

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК
СЕКЦІЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ПРОЕКТУВАННЯ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему: «Інформаційний чат-бот для соціальної мережі Telegram з використанням Google Assistant та Google Search API»

за спеціальністю 122 «Комп'ютерні науки та інформаційні технології»,
освітньо-професійна програма «Інформаційні технології проектування»

Виконавець роботи: студент групи ІТ-51-6 Кіншаков Едуард Віталійович

**Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою**

«___» _____ 2019 р.

Науковий керівник

_____ професор Лавров Є. А.
(підпис) (науковий ступінь, вчене звання, прізвище та ініціали)

Голова комісії

_____ Шифрін Д. М.
(підпис) (науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____
(підпис)

Суми-2019

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук
Секція інформаційних технологій проектування
Спеціальність 122 «Комп'ютерні науки та інформаційні технології»
Освітньо-професійна програма «Інформаційні технології
проектування»

ЗАТВЕРДЖУЮ

Зав. секцією ІТП

_____ В. В. Шендрик

«_____» _____ 2019 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

Кіншакова Едуарда Віталійовича

1 Тема роботи «Інформаційний чат-бот для соціальної мережі Telegram з використанням Google Assistant та Google Search API»

керівник роботи Лавров Євгеній Анатолійович,

затверджені наказом по університету від «17» травня 2019 р. №0834-III

2 Строк подання студентом роботи «3» червня 2019 р.

3 Вхідні дані до роботи інформаційний чат-бот в месенджері Telegram

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) аналіз стану досліджень з питань розробки чат-боту, постановка задачі та методи дослідження, проектування інформаційного чат-боту, розробка інформаційного чат-боту

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) актуальність, мета і задачі, дослідження аналогів, функціональні вимоги до чат-боту, чат-бот види, телеграм чому, переваги Telegram від інших месенджері, архітектура чат-боту, засоби реалізації, схема роботи чат-боту, функціональне моделювання системи, моделювання варіантів використання чат-боту, демонстрація роботи чат-боту, апробація, акт впровадження, перспективи досліджень, висновки.

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
<i>Аналіз стану досліджень</i>	<i>Лавров Є.А.</i>		
<i>Постановка задачі та методи</i>	<i>Лавров Є.А.</i>		
<i>Проектування інформаційного чат-боту</i>	<i>Лавров Є.А.</i>		
<i>Розробка інформаційного чат-боту</i>	<i>Лавров Є.А.</i>		

7.Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Дослідження ринку чат-ботів	15.12.19-20.01.19	
2	Мета та задачі	20.01.19-31.01.19	
3	Формування технічних вимог	05.02.19-12.02.19	
4	Аналіз документації Telegram	13.02.19-25.02.19	
5	Аналіз документації Google Search та Google Assistant	27.02.19-21.03.19	
6	Аналіз документації Dialogflow	24.03.19-02.04.19	
7	Аналіз документації Speech-to-text recognition	05.04.19-14.04.19	
8	Проектування алгоритму роботи чат-боту	16.04.19-20.04.19	
9	Розробка чат-боту	21.04.19-01.05.19	
10	Формування документації	03.05.19-28.05.19	

Студент

(підпис)

Кіншаков Е.В..

Керівник роботи

(підпис)

професор Лавров Є.А.

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ СТАНУ ДОСЛІДЖЕНЬ З ПИТАНЬ РОЗРОБКИ ЧАТ-БОТУ.....	8
1.1 Дослідження ринку чат-ботів	8
1.2 Аналіз аналогів та визначення проблеми.....	12
2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ.....	17
2.1 Мета та задачі	17
2.2 Методи реалізації	19
3 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОГО ЧАТ-БОТУ	23
3.1 Вимоги до інформаційної системи	24
3.2 Структура інформаційної системи	24
3.3 Моделювання інформаційної системи	26
3.4 Моделювання варіантів використання чат-боту.....	30
4 РОЗРОБКА ІНФОРМАЦІЙНОГО ЧАТ-БОТУ	32
4.1 Розробка логічної структури чат-боту	32
4.2 Опис функціонування чат-боту	38
ВИСНОВКИ	48
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	50
Додаток А. Технічне завдання.....	53
Додаток Б. Планування робіт	57
Додаток В. Приклади роботи Чат-боту.....	68
Додаток Г. Інструкція користувача	85
Додаток Д. Лістинг коду	86
Додаток Е. Копії публікацій	91
Додаток Є. Копії грамот переможця конкурсу наукових робіт.....	99
Додаток Є. Копії акту впровадження.....	102

ВСТУП

Актуальність. Пошук інформації є невід'ємною частиною кожного користувача смартфона. Всі верстви населення так чи інакше звертаються до пошукових систем. Використання сучасних месенджерів набуває великого попиту. Одним із таких месенджерів є Telegram. Інформаційна система Telegram є основоположником такої програми, як чат-бот. Тому була поставлена актуальна задача по розробці пошукової системи в месенджері Telegram, інакше кажучи створення комп'ютерної програми чат-бот. Дана програма надаватиме широку можливість для тих, на чиїх девайсах не існує голосових помічників, або оптимізувати пошук до однієї цілої системи за допомогою сервісів компанії Google.

Об'єкт дослідження. Чат-бот, як пошукова система.

Предмет дослідження. Алгоритм пошуку інформації та спілкування в Telegram.

Гіпотеза дослідження. Якщо використати технологію сумісного використання сервісів Google, то можна розробити алгоритми для спілкування та пошуку без використання пошукової системи в смартфоні.

Мета. Розробка інформаційного чат-боту в месенджері Telegram.

Публікації. Оpubліковано 3 наукові роботи. Копії публікацій наведено в додатку Е.

Апробація результатів. Результати доповідались на 3 наукових конференціях:

- Інформатика, математика, автоматика м. Суми, 5-9 лютого 2018 р
- Всеукраїнська науково-практична конференція молодих науковців і студентів «Інтелектуальний потенціал - 2018» , 14-16 листопада 2018р., м. Хмельницький

- Інформатика, математика, автоматика м. Суми, 23-26 квітня 2019
р

Впровадження. Результати впроваджено в виробничий процес Сервісного центру «СумиТехсервіс» ФОП Полунов.

Участь в конкурсах студентських наукових робіт. Результати були представлені на двох конкурсах наукових студентських робіт. Копії дипломів переможця наведено в додатку Є.

1 АНАЛІЗ СТАНУ ДОСЛІДЖЕНЬ З ПИТАНЬ РОЗРОБКИ ЧАТ-БОТУ

1.1 Дослідження ринку чат-ботів

На сьогоднішній день чат-боти як сучасні інструменти комунікацій стали широко використовуватися в багатьох сферах життєдіяльності людини з метою встановлення контакту з користувачами мережі Інтернет. Найбільшу популярність чат-боти отримали, коли почалося їх використання в месенджерах і соціальних мережах (наприклад, в Telegram, Viber, Facebook,). Завдяки новому інструменту віртуальної комунікації з'явилася можливість дізнаватися про спеціальні пропозиції онлайн, отримувати посилання свіжих новин та спеціальних пропозицій в сфері товарів і послуг, здійснювати більш складні операції. В даний час найбільше значення чат-боти мають у формуванні такої моделі поведінки, яка буде максимально наближена до людської.

Під чат-ботами в загальному сенсі зазвичай розуміють спеціальні програми, які здійснюють інтернет-спілкування найчастіше з одним або декількома користувачам, використовуючи штучний інтелект. Сучасні вигди інтернет-комунікацій здатні виступати в якості віртуального співрозмовника і повторювати і відтворювати письмовий набір знаків людини, надаючи запрограмовану відповідь на задані питання. В силу того, що контакт з чат-ботами є не тільки цікавим заняттям, але і корисним для людини з точки зору отримання нової інформації і рішення споживчих завдань, варто відзначити, що інтернет-аудиторія прийняла їх появу доброзичливо.

Багато організацій в свою чергу побачили в інтересі аудиторії до чат-ботів можливість для власного просування і стали розробляти власні чат-боти, використовуючи їх для залучення уваги споживачів до компанії за

рахунок використання сучасного інструменту комунікації. Чат-бот стає каналом просування або частиною рекламної кампанії як для великих брендів, так і для новачків. Потрібно позначити таку тенденцію: спочатку деякі компанії почали використовувати їх тільки як визначення трендів, тобто стала формуватися певна мода на використання чат-ботів без урахування їх практичної значущості. Можливості інтерфейса програми були досить зрозумілі і прості. На основі заздалегідь завантажених шаблонів, запам'ятовуванні слів, введених самим користувачем, і відповіді на відповідний запит був заснований алгоритм роботи чат-бота.

Чат-боти можуть використовуватися для спілкування між користувачами, а також в розважальних, інформаційних цілях (чат-боти можуть повідомити погоду, курс валют, записати клієнта на прийом, здійснити реєстрацію дзвінків) і в службах підтримки.

Поняття «чат-бот» походить від двох англійських слів: *tochat* - невимушена розмова в мережі Інтернет, *bot (robot)* - скорочено робот, з чого випливає, що це роботи, призначені для здійснення комунікацій з користувачами в мережі Інтернет, які виконують дії відповідно до закладеному сценарію. Така програма інтернет-комунікації як чат-бот заснована на сучасних технологіях. В її основі лежать такі інформаційно-комунікаційні технології, як штучний інтелект, нейронні мережі, що дозволяє наближати комунікації чат-ботів до людського віртуального спілкування і вирішувати з їх допомогою комерційні та маркетингові задачі [1].

Виходячи з поточної оцінки стану ринку, чат-боти, особливо для месенджерів – дуже перспективний напрям, який в даний час переживає бурхливе зростання (рис. 1.1).

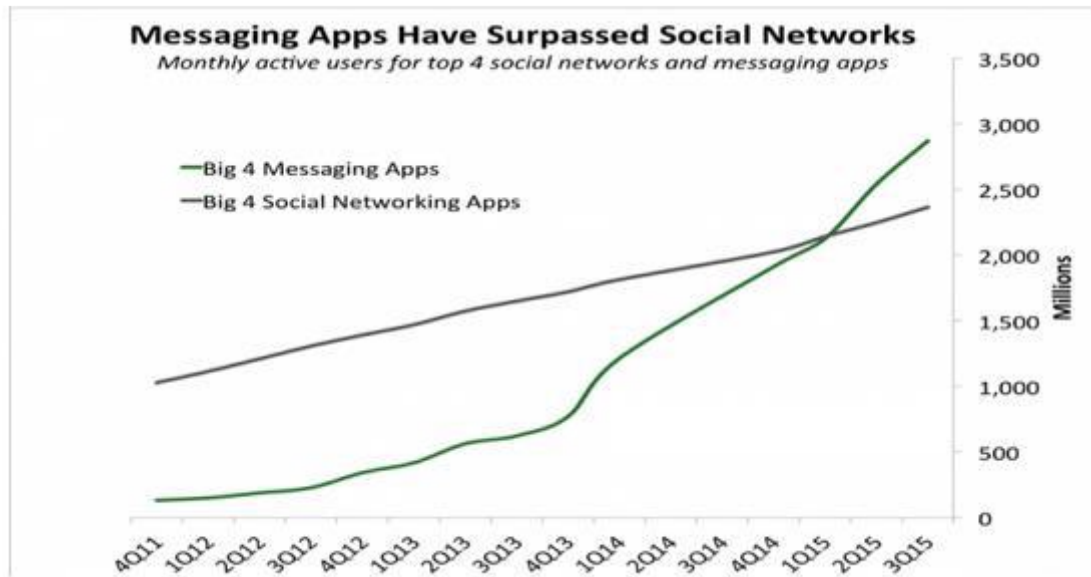


Рисунок 1.1 – Тенденції ринку чат-ботів

За оцінкою аналітиків у найближчому майбутньому чат - боти будуть мати все більшу значимість. Вони можуть повністю замінити класичні пошукові движки і соціальні мережі [2]. Переваги ботів забезпечують зручність людино-комп'ютерної взаємодії, швидкість реакції, можливість їх налаштування під користувача. Прикладом може бути Facebook чат-бот M, який в поточний момент знаходиться в режимі бета-тестування – це персональний цифровий асистент всередині месенджера, що виконує прості задачі (в т.ч. рекомендації з надання послуг). Технічно Facebook-M оснований на сучасних технологіях машинного навчання (глибинні нейромережі). Крім Facebook, в напрямку розвитку технологій чат-ботів активно рухається Google і Apple [3].

Основні напрямки використання чат-ботів:

1. У сфері бізнесу та електронної комерції. У даних сферах чат-боти мають найбільший функціонал і можуть вирішувати багато завдань, включаючи онлайн-консультування, рекламні комунікації, продажу та ін. [4].
2. У сфері отримання інформації. Такі чат-боти використовуються в електронних засобах масової інформації для надання користувачам цікавої їм

новинної інформації. Наприклад, доктор-бот може порадити лікування на основі озвучених симптомів або підказати адреси найближчих лікарень [5].

3. У сфері навчання іноземним мовам та інших дисциплін, так звані бот-вчителя. Вони надають навчальний матеріал (найчастіше це невеликі за обсягом тексти) з дисципліни, оцінюють знання учнів. В даний час чат-боти допомагають самостійно вивчати іноземні мови [6].

4. У сфері розваг. Чат-боти в сфері розваги можуть вести діалог з користувачем, надавати йому цікаву інформацію, грати з ним і виконувати інші розважальні функції [7].

Таким чином, чат-боти активно використовуються в бізнесі, освіті, ЗМІ та сфері розваг, що характеризує їх як досить універсальний інструмент комунікацій. Чат-боти в сфері бізнесу можуть виконувати різні функції, пов'язані із здійсненням внутрішніх та зовнішніх комунікацій в організації: автоматична розсилка документів, повідомлень, призначення зустрічей, обслуговування та консультування клієнтів, з'ясування потреб клієнтів, уявлення відповідей на прості питання, пошук про товар для економії часу і швидкого прийняття рішення про покупку. Чат-боти можуть аналізувати і збирати всю отриману під час комунікацій з користувачами інформацію для подальшого використання з метою поліпшення роботи чат-бота, так як у досліджуваного механізму є функція запам'ятовування отриманої інформації. Тобто чат-боти можуть автоматично збирати дані про клієнтів і при повторному зверненні можуть відтворювати власну базу, щоб не ставити клієнту питання раніше чим надати інформацію про запитовані товари або послугу [8].

Даний дипломний проект буде корисним та актуальним у побутовому використанні. Адже використання боту в інформаційній системі Telegram звужує ресурс використання інших додатків, котрі можуть бути зайвими. В країні у кожного користувача є девайс, але більша частина смартфонів не в змозі підтримувати голосових асистентів. Тому була поставлена задача

автоматизувати роботу пошуку в мережі Інтернет. Він є ефективним та зручним у використанні. Також завдяки зручним надбудовам в месенджері бот має змогу працювати при низькому трафіку, а також підтримувати спілкування з користувачем.

1.2 Аналіз аналогів та визначення проблеми

Перед тим як почати розробляти свій продукт, необхідно було ознайомитись з ринком чат-ботів, адже бот, котрий має будь-який інформаційний напрям або, інакше кажучи, якщо є зворотня відповідь, то він відразу вважається інформаційним. Важливо зазначити, що аналогів серед багатьох телеграм ботів знайдено не було. Навіть якщо вони існують, то безпосередньо це комерційні проекти, котрі зберігаються лише конфіденційно. Оцінювати його серед інших ботів теж неможливо, адже кожний бот виконує певну задачу і працює безпосередньо в своїй предметній області. Оскільки майже всі боти по своїй структурі інтерфейсу є прості та зрозумілі, тому наявність інструкційне є необхідною. Зараз будуть наведені аналоги інформаційних чат-ботів.

Кіноман 2.0 (@kinoman) шукає фільми і серіали, радить рандомні картини для перегляду, стежить за ситуацією на торентах. (рис 1.2) .

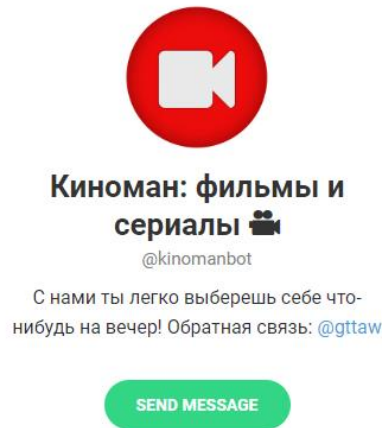


Рисунок 1.2 – Бот Киноман 2.0

MyCookBot (@mycookbot) на жаль, цей бот не вмiє доводити продукти до готового стану за допомогою нейромереж, але він підказує рецепти і вибирає страви з ваших iнгредiєнтiв. Бот зображений на рисунку 1.3.



Рисунок 1.3 – Бот MyCookBot

SoundCloud (@scloud_bot) бот, що iмітує клієнт, музичного сервісу SoundCloud прямо в діалоговому вікні. Все по-дорослому: скачування треків, чарти по жанрам, підключення акаунта (рис 1.4).

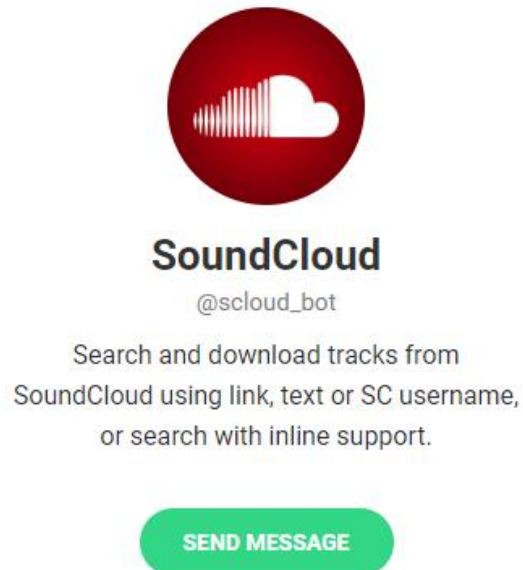


Рисунок 1.4 – Бот SoundCloud

Wolfram Alpha (@WolframBot) вирішує прості математичні задачки. Більше не треба завантажувати дурні математичні калькулятори. Він навіть графіки побудує і відправить. Розпізнає також і інші запити. Бот зображений на рис.1.5.



Рисунок 1.5 – Бот Wolfram Alpha

Дані боти є у відкритому доступі для завантаження на смартфон. Вони є досить корисними у сфері свого використання. Звісно, їх можна назвати інформаційними, адже окрім того, що вони ведуть діалог, вони повертають результат: чи то рецепт, музичний файл або відповідь математичної задачі. У мережі є ще дуже багато різноманітних ботів, і всі вони за своїм змістом унікальні [9].

Тому було обране рішення розробити саме інформаційний бот, котрий позиціонує себе як пошукова система, яка може спілкуватися зі своїм користувачем. Дана пошукава система є дипломним проектом і називається assistant. Вона зображена на рис.1.6.

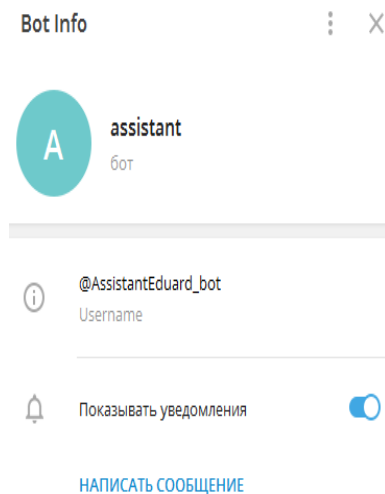


Рисунок 1.6 – Бот assistant

І для того, щоб зрозуміти для якої верстви населення бот буде актуальним та цікавим, необхідно його віднести до певної категорії. Подібні програми мають різні функціонали, тому сфера використання віртуальних помічників постійно підвищується. За цільовим призначенням можна виділити чотири основні типи роботів. Вони показані в табл. 1.1.

Таблиця 1.1 – Відношення до типу чат - бота

Види чат - ботів	Категорії			
	Новини	Квитки	Їжа	Пошуковий
Розважальні	-	-	-	+
Консультанти	-	-	-	-
Помічники	-	-	-	+
CRM-система	-	-	-	-

2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ

2.1 Мета та задачі

Головна мета – розробити інформаційний чат-бот в соціальній мережі Telegram. Дана соціальна мережа є першою, яка запровадила розроблення чат-ботів у месенджері.

Даний бот має можливість виконувати такі функції:

- симуляція живого спілкування з користувачем;
- пошук інформації за допомогою тексту;
- пошук інформації за допомогою голосового повідомлення.

Даний бот може зацікавити не тільки з точки зору реалізації, а й з точки зору подальших вдосконалень. Даний бот має необмежені можливості з вдосконалення, наприклад:

- додати нові мови для розпізнавання;
- додати відображення гарячих новин;
- вдосконалити команди для розпізнавання за допомогою Dialogflow;
- додати персональні моделі поведінки боту (наприклад, відображення ринку криптовалют);
- додавати нові продукти компанії Google та інших компаній та сервісів.

Даний бот повинен мати зручний інструмент пошуку та відображення необхідної інформації.

Виходячи з аналізу предметної області, а також аналогів даної інформаційної системи, можна представити даний бот як інформаційна система в інформаційній системі. Існуючі аналоги дали зрозуміти, що дана

розробка буде актуальною та необхідною у ряді повсякденного використання. Вище зазначені пункти дали зрозуміти, що в Telegram можна знайти бота будь-якого характеру, але не тільки інформаційного з точки зору пошуку та збору інформації.

Telegram Messenger - це програма для обміну повідомленнями, яка працює через Інтернет так само, як WhatsApp та Facebook Messenger. Це означає, що людина може безкоштовно відправляти повідомлення за допомогою з'єднання Wi-Fi або мобільної передачі даних (якщо даних достатньо).

Додаток прив'язується до номера телефону, потім приходить перевіряюче повідомлення з кодом (на мобільній версії для Android код підтягується автоматично). Спілкуватися можна як з конкретними користувачами, які вже встановили додаток, так і організовувати групові чати до 200 чоловік і відправляти запрошення, повідомлення з закликом. Щемість - двісті чоловік в чаті. Розробники вважають це однією з найголовніших фішок свого застосування.

Завдяки захищеності месенджера і захисту від перехоплення додаток став популярним в Гонконзі під час протестів. Спочатку про мітинг домовлялися через Facebook, але цю інформацію легко дістати.

Основним пунктом продажу телеграми є безпека. Це означає, що всі дії, включаючи чати, групи та медіа-зашифровані, не будуть видимі, не будучи розшифрованими першими, навіть якщо вони перехоплені.

Telegram може використовуватися на смартфонах, планшетах, ноутбуках і настільних комп'ютерах. Програма Telegram доступна для Android, iOS, Windows Phone, WindowsNT, MacOS і Linux.

2.2 Методи реалізації

Для реалізації необхідно визначитись з мовою програмування, адже не запрограмувати бота такого рівня неможливо. Були проаналізовані мови програмування і обрано мову Python.

Python - це мова програмування загального призначення, націлена в першу чергу на підвищення продуктивності самого програміста, ніж коду, який він пише. Говорячи простою людською мовою, на Python можна написати практично що завгодно (веб, настільні додатки, ігри, скрипти по автоматизації, комплексні системи розрахунку, системи управління життєзабезпеченням, чат – боти і багато іншого) без значних проблем [10].

Більш того, поріг входження низький, а код багато в чому лаконічний і зрозумілий навіть тому, хто ніколи його не використовував. За рахунок простоти коду подальший супровід програм, написаних на Python, стає легше і приємніше в порівнянні з Java або C ++. З точки зору бізнесу перевага полягає у скороченні витрат і збільшенні продуктивності праці співробітників [11].

Крім сотні тисяч індивідуальних розробників і невеликих компаній, Python підтримують такі гіганти ІТ як:

- Google;
- Dropbox;
- Mozilla;
- Facebook;
- Yandex;
- RedHat;
- Telegram.

І саме головне на даній мові програмування компанія Google та месенджері Telegram надають зручні API на мові Python. Написання боту

здійснювалось в середовищі програмування JetBrains PyCharm Community Edition 2018. Середовище зображено на рис. 2.3.

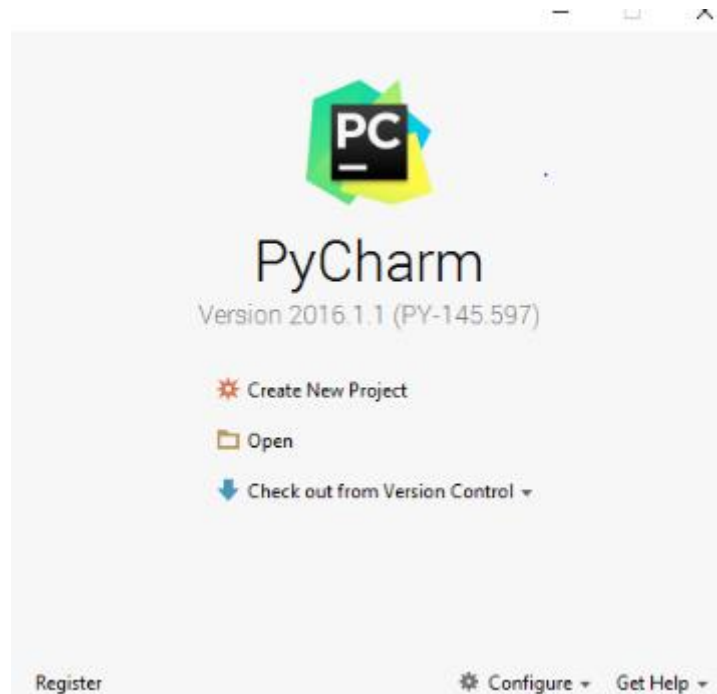


Рисунок 2.3 – JetBrains PyCharm Community Edition 2018

Для побудови будуть використовуватись продукти компанії Google, а саме: Google Assistant, Dialogflow, Google Speech-to-Text, Google Search.

Асистент є голосовим помічником Google. Спочатку це було розширення Google Now, яке було особистим-у той час як розширення на існуючих Google "OK Google" вже мала можливість голосового управління. Спочатку Google Now вмiло витягувала релевантну інформацію: вона знала, де людина працює, її зустрічі і плани подорожей, спортивні команди, які їй сподобалися, і те, що їй цікавило. Він міг представити особисту інформацію, яка мала значення. Google вже давно може замінити Google Now, але помічник чудово працює в одному просторі, поєднуючи ці персоналізовані елементи з широким спектром голосового управління.

Google Assistant підтримує як текстовий, так і голосовий запис, і з задоволенням слiдкує за розмовою, незважаючи на метод запису [12].

Сторінка "OK Google" охоплює голосові команди, голосовий пошук і керування пристроєм, що активується голосом, дозволяючи нагадувати, надсилати повідомлення, перевіряти зустрічі і інше на пристрої Android, як і Siri на Apple i Phone або iPad.

Google Assistant надає такі можливості як:

- керувати своїми пристроями та розумним домом;
- доступ до інформації з календарів та іншої особистої інформації;
- знайти інформацію в Інтернеті: від бронювання ресторанів до напрямків, погоди та новин;
- керувати музикою;
- відтворювати вміст Chromecast або інших сумісних пристроїв;
- запускати таймери та нагадування;
- призначення та відправлення повідомлень;
- відкриття програми на телефоні;
- читати свої сповіщення;
- переклади в режимі реального часу.

Продовження розмови означає, що людині не доведеться говорити "Hey Google" для подальших запитів. Google також може розпізнавати голосові профілі для різних людей, тому він знає, хто з ним розмовляє, і може адаптувати відповіді, що інші системи також починають пропонувати [13].

Dialogflow – це комплекс розгортання для створення розмовних інтерфейсів для веб-сайтів, мобільних додатків, популярних платформ обміну повідомленнями та пристроїв IoT. Його можна використовувати для створення інтерфейсів (наприклад, чат-ботів і розмовних IVR), які дозволяють здійснювати природні та багаті взаємодії між користувачами та бізнесом. Користувачі Dialogflow Enterprise Edition мають доступ до Google Cloud Support і угоди про рівень обслуговування (SLA) для розгортання продукції [14].

Працює за допомогою машинного навчання Google. Звичайне розуміння мови розпізнає намір користувача і витягує попередньо побудовані об'єкти такі, як час, дата і цифри. Людина може навчити свого агента ідентифікувати типи спеціальних об'єктів, надавши невеликий набір прикладів.

Можна також використовувати готові агенти як шаблони. Розгортати на різних платформах і мовах. Dialogflow підтримує 20 мов і інтеграцію з одним натисканням на 14 різних платформ. Залучивши більше аудиторій, будуючи багатомовним агентом з глобальним доступом.

Застосування найперевішних алгоритмів нейронної мережі та глибокого навчання до звуку для розпізнавання мови з безпрецедентною точністю. Хмарна точність від слова до тексту з часом поліпшується, оскільки Google покращує внутрішню технологію розпізнавання мови, яка використовується продуктами Google. Автоматично ідентифікує розмовну мову.

Використовуючи CloudSpeech-to-Textrecognition, можна визначити, якою мовою говорять у висловленні (обмеження до чотирьох мов). Це можна використовувати для голосового пошуку (наприклад, «Яка температура в Парижі?»). Повертає текстову транскрипцію в режимі реального часу для короткого або довгого звуку.

Cloud Speech-to-Text recognition може надсилати текстові результати, відразу ж повертаючи текст, оскільки він розпізнається з потокового аудіо або як говорить користувач. Крім того, CloudSpeech-to-Textrecognition може повертати розпізнаний текст з аудіофайлу, що зберігається у файлі. Він здатний аналізувати короткі і довгі аудіо.

Автоматично транскрибує власні іменники і контекстне форматування. Cloud Speech-to-text recognition, призначений для ефективною роботи з реальною мовою, може точно транскрибувати власні іменники і належним чином формувати мову (наприклад, дати, номери телефонів). Google підтримує

більш 10 власних іменників у порівнянні з кількістю слів у всьому Оксфордському словнику англійської мови [15].

Google Search, який також називається Google Web Search або просто Google – це веб-пошукова система, розроблена компанією Google LLC . Порядок результатів пошуку, повернутих Google, базується, зокрема, на системі рангу пріоритетів, яка називається " PageRank". Пошук Google також надає різні варіанти налаштування пошуку, використовуючи символи, які включають, виключають, визначають та вимагають певної поведінки пошуку, а також надають спеціалізований інтерактивний досвід, наприклад, статус рейсу та відстеження пакетів, прогноз погоди, конверсії валюти, одиниці та час визначення та багато іншого.

Google індексує сотні терабайт інформації з веб-сторінок. Для недоступних веб-сайтів Google надає посилання на кешовані версії сайту, сформовані за допомогою останньої індексації цієї сторінки пошуковою системою. Користувачі також можуть активувати " Безпечний пошук" - технологія фільтрації, спрямована на запобігання появі явних і порнографічних даних у результатах пошуку [16].

3 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОГО ЧАТ-БОТУ

3.1 Вимоги до інформаційної системи

Інформаційна система, яка має вигляд чат-боту, повинна робити пошук різноманітної інформації у месенджері Telegram. Бот повинен мати зручний інтерфейс без зайвих надбудов. Користувач повинен одразу зрозуміти, яким чином виконувати пошук. Дана система оснащена новітніми сервісами компанії Goggle, що надає можливість розробнику зробити якісний, зручний, мобільний, актуальний чат-бот. З аналізу задач, огляду методів та програмного забезпечення можна сформулювати вимоги до інформаційної системи.

Система повинна:

- працювати на всіх версіях месенджера Telegram;
- робити голосові запити щодо пошуку;
- робити текстові запити щодо пошуку;
- вести діалог з системою;
- підтримувати 2 мови для запитів пошуку.

3.2 Структура інформаційної системи

Для розуміння роботи чат-бота та її складових побудована її структура роботи за допомогою абстрактної моделі (рис. 3.1). Система включає в себе сервіси, API Google та месенджер Телеграм, котрий завантажений на девайсі користувача.

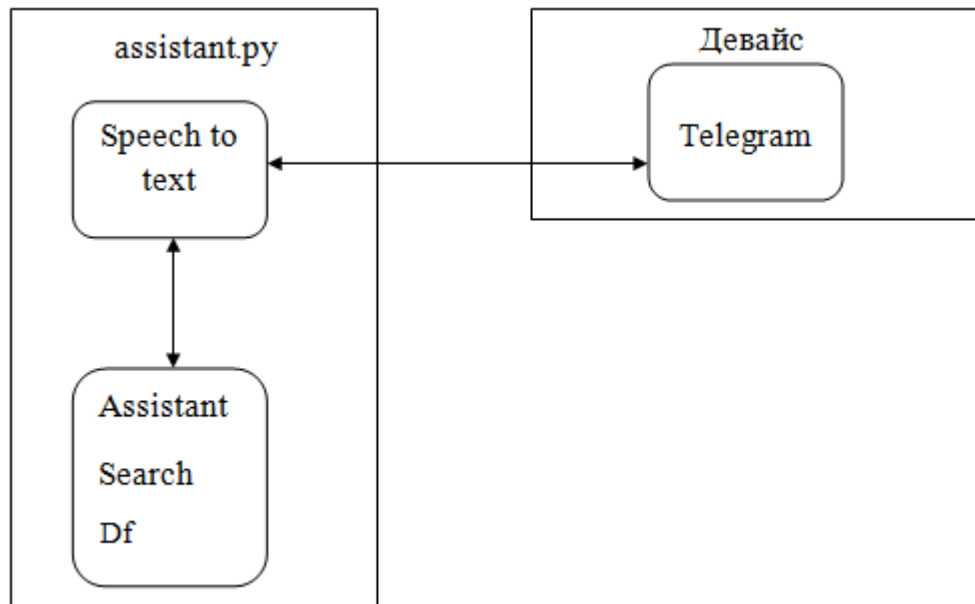


Рисунок 3.1 – Структура чат-боту

В даній структурі чітко можна побачити принцип роботи самого чат-боту. В розділах вище описані всі сервіси Google, які використовуються в даній інформаційній системі. В даній структурі зображені блоки з їх логічними структурами та додатками.

Додаток Telegram відіграє функцію платформи для розробки боту, а також його розміщення. assistant.py - файл з логікою роботи більшості сервісів, котрі є у самому боті. Speech-to-text recognition відіграє роль синтезу мови, котра після обробки аудіо чи тексту робить запит на сервіси Assistant, Search, Dialogflow. Після обробки запиту сервіси надсилають результат вже на девайс, а сам результат буде знаходитись у самому боті, де і був проведений запит. Пошук проводиться у вигляді спілкування.

3.3 Моделювання інформаційної системи

Перед тим, як розробити інформаційну систему, необхідно розробити модель її роботи, продемонструвати усі процеси, послідовність їх виконання та можливі варіанти розвитку подій всередині самої системи. Згідно з цим необхідно побудувати контекстну діаграму IDEF0 та її декомпозицію, а також варіанти використання (usecase).

В першу чергу необхідно побудувати контекстну діаграму IDEF0(рис. 3.2). Головним процесом є створення чат-боту з використанням сервісів Google. На вході маємо API key для доступу до сервісів Google, а також технічне завдання для розробки чат-боту. Відповідно на виході отримуємо вже готову інформаційну систему. Механізмами виступають Speech-to-text-recognition, з якого можна перетворити голос в текст, розробник, який розробляє саму систему, технічне забезпечення, мова програмування python, а також API Google. Усі процеси системи регулюються документацією сервісів Google, які будуть використані при розробці, а також документацією Telegram.

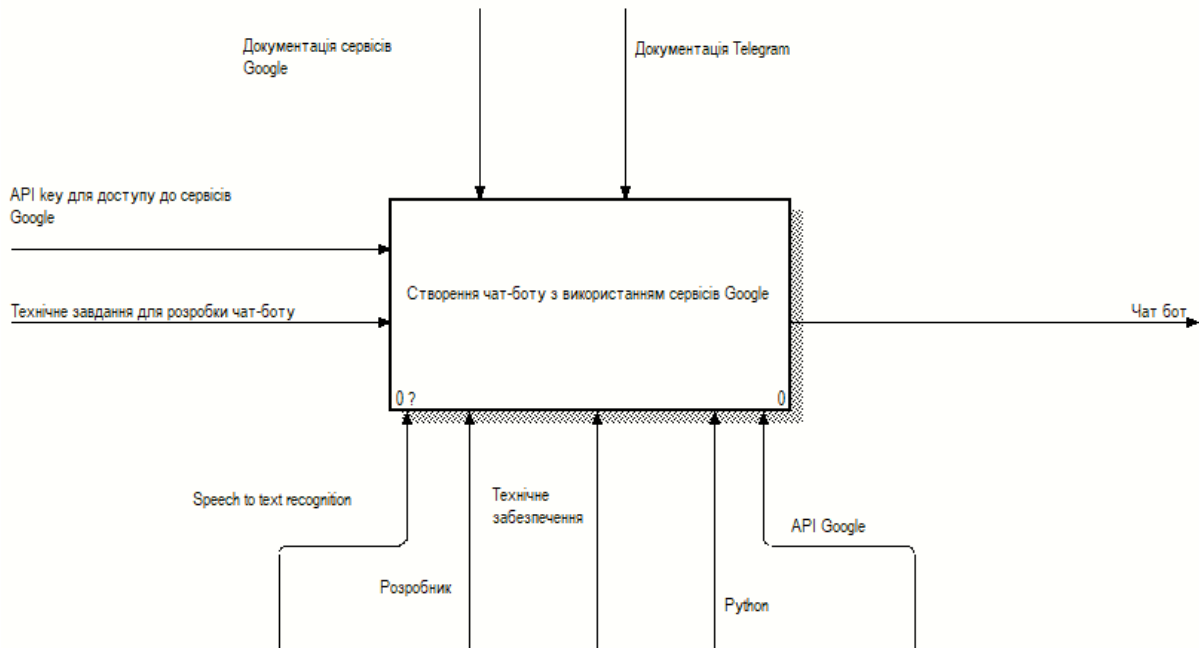


Рисунок 3.2 – Діаграма IDEF0

Контекстна діаграма дає загальний опис системи. Для того щоб побачити повний опис системи, необхідно виконати декомпозицію. Наступна дія дозволить детально ознайомитись з логікою послідовності виконання робіт для реалізації вихідного продукту.

Діаграма А-0 була розбита на три підрівні: отримання доступу до Telegram, підключення сервісів Google, створення алгоритму. Це обумовлено тим, що процес створення функціонального веб-ресурсу поділяється на два етапи:

- Перший етап: надання доступу до месенджера для здійснення маніпуляцій .

- Другий етап: з'єднання з кожним сервісом окремо.

- Третій етап: створення логічної структури роботи чат-боту.

Опис першого етапу формування даних:

- Вхідні дані: технічне завдання для розробки чат - боту.

- Вихідні дані: отримання зв'язку з Telegram.

- Управління: документація Telegram, документація сервісів Google.

- Механізми: розробник, технічне забезпечення, python.
- Опис другого етапу формування даних:
- Вхідні дані: отримання зв'язку з Telegram.
- Вихідні дані: отримання зв'язку з Google.
- Управління: документація сервісів Google.
- Механізми: розробник, python, speech to text recognition.

Опис третього етапу формування даних:

- Вхідні дані: отримання зв'язку з Google, API key для доступу до сервісів Google.
- Вихідні дані: чат -бот.
- Управління: документація Telegram.
- Механізми: розробник, python, API Google.

Діаграма другого рівня представлена на (рис.3.3).

Після декомпозиції контекстної діаграми на три процеси було виконано розбиття кожного з них на певний перелік робіт, результатом кожної з яких є вхідними даними для наступної.

Процес підключення до сервісів містить в собі такі етапи: підключення до Google Assistant, підключення до Google Search, отримання готової відповіді на запит. Це зумовлено тим, що перед тим, як підключитись до Google, необхідно зробити ряд заходів за допомогою мови програмування.

Для процесу підключення до сервісів Google Assistant були визначені такі дані:

- Вхідні дані: отримання зв'язку з Telegram.
- Вихідні дані: результат пошуку інформації.
- Управління: розробник, python, API Google.
- Механізми: документація сервісів Google.

Оскільки в даній системі використовується Google Assistant не масштабної версії, то необхідно додати пошукову систему Google Search для того, щоб бот працював безвідмовно. Були визначені такі дані:

- Вхідні дані: результат пошуку інформації.
- Вихідні дані: масштабований пошук.
- Управління: розробник, python, speech-to-text-recognition.
- Механізми: документація сервісів Google.

Останнім процесом є отримання відповіді на запит. Інакше кажучи, готова інформаційна система, яка включає в себе такі дані:

- Вхідні дані: масштабований пошук.
- Вихідні дані: отримання зв'язку Google.
- Управління: API Google, speech-to-text-recognition.
- Механізми: документація сервісів Google.

Діаграма декомпозиції процесу підключення до сервісів Google зображена на (рис.3.4).

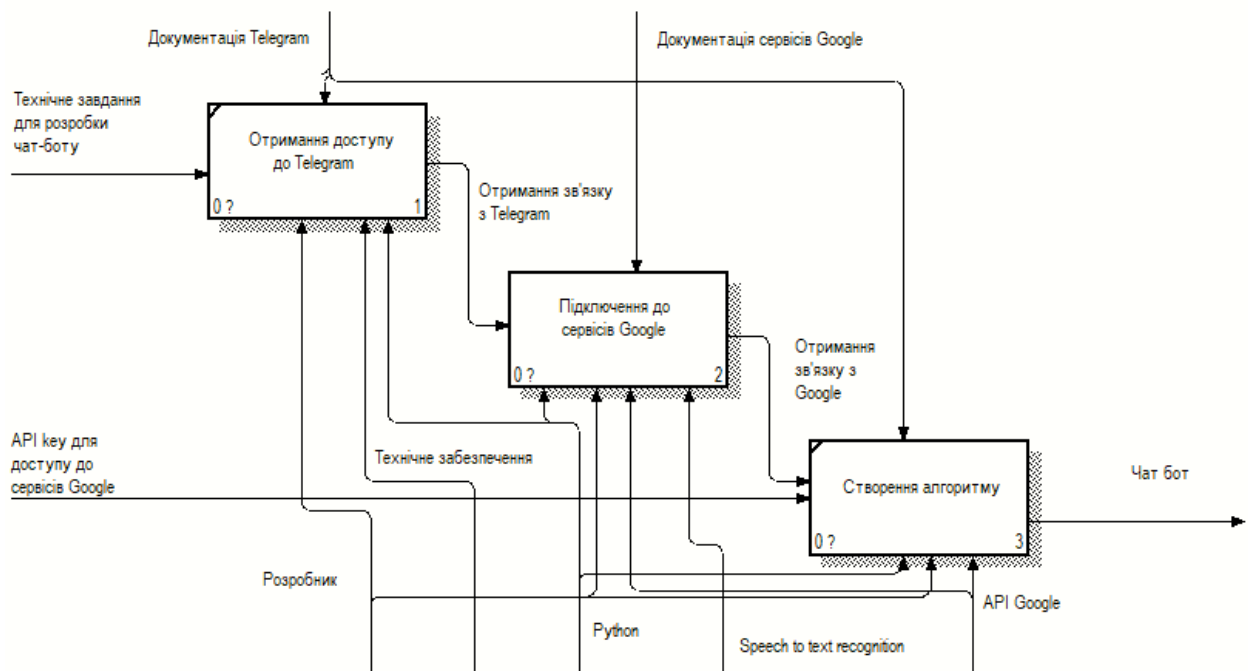


Рисунок 3.3 – Діаграма другого рівня

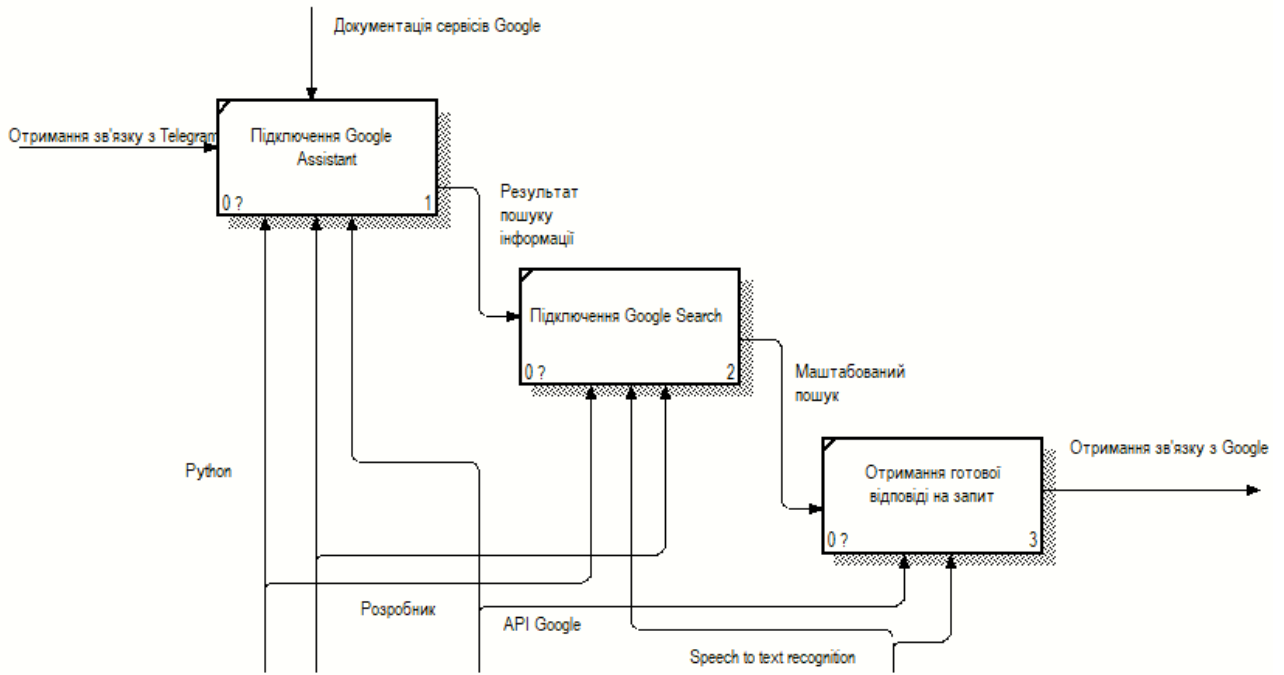


Рисунок 3.4 – Діаграма декомпозиції другого етапу

3.4 Моделювання варіантів використання чат-боту

Конструкції UML створюються з багатьох модельних елементів, які позначають різні частини системи програмного забезпечення. Елементи UML використовуються для побудови діаграм, які відповідають певній частині системи або точці зору на систему.

Для розробки діаграми Use Case були визначені такі актори:

- Admin-розробник, який повністю може маніпулювати чат-ботом, забезпечуючи технічну модернізацію системи.
- User-користувач, який може користуватись лише тими функціями бота, які в нього вбудовані.

Після того, як були визначені всі актори, які будуть взаємодіяти з системою, необхідно сформулювати перелік варіантів використання. Вони відповідають вимогам, яким повинен відповідати чат-бот, що зазначені у Додатку А.

Варіанти використання для чат-боту:

- Telegram;
- текстовий пошук;
- голосовий пошук;
- додавання надбудов;
- зміна правил;
- зміна конфігурацій;
- Dialogflow.

Виходячи з сформованих даних про акторів та всіх можливих варіантів використання чат-боту, розроблена Use Case діаграма. Вона представлена на рисунку 3.5.

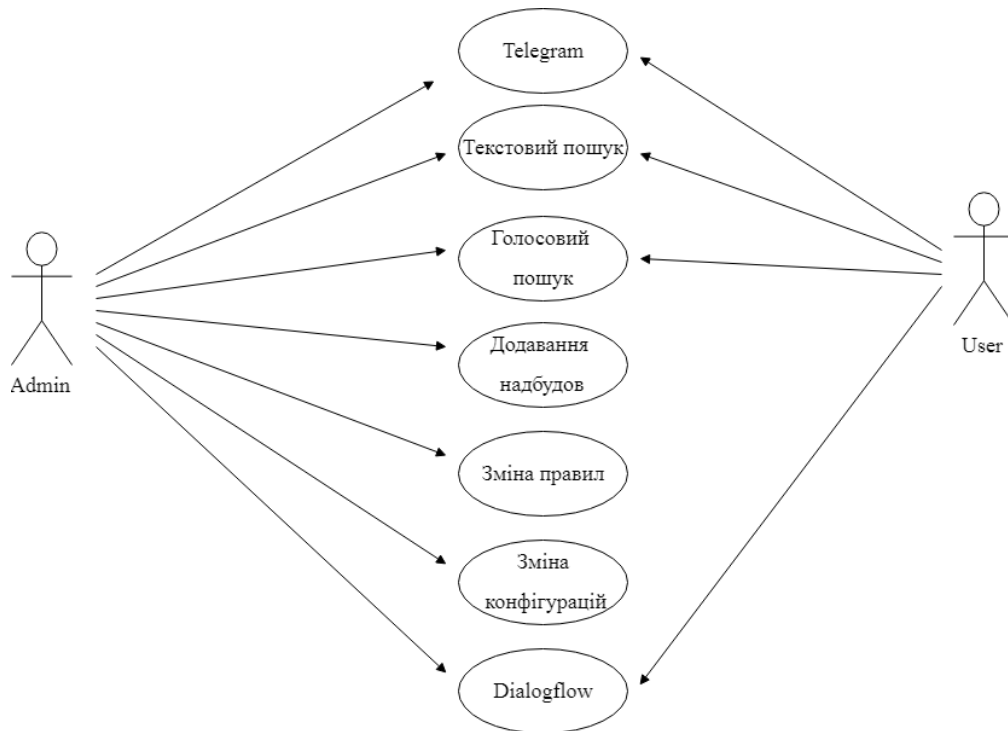


Рисунок 3.5 – Діаграма варіантів використання

4 РОЗРОБКА ІНФОРМАЦІЙНОГО ЧАТ-БОТУ

4.1 Розробка логічної структури чат-боту

Описаний у розділі 2 функціонал робить його унікальним та досить простим у використанні. Щоб зрозуміти, яким чином буде функціонувати даний чат-бот, була розроблена схема, яка зображена на рисунку 4.1.

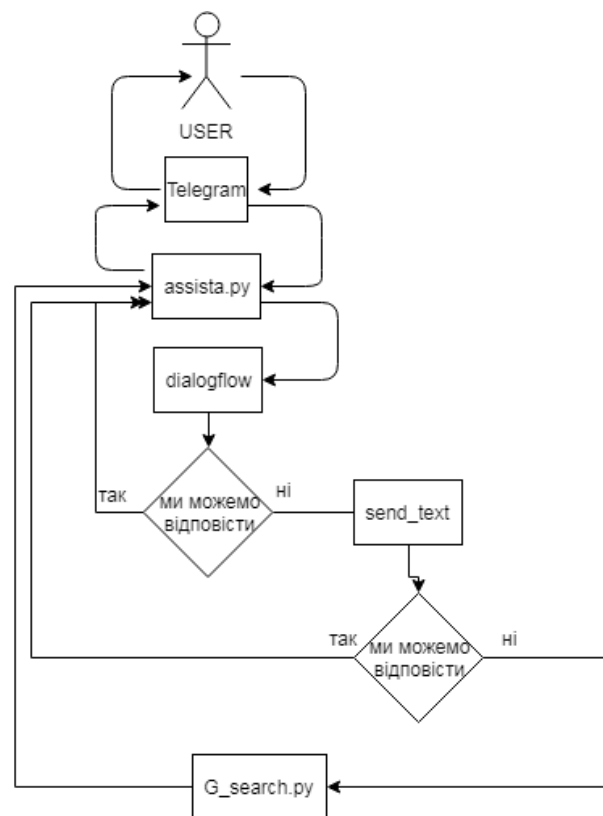


Рисунок 4.1 – Схема роботи чат – бота

Виходячи з цього, ми бачимо його унікальність. Дослідження полягає в тому, щоб дослідити, яким чином будуть працювати API як одне ціле. Адже на сьогоднішній час є актуальною робота з API, вони набагато полегшують проектування будь-якого проекту для будь-яких задач.

Першим кроком необхідно було створити бота в системі телеграм та отримати доступ до месенджеру. Отримання доступу здійснювалось стандартним методом. Отримання API токenu зображено на рисунку 4.2.

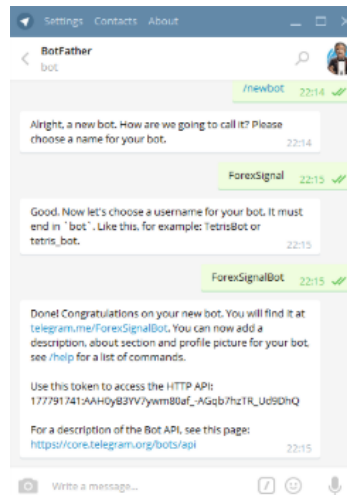


Рисунок 4.2 – Отримання доступу до Telegram боту

Наступний крок-створення скрипту, який об'єднає продукти компанії Googleз месенджером Telegram. На (рис.4.3) представлена схема папок проекту на мові програмування Python.

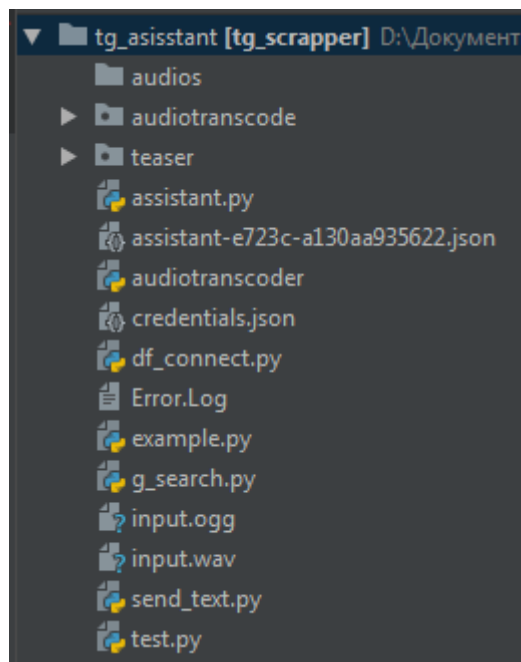


Рисунок 4.3 – Структура проекту

Папка `audios` використовується для збереження усіх аудіо повідомлень від користувача. Папка `audiotranscode` використовується для імпорту бібліотеки, яка змінює формат аудіофайла. Фрагмент програмного коду для збереження аудіо повідомлень:

```
at=audiotranscode.Audiotranscode()
at.transcode('{} .ogg'.format(name), '{} .wav'.format(name))
```

На даний момент папка `teaser` не використовується, але може бути підключена для створення короткого опису великого тексту.

Файл `assistant.py` виконує роль моста між різними продуктами Google та месенджером і є його основною частиною, яка приймає та відправляє повідомлення користувачів та продуктів Google. Це було досягнуто за допомогою бібліотеки `telebot`. Фрагмент програмного коду для ініціалізації боту:

```
bot = telebot.TeleBot('632423476907230:AAEoEya4zob2ZSWgYSr7Hk6JG6nKVUvPLk4')
```

Бот має функціонал для привітання при старті розмови. Фрагмент коду для привітання користувача:

```
@bot .mes_handler (commands3= ['start'], cont_types=['text'])
def send_welcome (mes) :
bot.send_mes(mes.chat.id, "Hello, I'm Scrapper, your personal assistant.")
```

Основна частина боту, яка відповідає за обробку повідомлень-запитів користувача та відправлення відповіді бота. Фрагмент коду для обробки усіх повідомлень користувачів:

```
@bot.mes_handler(cont_types=['voice', 'text'])
def get_voice(mes):
bot.send_mes(mes.chat.id, 'Ok, let me check what you wrote.')
```

Bot.polling- це команда, яка запускає сервер боту та дозволяє зберігати повідомлення про помилки у файл. Фрагмент коду для старту бота

```
def telegram_pollng():
    try:
        bot.pollng(none_stop=True, timeout=60) #
    except:
        traceback_error_string=traceback.format_exc()
        with open("Error.Log", "a") as myfile:
            myfile.write("\r\n\r\n" + time.strftime("%c")+ "\r\n<<ERROR pollng>>\r\n"+
            traceback_error_string + "\r\n<<ERROR pollng>>")
        bot.stop_pollng()
        time.sleep(10)
        telegram_pollng()
```

Файли з розширенням json - це службові файли, які використовуються для авторизації та взаємодії з продуктами Google.

Файл df_connect.py використовується для взаємозв'язку між скриптом та Dialogflow. Фрагмент коду для взаємодії бота та Dialogflow.

```
import dialogflow_v2 as dialogflow
sesion_client = dialogflow.SesionClient()
sesion = sesion_client.sesion_path('assistant-e723c', 'assistant-e723c')
text_input = dialogflow.types.TextInput(
    text=text, language_code=lang_code)
quer_input = dialogflow.types.QuerInput(text=text_input)
respon = sesion_client.detect_intent(
    sesion=sesion, quer_input=quer_input)
print(respon)
if respon.quer_rezult.intent.display_name == 'Default Fallback Intent':
    return None
```

Error.log файл зберігає звіти про помилки, щоб мати можливість відстежити, де, в який час та за яких обставин виникла та чи інша помилка. Фрагмент коду для звіту про помилки.

```
Sat Jun 2 01:15:16 2018 «ERROR pollng»
```

Traceback (most recent call last):

```
File "/usr/local/lib/python3.5/dist-packages/urllib3/connectionpool.py", line 387, in _make_request six.raise_from(e, None)
```

```
File "<string>", line 2, in raise_from
```

```
File "/usr/local/lib/python3.5/dist-packages/urllib3/connectionpool.py", line 383, in _make_request httplib_response = conn.getresponse()
```

```
File "/usr/lib/python3.5/http/client.py", line 1197, in getresponse response.begin()
```

File "/usr/lib/python3.5/http/client.py", line 297, in begin version, status, reason = self._read_status()

File "/usr/lib/python3.5/http/client.py", line 258, in _read_status line = str(self.fp.readline(_MAXLINE + 1), "iso-8859-1")

File "/usr/lib/python3.5/socket.py", line 575, in readinto return self._sock.recv_into(b)

File "/usr/local/lib/python3.5/dist-packages/urllib3/contrib/pyopenssl.py", line 294, in recv_into raise timeout("The read operation timed out") socket.timeout: The read operation timed out

test.py та example.py файл використовується для проведення тестів та вирішення проблем пов'язаних з конфліктами між модулями.

Файл g_search.py дозволяє відправляти текстові повідомлення в Google Search API та повертає результат пошуку. Фрагмент коду файлу g_search.py:

```
def search_result(query):
    print('GOT QUERY: ', query)
    search_results = google.search(query, num_page)
    return_results = []
    for result in search_results:
        return_results.append('{}\n{}\nURL: {}'.format(result.name, result.description, result.link))
        print(result.name, result.link, result.description)
    result = {'response': return_results}
```

Файл send_text.py використовується для відправлення запитів в Google Assistant API та відображення результату. Спочатку вказуються налаштування асистенту. Фрагмент коду для налаштування Google Assistant:

```
def iter_assist_zaproq():
    conf = embedded_assist_pb2.AssistConf(
        audio_out_conf=embedded_assist_pb2.AudioOutConf(
            encoding='LINEAR16',
            sample_rate_hertz=16000,
            volume_percentage=0,
        ),
        dialog_state_in=embedded_assist_pb2.DialogStateIn(
            leng_coding=ego.leng_coding,
            conv_state=ego.convn_state,
            is_new_conv=ego.is_new_conv,
        ),
        dece_conf=embedded_assist_pb2.DeceConf(
            dece_id=ego.dece_id,
            dece_model_id=ego.dece_model_id,
        ),
        txt_mas=txt_mas,
    )
```

Наступний крок - надіслати запит до Google Assistant API та повернути результат. Фрагмент коду для запиту в Google Assistant API:

```
grpc_channel = google.auth.transp.grpc.secure_authorized_channel(
    credentials, http_zaproq, api_endpoint)
logging.info('Connecting to %s', api_endpoint)
with SampleTxtAssist(lang, dece_model_id, dece_id, ekr,
    grpc_channel, grpc_deadline) as assist:
    while True:
        mas = mas
        vidp_txt, vidp_html = assist.assist(txt_mas=mas)
    return vidp_txt
```

4.2 Опис функціонування чат-боту

Помічник створений для постійного двостороннього діалогу між людиною і Google, який розуміє наш світ і допомагає виконувати необхідні завдання. Він дозволяє полегшити покупку квитків в кіно, коли людина знаходиться в русі, або пошук ідеального ресторану для родини, щоб влаштувати швидкий перекус перед початком показу фільму, а потім допомогти дістатися до кінотеатру. Таким чином був підключений даний сервіс, але він є обмеженим. Оскільки API не розширене і немає синхронізації з іншими сервісами, то Google Assistant видає результат, зображений на рис. 4.14 - 4.15.

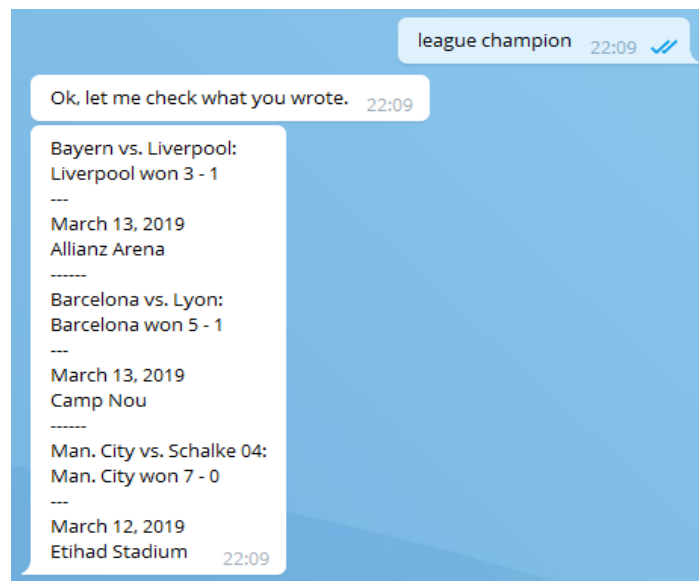


Рисунок 4.14 – Результат від Google Assistant

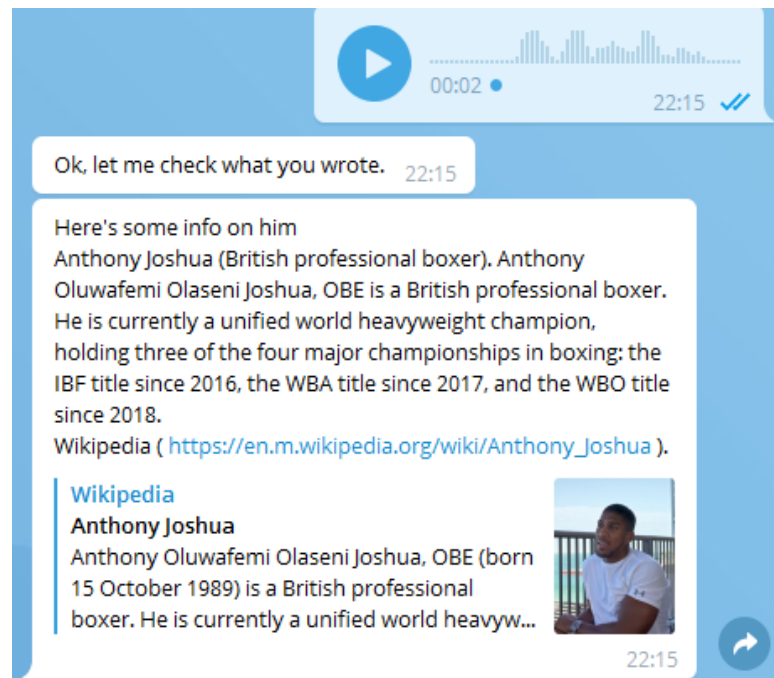


Рисунок 4.15 – Результат від Google Assistant

Google Assistant створена за допомогою нейронних мереж , завдяки котрим ми бачимо саме таку відповідь на наші запити. Асистент спростовує відповідь і робить її формалізованою та чіткою.

Google Search - це пошукова система, розроблена компанією Google. Найбільша пошукова система в світі, котра є на першому місці по результатам точності відповідей на запити. Google отримує інформацію з різних джерел. Ось приклади деяких з них:

- веб-сторінки;
- контент, доданий користувачами, наприклад, опублікований ними в сервісах "Google Мій бізнес" і "Google Карти";
- відскановані книги;
- відкриті бази даних в Інтернеті;
- безліч інших джерел.

Таким чином в даній роботі дана система відіграє другорядну роль. Тобто, якщо асистент неспроможний дати відповідь, одразу відповідь буде надаватись у формі 10 перших посилань сторінки. Це зроблено для того, щоб

чат-бот був максимально приближений у наданні відповіді. Приклад результату Google Search зображений на (рис 4.16).

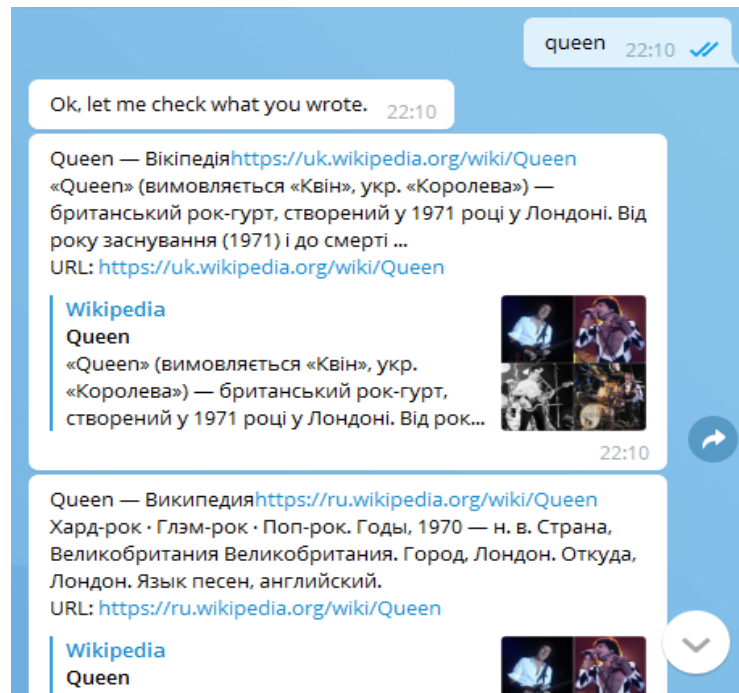


Рисунок 4.16 – Результат від Google Search

Для того щоб бот був більш живим, його необхідно навчити певному спілкуванню з користувачем. Цю задачу виконує Google Assistant, за таким принципом зроблена сама API. Але API працює тільки на англійській мові, спілкування відбувається на англійській мові, і запити виконуються на англійській мові. Приклад спілкування зображений на рисунку 4.17.

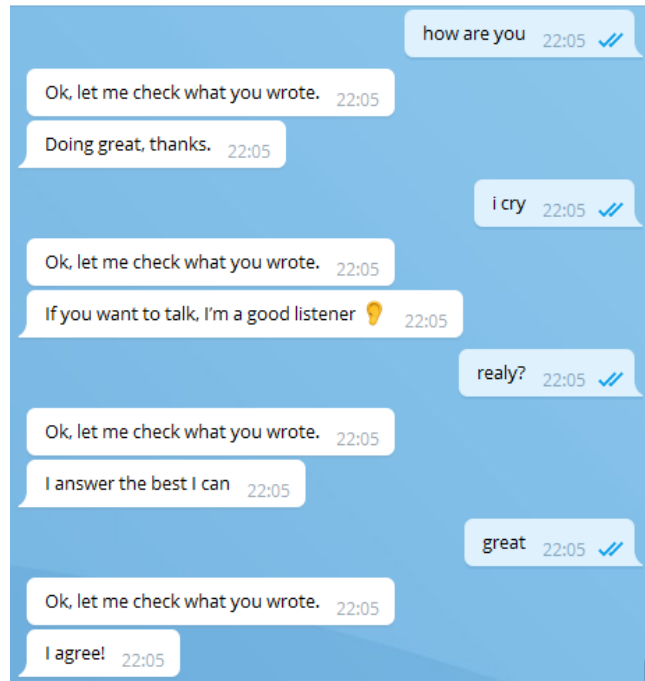


Рисунок 4.17 – Спілкування з чат - ботом

Для того щоб бот проводив спілкування зрозумілою мовою, був підключений сервіс компанії Google Dialogflow.

Dialogflow – це онлайн сервіс від Google, який дозволяє створювати своїх чат-ботів або функції для Googleасистента (це називається agent). На Хабре вже є кілька непоганих статей, в яких розглядаються основні інструменти цього продукту. Ось одна з них. По суті, Dialogflow вмie зіставити фразу користувача на природній мові (і російською теж) з деяким «немовним» значенням (званим intent) і що-небудь відповісти.

До того ж він ще й тримає контекст діалогу, щоб на наступну фразу можна було зреагувати в контексті попередніх реплік. Інструмент досить простий в управлінні і, крім того, зберігає всю статистику запитів, щоб можна було в один клік збільшити навчання агента новими intent або поповнювати існуючі. Власне так і відбувається навчання: людина просто показує агенту приклади запитів, які відносяться до того чи іншого intent, а він автоматично будує модель і все краще, і краще розуміє користувача. Також він може виділити з фрази якісь сутності, наприклад, дати або міста, якщо це необхідно для виконання якоїсь функції. Із всіх можливих функцій

була використана функція Small Talk. Вона використовується для надання відповідей на випадкові розмови.

Ця функція може значно покращити досвід користувачів, висвітлюючи поширені питання, які можуть не стосуватися намірів агента. В першу чергу, щоб під'єднатись до сервісу, необхідно перейти за посиланням (рис 4.18).

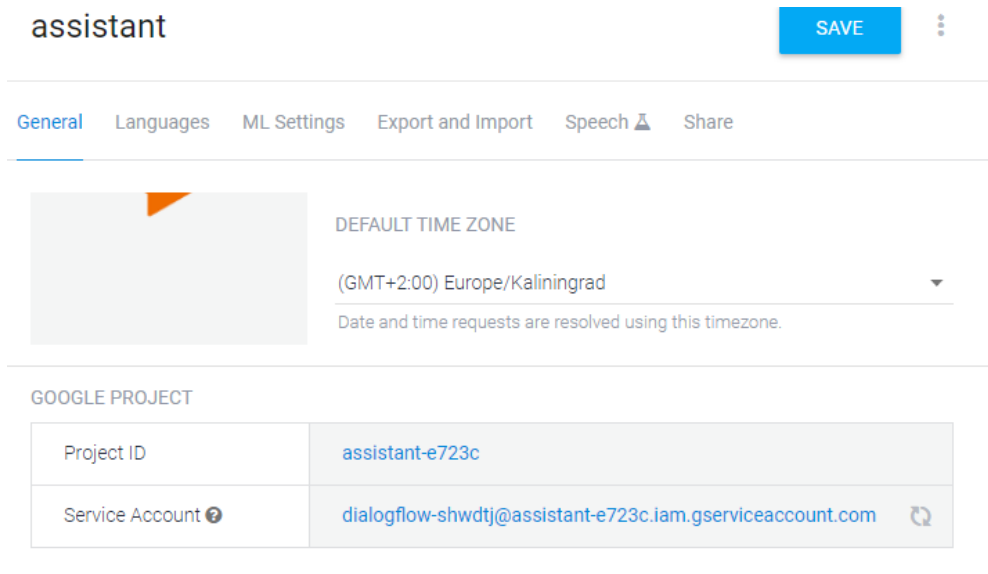


Рисунок 4.18 – Сервісний акаунт

Наступним кроком йде створення сервісного акаунта для того, щоб отримати ключ доступу в форматі json для взаємодії з Dialogflow ботом через python скрипт (рис 4.19). Також можна побачити сервісні акаунти, які взаємодіють з ботом (рис 4.20). Далі необхідно створити власний сервісний акаунт (рис 4.21).

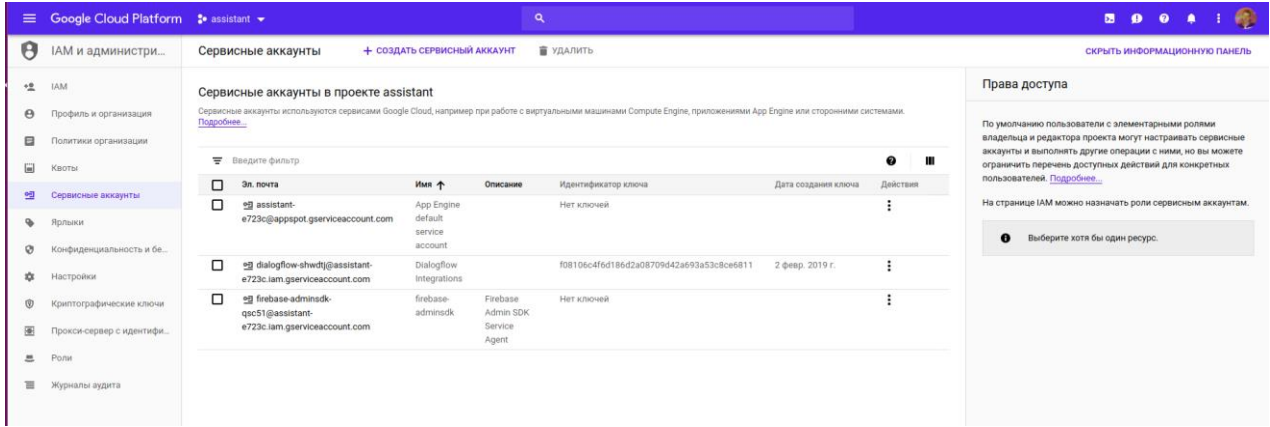


Рисунок 4.19 – Створення сервісного акаунта

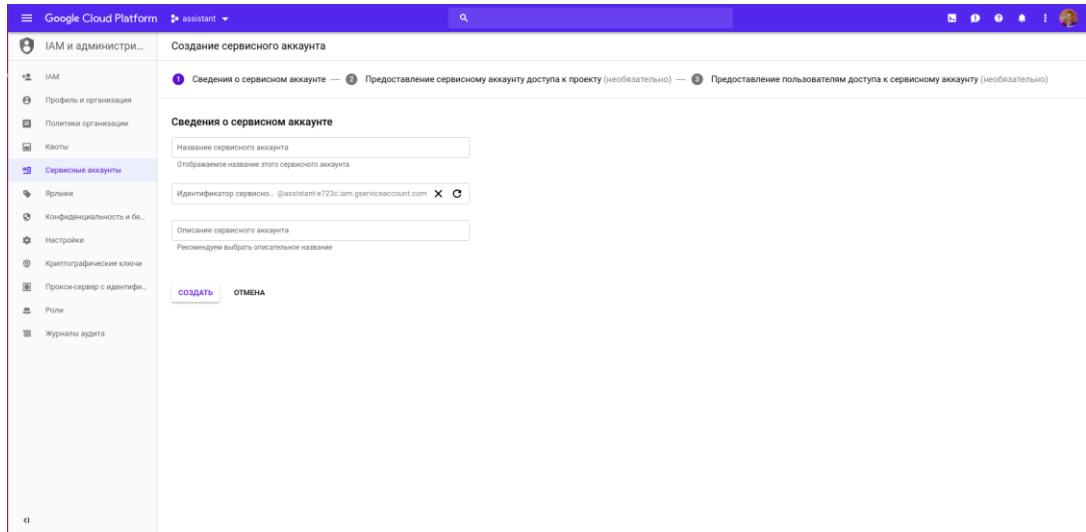


Рисунок 4.20 – Існуючі сервісні акаунти

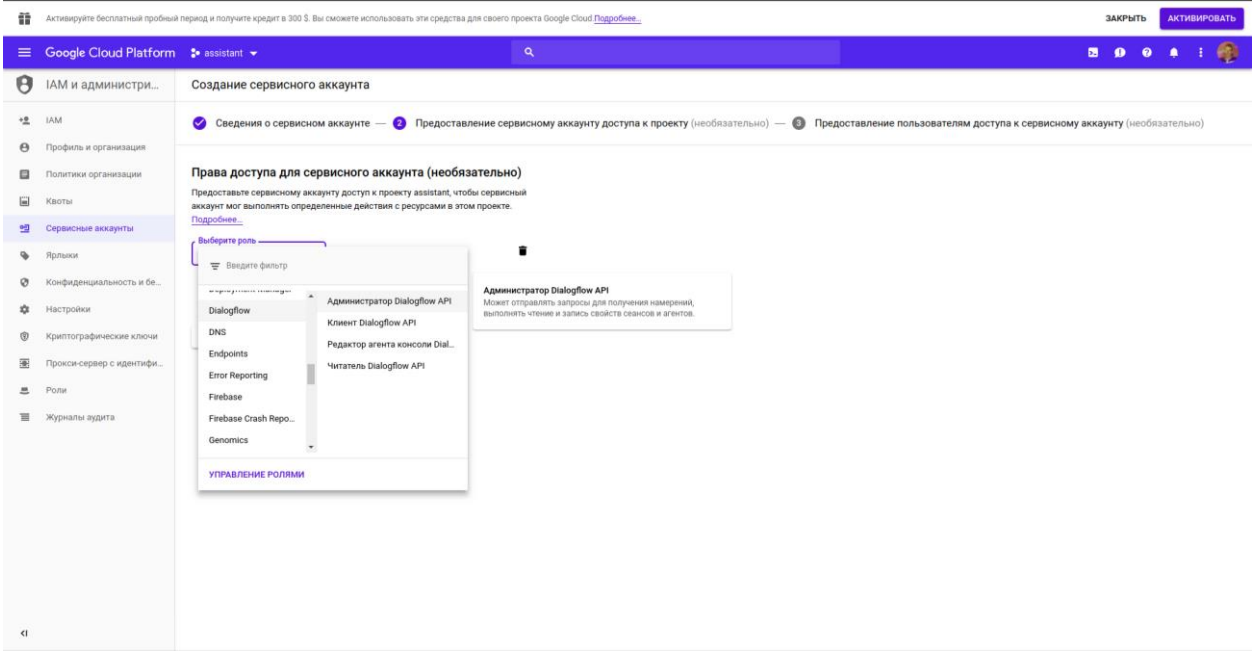


Рисунок 4.20 – Особистий сервісний акаунт

Далі налагоджуємо особистий рівень доступу сервісного акаунту (рівень адміністратора). Створюємо ключ у форматі json файла (рис. 4.21).

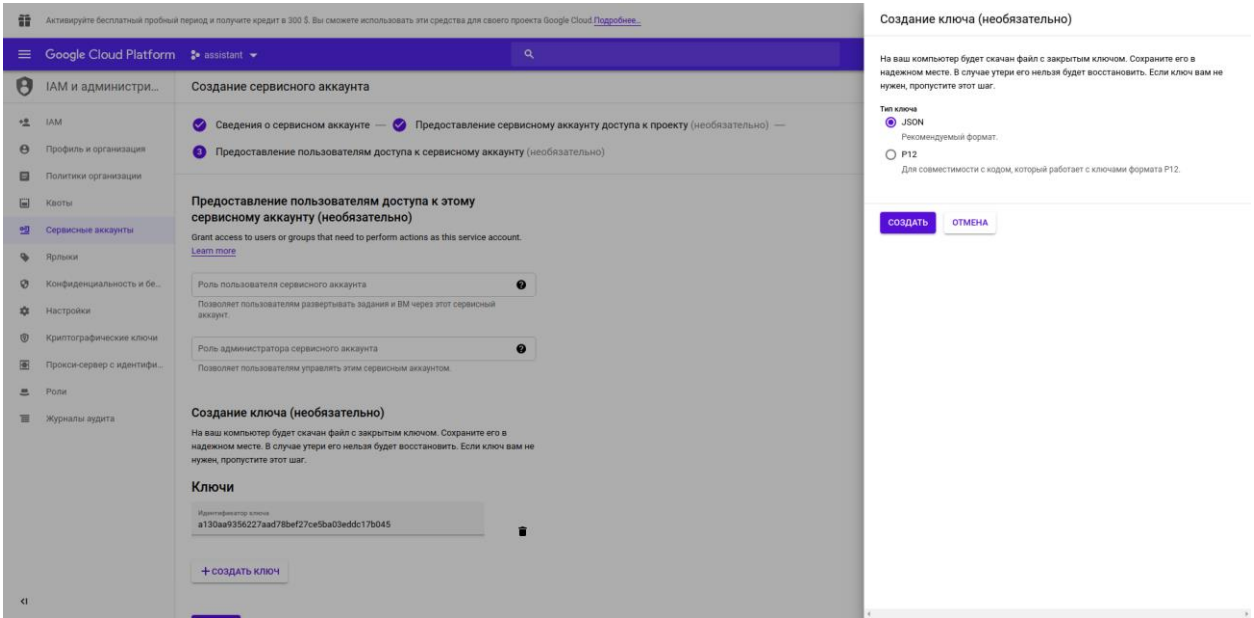


Рисунок 4.21 – Рівень доступу

З метою широкого спілкування можна обрати певні категорії, в яких система може вільно вести бесіду.

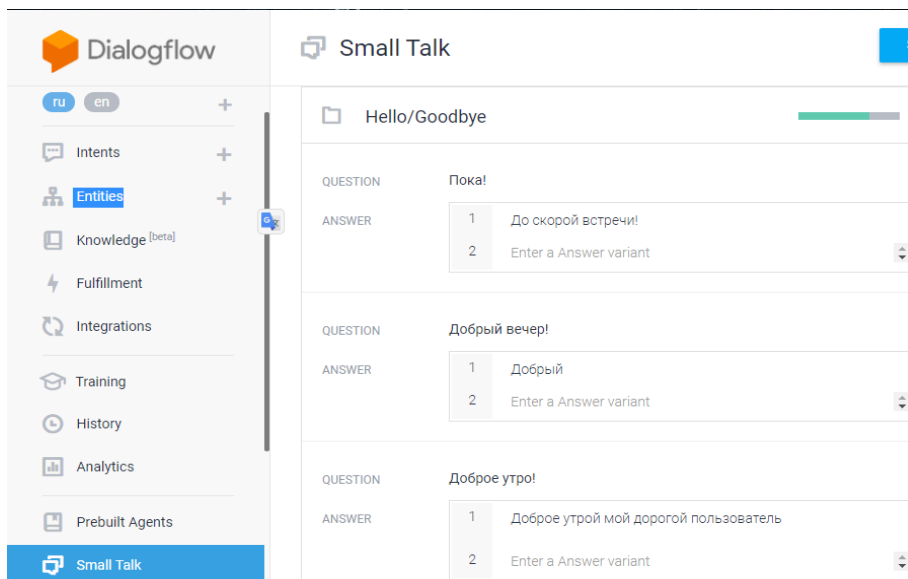


Рисунок 4.22 – Налаштування категорій Dialogflow

Результат роботи Dialogflow в месенджері Telegram зображений на рис.4.23-4.24.

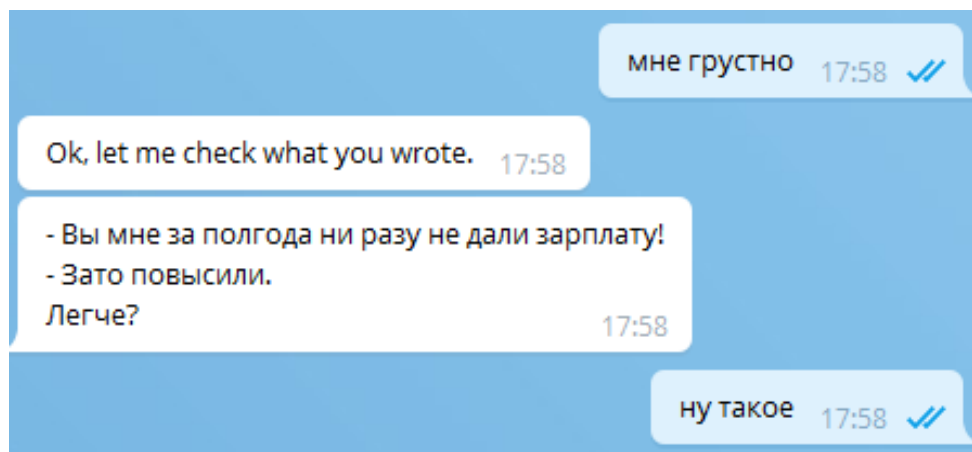


Рисунок 4.23 – Результат роботи Dialogflow

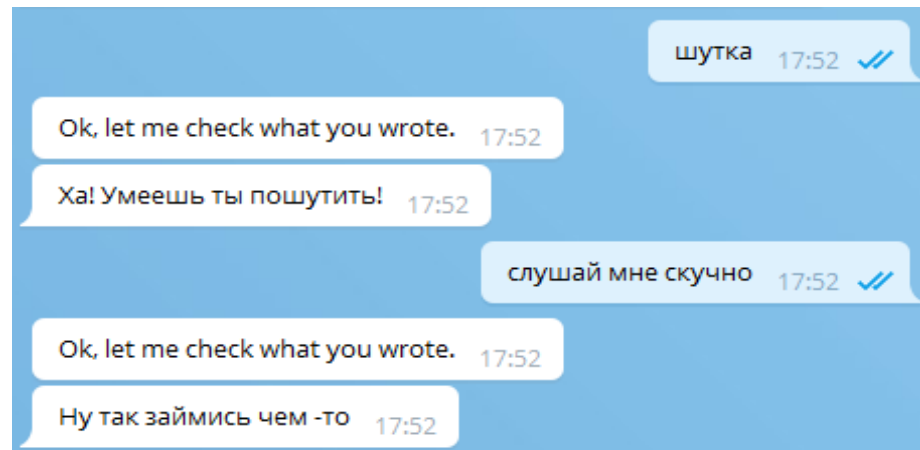


Рисунок 4.24 – Результат работы Dialogflow

Для того щоб пошук здійснювати можна було не лише текстом, а і голосом, була підключена API Speech-to-Text recognition, котра дозволяє синтезувати мову в текст, що робить чат-бота зручнішим та мобільним. Насправді подібних синтезів мови доволі багато, але був здійснений їх аналіз та було прийняте рішення обрати Speech-to-Text recognition.

Результат підключення одразу можна побачити у консольному вікні середовища програмування, яке зображене на рис.4.26. В месенджері даний синтез мови працює таким чином, що необхідно відправити аудіо повідомлення. Приклад зображений на рис. 4.27.

```
GOT QUERY: кто такая леди гага?
Lady Gaga – Вікіпедія https://uk.wikipedia.org/wiki/Lady\_Gaga
Леди Гага – Вікіпедія https://ru.wikipedia.org/wiki/Леди\_Гага
Леди Гага (Lady Gaga) фото, биография, последние новости ... https://uznayvse.ru/znamenitosti/bi
Леди Гага – биография, фото, личная жизнь, новости, песни 2019 ... https://24smi.org/celebrity/6
Леди Гага | Биография Lady Gaga | Фото | Личная жизнь | Журнал ... https://www.cosmo.ru/stars/bi
Леди Гага - все новости, личная жизнь и биография звезды https://viva.ua/stars-bio/ledi-gaga
30 фактов о Леди Гаге - Фактрум https://www.factroom.ru/facts/5160
Леди Гага (Lady Gaga) (Актриса, Музыкант): фото, биография ... https://www.vokrug.tv/person/sho
FROM GOOGLE SEARCH
meest ukrainian
```

Рисунок 4.26 – Результат роботи Speech-to-Text recognition

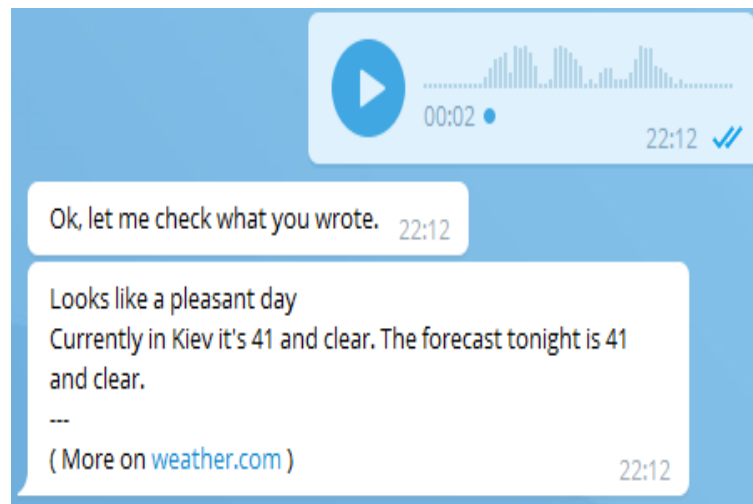


Рисунок 4.27 – Результат роботи Speech-to-Text recognition в Telegram

ВИСНОВКИ

Цифровізація економіки і нова тенденція орієнтації систем підтримки прийняття рішень на використання мобільних пристроїв ставлять задачу розробки нового покоління чат - ботів.

Виходячи з аналізу предметної області, а саме дослідження актуальності та необхідності чат-ботів у сучасному житті та їх використання в сучасних месенджерах, було вирішено розробити бота, який зможе відповісти на будь-який інформаційний запит.

Аналіз аналогів дав можливість зрозуміти, які типи та які категорії сучасних чат – ботів можна знайти у мережі. Завдяки даному аналізу було вирішено питання актуальності та працездатності боту. Дана розробка буде безкоштовна і буде у відкритому доступі для всіх верств населення, у яких є смартфон, підключення до мережі Інтернет, а також месенджер Телеграм.

Функціональні можливості чат-бота:

- володіти спілкуванням з користувачем;
- пошук інформації за допомогою тексту;
- пошук інформації за допомогою голосового повідомлення.

При реалізації продукту створено WBS-графічне представлення усіх робіт проекту, OBS-схема виконавців цих робіт та на їх основі матрицю відповідальності особи до елементарної роботи, побудовано діаграму Ганта та PDM-мережу, які показують терміни виконання усіх робіт та їх порядок виконання. Визначення можливих ризиків допоможе уникнути їх, а прорахований ступінь впливу на реалізацію проекту дозволить визначити план дій з усунення фактору їх прояву.

В зв'язку з проблемами ефективного використання Telegram для ряду девайсів були виконані та описані принципи побудови ефективної пошукової системи в соціальній мережі Telegram. Реалізовані алгоритми

взаємодії користувача з всесвітньою мережею. Чат-бот, реалізований за допомогою скриптової мови Python, а також сервісів Google. Дослідна експлуатація чат боту довела конструктивність підходу, намічено шляхи його вдосконалення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Чат-бот - что это такое, как работает, виды, преимущества и недостатки, как создать [Электронный ресурс] – Режим доступа: <http://kak-bog.ru/chat-bot-chto-eto-takoe-kak-rabotaet-vidy-preimushchestva>
2. Чат-боты: обзор и состояние технологий в отрасли m [Электронный ресурс] – Режим доступа: https://webguru.pro/blog/chat_bots_development/
3. Чат-боты: обзор и состояние технологий в отрасли [Электронный ресурс] – Режим доступа: <http://nlpx.net/archives/425>
4. Чат-бот для бизнеса [Электронный ресурс] – Режим доступа: <https://www.integrity.com.ua/activities/chatbots-for-business/>
5. Exploring chatbots for news [Электронный ресурс] – Режим доступа: <https://medium.com/nos-digital/exploring-chatbots-for-news-7f12d226c53d>
6. AI and Chatbots in Education [Электронный ресурс] – Режим доступа: <https://chatbotsmagazine.com/ai-and-chatbots-in-education-what-does-the-futurehold-9772f5c13960>
7. Advantages of Chatbot Integration into Entertainment Industry [Электронный ресурс] – Режим доступа: <https://chatbotslife.com/advantages-of-chatbot-integration-into-entertainment-industry-f62bfed3c003>
8. Чат-бот : зачем создавать, кому использовать [Электронный ресурс] – Режим доступа: <https://blog.click.ru/chat-bot-na-sajte-zachem-sozdavat-komu-ispolzovat-i-kak-nastroit/>
9. 50 лучших чат-ботов для Telegram [Электронный ресурс] – Режим доступа: <https://uip.me/2016/04/50-popular-telegram-bots/>
10. python-telegram-bot [Электронный ресурс] – Режим доступа: <https://github.com/python-telegram-bot/python-telegram-bot>

11. Лутц М. Программирование на Python, том II, 4-е издание. – Пер. сангл. – СПб.: Символ-Плюс, 2011. – 992 с.
12. How Assistant work & Introduction to Dialogflow [Электронный ресурс] – Режим доступа: <https://codeburst.io/2-how-assistant-work-introduction-to-dialogflow-319a72ba2db>
13. Introduction to the Google Assistant Library[Электронный ресурс] – Режим доступа:<https://developers.google.com/assistant/sdk/guides/library/python>
14. Messaging, simplified [Электронный ресурс] - Режим доступа: https://www.nexmo.com/products/sms/build?utm_source=google_search&utm_medium=paid&utm_campaign=EMEA_EURO_EN_SMS_General_B&utm_term
15. SpeechRecognition 3.8.1[Электронный ресурс] – Режим доступа: <https://pypi.org/project/SpeechRecognition/>
16. Enhance Your Presence on Search [Электронный ресурс] - Режим доступа: <https://developers.google.com/search/>
17. Telegram APIs [Электронный ресурс]-Режим доступа: <https://core.telegram.org/>
18. pyTelegramBotAPI [Электронный ресурс]-Режим доступа: <https://github.com/eternnoir/pyTelegramBotAPI>
19. Пишем telegram-бота на python с помощью библиотеки telebot [Электронный ресурс] - Режим доступа: <https://habr.com/ru/post/448310/>
20. Cloud Speech-to-Text documentation [Электронный ресурс] - Режим доступа: <https://cloud.google.com/speech-to-text/docs/>
21. Как создать Telegram бота с помощью Python [Электронный ресурс] - Режим доступа: <https://medium.com/nuances-of-programming/>
22. Роботы: информация для разработчиков [Электронный ресурс] - Режим доступа: <https://tlgrm.ru/docs/bots>
23. Инструкция: Как создавать ботов в Telegram [Электронный ресурс] - Режим доступа: <https://habr.com/ru/post/262247/>

24. Telegram Bot API [Электронный ресурс] - Режим доступа: <https://core.telegram.org/bots/api>
25. Створюємо Telegram бота на Python. Частина 1 [Электронный ресурс] - Режим доступа: <https://codeguida.com/post/410>
26. Welcome to Python Telegram Bot's documentation [Электронный ресурс] - Режим доступа: <https://python-telegram-bot.readthedocs.io/en/stable>
27. Learn to build your first bot in Telegram with Python n [Электронный ресурс] - Режим доступа: <https://medium.freecodecamp.org/learn-to-build-your-first-bot-in-telegram-with-python-4c99526765e4>
28. Пишем Telegram бота на Python хостинг на Heroku [Электронный ресурс] - Режим доступа: https://teletype.in/@cozy_codespace/Hk70-Nt14
29. Performing Google Search using Python code [Электронный ресурс] - Режим доступа: <https://www.geeksforgeeks.org/performing-google-search-using-python-code/>
30. How to use Google API in Python [Электронный ресурс] - Режим доступа: <https://boostlog.io/@bily809/how-to-use-google-api-in-python>
31. Python Library for Google Search [Электронный ресурс] - Режим доступа: <https://catonmat.net/python-library-for-google-search>

Додаток А. Технічне завдання

ТЕХНІЧНЕ ЗАВДАННЯ на розробку інформаційної системи «Інформаційний чат-бот для соціальної мережі Telegram з використанням Google Assistant та Google Search API»

Додаток А

Технічне завдання

Назва чат- боту «assistant».

Область застосування чат-боту- чат-бот,призначений для пошуку інформації та спілкування у режимі реального часу в інформаційній системі Telegram. Призначений для побутового використання.

Об'єкт, у якому використовують програму-користувачі з месенджером Telegram.

1 Основи для розробки

Розробка виконується на фоні того, що чат-боти являються на сьогоднішній час популярною автоматизованою системою.

2 Призначення розробки

Розроблений бот повинен надавати можливість користувачам котрі не мають голосових асистентів на смартфонах, та автоматизувати пошук, та розробити систему в системі, що є зручним та мобільним.

Назва організації: Кіншаков Едуард, IT-51, кафедра Комп'ютерних Наук.

Тема проекту: «Інформаційний чат-бот для соціальної мережі Telegram з використанням Google Assistant та Google SearchAPI»

3 Вимоги до веб-ресурсу

Чат -бот повинен мати звичайний вигляд без різних надбудов. Головна панель з клавіатурою, та кнопка для голосового повідомлення. Функціональні можливості боту представлені у пункті 2.1.

3.1 Вимоги до чат - боту

Оскільки будуть використовуватись API сервіси Google скрипковою мовою Python, то логічна структура звичайно буде прописана мовою програмування Python.

3.2 Вимоги до програмного забезпечення

Головною вимогою є наявність на смартфоні або десктопна версія на ПК месенджері Telegram, версія неважлива. Обов'язкове підключення до мережі інтернет.

3.3 Вимоги до функціональних характеристик

Чат-бот повинен забезпечувати наступні функції:

- текстовий пошук;
- голосовий пошук;
- текстовий діалог;
- голосовий діалог.

4 Перелік програмної документації

- опис проекту продукту із використанням UML-діаграм;
- алгоритм роботи чат - бота;
- програмний код розробки бота;
- технічне завдання;

5 Порядок виконання робіт і етапи розробки

Стадії та етапи розробки повинні відповідати ГОСТ 19.101-77 і складатися з наступних пунктів:

- оформлення завдання для дипломної роботи;
- планування роботи. Розроблення ТЗ ,побудова мережевого графіку та діаграми Ганта;
- розроблення структури чат - боту;
- розроблення функціоналу ;
- робота в режимі реального часу вести тестування чат-боту;
- інструкція користувача;
- оформлення пояснювальної записки про виконання дипломної роботи;
- задача пояснювальної записки до дипломної роботи та розробленого чат-боту;

- презентація роботи та її захист.

5 Порядок контролю та приймання

Контроль коректності функціонування та придатності чат-боту здійснюється замовником (секцією Інформаційні технології проектування) на основі наданої пояснювальної записки до дипломної роботи та програмних файлів. Контроль ходу виконання проекту здійснюється на основі календарного плану виконання дипломної роботи:

- перевірка завдання дипломної роботи;
- перевірка ТЗ, мережевого графіка та діаграми Ганта;
- перевірка структури чат-боту;
- перевірка чат-боту в месенджері;
- перевірка наявності функціоналу в режимі реального часу;
- інструкція користувача;
- задача ПЗ;
- презентація.

Додаток Б

Планування робіт

Чат - бот насамперед представляє собою не лише, як систему котра буде вести спілкування з людиною, насправді це скрипт котрий автоматизує той чи інший інформаційний процес. Кожного дня людина користується своїми девайсами, та проводить певний час у різних додатках, соціальних мережах, чат-ботах.

У 2015 році поняття бот набуло великої популярності. Дуже багато бізнес компаній, консалтингових компаній мали необхідність придбати собі бота, котрий виконував саме ті задачі, які було необхідно. Завдяки месенджері Telegram, поняття бот викликало велику прихильність у звичайних користувачів смартфонів. Завдяки цим ботам , можна проглянути абсолютно все, що буде цікавити людину.

Головна задача та ціль проекту, розробити інформаційну систему в інформаційній системі, котра буде робити запит на пошук інформації та повертати її як звичайне повідомлення вже зі знайденою інформацією.

Загальний вигляд мети полягає в тому, щоб людина без зайвих рухів могла знаходитись в одній системі, і щоб вона була, як розважальна, наукова, інформаційна.

Деталізація методом Smart є ефектом знання мови Python та повної логічної структури системи пошуку та електронного помічника, а також володіння технічною документацією Google, буде створена реалізація в форматі чат – боту. А саме пошуковий бот.

Найбільш важливими результатами є:

- підключення Speech-to-Text recognition, налаштування його, щоб йшло розпізнавання голосу у текст;
- підключення API Google Assistant та Google Search;

- паралельна робота асистента та пошукової системи;
- повернення результату.

Планування змісту робіт. Структурна декомпозиція робіт (Work Break down Structure) - називають представлення проекту, виконане у вигляді ієрархічної структури робіт, що досягається за допомогою послідовної декомпозиції. Інструмент спрямований на детальне планування, оцінку вартості, визначення та розподіл персональної відповідальності виконавців. Тобто, на основні робіт і результатів, що визначають зміст проекту. WBS структура рисунок Б.1.

Говорячи про організаційну структуру, ми маємо на увазі і концептуальну схему, навколо якої організовується група людей, основу, на якій тримаються всі функції. Організаційна структура підприємства - це, по суті, керівництво для користування, яке пояснює, як організація збудована і як вона працює. Якщо говорити конкретніше, то організаційна структура описує, як в компанії приймаються рішення і хто є її лідером. OBS структура рисунок Б.2.

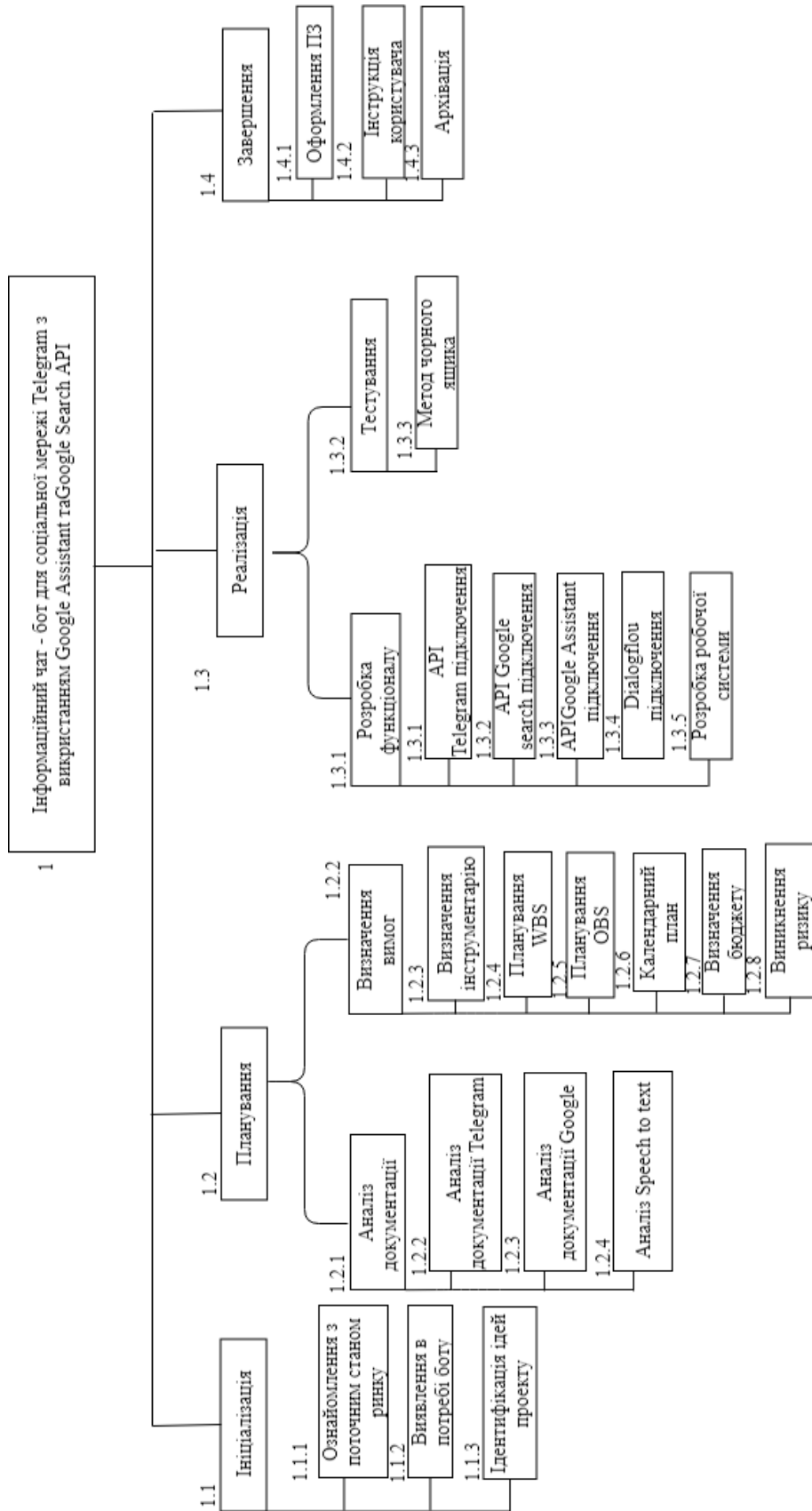


Рисунок Б.1 – WBS структура системи

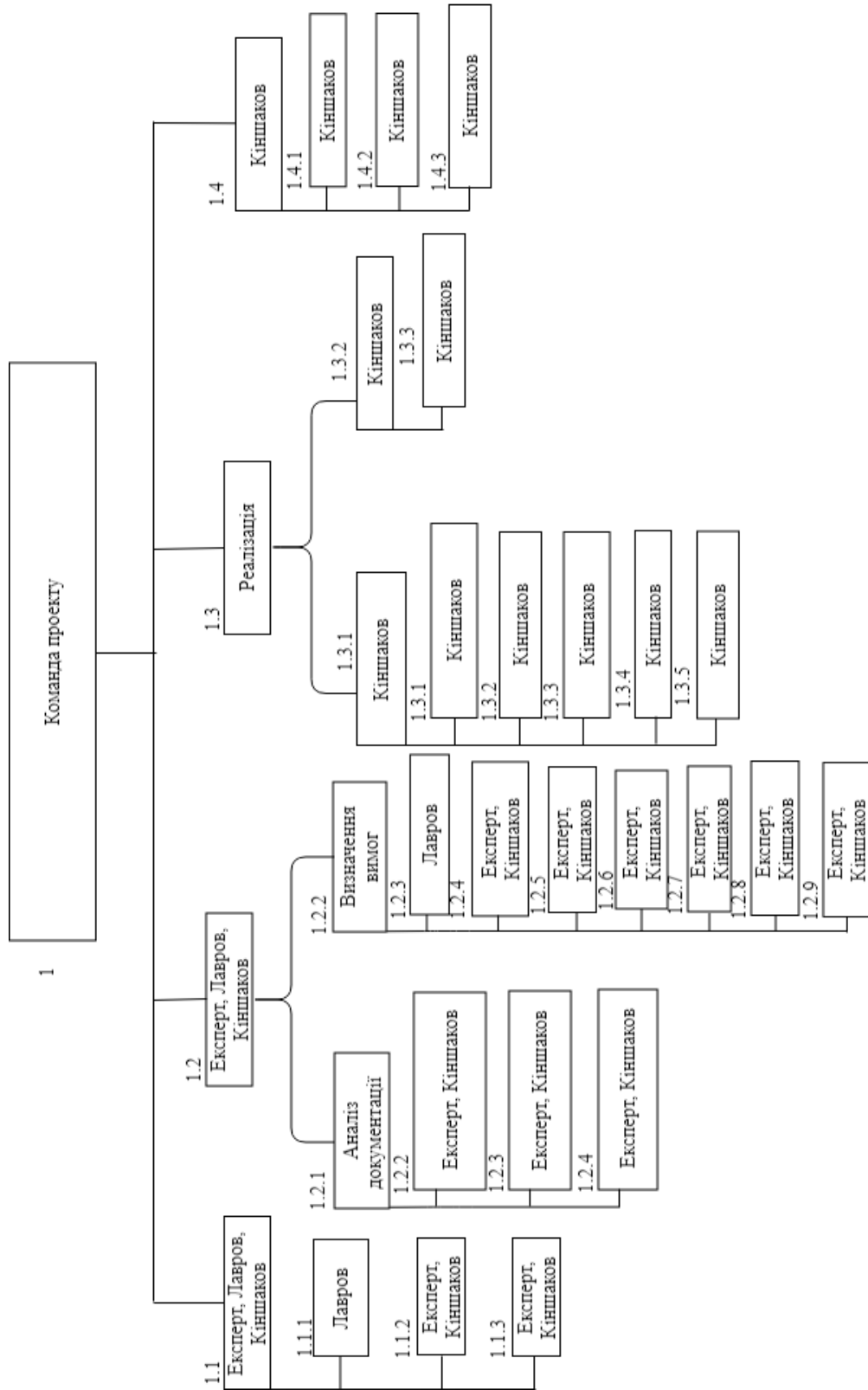


Рисунок Б.2 – OBS структура системи

Матриця відповідальності. Матриця RACI є простий інструмент, який використовується для визначення значень і обов'язків та уникнення плутанини при виконанні завдань або процесів. Використовується при управлінні проектами і для показу обов'язків в станах "AS-IS" і "TO-BE".

Матриця відповідальності являє собою особливий метод визначення функціональних областей, ключових напрямків діяльності, критеріїв прийняття управлінських рішень, де існують неясності. Всі розбіжності, що виникають в ході данного процесу, можуть бути винесені на загальне обговорення та згодом вирішені шляхом прийняття колективного рішення.

Беручи за основу OBS та WBS структури була побудована матриця відповідальності проекту, яка реалізована у вигляді таблиці.

Таблиця Б.1 – Матриця відповідальності

WBS\OBS	Кіншаков	Лавров	Експерт
Ознайомлення з поточним станом ринку			
Виявлення в потребі боту			
Ідентифікація ідей проекту			
Аналіз документації Telegram			
Аналіз документації Google			
Аналіз Speech to text			
Вимоги			
Визначення інструментарію			
Планування WBS, OBS			
Календарний план			
Визначення бюджету			
Виникнення ризику			
Розробка функціоналу			
API Telegram підключення			
API Google підключення			
API Google Assistant підключення			
Dialogflow підключення			
Розробка робочої системи			
Тестування			
Метод чорного ящика			
Оформлення ПЗ			
Інструкція користувача			

PDM мережа. PDM мережа була побудована за допомогою надбудови програми GanttProject. Ця надбудова має назву Pert діаграма. PERT призначений для дуже масштабних, одноразових, складних, нерутинних проектів. Метод має на увазі наявність невизначеності, даючи можливість розробити робочий графік проекту без точного знання деталей і необхідного часу для всіх його складових. PERT був розроблений головним чином для спрощення планування на папері і складання графіків великих і складних проектів. Метод особливо націлений на аналіз часу, який потрібен для

виконання кожної окремої задачі, а також визначення мінімального необхідного часу для виконання всього проекту. Мережу зображено (рис. Б.3-Б.4).

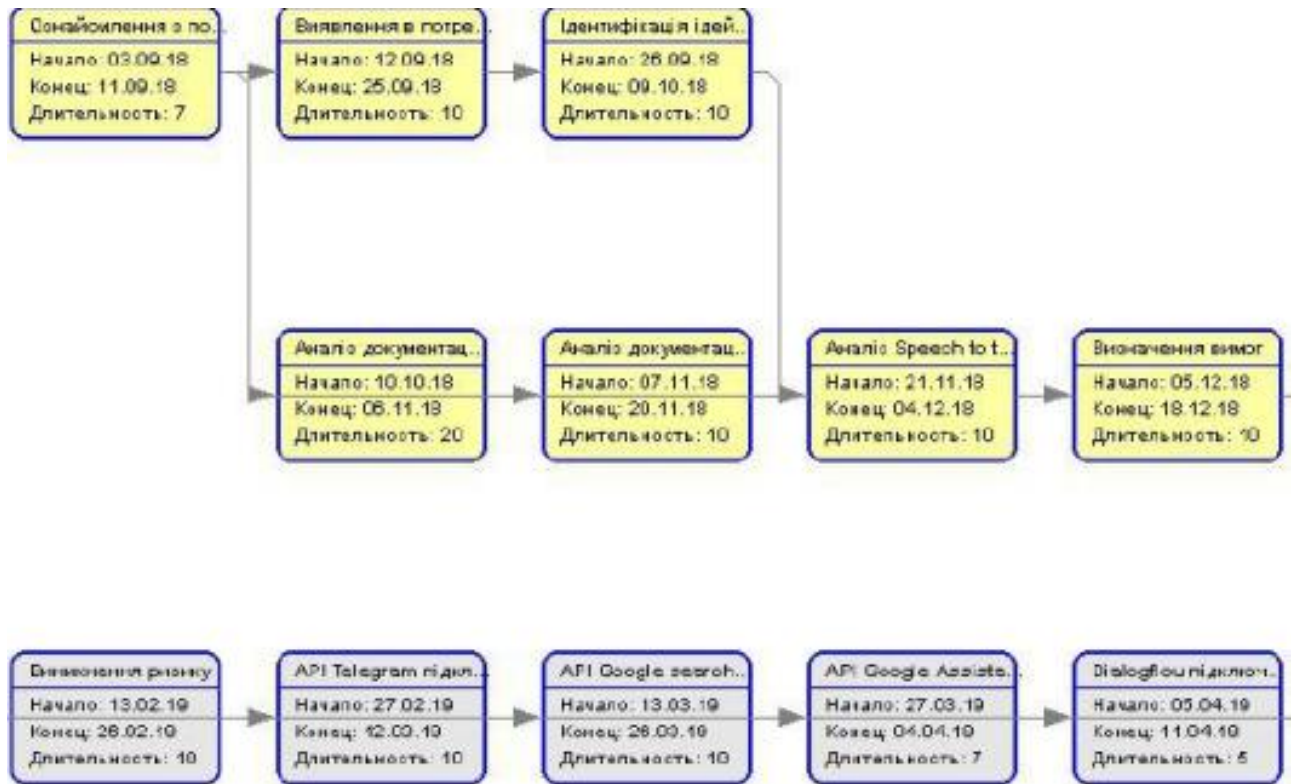


Рисунок Б.3 – PDM-мережа

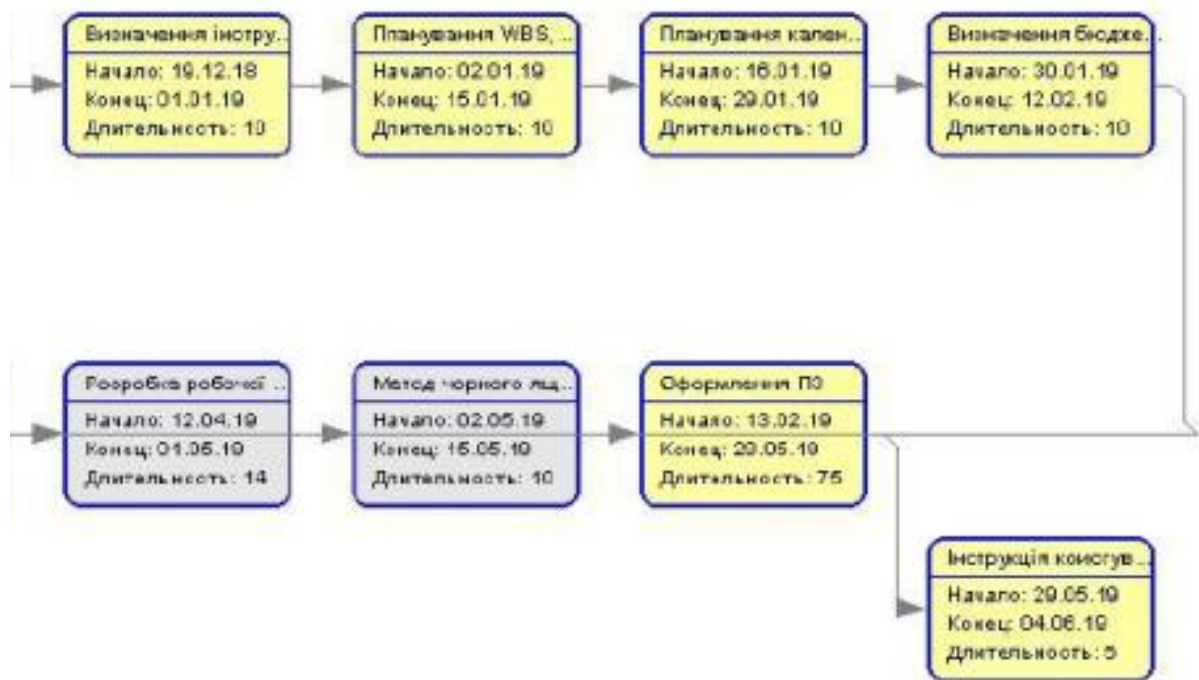


Рисунок Б.4 – Продовження PDM-мережі

Діаграма Ганта. Діаграма Ганта - це візуальний спосіб відображення запланованих завдань. Горизонтальні графіки широко використовуються для планування проектів будь-яких розмірів в різних галузях і сферах. Це зручний спосіб показати, яка робота планується до виконання в певний день і час. Дана діаграма також допомагає командам і менеджерам проектів контролювати дати початку і закінчення будь-якого проекту. Все в одному просторі. Діаграму Ганта наведено на рис. Б.5.

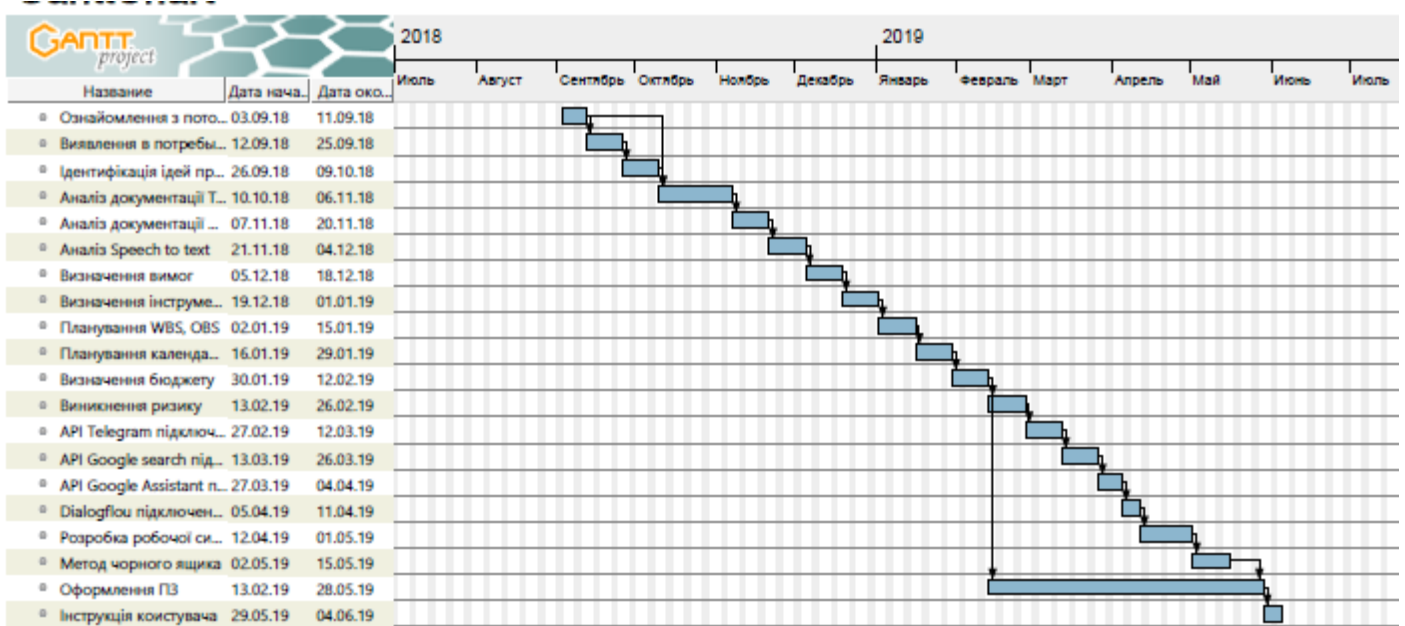


Рисунок Б.5 – Календарний графік робіт

Управління ризиками. Коли були визначенні всі роботи даного проекту, та спеціалістів котрі будуть розробляти даний продукт, необхідно передбачити всі можливі ризики котрі можуть вплинути на якість та час розробки даного проекту.

Були виділені такі ризики як:

- R1 – зміна ТЗ на етапі розробки;
- R2 – пропущені помилки у ході розробки;
- R3 – недотримання календарного плану;
- R4 – зміна функціоналу сервісів google;
- R5 – хвороба розробника;
- R6 – некоректне тестування;
- R7 – некоректна робота апаратного забезпечення.

Наступним кроком за допомогою оцінки експертів була визначена ймовірність появи все можливих ризиків, згідно з цим було побудовано таблицю.

Таблиця Б.2 – Ймовірність виникнення ризиків

Ймовірність виникнення	R1	R2	R3	R4	R5	R6	R7
Слабо ймовірний							
малоймовірний							
ймовірний							
дуже ймовірний							
майже можливий							

Далі було побудовано таблицю можливих втрат при виникненні ризиків.

Потім було зроблено таблицю тих втрат котрі можуть виникнути в процесі проектування та розробки.

Таблиця Б.3 – Втрати при виникненні ризиків

Значимість впливу	R1	R2	R3	R4	R5	R6	R7
мінімальна							
низька							
середня							
висока							
максимальна							

Виходячи з цих двох, була побудована Матриця впливу (ймовірностей і наслідків) ризиків. В матриці світлим кольором позначено неважливі ризики-білий колір, темнішим – помірні, темним – критичні.

Таблиця Б.4 – Матриця ймовірність-втрати

Ймовірність		R2	
		R4	R6
	R3		
		R1	R5
	R7		
Вплив			

Виходячи з цього, було визначено три критичних ризики, такі як:

- R2 – пропущені помилки у ході розробки;
- R4 – зміна функціоналу сервісів google;
- R6 – некоректне тестування;

Що до першого ризику, його можна уникнути лише за допомогою програміста, а саме зміна коду. Другий ризик взагалі він малоімовірний, але якщо він все ж таки виникне, то знову ж таки необхідне втручання розробника, щоб певний сервіс замінити на інший. Третій ризик, це не коректне тестування, що може повести за собою певні проблеми з замовником.

Формування бюджету. Завершенням планування проекту є етап розподіл бюджету даного проекту. Перед тим, як визначити бюджет, необхідно виділити тих виконавців котрі брали безпосередню участь в даній роботі. Були визначені такі працівники як:

- Розробник;
- Керівник;
- Експерт.

Додаток В

Приклади роботи Чат-боту

Приклад 1

Діалог з ботом через Google Assistant

Оскільки у відкритому доступі API Google Assistant є на англійській мові, то спілкування буде проводитись виключно англійською мовою. Починати діалог, можна з відомої ключової фрази «OK Google», результат зображений на рис.В.1.

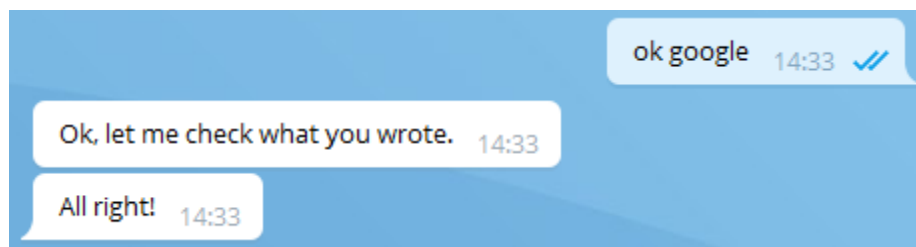


Рисунок В.1 – Початок діалогу

Як можна побачити на рис. В.2. бот відповідає на питання та вітається, але також здатний відповідати на звичайні фрази (рис.В.3).

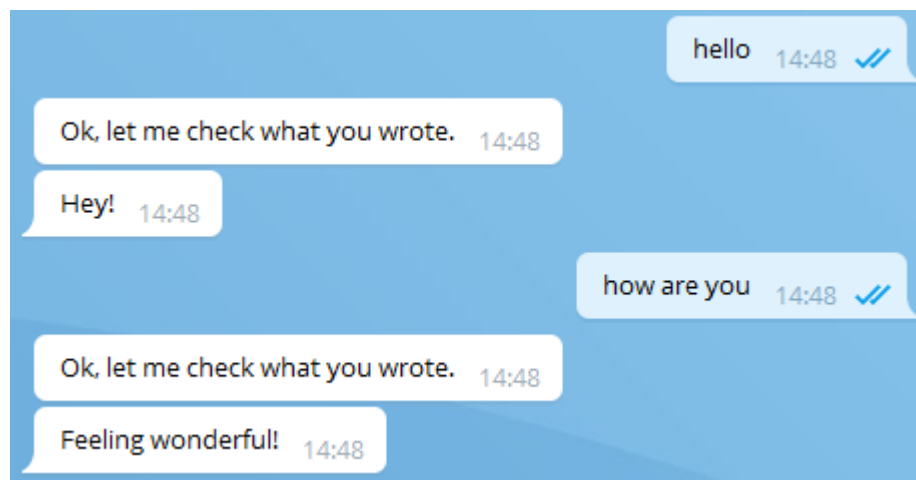


Рисунок В.2 – Відповіді на питання

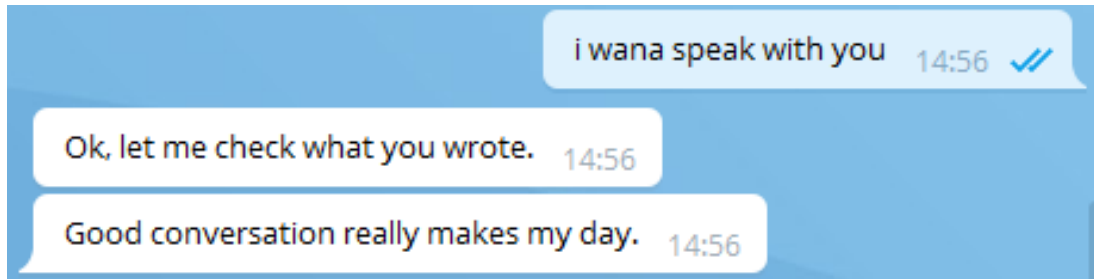


Рисунок В.3 – Підтримка контексту діалогу

Google Assistant представляє собою гарно навчену та логічно структуровану нейронну мережу. Вона генерує відповідь не лише однією фразою на одне і теж питання (рис. В.4- В.5).

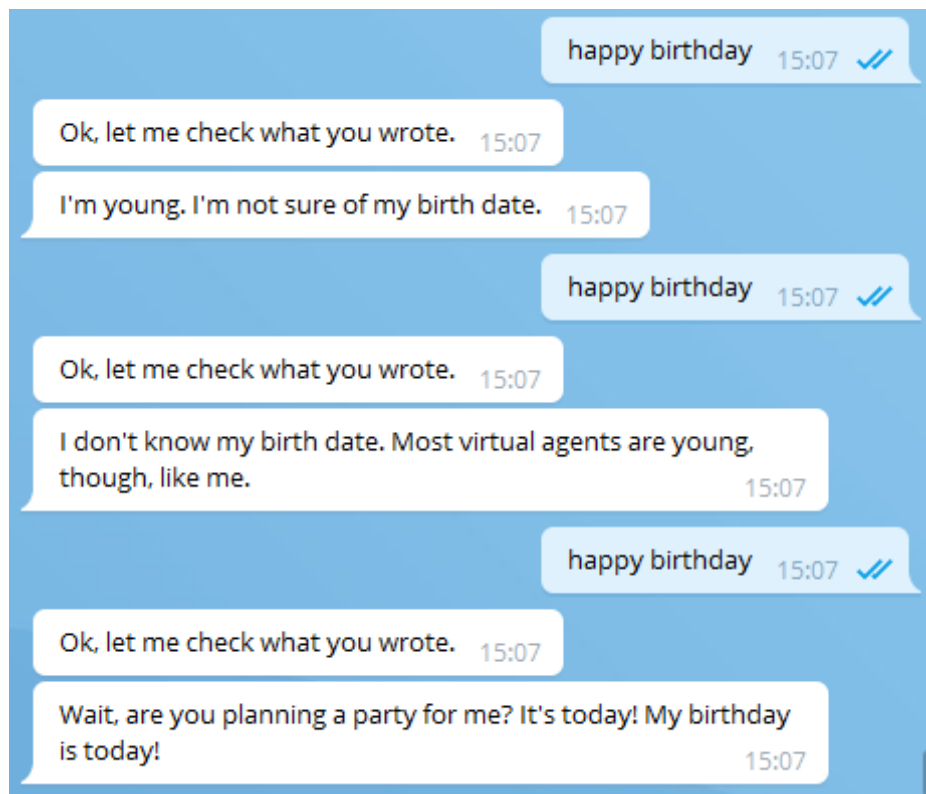


Рисунок В.3 – Приклад варіантів відповіді чат-боту

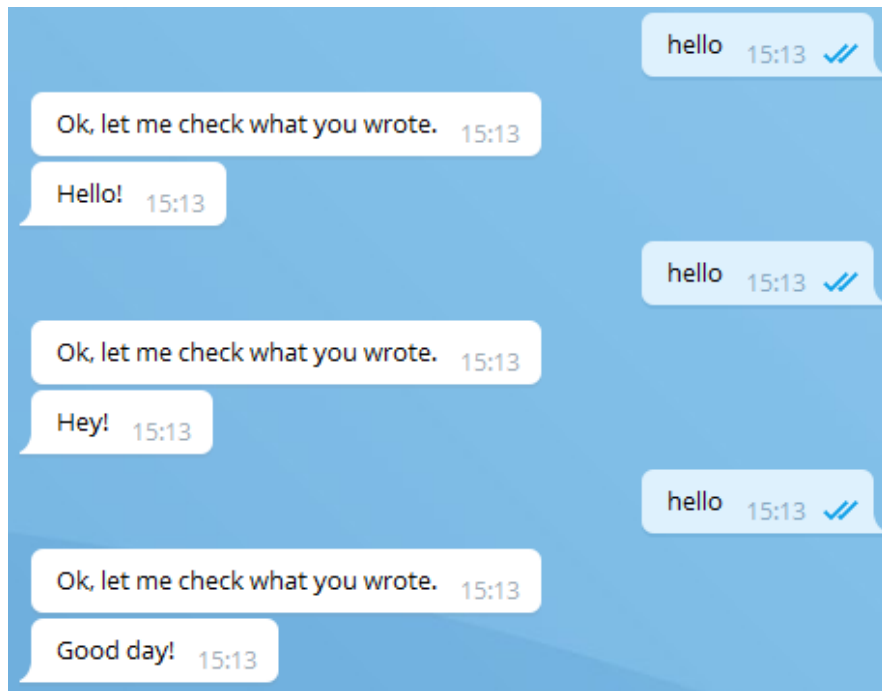


Рисунок В.4 – Приклад варіантів відповідей на питання

На рис. В.5, можна побачити що асистент, веде діалог зі зворотнім питанням. Тобто максимально наближене до людської розмови.

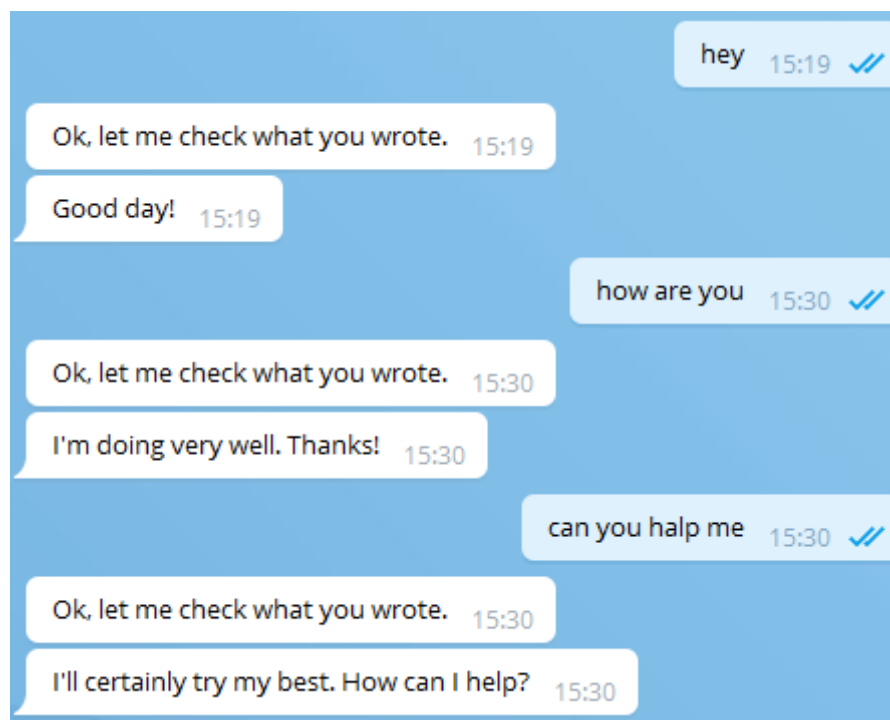


Рисунок В.5 – Зворотні питання від боту

Якщо асистент, не може відповісти на питання або не розуміє зміст питання, то він про це повідомляє наступним чином, результат повідомлення зображений на рис В.6.

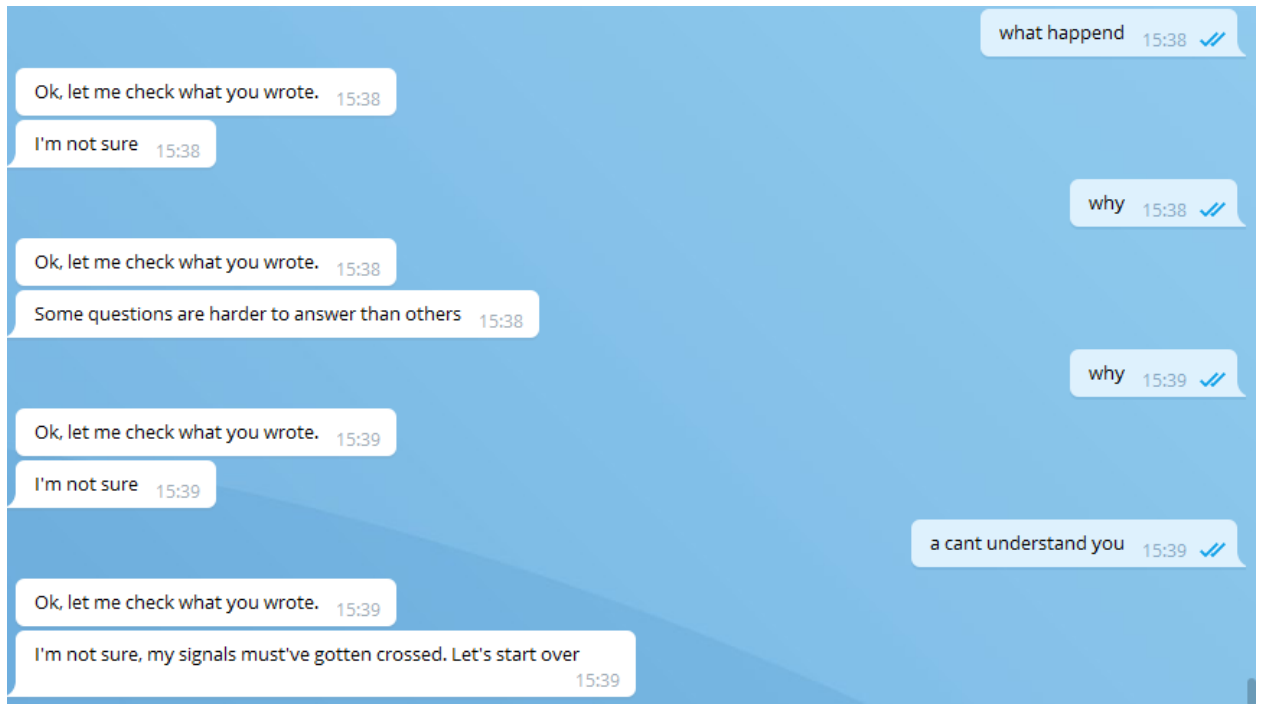


Рисунок В.6 – Приклад роботи з проблемними ситуаціями

Діалог с ботом при підтримці Google Assistant, може мати певний контекст, або вільне спілкування, у форматі чату, результати спілкування з ботом зображені на рис. В.7-В.12.

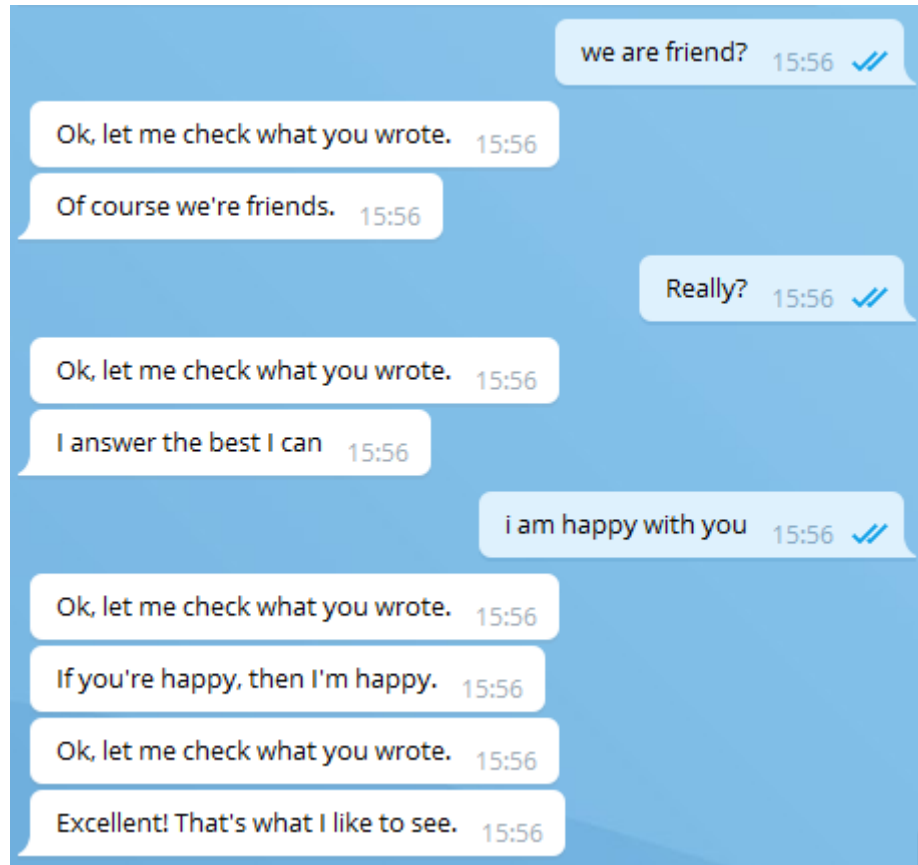


Рисунок В.7 – Спілкування з асистентом



Рисунок В.8 – Спілкування з асистентом



Рисунок В.9 – Спілкування з асистентом

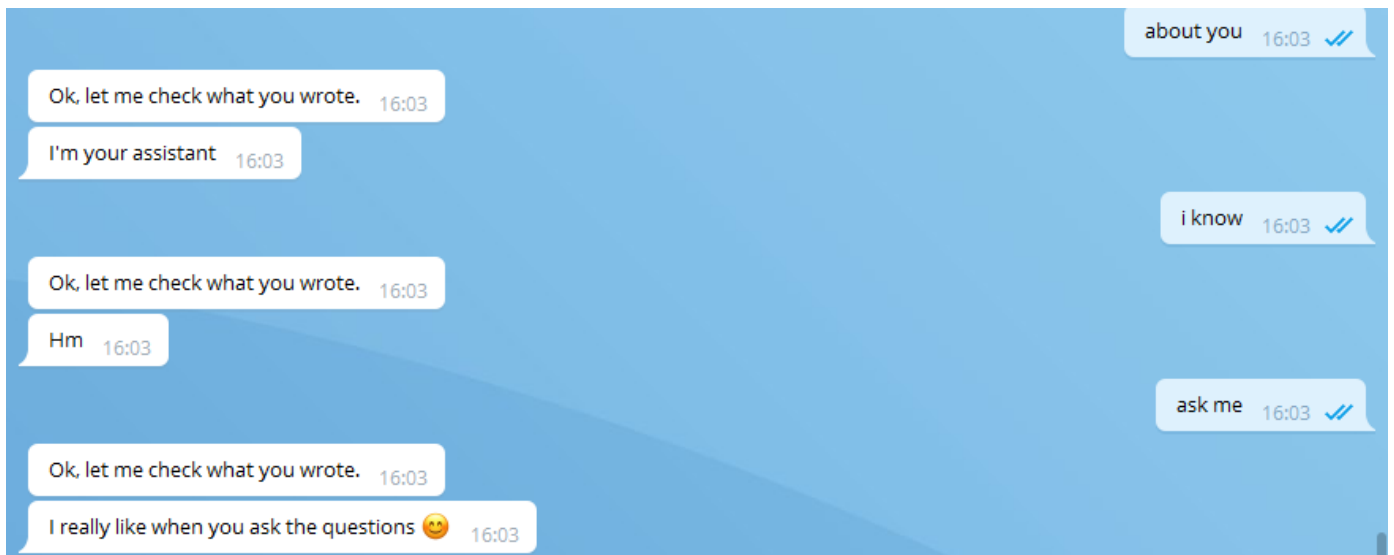


Рисунок В.10 – Спілкування з асистентом

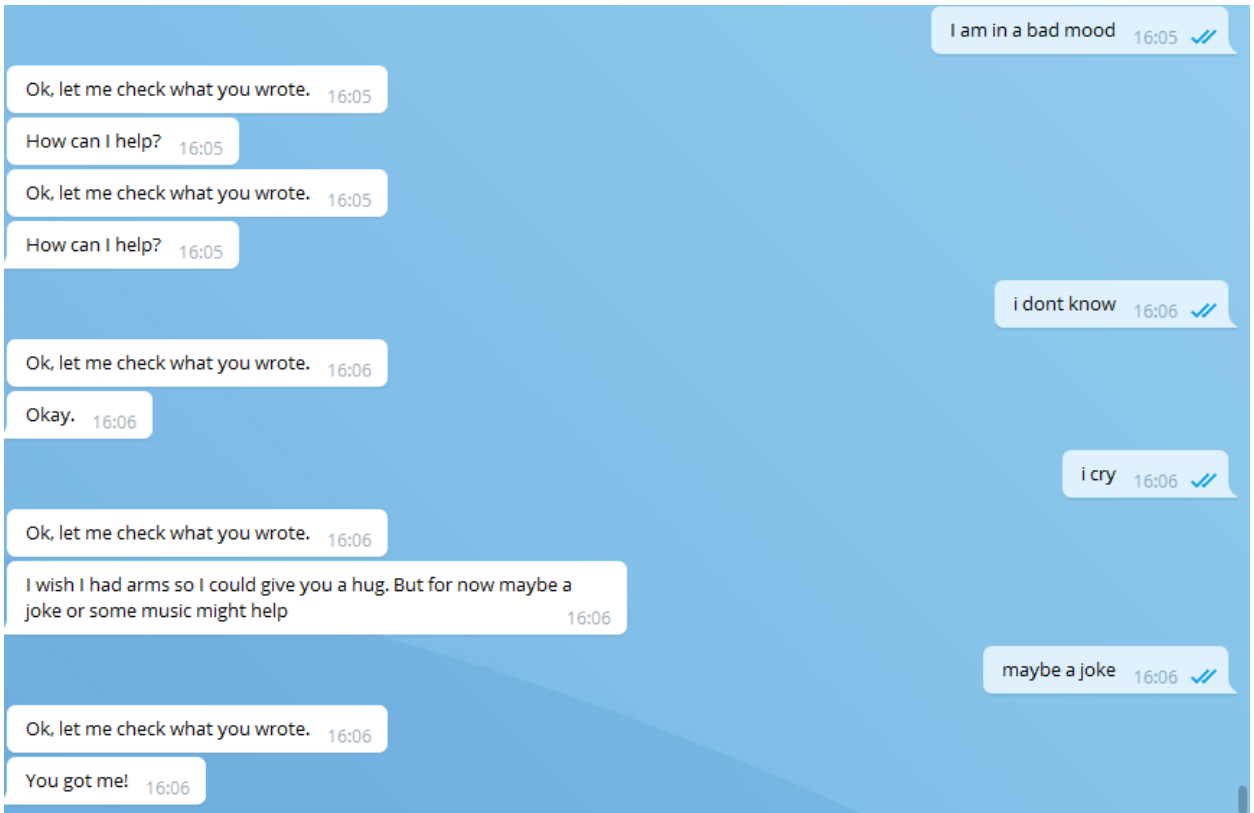


Рисунок В.11 – Спілкування з асистентом

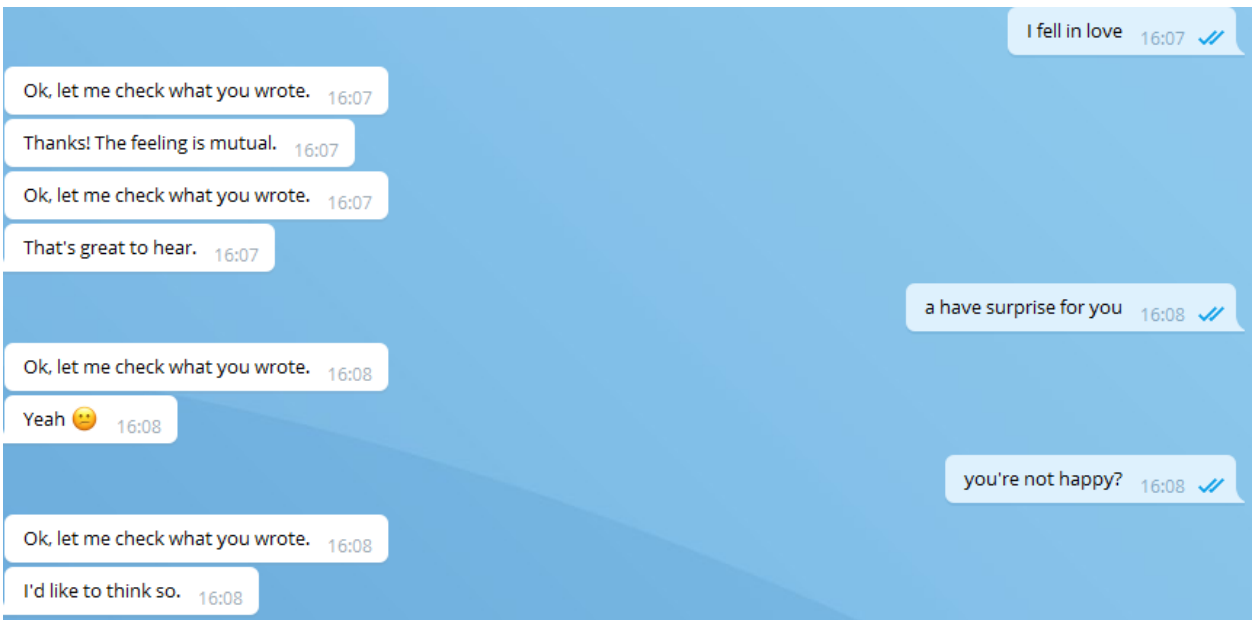


Рисунок В.12 – Спілкування з асистентом

Приклад 2

Діалог з ботом через Dialogflow

За допомогою сервісу Dialogflow була реалізоване спілкування за допомогою російської мови. Dialogflow був підключений для того, щоб бот був різностороннім та багато задачним. В цьому випадку всі відповіді прописані раніше розробником на ті питання, які запропонував саме сервіс (В.13).

QUESTION	Я очень зол.
ANSWER	<ol style="list-style-type: none"> 1 Успокойся! Просто расслабся и отдохни. 2 Enter a Answer variant
QUESTION	Я вернулся.
ANSWER	<ol style="list-style-type: none"> 1 Куда ходил? 2 Агде ты был? 3 Enter a Answer variant
QUESTION	Мне скучно.
ANSWER	<ol style="list-style-type: none"> 1 Ну так займись чем -то 2 Enter a Answer variant

Рисунок В.13 – Заповнення полів answer

Відповіді будуть генеруватись відповідно питанню. Приклади спілкування з ботом через сервіс Dialogflow зображені на В.14-В.18.

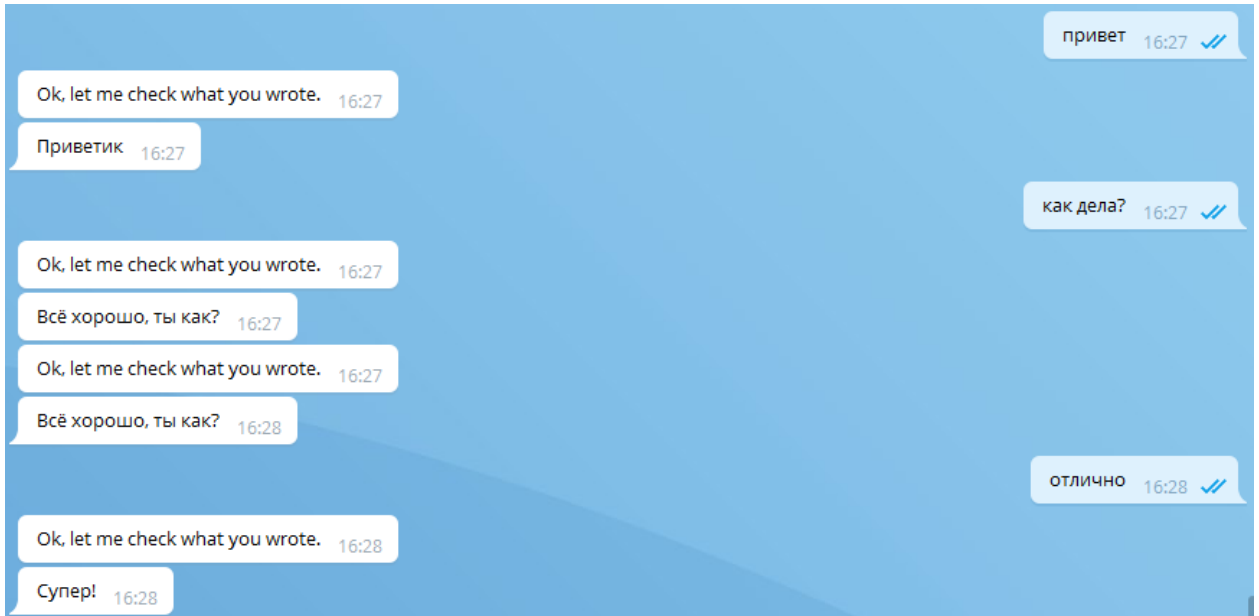


Рисунок В.14 – Спілкування з ботом через сервіс Dialogflow

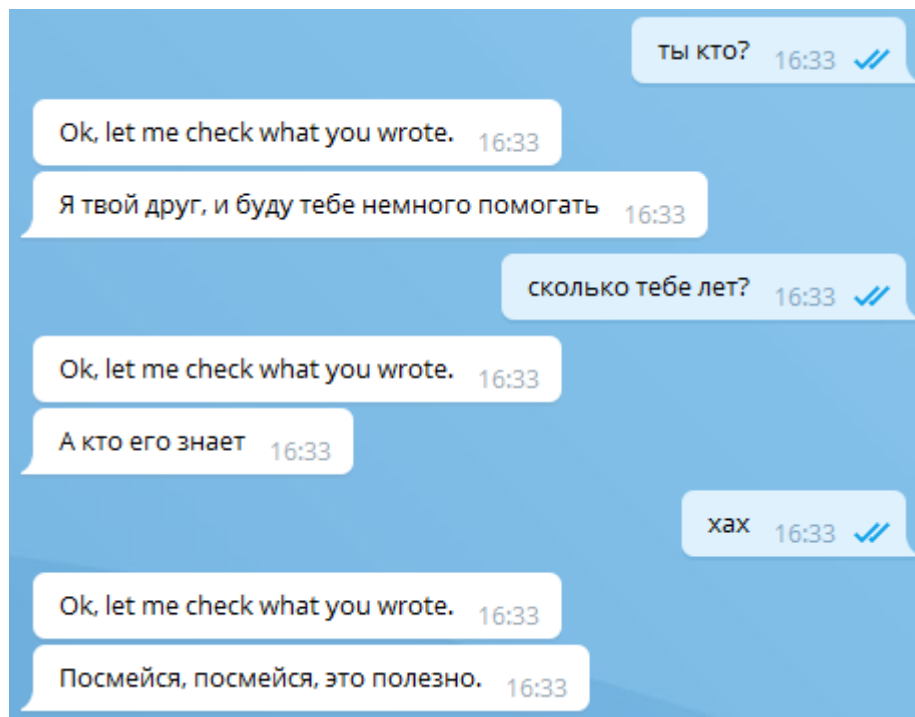


Рисунок В.15 – Спілкування з ботом через сервіс Dialogflow

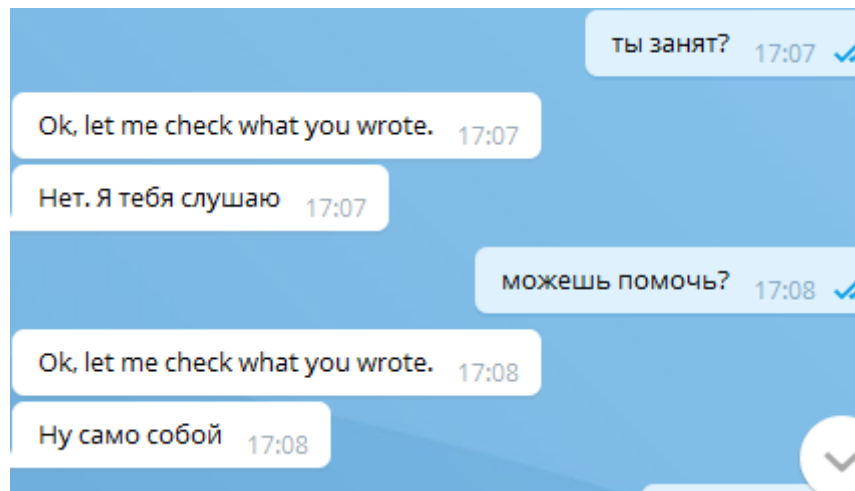


Рисунок В.16 – Спілкування з ботом через сервіс Dialogflow

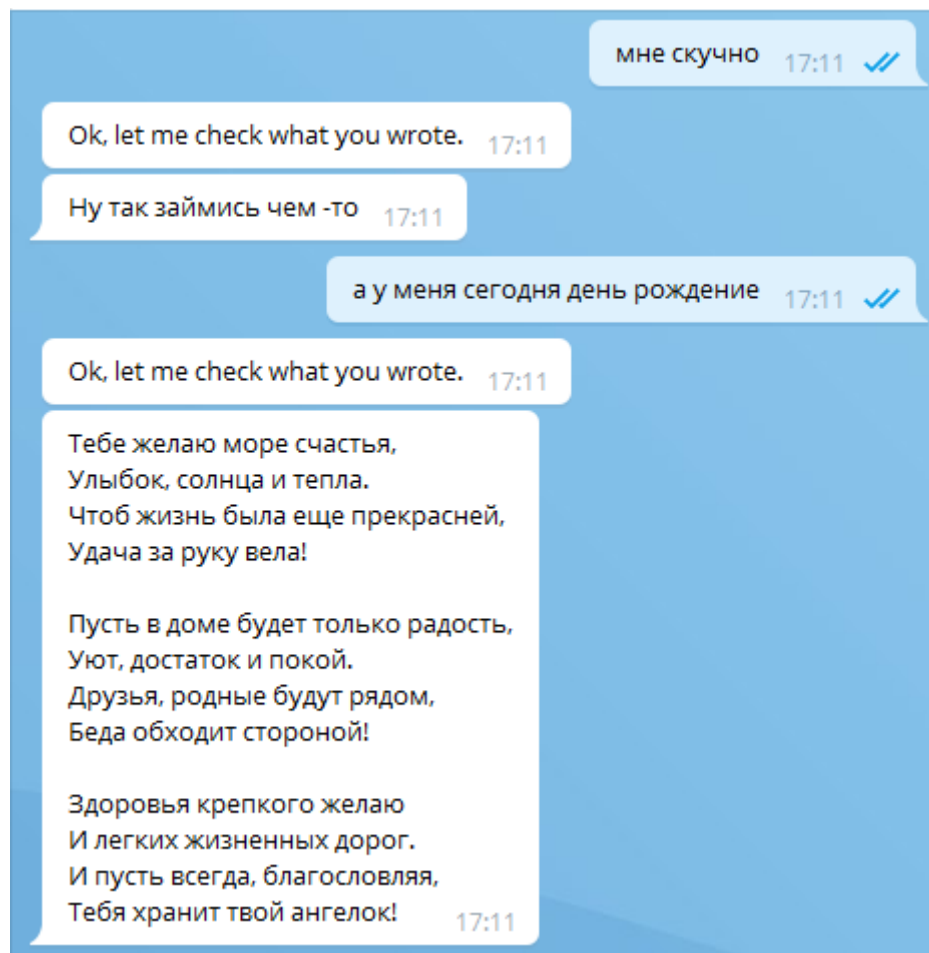


Рисунок В.17 – Спілкування з ботом через сервіс Dialogflow

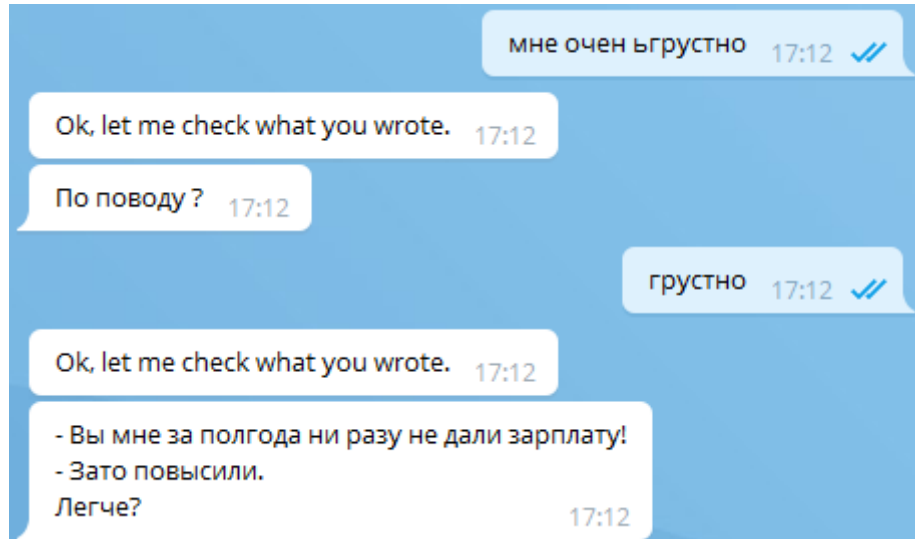


Рисунок В.18 – Спілкування з ботом через сервіс Dialogflow

Приклад 3

Пошук інформації

Пошук інформації здійснюється за допомогою голосового запиту, а також текстовим запитом. Результат залежить від того, яким чином буде задане питання. Відповідь на запит буде або від Google Search або Google Assistant. Google Assistant працює англійською мовою, як аудіо запит так і текстовий. Google Search голосовий запит англійською мовою, текстовий запит англійською та російською. Обробка запиту від Google Assistant голосові та текстові зображені на рис. В.19 - В.27.

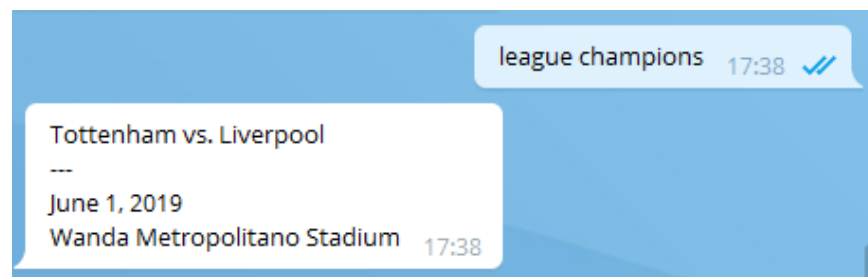


Рисунок В.19 – Текстовий пошук від Google Assistant



Рисунок В.20 – Текстовий пошук від Google Assistant

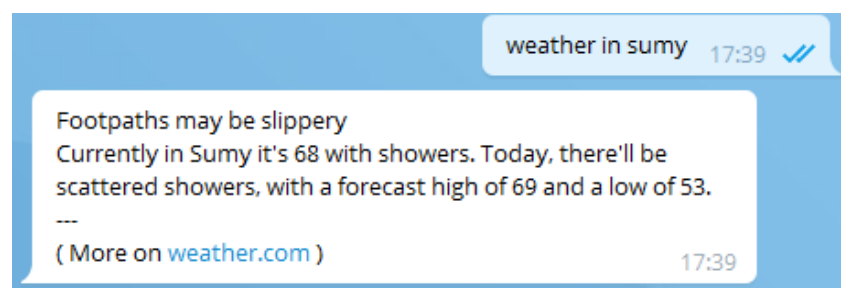


Рисунок В.21 – Текстовий пошук від Google Assistant

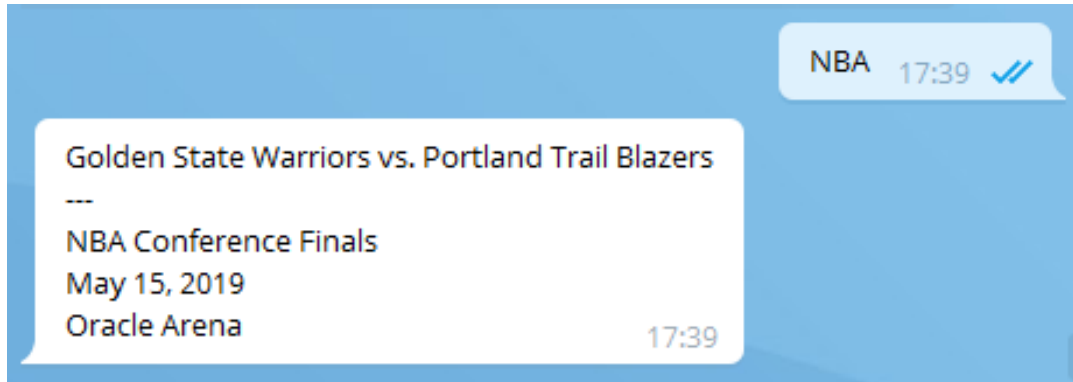


Рисунок В.22 – Текстовий пошук від Google Assistant



Рисунок В.23 – Текстовий пошук від Google Assistant



Рисунок В.24 – Голосовий пошук від Google Assistant

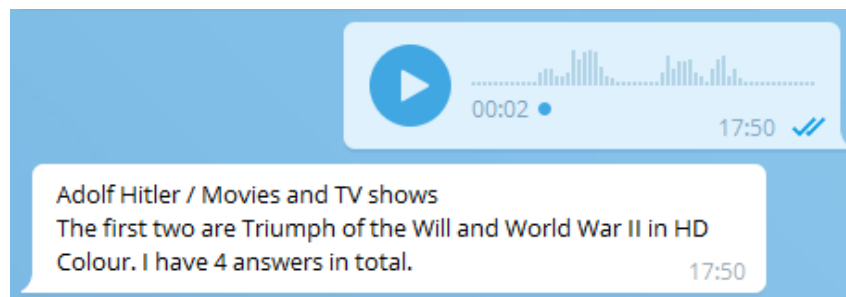


Рисунок В.25 – Голосовий пошук від Google Assistant

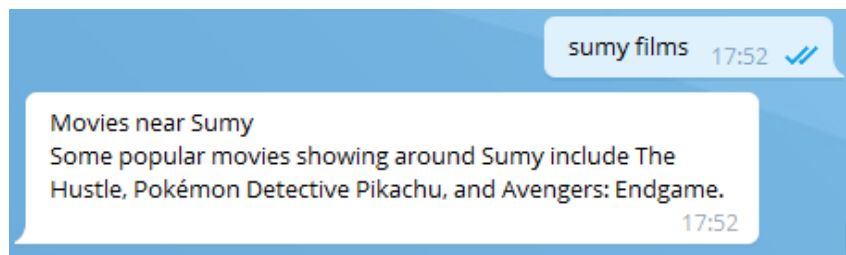


Рисунок В.26 – Текстовий пошук від Google Assistant

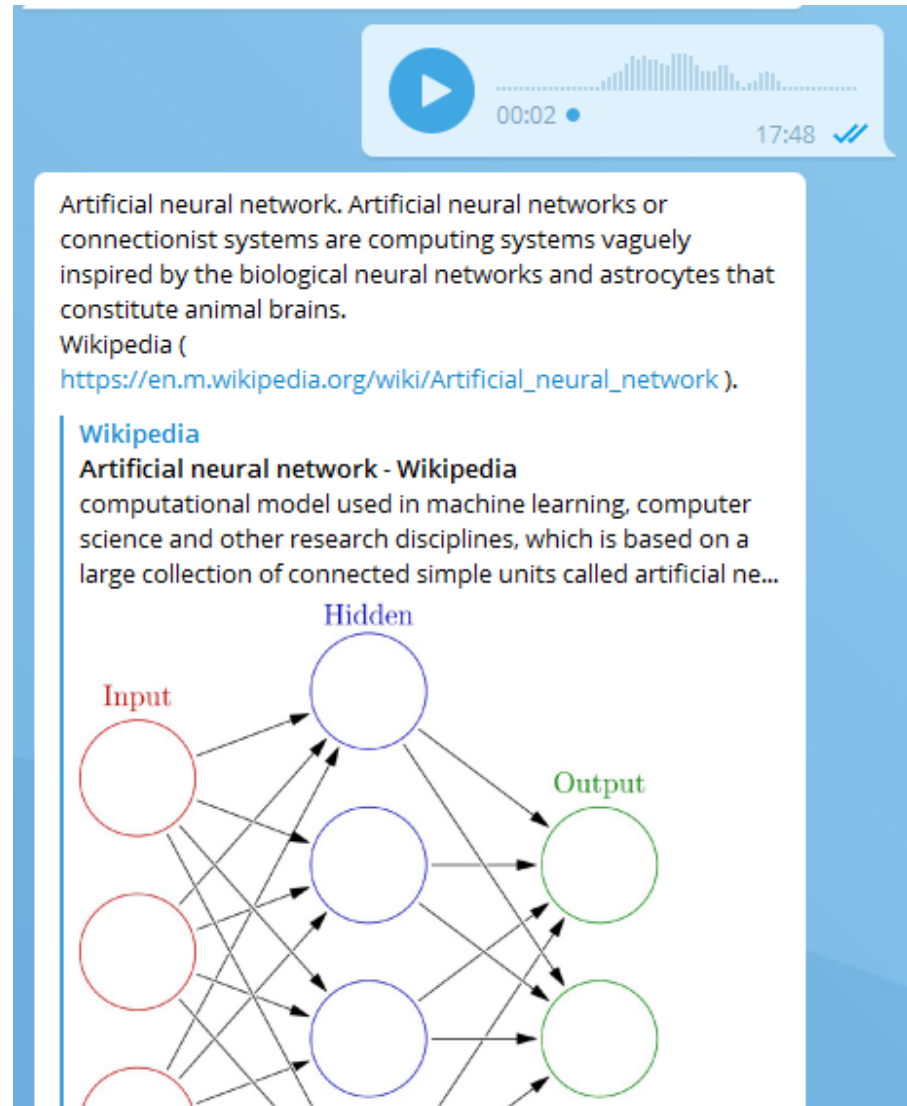


Рисунок В.27 – Голосовий пошук від Google Assistant

Обробка запиту від Google Search зображені на рис В.27 – 32.

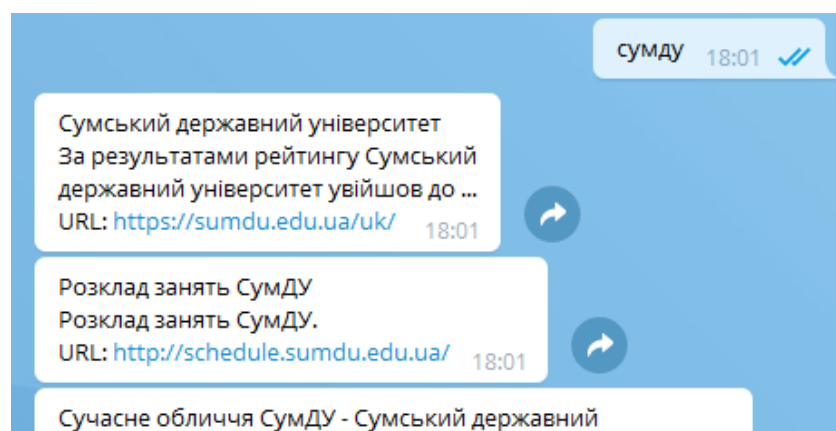


Рисунок В.28 – Текстовий пошук від Google Search

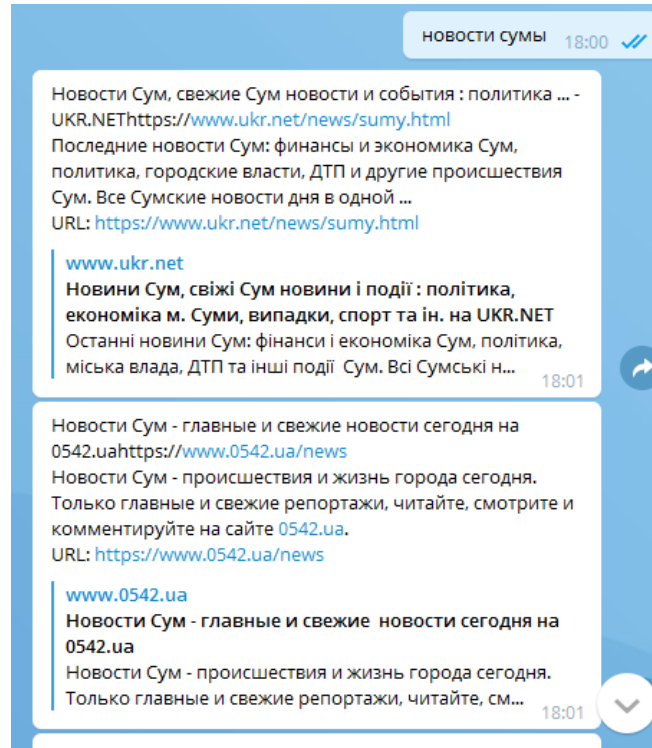


Рисунок В.29 – Текстовий пошук від Google Search

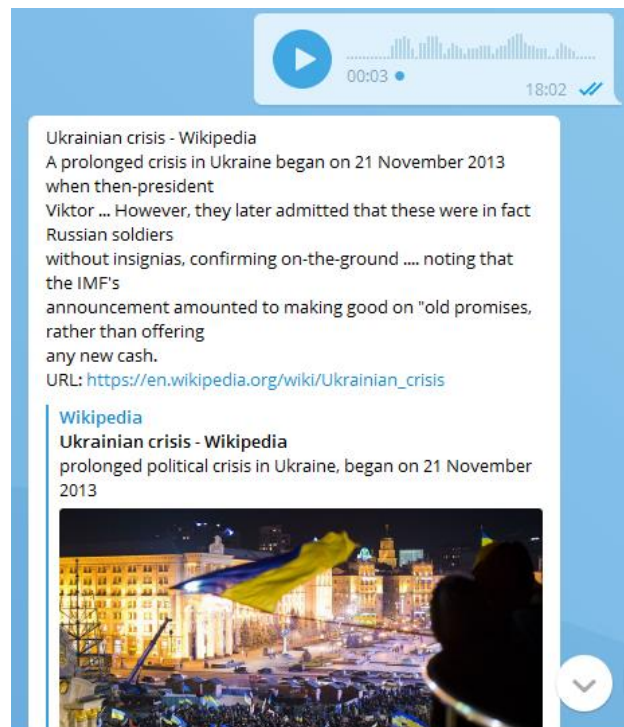


Рисунок В.30 – Голосовий пошук від Google Search

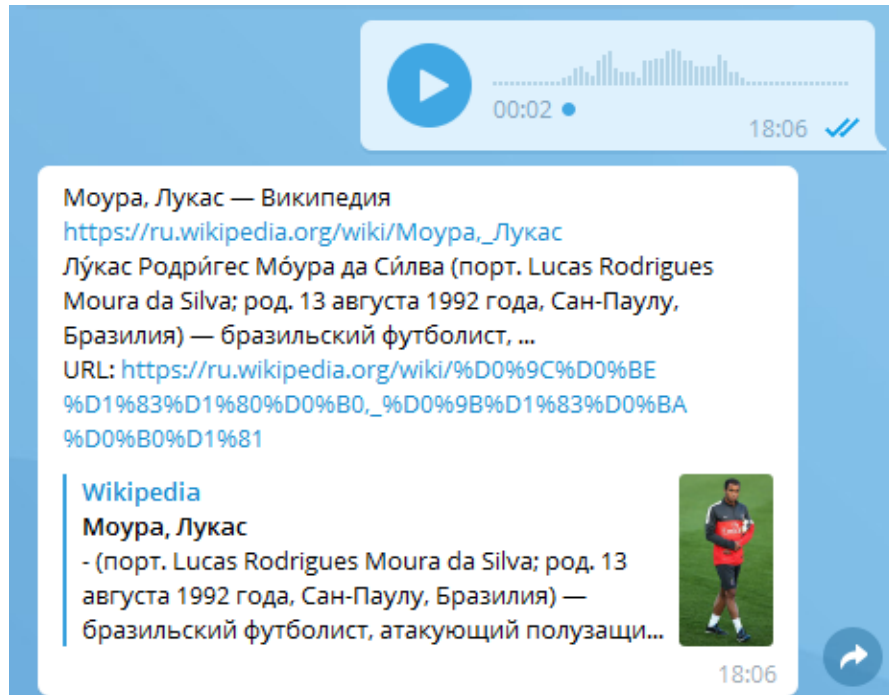


Рисунок В.31 – Голосовий пошук від Google Search

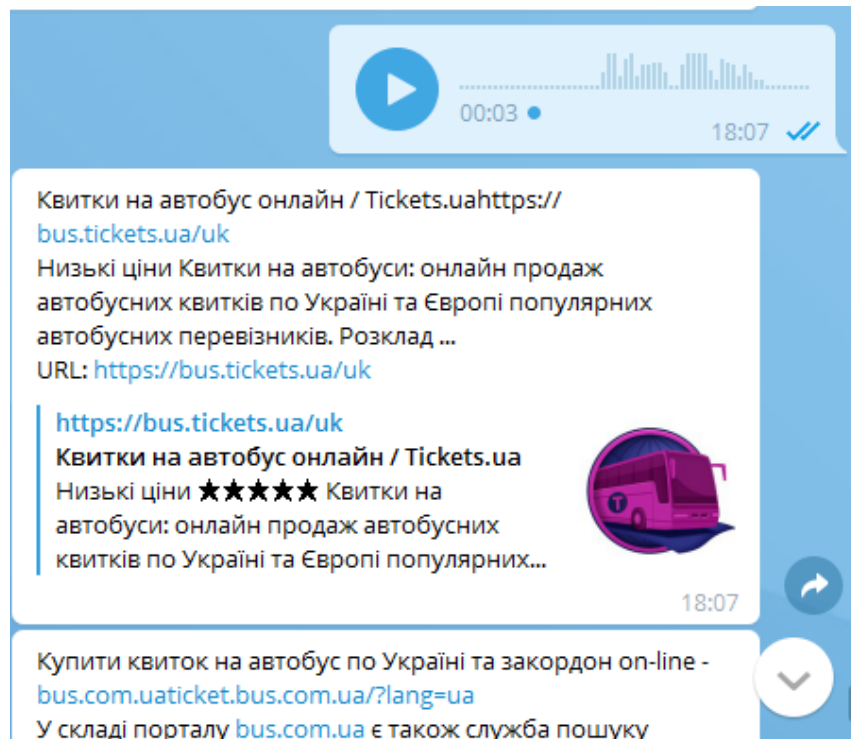


Рисунок В.32 – Голосовий пошук від Google Search

Додаток Г

Інструкція користувача

Для користування даним чат-ботом необхідно перейти за посиланням [@AssistantEduard_bot](#), в месенджері Telegram (рис.Г.1).

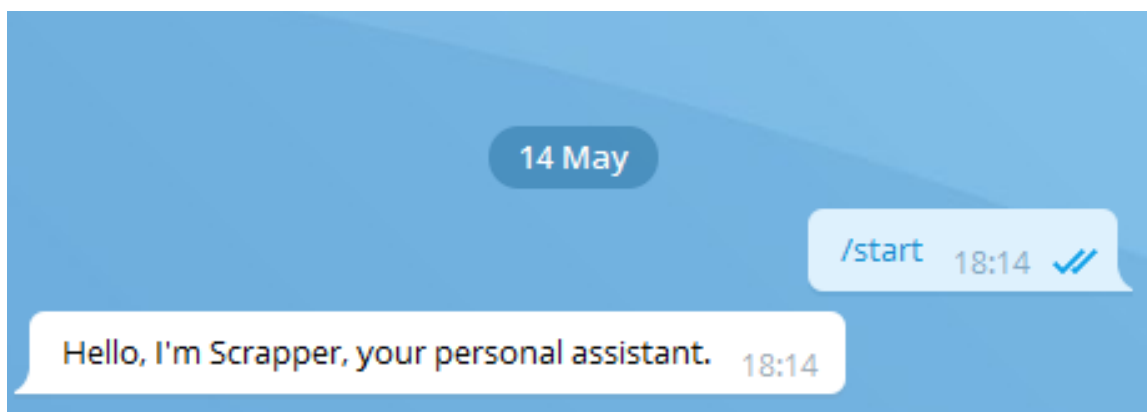


Рисунок Г.1 – Привітальне повідомлення від бота

Додаток Д

Файл коду assistant.py

```

import os
import traceback
import time

import telebot
import detectlanguage
import audiotranscode
import speech_recognition as sr

from pydub import AudioSegment

from send_text import main
from g_search import search_result
from df_connect import detect_intent_texts

from gtts import gTTS

APIKEY = 'eea82a6c343247408c73423b1b2f2d5c16e'

bot = telebot.TeleBot('632423476907230:AAEoEya4zob2ZSWgYSr7Hk6JG6nKVUvPLk4')

detectlanguage.configuration.api_key = "f01173453455c679cd1f096b2afre97a1cegnhtyjfb4ed67"

CHANNELS = 1
swidth = 2

@bot.mes_handler(commands=['start'], cont_types=['text'])
def send_welcome(mes):
    bot.send_mes(mes.chat.id, "Hello, I'm Scrapper, your personal assistant.")

@bot.mes_handler(commands=['help'])
def command_help(mes):
    help_text = "Welcome to help center"

    bot.send_mes(mes.chat.id, help_text)

@bot.mes_handler(commands=['here', 'example'])
def stop_command(mes):
    bot.send_mes(mes.chat.id, 'Who\'s Lady Gaga')

@bot.mes_handler(cont_types=['voice', 'text'])
def get_voice(mes):
    bot.send_mes(mes.chat.id, 'Ok, let me check what you wrote.')

if mes.cont_type == 'voice':
    file = bot.get_file(mes.voice.file_id)

    download_file = bot.download_file(file.file_path)

    name = 'input'

```

```

with open('{} .ogg'.format(name), 'wb') as new_voice:
    new_voice.write(download_file)

at = audiotranscode.AudioTranscode()
at.transcode('{} .ogg'.format(name), '{} .wav'.format(name))

result = recognize_audio(name, mes.cont_type)
elif mes.cont_type == 'text':
    result = recognize_audio(mes.text, mes.cont_type)
else:
    result = 'Unsupported type of mes.'

if type(result) == dict:
    print('FROM GOOGLE SEARCH')
    for respon in result['respon']:
        bot.send_mes(mes.chat.id, respon)

    bot.send_mes(mes.chat.id, 'That\'s all I found for your query. I\'m waiting for new tasks!')
else:
    bot.send_mes(mes.chat.id, result)

def recognize_audio(name, cont_type):
    if cont_type == 'voice':
        try:
            r = sr.Recognizer()
            with sr.AudioFile('{} .wav'.format(name)) as source:
                audio = r.record(source)

                text = r.recognize_google(audio)
                text = text.lower()
            except sr.UnknownValueError:
                return "I can't understand what are you talking about, sorry. Try again."
            except sr.RequestError as e:
                return "Could not request results from Google Speech Recognition service; {0}".format(e)
        else:
            text = name.lower()

    print(text)

    lang_code = detectlanguage.detect(text)
    print(lang_code)

    # try:
    if lang_code[0]['language'] == 'en':
        df_answer = detect_intent_texts(text, 'en')
    else:
        df_answer = detect_intent_texts(text, 'ru')

    if df_answer != None:
        return df_answer

    respon = None

    if lang_code[0]['language'] == 'en':
        respon = main(query=text)

    if respon != None:
        return respon

    return search_respon(text)

```

```

def telegram_pollng():
    try:
        bot.pollng(none_stop=True, timeout=60) #
    except:
        traceback_error_string=traceback.format_exc()
        with open("Error.Log", "a") as myfile:
            myfile.write("\r\n\r\n" + time.strftime("%c")+ "\r\n<<ERROR pollng>>\r\n" + traceback_error_string +
            "\r\n<<ERROR pollng>>")
        bot.stop_pollng()
        time.sleep(10)
        telegram_pollng()

if __name__ == '__main__':
    telegram_pollng()

```

Файл коду df_connect.py

```

import os

os.environ["GOOGLE_APPLICATION_CREDENTIALS"]="./assistant-e723c-a130aa935622.json"

def detect_intent_texts(text, lang_code):
    import dialogflow_v2 as dialogflow
    session_client = dialogflow.SesionClient()

    session = session_client.session_path('assistant-e723c', 'assistant-e723c')

    text_input = dialogflow.types.TextInput(
        text=text, language_code=lang_code)

    quer_input = dialogflow.types.QuerInput(text=text_input)

    respon = session_client.detect_intent(
        sesion=session, quer_input=quer_input)

    print(respon)
    if respon.quer_rezult.intent.display_name == 'Default Fallback Intent':
        return None

    print('=' * 20)
    print('Quer text: {}'.format(respon.quer_rezult.quer_text))
    print('Detected intent: {} (confidence: {})\n'.format(
        respon.quer_rezult.intent.display_name,
        respon.quer_rezult.intent_detection_confidence))
    print('Fulfillment text: {}\n'.format(
        respon.quer_rezult.fulfillment_text))
    print('=' * 20)

    return response.query_rezult.fulfillment_text

```

Файл коду g_search.py

```
import sys
sys.path.append('D:/diplom/Google-Search-API-master')

from google import google

num_page = 1

def search_result(query):
    print('GOT QUERY: ', query)
    search_results = google.search(query, num_page)

    return_results = []

    for result in search_results:
        return_results.append('{}\n{}\nURL: {}'.format(result.name, result.description, result.link))
        print(result.name, result.link, result.description)

    result = {'response': return_results}

    return result
```

Файл коду send_text.py

```
import os
import logging
import json

import click
import google.auth.transp.grpc
import google.auth.transp.zaproq
import google.oauth2.credentials

from google.assist.embedded.v1alpha2 import (
    embedded_assist_pb2,
    embedded_assist_pb2_grpc
)

ASSIST_API_ENDPOINT = 'embeddedassist.googleapis.com'
DEFAULT_GRPC_DEADLINE = 60 * 3 + 5
PLAYING = embedded_assist_pb2.ScreenOutConf.PLAYING

class SampleTxtAssistant(object):

    def __init__(ego, leng_coding, dece_model_id, dece_id,
                 ekr, channel, deadline_sec):
        ego.leng_coding = leng_coding
        ego.dece_model_id = dece_model_id
        ego.dece_id = dece_id
        ego.conv_state = None

        ego.is_new_conv = True
        ego.ekr = ekr
        ego.assist = embedded_assist_pb2_grpc.EmbeddedAssistStub(
```



```

        channel
    )
    ego.deadline = deadline_sec

def __enter__(ego):
    return ego

def __exit__(ego, etype, e, traceback):
    if e:
        return False

def assist(ego, txt_mas):

    def iter_assist_zaproq():
        conf = embedded_assist_pb2.AssistConf(
            audio_out_conf=embedded_assist_pb2.AudioOutConf(
                encoding='LINEAR16',
                sample_rate_hertz=16000,
                volume_percentage=0,
            ),
            dialog_state_in=embedded_assist_pb2.DialogStateIn(
                leng_coding=ego.leng_coding,
                conv_state=ego.convn_state,
                is_new_conv=ego.is_new_conv,
            ),
            dece_conf=embedded_assist_pb2.DeceConf(
                dece_id=ego.dece_id,
                dece_model_id=ego.dece_model_id,
            ),
            txt_mas=txt_mas,
        )

        ego.is_new_conv = False
        if ego.ekr:
            conf.screen_out_conf.screen_mode = PLAYING
            zap = embedded_assist_pb2.AssistZaproq(conf=conf)

            yield zap

    txt_quar = None
    html_vidp = None
    for resp in ego.assist.Assist(iter_assist_zaproq(),
                                  ego.deadline):
        if resp.screen_out.data:
            html_vidp = resp.screen_out.data
        if resp.dialog_state_out.conv_state:
            conv_state = resp.dialog_state_out.conv_state
            ego.conv_state = conv_state
        if resp.dialog_state_out.supplemental_ekr_txt:
            txt_vidp = resp.dialog_state_out.supplemental_ekr_txt
    return txt_vidp, html_vidp

def main(mas,
        api_endpoint=ASSIST_API_ENDPOINT,
        credentials='/home/kirillkiyko/.conf/google-oauthlib-tool/credentials.json',

```

```

dece_model_id='yantoassist-yantoproduct-huxtov',
dece_id='ED82E54B6A31113A5CFB2578F908D807',
ekr=False,
grpc_deadline=DEFAULT_GRPC_DEADLINE,
verbose=False,
lang='en-US'):

logging.basicConfig(level=logging.DEBUG if verbose else logging.INFO)
try:
    with open(credentials, 'r') as f:
        credentials = google.oauth2.credentials.Credentials(token=None,
                                                            **json.load(f))

        http_zaproq = google.auth.transp.zaproq.Zaproq()
        credentials.refresh(http_zaproq)
except Exception as e:
    logging.err('Err loading credentials: %s', e)
    logging.err('Run google-oauthlib-tool to initialize '
               'new OAuth 2.0 credentials.')
return

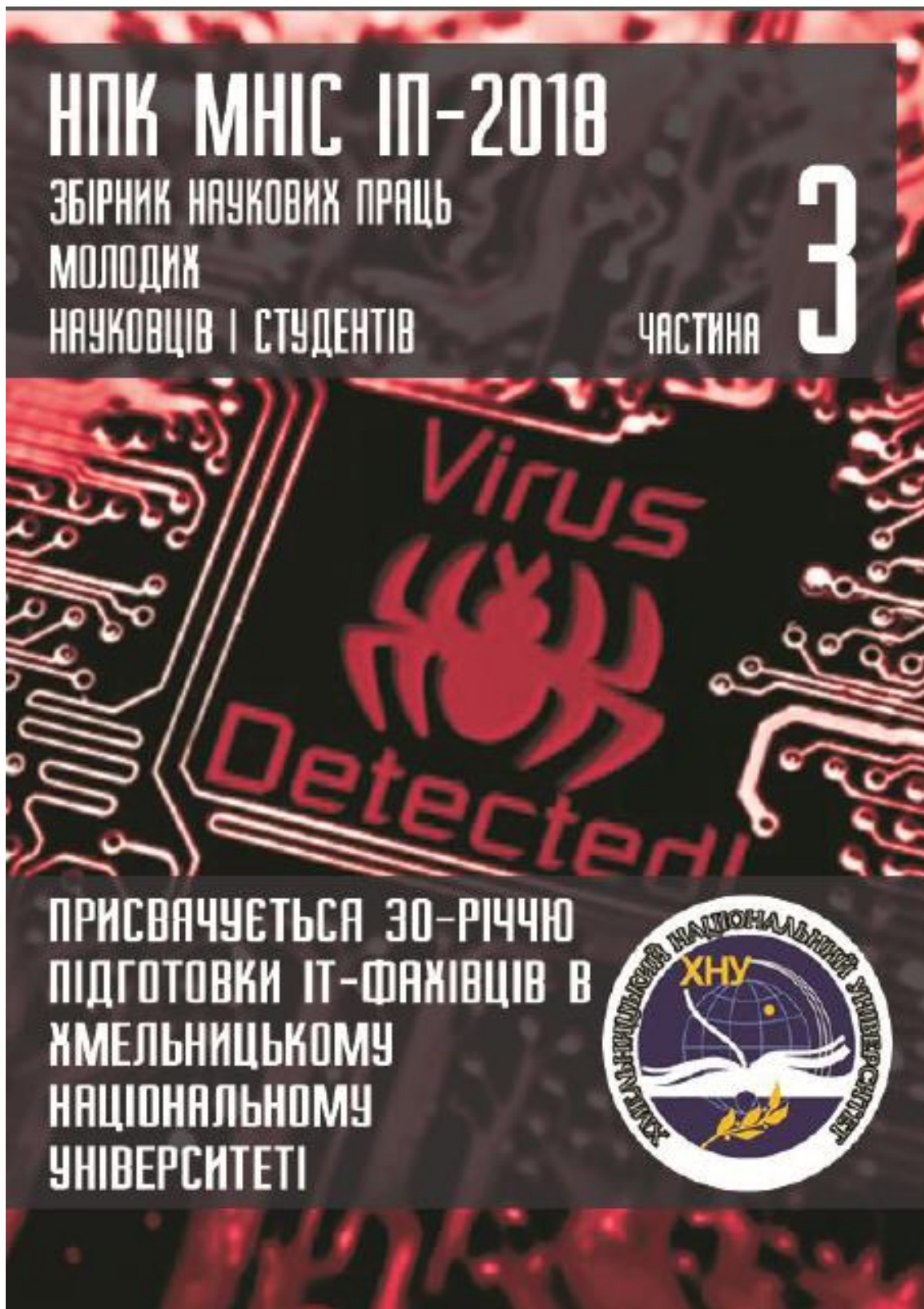
grpc_channel = google.auth.transp.grpc.secure_authorized_channel(
    credentials, http_zaproq, api_endpoint)

logging.info('Connecting to %s', api_endpoint)
with SampleTxtAssist(lang, dece_model_id, dece_id, ekr,
                    grpc_channel, grpc_deadline) as assist:
    while True:
        mas = mas
        vidp_txt, vidp_html = assist.assist(txt_mas=mas)
        return vidp_txt

```

Додаток Е

Копії публікацій



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Хмельницький національний університет

Військовий інститут Київського національного університету
ім. Тараса Шевченка

ПВНЗ “Університет економіки і підприємництва”

Тернопільський інститут агропромислового виробництва

Інтелектуальний потенціал - 2018

збірник наукових праць молодих науковців і студентів

**Присвячується 30-річчю підготовки ІТ- фахівців в
Хмельницькому національному університеті**

сформовано за матеріалами

Всеукраїнської науково-практичної конференції
молодих науковців і студентів «Інтелектуальний потенціал – 2018»

14-16 листопада 2018р.

Частина 3

Кібербезпека та актуальні проблеми комп'ютерних систем і мереж

Хмельницький
2018

Аналіз проблем людського фактору в задачах забезпечення кібербезпеки

Кіншаков Е., Щербань Т.

Науковий керівник – професор Лавров Е.А.

Сумський державний університет, Суми, Україна

Проблеми кібербезпеки, набули надзвичайної актуальності. Інформаційна безпека (ІБ) складається з цілого комплексу різних заходів і дій. Це, перш за все, контроль дій різного роду суб'єктів - рядових співробітників компанії, привілейованих користувачів, ІТ-аутсорсерів, контрагентів. Крім того, це чітке розмежування прав доступу всередині компанії, використання резервного копіювання даних, а також наявність простої, зрозумілої і доведеної до відома працівників політики безпеки. У поточних реаліях захист повинен бути досить гнучким, щоб забезпечити і достатній рівень захищеності, і виконання бізнес-цілей. Згідно з інформацією, яка міститься в дослідженні Lloyd's of London і Cyence, фінансові втрати від масштабної кібератаки можуть коштувати світовій економіці від 15,6 млрд до 121 млрд доларів. Якщо розглядати найбільш песимістичний сценарій розвитку подій, то втрати від кібератак можуть перевищити економічний збиток від урагану «Катріна», який став найбільш руйнівним в історії Сполучених Штатів. Втрати від нього склали 108 млрд доларів. У доповіді вказуються два потенційних сценарію розвитку глобальної кібератаки: злом провайдерів хмарних сховищ або використання можливих вразливостей в операційних системах.

У першому сценарії хакери модифікують «гіпервизор», керуючу систему хмарних сховищ, в результаті чого всі зберігаються файли виявляються загубленими. У другому варіанті розглядається гіпотетичний випадок, коли кібераналітик випадково забуває в поїзді сумку, в якій зберігається доповідь про уразливість всіх версій операційної системи, встановленої на 45% всіх світових пристроїв. Ця доповідь згодом продається кримінальним групам. Мінімальний збиток при першому сценарії складе від 4,6 млрд до 53,1 млрд доларів. При другому сценарії втрати складуть від 9,7 млрд до 28,7 млрд доларів.

Людський фактор. Саме проблема «надійних рук» або, кажучи іншими словами, кваліфікованих кадрів є однією з найбільш нагальних. Вона має особливу актуальність протягом усіх останніх років, тому що на сьогоднішній день людина залишається найбільш уразливим ланкою в ІТ-інфраструктурі. Найслабша ланка в інформаційній безпеці банку - це співробітник компанії. Якщо співробітники не дотримуються правил безпеки, то технології не зможуть допомогти захиститися.

Так, при використанні соціальної інженерії зловмисники можуть змусити співробітника організації здійснити якусь дію, яке спростить проведення атаки, пояснює експерт. «Часто, щоб підібрати пароль до аккаунту, зловмиснику не обов'язково його «зламувати» - вся інформація про

пароль є в профілі соціальних мереж або поруч з робочим столом. Навіть співробітники на керівних позиціях виробляють маніпуляції, спровоковані зловмисниками. Окремим рядком можна привести небажання працівників слідувати політиці і вимогам по ІБ заради спрощення своєї роботи». Щоб мінімізувати вплив людського фактора, потрібно постійно підвищувати обізнаність співробітників в області ІБ, а також впроваджувати систему контролів і моніторингу дотримання політик і вимог в області ІБ. Серед основних способів мінімізації загрози ІБ -підвищення обізнаності персоналу в питаннях ІБ, проведення тестів, ділових ігор, кібернавчаль.

У зв'язку з проблемою ризиків, які несе людський фактор, цікаво згадати дослідження антивірусної компанії ESET, опубліковане в липні 2017 року. Чотири компанії з п'яти недооцінюють ризики ІБ, пов'язані з людським фактором. Такий висновок зробили співробітники ESET після опитування інтернет-користувачів з СНД. Респондентам запропонували вибрати відповідь на питання «Чи проходили ви на роботі тренінг з інформаційної безпеки?». Негативна відповідь лідирує з великим відривом. 69% респондентів ніколи не проходили навчання основам кібербезпеки в своїх компаніях. Ще 15% учасників опитування повідомили, що їх роботодавці обмежилися мінімальним обсягом інформації. Навчання не виходило за рамки «в разі неполадок перезавантажте комп'ютер», правила кібербезпеки не зачіпалися. Тільки 16% респондентів пройшли якісні тренінги з докладною розповіддю про інформаційну безпеку. Для порівняння: більше 60% учасників аналогічного опитування в США повідомили, що їх роботодавці організували для них навчання з кібербезпеки.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

ІНФОРМАТИКА, МАТЕМАТИКА,
АВТОМАТИКА

ІМА :: 2018

**МАТЕРІАЛИ
та програма**

НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

(Суми, 5-9 лютого 2017 року)



Суми
Сумський державний університет
2018

Аналіз проблем людського фактору в задачах забезпечення кібербезпеки

Щербань Т., студент; Кіншаков Е. студент; Лавров Е.А., професор
Сумський державний університет, Суми, Україна

Проблеми кібербезпеки, набули надзвичайної актуальності. Інформаційна безпека (ІБ) складається з цілого комплексу різних заходів і дій. Це, перш за все, контроль дій різного роду суб'єктів - рядових співробітників компанії, привілейованих користувачів, ІТ-аутсорсерів, контрагентів. Крім того, це чітке розмежування прав доступу всередині компанії, використання резервного копіювання даних, а також наявність простої, зрозумілої і доведеної до відома працівників політики безпеки. У поточних реаліях захист повинен бути досить гнучким, щоб забезпечити і достатній рівень захищеності, і виконання бізнес-цілей. Згідно з інформацією, яка міститься в дослідженні Lloyd's of London і Cyence, фінансові втрати від масштабної кібератаки можуть коштувати світовій економіці від 15,6 млрд до 121 млрд доларів. Якщо розглядати найбільш песимістичний сценарій розвитку подій, то втрати від кібератак можуть перевищити економічний збиток від урагану «Катріна», який став найбільш руйнівним в історії Сполучених Штатів. Втрати від нього склали 108 млрд доларів. У доповіді вказуються два потенційних сценарію розвитку глобальної кібератаки: злом провайдерів хмарних сховищ або використання можливих вразливостей в операційних системах.

У першому сценарії хакери модифікують «гіпервизор», керуючу систему хмарних сховищ, в результаті чого всі зберігаються файли виявляються загубленими. У другому варіанті розглядається гіпотетичний випадок, коли кібераналітик випадково забуває в поїзді сумку, в якій зберігається доповідь про уразливість всіх версій операційної системи, встановленої на 45% всіх світових пристроїв. Ця доповідь згодом продається кримінальним групам. Мінімальний збиток при першому сценарії складе від 4,6 млрд до 53,1 млрд доларів. При другому сценарії втрати складуть від 9,7 млрд до 28,7 млрд доларів.

Людський фактор. Саме проблема «надійних рук» або, кажучи іншими словами, кваліфікованих кадрів є однією з найбільш

нагальних. Вона має особливу актуальність протягом усіх останніх років, тому що на сьогоднішній день людина залишається найбільш уразливим ланкою в IT-інфраструктурі. Найслабша ланка в інформаційній безпеці банку - це співробітник компанії. Якщо співробітники не дотримуються правил безпеки, то технології не зможуть допомогти захиститися.

Так, при використанні соціальної інженерії зловмисники можуть змусити співробітника організації здійснити якусь дію, яке спростить проведення атаки, пояснює експерт. «Часто, щоб підібрати пароль до аккаунту, зловмиснику не обов'язково його «зламувати» - вся інформація про пароль є в профілі соціальних мереж або поруч з робочим столом. Навіть співробітники на керівних позиціях виробляють маніпуляції, спровоковані зловмисниками. Окремим рядком можна привести небажання працівників слідувати політиці і вимогам по ІБ заради спрощення своєї роботи». Щоб мінімізувати вплив людського фактора, потрібно постійно підвищувати обізнаність співробітників в області ІБ, а також впроваджувати систему контролів і моніторингу дотримання політик і вимог в області ІБ. Серед основних способів мінімізації загрози ІБ - підвищення обізнаності персоналу в питаннях ІБ, проведення тестів, ділових ігор, кібернавчань.

У зв'язку з проблемою ризиків, які несе людський фактор, цікаво згадати дослідження антивірусної компанії ESET, опубліковане в липні 2017 року. Чотири компанії з п'яти недооцінюють ризики ІБ, пов'язані з людським фактором. Такий висновок зробили співробітники ESET після опитування інтернет-користувачів з СНД. Респондентам запропонували вибрати відповідь на питання «Чи проходили ви на роботі тренінг з інформаційної безпеки?». Негативна відповідь лідирує з великим відривом. 69% респондентів ніколи не проходили навчання основам кібербезпеки в своїх компаніях. Ще 15% учасників опитування повідомили, що їх роботодавці обмежилися мінімальним обсягом інформації. Навчання не виходило за рамки «в разі неполадок перезавантажте комп'ютер», правила кібербезпеки не зачіпалися. Тільки 16% респондентів пройшли якісні тренінги з докладною розповіддю про інформаційну безпеку. Для порівняння: більше 60% учасників аналогічного опитування в США повідомили, що їх роботодавці організували для них навчання з кібербезпеки.



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ



ІНФОРМАТИКА, МАТЕМАТИКА, АВТОМАТИКА

ІМА - 2019

**МАТЕРІАЛИ
та програма**

**НАУКОВО-ТЕХНІЧНОЇ
КОНФЕРЕНЦІЇ**

(Суми, 23-26 квітня 2019 року)

Суми,
Сумський державний університет
2019

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

ІНФОРМАТИКА, МАТЕМАТИКА,
АВТОМАТИКА

ІМА :: 2019

МАТЕРІАЛИ
та програма

НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

(Суми, 23–26 квітня 2019 року)

Суми
Сумський державний університет
2019

Пошуковий чат-бот для мережі Telegram

Кіншаков Е.В., студент; Лавров Є.А., професор
Сумський державний університет, м. Суми, Україна

Величезна кількість часу, який люди проводять в Facebook, Telegram, Viber, робить месенджери швидко зростаючим сегментом на ринку додатків. Google Assistant на сьогоднішній день у смартфонах є невід'ємною частиною. Він окрім пошуку інформації, синхронізує всі сервіси девайса. Для користувачів, телефони яких не підтримують жодного асистента, виникає суттєва проблема в інформаційному забезпеченні. В зв'язку з масовим переходом користувачів до використання месенджеру Telegram, загострюється проблема пошуку інформації, що спонукає до розробки відповідного чат-боту. В зв'язку з проблемами ефективного використання Telegram для ряду девайсів, необхідно: розробити принципи побудови ефективної пошукової системи в соціальній мережі Telegram; реалізувати алгоритми взаємодії користувача з всесвітньою мережею; дослідити ефективність чат-боту і намітити шляхи його вдосконалення.

Чат-бот, реалізований за допомогою скриптової мови Python, забезпечує: голосовий пошук; текстовий пошук; спілкування. Логічна структура чат-боту показана на рисунку 1

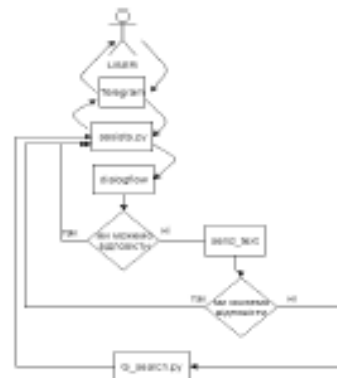


Рисунок 1 – Принцип роботи чат-бота

Дослідна експлуатація чат боту довела конструктивність підходу. Напрямок перспективних досліджень – розширення можливостей API.

Аналіз тенденцій ринку і технологій створення чат-ботів

Кіншаков Е.В., студент; Лавров Є.А., професор
Сумський державний університет, м. Суми, Україна

Постановка задачі. Цифровізація економіки і нова тенденція орієнтації систем підтримки прийняття рішень на використання мобільних пристроїв ставлять задачу розробки нового покоління чат - ботів. Метою даної роботи є дослідження тенденцій і перспективних засобів створення чат-ботів для індивідуального користування і бізнесу.

Стан світового ринку чат-ботів і близьких технологій. Виходячи з поточної оцінки стану ринку, чат-боти, особливо для месенджерів – дуже перспективний напрям, який в даний час переживає бурхливе зростання (рис 1.)

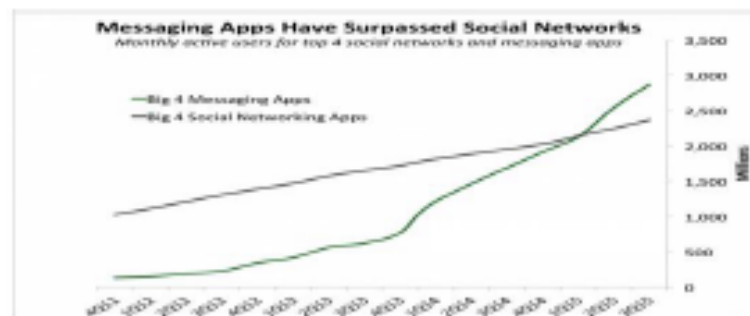


Рисунок – 1 Тенденції ринку чат-ботів

За оцінкою аналітиків у найближчому майбутньому чат - боти будуть мати все більшу значимість. Вони можуть повністю замінити класичні пошукові двигачи і соціальні мережі. Переваги ботів забезпечують зручність людино-комп'ютерної взаємодії, швидкість реакції, можливість їх налаштування під користувача. Прикладом може бути Facebook чат-бот M, який в поточний момент знаходиться в режимі бета-тестування – це персональний цифровий асистент всередині месенджера, що виконує прості задачі (в т.ч. рекомендації з надання послуг). Технічно Facebook-M оснований на сучасних технологіях машинного навчання (глибинні нейромережі). Крім Facebook, в напрямку розвитку технологій чат-ботів активно рухається Google і Apple. Приблизні аналоги Facebook-M вже створені в Китаї - наприклад,

Xiaolce. На ринку більш простих ботів лідерами є такі сервіси, як Slack і Telegram. Для них було створено безліч простих допоміжних чат-ботів, наприклад, для пошуку картинок в інтернеті або нагадування про важливі справи.

Аналіз фреймворків і технологій для створення чат-ботів. Чат-боти для месенджерів. Існує безліч бібліотек для різних мов програмування, а також API для створення чат - ботів у Slack. Приклади бібліотек та API для створення ботів в Telegram наведені в [1]. Фреймворки для створення чат-ботів. API для створення чат-ботів і послуг по їх створенню надають такі іноземні компанії як Pandorabots [2] і Synthetic Intelligence Network [3]. Pandorabots надає доступ до свого API на базі достатньо простої XML-подібної скриптової мови AIML (Artificial Intelligence Markup Language). Synthetic Intelligence Network представляє мову SIML (Synthetic Intelligence Markup Language), концептуально схожу з AIML і створену на базі її, а також свою власну розробку ботів – Syn Chatbot Studio під dot-net [4]. Також досить відомими фреймворками для створення власних ботів є Eitbot на Python [5] і Hubot [6] (можуть бути розміщені на Slack, Telegram, IRC, тощо).

Висновки. Створення чат-бота, навіть з урахуванням використання останніх технологічних розробок, передбачає велике людське втручання на всіх етапах розробки (в першу чергу для задач шаблонів реакцій на повідомлення користувача). Створення чат-бота (наприклад, для розвантаження онлайн-консультантів при відповідях на стандартні запитання) не вимагає складних технологій, достатньо базових технологій обробки мови. Існує достатня кількість фреймворків і API, які можуть бути використані для створення чат-ботів. Крім того, фреймворк для обробки мови для комерційного чат-бота можна створити самостійно на базі різних програмних бібліотек з відкритим вихідним кодом.

1. <https://vc.ru/p/telegram-bots-business>
2. <http://www.pandorabots.com>
3. <http://simlbot.com>
4. https://www.chatbots.org/ai_zone/viewthread/1903/
5. <http://errbot.io/en/latest/>
6. <https://hubot.github.com/>

Додаток Є

Копії грамот переможця конкурсу наукових робіт



ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ

АВТОМОБІЛЬНО-ДОРОЖНИЙ



ДИПЛОМ

нагороджується

ДИПЛОМОМ **III** СТУПЕНЯ

студент

Сумського державного університету

Кіншаков Едуард Віталійович

за наукову роботу у Всеукраїнському конкурсі студентських наукових робіт з

ОХОРОНИ ТРАЦІ

19 квітня 2019 р.



Ректор

А.М.Туренко



Додаток Є.**Копії акту впровадження**

Довідка про
впровадження в ТОВ "СУ" Сумтехсервіс
результатів дипломного проекту

студента групи ІТ-51-6 Сумського державного університету
Кіншакова Едуарда Віталійовича
на тему
«Інформаційний чат-бот для соціальної мережі Telegram з використанням
Google Assistant та Google Search API»

Передано комп'ютерну програму – чат-бот «Assistant», для підтримки
діяльності керівників і служб підприємства.

Керівник підприємства

