдосконалюються, впроваджуючи нові можливості для своїх користувачів, що сприяє їхньому зростанню та популярності на глобальному рівні.

Топ-5 найбільш популярних месенджерів серед жителів українських міст:

- Вайбер;
- Фейсбук Messenger;
- Телеграм;
- Ватсап;
- Скайп.

В Україні спостерігається активне використання месенджерів, зокрема Вайбер, Фейсбук Messenger, Телеграм, Ватсап та Скайп, які займають топові позиції серед популярних комунікаційних платформ. Згідно з даними, Вайбер лідирує на українському ринку, оскільки 94% власників смартфонів користуються цим додатком хоча б раз на місяць. В середньому, користувач проводить близько 30 хвилин на день у Вайбер, причому 45% цього часу витрачається на дзвінки та відеодзвінки. Користувачі Вайбер надсилають понад 20 повідомлень щодня, а 35% активно використовують стікери [25]-[30].

Розглянемо більш детально лідерів українського ринку:

- Телеграм, який має охоплення 28% користувачів, поступається лише Вайбер, але перевершує WhatsApp, який використовується лише 22% українців. Телеграм виділяється своїм акцентом на безпеку та приватність, що робить його привабливим для користувачів, які шукають конфіденційності в спілкуванні. Додаток також активно використовується для створення чат-ботів і каналів, що дозволяє брендам і бізнесам взаємодіяти зі споживачами;
- Фейсбук Messenger залишається важливим інструментом для спілкування, але його популярність у Україні не така висока, як у світі. Згідно з даними, використання Messenger не є основним, проте він залишається платформою для обміну інформацією між користувачами Фейсбук;
- Скайп, хоча і має менше використання в Україні, все ще популярний серед користувачів, які цінують відеозв'язок, особливо в умовах віддаленої роботи та навчання.

Загалом, месенджери в Україні демонструють тенденцію до зростання, з акцентом на інтеграцію нових функцій, що відповідають потребам користувачів у швидкому та ефективному спілкуванні. Користувачі прагнуть до інтерактивності та різноманітності, що підштовхує розробників додатків впроваджувати нові можливості в свої платформи [30]-[33].

Порівняння найбільш популярних месенджерів наведено у таблиці 1.2

Таблиця 1.2 – Порівняння найбільш популярних месенджерів

		Фейсбук Messenger	Телеграм Messenger	Вайбер	Ватсап
1	2	3	4	5	6
Загальна інформація	Доступність на ПК	+	+	+	+
	Безкоштовне використання	+	+	+	+
	Наявність реклами	+	+	—	+
	Можливість персоналізації	+	+	—	+
	Сумісність з Android	+	+	+	+
Користувачі	Рейтинг в Google Play	4.2	4.4	4.4	4.3
	Щомісячні користувачі	130 млн	500 млн	823 млн	1500 млн
	Рейтинг в App Store компанії Apple	4.5	4	4.5	4
Контакти	Використовує функцію "двосторонньої відмови"	—	+	—	—
	Користувач може надіслати контактну інформацію повністю	—	+	+	+
	Має інтеграцію існуючих контактів	+	+	+	+

Продовження таблиці 1.2

		Фейсбук Messenger	Телеграм Messenger	Вайбер	Ватсап
1	2	3	4	5	6
	Має налаштування з обраними контактами	+	—	+	—
	Використовує GPS для зустрічі з людьми	—	+	—	—
Функції обміну повідомленнями	Має тимчасові зображення	—	+	—	—
	Має тимчасові текстові повідомлення	—	+	+	—
	Показує коли в останній раз користувач був активним	—	+	+	+
	Має функцію відправки повідомлень за графіком	—	+	—	—
	Двостороннє шифрування	—	+	+	+
	Немає необхідності у відкритті акаунта	—	—	—	—
	Має налаштування конфіденційності	—	+	+	+

На основі проаналізованих месенджерів для роботи було обрано платформу Телеграм, яка забезпечує оптимальний баланс функціональності, безпеки та конфіденційності. Телеграм підтримує широкий спектр мов, дозволяє використовувати анонімний вхід без SIM-карти та пропонує двостороннє шифрування. Завдяки можливості надсилати тимчасові повідомлення, зберігати конфіденційність і персоналізувати налаштування, цей месенджер відповідає ключовим вимогам проєкту. Крім того, він демонструє стабільне зростання популярності серед користувачів і має високий рейтинг у магазинах додатків, що свідчить про його надійність та зручність у використанні.

1.4 Висновки за розділом 1

У ході виконання розділу 1 були досягнуті наступні результати:

- а) вивчено визначення поняття «чат-бот» та його історичне походження;
- б) здійснено аналіз різних типів чат-ботів та їх функціональних можливостей;
- в) проведено огляд популярних месенджерів, визначено їхні сильні та слабкі сторони;
- г) для подальшої роботи було обрано платформу «Телеграм».

2 ВИБІР СЕРЕДОВИЩА РОЗРОБКИ

Для реалізації бота-консультанта був обраний месенджер Телеграм, адже він є зручним і універсальним кросплатформним застосунком. Цей месенджер дозволяє користувачам обмінюватися повідомленнями, а також надсилати файли різних форматів, що значно розширює можливості комунікації. Телеграм має низку переваг, серед яких – доступність на всіх основних платформах, таких як Android, iOS, Windows Phone, Linux тощо. Така доступність робить його зручним для широкого кола користувачів, забезпечуючи стабільний доступ до бота-консультанта з будь-якого пристрою.

2.1 Аналіз існуючих середовищ для розробки чат-ботів

У сучасних умовах цифровізації все більше організацій та підприємств інтегрують чат-боти для автоматизації комунікації з клієнтами та підвищення ефективності роботи. Завдяки розвиткові технологій, розробка чат-ботів стала доступнішою, а різноманітність платформ дозволяє вибирати рішення під конкретні потреби. Розглянемо найпопулярніші середовища для розробки ботів та проаналізуємо їхні особливості.

- microsoft Bot Framework є одним з найбільш відомих середовищ для створення ботів. Це рішення пропонує широкий спектр інструментів для розробки, тестування і розгортання ботів. Окрім цього, Microsoft Bot Framework підтримує інтеграцію з хмарними сервісами штучного інтелекту Azure AI, що дозволяє створювати боти з обробкою природної мови. Платформа підходить для різноманітних бізнес-задач, зокрема, для обслуговування клієнтів, оскільки вона підтримує популярні канали комунікації, такі як Microsoft Teams, Скайп та Slack [34];

- dialogflow від Google – це потужне середовище для розробки чат-ботів з підтримкою штучного інтелекту. Основною перевагою платформи є її інтеграція з Google Assistant та підтримка численних платформ, включаючи

Фейсбук Messenger. Завдяки підтримці машинного навчання та обробки природної мови, Dialogflow дозволяє створювати боти, які можуть вести природний діалог з користувачами. Платформа також пропонує багатомовну підтримку, що робить її корисною для глобальних рішень [35];

- ibm Watson Assistant забезпечує високий рівень інтелектуальної обробки та може інтегруватися з різними бізнес-платформами, такими як Slack та Фейсбук Messenger. Цей сервіс пропонує широкий набір функцій для створення та налаштування ботів, що можуть відповідати на складні питання, аналізувати дані і надавати аналітику. IBM Watson Assistant часто використовується у великих компаніях для автоматизації обслуговування клієнтів та інших бізнес-процесів [36];

- manyChat орієнтований на створення ботів для соціальних мереж і маркетингових завдань, зокрема для Фейсбук Messenger та Instagram. Платформа відрізняється простотою у використанні, не вимагає навичок програмування, тому є популярною серед маркетологів і бізнесів, які бажають автоматизувати комунікацію з клієнтами. Основний фокус ManyChat – це автоматизація маркетингових процесів та покращення взаємодії з клієнтами [37];

- rasa є відкритою платформою для розробки ботів з високим рівнем гнучкості та можливістю повного контролю над процесом створення чат-ботів. Rasa дозволяє будувати індивідуальні рішення, що відповідають конкретним вимогам бізнесу, що є важливим для компаній, які прагнуть зберегти дані в межах власної інфраструктури. Ця платформа особливо підходить для складних рішень та забезпечує підтримку обробки природної мови [38].

Для об'єктивного аналізу розглянемо кілька основних критеріїв, які допоможуть оцінити кожне середовище:

- підтримка природної мови (NLP) – наскільки добре середовище підтримує обробку природної мови для створення розмовних інтерфейсів;

- інтеграція з платформами – можливість підключення до різноманітних месенджерів, таких як Фейсбук Messenger, Slack, Google Assistant та інші;
- гнучкість налаштувань – рівень контролю над функціональністю та можливостями налаштування чат-бота.
- простота використання – зручність у використанні як для початківців, так і для досвідчених розробників.
- цільове використання – оптимальні сфери застосування платформи, такі як маркетинг, обслуговування клієнтів, автоматизація бізнес-процесів тощо;
- серед існуючих середовищ для розробки ботів кожне має свої особливості та переваги, які відрізняються залежно від цілей використання. Розглянемо кожне з них окремо;
- microsoft Bot Framework ідеально підходить для корпоративного середовища, особливо для компаній, які активно користуються сервісами Microsoft, адже воно забезпечує чудову інтеграцію, високу підтримку NLP і надійний захист даних. Ця платформа підходить для великих компаній, які потребують стабільного рішення для обслуговування клієнтів, хоча її рівень гнучкості може бути дещо обмеженим;
- dialogflow є потужним інструментом для створення інтелектуальних помічників із високою підтримкою NLP, відмінною інтеграцією з Google Assistant та іншими популярними платформами. Це середовище зручно використовувати для багатомовних рішень, що розширює можливості розробників. Хоча в безкоштовній версії функціонал обмежений, платна модель дозволяє користувачам максимально розкрити потенціал цієї платформи;
- ibm Watson Assistant є чудовим вибором для компаній, які шукають високий рівень кастомізації та безпеки. Це рішення відмінно інтегрується з бізнес-сервісами та дозволяє створювати глибоко персоналізовані боти. Завдяки підтримці багатомовності, наявності аналітики

та високому рівню захисту даних, цей інструмент підходить для великих проєктів. Однак для невеликих проєктів платна модель може бути недоцільною;

– manyChat — простий і доступний інструмент, який добре працює з соціальними мережами, такими як Фейсбук Messenger та Instagram. Ця платформа не має складних можливостей NLP чи потужної безпеки, однак через свою простоту та доступність, особливо у безкоштовній версії, вона є популярною для малого бізнесу та стартапів, які фокусуються на швидкому маркетинговому запуску [38].

Порівняння конструкторів представлено у таблиці 2.1

Таблиця 2.1 – Порівняння найпопулярніших конструкторів

	Microsoft Bot Framework	Dialogflow	IBM Watson Assistant	ManyChat
1	2	3	4	5
Підтримка NLP	+	+	+	±
Інтеграція з платформами	+	+	+	+
Гнучкість налаштувань	±	±	+	—
Простота використання	+	+	±	++
Цільове використання	Обслуговування клієнтів	Інтелектуальні помічники	Бізнес-додатки	Маркетинг
Підтримка багатомовності	±	+	+	±
Безпека даних	+	+	++	—

Продовження таблиці 2.1

	Microsoft Bot Framework	Dialogflow	IBM Watson Assistant	ManyChat
Вартість використання	Безкоштовно, платно для Azure	Обмежено безкоштовно, платно за розширені функції	Платне використання, безкоштовний тестовий період	Основні функції безкоштовні, платні опції
Технічна підтримка	+	+	+	±
Аналітика та звітність	+	+	++	+
Рівень кастомізації	±	±	+	±

Пояснення позначень до таблиці 2.1:

- а) ++ – дуже висока підтримка або дуже добра сумісність;
- б) + – висока підтримка або добре налаштована сумісність;
- в) ± – середня підтримка або можливість налаштування;
- г) – – низька підтримка або обмежена сумісність.

Враховуючи, що бот є соціальним проектом без комерційної складової, можливостей безкоштовної версії виявилось недостатньо, а платні тарифи мають надто високу вартість, було прийнято рішення створити власне рішення.

2.2 Вибір засобів розробки вебзастосунку

У розробці Телеграм-ботів програмісти обирають між кількома популярними мовами програмування, серед яких Python, JavaScript (Node.js) і

РНР. Щоб визначити, яка з них найбільше підходить для розробки Телеграм-ботів, розглянемо їх за такими ключовими критеріями:

- легкість вивчення. Цей критерій важливий для новачків, оскільки проста мова забезпечує швидке освоєння базового синтаксису і прискорює перехід до практичної реалізації;
- підтримка Telegram API. Telegram API є основним інструментом взаємодії з Телеграм-сервісом, що дозволяє обробляти повідомлення, взаємодіяти з користувачами та реалізовувати різноманітні функції бота. Чим більш розвинені бібліотеки підтримує мова, тим простіше реалізувати потрібний функціонал;
- підтримка асинхронності. Ця характеристика є важливою для ефективного виконання багатозадачних операцій, адже Телеграм-боти часто обробляють запити у реальному часі від великої кількості користувачів. Асинхронність дозволяє розподіляти навантаження та оптимізувати час обробки запитів;
- швидкість виконання. Для більш складних ботів або програм із великою кількістю запитів швидкість виконання є критичним фактором, оскільки тривала обробка даних може знижувати якість користувацького досвіду;
- розвиненість спільноти. Розвинена спільнота розробників забезпечує доступ до навчальних матеріалів, готових рішень та бібліотек, що спрощує процес розробки і сприяє швидкому вирішенню технічних проблем завдяки обміну досвідом.

На основі цих критеріїв порівняємо найпоширеніші мови програмування для створення Телеграм-ботів:

- python є однією з найпопулярніших мов для розробки ботів завдяки простоті синтаксису та доступності численних бібліотек, що підтримують Telegram API. Python є легким для вивчення, що дозволяє швидко освоїти його базові принципи і приступити до роботи над ботом. Завдяки бібліотекам, як-от `python-telegram-bot` і `pyTelegramBotAPI`, Python надає

можливості для створення гнучкого функціоналу. Водночас Python має певні обмеження щодо швидкості виконання та асинхронної обробки, що може бути проблемою для великих проєктів. Розвинена спільнота Python сприяє швидкому вирішенню труднощів і підтримує великий обсяг навчальних ресурсів [39];

– JavaScript (Node.js) є чудовим вибором для розробників, які вже знайомі з веб-технологіями, оскільки його підтримка асинхронного виконання коду дозволяє ефективно обробляти паралельні запити у реальному часі. Це робить JavaScript актуальним вибором для Телеграм-ботів, які потребують одночасної обробки численних запитів. Завдяки бібліотекам, як-от `node-telegram-bot-api` і `telegraf`, створення бота на JavaScript є зручним і швидким. Водночас, JavaScript має середній рівень легкості для вивчення і слабку типізацію, що може ускладнити розробку великих проєктів [40];

– php традиційно використовується в веб-розробці та підходить для створення простих ботів у веб-середовищі. Бібліотеки, як-от `telegram-bot-sdk`, дозволяють інтегрувати ботів у веб-додатки. Проте PHP менш поширений у розробці Телеграм-ботів порівняно з Python чи JavaScript, оскільки не підтримує високу асинхронність і має обмежені можливості для масштабованих застосунків. PHP залишається актуальним для проєктів, де Телеграм-бот є доповненням до веб-сайту [41]-[42].

Порівняння мов програмування представлено у таблиці 2.2

Таблиця 2.2 – Порівняння мов програмування

	Python	Java Script	PHP
Легкість вивчення	+	–	–
Підтримка Telegram API	+	+	+
Підтримка асинхронності	–	+	–
Швидкість виконання	–	+	–
Розвиненість спільноти	+	+	–

Для створення програми було вирішено використовувати мову програмування PHP (Personal Home Page Tools) — вибір, що не випадковий, а

ретельно зважений. Ця мова не лише відповідає ключовим критеріям сучасної веброзробки, а й зарекомендувала себе як одна з найефективніших у цій сфері. PHP — це інструмент, який дозволяє програмісту зануритися в предметну область із максимальною ефективністю, сприяючи продуктивності та відкриваючи шлях до створення високоякісних рішень.

Варто зауважити, що саме PHP стоїть за функціонуванням деяких із найвідоміших платформ у світі. Зокрема, Wikipedia — глобальна енциклопедія, Фейсбук — соціальна мережа-гігант, Tumblr — блог-платформа, яка збирає мільйони користувачів щодня. І, звісно ж, WordPress CMS — система управління контентом, що підкорила інтернет. Ці приклади демонструють не просто популярність, а надійність і потужність PHP у реальних умовах, де вимоги до продуктивності й стабільності неймовірно високі.

Таким чином, обравши PHP, розробник отримує в своє розпорядження не просто мову програмування, а ключ до світу, де створюються інновації та успіхи, що змінюють цифровий ландшафт.

2.3 Вибір сховища даних

Для кожного програмного застосунка вибір сховища даних є одним із найважливіших етапів розробки, оскільки він безпосередньо впливає на продуктивність, надійність та ефективність роботи всієї системи. При цьому важливо зазначити, що критерії вибору рішення для зберігання даних можуть значно варіюватися в залежності від вимог конкретного проєкту, специфіки бізнес-логіки та характеристик навантаження, з яким система повинна працювати.

Одним з основних факторів, які визначають вибір типу сховища даних, є характер даних, які необхідно зберігати. Це можуть бути документи з довільною структурою, дані з чітко визначеною схемою або навіть медіафайли. Наприклад, для даних, що вимагають суворого дотримання

схеми, доцільно обирати реляційну базу даних (RDBMS), а для зберігання даних без жорстких обмежень у структурі — документоорієнтовану базу даних, таку як MongoDB [43].

Більш того, вибір сховища даних часто залежить від вимог до масштабованості та відмовостійкості системи. Наприклад, для великих обсягів даних, які потребують високої пропускної здатності та обробки різних типів навантаження (читання чи запис), можуть бути потрібні спеціалізовані рішення, які забезпечують високу продуктивність та можливості для горизонтального масштабування [44].

Зважаючи на ці аспекти, було прийнято рішення використовувати реляційну базу даних для зберігання користувацької інформації, такої як підписки на розсилки, статистика використання розділів та запитання, задані фахівцям. У даному випадку оптимальним вибором є використання безкоштовного та надійного реляційного сховища даних, такого як MySQL або PostgreSQL [45].

Таблиця 2.3 наводить порівняння основних типів систем зберігання даних, де можна побачити ключові переваги кожного з варіантів.

Таблиця 2.3 – Порівняння систем баз даних

	RDBMS (MySQL or PostgreSQL)	Document-Oriented database (MongoDB)	File Storage
Підтримка транзакцій	+	—	—
Масштабованість	+	+	+
Швидкість читання	+	+	—
Швидкість запису	—	+	—
Підтримка складних запитів	+	—	—
Складність налаштування	—	+	+
Безкоштовний продукт	+	+	+

Вибір реляційної бази даних, зокрема PostgreSQL або MySQL, був зроблений через їх здатність ефективно працювати з типами даних, що

потребують стабільної та структурованої організації. Окрім того, важливим фактором був доступ до безкоштовних версій цих рішень з відкритим вихідним кодом, що є особливо корисним для проєктів на платформі PHP. MySQL, як частина LAMP-стеку (Linux, Apache, MySQL, PHP), є популярним вибором завдяки своїй поширеності та підтримці у розробницькому середовищі, а також можливості знизити витрати на ліцензії.

2.4 Висновки за розділом 2

У ході виконання дипломної роботи магістра було вирішено низку дослідницьких завдань, спрямованих на вивчення можливостей використання чат-ботів у металургійній галузі.

- проведено аналіз сучасних популярних мов програмування. Цей аналіз дозволив окреслити найкращі варіанти з точки зору продуктивності, зручності інтеграції та підтримки в межах обраної архітектури;
- відібрано три найбільш популярні мови програмування, які відповідають визначеним вимогам, і складено детальну порівняльну таблицю. Ця таблиця надала можливість об'єктивно оцінити переваги й недоліки кожного кандидата;
- для розробки чат-бота рекомендовано мову програмування PHP, яка продемонструвала найкращі показники відповідності критеріям;
- описано критерії вибору сховища даних, включно з параметрами продуктивності, надійності та масштабованості. Цей етап допоміг встановити вимоги до бази даних для оптимальної роботи чат-бота;
- складено порівняльну таблицю сховищ даних, у якій проаналізовано декілька варіантів на відповідність зазначеним критеріям. Проведене дослідження дозволило вибрати найбільш відповідне рішення для проєкту;

– обрано систему керування базами даних MySQL, що забезпечує необхідний рівень продуктивності та стабільності, а також відповідає архітектурним вимогам застосунку.

Крім того, було досліджено популярні платформи для розробки чат-ботів і визначено критерії оцінки сучасних інструментів для створення таких додатків. На основі проведеного аналізу розроблено методичні рекомендації зі створення бота-консультанта для металургійного підприємства. Ці рекомендації включають оптимальні технології, платформи та функціональні можливості, які найбільше відповідають потребам галузі.

3 ПРОЄКТУВАННЯ ВЕБЗАСТОСУНКУ

3.1 Архітектура застосунка

Застосунок функціонує на основі чітко визначеної точки входу — Controller, який отримує параметри HTTP-запиту. Сам по собі Контролер не занурюється у лабіринти бізнес-логіки; його роль скоріше схожа на диспетчера, що передає складні обчислення внутрішньому сервісу — BotEngine. Вся взаємодія між цими компонентами відбувається винятково через публічні методи, підтримуючи чітку архітектурну дисципліну та модульність.

Ізоляція сервісів бізнес-логіки від шару обробки запитів додає системі гнучкості та масштабованості. Завдяки такому підходу ядро застосунку стає універсальним інструментом, здатним обслуговувати різноманітні API-клієнти одночасно, не втрачаючи при цьому продуктивності.

Процес взаємодії виглядає наступним чином: користувач надсилає повідомлення боту через Телеграм, після чого сервер Телеграм транслює HTTP-запит із цим повідомленням на вебсервер. Вебсервер, у свою чергу, звертається до застосунку для обробки запиту. Застосунок аналізує отриману інформацію, виконує необхідні дії та формує відповідь. Ця відповідь відправляється знову у вигляді HTTP-запиту назад на сервер Телеграм, який передає її кінцевому користувачеві.

Таким чином, кожен етап процесу — від користувацького введення до повернення відповіді — є логічно відокремленим і структурно незалежним, що сприяє не лише ефективності, а й підтримці надійності системи у цілому (рисунок 3.1)

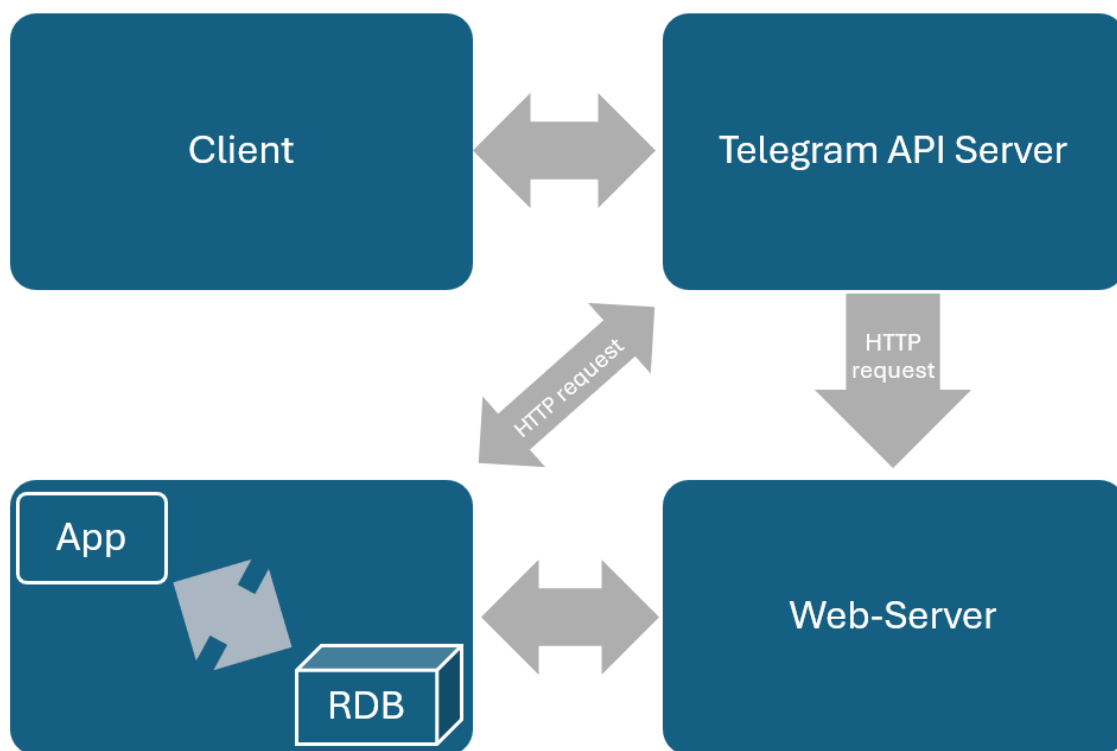


Рисунок 3.1 – Загальна архітектури системи

3.2 Взаємодія компонентів застосунку

Telegram-бот функціонує у форматі вебзастосунку. Головною точкою входу до застосунку виступає контролер, що обробляє вхідні HTTP-запити. Після отримання запиту система підключає конфігураційний файл із необхідними параметрами та активує BotEngine — центральний двигун, у якому сконцентрована вся бізнес-логіка (рисунок 3.2).

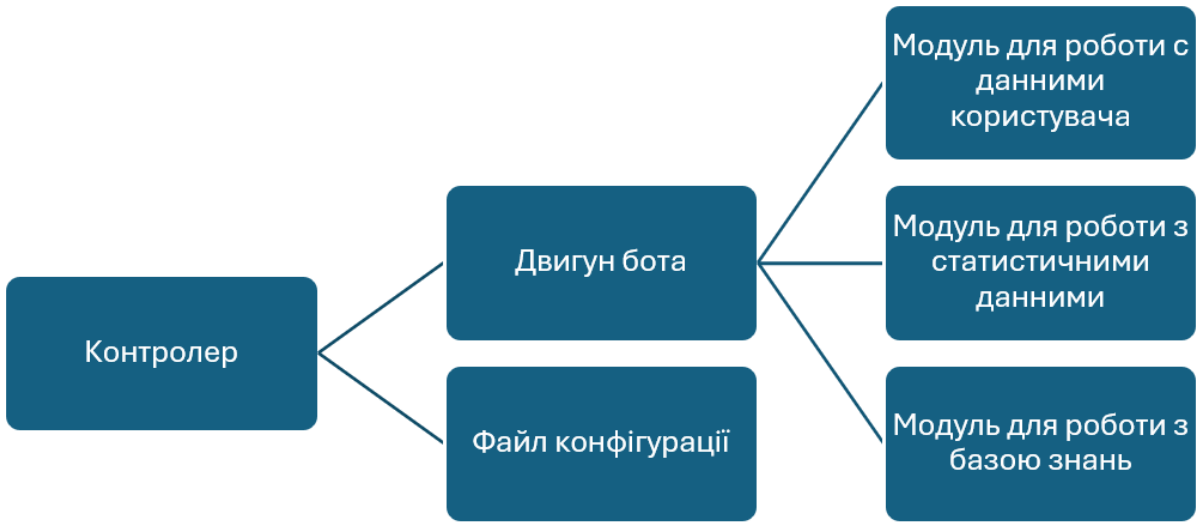


Рисунок 3.2 – Схема взаємодії компонентів застосунку

Модуль для обробки даних користувачів відіграє ключову роль у забезпеченні коректної взаємодії із базою даних (рисунок 3.3). Він не лише відповідає за надання необхідної інформації про користувачів, але й гарантує її надійне збереження у структурованому вигляді. Завдяки цьому модулю система може оперативно отримувати та оновлювати дані, підтримуючи актуальність інформації та забезпечуючи безперебійний доступ до профілів користувачів.

info	
id	INTEGER PRIMARY KEY AUTOINCREMENT
topic	TEXT UNIQUE
information	TEXT

Рисунок 3.3 – Структура бази даних

Модуль для роботи зі статистичними даними спеціалізується на обробці та збереженні відомостей. Цей компонент є незамінним при виконанні команди «Статистика», оскільки він надає необхідні дані для аналізу та

візуалізації показників. Завдяки модулю забезпечується оперативний доступ до актуальних статистичних відомостей, що дозволяє користувачам отримувати вичерпну інформацію про поточну ситуацію.

Модуль для роботи з базою знань виконує функції централізованого сховища інформації, відповідаючи за пошук і збереження даних у базі знань. Цей модуль є основою для реалізації команди «Пошук по базі знань», дозволяючи швидко знайти необхідну інформацію за заданими параметрами. Завдяки йому система забезпечує ефективний доступ до великих обсягів інформації, що сприяє точному та швидкому виконанню запитів користувачів.

3.3 Діаграма варіантів використання

Діаграма варіантів використання для Телеграм-бота металургійного підприємства відображає взаємодію користувача з системою та основні сценарії використання. У цій діаграмі визначені ключові варіанти використання, актор (користувач) та логіка взаємодії між ними.

Головним є користувач, який взаємодіє з ботом через Телеграм. Взаємодія починається з того, що користувач надсилає команду /start. У відповідь бот формує список доступних тем і надсилає їх користувачу у вигляді інтерактивної клавіатури. Це відображає перший варіант використання: виконання команди /start.

Після отримання списку тем користувач має можливість вибрати одну з них. Цей процес описується варіантом використання "Обрати тему". Вибір теми ініціює обробку запиту, під час якої бот надсилає відповідну інформацію, що відповідає вибраній темі. Цей сценарій описується як "Отримати інформацію".

У разі, якщо користувач надсилає некоректну команду або текст, який не відповідає доступним темам, бот інформує користувача про помилку. Цей сценарій описується варіантом використання "Помилка команди".

Взаємодія між елементами діаграми демонструє логічну послідовність дій, які виконуються у процесі роботи бота. Таким чином, діаграма дозволяє наочно відобразити, як користувач може взаємодіяти з ботом і які результати отримати в різних сценаріях (рисунок 3.4).



Рисунок 3.4 – Схема варіантів використання

3.4 Діаграма класів вебзастосунку

Діаграма класів Телеграм-бота для металургійного підприємства має ключові елементи системи та способи їх взаємодії. У структурі виділено основні класи, кожен із яких виконує конкретну роль у забезпеченні функціональності бота (рисунок 3.5).

Центральною складовою є клас TelegramBot, який відповідає за інтеграцію з Telegram API. Цей клас обробляє запити від користувачів та надсилає відповіді. Основні його методи дозволяють отримувати оновлення

через API, опрацьовувати команди та формувати відповіді, зокрема текстові повідомлення.

Клас `DatabaseManager` відповідає за взаємодію з базою даних. Його функціонал охоплює встановлення з'єднання, виконання SQL-запитів та обробку даних. Наприклад, метод `fetchTopics()` використовується для отримання списку тем, а `fetchInformation(topic)` — для вибірки інформації за конкретною темою. Завдяки цьому, вся робота з базою даних ізольована від решти логіки, що забезпечує чітку структуру і спрощує управління даними.

Клас `Keyboard` відповідає за створення інтерактивної клавіатури для Телеграм. Ця клавіатура дає змогу користувачеві вибирати теми або виконувати дії відповідно до контексту. Клас тісно взаємодіє з базою даних, перетворюючи отримані теми у зручні кнопки для вибору.

Ключовим елементом взаємодії між класами є функція `handleMessage`. Вона обробляє повідомлення від користувачів, аналізує текст команди та визначає подальші дії. Наприклад, при отриманні команди `/start` функція звертається до `DatabaseManager` для отримання списку тем, передає їх у `Keyboard` для створення клавіатури та надсилає результат користувачеві через `TelegramBot`. У разі вибору конкретної теми функція виконує пошук інформації у базі даних і повертає відповідь у текстовому форматі. Якщо команда некоректна, бот генерує повідомлення про помилку.

Загалом, класи працюють у тісній взаємодії, формуючи злагоджену систему. `TelegramBot` виступає посередником між користувачем і бізнес-логікою, реалізованою в `DatabaseManager` та `Keyboard`. Така архітектура забезпечує високу ефективність у розробці, зручність у підтримці та легкість розширення функціоналу Телеграм-бота, адаптованого до потреб металургійного підприємства.

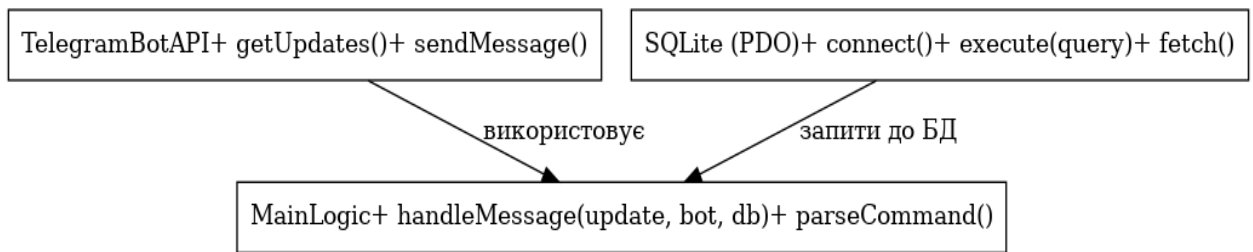


Рисунок 3.5 – Діаграма класів

3.3 Висновки за розділом 3

У розділі було розглянуто проєктування Телеграм-бота для металургійного підприємства, який базується на модульній архітектурі. Це забезпечує чітку взаємодію між основними компонентами системи. Центральним елементом є контролер, який виконує роль диспетчера HTTP-запитів, передаючи їх обробку внутрішнім сервісам, таким як BotEngine, модуль обробки користувацьких даних, статистичних даних та бази знань.

Розроблена схема взаємодії компонентів застосунку. Використання чітко визначених ролей для кожного з компонентів дозволяє досягти високої гнучкості системи, її масштабованості та структурної незалежності етапів обробки запитів. Такий підхід не лише полегшує інтеграцію нових функціональних модулів, але й сприяє підтримці стабільності та надійності роботи бота в різних сценаріях взаємодії з користувачами.

Крім того, чітка розподіленість функцій між модулями забезпечує не лише простоту підтримки та розширення функціоналу, а й можливість швидкої адаптації системи до змінних потреб підприємства. Це включає додавання нових команд, покращення алгоритмів обробки запитів і впровадження інноваційних підходів до роботи з великими обсягами інформації. Завдяки такій архітектурі система залишається актуальною навіть за умови еволюції бізнес-вимог та технологічного середовища.

Була проведена візуалізація структури через діаграми варіантів використання та класів, що демонструє як відбувається взаємодія між компонентами та які сценарії реалізує система. Такий підхід є оптимальним

для вирішення задач металургійного підприємства, забезпечуючи ефективну роботу з даними, простоту користування та швидкість обробки запитів.

Загалом, обраний архітектурний підхід закладає міцну основу для подальшого розвитку та адаптації системи під змінні потреби підприємства, водночас відповідаючи високим стандартам надійності й продуктивності.

4 РОЗРОБКА ТЕЛЕГРАМ-БОТА

4.1 Реєстрація бота в Телеграм

Етапами створення чат-боту є:

- реєстрація акаунта в Телеграм для створення чат-бота

Першим етапом у процесі розробки чат-бота є реєстрація акаунта в Телеграм. Це необхідно для того, щоб мати змогу взаємодіяти з платформою та використовувати її API для створення бота;

- пошук і взаємодія з ботом @BotFather

Для створення бота в Телеграм розробник повинен знайти офіційного бота @BotFather через пошук у Телеграм. @BotFather є основним інструментом для створення та налаштування нових ботів;

- активація бота за допомогою команди /start

Після знаходження бота @BotFather необхідно активувати його, надіславши команду "/start". Ця команда дозволяє почати взаємодію з ботом та отримати доступ до функцій для створення нового бота;

- створення бота за допомогою команди /newbot

Для створення нового бота потрібно виконати команду "/newbot", вказавши ім'я бота. Після цього система надає унікальний токен доступу, який використовуватиметься для взаємодії з ботом через API;

- копіювання токена доступу

Після виконання команди створення бота, @BotFather надає токен доступу. Цей токен є унікальним і необхідний для підключення до API та управління ботом. Важливо зберегти цей токен у безпечному місці;

- налаштування опису бота за допомогою команди /setdescription

Розробник може налаштувати опис бота за допомогою команди "/setdescription". Опис допомагає користувачам зрозуміти функціональність бота, що важливо для ефективного використання;

- додавання зображення профілю за допомогою команди /setuserpic

Для покращення ідентифікації бота розробник може додати зображення

профілю, використовуючи команду `"/setuserpic"`. Це дозволяє зробити бота більш привабливим і допомагає користувачам впізнавати його;

- вибір створеного бота для подальшої роботи

Після виконання попередніх етапів необхідно вибрати створеного бота для налаштування додаткових параметрів та продовження роботи з ним;

- написання та редагування опису бота

Розробник може додавати або редагувати опис бота, щоб він чітко відображав його функціональність та призначення;

- завантаження зображення профілю

На завершальному етапі розробник завантажує зображення профілю бота, яке буде відображатися в Телеграм. Це допоможе зробити бота більш привабливим та підвищити його впізнаваність серед користувачів [46].

Послідовність створення чат-боту зображена на рисунці 4.1.



Рисунок 4.1 – Діаграма процесу реєстрації бота в Телеграм

Перед початком розробки бота необхідно його зареєструвати та отримати унікальний ідентифікатор, який також є токеном. Для цього в Телеграм використовується спеціальний бот — @BotFather (рисунок 4.2) [46].

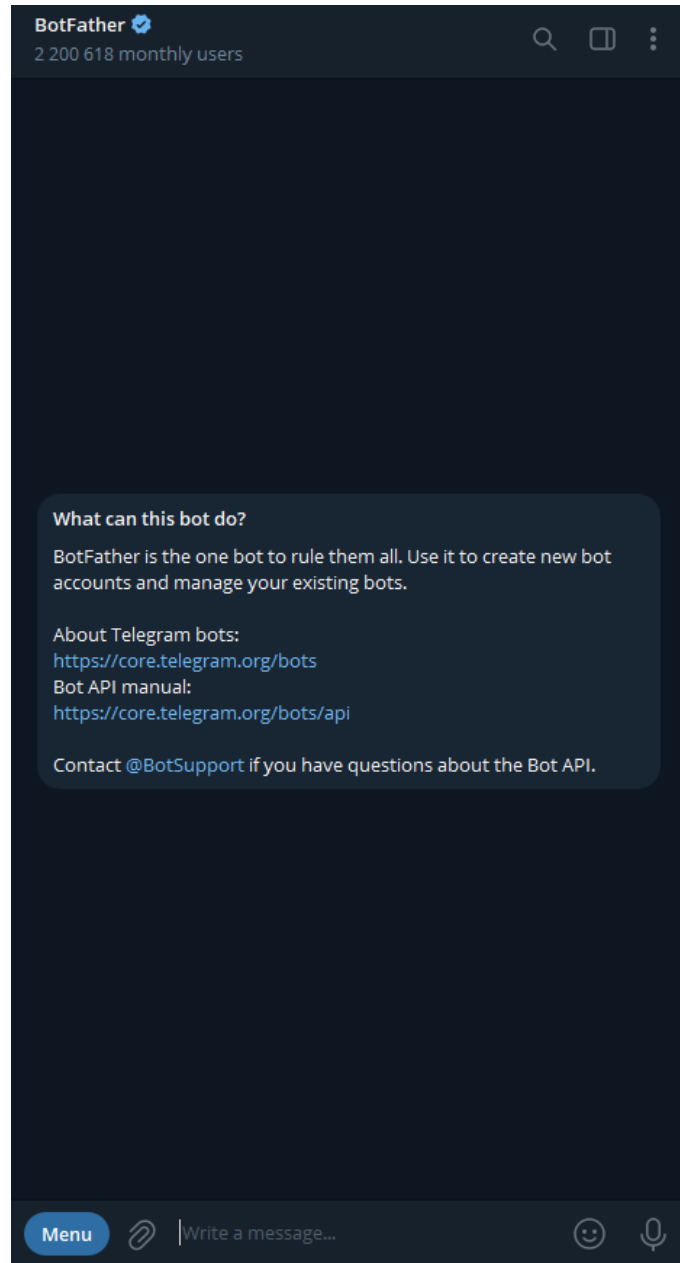


Рисунок 4.2 – Вигляд @BotFather

Для активації бота @BotFather необхідно надіслати команду `"/start"`. Після цього бот надасть користувачеві список доступних команд, які можна використовувати для створення та налаштування нового бота. Цей список команд дозволяє вибрати необхідну опцію, наприклад, створити нового бота,

налаштувати його опис або зображення профілю. Після надсилання команди `"/start"`, користувач отримає інтерфейс, який чітко вказує доступні функції для подальшої взаємодії з ботом (рисунок 4.3) [46].

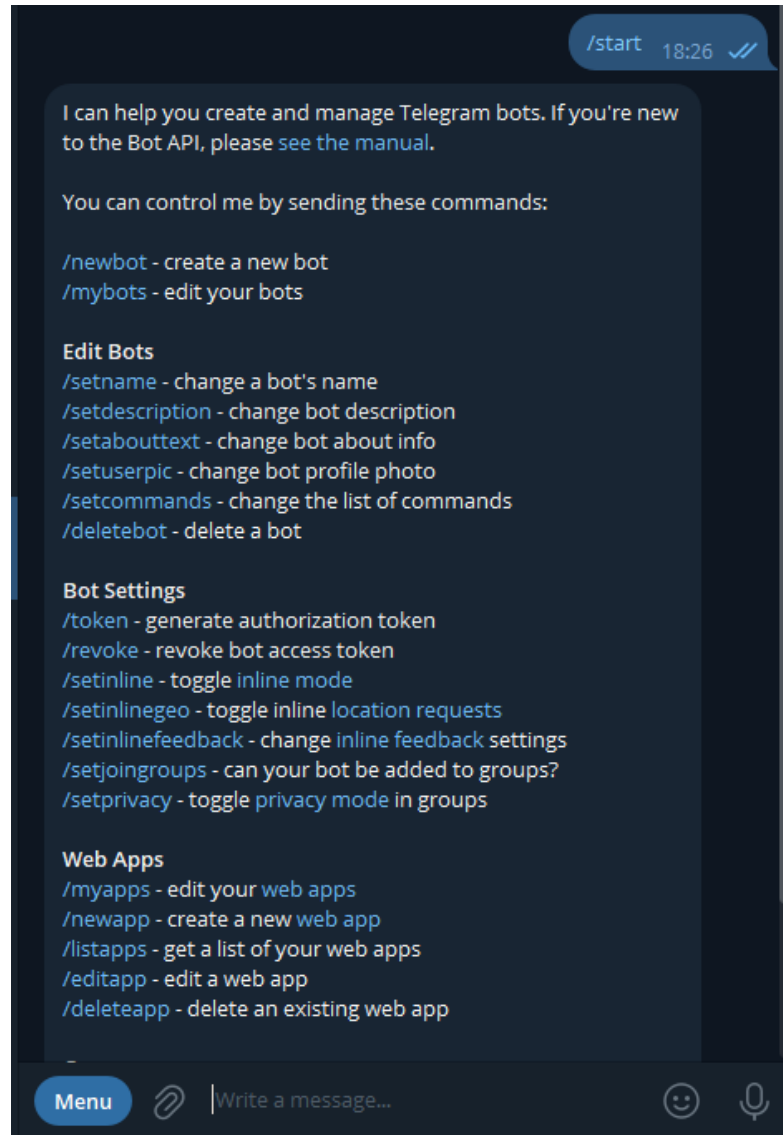


Рисунок 4.3 – Список команд @BotFather

Основною командою є `/newbot`. Після її відправлення бот попросить вказати ім'я для нового бота. Єдине обмеження для імені — воно повинно завершуватись на «bot». Якщо все пройшло успішно, @BotFather надасть токен бота та посилання для швидкого додавання бота до контактів (рисунок 4.4) [46].

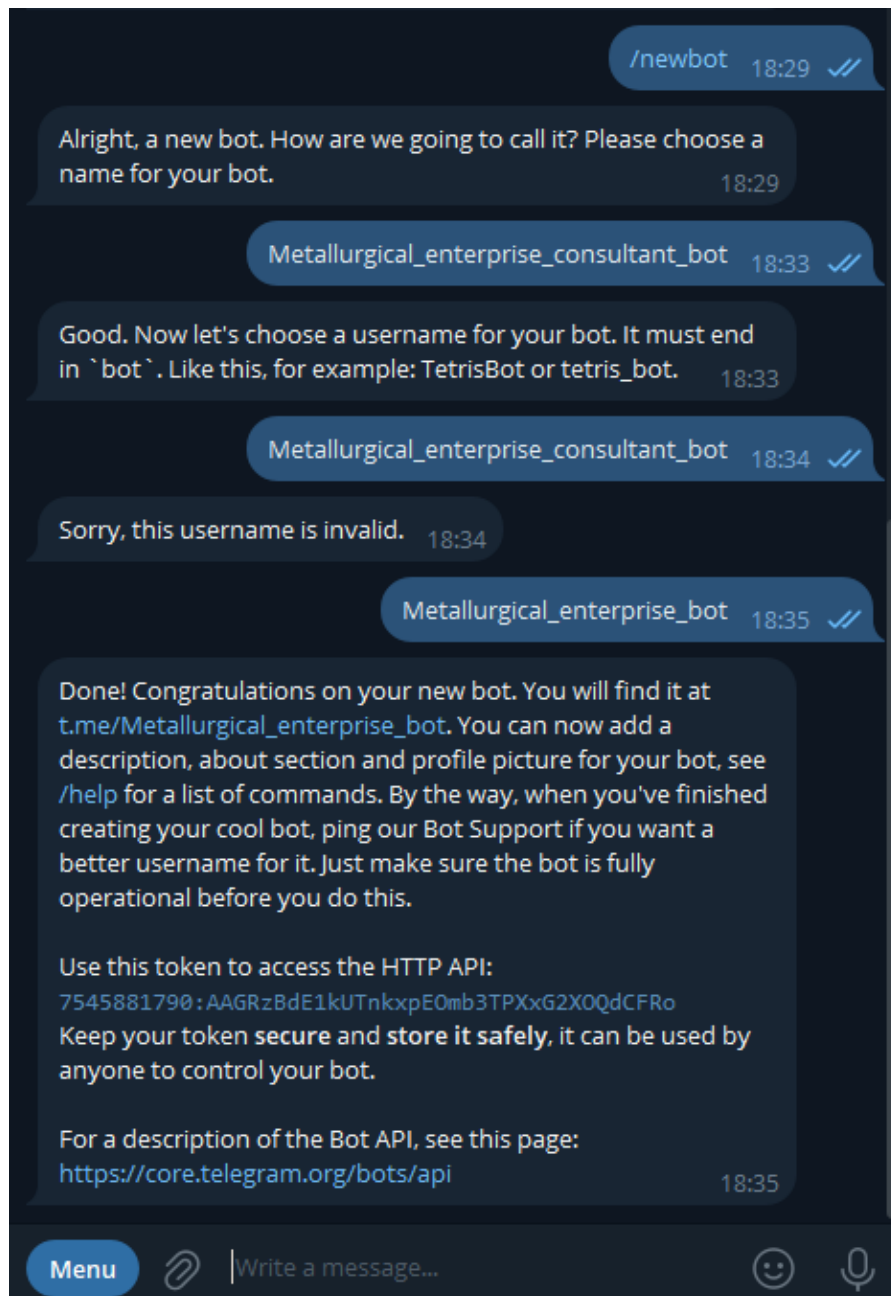


Рисунок 4.4– Реєстрація бота в @BotFather

Для налаштування додаткових параметрів, таких як іконка чат-бота, вітальне повідомлення, опис чат-бота, а також для видалення наявних чат-ботів, існують спеціальні команди (таблиця 4.1). Окрім команд для зміни основних параметрів, є також ряд команд, що дозволяють отримати незмінні параметри, такі як токен, а також призначати значення для певних параметрів, що наведені в таблиці 4.2.

Таблиця 4.1 – Доступні команди для зміни чат-ботів

Команда	Опис
1	2
/setname	Змінює існуюче ім'я бота
/setdescription	Встановлює текст, який відображатиметься при першому відкритті чат-бота
/setabouttext	Встановлює текст в розділі «Про чат-бота»
/setuserpic	Додає вибране зображення профілю для чат-бота
/setcommands	Створює список доступних команд для чат-бота
/deletebot	Видаляє вибраного чат-бота

Таблиця 4.2 – Доступні команди для налаштування чат-бота

Команда	Опис
/token	Повертає токен доступу, отриманий раніше для чат-бота
/revoke	Анулює токен доступу до чат-бота
/setinline	Включає або вимикає можливість виклику бота з інших чатів
/setinlinefeedback	Дозволяє отримувати інформацію про кількість обраних користувачами команд
/setjoingroup	Визначає, чи може бот бути доданий до групових чатів
/setprivacy	Включає режим конфіденційності для бота, обробляючи інформацію окремо для кожного користувача в чаті

Після того, як усі налаштування виконано в Телеграм і отримано токен доступу, можна переходити до наступного етапу — розробки програмної частини чат-бота. Цей етап включає створення програмного забезпечення, яке буде взаємодіяти з API Telegram, обробляти запити користувачів та

виконувати необхідні функції бота, використовуючи отриманий токен для аутентифікації і доступу до ресурсів платформи.

4.2 Структура та реалізація бота

Telegram-бот — це вебзастосунок, що здійснює обмін даними з сервером Telegram через HTTP-запити. Бот приймає повідомлення від користувачів, обробляє їх і повертає відповіді відповідно до закладеної логіки. У межах цього проєкту було розроблено бот для працівників металургійного підприємства, який забезпечує доступ до важливої інформації, такої як правила техніки безпеки, графік роботи, соціальні гарантії тощо.

Основним елементом проєкту є файл `index.php`, який виступає точкою входу до програми. Саме через цей файл бот взаємодіє з користувачами та сервером Telegram. У ньому налаштовується підключення до бази даних SQLite, яка зберігає всю необхідну інформацію, зокрема, теми та відповідні відповіді. Крім того, у цьому файлі відбувається підключення до Telegram API за допомогою спеціалізованої бібліотеки, що дозволяє приймати повідомлення від користувачів і надсилати їм відповіді.

Перш за все, бот підключає необхідні бібліотеки, що надаються через менеджер залежностей Composer. Це забезпечує доступ до основних компонентів для роботи з Telegram API і іншими необхідними пакунками. Після цього налаштовується з'єднання з базою даних SQLite. Якщо таблиця для збереження інформації ще не існує, вона створюється автоматично, що дозволяє ботам ефективно працювати навіть при першому запуску програми. У цій таблиці зберігається інформація, яку бот буде надавати користувачам.

Основний цикл роботи бота, розміщений у цьому ж файлі, постійно опитує сервери Telegram, отримує нові повідомлення та передає їх на обробку. Ця архітектура дозволяє боту працювати безперервно, обробляючи кожне повідомлення користувачів у реальному часі (рисунок 4.5)

```

<?php

require __DIR__ . '/vendor/autoload.php';

use ...

// --- Database Connection (PDO) ---
try {
    $db = new PDO( 'dsn: 'sqlite:metallurgy_info.db');
    $db->setAttribute( attribute: PDO::ATTR_ERRMODE, value: PDO::ERRMODE_EXCEPTION);
    $db->exec( statement: "CREATE TABLE IF NOT EXISTS info (
        id INTEGER PRIMARY KEY AUTOINCREMENT,
        topic TEXT UNIQUE,
        information TEXT
    )");
}

```

Рисунок 4.5 – Структура проєкту в JetBrains IDE PhpStorm

Одна з ключових частин коду — це постійний цикл, який перевіряє наявність нових оновлень від Телеграм за допомогою методу `getUpdates()`. Цей цикл безперервно опитує сервер Телеграм, щоб отримати нові повідомлення від користувачів і передати їх на обробку. Якщо з'являється нове повідомлення, викликається функція `handleMessage`, яка займається безпосередньою обробкою запиту. У цій функції відбувається перевірка, чи є в повідомленні команда `/start`, і у разі її наявності бот формує інтерактивну клавіатуру з доступними темами, які зберігаються в базі даних. Користувач може вибрати одну з тем, і бот відповідає йому відповідною інформацією (рисунок 4.6)

```

$telegram = new Api('7743347924:AA6nU8WXHxjchEpj6LqcW32f16Z6UC6u82I');
$lastUpdateId = 0;

while (true) {
    try {
        $updates = $telegram->getUpdates(['offset' => $lastUpdateId + 1]);
        foreach ($updates as $update) {
            if ($update->getMessage()) {
                handleMessage($update, $telegram, $db);
            }
            $lastUpdateId = $update->getUpdateId();
        }
    } catch (Telegram\Bot\Exceptions\TelegramSDKException $e) {
        error_log( message: "Помилка Telegram API: " . $e->getMessage());
    } catch (PDOException $e) {
        error_log( message: "Помилка бази даних: " . $e->getMessage());
    } catch (Exception $e) {
        error_log( message: "Інша помилка: " . $e->getMessage());
    }
    sleep( seconds: 1);
}

```

Рисунок 4.6 – Використання методу `getUpdates`

Кожне повідомлення користувача обробляється шляхом виконання SQL-запиту до бази даних, щоб отримати текстову відповідь для обраної теми. Якщо ж тема не була знайдена в базі, бот відправляє повідомлення про помилку. Бот також здатний відправляти повідомлення у форматі HTML, що дозволяє зберігати форматування тексту та робить взаємодію з користувачем більш зручною. Якщо ж повідомлення не є відомою командою, бот повідомляє про некоректний запит (рисунок 4.7)

```
} catch (Telegram\Bot\Exceptions\TelegramSDKException $e) {
    error_log( message: "Помилка Telegram API: " . $e->getMessage());
} catch (PDOException $e) {
    error_log( message: "Помилка бази даних: " . $e->getMessage());
} catch (Exception $e) {
    error_log( message: "Інша помилка: " . $e->getMessage());
}
```

Рисунок 4.7 – Помилки які можуть виникнути

У функції `handleMessage` ми реалізуємо логіку обробки повідомлень, що надходять від користувачів бота. Спочатку ми отримуємо дані з нового повідомлення, зокрема текст, за допомогою методу `$message->getText()`. Якщо текст повідомлення містить команду `/start`, ми виконуємо запит до бази даних, щоб отримати список доступних тем для користувачів. Після цього генеруємо клавіатуру з цими темами, щоб користувач міг вибрати одну з них.

Якщо команда не є `/start`, ми перевіряємо, чи є текст повідомлення темою, яка зберігається в базі даних. Для цього виконуємо SQL-запит, щоб знайти інформацію, що відповідає введеній темі. Якщо така інформація є, ми надсилаємо користувачеві відповідь з отриманим текстом у форматі HTML. Якщо ж тема не знайдена, ми відправляємо повідомлення з повідомленням про помилку, інформуючи користувача, що команда або тема не існує.

Таким чином, у функції `handleMessage` здійснюється обробка запитів, взаємодія з базою даних і відповідь користувачу з відповідною інформацією чи повідомленням про помилку (рисунок 4.8)

```
function handleMessage($update, $telegram, $db)
{
    $message = $update->getMessage();
    $chatId = $message->getChat()->getId();
    $text = $message->getText();

    if ($text == '/start') {
        $stmt = $db->query('SELECT topic FROM info');
        $topics = $stmt->fetchAll(PDO::FETCH_COLUMN);
        $keyboard = Keyboard::make()
            ->row(array_map(function ($topic) {
                return Keyboard::button(['text' => $topic, 'request_contact' => false]);
            }, $topics));

        $telegram->sendMessage([
            'chat_id' => $chatId,
            'text' => 'Вітання! Цей чат-бот створений для зручного та швидкого доступу до важливої інформації для працівників металургійного підприємства. Обирайте розділ, що вас цікавить..',
            'reply_markup' => $keyboard,
        ]);
    } else {
        $stmt = $db->prepare('SELECT information FROM info WHERE topic = ?');
        $stmt->execute([$text]);
        $information = $stmt->fetchColumn();

        if ($information) {
            $telegram->sendMessage([
                'chat_id' => $chatId,
                'text' => $information,
                'parse_mode' => 'HTML',
            ]);
        }
    }
}
```

Рисунок 4.8 – Функція handleMessage

У функції обробки повідомлень від користувачів, бот перевіряє, чи є команда /start. У разі її наявності, бот формує і відправляє користувачеві інтерактивну клавіатуру з доступними темами, які зберігаються в базі даних. Це дає можливість користувачам вибрати одну з тем, і отримати необхідну інформацію.

Основний цикл програми забезпечує безперервну роботу бота, перевіряючи оновлення від Телеграм і обробляючи їх. Це дозволяє боту діяти в реальному часі, оперативно реагуючи на запити користувачів. Весь код логічно поєднує обробку повідомлень, роботу з базою даних і взаємодію з Telegram API, створюючи ефективний інструмент для доступу до важливої інформації працівників металургійного підприємства (рисунок 4.9).

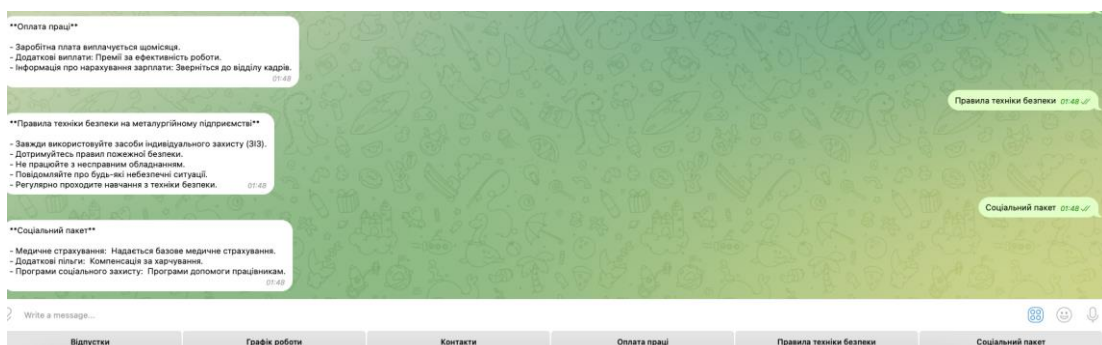


Рисунок 4.9 – Робота бота

У використаному коді реалізація функціональності бота здійснена через кілька основних функцій, кожна з яких виконує специфічну роль у забезпеченні взаємодії з користувачами, базою даних і Telegram API.

Ключовою функцією є `handleMessage`, яка відповідає за обробку повідомлень від користувачів. Вона отримує оновлення, перевіряє наявність повідомлень і, залежно від їх вмісту, виконує необхідні дії. Якщо в повідомленні міститься команда `/start`, функція генерує клавіатуру з доступними темами, отриманими з бази даних. Якщо текст повідомлення відповідає одній з доступних тем, виконується SQL-запит до бази для отримання детальної інформації, яка потім надсилається користувачеві у форматі HTML.

Ще одна важлива складова функціональності — це взаємодія з базою даних за допомогою SQL-запитів через об'єкт `$db`. При команді `/start` бот запитує всі доступні теми, які відображаються на клавіатурі у вигляді кнопок. Якщо користувач обирає одну з тем, бот виконує запит для отримання відповідної інформації з бази даних.

Також передбачено обробку помилок. Якщо запит користувача не відповідає жодній з тем у базі даних, бот повідомляє про некоректну команду. Важливо, що функція `handleMessage` не лише генерує відповіді, а й реалізує логіку, що дозволяє забезпечити динамічну взаємодію з користувачем, залежно від його вибору (рисунок 4.10).

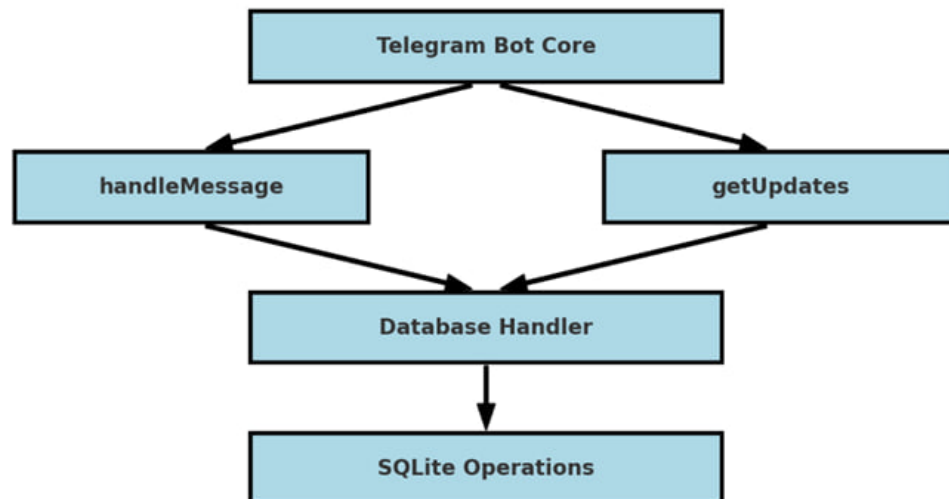


Рисунок 4.10 – Діаграма програмних компонентів

Усі ці функції працюють в межах основного циклу, що періодично отримує оновлення від Телеграм через метод `getUpdates`. Цей цикл забезпечує безперервну роботу бота, дозволяючи оперативно реагувати на дії користувачів

Функціональність Телеграм-бота реалізована через взаємодію ключових функцій, які забезпечують обробку користувацьких запитів, роботу з базою даних та інтеграцію з Telegram API. Центальною є функція `handleMessage`, яка відповідає за обробку вхідних повідомлень: вона аналізує їхній вміст, виконує відповідні дії та генерує відповіді на основі даних із бази. Ця функція об'єднує маршрутизацію команд, створення інтерактивного інтерфейсу та пошук необхідної інформації.

Безперервна робота бота забезпечується основним циклом, що використовує метод `getUpdates` для постійного опитування серверів Телеграм у реальному часі. Взаємодія з базою даних організована за допомогою SQL-запитів, які дозволяють динамічно зберігати та отримувати дані.

Усі функції розроблені для досягнення максимальної ефективності, що гарантує оперативну реакцію бота на запити користувачів і забезпечує зручність у його використанні.

4.3 Тестування вебзастосунку

Для забезпечення стабільної роботи Телеграм-бота для металургійного підприємства було проведено комплексне тестування функціональності на всіх етапах розробки та впровадження. Воно включало перевірку обробки повідомлень, взаємодії з базою даних, роботи клавіатури та загальної стійкості до помилок.

Основні етапи тестування:

- перевірка команд і клавіатури. Тестувалася команда */start* для генерації кнопок відповідно до тем, доступних у базі даних. Перевірялися коректність відображення кнопок в інтерфейсі користувача та правильність передачі даних на сервер;
- тестування запитів до бази даних. Всі SQL-запити перевірялися на відповідність очікуваним результатам. Тестувалися сценарії, коли тема відсутня, дані введені некоректно або частково. Додатково перевірено механізм унікальності записів у базі завдяки атрибуту *topic*;
- перевірка взаємодії з Telegram API. У процесі тестування циклу отримання оновлень аналізувалася коректність обробки як очікуваних, так і неочікуваних вхідних даних. Помилки API фіксувалися в логах;
- тестування форматування відповідей. Відповіді бота перевірялися на правильне відображення у форматі HTML, включаючи розділи, списки та стилі;
- тестування проводилося вручну та автоматизовано з використанням спеціалізованих бібліотек для Телеграм-ботів. Результати показали, що бот коректно обробляє запити, стійкий до некоректних даних і відповідає очікуванням користувача (рисунок 4.11).

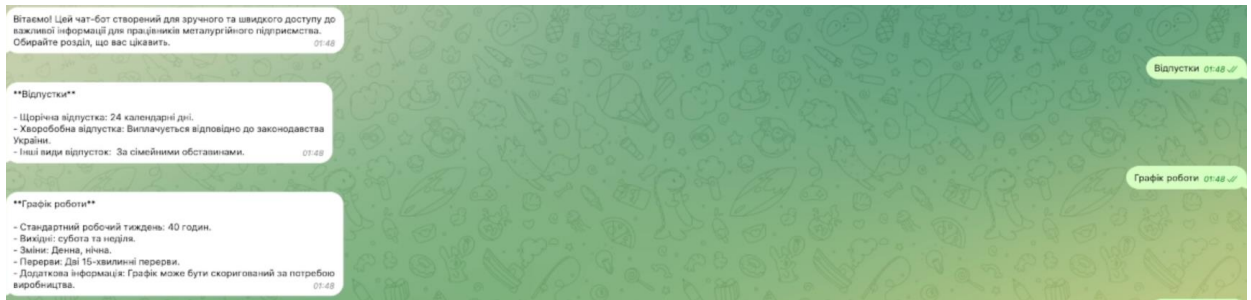


Рисунок 4.11 – Тестування функцій бота

4.4 Опис роботи вебзастосунку

Telegram-бот функціонує як єдиний програмний файл, що виконується у серверному середовищі PHP. Бот працює в режимі постійного опитування серверів Telegram за допомогою методу `getUpdates`, що забезпечує оперативну обробку повідомлень і швидку реакцію на запити.

Адміністраторам надається доступ до бази даних SQLite, що дозволяє додавати, редагувати чи видаляти інформацію за темами. Додавання нових функцій або тем здійснюється через оновлення відповідних таблиць у базі даних.

Основні вимоги для розгортання бота:

- сервер: Наявність серверного середовища PHP версії 7.4 або новішої;
- залежності: Встановлення бібліотеки Telegram Bot API через Composer;
- база даних: SQLite, яка створюється автоматично при запуску програми;
- інтернет-з'єднання: Стабільне з'єднання для роботи з Telegram API;
- функціональність бота під час роботи;
- приймає повідомлення від користувачів, аналізує текст і виконує команди;

- знаходить необхідну інформацію в базі даних, формує відповіді та надсилає їх користувачу;
- у разі помилок або некоректних запитів інформує користувача про неможливість виконання запиту.

Система проста у використанні та не потребує спеціальних знань для базового управління. За потреби бот може бути інтегрований з іншими сервісами або переведений на більш продуктивну базу даних для розширення функціоналу.

4.5 Висновки за розділом 4

У процесі розробки чат-бота було виконано низку ключових завдань, які забезпечили його функціональність та відповідність поставленим вимогам. Спершу здійснено реєстрацію бота на платформі Телеграм за допомогою @BotFather, що дозволило отримати унікальний токен для доступу до API Telegram. Це стало основою для подальшої інтеграції програмного забезпечення із сервісами Телеграм.

Наступним етапом було налаштування основних параметрів бота, зокрема його імені, опису, аватару, а також списку доступних команд. Ці дії забезпечили зручність ідентифікації бота користувачами та підвищили його інтерактивність.

Основна робота полягала у створенні програмної частини бота, яка включала налаштування взаємодії з базою даних SQLite для зберігання та отримання інформації. У межах цієї задачі було створено таблиці для організації даних та розроблено логіку SQL-запитів, які дозволяють динамічно обробляти запити користувачів.

5 ЕКОНОМІКО-ОРГАНІЗАЦІЙНА ЧАСТИНА

5.1 Розрахунок основної заробітної плати

З розвитком цифрових технологій та автоматизації процесів, металургійна галузь потребує все більше інноваційних рішень, які можуть значно покращити ефективність роботи. Одним із таких рішень є впровадження чат-бота-консультанта для співробітників підприємства. Такий чат-бот забезпечує автоматизовану допомогу в рутинних питаннях, покращує комунікацію між різними підрозділами, а також дозволяє зекономити час та зусилля працівників.

Проект розробки чат-бота для металургійного підприємства передбачає створення інструменту, який зможе надавати консультації, розв'язувати питання, що часто виникають у співробітників, та допомагати в обробці внутрішньої документації. Це дозволить зменшити навантаження на менеджерів та інших спеціалістів, дозволяючи їм зосередитись на важливіших завданнях. Опис ідеї проекту представлений у таблиці 5.1.

Таблиця 5.1 – Опис ідеї

Зміст ідеї	Напрямки застосування	Вигоди для споживачів (користувачів)
Створити чат-бота для консультацій на підприємстві	Металургійні підприємства	Зменшити навантаження на персонал, підвищити ефективність роботи
	Працівники підприємства	Скоротити час на вирішення стандартних питань
	Керівництво підприємства	Полегшити процес прийняття управлінських рішень

Як видно з таблиці, потенційний ринок для застосування такого чат-бота виявляється дуже широким. Інтерес до подібних рішень можна очікувати з боку будь-якої металургійної компанії, яка прагне підвищити продуктивність і знизити операційні витрати.

Водночас варто зауважити, що на ринку вже існують схожі рішення, і конкуренція в цій галузі досить висока. Однак, ефективність та інноваційність продукту, а також якість обслуговування користувачів можуть стати основними конкурентними перевагами. Ось попередня характеристика потенційного ринку для чат-бота (таблиця 5.2).

Таблиця 5.2 – Попередня характеристика потенційного ринку

№	Показники стану ринку (найменування)	Характеристика
1	Головні конкуренти	Різноманітні чат-боти для корпоративного використання
2	Динаміка ринку (якісна оцінка)	Зростає
3	Наявність обмежень для входу	Складність конкуренції з уже існуючими рішеннями
4	Специфічні вимоги до стандартизації та сертифікації	Вимоги до відповідності стандартам безпеки та якості

Попередня характеристика потенційних клієнтів представлена у таблиці 5.3 [47].

Таблиця 5.3 – Попередня характеристика потенційних клієнтів

№ п/ п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів (користувачів)
1	2	3	4	5
1.	Зниження навантаження на персонал	Менеджери, керівники підрозділів	Різні рівні навантаження в залежності від підрозділу	Потрібна швидка допомога в рутинних питаннях
2.	Спрощення внутрішніх процесів	Оператори	Потрібен швидкий доступ до документів та інструкцій	Легкість у використанні, швидкість доступу до інформації
3.	Підвищення ефективності	Всі співробітники підприємства	Потрібності в залежності від задачі	Інтуїтивно зрозумілий інтерфейс, доступність 24/7

Важливо зазначити, що розробка і впровадження чат-бота не є абсолютно безперешкодним процесом. Слід врахувати деякі труднощі та потенційні загрози. Детальніше це описано в таблиці 5.4 [47].

Таблиця 5.4 – SWOT-аналіз

Сильні сторони:	Слабкі сторони:
- простота у використанні; - доступ 24/7; - портативність; - швидкодія.	Складність інтеграції в існуючі системи

Продовження таблиці 5.4

Можливості:	Загрози:
- підвищення ефективності роботи - автоматизація рутинних процесів	Ризик недостатнього розуміння технології користувачами

SWOT-аналіз показує, що основними сильними сторонами продукту є його простота у використанні, доступність 24/7, портативність і висока швидкодія. Ці характеристики роблять його зручним і ефективним у роботі.

Однак існують і слабкі сторони, зокрема складність інтеграції в існуючі системи підприємства, що може потребувати додаткових зусиль і ресурсів.

Серед можливостей варто виділити автоматизацію рутинних процесів, що дозволить підвищити ефективність роботи та зменшити ризик помилок.

Основною загрозою є можливе недостатнє розуміння технології користувачами, що може уповільнити процес впровадження та знизити очікувану ефективність.

Наступним етапом є розробка ринкової стратегії. Спершу необхідно визначити стратегію охоплення ринку, зокрема описати цільові групи потенційних споживачів (таблиця 5.5) [47].

Таблиця 5.5 – Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Рівень конкуренції в сегменті	Простота входу у сегмент
1.	Оператори металургійного обладнання, технічний персонал – люди, які відповідають за контроль за обладнанням	Середня готовність, зважаючи на потребу в ефективних рішеннях для автоматизації та оптимізації виробничих процесів.	Високий попит на автоматизацію виробничих процесів, що знижує витрати та підвищує ефективність.	Висока конкуренція через наявність подібних рішень на ринку.	Висока складність, потребує значних інвестицій у технології.
2.	Постачальники сировини – компанії, що забезпечують металургійні підприємства необхідними матеріалами.	Висока готовність до співпраці для оптимізації процесів постачання та скорочення витрат.	Стабільний попит, зокрема на оптимізацію логістики та забезпечення постійного потоку сировини.	Помірна конкуренція, але є можливість для поліпшення умов співпраці.	Легкий вхід через встановлені канали постачання.
3.	Керівники виробництва, які мають потребу в рішенні для моніторингу та управління виробничими процесами.	Висока готовність до впровадження новітніх технологій, що дозволяють підвищити ефективність.	Помірний попит на інноваційні рішення для моніторингу і оптимізації виробничих процесів.	Низька конкуренція, оскільки більшість підприємств ще не повністю автоматизовані.	Простота входу залежить від технологічних вимог підприємства.

Для успішного просування цього продукту необхідно чітко визначити основних зацікавлених осіб проєкту. У випадку з розробкою програмного забезпечення для металургійного підприємства слід виокремити ключових користувачів, таких як інженери, технічний персонал, керівники виробничих процесів та інші співробітники. Саме їхні потреби та очікування повинні бути враховані під час створення і впровадження системи. [47].

5.2 Розрахунок основної заробітної плати

Витрати розраховуються з урахуванням запланованого бюджету для всіх працівників, залучених до реалізації проєкту. У розробці беруть участь програміст, який працює протягом 2 місяців, та консультант, залучений на 0,25 місяця. Детальні розрахунки наведено в таблиці 5.6.

Таблиця 5.6 – Розрахунок основної заробітної плати

Посада виконавця	Чисельність, чоловік	Місячний оклад, грн.	Кількість місяців роботи	Сума заробітної платні, грн
Програміст	1	9000	2	18000
Консультант	1	5000	0,25	1250
Підсумок	2			19250

Заробітна плата основних виробничих працівників становить 19250 грн. Витрати за цією статтею складаються з планового фонду зарплати всіх категорій працівників, зайнятих у розробці програми.

5.3 Розрахунок додаткової заробітної плати

Додаткова заробітна плата встановлюється на рівні 10% від основної заробітної плати працівників та обчислюється за формулою (5.1) [47]:

$$ЗП_{\text{дод}} = ЗП_{\text{осн}} \cdot 0,1 \quad (5.1)$$

У нашому випадку отримаємо:

$$ЗП_{\text{дод}} = 19250 \cdot 0,1 = 1925 \text{ грн}$$

5.4 Єдиний соціальний внесок

Єдиний соціальний внесок (ЄСВ) є важливим компонентом соціального забезпечення, який сплачується роботодавцями та фізичними особами-підприємцями в Україні. Цей внесок спрямований на забезпечення фінансування соціальних програм, таких як пенсійне забезпечення, страхування на випадок безробіття, тимчасової непрацездатності та інших соціальних виплат.

Єдиний соціальний внесок становить 22% і береться від основної та додаткової заробітної плати на основі формули (5.2):

$$ВТ_{\text{соц}} = (ЗП_{\text{осн}} + ЗП_{\text{дод}}) \cdot 0,22 \quad (5.2)$$

Далі отримаємо:

$$ВТ_{\text{соц}} = (19250 + 1925) \cdot 0,22 = 4658,50 \text{ грн.}$$

5.5 Визначення витрат

До накладних витрат належать усі інші витрати, зокрема витрати на канцелярські товари, утримання та обслуговування будівель тощо. Вони обчислюються у відсотковому співвідношенні до основної заробітної плати виробничого персоналу (50-100%).

Отримаємо:

$$19250 \cdot 0,5 = 9625 \text{ грн.}$$

Додаткові витрати становлять 1-2% від загальної суми попередніх витрат. На основі проведених розрахунків формується кошторис витрат на виконання робіт, наведений у таблиці 5.7.

Таблиця 5.7 – Кошторис витрат

Калькуляційні статті	Витрати	
	у розрахунку на одиницю продукції, грн.	Структура, %
1	2	3
Заробітна плата основних виробничих працівників	19250	54,2
Додаткова заробітна плата	1925	5,4
Єдиний соціальний внесок	4658,0	13,1
Накладні витрати	9625	27,1
Інші витрати	80,90	0,2
Повна собівартість	35539,40	100

Оскільки розроблене програмне забезпечення для металургійного підприємства не має комерційного характеру, підрахунок очікуваного доходу в даному випадку є неможливим, оскільки це не передбачає продажу або монетизації через традиційні фінансові канали. Однак, здійснивши попередні розрахунки витрат на розробку, а також визначивши потенційну зацікавлену аудиторію серед металургійних підприємств, можна зробити висновок про економічну вигідність проекту.

Враховуючи високий попит на автоматизацію процесів, підвищення ефективності та безпеку в галузі, чат-бот, що надає інформаційно-довідкові послуги, має великий потенціал для впровадження в поточні виробничі ланцюги. Тому, хоча прямий фінансовий ефект наразі важко оцінити, є переконливі підстави стверджувати, що цей чат-бот матиме попит на ринку, сприятиме підвищенню продуктивності та зменшенню витрат на обробку стандартних запитів, що позитивно вплине на ефективність роботи підприємства в цілому.

5.6 Висновки за розділом 5

На основі проведених розрахунків основної заробітної плати можна зробити висновок, що витрати на оплату праці працівників, залучених до розробки проекту, є оптимальними для даного етапу реалізації. Загальна сума становить 19 250 грн, що включає оплату праці програміста та консультанта. Ці витрати є базовими й відображають реальний рівень залучення фахівців до створення чат-бота.

Варто зазначити, що обґрунтований підхід до формування основної заробітної плати дозволяє уникнути надмірних витрат і водночас забезпечити необхідну мотивацію для працівників. Завдяки цьому вдалося закласти фінансову основу для подальших етапів розробки, зокрема обчислення додаткових витрат, накладних витрат і єдиного соціального внеску.

Таким чином, етап розрахунку основної заробітної плати підтвердив доцільність і реалістичність обраної фінансової стратегії, що сприяє успішному впровадженню інноваційного рішення для металургійного підприємства.

ВИСНОВКИ

У ході виконання дипломної роботи магістра були вирішені наступні завдання: проведено аналіз існуючих чат-ботів та їх особливостей, досліджено популярні платформи для розробки чат-ботів, визначено критерії оцінки функціональних можливостей сучасних конструкторів для створення чат-ботів, розроблено рекомендації щодо створення чат-бота для потреб металургійного підприємства, реалізовано програмний прототип бота-консультанта для оптимізації виробничих та інформаційних процесів.

У результаті виконаної роботи був створений чат-бот для металургійного підприємства, який виконує такі функції: надає інформацію про виробничі процеси, оперативно консультує працівників щодо технічних регламентів, забезпечує доступ до актуальних виробничих показників, а також спрощує процес комунікації між співробітниками різних відділів.

Практичне значення розробки полягає у створенні інструменту, що дозволяє підвищити ефективність обміну інформацією на підприємстві, автоматизувати рутинні процеси, а також сприяти покращенню управління технологічними процесами та персоналом.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Петров, І. І. Чат-боти в сучасному бізнесі: виклики та можливості. – Київ: Видавництво "Наука", 2021. – 200 с.
2. Сидоренко, О. О. Автоматизація комунікацій у металургійній галузі: технології та рішення. – Львів: Видавництво "Техніка", 2022. – 180 с.
3. Чат-боти та їх використання у сучасних інформаційних системах / Автори статті // Інформаційні технології та управління. – 2020. – № 1. – С. 35-41.
4. Новітні технології в сфері чат-ботів / П. І. Сидоренко // Вісник технічних наук. – 2021. – № 3. – С. 22-28.
5. Турінг А. В. Мислення машин і його перевірка / А. В. Турінг // Журнал математичних обчислень. – 1950. – № 4. – С. 433-460.
6. Вейценбаум Дж. Програма ELIZA як революція в обробці природної мови / Дж. Вейценбаум // Технології та інновації. – 1966. – № 2. – С. 12-19.
7. Колбі К. PARRY: штучний інтелект у психологічних дослідженнях / К. Колбі // Психологічний огляд. – 1972. – № 1. – С. 89-95.
8. Розробка чат-ботів: сучасні підходи та алгоритми / П. А. Ковальчук // Програмування та штучний інтелект. – 1997. – № 5. – С. 50-57.
9. Чат-бот Dr. Sbaitsa та його внесок у розвиток штучного інтелекту / О. І. Іваненко // Наукові досягнення у сфері інформаційних технологій. – 1992. – № 6. – С. 72-80.
10. IBM Watson [Electronic resource]. – Access mode: <https://www.ibm.com/watson>.
11. Google Dialogflow [Electronic resource]. – Access mode: <https://cloud.google.com/dialogflow>.
12. Messenger Developer Blog. Building Bots on [Electronic resource]. – Access mode:– Режим доступу: <https://developers.facebook.com>.
13. Telegram FAQ. How to create a Telegram Bot [Electronic resource]. – Access mode: <https://core.telegram.org/bots>.
14. Novak D. Technologies for Automated Commerce [Electronic resource]. –

Access mode: <https://example.com>.

15. Novak D. Technologies for Automated Commerce. Kyiv: 2021.
16. Smith R. AI in Customer Engagement // Business Review. 2022. № 8. С. 45-47.
17. Ivanov A. N. Educational Bots and Their Role in Modern Learning. Lviv: LNU Press, 2023.
18. O'Connor M. Privacy Concerns in AI-Driven Communication // Journal of Ethics. 2023. № 5. С. 89-92.
19. Rahman T. Chatbots and the Future of IoT // Technology Insights. 2024. № 2. С. 56-59.
20. Бондаренко І. Вплив чат-ботів на клієнтський сервіс компаній [Електронний ресурс] // Економічний вісник. – 2023. – Режим доступу: <https://www.retailweek.com/technology/chatbots-in-ecommerce-how-they-are-changing-customer-service/7037206.article>
21. Петренко О. Використання чат-ботів у роздрібній торгівлі [Електронний ресурс] // Журнал бізнес-інновацій. – 2022. – Режим доступу: [https://itc.ua/articles/yak-vprovadzhennya-chat-botiv-zminyuye obslugovuvannya-kliyentiv-v-ukrayini/](https://itc.ua/articles/yak-vprovadzhennya-chat-botiv-zminyuye-obslugovuvannya-kliyentiv-v-ukrayini/)
22. Ковальчук Л. Роль чат-ботів у сучасному бізнесі [Електронний ресурс] //Сучасний менеджмент. – 2021. – Режим доступу: <https://www.forbes.com/sites/forbestechcouncil/2022/02/28/chatbots-a-new-trend-in-customer-service/?sh=44a9c66051f4>
23. Іванова Н. Переваги автоматизованих систем підтримки клієнтів [Електронний ресурс] // Технології в бізнесі. – 2023. – Режим доступу: <https://www.businessinsider.com/chatbots-the-new-business-tool-changing-the-industry-2021-3>
24. Мартиненко А. Як Rozetka використовує чат-боти для покращення клієнтського досвіду [Електронний ресурс] // TechUkraine. – 2024. – Режим доступу: <https://www.techukraine.org/rozetka-chatbots/>
25. Андросюк, Е. "Що таке чат-боти і в чому їх перевага як каналу зв'язку"

// Prostobank. - 2020. - Режим доступу: prostobank.ua.

26. "Чат-боти: можливості і алгоритми NLP" // Avada-Media. - 2021. – Режим доступу: avada-media.com.

27. Similarweb. Most Popular Messaging Apps In 2023. [Electronic resource]. – Access mode: <https://www.similarweb.com/blog/popular-messaging-apps/>.

28. Sinch. The most popular messaging apps worldwide by country. [Electronic resource]. – Access mode: <https://engage.sinch.com>.

29. Umnico. Most Popular Messaging Apps in 2024: Features, Number of Users and Trends [Electronic resource]. – Access mode: <https://umnico.com>.

30. Statista. Number of monthly active WhatsApp users worldwide from 1st quarter 2013 to 2nd quarter 2024. [Electronic resource]. – Access mode: <https://www.statista.com/statistics/260919/number-of-monthly-active-whatsapp-users/>.

31. BusinessofApps. WeChat Revenue and Usage Statistics (2024). [Electronic resource]. – Access mode: <https://www.businessofapps.com/data/wechat-statistics/>.

32. Мейєр Е., Вейл Е. CSS: повний довідник / Е. Мейєр. - Львів: ТОВ «Діалектика», 2019. 1088 с. - ISBN 978-5-907114-56-2.

33. Ітан Маркотт Чуйний вебсайт / Маркотт І. - М.: Манн, Іванов і Фербер, 2019. 176 с. - ISBN: 978-5-91657-385-5.

34. Ши Д. Філософія CSS дизайну/Д. Ши. - Київ : НТ Прес, 2017. - 312 с. - ISBN 5-477-00122-4.

35. Дронов В.А. Розробка сучасних Web-сайтів / В.А. Дронов - Київ: 2019. - 414 с.

36. Печніков В.М. Створення Web-сторінок та Web-сайтів. Самовчитель/В.М. Пічників. - Харків.: Тріумф, 2019. - 470 с. - ISBN: 978-5-89392-143-4.

37. Берковський В.В., Радіонов В.О., Бурковський В.С. Зіставлення технологій створення сайтів з їх цільовим призначенням / В.В. Берковський // Наука та техніка Повітряних Сил Збройних Сил України. - 2017. - № 2 (19). - С. 122-124.

38. Борисов Р. І. Побудуйте професійний сайт самі / Р. І. Борисов. - Київ:

2019. 304 с.

39. Ажгіхіна С.Г., Марченко М.М. Проектування фірмового стилю / С. Г. Ажгіхіна. -Київ : НТ Прес, 2015. 312 с. - ISBN 5-477-00122-4.

40. Ітан Маркотт. Чуйний веб-дизайн / Маркотт І. - М.: Манн, Іванов та Фербер, 2018. 176 с.

41. Пітер Фіелл. Енциклопедія дизайну/Ф. Харків.: «Аст-Астрель», 2021 321 с.

42. Гладкий А.В. Веб-Самоделкін. Як самому створити сайт швидко та професійно / А.В. Гладкий. - Київ: 2018. 414

43. Петрова О. А. Реляційні бази даних та їх застосування: навч. посіб. — Харків: ХНУ, 2021. — 345 с.

44. Сидоренко В. П. MongoDB: Архітектура та застосування. — Львів: ЛДУ, 2023. — 278 с.

45. Іванов І. І. Основи розробки вебзастосунків з використанням MySQL. — Київ: Наукова думка, 2022. — 410 с.

46. Habr [Електрон. ресурс]. — Режим доступа: <https://habr.com/ru/post/262247>.

47. Методичні вказівки по економічному обґрунтуванню дипломних проектів для студентів спеціальностей 8.05010301 «Програмне забезпечення автоматизованих систем» та 8.05010201 «Інформаційні технології проектування» для усіх форм навчання [Текст] / В.В. Остапенко, Є.М. Сердюк. — Запоріжжя : ЗНТУ, 2015. — 22 с.

ДОДАТОК А
Технічне завдання

Вступ

Сучасна автоматизація бізнес-процесів є ключем до підвищення ефективності металургійних підприємств. Використання чат-ботів стає актуальним інструментом, що забезпечує оперативне вирішення питань, надання консультацій і підвищення рівня обслуговування клієнтів та співробітників.

Метою роботи є розробка бота-консультанта, що надає інформацію про виробничі процеси, технічні довідки та інструкції з безпеки. Це сприятиме зниженню навантаження на персонал і підвищенню продуктивності праці.

А.3 Основні вимоги до програми

А.3.1 Функціональні характеристики

Чат-бот має:

- надавати технічну інформацію;
- шукати дані в базі знань;
- надавати інструкції з охорони праці;
- інформувати про розклад і контакти;
- генерувати звіти за запитом.

А.3.2 Надійність

Надійність забезпечується стабільною роботою серверів, мережі Інтернет і резервним копіюванням даних.

А.3.3 Умови експлуатації

Програма працює на Windows або Linux із Python 3.7+, вебсервером Nginx або Apache та СУБД MySQL або PostgreSQL.

A.3.4 Технічні засоби

Потрібні:

- процесор 2 ГГц;
- озп 2 Гб;
- дисковий простір 500 Мб;
- інтернет 1 Мб/с.

A.3.5 Транспортування та збереження

Програмне забезпечення транспортується та зберігається згідно з правилами поводження з цифровими носіями. Резервне копіювання здійснюється автоматично.

A.3.6 Програмна документація

Документація включає технічне завдання, інструкцію з експлуатації та опис коду програми.

A.3.7 Маркування та упакування

Носії мають містити назву проєкту "Розробка бота-консультанта для металургійного підприємства" і версію ПЗ. Дані упаковані в архів із документами та файлами.

ДОДАТОК Б
Текст програми

```

<?php

require __DIR__ . '/vendor/autoload.php';

use Telegram\Bot\Api;
use Telegram\Bot\Keyboard\Keyboard;

// --- Database Connection (PDO) ---
try {
    $db = new PDO('sqlite:metallurgy_info.db');
    $db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $db->exec("CREATE TABLE IF NOT EXISTS info (
        id INTEGER PRIMARY KEY AUTOINCREMENT,
        topic TEXT UNIQUE,
        information TEXT
    )");
} catch (PDOException $e) {
    die("Помилка бази даних: " . $e->getMessage());
}

// --- Initial Information (Ukrainian) ---
$metallurgyInfo = [
    ['topic' => 'Правила техніки безпеки', 'information' => "***Правила техніки безпеки на металургійному підприємстві**\n\n Завжди використовуйте засоби індивідуального захисту (ЗІЗ).\n- Дотримуйтесь правил пожежної безпеки.\n- Не працюйте з несправним обладнанням.\n- Повідомляйте про будь-які небезпечні ситуації.\n- Регулярно проходите навчання з техніки безпеки."],
    ['topic' => 'Графік роботи', 'information' => "***Графік роботи**\n\n Стандартний робочий тиждень: 40 годин.\n- Вихідні: субота та неділя.\n- Зміни: Денна, нічна.\n- Перерви: Дві 15-хвилинні перерви.\n- Додаткова інформація: Графік може бути скоригований за потребою виробництва."],
    ['topic' => 'Контакти', 'information' => "***Контакти**\n\n Служба технічної підтримки: +380-50-123-45-67\n- Відділ кадрів: +380-50-987-65-43\n- Адміністрація: +380-50-111-22-33"],
    ['topic' => 'Оплата праці', 'information' => "***Оплата праці**\n\n Заробітна плата виплачується щомісяця.\n- Додаткові виплати: Премії за ефективність роботи.\n- Інформація про нарахування зарплати: Зверніться до відділу кадрів."],
    ['topic' => 'Відпустки', 'information' => "***Відпустки**\n\n Щорічна відпустка: 24 календарні дні.\n- Хворобобна відпустка: Виплачується відповідно до законодавства України.\n- Інші види відпусток: За сімейними обставинами."],
    ['topic' => 'Соціальний пакет', 'information' => "***Соціальний пакет**\n\n Медичне страхування: Надається базове медичне страхування.\n- Додаткові пільги: Компенсація за харчування.\n- Програми соціального захисту: Програми допомоги працівникам."],
];

```

```
// --- Update Database with Metallurgy Information ---
try {
    $db->exec('DELETE FROM info');

    $stmt = $db->prepare("INSERT INTO info (topic, information) VALUES (?, ?)");
    foreach ($metallurgyInfo as $info) {
        $stmt->execute([$info['topic'], $info['information']]);
    }
} catch (PDOException $e) {
    die("Помилка оновлення бази даних: " . $e->getMessage());
}

// --- Telegram Bot ---
$telegram = new Api('7743347924:AAGnU8WXHXjchEpjGLqcW32f16Z6UC6u82I');
$lastUpdateId = 0;

while (true) {
    try {
        $updates = $telegram->getUpdates(['offset' => $lastUpdateId + 1]);
        foreach ($updates as $update) {
            if ($update->getMessage()) {
                handleMessage($update, $telegram, $db);
            }
            $lastUpdateId = $update->getUpdateId();
        }
    } catch (Telegram\Bot\Exceptions\TelegramSDKException $e) {
        error_log("Помилка Telegram API: " . $e->getMessage());
    } catch (PDOException $e) {
        error_log("Помилка бази даних: " . $e->getMessage());
    } catch (Exception $e) {
        error_log("Інша помилка: " . $e->getMessage());
    }
    sleep(1);
}

function handleMessage($update, $telegram, $db)
{
    $message = $update->getMessage();
    $chatId = $message->getChat()->getId();
    $text = $message->getText();

    if ($text === '/start') {
        $stmt = $db->query('SELECT topic FROM info');
        $topics = $stmt->fetchAll(PDO::FETCH_COLUMN);
    }
}
```

```

$keyboard = Keyboard::make()
->row(array_map(function ($topic) {
    return Keyboard::button(['text' => $topic, 'request_contact' => false]);
}, $topics));

$telegram->sendMessage([
    'chat_id' => $chatId,
    'text' => 'Вітаємо! Цей чат-бот створений для зручного та швидкого
доступу до важливої інформації для працівників металургійного підприємства.
Обирайте розділ, що вас цікавить.',
    'reply_markup' => $keyboard,
]);
} else {
    $stmt = $db->prepare("SELECT information FROM info WHERE topic = ?");
    $stmt->execute([$text]);
    $information = $stmt->fetchColumn();

    if ($information) {
        $telegram->sendMessage([
            'chat_id' => $chatId,
            'text' => $information,
            'parse_mode' => 'HTML',
        ]);
    } else {
        $telegram->sendMessage([
            'chat_id' => $chatId,
            'text' => 'Невідома команда.',
        ]);
    }
}
}

$db->close();

?>

```

ДОДАТОК В
Слайди презентації

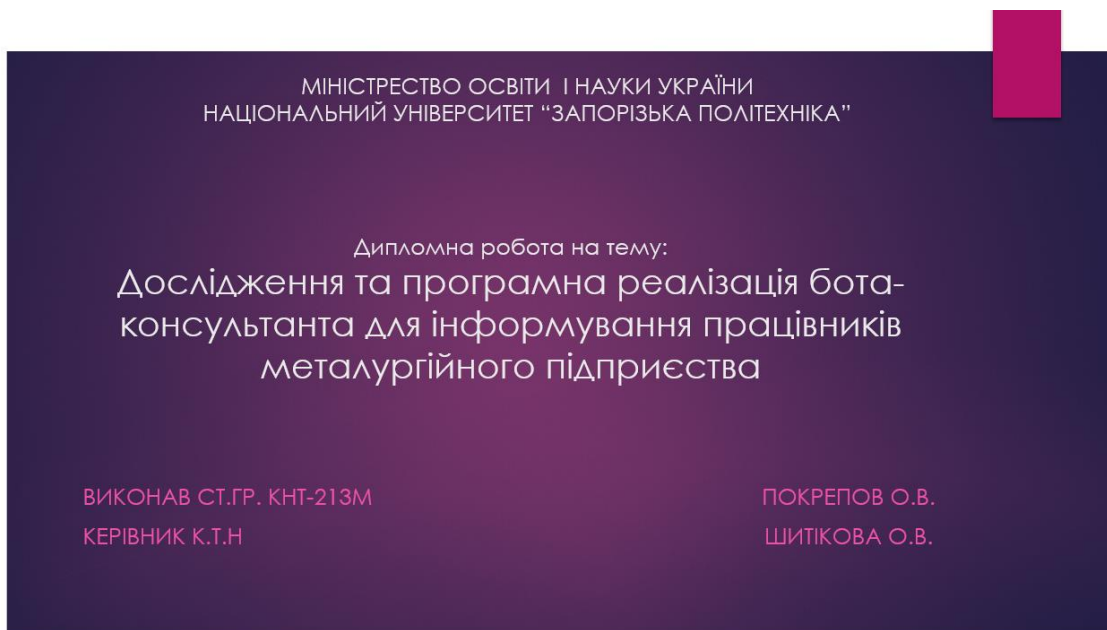


Рисунок В.1 – Слайд 1



Рисунок В.2 – Слайд 2

Аналіз типів ботів

Тип бота	Галузі застосування
Бот як асистент або фахівець технічної підтримки	Технічна підтримка, навчання та наставництво, збір зворотного зв'язку
Бот як інтерфейс для віддаленого управління	Системи розумного будинку, індустрія IoT, управління бізнес-процесами
Транзакційні чат-боти	Фінансові послуги, електронна комерція, замовлення послуг
Інформаційні чат-боти	Новинні агрегатори, підтримка користувачів

Рисунок В.3 – Слайд 3

Огляд месенджерів



Процент щомісячних користувачів

Джерело: <https://www.similarweb.com/blog/popular-messaging-apps/>.

Рисунок В.4 – Слайд 4

Порівняння популярних месенджерів

1	2	Фейсбук Messenger	Telegram Messenger	Вайбер	Ватсап
Загальна інформація	Доступність на ПК	+	+	+	+
	Безкоштовне використання	+	+	+	+
	Наявність реклами	+	+	-	+
	Можливість персоналізації	+	+	-	+
	Сумісність з Android	+	+	+	+
Користувачі	Рейтинг в Google Play	4.2	4.4	4.4	4.3
	Щомісячні користувачі	130 млн	500 млн	823 млн	1500 млн
Контакти	Рейтинг в App Store компанії Apple	4.5	4	4.5	4
	Використовує функцію "двосторонньої відмови"	-	+	-	-

Рисунок В.5 – Слайд 5

Загальна архітектура взаємодії

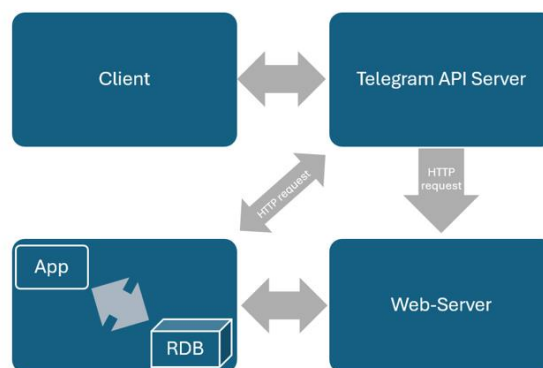


Рисунок В.6 – Слайд 6

Схема компонентів застосунку



Рисунок В.7 – Слайд 7

Послідовність реєстрації бота

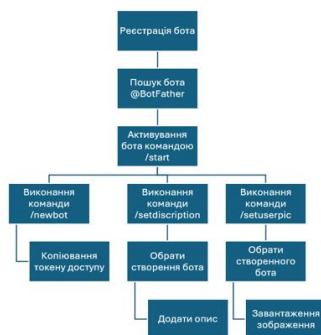


Рисунок В.8 – Слайд 8

Зовнішній вигляд розробленого бота

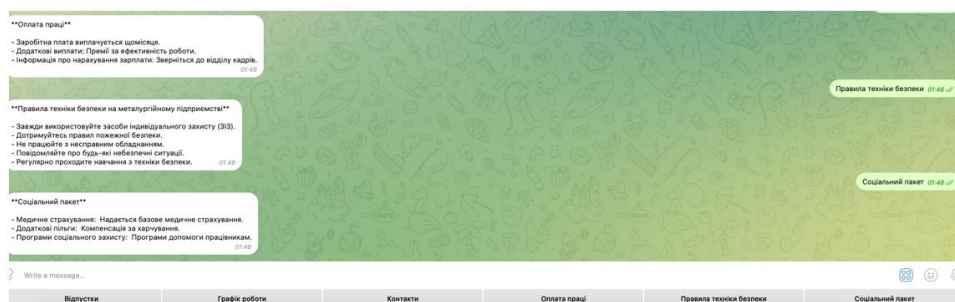
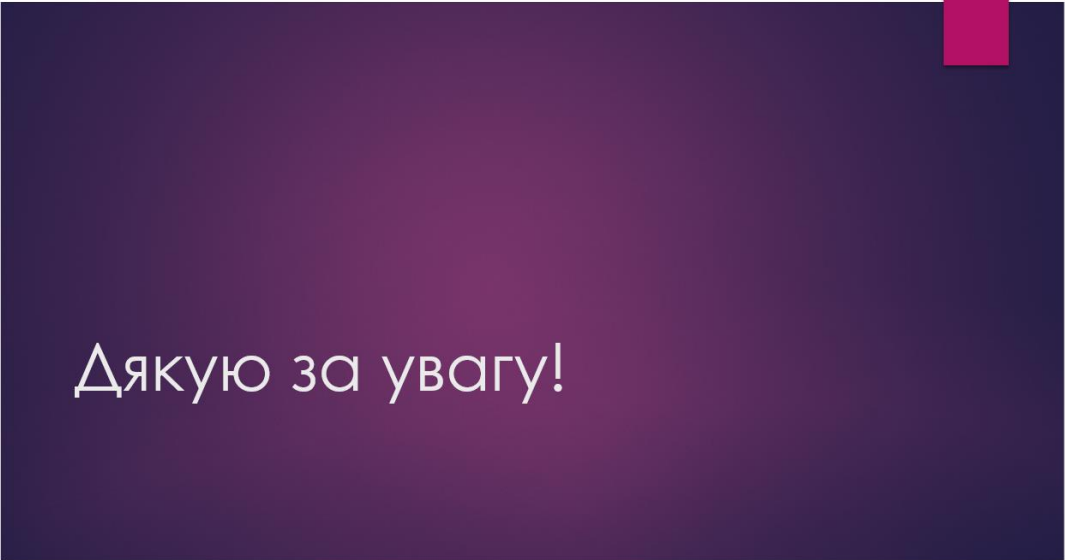


Рисунок В.9 – Слайд 9

Висновки

- У ході виконання дипломної роботи магістра були вирішені наступні завдання: проведено аналіз існуючих чат-ботів та їх особливостей, досліджено популярні платформи для розробки чат-ботів, визначено критерії оцінки функціональних можливостей сучасних конструкторів для створення чат-ботів, розроблено рекомендації щодо створення чат-бота для потреб металургійного підприємства, реалізовано програмний прототип бота-консультанта для оптимізації виробничих та інформаційних процесів.
- У результаті виконаної роботи був створений чат-бот для металургійного підприємства, який виконує такі функції: надає інформацію про виробничі процеси, оперативно консультує працівників щодо технічних регламентів, забезпечує доступ до актуальних виробничих показників, а також спрощує процес комунікації між співробітниками різних відділів.

Рисунок В.10 – Слайд 10



Дякую за увагу!

Рисунок В.11 – Слайд 11