

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний університет «Запорізька політехніка»

Інститут інформатики та радіoeлектроніки,  
Факультет комп'ютерних наук і технологій

(повне найменування інституту, назва факультету)

Кафедра програмних засобів

(повне найменування кафедри )

**Пояснювальна записка**

до дипломного проекту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ЗАПИТІВ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ЧАТБОТУ  
TAKSI  
RESEARCH OF REQUESTS AND SOFTWARE IMPLEMENTATION OF A TAXI  
CHATBOT

Виконав: студент(ка) 2 курсу, групи КНТ-110м  
Спеціальності 121 Інженерія програмного  
забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

Назаренко Н.В.

(прізвище та ініціали)

Керівник Касьян М.М.

(прізвище та ініціали)

Рецензент Поляков М.О.

(прізвище та ініціали)

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

Інститут, факультет ПРЕ, ФКНТ  
 Кафедра програмних засобів  
 Ступінь вищої освіти магістр  
 Спеціальність 121 Інженерія програмного забезпечення  
 (код і найменування)  
 Освітня програма (спеціалізація) Інженерія програмного забезпечення  
 (назва освітньої програми (спеціалізації))

**ЗАТВЕРДЖУЮ**

Завідувач кафедри ПЗ, д.т.н, проф.  
С.О. Субботін  
 “ ” 20\_\_ року

**З А В Д А Н Н Я**

**НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)**

Назаренко Нікіти Вадимовича

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Дослідження запитів та програмна реалізація чатботу таксі. Research of Requests and Software Implementation of a Taxi Chatbot  
 керівник проекту (роботи) Касьян Микола Миколайович, к.т.н., доцент,  
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “05” листопада 2021 року № 435

2. Строк подання студентом проекту (роботи) 1 грудня 2021 року

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

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Дослідження методів роботи чатботів. 3. Розробка архітектури чатботу. 4. Основні рішення щодо реалізації компонентів чатботу. 5. Тестування чатботу. Додатки.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Слайди презентації

## 6. Консультанти розділів проєкту (роботи)

| Розділ              | Прізвище, ініціали та посада консультанта | Підпис, дата   |                           |
|---------------------|-------------------------------------------|----------------|---------------------------|
|                     |                                           | завдання видав | прийняв виконане завдання |
| 1-5 Основна частина | Касьян М.М., доцент                       |                |                           |
| Нормоконтролер      | Белова А.В., асистент                     |                |                           |
|                     |                                           |                |                           |
|                     |                                           |                |                           |

7. Дата видачі завдання «09» вересня 2021 року.

## КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів дипломного проєкту (роботи)                                  | Строк виконання етапів проєкту (роботи) | Примітки     |
|-------|---------------------------------------------------------------------------|-----------------------------------------|--------------|
| 1     | Постановка завдання роботи.                                               | 1 тиждень                               | Завдання, ТЗ |
| 2     | Аналіз предметної області.                                                | 2 тиждень                               | Розділ 1     |
| 3     | Розробка та удосконалення методів, моделей й алгоритмів вирішення задачі. | 3-4 тижні                               | Розділ 2     |
| 4     | Розробка архітектури чатботу.                                             | 5 тиждень                               | Розділ 3     |
| 5     | Розробка чатботу.                                                         | 6-7 тижні                               | Розділ 4     |
| 6     | Тестування запитів до чатботу.                                            | 8 тиждень                               | Розділ 5     |
| 7     | Оформлення пояснювальної записки та документів до неї.                    | 9-10 тижні                              | Додатки      |
| 8     | Нормоконтроль та рецензування.                                            | 11-12 тижні                             |              |
| 9     | Захист роботи.                                                            | 13 тиждень                              |              |
|       |                                                                           |                                         |              |
|       |                                                                           |                                         |              |

Студент (ка) \_\_\_\_\_ Назаренко Н.В.  
(підпис) (прізвище та ініціали)

Керівник проєкту (роботи) \_\_\_\_\_ Касьян М.М.  
(підпис) (прізвище та ініціали)

## РЕФЕРАТ

Пояснювальна записка до кваліфікаційної роботи магістра: 75 с., 11 табл., 46 рис., 2 дод., 31 джерело.

АНАЛІЗ, БАЗА ДАНИХ, БОТ, ІНТЕРФЕЙС, МЕСЕНДЖЕР, МОБІЛЬНИЙ ДОДАТОК, ТАКСІ, PYTHON, TELEGRAM, UML.

Об'єкт дослідження – чатбот.

Предмет дослідження – телеграм-бот таксі.

Мета роботи – провести дослідження запитів створення чатботу та розробка чатботу для замовлення таксі.

Матеріали, методи та технічні засоби: мова моделювання UML, мова програмування Python, база даних SQLite3, static maps, geocoder, сервер Heroku.

Результати. На основі аналізу предметної області та дослідження запитів до створення чатботів розроблено програмний додаток, призначений для підвищення продуктивності роботи підприємства.

Висновки. Розроблена система дозволяє автоматизувати роботу таксі.

Галузь використання – перевезення пасажирів.

## **ABSTRACT**

Explanatory note to the diploma qualifying work of the master: 75 pages, 11 tables, 46 figures, 2 appendices, 31 sources.

ANALYSIS, DATABASE, BOT, INTERFACE, MESSAGE, MOBILE APP, TAXI, PYTHON, TELEGRAM, UML.

The object of research is a chatbot.

The subject of research - a telegram-bot taxi.

The purpose of the work is to conduct research on requests to create a chatbot and develop a chatbot to order a taxi.

Materials, methods and technical means: UML modeling language, Python programming language, SQLite3, static maps, geocoder, Heroku server.

Results. Based on the analysis of the subject area and the study of requests to create chatbots developed a software application designed to increase the productivity of the enterprise.

Conclusions. The developed system allows to automate the work of taxis.

Field of use - transportation of passengers.

## ЗМІСТ

|                                                             |    |
|-------------------------------------------------------------|----|
|                                                             | С. |
| Перелік скорочень та умовних позначок .....                 | 8  |
| Вступ.....                                                  | 9  |
| 1 Аналіз предметної області.....                            | 11 |
| 1.1 Поняття та класифікація ботів .....                     | 11 |
| 1.2 Аналіз сучасних месенджерів .....                       | 13 |
| 1.3 Переваги використання чатботу .....                     | 17 |
| 1.4 Висновки до розділу .....                               | 21 |
| 2 Дослідження запитів чатботів.....                         | 22 |
| 2.1 Методи розробки чатботу .....                           | 22 |
| 2.2 Порівняння чатботів з мобільними додатками.....         | 25 |
| 2.3 Висновки до розділу .....                               | 28 |
| 3 Розробка архітектури чатботу.....                         | 29 |
| 3.1 Структура чатботу .....                                 | 29 |
| 3.2 Функціональні вимоги до чат-боту .....                  | 30 |
| 3.3 Моделювання чатботу.....                                | 31 |
| 3.4 Висновки до розділу .....                               | 34 |
| 4 Основні рішення щодо реалізації компонентів чат-боту..... | 35 |
| 4.1 База даних .....                                        | 35 |
| 4.2 Вибір мови програмування та налаштування бібліотек..... | 39 |
| 4.3 Реєстрація бота в Telegram .....                        | 43 |
| 4.4 Висновки до розділу .....                               | 47 |
| 5 Тестування чатботу .....                                  | 48 |
| 5.1 Розміщення чатбота на віддаленому сервері.....          | 48 |
| 5.2 Тестування чат-боту .....                               | 49 |
| 5.3 Висновки до розділу .....                               | 53 |

|                                    |    |
|------------------------------------|----|
|                                    | 7  |
| Висновки .....                     | 54 |
| Перелік джерел посилання .....     | 56 |
| Додаток А Програмний код.....      | 59 |
| Додаток Б Слайди презентації ..... | 68 |

**ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК**

|       |   |                                     |
|-------|---|-------------------------------------|
| API   | – | application programming interface;  |
| JSON  | – | JavaScript Object Notation;         |
| HTTP  | – | HyperText Transfer Protocol;        |
| HTTPS | – | HyperText Transfer Protocol Secure; |
| NLG   | – | Natural Language Generating;        |
| NLP   | – | Natural Language Processing         |
| SQL   | – | Structured Query language;          |
| XML   | – | Extensible Markup Language          |
| IT    | – | інформаційні технології;            |
| БД    | – | база даних;                         |
| ОС    | – | операційна система;                 |
| ПЗ    | – | програмне забезпечення.             |

## ВСТУП

Розвиток технологій у світі сприяє швидкому обміну інформацією та появі нових каналів комунікації. Для взаємодії зі споживачами компаніям доводиться поступово переходити з традиційних каналів комунікації, таких як телебачення, радіо та друковані канали інформації у цифрові [1]. Це пов'язано з тим, що клієнту важливо бути на зв'язку у зручний для нього час та отримувати інформацію зручним для нього способом. Саме тому компаніям і брендам зараз важливо дотримуватися переваг своїх споживачів, щоб вести з ними комунікацію і своєчасно доносити інформацію про свій продукт або послугу, застосовувати маркетингові інструменти.

Одним з каналів комунікації, що розвиваються для зв'язку зі споживачем на сьогоднішній день є чатбот. Це зумовлено тенденцією у сучасній комунікації: переходом аудиторії до месенджерів. За даними дослідження Online Market Intelligence [2], месенджери є найбільш популярним каналом комунікації. У 83% власників смартфонів встановлений хоча б один месенджер. Так, у зв'язку зі зростанням популярності месенджерів, багато брендів стали використовувати на їхній платформі чатботи як новий канал комунікації.

Зрозумівши потенціал цих мобільних додатків, більшість компаній почали використовувати месенджери в якості допоміжного інструменту для роботи підприємства. Адже смартфони цілодобово знаходяться поряд з клієнтом на відстані витягнутої руки, а це означає, що повідомлення будуть помічені майже зі 100% ймовірністю.

До переваг месенджерів можна віднести: охоплення більшого відсотку користувачів, можливість ділитися різним типом контенту (від великих текстових файлів, зображень, відеофайлів до геолокації) та максимальну швидкість реакції на повідомлення від користувачів (завдяки push-повідомленням) [3].

Також головними перевагами месенджерів є можливість завжди залишатися на зв'язку, забезпечуючи цілодобову підтримку та найвищого рівня приватності у вирішенні проблем користувача. В літературі розглядається визначення бот як спеціальна програма для виконання функцій запрограмованого алгоритму. Це своєрідні симулятори дій живого персоналу підприємства, іноді вони настільки складні, що їх неможливо відрізнити від реального користувача [4]. При отриманні запиту від клієнта програми працюють за попередньо розробленим сценарієм, правила якого не змінюються. Розумний бот призначений для виконання монотонної та рекурсивної роботи.

Мета даної магістерської роботи – розробка чатботу для замовлення таксі.

Реалізація поставленої мети передбачає вирішення наступних завдань:

- провести аналіз месенджерів та сучасних чатботів таксі;
- дослідити методи розробки чатботів;
- дослідити переваги використання чатботів ніж мобільних додатків;
- розробити концептуальну модель застосунку;
- провести аналіз існуючого інструментарію проєктування чатботів;
- розробити програмне забезпечення чатботу;
- проаналізувати результати запитів.

Предметом дослідження дипломної кваліфікаційної роботи магістра є розробка бота у месенджері.

Практична цінність роботи полягає в тому, що на підставі досліджень розроблено програмне забезпечення для підвищення працездатності обробки замовлень таксі.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Поняття та класифікація ботів

Бот – це програмний додаток, у якому запрограмовано ряд функцій до виконання певних завдань. Боти автоматизовані – це означає, що вони працюють відповідно до своїх інструкцій, що не потребують безпосереднього втручання людини в їхню роботу. Зазвичай вони виконують систематичні завдання і працюють набагато швидше, ніж люди [5].

Боти найчастіше працюють по мережі. Більше половини інтернет-трафіку – це боти, які сканують вміст вебсторінок, спілкуються з людьми або шукають нові цілі для атак на користувачів. Деякі боти корисні. Наприклад, боти пошукових систем, що індексують контент для пошуку корисної інформації, або боти для обслуговування клієнтів, які допомагають користувачам у вирішенні їх низки питань. Інші боти є "поганими". Вони запрограмовані, щоб зламувати облікові записи користувачів, щоб сканувати вебсторінки у пошуках контактної інформації для розсилки спаму або виконувати інші шкідливі дії [6].

Боти бувають чотирьох видів:

- чатботи - боти, які імітують людську бесіду, відповідаючи на певні фрази вже запрограмованими відповідями;
- сканерботи - боти, які сканують контент на веб-сторінках по всьому інтернету;
- соціальні медіаботи - боти, що працюють на платформах соціальних мереж;
- шкідливі боти – боти, які сканують контент, розповсюджують спам-інформацію, а також можуть виконувати атаки на базу даних облікових записів користувачів [2].

Чатботи класифікують за алгоритмом, видом та функціональністю.

За алгоритмом:

- обмежений. Такий чатбот взаємодіє з відвідувачами за заздалегідь підготовленим скриптом, що складається з найпоширеніших питань;
- саморозвинений («розумний») бот. В основі цієї програми – штучна нейронна мережа, яка розуміє сенс розмови. Розмова з таким чатботом реалістичніша, він здатний жартувати і відправляти співрозмовнику смішні картинки, а релевантність його відповідей з часом зростає [3].

Чатботи за видом:

- кнопковий. Для спілкування з ботом користувач використовує встановлені кнопки. Більшість обмежених чатботів працює саме за таким принципом;
- текстовий. Чатбот розпізнає слова, що часто зустрічаються в запиті відвідувача, уточнює питання і пропонує рішення [6].

За функціоналом:

- комунікаційні. Їх застосовують виключно для спілкування – щоб відповісти на запитання, поділитися інформацією про спеціальні пропозиції та знижки, допомогти підібрати товар чи послугу, тощо;
- функціональні. Такі чат-боти дають можливість одразу виконати ті чи інші дії – наприклад, переказати гроші на рахунок, уточнити статус замовлення за його номером, тощо [7].

Чатботи допомагають:

- оптимізувати роботу. Бот відповідає на поширені запитання відвідувачів сайту, перекладаючи їх на фахівців лише для вирішення нестандартних завдань;
- скоротити тимчасові трудовитрати. Одна програма, яка вирішує більшу частину питань дає змогу скоротити до мінімуму штат співробітників, які приймають дзвінки. Інтелектуальна система може практично повністю замінити

операторів технічної підтримки та call-центру, надаючи користувачам довідкову інформацію та відповіді на популярні запитання;

- організувати цілодобову взаємодію із клієнтами;

- підвищити рівень продажів. Все більше чатботів, створених для конкретних компаній, здатні відповідати на запитання відвідувачів та пропонувати можливість безпечно сплатити замовлення, не залишаючи діалогове вікно;

- персоналізувати спілкування з клієнтом. Кожен потенційний покупець (лід), котрий відвідав ваш інтернет-ресурс, розраховує на індивідуальний підхід. Чат-бот здатний аналізувати особисті дані відвідувача та його запити в мережі;

- виключити людський чинник. Чат-бот не страждає "поганим" настроєм або відсутністю досвіду у вирішенні задач. Його відповіді завжди мають задану тональності та містять конкретні грамотні вирази. Особливістю інтелектуальних систем є здатність самостійно навчатися та вдосконалювати прийоми спілкування на базі проведених діалогів;

- аналізувати та систематизувати результати. За запитом система забезпечить вас аналітичною інформацією та цифрами про результати виконаної роботи, систематизує процес продажу.

## **1.2 Аналіз сучасних месенджерів**

Переваги використання месенджерів для налагодження спілкування з клієнтом [8]:

- 85% мобільних клієнтів користуються послугами месенджерів щомісяця;

- 15 днів на місяць середній мобільний користувач проводить у месенджерах та соціальних мережах;

- 32% мобільного трафіку займає обмін миттєвими повідомленнями;
- 49% користувачів чату – це зрілі люди віком від 35 до 64 років.

Хочеться звернути увагу на те, що месенджери вже випередили соціальні мережі за першими двома показниками.

За даними онлайн-опитування, найпопулярнішим і використовуваним є Viber, яким користуються 83% респондентів (рис. 1.1) [9].



Рисунок 1.1 – Популярність месенджерів [9]

На другому місці Facebook Messenger – 54%. На третьому місці Skype – 44%, найбільше користуються люди старшого віку (45-49 років) – 53%. 19% користуються додатком Telegram, він найбільш популярний серед респондентів 16-29 років. Месенджер WhatsApp – 18%. Але лише 2% користуються новим додатком від Київстар – Veon [9].

Розглянемо детальніше кожен месенджер.

Facebook Messenger – це чатбот платформа з аудиторією понад 1,2 мільярда активних користувачів на місяць та понад 100 тисяч активних ботів по всьому світу. Провідна платформа для служб обміну миттєвими повідомленнями в США. Набір функцій є найбільш розвиненим для розробників: широкий

спектр елементів взаємодії (шаблони списку продуктів, розширення, елементи управління, шаблони реєстрації), автентифікація, прийом платежів, надсилання сервісних повідомлень за номером телефону [10].

Viber – в цьому месенджері облікові записи для бізнесу стали доступні зовсім недавно. Компанії можуть встановити діалог із передплатниками свого облікового запису через бота або CRM.

Telegram – одним із перших відкрив можливість створення ботів у месенджері. Загалом Telegram створив близько 100 тисяч ботів [10].

Для середнього та великого бізнесу зробили WhatsApp Business API – прямий доступ до API месенджера, що дозволяє реалізовувати служби підтримки у месенджері, ботах тощо.

Розглянемо, чим відрізняється створення чатботів у Telegram, WhatsApp, Viber, Facebook.

Чатбот в Telegram. Основною перевагою цієї програми для обміну повідомленнями є його API. API відкритий для всіх та пропонує більше можливостей і дозволяє реалізувати всі функціональні вимоги замовника [11].

Як і у випадку з Telegram, Viber не має обмежень API, на відміну від WhatsApp. Viber пропонує деякі унікальні функції, яких немає в інших месенджерів. Наприклад, можна повністю розробити власний дизайн кнопок, фону та ін [12].

Facebook Messenger налічує найбільшу кількість активних ботів. Facebook Messenger має широкий спектр інструментів та функцій для розробки. Крім того, обмеження на надсилання повідомлень у Facebook Messenger можуть розглядатися рекламодавцями як незручність [13].

Чат-бот у WhatsApp. API для цього месенджера є платним. Крім того, чатботи в цьому додатку засновані на принципі надсилання повідомлень. Фактично, це не чатбот, а платформа для автоматичних повідомлень та миттєвих привітань. Все це робить WhatsApp найменш популярною платформою для

запуску чатбота [13].

Надалі розглянемо наступні технічні параметри для аналізу: форма використання, доступи, інтерфейси, механізми зв'язку, протоколи передачі даних, формати передачі даних. У таблиці 1.1 наведено порівняльну інформацію про популярні месенджери з підтримкою платформи чатбота.

Таблиця 1.1 – Порівняння популярних месенджерів для чатботів

| Параметри                | Facebook                                                       | Telegram                                                       | Skype                                      | Viber                             |
|--------------------------|----------------------------------------------------------------|----------------------------------------------------------------|--------------------------------------------|-----------------------------------|
| Форма використання       | Умовно безкоштовна                                             | Безкоштовна                                                    | Умовно безкоштовна                         | Умовно безкоштовна                |
| Доступ до чатботу        | Доданий як новий користувач, за підпискою, вбудований в діалог | Доданий як новий користувач, за підпискою, вбудований в діалог | Доданий як новий користувач, за підпискою, | За підпискою, вбудований в діалог |
| Протоколи передачі даних | https                                                          | https                                                          | https                                      | https                             |
| Інтерфейс користувача    | Текстовий, кнопочний, голосовий                                | Текстовий, кнопочний, голосовий                                | Текстовий, кнопочний, голосовий            | Текстовий, кнопочний, голосовий   |
| Механізм зв'язку         | webhook                                                        | Webhook, polling                                               | webhook                                    | webhook                           |
| Формати передачі даних   | JSON<br>Multipart/form-data                                    | JSON, URL, query<br>Multipart/form-data                        | JSON<br>Multipart/form-data                | JSON<br>Multipart/form-data       |

За розглянутими характеристиками можна зробити висновок, що зараз існує два месенджера, що мають багато функцій і можливостей, це Telegram і Facebook Messenger.

За результатами аналізу робимо висновок, що для досягнення цілей магістерської роботи доцільно використовувати месенджер Telegram, оскільки він порівняно з іншими месенджерами, має такі переваги:

- має повністю безкоштовну форму використання;
- підтримує комунікаційний механізм Polling, що значно спрощує розробку та налаштування системи на початкових етапах;
- підтримує повний перелік функціональних можливостей, які дозволять реалізувати всі вимоги системи, а при необхідності покращити та розширити їх.

### **1.3 Переваги використання чатботу**

220 тис. залучених у перевезення водіїв і 280 млн поїздок на рік — так три роки тому оцінював таксоринок компанія BRDO. Сьогодні ж його обсяг уже досяг 60 млрд грн — такі дані Української таксомоторної асоціації. Водночас більше ніж 90% водіїв працюють без ліцензій [14].

Процесу зростання залучених до перевезення водіїв сприяє дедалі більша цифровізація процесу спілкування клієнтів і тих, хто постачає послуги таксі. Онлайн-агрегатори замовлення стартували в Україні в середині 2016 року, з приходом американської компанії Uber. Ця компанія і ще дві подібні, заснований у 2010-му український Uklon, а також естонський Bolt, сьогодні стали найбільшими гравцями в країні. Разом вони істотно змінили ринок, спростивши замовлення машини до кількох торкань на екрані смартфона, тим самим скоротивши загальний час очікування автомобіля. Тепер клієнти оцінюють водіїв, можуть відстежити їхній рейтинг і шлях замовленого автомобіля. Лідери активно залучають усе нових користувачів, витісняючи традиційні служби. За різними оцінками, онлайн-замовлення становлять уже 60–80% від усіх випадків звернення клієнтів по таксі у Києві [14].

Крім того, жодга з перелічених компаній не використовує технології чатботів у месенджерах.

Але у месенджерах працює безліч чатботів для вирішення різноманітних завдань, наведемо приклади деяких з них для великих українських компаній.

Для прискорення та автоматизації грошових переказів розроблений бот в Telegram (рис. 1.2).



Рисунок 1.2 – Бот «ПриватБанку»

Приватбанком розроблений бот для оплати комунальних платежів (рис. 1.3).

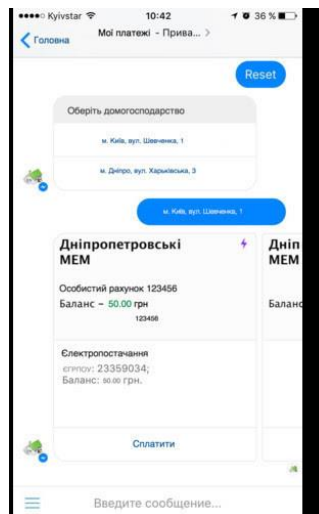


Рисунок 1.3 – Бот «Мої платежі»

Телеграмбот зі штучним інтелектом «Зоряна» від телекомунікаційної

компанії «Київстар» дозволяє клієнтам отримати відповіді на найпоширеніші питання та послуги компанії (рис. 1.4).

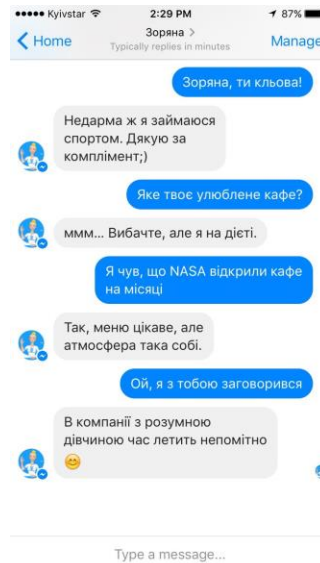


Рисунок 1.4 – Бот «Зоряна»

Whereisnpbot бот допомагає користувачам відстежувати посилки компанії «Нова Пошта». Також існує телеграмбот для відстеження посилок від всіх великих поштових компаній України (рис. 1.5).

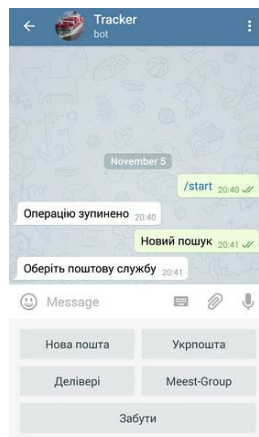


Рисунок 1.5 – Бот для відстеження посилки

iGovBot – телеграм-бот, який шукає на порталі держпослуг iGov

інформацію про компанії за назвою, кодом ОКПО або імені директора і повідомляє про зміни в тих компаніях, на які підписаний користувач (рис. 1.6) [8].



Рисунок 1.6 – Бот на порталі державних послуг

Перелічимо причини використання чатботів для підприємств.

Перший головний критерій використання чатботу це економічна вигода для компанії. Багато простих функцій, які виконує оператор або менеджер, чатбот може реалізовувати за декілька кліків.

Другий фактор використання боту полягає у відсутності черг та необхідності відвідувати спеціалізовані приміщення. Кількість клієнтів, які одночасно обслуговуюються можуть нараховувати тисячі.

Режим роботи чатботу може бути цілодобовим. Ця перевага програмного забезпечення дозволяє клієнтам отримувати консультацію або відповіді в зручний для них час.

#### **1.4 Висновки до розділу**

Через збільшення конкуренції між компаніями потрібно використовувати сучасні методи роботи з клієнтами. Чатбот належить до найбільш перспективних розробок. У цьому розділі було розглянуто загальні поняття чатбота у різних месенджерах. Аналіз сучасних розробок дозволив скласти класифікацію всіх ботів, що існують на даний момент. Також були виділені переваги використання месенджерів у бізнесі та сучасні чатботи.

## 2 ДОСЛІДЖЕННЯ ЗАПИТІВ ЧАТБОТІВ

### 2.1 Методи розробки чатботу

Найпоширеніший варіант розробки чатботу – інтерактивна форма, найбільш складний – програма зі штучним інтелектом, що імітує співрозмовника, вміє підтримати розмову на будь-яку тему [15].

Створити чатбот можна двома способами:

- програмувати самостійно;
- використовувати drag and drop конструктор.

Якщо узагальнити, вибір способу залежить від цілей, досвіду та кількості витрат на розробку. Таблиця 2.1 показує переваги та недоліки розробки чатботів кожним способом.

Таблиця 2.1 – Методи розробки чатботів

| Програмування самостійно                                                                                                                                                                                                                                                                                              | Використання конструктора                                                                                                                                                                                                                                                                                                                               |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Переваги                                                                                                                                                                                                                                                                                                              |                                                                                                                                                                                                                                                                                                                                                         |
| <ul style="list-style-type: none"> <li>– інтеграція складних унікальних функцій;</li> <li>– вибір будь-яких платформ та технічних рішень;</li> <li>– розробка інтерфейсу користувача, відповідного очікуванням ЦА;</li> <li>– швидке виправлення недоліків, технічне обслуговування та покращення чатбота.</li> </ul> | <ul style="list-style-type: none"> <li>– легкий старт за допомогою відеоінструкцій, баз знань та допомоги техпідтримки;</li> <li>– швидкість створення чат-ботів;</li> <li>– можливість спробувати новий інструмент безкоштовно або за умовні \$5-10 на місяць;</li> <li>– проста інтеграція зі сторонніми сервісами, якщо вона передбачена.</li> </ul> |

Продовження таблиці 2.1

| Програмування самостійно                                                                                                                                                                                                                        | Використання конструктора                                                                                                                                                                                                                                                                                            |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Недоліки                                                                                                                                                                                                                                        |                                                                                                                                                                                                                                                                                                                      |
| <ul style="list-style-type: none"> <li>– вартість;</li> <li>– необхідність підтримувати працездатність хостингу та сервера;</li> <li>– тривалість підготовки, реалізації та тестування;</li> <li>– складності при зміні розробникао.</li> </ul> | <ul style="list-style-type: none"> <li>– не на всіх платформах можна експортувати передплатників;</li> <li>– обмежена кількість месенджерів та інтеграцій із зовнішніми сервісами;</li> <li>– розробники довго впроваджують нові потрібні функції;</li> <li>– залежність від сторонньої служби підтримки.</li> </ul> |

Інтерактивний бот можна розробити за допомогою наступних зручних сервісів:

– Flow XO – конструктор чат-ботів, що не вимагає знання коду. Сервіс простий у використанні, з інтуїтивно зрозумілим візуальним редактором. Є можливість використовувати готові шаблони і фільтри (рис. 2.1). Підключення до поштових баз даних для організації розсилок [16];

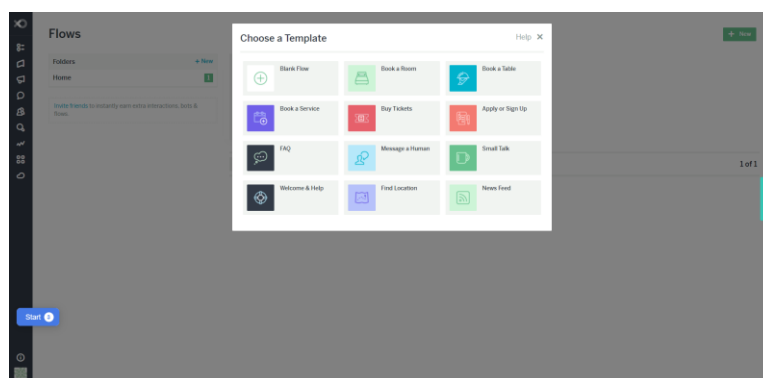


Рисунок 2.1 – Конструктор ботів Flow XO

– Chatfuel – дозволяє створити і запустити чатбота всього за 3 кроки:

створити, інтегрувати з бажаним додатком, запустити. Сервіс умовно безкоштовний та дуже популярний [17] (рис. 2.2.);

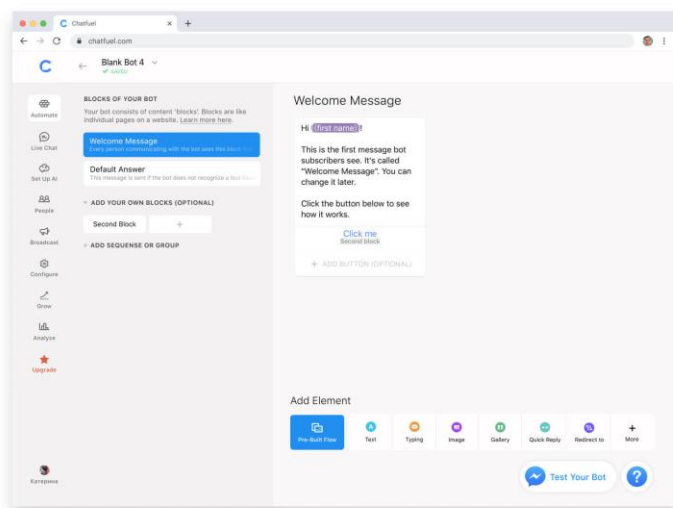


Рисунок 2.2 – Сервіс розробки ботів Chatfuel

– Dialogflow від корпорації Google. Стандартний тариф безкоштовний, він охоплює більшість потреб розробників. На сайті є повноцінне керівництво в тому, як почати роботу і зробити свого першого бота (рис. 2.3) [18].

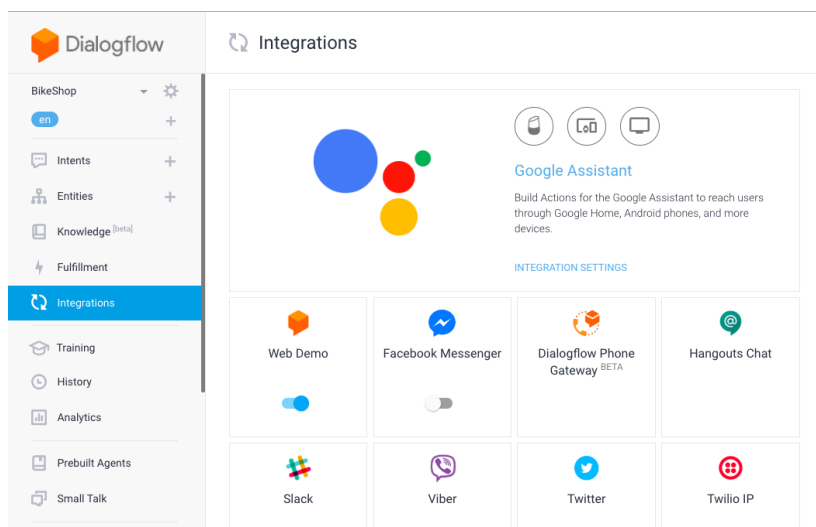


Рисунок 2.3 – Сервіс розробки ботів Dialogflow

Для створення ботів без використання сервісу, використовують API платформи.

Існує два протилежних за логікою способи отримання оновлень від бота. Long pulling – додаток автоматично опитує сервера Telegram на наявність будь-яких оновлень для бота [16]. Webhook – сервера Telegram самі сповіщають додаток на сервері як тільки з'являться будь-які оновлення. Незалежно від способу отримання оновлення, у відповідь відправляється об'єкт JSON [17].

Завдання, які вирішує чатбот такі:

- зменшення витрат та часу. Завдяки встановленій програмі можна значно скоротити кількість менеджерів, які відповідають базові питання клієнтів;
- оптимізація робочого процесу. Чатбот бере на себе відповіді на питання, що часто задаються. До обов'язків менеджерів входить лише вирішення складних ситуацій, які потребують втручання людини;
- відповіді на замовлення у цілодобовому режимі;
- збільшення кількості замовлень. Удосконалення чатботів дозволило багатьом бізнесам вбудувати в систему діалогу з клієнтом можливість швидкої та безпечної оплати обраного продукту;
- відсутність помилки через людський фактор;
- аналітика, звітність, систематизація. Чатбот зберігає показники про виконану роботу, тому легко зможете отримати важливі аналітичні дані при необхідності. Він також автоматично класифікує та систематизує процеси.

## **2.2 Порівняння чатботів з мобільними додатками**

Зростає кількість створених програм (для Android існує більше 2,8 мільйонів програм, для iOS — більше 2,2 мільйонів), але не кількість використовуваних. Більшість часу користувачі проводять у п'яти улюблених

додатках. Одне, як правило, месенджер [19].

Для того щоб зробити висновок яка розробка необхідніше для розвитку компанії треба порівняти переваги та недоліки кожного сервісу зі сторони користувачів та замовників.

Перелічено переваги чат-ботів зі сторони користувачів:

- боти не вимагають встановлення. Програму потрібно знайти, завантажити та встановити. Це триває час. Адже користувач не завжди навіть впевнений, що програма зручна і там є функції, які йому потрібні. Чатбота можна просто додати в месенджері та одразу користуватися ним;

- боти не займають місця на телефоні. Можливості смартфонів постійно зростають. Чат-боти не займають пам'ять телефону та місце на робочому столі;

- боти пропонують природну взаємодію. Додати робота, як додати нового знайомого в Telegram - це простий і природний крок.

Для власників бізнесу наступні переваги використання боту:

- бота можна розробити швидше та дешевше. Якщо говорити про подібний функціонал, то створення чат-бота для бізнесу буде в середньому однозначно швидшим і дешевшим. Так, розробка простого робота займає від тижня і коштує від \$500. Програма розробляється щонайменше 2 місяці (як правило, від 4) і коштує від \$2 000;

- бота не потрібно постійно оновлювати, на відміну додатків. Щоб не відстати від оновлень ОС, потрібно постійно оновлювати програми. Чатботи не потребують постійного оновлення.

Враховуючи всі ці та інші фактори ми розробили порівняльну таблицю 2.2.

Таблиця 2.2 – Порівняння чатботів та мобільних додатків

| Параметри                              | Чат-бот                                         | Мобільний додаток                                      |
|----------------------------------------|-------------------------------------------------|--------------------------------------------------------|
| Інсталяція                             | Не потрібна. Достатньо знайти бота в месенджері | Потрібна                                               |
| Використання фізичної пам'яті пристрою | -                                               | +                                                      |
| Авторизація                            | Достатньо мобільного номеру                     | Необхідні дані профілю                                 |
| Швидкість розробки                     | Від 2 тижнів                                    | Від 2 місяців                                          |
| Оновлення                              | Автоматично разом із месенджером                | Самостійно оновлювати при внесені змін або оновлені ОС |
| Топ мов програмування                  | Python, Java, C, PHP, Clojure, Ruby             | Python, Swift, JS, C#, Java, Objective-C               |
| NLP                                    | +                                               | -                                                      |
| Анімація                               | -                                               | +                                                      |
| Інтерфейс                              | Розмовний                                       | Унікальний дизайнерський                               |
| Платформа                              | Месенджери (Facebook, telegram, viber)          | ОС (Android, iOS)                                      |

Наведемо деякі зауваження, чому не можна однозначно обирати чатботів для розробки.

Різний функціонал ботів та додатків. Боти не завжди можуть забезпечити весь функціонал, який надає програма. Наприклад, у програмі Uber можна відразу бачити чи є вільні машини неподалік. З іншого боку, для служби підтримки звичайно використовувати чат. При цьому на багато питань може відповісти чатбот і практично непомітно підключити до розмови живого

оператора, якщо це необхідно;

Різний інтерфейс. Інтерфейс чатбота включає графічні елементи, але в цілому він вважається розмовним і це є перевага ботів. По-перше, звичайна людська мова - найприродніший спосіб взаємодії для людей. По-друге, інтерфейс чатботів є універсальним. Коли людина встановлює нові програми, спочатку потрібно зрозуміти, як ним користуватися і де яка функція знаходиться. З іншого боку, створюючи додаток, простіше виділитися серед конкурентів за рахунок яскравого дизайну. Крім того, за допомогою графічного інтерфейсу додатків деякі речі все ж таки можна зробити швидше.

Історія повідомлень у додатку та чатботі. У чатботі можна повернутися і переглянути історію повідомлень та зроблені вибори.

Порівнювати можна продукти однакової якості. Розробка чатботу або додатку можуть бути марними при поганому дизайні або неуважному тестуванні. Обидві програми можна зробити незрозумілим для користувача. Якщо при замовленні програми виникає питання, зробити його на iOS, Android або кросплатформним, то при розробці чатбота доведеться вибирати між месенджерами або також, зробити його кросплатформним.

### **2.3 Висновки до розділу**

В даному розділі проаналізовані метода та фреймворки розробки чатботів. Проведено дослідження факторів вибору між чатботом та мобільним додатком. Наведені параметри аналізу, які показують переваги та недоліки чатботів. Розробка чатботу або додатку можуть бути марними при поганому дизайні або неуважному тестуванні.

## 3 РОЗРОБКА АРХІТЕКТУРИ ЧАТБОТУ

### 3.1 Структура чатбота

Архітектура чатбота - це найголовніша та найвідповідальніша частина розробки чатботів [15]. Залежно від зручності використання та контексту бізнес-операцій, архітектура, пов'язана з програмуванням чатбота, може різко змінюватися. Незважаючи на різні вимоги клієнта зміни чи перетворення того чи іншого елемента архітектури не впливають на основний потік комунікації чатбота та клієнту.

На рисунку 3.1 зображено узагальнену архітектуру чатбота.

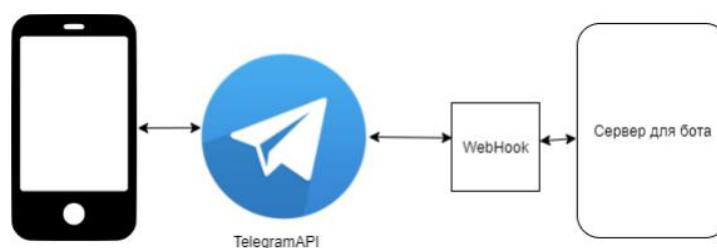


Рисунок 3.1 – Узагальнена архітектура чатбота

Дана схема включає наступні елементи.

Сервер зберігає програмний код де прописані всі операції та логіка чатбота. Під час своєї активності чатбот використовує дані зі сховища, взаємодіє з веб-сервісами, за допомогою яких і надає інформацію користувачам.

База даних – одне з найважливіших елементів аналізованої архітектури, що зберігає всі дані та необхідні налаштування для повноцінної роботи чатбота.

Telegram API – у межах цієї структури він виконують роль інтерфейсу за допомогою якого чатбот може взаємодіяти з користувачами.

Всі перераховані вище елементи, які представлені в цій комунікації є невід'ємними частинами всієї структури чат-бота. А сама архітектура –

еталонною.

### 3.2 Функціональні вимоги до чат-боту

Вимоги до чатботу – набір вимог щодо властивостей, якості та функцій ПЗ, що буде розроблено. Вимоги визначаються в процесі аналізу та фіксуються в специфікації вимог, діаграмах прецедентів та інших артефактах [20].

Перелічимо функціональні можливості розробляемого чатботу:

- функція замовлення таксі;
- дружелюбний інтерфейс;
- навігація в функціях чатботу;
- швидкість роботи кнопок чатботу;
- функція перегляду історії відповідей;
- функція обрати замовлення;
- функція реєстрації, авторизації користувачів;
- функція відправки геолокації.

Вимоги чатботу для водіїв:

- функція надання інформації про замовлення;
- функція обирати замовлення;
- функція зміни інформації про автівку;
- авторизація водіїв;
- наявність навігатора.

В таблиці 3.1 наведені характеристики всіх користувачів чатботу.

Таблиця 3.1 – Користувачі чатботу замовлення таксі

| Користувач | Характеристика                                                       |
|------------|----------------------------------------------------------------------|
| Пасажир    | Особа, яка замовляє таксі.                                           |
| Водій      | Особа, яка виконує послугу перевезення.                              |
| Розробник  | Особа, яка забезпечує коректну роботу і регулярне оновлення чатботу. |

Для того щоб чатбот правильно відображався у користувачів необхідно мати оновлений месенджер.

### 3.3 Моделювання чатботу

Процес моделювання є процес переходу з реальної області в віртуальну (модельну) за допомогою формалізації, далі відбувається вивчення моделі (власне моделювання) і інтерпретація результатів як зворотний перехід з віртуальної області в реальну. Цей шлях замінює пряме дослідження об'єкта у реальній області, тобто лобове чи інтуїтивне вирішення завдання [20].

Розробка моделі завжди передуює її створенню чи оновленню. Це необхідно хоча б для того, щоб ясніше уявити вирішуване завдання. Продумані моделі дуже важливі для взаємодії всередині команди розробників, і для взаєморозуміння з замовником. Зрештою це дозволяє переконатися в "архітектурній узгодженості" проекту до того, як він буде реалізований в коді.

UML (Unified Modeling Language) призначений для спрощення спілкування та взаємодії учасників проекту, скорочення часу на пояснення та присвоєння інформації, полегшення документування.

На рисунку 3.2 приводиться діаграма варіантів використання чатботу пасажиром таксі. Чатбот реалізує мінімум 6 функцій для здійснення перевезення.



Рисунок 3.2 – Діаграма варіантів використання «Пасажир»

Чатботом також користуються водії. Це дає змогу виключити необхідність наймати на роботу диспетчера таксі. Водій, який користується чатботом повинен зареєструватись в системі для подальшої авторизації та отримання замовлень (рис. 3.3).



Рисунок 3.3 – Діаграма варіантів використання «Водій»

Водій та пасажир використовують майже однакові функції чатботу (рис. 3.4). Але водій може обрати замовлення серед створених на власний розсуд.

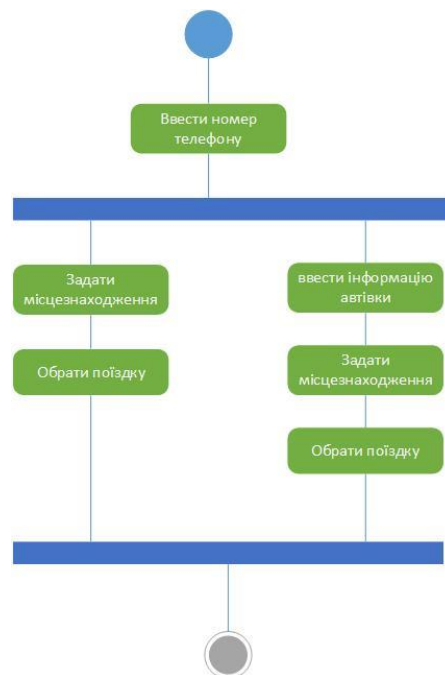


Рисунок 3.4 – Діаграма станів системи чатботу таксі

### **3.4 Висновки до розділу**

В даному розділі розроблена структура чатботу таксі. Вона складається з основних елементів: сервера, бази даних та месенджера в якості інтерфейсу системи. Описані функціональні вимоги до користувачів чатботу. Даний чатбот розробляється для організації перевезень, тому користувачами додатку будуть пасажери та водії. Проведено моделювання основних прецедентів системи за допомогою UML-діаграм використання.

## 4 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ ЧАТ-БОТУ

### 4.1 База даних

SQL і NoSQL це два найбільш поширених типи баз даних. SQL – це мова структурованих запитів, що у більшості сучасних систем управління реляційними базами даних. NoSQL зазвичай використовується в нереляційній базі даних (тобто в тій, де не підтримуються первинні – вторинні ключі та з'єднання між таблицями). Відмінності між базами даних SQL і NoSQL включають те, як вони побудовані структури і типи даних.

Реляційні бази даних використовують жорстку структуру таблиць зі стовпцями та рядками. У кожному рядку є один запис, і кожен стовпець містить певну інформацію. Структуровані дані вимагають нормалізації, що знижує надмірність даних та підвищує їх надійність [21].

Нереляційні бази даних часто реалізуються як систем NoSQL. Існує безліч типів баз даних NoSQL – від сховищ значень ключів до сховищ документів, баз даних графіків, баз даних тимчасових рядів та сховищ із широкими стовпцями. Деякі системи NoSQL також підтримують «багатомодельні» схеми - це означає, що вони можуть підтримувати в собі більше однієї схеми даних [22].

Порівняння NoSQL і SQL базується на таких умовах, як безперервність, доступність та швидкість. Як правило, розробники самі визначають, який тип системи баз даних слід використовувати, виходячи від своїх потреб (рис. 4.1).

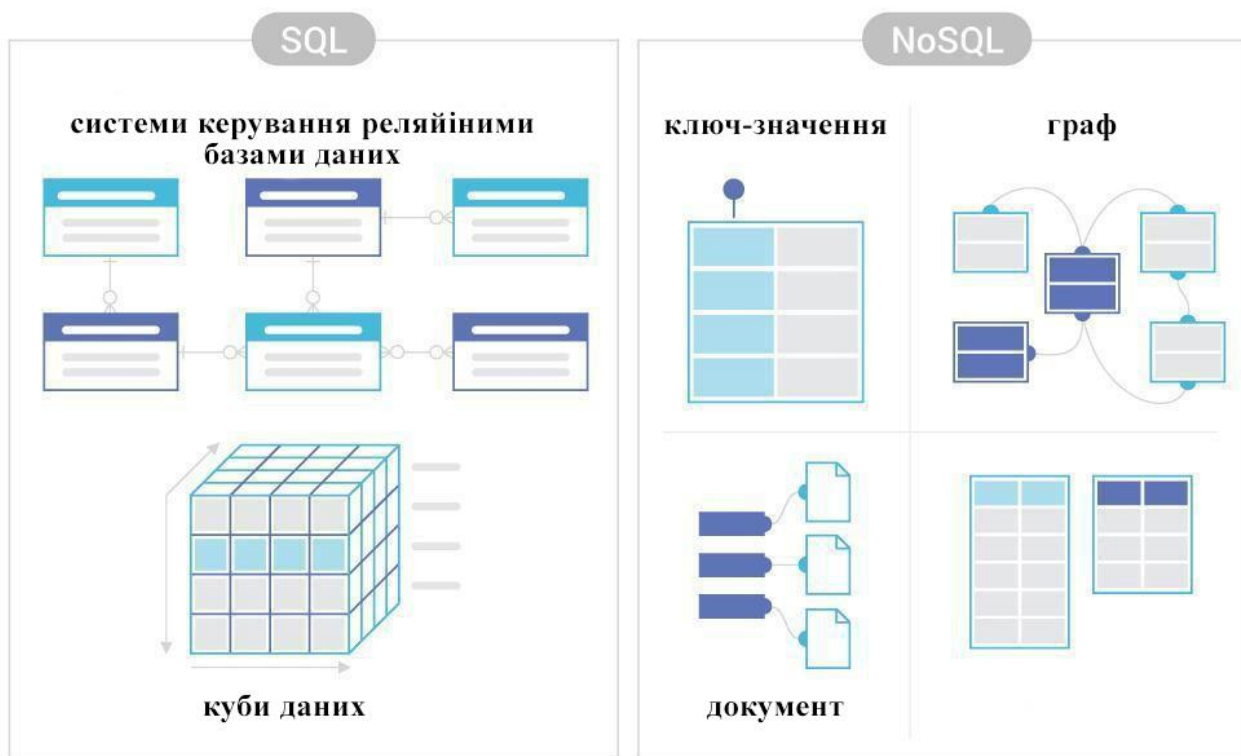


Рисунок 4.1 – Ключові відмінності між SQL та NoSQL [22]

Реляційна база даних підходить для обробки моделей даних, які добре опрацьовані та незмінні протягом великої кількості часу. Реляційні моделі мають високу швидкість обробки та передачі даних.

Нереляційна база даних використовується там, де постійно стикаються з мінливими вимогами до зберігання даних, що швидко розвиваються. Використання таких баз необхідно, коли потрібно мати справу з декількома типами даних і великими обсягами. потоку інформації.

Список найпопулярніших реляційних баз даних включає себе:

- Oracle Database;
- MySQL;
- SQLite;
- PostgreSQL;
- Microsoft SQL Server.

Список популярних нереляційних баз даних включають:

- Apache Cassandra;
- Apache HBase;
- Scylla;
- MongoDB;
- Redis.

Задля реалізації чатботу таксі виникла потреба у базі даних (БД). У проєкті використовується SQLite 3.0. База даних чатбота складається з трьох таблиць. Діаграма структури БД представлена рисунку 4.2.



Рисунок 4.2 – Діаграма структури бази даних

Кожна таблиця виконує свою функцію:

- Drivers зберігає інформацію про водіїв (табл. 4.1);
- Orders зберігає інформацію про замовлення (табл. 4.2);
- Passangers дані пасажира таксі (табл. 4.3).

Таким чином, було реалізовано базу даних для чатбота на платформі Telegram.

Таблиця 4.1 – Сутність Drivers

| Атрибут      | Тип даних |
|--------------|-----------|
| phone        | Числовий  |
| machine_firm | Текстовий |
| longitude    | Числовий  |
| latitude     | Числовий  |
| photo_car    | Числовий  |

Таблиця 4.2 – Сутність Orders

| Атрибут         | Тип даних |
|-----------------|-----------|
| phone           | Числовий  |
| longitude_start | Числовий  |
| latitude_start  | Числовий  |
| longitude_end   | Числовий  |
| latitude_end    | Числовий  |
| Price           | Числовий  |
| length_way      | Числовий  |
| time_way        | Числовий  |

Таблиця 4.3 – Сутність Passengers

| Атрибут | Тип даних |
|---------|-----------|
| phone   | Числовий  |

## 4.2 Вибір мови програмування та налаштування бібліотек

Для розробки чатботу таксі можна використовувати такі мови програмування, як Java, Python, C, PHP. Розглянемо переваги кожною мови.

Java – завжди відмінний вибір з огляду на її здатність створювати високорівневі функції. Як об'єктно-орієнтована мова програмування, переносимість Java робить його ідеальним для розробки чат-ботів [23].

Завдяки компактному та портативному розміру Java також пропонує вбудований засіб збору коду. Крім того, Java має кілька чудових функцій для створення чатботів на базі штучного інтелекту. Деякі з ключових функцій, що роблять Java розповсюдженою:

- Java має високу масштабованість;
- Java компактна і дуже портативна;
- Java чудово підходить для простого кодування алгоритмів з розширеними функціями.

Clojure - це мова програмування, що використовується для спільних цілей і дозволяє отримати доступ до кількох фреймворків Java з виведенням типів та підказками для додаткових типів [24].

Ця мова вважається частиною сімейства мов Lisp. Як і Lisp, вона побудована з використанням підходу «код як дані». Ця мова, крім роботи на JVM, дозволяє розробникам кодувати програму з хронологічним рядом функцій.

Погляньмо на деякі ключові переваги Clojure:

- чистий та точний код;
- Clojure пропонує багатопоточний та правильний код завдяки системі транзакційної пам'яті та системі реактивних агентів;
- мова вважається простою у використанні через функції вищого порядку замість циклу на основі побічних ефектів.

Python на даний момент вважається найбільш широко використовується мовою програмування для створення чатботів. Це визначено тим, що Python оснащений AIML або мовою розмітки штучного інтелекту, що спрощує створення синтаксису для складних функцій чатбота [24].

Приведемо основні переваги Python для розробки чатботів:

- Python використовується на серверній стороні додатку;
- має великий обсяг бібліотек для обробки природної мови;
- універсальна мова.

Якщо обирати мову розробки за швидкістю та продуктивністю можна казати про мову C++ [25].

Переваги використання мови C++:

- усі алгоритми чатбота на базі штучного інтелекту можуть бути повністю написані на C++;
- C++ пропонує структуровану абстракцію високого рівня.

Створення чатбота мовою PHP обходиться дешево, тому що мова є повністю відкритою і надзвичайно простою у використанні. Як мову сценаріїв високого рівня її можна використовувати для створення чатбота з меншими витратами [26].

Переваги використання PHP для чатботів:

- PHP - дуже швидка мова сценаріїв у порівнянні з багатьма іншими;
- використовуючи PHP, отримаємо широкий спектр бібліотек, доступних через API;
- чатбот, створений на PHP, може працювати на всіх платформах.

Ruby - ще одна потужна та високорівнева мова програмування, яка використовується для створення чат-бота. Як об'єктно-орієнтована мова програмування Ruby дуже схожа на Python у тому, що стосується простоти створення синтаксису [27].

Мова Ruby добре оснащена всіма інструментами, необхідні для створення складного чатбота:

- Ruby пропонує широкий спектр якісних бібліотек для всіх функцій чатбота;
- Ruby забезпечує чудову продуктивність усіх функцій;
- як об'єктно-орієнтована мова програмування Ruby пропонує дуже чистий та точний синтаксис.

Lisp – одна з найстаріших мов програмування високого рівня, яка пройшла багаторівневу еволюцію і стала високоефективною та динамічною мовою. Завдяки своїм динамічним властивостям вона широко використовується розробниками для створення інтелектуальних та діалогових чатботів усіх типів.

Переваги цієї мови:

- Lisp має можливість динамічного прототипування;
- добре адаптується до конкретних завдань і може використовуватися для створення ботів для певних цілей.

З огляду на всі перелічені переваги кожної мови для розробки проєкту оберемо Python. Інсталяція бібліотеки дуже проста. Бібліотека `pyTelegramBotApi` сама відповідає за всі нюанси відправлення та отримання запитів, дозволяє зосередитися безпосередньо на логіці.

Перед встановленням та запуском потрібно встановити WebHook на потрібну адресу та зловити вихідний сигнал. Це сприяє поверненню поведінки з розвантаженням подій вручну.

Telegram контролює повністю повідомлення. Необхідність ставити перед командами “/” створено лише для того, щоб користувачам було зручно. Це дозволяє навчити програму відповідати по-людськи. Для того, щоб дати їй можливість отримувати у групах всі повідомлення, потрібно написати `/setprivacy` помічнику `@BotFather`, відключаючи таким чином приватність.

Основний код додатку наведений у лістингу А.1. Далі розглянемо

ключові програмні моменти реалізації чат-боту.

На рисунку 4.3 наведений програмний код стартової команди. Для того, щоб почати роботу з ботом необхідно надіслати власний номер.

```
@bot.message_handler(commands=["start"])
def start(message):
    name = message.text
    bot.send_message(message.chat.id,
        "Привіт <b>{first_name}</b> Надішли мені свій номер виконавши команду /phone".format(first_name=message.from_user.first_name),
        parse_mode='HTML', reply_markup=types.ReplyKeyboardRemove())
```

Рисунок 4.3 – Команда «/start»

Власний номер можна надіслати боту натиснувши кнопку "Надіслати номер телефона". Телеграм –бот запитає згоду надіслати власний номер у чат. Код команди наведений у рисунку 4.4.

```
@bot.message_handler(commands=["phone"])
def phone(message):
    user_markup = types.ReplyKeyboardMarkup(row_width=1, resize_keyboard=True, one_time_keyboard=True)
    button_phone = types.KeyboardButton(text="Надіслати номер телефона", request_contact=True)
    user_markup.add(button_phone)
    msg = bot.send_message(message.chat.id, "Згодні надати номер для реєстрації?", reply_markup=user_markup)
    bot.register_next_step_handler(msg, reg_or_auth)
```

Рисунок 4.4 – Команда «Надіслати номер телефона»

Після авторизації за номером чатбот має два варіанти команд для пасажирів та водія (рис. 4.5), (рис. 4.6).

```
# keyboard for auth passenger
buttons_actions = types.ReplyKeyboardMarkup(resize_keyboard=True, one_time_keyboard=True)
button_history_ways = types.KeyboardButton(text="Мої поїздки")
button_add_order = types.KeyboardButton(text="Нова поїздка")
buttons_actions.add(button_history_ways)
buttons_actions.add(button_add_order)
mess = bot.send_message(message.chat.id, "Обери дію", reply_markup=buttons_actions)
bot.register_next_step_handler(mess, choose_action_passenger, input_phone, message.chat.id)

return '' # stop function
```

Рисунок 4.5 – Меню команд для пасажирів

```

# keyboard for auth taxi driver
buttons_actions = types.ReplyKeyboardMarkup(resize_keyboard=True, one_time_keyboard=True)
button_settings = types.KeyboardButton(text="Налаштування")
button_choose_order = types.KeyboardButton(text="Обрати поїздки")
buttons_actions.add(button_settings)
buttons_actions.add(button_choose_order)
mess = bot.send_message(message.chat.id, "Обрати дію.", reply_markup=buttons_actions)
bot.register_next_step_handler(mess, choose_action_taxi_driver, input_phone, message.chat.id)

return '' # stop function

```

Рисунок 4.6 – Меню команд для водія

Структура проекту складається з наступних файлів:

- база даних «base.db»;
- головний файл додатку «app.py»;
- файли роботи з картами та геоданими «geocoder\_coords.py», «static\_map\_pas.py».

### 4.3 Реєстрація бота в Telegram

Для початку необхідно зареєструвати саму оболонку роботи в месенджері, отримати присвоєний їй ID, який є також токеном. У телеграмі допомогу при такій реєстрації надає віртуальний помічник @BotFather.

Токен отримали за наступними кроками:

- ввели команду /start для запиту інструкції (рис.4.7);

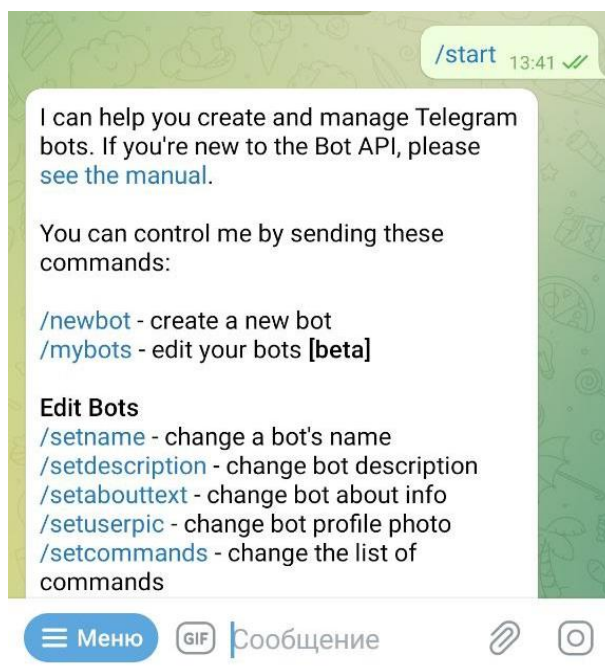


Рисунок 4.7 – Початок роботи з @BotFather

– надсилання установки /newbot (рис. 4.8);

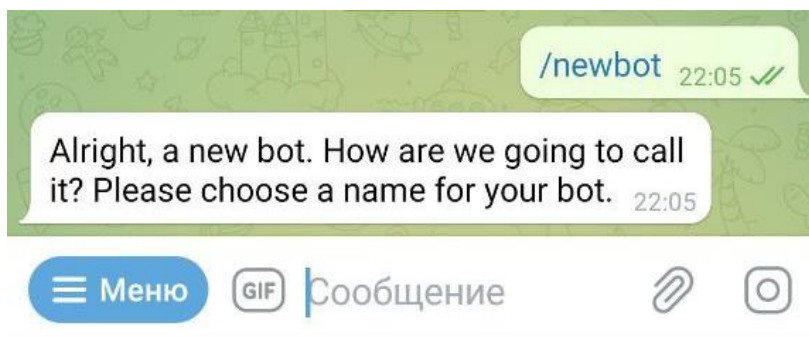


Рисунок 4.8 – Запит на створення нового бота

– @BotFather пропонується придумати ім'я для створюваного робота (рис. 4.9);



Рисунок 4.9 – Створення нового бота

– якщо ім'я підходить, пропонується додати до нього закінчення "bot" (рис. 4.10);



Рисунок 4.10 – Нове м'я бота

– далі помічник надсилає токен нового віртуального користувача та url для додавання його до списку контактів. Отриману комбінацію бажано протестувати за посиланням: [api.telegram.org/bot/HomeTaxi\\_bot](https://api.telegram.org/bot/HomeTaxi_bot). Останній крок це отримання токена для подальшої роботи (рис. 4.11).

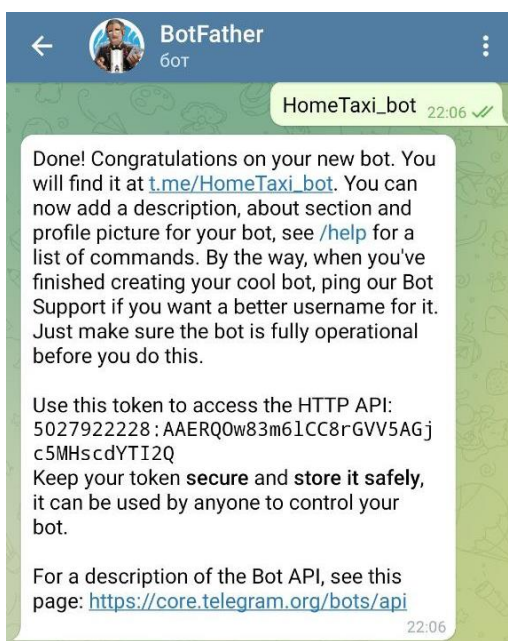


Рисунок 4.11 – Токен нового бота

API телеграму має багато функцій для встановлення параметрів розробляемого боту. До обов'язкових налаштувань чатбота можна віднести: збереження іконки, опис чатбота, текст на старті роботи боту, список використовуваних команд (табл. 4.4).

Таблиця 4.4 – Команди зміни параметрів чат-бота

| Команда        | Функції                                          |
|----------------|--------------------------------------------------|
| /setname       | Зміна імені                                      |
| /setdecription | Текст при старті бота                            |
| /setusertext   | Задає текст в інформаційному описі боту          |
| /setcommands   | Дозволяє створити список використовуваних команд |
| /deletebot     | Видалення бота                                   |
| /setuserpic    | Задає картинку боту                              |

Для налаштування боту також є специфічні команди. Ці команди задають уніфіковані параметри доступу до боту, видалення, режим конфіденційності та інші (табл. 4.5).

Таблиця 4.5 – Команди зміни параметрів чатботу

| Команда            | Функції                                         |
|--------------------|-------------------------------------------------|
| /token             | Повертає призначений токен                      |
| /revoke            | Анулює токен                                    |
| /setinline         | Вимикає можливість викликати бота з інших чатів |
| /setinlinegeo      | Вимикає можливість передавати посилання на бот  |
| /setinlinefeedback | Інформацію про кількість обраних команд         |
| /setjoininggroup   | Бот може використовуватись у групах             |
| /setprivacy        | Режим конфіденційності                          |

#### 4.4 Висновки до розділу

Четвертий розділ описує програмні рішення реалізації телеграм-боту таксі. Розроблена SQL база даних, яка зберігає інформацію про водіїв, пасажирів та їх поїздки. В якості мови програмування обрана мова Python. Реєстрація бота виконана за допомогою віртуального помічника Telegram @BotFather, де отриман власний токен чатботу.

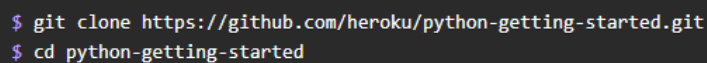
## 5 ТЕСТУВАННЯ ЧАТБОТУ

### 5.1 Розміщення чатбота на віддаленому сервері

Для постійного функціонування чатботу його розташовують на сервері, який конфігурує його роботу. Для комфортної інфраструктури та можливості розширення чат-боту варто розглянути хостинг з наявністю бази даних, веб-сервера, SSL-сертифіката. В якості платформи був обраний сервер Heroku.

Heroku — це хмарна платформа як послуга (PaaS), яка підтримує безліч мов програмування. Історія Heroku почалася в 2007 році, і тоді першим мовою програмування був Ruby. Тепер вона підтримує Java, Node.js, Scala, Clojure, Python, PHP і Go.

Перший крок перед розгортання чатботу це підготовка програмного коду. Обов'язковою умовою є встановлення Git. Це потрібно для того, щоб клонувати приклад програми. Для локальної версії коду, яку треба розгорнути на Heroku, виконаємо наступні команди (рис.5.1).



```
$ git clone https://github.com/heroku/python-getting-started.git
$ cd python-getting-started
```

Рисунок 5.1 – Налаштування локальної версії коду

Тепер у репозиторію зберігаються файли runtime.txt, що вказує версію Python, і файл requirements.txt, який використовується менеджером залежностей Python Pip.

Наступний крок створення локального клієнту Heroku, який готує сервер для отримання вихідного коду (рис. 5.2).

```
$ heroku create  
Creating app... done, ● serene-caverns-82714  
https://serene-caverns-82714.herokuapp.com/ | https://git.heroku.com/serene-caverns-82714.git
```

Рисунок 5.2 – Налаштування Heroku

Остання команда для розгортання чатботу наведена на рисунку 5.3.

```
$ git push heroku main
```

Рисунок 5.3 – Розміщення чатботу на сервері

## 5.2 Тестування чатботу

Тестування чатбота таксі здійснювалося вручну. Тестування працездатності перевірялось по наступним параметрам:

- швидкість відгуку чатбота на повідомлення /start;
- коректна обробка повідомлень від водія та пасажирів;
- коректне відображення діалогу;
- обробка натискання на кнопки;
- швидкість відображення карти;
- швидкість роботи запитів до БД.

Робота користувача з ботом починається після введення команди /start. У відповідь клієнт отримує опис чатбота (рис. 5.4).

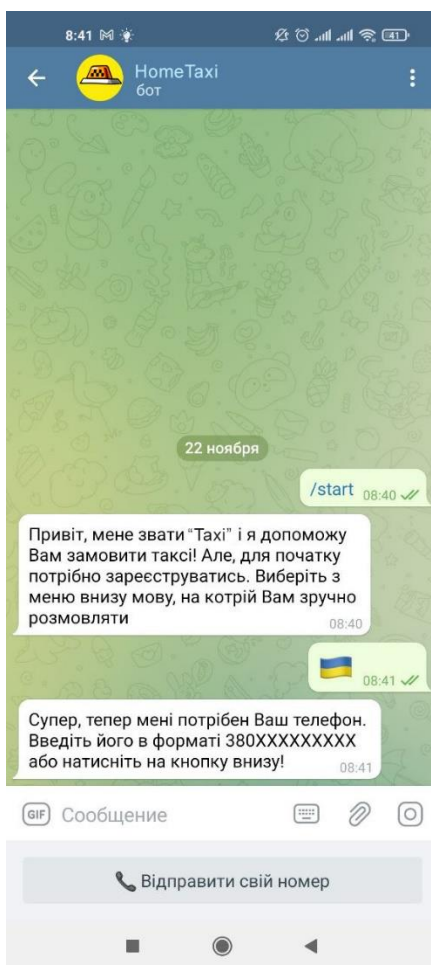


Рисунок 5.4 – Перша команда взаємодії користувача з ботом

Після старту для клієнтів з'являється функція авторизуватися, натиснувши кнопку «Надіслати номер телефона». Ця дія потрібна, щоб розрізняти водіїв та пасажирів, а також ідентифікувати кожного користувача. У відповідь чатбот перепитає дозвіл на отримання інформації про власний мобільний номер (рис. 5.5).



Рисунок 5.5 – Авторизація користувача

Якщо власником мобільного номеру є збережений в базі водій, він отримає доступ до категорії замовлень. Якщо у чатботі авторизувався пасажир, то далі включається механізм замовлення таксі на певну адресу. Ввівши адресу, яка потрібна користувачеві, чатбот вирахує ціну поїздки, час та вістань до місця призначення та збереже цю інформацію для водіїв (рис. 5.6).

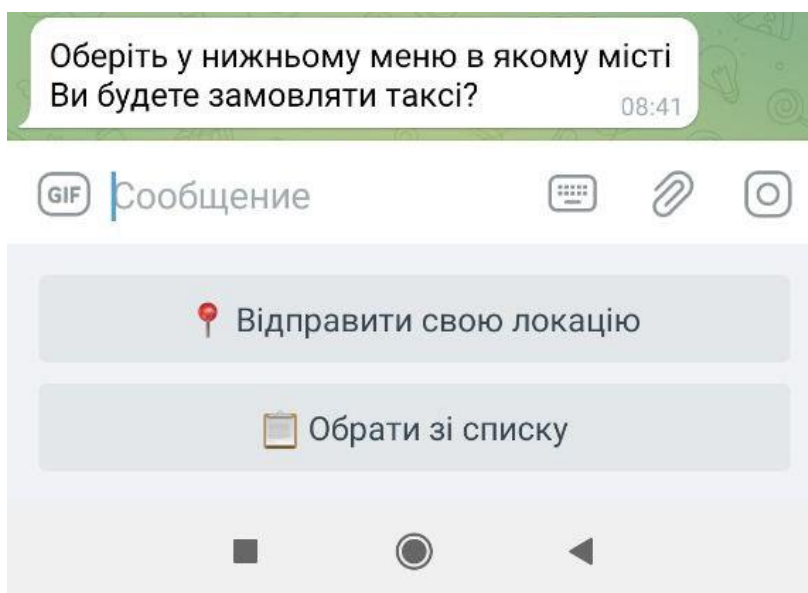


Рисунок 5.6 – Замовлення таксі

Пасажира може обрати вже збережену поїздку, якщо вона була здійснена з його мобільного номеру (рис. 5.7).

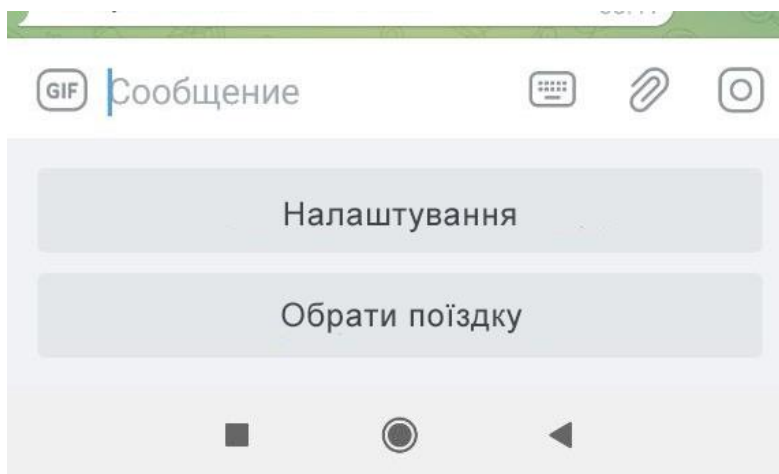


Рисунок 5.7 – Меню пасажира

Тестування чатботу проводилось вручну. Для кожного варіанта використання розроблена відповідна функція, яка працює успішно. В таблиці 5.1 наведені результати тестування.

Таблиця 5.1 – Тестування чатботу

| № | Параметр                                                      | Значення |
|---|---------------------------------------------------------------|----------|
| 1 | Середня швидкість відгуку чатбота на повідомлення /start, сек | 0,1      |
| 2 | Середня тривалість сесії замовлення таксі, хв                 | 3        |
| 3 | Середня швидкість відображення карти, сек                     | 1,5      |
| 4 | Середня швидкість запитів до БД, сек                          | 0,3      |
| 5 | Коректна обробка повідомлень водія                            | Успішно  |
| 6 | Коректна обробка повідомлень пасажирів                        | Успішно  |
| 7 | Коректне відображення діалогу                                 | Успішно  |
| 8 | Коректна робота кнопок                                        | Успішно  |

Чатбот допомагає не лише скоротити витрати, але й підвищити продуктивність завдяки автоматизації обміну повідомленнями у месенджері. Можна масштабувати обробку звернень для роботи у режимі 24/7 без додаткових фінансових витрат. Розгортання чатботу призводить до значного скорочення часу відгуку. Можна автоматизувати вітальні повідомлення, аналізувати вхідні повідомлення для сегментації клієнтів та інше — все це сприяє підвищенню задоволеності клієнтів.

### 5.3 Висновки до розділу

Для постійного та швидкого отримання доступу до чатботу він був розташований на сервері Негокі. Тестування телеграм бота здійснювалося вручну. Працездатність бота перевірялась за наступними параметрами: середня швидкість відгуку чатботу на повідомлення, середня тривалість сесії, середня швидкість відображення карти, коректна обробка повідомлень, діалогів та настикання кнопок.

## ВИСНОВКИ

Розвиток технологій у світі сприяє швидкому обміну інформацією та появі нових каналів комунікації, таких як месенджери та мобільні додатки. Месенджери надають багато переваг, але найважливішою є цілодобова підтримка клієнтів компаній. Месенджери забезпечують приватну взаємодію між користувачем та послугою.

В атестаційній роботі проаналізовані методи та фреймворки розробки чатботів. Зроблено висновок, що найпоширеніший варіант розробки це інтерактивна форма. Проведено дослідження факторів вибору між чатботом та мобільним додатком. Наведені параметри аналізу, які показують переваги та недоліки використання чатботів для підприємств.

Архітектура чатбота - це найголовніша та найвідповідальніша частина розробки. Узагальнена архітектура складається з трьох елементів: програмний код, база даних, месенджер. Описані функціональні вимоги до користувачів сервісом Проведено моделювання основних класів системи за допомогою UML-діаграм.

Програмна частина телеграм-боту складається з бази даних та додатку. Використана SQL база даних, яка зберігає інформацію про водіїв, пасажирів та їх поїздки. В якості мови програмування обран Python. Реєстрація бота проведена за допомогою віртуального помічника Telegram @BotFather.

В якості сервера для розташування телеграмботу використан Heroku. Він підтримує всі функції адміністрування, технічної підтримки та роботи бази даних. Тестування телеграм бота здійснювалося вручну. Кожен етап розробки перевірявся за наступними параметрами: середня швидкість відгуку чатботу на повідомлення, середня тривалість сесії, середня швидкість відображення карти, коректна обробка повідомлень, діалогів та настикання кнопок.

В даній кваліфікаційній роботі можна зробити висновок, що поставлені

задачі були вирішені.

Практична цінність роботи полягає в тому, що на підставі аналізу предметної області та дослідження запитів до створення чатботів розроблено програмний застосунок, призначений для підвищення продуктивності роботи підприємства. Розроблений телеграм-бот дозволяє автоматизувати роботу таксі.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Bot in the Bunch: Facilitating Group Chat Discussion by Improving Efficiency and Participation with a Chatbot / Soomin Kim, Jinsu Eun, Changhoon Oh, Bongwon Suh, Joonhwan Lee // CHI '20: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. – April 2020. – Vol.2. – P. 1-13.
2. Будущее чат-ботов: 10 исследований и прогнозы экспертов [Электронный ресурс]. – Режим доступа: <https://www.carrotquest.io/chatbot/future-of-chatbots/>
3. Инструкция: как создавать ботов [Электронный ресурс] / Мю Музафаров. – Режим доступа: <https://habr.com/ru/post/262247/>
4. Chatbot: Modeling the determinants of users' satisfaction and continuance intention of AI-powered service agents / Muhammad Ashfaqa, Jiang Yuna, ShubinYub, Sandra Maria, CorreiaLoureiroc // Telematics and Informatics. – 2020. – Vol. 54. – P. 108-121.
5. Кузнецов, В.В. Перспективы развития чат-ботов / В.В. Кузнецов // Успехи современной науки. – 2016. – Т. 8. – № 12. – С. 16-19.
6. An Overview of Chatbot Technology / Eleni Adamopoulou, Lefteris Moussiades // AIAI 2020: Artificial Intelligence Applications and Innovations. – 2020. Vol. 1. – P. 373-383.
7. Shawar, B.A. Using corpora in machinelearning chatbot systems / B.A. Shawar, E. Atwell // International Journal of Corpus Linguistics. – 2005. – Vol. 10(4). – P. 489–516.
8. Цыганова, Е.В. Чат-боты / Е.В. Цыганова // Студенческий вестник. – 2020. – № 31-2 (129). – С. 34-36.
9. Які мобільні додатки є найбільш популярними? [Електронний ресурс].

– Режим доступу:

<https://www.kiis.com.ua/?lang=ukr&cat=reports&id=1027&page=1&t=5>

10. В яких додатках українці спілкуються найбільше? [Електронний ресурс] / К. Хорошак. – Режим доступу:

<https://life.pravda.com.ua/society/2021/04/19/244623/>

11. Порівнюємо Месенджери [Електронний ресурс]. – Режим доступу: <https://androidclub.com.ua/porivnyuyemo-mesendzhery/>

12. Який месенджер найбезпечніший? [Електронний ресурс]. – Режим доступу: <https://yak.dslua.org/articles/yakuu-mesendzher-naybezpechnishyy/>

13. Топ месенджерів в Україні та світі [Електронний ресурс] / А. Юдін. – Режим доступу: <https://marketer.ua/ua/top-messengers-in-ukraine-and-the-world/>

14. Колеса, що ростуть зі смартфонів. Як український ринок таксі повністю змінився за п'ять років і тепер очікує на прихід китайського гіганта [Електронний ресурс]. – Режим доступу: <https://biz.nv.ua/ukr/markets/uber-bolt-uklon-yak-roste-i-v-yakomu-napryamku-rozvivayetsya-rinok-taksi-v-ukrajini-50160887.html>

15. Біловодська, О.А. Системне дослідження використання чат-боту в комунікації з клієнтами / О.А. Біловодська, К.О. Лагута // Формування ринкових відносин в Україні. – 2020. – No 5(228). – Р. 62-68.

16. Flow XO — обзор сервиса [Электронный ресурс]. – Режим доступа: <https://startpack.ru/application/flowxo>

17. 10 этапов разработки качественного чат-бота для клиентской поддержки [Электронный ресурс]. – Режим доступа: <https://chatbotconf.com.ua/ru/article/10-etapov-razrabotki-kachestvennogo-chatbota-dlya-klientskoy-poddergki-69338>

18. Kosova, E. M. Chatbot integration: analysis of energy industry cases / E. M. Kosova // Innovation & Investment. – 2020. – No 4. – Р. 251-254.

19. Energi Enhances Customer Experience with Chatbot Ellis [Electronic

resource] / Jenny. Göteborg. – Access mode:  
<https://www.getjenny.com/customerservice-chatbot-enhances-customer-service>

20. Chatbots: The Definitive Guide // Artificial Solutions. – 2020. – 33 p.
21. Диков, А.В. Клиентские технологии веб-программирования: JavaScript и DOM / А.В. Диков. – СПб.: Лань, 2020. – 440 с.
22. Садаладж, П. NoSQL. Методология разработки нереляционных баз данных / П. Садаладж, М. Фаулер. – К.: Диалектика, 2020. – 192 с.
23. Эванс, Б. Java. Справочник разработчика / Б. Эванс, Д. Флэнаган. – М.: Вильямс, 2019. – 592 с.
24. Яворски, М. Python. Лучшие практики и инструменты / М. Яворски, Т. Зиаде. – СПб.: Питер, 2021. – 217 с.
25. Kosova, E.M. Chatbot integration: analysis of energy industry cases / E.M. Kosova // Innovation & Investment. – 2020. – No 4. – P. 251-254.
26. Котеров, Д.В. PHP 7: Наиболее полное руководство / Д.В. Котеров, И.В. Симдянов. – СПб.: БХВ-Петербург, 2019. – 1088 с.
27. Метц, С. Ruby. Объектно-ориентированное проектирование / С. Метц. – СПб.: Питер, 2018. – 304 с.
28. Джанарсанам, С. Разработка чат-ботов и разговорных интерфейсов / С. Джанарсанам. – М.: ДМК-Пресс, 2019. – 340 с.
29. Лутц, М. Изучаем Python. / М. Лутц – Пер. с англ. Киев.: ООО «Диалектика», 2020. – 870 с.
30. Реализация Python для Telegram Bot API [Электронный ресурс]. – Режим доступа: <https://pypi.org/project/pyTelegramBotAPI/>
31. Chatbot market – growth, trends and forecast 2020-2025 [Electronic resource]. – Access mode: <https://www.mordorintelligence.com/industry-reports/chatbotmarket>

**ДОДАТОК А**  
**Програмний код**

### Лістинг А.1 – Програмний код чатботу

```

import telebot
from telebot import types

from io import Bytesio
from PIL import image
import requests

import math
import sqlite3

from geocoder_coords import coordToAddr, addrToCoord
from static_map_pass import create_static_map_ride

token = "5027922228:AAerBKL VAT95c5MHscdYTi2Q"
bot = telebot.TeleBot(token)

#Зведення початкової команди
@bot.message_handler(commands=["begin"])
def begin(message):
    name = message.text
    bot.send_message(message.chat.id, "Привіт <b>{first_name}</b> Супер,
        тепер мені потрібен Ваш номер. Введіть його в форматі
        380XXXXXXXXX або натисніть на кнопку
        внизу".format(message.from_user.first_name),
        parse_mode='HTML', reply_markup=types.ReplykeyLocRemove())

#команда введення телефону
@bot.message_handler(commands=["telephone"])
def telephone(message):
    buttonPhone = types.keyLocButton(text="Відправити свій номер",
    request_contact=True)
    markUp.add(buttonPhone)
    mes = bot.send_message(message.chat.id, "Згодні надати номер для
    реєстрації?", reply_markup=markUp)
    bot.register_next_step_handler(mes, registration)

#реєстрація водія або пасажир
```

```
def registration(message):
    try:
        inputTelephone = message.contact.telephone_number
        my_db = sqlite3.connect('my_base.db')
        myTrip = my_db.cursor()
        myTrip.execute('SELET * FROM passUser')
        passUser = myTrip.fetchall()

        for user in passUser:
            table_telephone = user[1]
            if table_telephone == inputtelephone:
                buttonAction = types.ReplykeyLocMarkup(resize_keyLoc=True, one_time_keyLoc=True)
                buttonAddride = types.keyLocButton(text="Нова поїздка")
                buttonHistoryWay = types.keyLocButton(text="Мої поїздки")

                buttonAction.add(buttonHistoryWay)
                buttonAction.add(buttonAddride)
                mess = bot.send_message(message.chat.id, "обрати дію", reply_markup=buttonAction)
                bot.register_next_step_handler(mess, chooseActionPas, inputtelephone, message.chat.id)
                return "

        myTrip.execute('SELECT * FROM driverUser')
        driverUser = myTrip.fetchall()
        print(driverUser)
        for user in driverUser:
            table_telephone = user[1]
            if table_teteletelephone == inputteteletelephone:
                buttonAction = types.ReplykeyLocMarkup(resize_keyLoc=True, one_time_keyLoc=True)
                buttonChooseride = types.keyLocButton(text="обрати поїздку")
                buttonSetting = types.keyLocButton(text="Налаштування")
                buttonAction.add(buttonChooseride)
                buttonAction.add(buttonSetting)

                mess = bot.send_message(message.chat.id, "обрати дію"
```

```

reply_markup=buttonAction)
    bot.register_next_step_handler(mess, buttonActionTaxi, inputtelephone,
message.chat.id)

    return "

    buttonSetting = types.ReplykeyLocMarkup(resize_keyLoc=True)
    buttonPas = types.keyLocButton(text="Пасажир")
    buttonTaxiDr = types.keyLocButton(text="Водій")
    buttonSetting.add(buttonPas)
    buttonSetting.add(buttonTaxiDr)

    mess = bot.send_message(message.chat.id, "оберіть хто ви",
reply_markup=buttonSetting)
    bot.register_next_step_handler(mess, chooseSetting, inputtelephone)

except:
    mess = bot.send_message(message.chat.id, "Помилка. Надайте свій номер
телефона")
    bot.register_next_step_handler(mess, registration)

#вибір поїздки з "Мої поїздки" або "Нова поїздка"
@bot.message_handler(content_types=['text'])
def chooseActionPas(message, user_telephone, idUser):
    if message.text == 'Мої поїздки':
        my_db = sqlite3.connect('my_base.db')
        myTrip = my_db.cursor()
        myTrip.execute('SELECT * FROM rides')
        rides = myTrip.fetchall()

        for ride in rides:
            if ride[1] == user_telephone:

                firstPoint = coordToAddr(ride[2], ride[3])
                secondPoint = coordToAddr(ride[4], ride[5])
                bot.send_message(message.chat.id,
                                f"<i><b>Замовлення
№{ride[0]}</b></i>\n\n<i><b>Начальна
{firstPoint}</b></i>\n\n<i><b>Кінцева
{secondPoint}</b></i>\n\n<i><b>Відстань:</b></i> {ride[7]} м\n\n<i><b>Час
руху:</b></i> {ride[8]} хв\n\n<b>Ціна:</b> {ride[6]} □", parse_mode='HTML',
reply_markup=types.ReplykeyLocRemove())

```

```

elif message.text == 'Нова поїздка':
    keyLoc = types.ReplykeyLocMarkup(resize_keyLoc=True)
    buttonLocation = types.keyLocButton(text="      моє місцезнаходження",
request_location=True)
    keyLoc.add(buttonLocation)
    mess = bot.send_message(message.chat.id, "Надіслати моє місцезнаходження
", reply_markup=keyLoc)
    bot.register_next_step_handler(mess, geo_location, user_telephone, 'Пасажир')

```

#Вибір замовлення для водія

```
@bot.message_handler(content_types=['text'])
```

```
def buttonActionTaxi(message, user_telephone, idUser):
```

```

    if message.text == 'обрати поїздку':
        my_db = sqlite3.connect('my_base.db')
        myTrip = my_db.cursor()
        print(user_telephone)
        myTrip.execute(f'SELECT * FROM driverUser')
        driverUser = myTrip.fetchall()
        for taxiDriver in driverUser:
            if taxiDriver[1] == user_telephone:
                print(taxiDriver)
                keyLoc = types.ReplykeyLocMarkup(resize_keyLoc=True)
                buttonLocation = types.keyLocButton(text="      моє місцезнаходження",
request_location=True)
                keyLoc.add(buttonLocation)
                mess = bot.send_message(message.chat.id, "Надати моє
місцезнаходження ", reply_markup=keyLoc)
                bot.register_next_step_handler(mess, geo_location, user_telephone,
'Tаксист', model=taxiDriver[2], numbers=taxiDriver[3], photo_car=taxiDriver[-2])
                break

```

# налаштування інформації про водія

```
@bot.message_handler(content_types=['text'])
```

```
def chooseSetting(message, user_telephone):
```

```

    if message.text == 'Водій':
        mess = bot.send_message(message.chat.id, "обрати марку машини",
reply_markup=types.ReplykeyLocRemove())
        bot.register_next_step_handler(mess, model, user_telephone)
    elif message.text == 'Пасажир':
        my_db = sqlite3.connect('my_base.db')
        myTrip = my_db.cursor()

```

```
formula = "INSERT INTO passUser ('telephone', 'idUser') VALUES (?,?)"
myTrip.execute(formula, (user_telephone, message.chat.id))
my_db.commit()
```

```
keyLoc = types.ReplykeyLocMarkup(resize_keyLoc=True)
buttonLocation = types.keyLocButton(text="      моє місцезнаходження",
request_location=True)
keyLoc.add(buttonLocation)
mess = bot.send_message(message.chat.id, "Надіслати моє місцезнаходження",
reply_markup=keyLoc)
bot.register_next_step_handler(mess, geo_location, user_telephone, 'Пасажир')
```

```
@bot.message_handler(content_types=['text'])
def avto_number(message, telephone):
    model = message.text
    mess = bot.send_message(message.chat.id, "Введіть номер машини",
reply_markup=types.ReplykeyLocRemove())
    bot.register_next_step_handler(mess, numbers, user_telephone, model)
```

#визначення геолокації

```
@bot.message_handler(content_types=['text'])
def geo_location(message, telephone, job, model=None, numbers=None,
photo_car=None):
    latitude = message.location.latitude
    longitude = message.location.longitude
    tripLeng = {}
```

```
    addrLocation = coordToAddr(longitude, latitude)
    bot.send_message(message.chat.id, addrLocation,
reply_markup=types.ReplykeyLocRemove())
```

```
my_db = sqlite3.connect('base.db')
myTrip = my_db.cursor()
```

```
if job == 'Таксист':
    ln = 0
    myTrip.execute(f'SELECT * FROM driverUser')
    driverUser = myTrip.fetchall()
    for driver in driverUser:
```

```

    if driver[1] != teletelephone:
        ln += 1
    if ln == len(driverUser):

        formula = "INSERT INTO driverUser ('telephone', 'model', 'numbers',
'longitude', 'latitude', 'photo', 'idUser') VALUES (?, ?, ?, ?, ?, ?)"
        myTrip.execute(formula, (etelephone, model, numbers, longitude, latitude,
photo_car, message.chat.id))
        my_db.commit()

        my_db = sqlite3.connect('my_base.db')
        myTrip = my_db.cursor()

rides = myTrip.fetchall()
for us in rides:
    user = us

    firstPoint = coordToAddr(user[2], user[3])
    secondPoint = coordToAddr(user[4], user[5])
    bot.send_message(message.chat.id,
        f"<i><b>Замовлення
№{user[0]}.</b></i>\n\n<i><b>Начальна
{firstPoint}\n\n<i><b>Кінцева
{secondPoint}\n\n<i><b>Відстань:</b></i> {user[7]}
пункту:</b></i> {user[8]} хв\n\n<b>Ціна:</b> {user[6]} □", parse_mode='HTML',
reply_markup=types.ReplykeyLocRemove())

    mess = bot.send_message(message.chat.id, "Введіть номер замовлення",
parse_mode='HTML', reply_markup=types.ReplykeyLocRemove())
    bot.register_next_step_handler(mess, chooseTrip)

elif job == 'Пасажир':
    mess = bot.send_message(message.chat.id, "<b>Куди їдемо?</b>",
parse_mode='HTML', reply_markup=types.ReplykeyLocRemove())
    bot.register_next_step_handler(mess, wayHome, telephone, longitude,
latitude)

#обрати замовлення для водія
@bot.message_handler(content_types=['text'])
def chooseTrip(message):
    numTrip = message.text

```

```

my_db = sqlite3.connect('base.db')
myTrip = my_db.cursor()
myTrip.execute('SELECT * FROM rides')
users = myTrip.fetchall()

pass = []
for us in users:
    if us[0] == int(numTrip):
        pass.append(us)

firstPoint = coordToAddr(pass[0][2], pass[0][3])
secondPoint = coordToAddr(pass[0][4], pass[0][5])
bot.send_message(message.chat.id, f"<i><b>Номер                пасажира:
{pass[0][1]}.</b></i>\n\n<i><b>Начальна                точка:</b></i>
{firstPoint}\n\n<i><b>Кінцева                точка:</b></i>
{secondPoint}\n\n<i><b>Відстань:</b></i>                {pass[0][7]} м\n\n<i><b>Час
pyxy:</b></i>                {pass[0][8]} хв\n\n<b>Ціна:</b>                {pass[0][6]} □",
parse_mode='HTML', reply_markup=types.ReplykeyLocRemove())

create_static_map_ride(f'{pass[0][2]},{pass[0][3]}')
bot.send_photo(message.chat.id, open('11mapPoint.png', 'rb'));
my_db = sqlite3.connect('_mybase.db')
myTrip = my_db.cursor()

user_taxi = myTrip.fetchall()
src_photo_car = user_taxi[0][6]

bot.send_photo(pass[0][-1], open(photo_car, 'rb'));

myTrip.execute(sql, (int(numTrip),))
my_db.commit()

@bot.message_handler(content_types=['text'])
def wayHome(message, telephone, beginLongitude, beginLatitude):
    goLocation = message.text
    endLongitude, endLatitude = [float(x) for x in addrToCoord(goLocation).split(' ')]

    mess = bot.send_message(message.chat.id, "<b>Запропонуйте ціну □.</b>",
parse_mode='HTML', reply_markup=types.ReplykeyLocRemove())
    bot.register_next_step_handler(mess, priceWay, telephone, beginLongitude,
beginLatitude, endLongitude, endLatitude)

```

```

@bot.message_handler(content_types=['text'])
def priceWay(message, telephone, beginLongitude, beginLatitude, endLongitude,
endLatitude):
    priceWay = int(message.text)
    x1, y1 = beginLongitude, beginLatitude
    x2, y2 = endLongitude, endLatitude
    a = math.radians((y1 + y2) / 2)
    x = math.cos(a)
    n = abs(x1 - x2) * 111000 * x
    n2 = abs(y1 - y2) * 111000
    lenWay = round(math.sqrt(n * n + n2 * n2))
    timeWay = round(lenWay / (40 * 1000) * 60)
    firstPoint = coordToAddr(beginLongitude, beginLatitude)
    secondPoint = coordToAddr(endLongitude, endLatitude)

    bot.send_message(message.chat.id,
    замовлення</b></i>\n\n<i><b>Начальна
    {firstPoint}\n\n<i><b>Кінцева
    {secondPoint}\n\n<i><b>Відстань:</b></i>          {lenWay}          м\n\n<i><b>Час
    руху:</b></i>          {timeWay}          хв\n\n<b>Ціна:</b>          {priceWay}          □",
    parse_mode='HTML', reply_markup=types.ReplykeyLocRemove())

    my_db = sqlite3.connect('my_base.db')
    myTrip = my_db.cursor()
    formula = "INSERT INTO rides ('teletelephone', 'beginLongitude',
'beginLatitude', 'endLongitude', 'endLatitude', 'price', 'lenWay', 'timeWay', 'idUser')
VALUeS (?,?,?,?,?,?,?,?,?)"
    myTrip.execute(formula, (teletelephone, beginLongitude, beginLatitude,
endLongitude, endLatitude, priceWay, lenWay, timeWay, message.chat.id))
    my_db.commit()

if __name__ == '__main__':
    bot.polling(none_stop=True, interval=0)

```

**ДОДАТОК Б**  
**Слайди презентації**

# ДОСЛІДЖЕННЯ ЗАПИТІВ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ ЧАТБОТУ ТАКСІ.

Автор: КНТ-110м

Назаренко Н.В.

Керівник: к.т.н. доцент

Касьян М.М.



Рисунок Б.1 – Слайд №1

## МЕТА, ОБ'ЄКТ, ПРЕДМЕТ І ЗАДАЧІ ДОСЛІДЖЕННЯ

Мета даної магістерської роботи – розробка чатботу для замовлення таксі.

Реалізація поставленої мети передбачає вирішення наступних завдань:

- провести аналіз месенджерів та сучасних чатботів таксі;
- дослідити методи розробки чатботів;
- дослідити переваги використання чатботів ніж мобільних додатків;
- розробити концептуальну модель застосунку;
- провести аналіз існуючого інструментарію проектування чатботів;
- розробити програмне забезпечення чатботу;
- проаналізувати результати запитів.

Предметом дослідження дипломної кваліфікаційної роботи магістра є розробка бота у месенджері.



Рисунок Б.2 – Слайд №2

## АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

| Месенджери | Відсоток користувачів, % | Найпопулярніший вік користувачів, |
|------------|--------------------------|-----------------------------------|
| Viber      | 83                       | 30-44                             |
| Messenger  | 54                       | 16-29                             |
| Skype      | 44                       | 45-59                             |
| Telegram   | 19                       | 16-29                             |
| WhatsApp   | 18                       | 16-29                             |

3/14

Рисунок Б.3 – Слайд №3

## АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

| Переваги                                  | Опис                                                                                                                               |
|-------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------|
| оптимізувати роботу                       | Бот відповідає на поширені запитання відвідувачів сайту, перекладаючи їх на фахівців лише для вирішення нестандартних завдань      |
| скоротити тимчасові трудовитрати          | Одна програма, яка вирішує більшу частину питань дає змогу скоротити до мінімуму штат співробітників, які приймають дзвінки        |
| цілодобова підтримка                      | Організувати цілодобову взаємодію із клієнтами                                                                                     |
| підвищити рівень продажів                 | Обслуговування клієнтів без черг                                                                                                   |
| персоналізувати спілкування з клієнтом    | Зберігаючи історію повідомлень чатбот може індивідуально вирішувати проблеми користувача                                           |
| виключити людський чинник                 | Його відповіді завжди мають задану тональності та містять конкретні грамотні вирази.                                               |
| аналізувати та систематизувати результати | За запитом система забезпечить вас аналітичною інформацією та цифрами про результати виконаної роботи, систематизує процес продажу |

4/14

Рисунок Б.4 – Слайд №4

## ПОРІВНЯННЯ ПОПУЛЯРНИХ МЕСЕНДЖЕРІВ ДЛЯ ЧАТБОТІВ

| Параметри                | Facebook                                                       | Telegram                                                       | Skype                                      | Viber                             |
|--------------------------|----------------------------------------------------------------|----------------------------------------------------------------|--------------------------------------------|-----------------------------------|
| Форма використання       | Умовно безкоштовна                                             | Безкоштовна                                                    | Умовно безкоштовна                         | Умовно безкоштовна                |
| Доступ до чатбота        | Доданий як новий користувач, за підпискою, вбудований в діалог | Доданий як новий користувач, за підпискою, вбудований в діалог | Доданий як новий користувач, за підпискою, | За підпискою, вбудований в діалог |
| Протоколи передачі даних | https                                                          | https                                                          | https                                      | https                             |
| Інтерфейс користувача    | Текстовий, кнопковий, голосовий                                | Текстовий, кнопковий, голосовий                                | Текстовий, кнопковий, голосовий            | Текстовий, кнопковий, голосовий   |
| Механізм зв'язку         | webhook                                                        | Webhook, polling                                               | webhook                                    | webhook                           |
| Формати передачі даних   | JSON<br>Multipart/form-data                                    | JSON, URL, query<br>Multipart/form-data                        | JSON<br>Multipart/form-data                | JSON<br>Multipart/form-data       |

5/14

Рисунок Б.5 – Слайд №5

## МЕТОДИ РОЗРОБКИ ЧАТБОТІВ

| Програмування самостійно                                                                                                                                                                                                                                                                                              | Використання конструктора                                                                                                                                                                                                                                                                                                                               |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Переваги</b>                                                                                                                                                                                                                                                                                                       |                                                                                                                                                                                                                                                                                                                                                         |
| <ul style="list-style-type: none"> <li>- інтеграція складних унікальних функцій;</li> <li>- вибір будь-яких платформ та технічних рішень;</li> <li>- розробка інтерфейсу користувача, відповідного очікуванням ЦА;</li> <li>- швидке виправлення недоліків, технічне обслуговування та покращення чатбота.</li> </ul> | <ul style="list-style-type: none"> <li>- легкий старт за допомогою відеоінструкцій, баз знань та допомоги техпідтримки;</li> <li>- швидкість створення чат-ботів;</li> <li>- можливість спробувати новий інструмент безкоштовно або за умовні \$5-10 на місяць;</li> <li>- проста інтеграція зі сторонніми сервісами, якщо вона передбачена.</li> </ul> |
| <b>Недоліки</b>                                                                                                                                                                                                                                                                                                       |                                                                                                                                                                                                                                                                                                                                                         |
| <ul style="list-style-type: none"> <li>- вартість;</li> <li>- необхідність підтримувати працездатність хостингу та сервера;</li> <li>- тривалість підготовки, реалізації та тестування;</li> <li>- складності при зміні розробника.</li> </ul>                                                                        | <ul style="list-style-type: none"> <li>- не на всіх платформах можна експортувати передплатників;</li> <li>- обмежена кількість месенджерів та інтеграцій із зовнішніми сервісами;</li> <li>- розробники довго впроваджують нові потрібні функції;</li> <li>- залежність від сторонньої служби підтримки.</li> </ul>                                    |

6/14

Рисунок Б.6 – Слайд №6

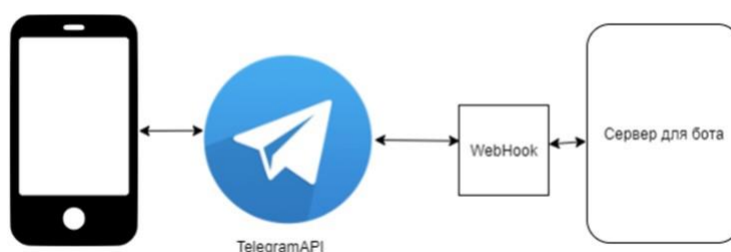
## ДОСЛІДЖЕННЯ ЗАПИТІВ РОЗРОБКИ ЧАТБОТІВ ЧИ МОБІЛЬНИХ ДОДАТКІВ

| Параметри                              | Чат-бот                                         | Мобільний додаток                                        |
|----------------------------------------|-------------------------------------------------|----------------------------------------------------------|
| Інсталяція                             | Не потрібна. Достатньо знайти бота в месенджері | Потрібна                                                 |
| Використання фізичної пам'яті пристрою | -                                               | +                                                        |
| Авторизація                            | Достатньо мобільного номеру                     | Необхідні дані профілю                                   |
| Швидкість розробки                     | Від 2 тижнів                                    | Від 2 місяців                                            |
| Оновлення                              | Автоматично разом із месенджером                | Самостійно оновлювати при внесенні змін або оновленні ОС |
| Топ мов програмування                  | Python, Java, C, PHP, Clojure, Ruby             | Python, Swift, JS, C#, Java, Objective-C                 |
| NLP                                    | +                                               | -                                                        |
| Анімація                               | -                                               | +                                                        |
| Інтерфейс                              | Розмовний                                       | Унікальний дизайнерський                                 |
| Платформа                              | Месенджери<br>(Facebook, telegram, viber)       | ОС<br>(Android, iOS)                                     |

7/14

Рисунок Б.7 – Слайд №7

## УЗАГАЛЬНЕНА АРХІТЕКТУРА ЧАТБОТА



8/14

Рисунок Б.8 – Слайд №8

# РОЗРОБКА СИСТЕМИ



Рисунок 1 – Діаграма варіантів використання «Пасажир»



Рисунок 2 – Діаграма варіантів використання «Водій»

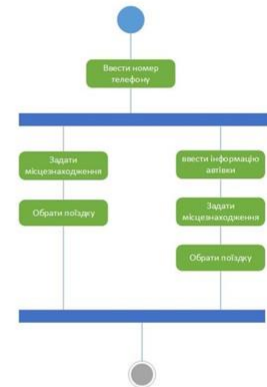
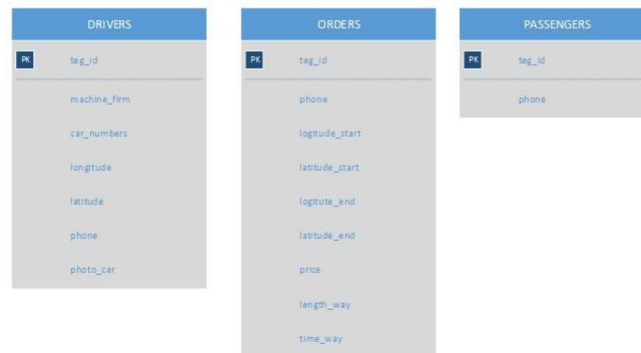


Рисунок 3– Діаграма станів системи чатботу таксі

9/14

Рисунок Б.9 – Слайд №9

## ДІАГРАМА СТРУКТУРИ БАЗИ ДАНИХ



10/14

Рисунок Б.10 – Слайд №10

# ПРОГРАМНА РЕАЛІЗАЦІЯ ЧАТБОТУ

Для розробки чатботу таксі можна використовувати такі мови програмування, як Java, Python, C, PHP. Розглянуті переваги та недоліки кожної мови, так для розробки чатботу обрано Python.

| Модулі системи     | Технологія                                                                              |
|--------------------|-----------------------------------------------------------------------------------------|
| База даних         | SQLite                                                                                  |
| Мова програмування | Python                                                                                  |
| Реєстрація чатбота | Реєстрація чатбота в Telegram проведена за допомогою віртуального помічника @BotFather. |
| Сервер             | Heroku                                                                                  |

11/14

Рисунок Б.11– Слайд №11

## ІНТЕРФЕЙС ЧАТБОТУ

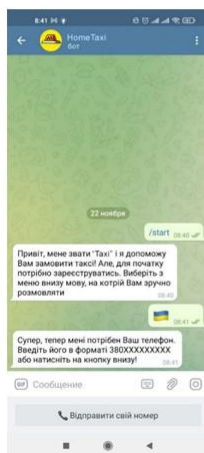


Рисунок 4 – Початок роботи з ботом



Рисунок 5 – Авторизація користувача

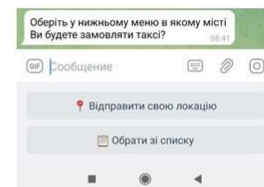


Рисунок 6 – Замовлення таксі

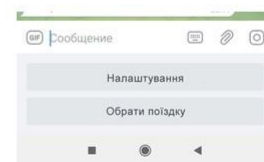


Рисунок 7 – Меню пасажирів

12/14

Рисунок Б.12– Слайд №12

## ТЕСТУВАННЯ ЧАТБОТУ

| № | Параметр                                                      | Значення |
|---|---------------------------------------------------------------|----------|
| 1 | Середня швидкість відгуку чатбота на повідомлення /start, сек | 0,1      |
| 2 | Середня тривалість сесії замовлення таксі, хв                 | 3        |
| 3 | Середня швидкість відображення карти, сек                     | 1,5      |
| 4 | Середня швидкість запитів до БД, сек                          | 0,3      |
| 5 | Коректна обробка повідомлень водія                            | Успішно  |
| 6 | Коректна обробка повідомлень пасажирів                        | Успішно  |
| 7 | Коректне відображення діалогу                                 | Успішно  |
| 8 | Коректна робота кнопок                                        | Успішно  |

13/14

Рисунок Б.13– Слайд №13

## ВИСНОВКИ

В даній роботі вирішені наступні завдання:

- проведено аналіз месенджерів та сучасних чатботів таксі;
- досліджені методи розробки чатботів;
- досліджені переваги використання чатботів ніж мобільних додатків;
- розроблена модель застосунку;
- проведено аналіз існуючого інструментарію проєктування чатботів;
- розроблено програмне забезпечення чатботу;
- проведено тестування чатботу.

В даній кваліфікаційній роботі можна зробити висновок, що поставлені задачі були вирішені.

Практична цінність роботи полягає в тому, що на підставі аналізу предметної області та дослідження запитів до створення чатботів розроблено програмний застосунок, призначений для підвищення продуктивності роботи підприємства. Розроблений телеграм-бот дозволяє автоматизувати роботу таксі.

14

Рисунок Б.14– Слайд №14