

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

Навчально-науковий інститут Інформаційних технологій

Кафедра інженерії програмного забезпечення

Пояснювальна записка

до бакалаврської роботи

на ступінь вищої освіти бакалавр

на тему: «Розробка чат-боту для туристичного агентства UNID TRAVEL
мовою Python »

Виконав: студент 5 курсу, групи ППЗ– 51
спеціальності

121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

Струпінський І.Ю.
(прізвище та ініціали)

Керівник Коба А.Б.
(прізвище та ініціали)

Рецензент_
(прізвище та ініціали)

Нормоконтроль_
(прізвище та ініціали)

Київ – 2022

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

Навчально-науковий інститут Інформаційних технологій

Кафедра Інженерії програмного забезпечення Ступінь вищої освіти «Бакалавр»

Напрямок підготовки -121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного забезпечення

- О.В. Негоденко

“16” лютого 2022 року

ЗАВДАННЯ

НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ

Струпінський Ілля Юрійович

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка чат-боту для туристичного агенства UNID TRAVEL мовою Python»

Керівник роботи Коба А.Б.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “18” лютого 2021 року № 22.

2. Строк подання студентом роботи 3.06.2022

3. Вихідні дані до роботи:

3.1. Існуючі Чат-боти

3.2. Середовище розробки Pycharm;

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити).

4.1. Проаналізувати та виділити переваги розробленого ПЗ;

4.2. Обґрунтувати обрані програмні застосунки та шляхи розробки додатку.

4.3. Розробити програмне забезпечення та користувацький інтерфейс.

5. Перелік графічного матеріалу

1. Діаграми діяльності, стану, схеми

2. Скріни програми.

6. Дата видачі завдання

КАЛЕНДАРНИЙ ПЛАН

№ з / п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Аналіз чат-ботів, які існують	11.04-14.04	
2	Встановлення вимог до чат-боту	15.04-17.04	
3	Розробити чат-бот для зібрання інформації	18.04-21.04	
4	Основні етапи технології розробки чат-боту	22.04-23.04	
5	Тестування розробки	24.04-05.05	
6	Оформлення матеріалів кваліфікаційної роботи	05.05-07.05	
7	Подання кваліфікаційної роботи керівникові	07.05-10.05	
8	Розробка обов'язкових демонстраційних матеріалів	11.05-13.05	
9	Попередній захист роботи	14.05-01.06	
10	Подання роботи в деканат	03.06	

Студент Струпінський І.Ю.

(підпис)

(прізвище та ініціали)

Керівник роботи _

Коба А.Б.

(підпис)

(прізвище та ініціали)

Реферат

Пояснювальна записка містить 63 сторінки, включає 40 рисунків, 1 таблиця та 14 посилань.

PYTHON, MONGODB, TELEGRAM, ЧАТ-БОТ, МЕСЕНДЖЕРА, PYTELEGRAMBOTAPI, BOTFATHER.

Актуальність - чат-бот збирає контактні данні, та дає змогу менеджеру не витратити час на стандартні запитання, тим самим пришвидшує свою роботу та показує гарний сервіс.

Мета роботи - підвищення якості роботи менеджера туристичної компанії за рахунок розробленого чат-бота для зібрання інформації про тур та отримання контактних даних від клієнта.

Об'єкт дослідження - процес роботи менеджера туристичних компаній.

Предмет дослідження - телеграм чат-бот для зібрання інформації та контактних даних.

У роботі досліджені чат-боти які є на даний момент часу. Встановлені вимоги до чат боту, основні функції які потрібні для чат-боту, етапи технології розробки боту та проведене тестування чат-боту.

Актуальність роботи полягає в створенні Telegram-боту, котрий буде призначений для збору інформації, за допомогою якої менеджер буде бачити всю важливу інформацію для підбору туру або відпочинку, і не буде втрачати свій час на базові і стандартні запитання. Цим самим, компанія зберігає номер клієнта для майбутньої роботи з ним та економить час, фінанси. В нашій країні така практика збору інформації для туру рідка, тому це виглядає цікаво та технологічно.

ЗМІСТ

ВСТУП.....	10
1. АНАЛІЗ І ВИДІЛЕННЯ ПЕРЕВАГИ РОЗРОБЛЕННОГО ПРОГРАМНОГО ЗАБЕСПЕЧЕННЯ	12
1.1 Загальні відомості про чат-боти.....	12
1.2 Опис предметного середовища.....	14
1.3 Огляд платформ, які підтримують чат-боти.....	17
1.3.1. Viber.....	18
1.3.2. WatsApp.....	19
1.3.3. Telegram.....	20
1.3.4. Facebook messenger.....	22
1.3.5 Twitter.....	23
1.3.6. Discord.....	25
1.4 Опис цільової галузі.....	28
1.5. Огляд аналогів у сфері туризм.....	31
1.5.1.Toursitebot.....	32
1.5.2. Eddy Travels.....	33
2.РОЗРОБКА ПРОГРАМНОГО ЗАБЕСПЕЧЕННЯ ДЛЯ БОТА.....	34
2.1. Постановка задачі для створення чат-бота.....	34
2.2. Мова програмування Python.....	36
2.3 Середовища розробки.....	40
2.4. База даних.....	45
2.5. Telegram Bot Api.....	47

2.6. Діаграма прецедентів.....	50
2.7. Діаграма станів.....	54
2.8. Діаграма діяльності.....	56
3. РЕАЛІЗАЦІЯ ТА СТВОРЕННЯ ЧАТ БОТУ.....	57
3.1. Основні задачі чат-боту.....	57
3.2. Створення чат-боту.....	62
3.3. Тестування чат бот.....	66
ВИСНОВКИ.....	72

Вступ

Зараз світ щодня швидко розвивається, технології також не стоять на місці. Через це щодня з'являються нові тенденції та ідеї, багато з яких сильні. Це впливає на ІТ-індустрію і безпосередньо впливає на наше повсякденне життя.

До недавнього часу про чат-боти які допомагають бізнесу ніхто і не знав, але зараз вони одні із лідерів на ринку як інструмент котрий підвищує ефективність, економить час та гроші компанії. Перевагою чат-боту є швидкість, тому що клієнт одразу отримує відповідь і може обробляти відразу дуже багато заявок. Тому, використання ботів на даний момент, є дуже актуальним для ведення бізнесу онлайн в наші часи.

Якщо казати про актуальність, то зараз дуже популярні месенджери і на цих платформах можна створити чат-бот. На даний момент є різні чат боти, наприклад : показують погоду, курс валют, магазин одягу та опитувальник. Чат- боти для ведення бізнесу наразі дуже популярні. В них не має як на сайтах безліч реклами та непотрібної інформації. Веб-сайти втрачають швидкість своїх продуктів, використовують більше інтернет-трафіку, а деякі банерні оголошення використовують трекери для відстеження активності користувачів, збору інформації про їхню діяльність та інтереси та показу більш релевантної реклами на основі інформації, яку вони отримують .

Метою роботи є підвищення якості роботи менеджера туристичної компанії за рахунок розробленого чат-бота для зібрання інформації про тур та отримання контактних даних від клієнта.

Об'єктом дослідження є процес роботи менеджера туристичних компаній.

Предметом дослідження є телеграм чат-бот для зібрання інформації та контактних даних.

Завдання :

1. Аналіз обраної предметної області
2. Порівняння аналогів існуючих чат-ботів
3. Визначення технічних завдань
4. Вибір середовища розробки та технологій
5. Розробка боту на платформі Telegram
6. Виконати тестування продукту

1. АНАЛІЗ І ВИДІЛЕННЯ ПЕРЕВАГИ РОЗРОБЛЕНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1 Загальні відомості про чат-боти.

Бот - це програма яка містить алгоритм штучного інтелекту, яка дозволяє йому взаємодіяти з користувачем. Боти можуть взаємодіяти незалежно і вони структуровані як справжні користувачі. Що цікаво, бот має профіль, але ним не керує реальна людина, він працює за алгоритмом.

Чат-бот - це віртуальний співрозмовник, програма, призначена для імітації поведінки людини під час спілкування з одним або кількома співрозмовниками. Більшою мірою ця тенденція зараз формується за кордоном – існує величезна кількість ботів для Facebook Messenger, Telegram, WhatsApp, Viber, які вирішують найрізноманітніші завдання – від пошуку квитків до управління невеликими командами розробників. І щоб отримати все це, користувачам навіть не потрібно виходити з пріоритетного месенджера.

На даний момент існують різні програми чат-ботів:

- Планова заміна - дозволяє виконувати певні функції без залучення людей, а робота буде виконана миттєво і бездоганно;
- Пошук та агрегація новин, аналітики, даних (Data-Driven Collaboration), даних, доступних на місці прийняття рішень - месенджери та всіх учасників, які їх потребують;
- Електронна комерція - для спонтанних покупок без тривалого пошуку, мобільна електронна комерція + візуальний пошук + чат-боти для спілкування з клієнтами;
- Перша лінія роботи з клієнтами, помічниками, консультантами, стандартні питання, телефонія;
- Просто для розваги – розваги, ігри.

Є нестандартні боти, які дуже дивують тим, що вони можуть дати людині, наприклад :

- Юрист. Безкоштовний чат-бот DoNotPay, створений студентом Стенфордського університету Джошуа Бродером у 2015 році, спочатку лише допоміг оскаржити штрафи за паркування. За час своєї діяльності DoNotPay, що працює в США та Великобританії, оскаржив лише 7,7 мільйонів фунтів стерлінгів у вигляді штрафу. Тепер функціонал бота-юриста значно розширений, він може надавати консультації в більш ніж тисячі різних напрямків. Поки що DoNotPay не може подавати в суд самостійно, але автоматично складає листи та запити та заповнює PDF-форми. Все це значно економить гроші для користувачів.
- Гід. Unexploredcity_bot — це чат-бот однойменної української екскурсійної платформи, завдяки якому ви можете дізнатися про цікаві пішохідні маршрути поблизу. Все, що вам потрібно зробити, це надіслати геолокацію, і бот покаже всі доступні маршрути, проведе вас по всіх цікавих місцях, надасть опис і покаже фото. На даний момент існує 62 маршрути та 530 місць у 14 містах.
- Синоптик. Чатбот Roncho, стилізований під kota-синоптика, що мешкає в Брукліні. Він може надсилати персональні прогнози погоди щодня у визначений час, або по запиту, розбавляючи їх жартами та мемами, завдяки цьому нагадує скоріш потішного друга, ніж бездушного робота.

1.2 Опис предметного середовища

На сьогоднішній день переважна більшість міжміського спілкування людей відбувається через Інтернет, який замінює користування мобільним зв'язком. Це пов'язано з активним розвитком Інтернет-технологій, підвищенням їх доступності, високим попитом на універсальні засоби передачі інформації різних типів і форматів.

Нішу цієї інтерактивної платформи займають соціальні мережі та месенджери. Згідно з дослідженням, у 2022 році 79% користувачів інтернету, рзвинених країн, були користувачами соціальних мереж, і очікується, що ця цифра зросте. Я вважаю що скоро взагалі не будет такого поняття як “мобільний зв’язок”, тому що технології беруть верх. Колись, ми і подумати не могли, що можна буде переписуватись і дзвонити через месенджери або соціальні мережі. Використання соціальних мереж є однією з найпопулярніших онлайн-діяльностей, з високим рівнем залучення користувачів і розширенням мобільних можливостей.

Telegram Messenger, один із представлених у нашій країні месенджерів, у 2021 році повідомив, що програма обміну повідомленнями вже має понад 500 мільйонів активних користувачів. Наведені вище показники цікаві як для великого, так і для малого бізнесу з точки зору кількості активних користувачів інтернету. Соціальні мережі та месенджери стають посередниками між клієнтами та бізнесом, інструментами для комунікації та впровадження послуг, які в свою чергу дають великої потік клієнтів.

Елементом роботи буде чат-бот - комп'ютерна програма, яка взаємодіє з користувачами в месенджері або соціальній мережі. Вперше термін «чат-бот» був використаний у 1994 році – так розробник Майкл Молдінг назвав свою електронну помічницю Юлію. Звичайно, у «Юлії» було мало шансів, тому що вона була найпершою у цьому світі. Вона аналізує текст, написаний користувачами, а потім публікує контент, який більш-менш відповідає темі. Чат-боти також були добре відомі користувачам інтернету наприкінці 1990-х і початку 2000-х років: у IRC та IM вони розповідають історії, організують вікторини та забороняють (блокують) зловмисних порушників правил чату. А зараз вже є 10000 варіантів як можна примінити бота для простих задач і для кожної сфери.

На цей час існують два типи ботів :

- Обмежені – відповідають на конкретні запити користувачів по заздалегідь розробленому скрипту, мають обмежену кількість відповідей;

- Розвиваються самі – працюють на основі штучної нейронної мережі, можуть розуміти суть бесіди та вести реалістичні розмови з покупцем, згодом навчаються та дають більш релевантні відповіді на запити.

Одна з переваг чат-бота - економія. Компанії які використовують ботів економлять тисячі доларів кожен місяць. Чат-боти не можуть повністю замінити співробітників, але вони дуже здатні автоматично відповідати на запитання клієнтів, заощаджуючи час співробітників і підвищуючи їх продуктивність, а ще менше стресу для клієнтів. Інтернет-маркетологи розглядають чат-боти як можливість додатково автоматизувати етап роботи з клієнтами, підвищити лояльність клієнтів, перекласти деякі основні завдання на «віртуальних продавців» та отримати прямі конверсії.

Зараз можна виділити, з якою метою створюються боти. Я виділив основні напрямки, такі як :

- Чат-боти з іграми, розважальним контентом;

Зараз вже є дуже багато ігор і в тому числі інтелектуальних як для дорослих, так і для розвитку діточок .

- Чат-боти персональні помічники;

За допомогою такого бота, можна не забути наприклад важливу дату.

- Чат-бот маркетингових операцій;

Можна просувати наприклад рекламу.

- Чат-бот для бізнесу;

Чат-боти інтернет магазинів.

- Чат-бот з технологією штучного інтелекту;

Чат-бот аналізу сайтів конкурента.

Найбільш активно розвиваються чат-боти для бізнесу та маркетингових операцій, приносять кошти, допомагають бізнесу та економить час. Тому ми розглянемо які бізнес задачі може взяти на себе чат бот :

- Оплата і оформлення замовлень;
- Можна спілкуватися в середині компанії;

Створити бот в якому буде інформація про компанію, коли нарада або збирання інформації про колег по роботі.

- Надання технічної підтримки клієнту;

Відповіді на питання клієнта.

- Поширення актуальної інформації;

Наприклад погоди або новин .

- Залучення клієнта в діалог;

З одного боку, чат-боти допомагають легко і швидко справлятися з щоденними завданнями, а з іншого – автоматично збирають і обробляють замовлення від клієнтів. Месенджери більше не є каналом особистого спілкування і використовуються в ділових комунікаціях для спілкування з партнерами та колегами. На цій ідеї побудована ідея чат-ботів.

1.3 Огляд платформ, які підтримують чат-боти

Зараз чат-боти підтримують такі платформи як:

- WhatsApp;
- Telegram;
- Viber;
- Facebook messenger;
- Discord;

- Twitter;

З одного боку, чат-боти допомагають легко і швидко справлятися з щоденними завданнями, а з іншого – автоматично збирають і обробляють замовлення від клієнтів. Месенджери більше не є каналом особистого спілкування і використовуються в ділових комунікаціях для спілкування з партнерами та колегами. На цій ідеї побудована ідея чат-ботів.

Зараз потрібно порівняти ці месенджери, проаналізувати який більш всього підійде для даної роботи.

1.3.1. Viber

Viber - у 2010 році випустили месенджер. У 2018 році аудиторія застосунку перевищила 1 млрд користувачів у 190 країнах і тільки у 2019 офіційно відкрилось представництво в Україні. Viber використовує наскрізне шифрування. Таким чином процеси додатку, захищені та лишаються конфіденційними. Коли користувач надсилає повідомлення з одного телефону на інший, половина ключів знаходиться в першому телефоні, а інша у другому. Кажуть що на сервері вайбер ключи не зберігаються, але на просторах інтернету, користувачі кажуть що це не так і їх данні зливають. Ми будемо розглядати ще один месенджер у якого з цим все добре. Тут можна і знімати відео і відправляти файли, голосові повідомлення і можна робити чат-боти але в них все це з'являється в останню чергу, якщо ми беремо до уваги інших конкурентів.

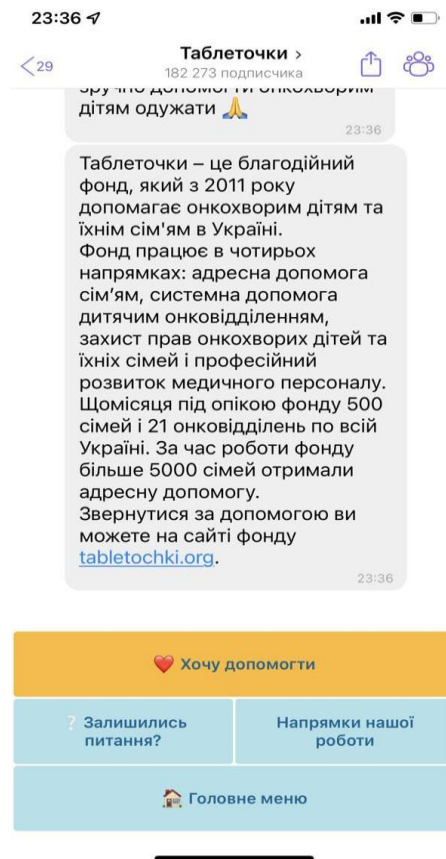


Рисунок 1.1 - Зовнішній вигляд боту Таблеточки.

Наприклад, чат-бот у Viber — таблеточки (рисунок 1.1), один із найбільших благодійних фондів в Україні. З 2011 року збирає гроші на лікування онкохворих дітей. Їхній офіційний бот – це платформа, за допомогою якої можна внести будь-яку суму на лікування неповнолітніх. Тож мінімальна плата (30 грн) піде на оплату трьох катетерів, а вартість препарату для знищення злоякісних клітин дорівнює вартості чотирьох чашок капучино плюс соєве молоко.

1.3.2. WhatsApp

WhatsApp - месенджер заснували у 2009 році американський програміст але українського походження з файного міста Фастів. Цей месенджер є найпотужнішим каналом обміну бізнес-повідомленнями. Більше ніж 1,6 млрд користувачів по всьому світу користуються цим месенджером.

Дуже великий плюс є те, що відсутній спам та реклама. WhatsApp має простий інтерфейс тому що це як аналог SMS, тільки для відправки в наявності має бути інтернет. Багато людей довіряють цьому месенджеру, тому що він безпечний та конфіденційний. І ось який є мінус - він дуже легкий у використанні але це і є проблема для чат-ботів, тому що немає великого розміру функцій. Тому, месенджер не підтримує розширені елементи інтерфейсу, такі як швидкі відповіді, кнопки і картки, які є критичними для навігації.

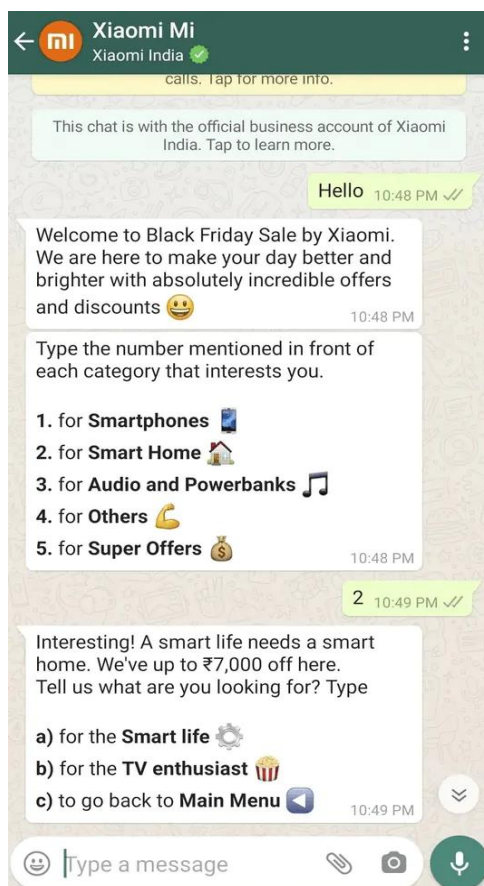


Рисунок 1.2 - Зовнішній вигляд боту Хіомі.

Наприклад, чат-бот Хіомі (Рисунок 1.2), це ще один яскравий приклад великого бренду, який проводить ефективну рекламну кампанію в WhatsApp без ризику бути заблокованим – Хіомі. Якщо ви надсилаєте масові повідомлення зі свого особистого облікового запису WhatsApp людям, з якими раніше не спілкувалися, ви можете заблокувати свій профіль. Тому, щоб зменшити ризик

засмічення: Підключіться до WhatsApp Business, отримайте підтверджений статус облікового запису компанії. Пропонується компаніям, які пройшли тест WhatsApp. У таких облікових записах біля імені, наприклад Xiaomi, з'явиться зелена галочка.

1.3.3. Telegram

Telegram – 2013 році був створений месенджер. За останній час з'являється кожного місяця майже 500 млн користувачів. Серед усіх месенджерів телеграм є найбагатшим за своїми можливостями. Вважається що це самий захищений месенджер з усіх. Це кажуть люди в інтернеті і на форуммах. Засновник не продає данні цього месенджеру. А чат-боти мають величезну кількість різноманітних елементів інтерфейсів користувача для створення цілісного досвіду спілкування для користувачів. Я можу сказати з упевненістю, що Telegram є потужним каналом обміну повідомленнями, з різними функціями, які він надає для людей які розроблять ботів. Чат бота можна оприлюднити за 3-4 кліка і конкурентів такого немає. Починають з чату усіх чатів BotFather, приблизно 5 хвилин ш можна запустити свого бота який буде готовий для клієнтів. А ще засновник Telegram Messenger Павло Дуров — влітку 2018 року запустив API Telegram Bot для розробників і магазин готових ботів для користувачів. Тому сьогодні Telegram пропонує тисячі чат-ботів на будь-який смак - від вивчення іноземних мов до спілкування з незнайомими людьми.

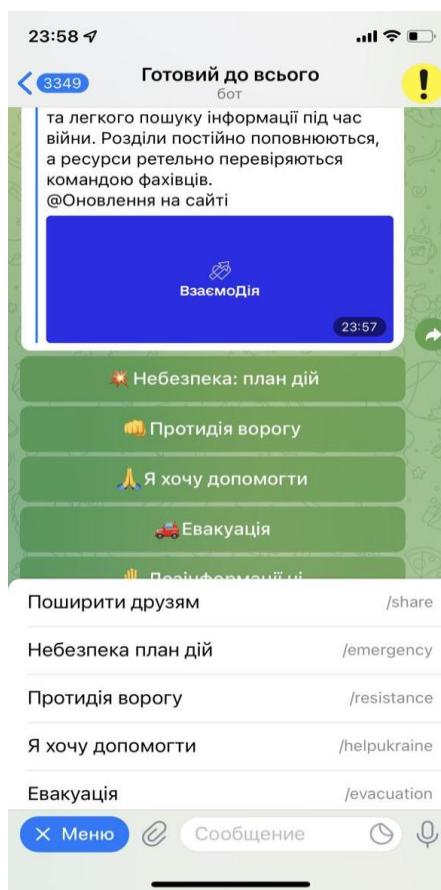


Рисунок 1.3 - Зовнішній вигляд боту Будь готовий до всього.

Будьте готові до всього - дуже корисний чат-бот(Рисунок 1.3). Він радить, як діяти під час війни: як спланувати переїзд чи убезпечити будинок з родиною; як реагувати на сповіщення та повідомлення з різних каналів зв'язку; де шукати перевірену інформацію; як убезпечити будинок та збирати тривожний портативний комплект або аптечку. Дуже крутий і корисний бот.

1.3.4. Facebook messenger

Facebook - найбільша соцмережа, вона ще почала працювати у 2004 році як мережа для студентів. Станом на грудень 2018 року Facebook дав данні що в них понад 2,3 млрд активних користувачів щомісяця. У червні 2017 року кількість українських користувачів різко зросла після заборони російських інтернет-ресурсів, серед яких Однокласники і Вконтакті. В Україні Facebook посів 3 місце в десятці найпопулярніших сайтів нашої країни. А ще у 2016 році Facebook показав щось серйозніше – власну платформу ботів. Тепер кожна компанія може

використовувати власний інструмент Facebook для створення власного чат-бота, заповнення його за допомогою Facebook Messenger і надання користувачам своїх продуктів і рекомендацій. Перевагою нової платформи перед іншими конкурентами є можливість надсилати користувачам не звичайний текст, а повну інтерактивну карту з картинками, посиланнями, кнопками тощо, дивовижну валізу, аптечку тощо.

Насправді, це сильний суперник як месенджер і в даній ситуації як для створення чат-бота. Але він насправді не такий популярний як його один із конкурентів. Є один дуже великий мінус, у Facebook не так сильно розвинений API. API - це інструмент, який дозволяє отримувати дані про дії користувачів на вашому сайті, а на основі - аналізувати статистику та оптимізувати рекламні кампанії. Тому цей месенджер не номер один у цьому списку.

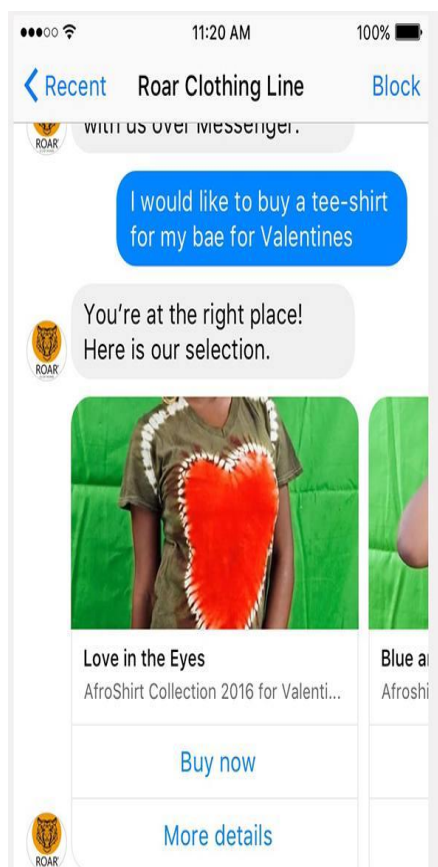


Рисунок 1.4 - Зовнішній вигляд боту Roar Clothing line.

Roar Clothing line - бізнес-бот який продає товари у великому обсязі, в основному одяг і раніше до Facebook bot магазин не мав таких оборотів але зараз обороти на місяць мають майже 100 тисяч доларів(Рисунок 1.4).

1.3.5 Twitter

Шлях Microsoft до робототехніки почався з невдач. Компанія запустила чат-бота в Twitter і запросила всіх бажаючих поспілкуватися з ним. Це розмовний чат-бот з вбудованим штучним інтелектом, який навчається з розмов з користувачами. Результати були трагічними — за один день робот на ім'я Тай перетворився з симпатичної дівчини в чоловіконенависника, шанувальника Дональда Трампа і таємного співробітника Секретної служби США. Виною всьому «недружнє» та нічим не обмежене інтернет-середовище, з якого чат-боти поглинають всю інформацію, незалежно від моральних аспектів. Ця невдача стала прецедентом, і ми побачимо її вплив на подальший розвиток «розумних» чат-ботів.

Але в Китаї прижився інший робот Microsoft — Xiaoice. Мільйони китайців знаходять час на листування з «17-річною дівчиною» і радіють. Сяобін поводить себе як дівчинка-підліток, легко розмовляє, посміхається і пам'ятає попередні розмови. Чат-боти будуть діяти як посередники між людьми, помічниками та іншими системами в рамках нової «Бесіди як платформи» від Microsoft, яка використовуватиметься в таких сферах, як інтернет-магазини.

Завдяки новим інструментам Microsoft Windows-програмісти з Редмонда та всього світу активно розвиватимуть можливості робота. Їх навчатимуть краще розуміти природну людську мову, тим самим допомагаючи людям стати більш ефективними. У форматі Twitter прямі повідомлення — це, по суті, той самий особистий чат, тому зацікавлені сторони та розробники можуть завантажувати своїх чат-ботів на платформу.

Ви також можете створювати чат-ботів за допомогою нового інструменту Microsoft Bot Framework від компанії. Роботи, створені таким чином, стануть

розумнішими пізніше (за умови використання алгоритмів машинного навчання). У розробці Facebook про таку платформу вже згадувалося вище, але Microsoft вже має принаймні одну перевагу - компанія не тільки зробить ще один чат-бот для своєї платформи: боти, створені за допомогою інструментів Microsoft, можуть бути схожими на рідні Twitter і Skype для Twitter, і ті, які не мають відношення до Microsoft Slack і Telegram.

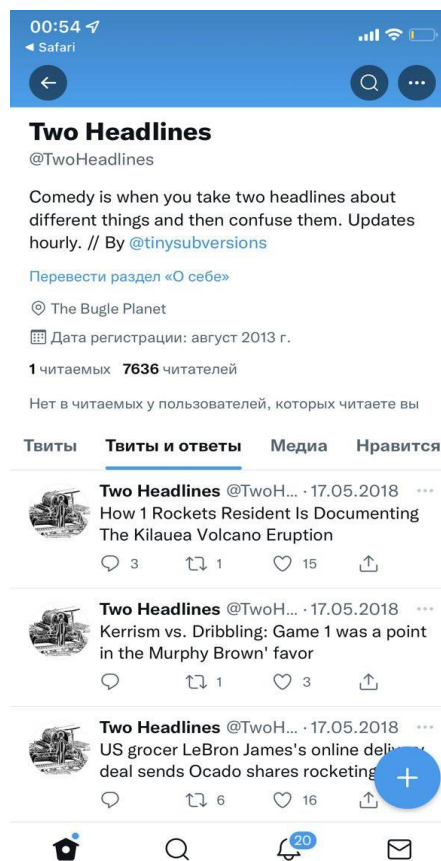


Рисунок 1.5 - Зовнішній вигляд бота Two Headlines.

Two Headlines, Автор бота спробував задати алгоритм для жартів, але для багатьох ця проста і зрозуміла ідея стала хроніками антиутопії, що розгортається в реальному часі(Рисунок 1.5). А ось що пише і сам автор бота Дариус Каземи: «Акаунти, які я створив, зараз існують окремо від мене, так що я в певному сенсі гордий за них, — каже він в інтерв'ю The Boston Globe. — Щоранку я прокидаюся, дивлюся на останні твіти TwoHeadlines і знову дивуюся». Я думаю скоро боти зможуть вийти з екрану ноутбука, та пожати мені руку.

1.3.6. Discord

За час свого існування Discord став платформою, яка об'єднує геймерів з усього світу. Як і майже у всіх популярних іграх, можна сказати, що в месенджері є суспільство і група, присвячені грі.

В екосистемі месенджерів є багато видів ботів. Персональні сервери з підтримкою Messenger можуть використовувати ботів для гри з друзями, тоді як загальнодоступні сервери можуть використовувати їх для модерації користувачів. Повнофункціональні музичні боти доступні на серверах обох типів, тому гравці можуть слухати музику в будь-який час, коли вони грають.

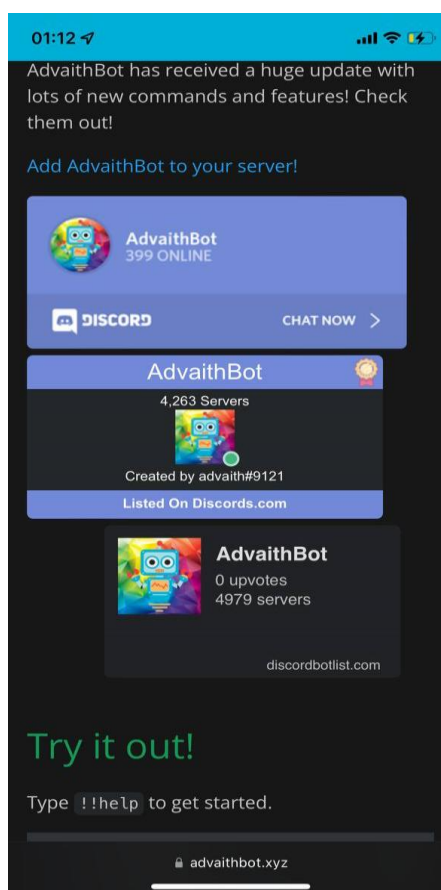


Рисунок 1.6 - Зовнішній вигляд Erisbot

Erisbot інтуїтивно зрозумілий і простий у використанні (Рисунок 1.6). Програма працює як музичний плеєр і містить багато функцій. Його навіть можна використовувати як окрему програму.

Таблиця 1.1 - Таблиця порівнянь функцій

Аналіз	WhatsA pp	Viber	Facebook	Telegra m	Discord	Twitter
відео	+	+	+	+	-	-
відеотран сляції	+	+	+	+	+	-
стікери	-	-	-	+	+	-
Текстові повідомле ння	+	+	+	+	+	+
Голосові повідомле ння	+	+	+	+	-	-
геолокаці ї	-	+	-	+	-	-
файли	+	+	+	+	-	-
GIF	-	+	-	+	+	+
API	-	+	-	+	+	+

Сильне, але вузьке співтовариство, таке як Discord, було визнано неможливим для впровадження платформи чат-бота для цільової галузі – подорожей, через визначення пріоритетів двох факторів, таких як розмір та різноманітність аудиторії. У свою чергу, незважаючи на високий статус Twitter у світі, він не набув необхідної популярності в Україні. Facebook - дуже повільний розвиток API і якщо дивитися на оточення то в нашій країні він не дуже популярний. WhatsApp – месенджер підходить для бізнесу, але він дуже простий. Він обмежений у своїх можливостях через відсутність розширеного інструментарію для створення кастомізованих ботів. Viber - насправді дуже потужний конкурент у напрямку ботів але месенджер трішки запізнюється і після телеграму, впроваджує у себе ці технології. Telegram – дуже швидко розвивається,

впроваджує один з перших нові технології, має великі можливості у створенні ботів і легко можна освоїти.

1.4 Опис цільової галузі

За темою дипломної роботи, ми будемо розбирати таку галузь як туризм.

За рейтингом Всесвітньої туристичної організації Україна посідає 8 місце у світі за кількістю туристів. Згідно з багатою археологічною та релігійною історією країни, сферою української індустрії гостинності є активний відпочинок та спортивний туризм (наприклад, лижний) та пізнавальний туризм. Численні оздоровниці, розташовані майже в усіх регіонах України, стають всесвітньо відомими. Я вважаю що одне із самих популярних місць в нашій країні це - Чорнобиль. Я був там сам на екскурсії і хочу сказати що це дивовижне місце. Там своя атмосфера, мертвий город дуже приваблює туристів. Коли я їздив на екскурсію то десь приблизно 70% людей були іноземці. Та є ще статистика що в 2019 році відвідали 120 тисяч туристів це місто. А у 2016 кажуть приблизно 60 тисяч. Ви бачите як стрімко росте популізація цього міста, пандемія трішки все зламала, але на мою думку статистика з кожнім роком будет рости у два, три рази. Туристів дуже приваблює наша Країна, тут дуже багато історії, природи, культури і так далі.

Туристична індустрія зараз швидко розвивається і навіть при нестабільному економічному становищі стрімко розвивається в нашій країні. Зараз можна розібрати що таке туристична компанія як бізнес. Турфірми є посередниками між туроператорами та клієнтами. Якщо туроператор відповідає за створення туристичного продукту та ефективної реклами в інформаційному просторі для цільової аудиторії туру, то на туристичне агентство покладаються організаційні питання, пов'язані з продажем квитків, турами та туристичним комфортом. Дуже популярний напрямок. Кажуть, якщо брати час до пандемии то за 8 років число компаній збільшилося у 3 рази. На сьогоднішній день у туризм в основному

інвестують національні та місцеві малі та середні підприємці. Держава робить це тому, що цього вимагає статус туризму як пріоритетної галузі економіки, малі та середні підприємці фактично вкладають кошти в туристичну інфраструктуру у свій бізнес. Навіть не дивлячись на кризу і пандемію, зараз дуже багато туристичних компаній. Зростає конкуренція, тому компанії придумують нові варіанти як залучити клієнтів, також за допомогою чат-ботів своїм функціоналом. На цій почві з'являються нові робочі місця, великі компанії дають рекламу нашої країни для закордонних туристів, облагороджують архітектурні спорудження.

Я тут згадав про чат-боти і можна розглянути актуальні причини створення чат ботів для туристичних компаній:

- Підтримка клієнта 24/7

Клієнт може задати питання або дізнатись певну інформацію у будь який час. В свою чергу наприклад менеджер не може відповісти в будь який час, тому що в нього є графік роботи. А ще клієнт зможе залишити контактні данні і час, коли до нього можна подзвонити щоб обговорити питання.

- Менше стресу для клієнта

Що б розуміти наскільки подобаються клієнтам боти, можна звернути увагу на опитування BusinessWire, 55% людей будуть спілкуватися через чат-бот, а ще 15% будуть спілкуватися через чат-бот якщо підвищать ефективність сервісу, скоротять час вирішення проблеми, скоротять час на переведення клієнта оператору без залучення телефону та пошти. Ти не потрапиш на погани настрої менеджера іноді людям поки вони не зацікавлені буває не зручно, треба приймати іноді відразу рішення.

- Купівля та бронювання готелю або транспорту

У чат-боті в режимі розмови ви можете вибрати маршрути, готелі та місця для бронювання. Все більше користувачів смартфонів вважають цю форму спілкування бажаною.

- Інтегровані платформи

Клієнту не треба встановлювати додаткові додатки. Зараз час коли все спілкуються в месенджерах і в даній ситуації клієнт зможе перейти до месенджеру який встановлений у нього в телефоні і почати бесіду с ботом.

- Всі відповіді залишаються у переписці;
- Месенджер доступний у всьому світі;

А зараз можна подивитися які плюси чат бот дає туристичній компанії:

- Збір даних;
- Розуміння які питання та на що звертає увагу більше всього клієнт;
- Економія ресурсів компанії;
- Робота 24/7;
- Бот може обробляти багато запитів;

В цей час менеджер може не займатись обробкою стандартних запитань, а буде задіяний для серйозних цілей.

- По переписці менеджер вже може розуміти чого хоче клієнт.

Це ще не весь список але тут основне. І я розумію що для туристичного агентства, цей бот дуже потрібен. Він зможе надати інформацію, що клієнтам більше подобається, на що треба звертати увагу. Можна зменшити штаб, тим самим менше витратити кошти. Подобатися своєму клієнту, тому що він отримав швидко відповідь на запитання. Зміг задати питання пізно ввечері і мати відповідь. Можна ефективно впроваджувати нові продукти та послуги, надавати клієнтам знижку. Чат-боти відмінно маскують збір даних під дружню бесіду, і тому є незамінним джерелом корисної інформації про біль та переваги клієнта. А найголовніше - вони використовують ці данні та підлаштовують під клієнта в режимі реального часу. Все це використовується не тільки для покращення якості сервіса, але ще для творення персоналізованішої та таргетованої реклами в майбутньому.

У цьому розділі було проведено опис цільової галузі. Було розглянуто що таке туристична компанія, які в неї плюси, що вона робить для країни і як ця сфера розвивається. Було проаналізовано що дає бот для клієнта і які плюси для туристичної компанії. Ми побачили що багато плюсів дає цей помічник проаналізували функціональні вимоги і я вважаю для того, щоб компанія ще більше розвивалась, кожній компанії потрібен такий бот.

1.5. Огляд аналогів у сфері туризм

Насправді в нашій країні дуже мало аналогів на цю тему, хоча за кодом це добре практикується а в нас це тільки з'являється. На разі для туристичної компанії в чат-ботах є ось такі сценарії :

- Бронювання готелів;

Допомагають підібрати готель, ціну та апартаменти.

- Інформаційні;

Що брати з собою, які треба документи, які правила в'їзду при пандемії.

- Бронювання квитків;

Можна купити або забронювати квиток на поїзд чи на літак.

- Збір даних

Збір даних для менеджера, щоб розуміти чого хоче клієнт і зможе надати компанія такий сервіс.

- Планування турів;

Планування наприклад турів для компанії в декілька країн та з екскурсіями.

Зараз буде наведенні декілька варіантів вже працюючих ботів для туристичних компаній.

1.5.1.Toursitebot

Українська компанія, чат-бот допомагає дізнатися всю інформацію яка потрібна для пошуку туру на сайті, компанія займається тільки закордонними відпочинками і надає не великий список компаній куди можна поїхати. Бот можна побачити на малюнку б

Плюси:

- Підтримка 24\7;
- Надає змогу відповідати на все варіантами кнопок;
- Дає змогу після надання інформації боту, відразу надати силку на сайт де під критерії вже буде підібрані готелі;
- Зібрання даних клієнтів для подільшої роботи з ними;

Мінуси:

- Мало країн для вибору клієнта;
- Не має Української мови;
- Не має турів по Україні;

Сайт іноді не коректно підбирає тури а іноді взагалі не дає варантів і не має куди подзвонити та клієнт не залишає номер для роботи менеджера (Рисунок 1.7)

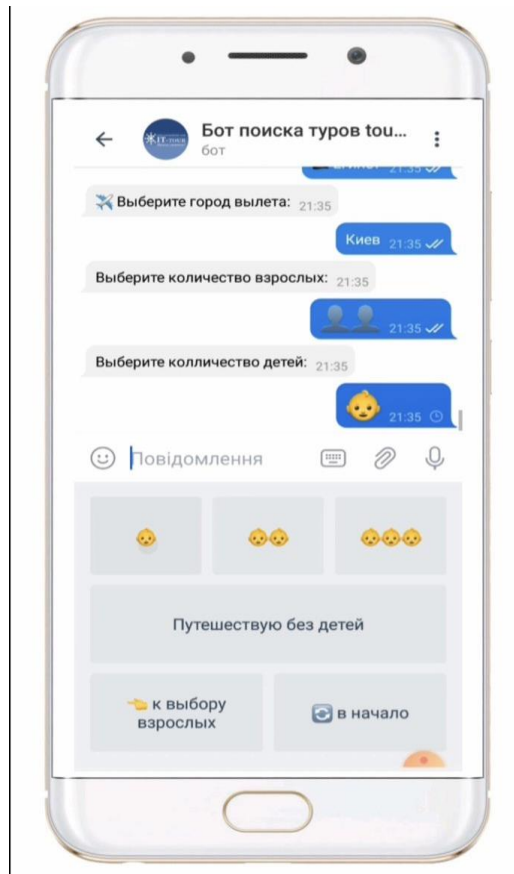


Рисунок 1.7 - Зовнішній вигляд ToursiteBot.

1.5.2. Eddy Travels

Eddy Travels — європейська компанія створила цей бот. Він розроблений для підтримки широкого спектру функцій планування подорожей. Дуже цікаво що для пошуку можна користуватися голосовими повідомленнями. Після запиту користуваччя він витягує з певних сайтів потрібну інформацію(Рисунок 1.8)

Плюси:

- Підтримка 24\7;
- Можна відповідати голосовими повідомленнями;
- Дозволяє відстежувати ціни на квитки або готелі;
- Можна підібрати для себе готель;

Мінуси:

- Не можна зв'язатися з менеджером, тому є трішки не довіра бронювання або покупки білетів;
- Є мови на вибір але все одно всі відповіді на англійській мові;
- Потрібна підписка щоб користуватися ним весь час;



Рисунок 1.8 - Зовнішній вигляд Eddy Travels

2. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ БОТА

2.1. Постановка задачі для створення чат-бота

Постановка задач буде формуватися через інтерв'ю. Інтерв'ю є невід'ємною частиною процесу розробки чат-бота. Коли ми працюємо на основі припущень або лише власного досвіду, ми, як правило, не зосереджуємось і не дуже розуміємо чи буде цей бот корисним. Це означає, що ми можемо легко втратити можливості покращити послуги або продукти для задоволення потреб клієнтів. Інтерв'ю повинно виконуватися на перших етапах роботи над продуктом, оскільки без нього проектування чат боту для туристичного агенства може базуватися лише на власному досвіді та припущеннях, які не можна вважати об'єктивними.

Під час розробки телеграм-боту я поспілкувався з власницею компанії “UNID TRAVEL” Струпінською Наталією Сергіївною, з якою ми обговорювали що має бути в телеграм-боті і для чого він взагалі потрібен. На основі аналізу і бесіди ми прийшли до висновку що бот потрібен для збору інформації, щоб менеджеру вже було відомо чого хоче клієнт. По друге збір даних, для розповсюдження реклами та цікавої інформації. Грунтуючись на цьому, ми розробили алгоритм питань які будуть задаватися клієнту :

- Яку країну хочете відвідати ?
- Тури по Україні ?
- Спокійний відпочинок (пляж, готель)?
- Активний відпочинок (сноуборт, лижі)?
- Яка країна ?
- Можливі дати поїздки ?
- Кількість ночей ?

- Рівень сервісу ?
- Кількість людей ?
- На який бюджет розраховуєте ?

В кінці можна попросити номер телефону для того щоб менеджер зв'язався с клієнтом.

За допомогою такого діалогу с ботом, менеджер буде бачити всю важливу інформацію для підбору туру або відпочинку і не буде втрачати свій час на базові і стандартні запитання. Цим самим компанія зберігає номер клієнта для майбутньої роботи з клієнтом.

Основна задача розроблюваного чат-боту полягає у чіткому та швидкому зборі інформація та отриманні контактів для подальшої роботи з клієнтом. На діаграмі можна побачити структуру клієнта з чат-ботом туристичної компанії (Рисунок 1.9)

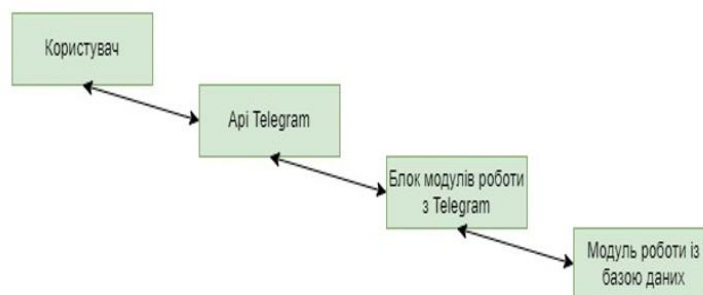


Рисунок 1.9 - Структура клієнта з чат-ботом компанії.

Після того як ми розібрали за яких умов і що має бути в боті можна перейти безпосередньо до аналізу технологій за допомогою яких можна створити чат-бот.

Мета роботи - Зібрання даних клієнта про подорож для туристичної компанії за допомогою чат боту який розроблений на мові програмування Python на базі месенджера Telegram та його бібліотек.

2.2. Мова програмування Python



Рисунок 2.0 - Логотип мови програмування Python

При створенні дипломного проекту важливою складовою є грамотно підібраний інструмент розробки. Тому, перед тим, як можна буде починати розробку нашого бота, треба спочатку проаналізувати мову програмування Python(Рисунок 2.0).

В якості мови програмування для написання програм було обрано Python. Вибрана мова ідеально підходить для написання чат-ботів за допомогою додаткова бібліотека

Python - це об'єктно-орієнтована мова програмування з інтегрованими динамічними структурами. В основному для підтримки програмного забезпечення Інтернет і розробки додатків. Це дуже привабливо в області, яка швидко розвивається, оскільки мова надає динамічні параметри введення тексту та Динамічне прив'язування функцій.

Python - відносно проста мова, вона вимагає унікального синтаксису, зосереджений на читанні. Розробники можуть читати та перекладати код Python набагато простіше, ніж інші мови. Це, у свою чергу, зменшує витрати на підтримку

та розвиток програми, оскільки дозволяє командам працювати без серйозних бар'єрів з точки зору мови та досвіду.

Ця мова програмування використовує мінімалістичний синтаксис, який збільшує швидкість запису програми, зменшуючи обсяг написаного коду та покращуючи його читабельність. В порівнянні з іншими мовами програмування це можна побачити на малюнку(Рисунок 2.1)

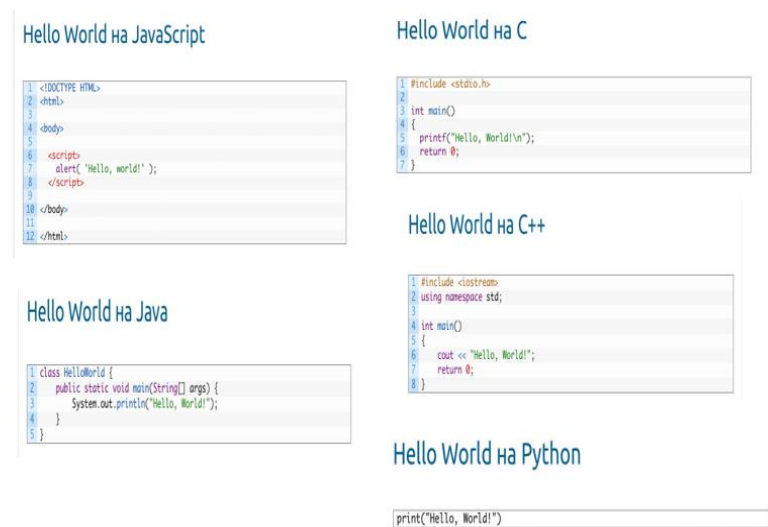


Рисунок 2.1 - Синтаксис різних мов програмування.

Крім того, Python підтримує використання модулів і пакетів, що означає, що програми можна розробляти модульно, а код можна повторно використовувати в різних проектах. Після того, як ви розробили правильні модулі або пакети, ви можете розширити їх для використання в інших проектах і легко імпортувати або експортувати ці модулі.

Мова Python є широко використовувана у великих компаніях тому, що вона підтримує різні пірадигми програмування: об'єктно орієнтовану, функціональну, структурну, інмеративну та аспектно орієнтовану. Python має велику стандартну бібліотеку яка є комплексна та має автоматизоване керування пам'яттю та динамічні функції. Щоб розуміти наскільки ця мова популярна, можна подивитися на NASA які літають у космос, вони також використовують цю мову. Переважна

більшість веб-додатків і платформ покладаються на Python, включаючи пошукову систему Google, YouTube і Facebook. Це ясно показує, що ця мова має достатньо серйозна та надійна. Майже 15% розробників використовують Python. Він доступний в таких операційних системах, як UNIX, Linux, Windows і Mac OS.

Іншою важливою перевагою мови Python є велика кількість модулів, які можна імпортувати для надання безлічі різних додаткових функцій. Ці модулі написані на C, сам Python може використовувати більше Досвідченні програмісти.

Бібліотеки в програмуванні - Набір класів, компонентів або модулів, що використовуються для вирішення різноманітних проблем. Завдяки їх використанню не потрібно щоразу вказувати стандартні елементи, такі як кнопки, запити тощо. Бібліотека — це набір коду, який був написаний кимось. Програмісти можуть включати ці бібліотеки в свої програми, вставляючи їх у свій програмний код відповідно до спеціальних алгоритмів, і ці бібліотеки можна використовувати повторно. Бібліотеки бувають вбудованими і додатковими. Кожна мова програмування має свою бібліотеку. Бібліотеки які використовують при розробці ботів та сайтів або чогось ще, є такі як наприклад у Java використовується JHipster та TelegramVots у C++ - Boost та Tgbot-cpp, Python - Simplejson або Telebot, тощо. Використання бібліотеки є невід'ємною частиною Функціональне програмування. Використання бібліотек скорочує час на написання програм і зменшує кількість помилок при написанні програм.

Якщо брати Python та його бібліотеки для розробки телеграмм бота то ми побачимо що їх не мало, але кожна бібліотека розрахована під різні задачі та проекти. Наприклад Telebot це бібліотке для невеликих проектів, а python-telegram-bot - це потужна бібліотека, яка активно розвивається і дає змогу робити великі проекти з великою кількістю функцій. Для професіоналів підійде aiogram- це дуже потужна бібліотека. З наведених вище варіантів була обрана бібліотека telebot: вона повністю реалізує функціонал бота, будучи максимально наближеною до API Telegram Bot і розвиваючись. Тому мовою розробки проекту є Python, і для моїх

цілей розробки телеграм бота для туристичного агенства ця бібліотека дуже підійде.

Єдиним недоліком, про який говорять самі розробники Python, є те, що програми Python повільні порівняно з іншими програмами. Але це не грає велику роль якщо порівнювати з перевагами мови при кодуванні програм не дуже критичних до швидкості виконання.

2.3 Середовища розробки

PyCharm необхідний для створення високоякісних програм з використанням мови програмування Python. Систему також можна використовувати для редагування сценаріїв Javascript і Coffeescript, а також файлів розмітки та стилів веб-документів.

Для створення якісного робочого коду програмісти, які використовують PyCharm, будуть використовувати класи та мовні методи, дуже зручні для пошуку та навігації в системі. Це прискорює написання коду. Код можна імпортувати з інших джерел, оскільки програмісти надають доповнення та шаблони програмування. Також перевіряє сумісність версії інтерпретатора. Автоматичне форматування та підсвічування коду допомагає структурно та логічно представити код, полегшує сприйняття коду, аналіз і виявлення синтаксичних помилок.

На малюнку можна побачити інтерфейс, який дуже зручний оскільки має багато функцій(Рисунок 2.2). Приємний для очей та його ще можна налаштувати під себе.

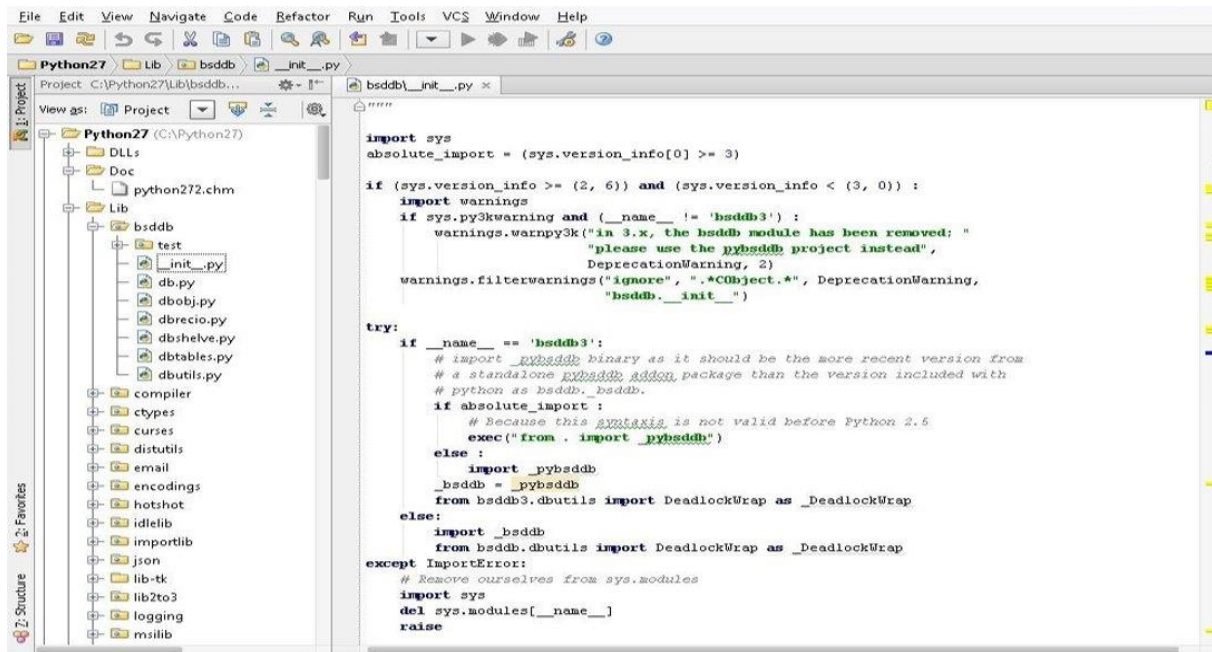


Рисунок 2.2 - Інтерфейс Pycharm.

Інтегрований графічний налагоджувач для налагодження та тестування. Можна працювати з командного рядка. Профайлер Python допоможе ідентифікувати та оптимізувати нерозпізнані фрагменти коду, виконання яких займає занадто багато часу. Доступні дистанційні перекладачі. Інтегровані інструменти для створення та налаштування віртуальних середовищ Vagrant та контейнерних середовищ Docker.

PyCharm підтримує модульне тестування, яке перевіряє правильність роботи різних частин коду, модулів і програм. Це часто прискорює тривалий і складний процес пошуку помилок програмування в тексті програмного коду, який може складатися з сотень і навіть тисяч рядків.

Під час створення доступів до бази даних PyCharm надає зручні інструменти для створення запитів SQL до Oracle, SQL Server, PostgreSQL, MySQL та деяких інших баз даних. Ви можете писати та редагувати код SQL та використовувати його для перегляду даних запиту.

Важливо, що PyCharm підтримує багато популярних фреймворків. Так, інтеграція з фреймворками Django та Flask дуже корисна для програмістів Python. Вважайте, що кожна з цих рамок має свою сферу застосування. Django часто

використовується для розробки багатокористувацьких програм, і він забезпечує відносно високий рівень безпеки. Flask корисний, наприклад, для роботи з базами даних і генерування складних запитів SQL. Розробники веб-додатків також можуть використовувати фреймворки Pyramid або web2py. Рефакторинг за допомогою PyCharm дозволяє швидко і безпечно оновлювати код і вносити зміни в проект.

На мові програмування Python розроблені якісні бібліотеки для проведення наукових розрахунків і PyCharm надає можливості для їх використання у нових програмах. Наприклад, набір бібліотек Anaconda допоможе написати програму для обробки великих масивів даних і машинного навчання. Бібліотека з відкритим вихідним кодом NumPy призначена для роботи з багатовимірними масивами, в тому числі і з використанням високорівневих математичних функцій. Візуалізувати дані за допомогою статичних, анімованих та інтерактивних графіків допоможе бібліотека Matplotlib.

Середовище розробки PyCharm надає багато зручностей, але його можна зробити ще зручнішим. Наприклад, ви можете змінити інтерфейс, використовувати приємні колірні схеми та підсвічування синтаксису, налаштувати гарячі клавіші та створити власну унікальну робочу область. Розробляйте плагіни для підтримки зовнішніх систем контролю версій, відстеження помилок та оновлень, інтеграцію з фреймворками та різними інструментами. Бібліотека плагінів постійно оновлюється і значно розширює функціональність PyCharm.

PyCharm розробляється та підтримується JetBrains. Існує дві основні версії цього програмного продукту. PyCharm Community Edition поширюється на умовах ліцензії Apache, PyCharm Professional Edition доступна для комерційного використання. Ви можете використовувати PyCharm Education Edition для вивчення програмування на основі Python.

Створення проекту в PyCharm:

Ознайомившись із середовищем програмування PyCharm, почнемо зі створення проекту. Для цього запустіть PyCharm. Якщо відкритих проектів немає, з'явиться

вітальне вікно (рисунок 2.3) з можливістю приєднатися до ISR. Нас цікавлять дві з них: створити новий проект, відкрити.



Рисунок 2.3 - Вітальне вікно.

Програма, яка розробляється в PyCharm, складається з файлів, кожний із яких має певне призначення.

Таким чином, проект — це набір файлів, створених користувачем або автоматично необхідних для запуску програми, розробленої в ISR. Виберіть опцію Створити новий проект та введіть назву проекту у діалоговому вікні, що відкриється (Рисунок 2.3). Буде створена папка проекту, що містить папки та файли, необхідні для подальшої розробки. Це відкриє головне вікно середовища розробки.

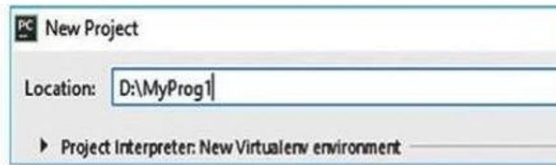


Рисунок 2.3 - Введення назви проекту.

Інтерфейс середовища розробки PyCharm Розглянемо елементи головного вікна PyCharm (рисунок 2.4).

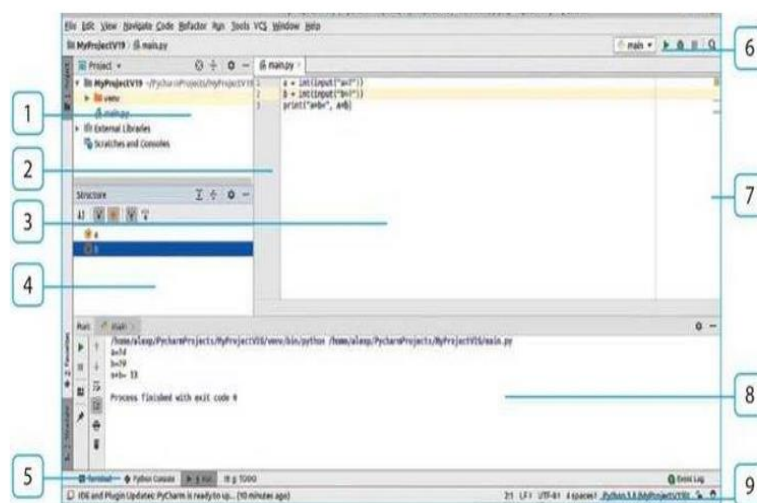


Рисунок 2.4 - Інтерфейс середовища Pycharm.

1 - панель проекту, що показує файли проекту;

2 - Ліва смуга - вертикальна смуга з лівого боку редактора, що показує номери рядків і точки розриву коду;

3 - Вікно текстового редактора, де написаний код і вкладки для зручної навігації між відкритими файлами;

4 - Панель структури (Structure), яка містить список змінних і функцій, створених у програмному коді;

5 - Панель інструментів вкладок і кнопка для перемикання між панелями. Якщо навести курсор на цю кнопку, з'явиться список доступних панелей;

6 - Навігаційні ваги з кнопками для запуску та запуску програми;

7 - Правий жолоб - вертикальна смуга з правого боку редактора, що показує результати перевірки програмного коду;

8 - вихідна панель для перегляду та введення повідомлень під час виконання програми;

9 - Лінійка стану (рядок стану), яка показує деякі параметри ISR.

Отже, ми можемо побачити що дане середовище дуже підходить для нашого проекту та побачили як його запускати і що за що відповідає.

2.4. База даних

Одним з найважливіших кроків у розробці дипломної роботи є запис і зберігання контактної інформації потенційних клієнтів та їх відповіді на запитання для подальшого спілкування з ними. Тому наступним кроком у розробці бота Telegram є вибір бази даних.

MongoDB — це міжплатформний та документоорієнтований відкритий документ, база даних NoSQL. Як база даних NoSQL, MongoDB уникає структури робочого столу реляційних баз даних на користь JSON-подібних документів з динамічними схемами, які називаються BSON.

Це робить інтеграцію даних швидшою та простішою для певних типів програм. MongoDB розроблено для масштабованості, високої доступності та

продуктивності від розгортання одного сервера до великих складних інфраструктур на багатьох сайтах.

Спочатку MongoDB була розроблена компанією MongoDB Inc., тоді відомою як 10gen, у жовтні 2007 року як основна частина PaaS (платформа як послуга), що пропонує схожі на Windows Azure та Google App Engine. Розробка перейшла на відкритий код у 2009 році.

MongoDB став однією з найпопулярніших баз даних NoSQL, використовуючи її як основу для багатьох основних веб-сайтів, включаючи eBay, Craigslist, SourceForge та The New York Times. MongoDB доступний під загальною публічною ліцензією GNU Affero, тоді як його мовні драйвери доступні під ліцензією Apache. Також пропонуються комерційні ліцензії.

Ось які є Особливості MongoDB:

- Спеціальні запити – підтримуються пошуки полів, пошук регулярних виразів і діапазони запитів.
- Індексування – будь-яке поле в документі BSON можна індексувати.
- Балансування навантаження — це метод, який дозволяє MongoDB масштабуватися по горизонталі, розподіляючи та поділяючи дані на діапазони, а потім зберігаючи їх у різних сегментах, які можуть бути на різних серверах. Шардування використовується для визначення способу розподілу даних.
- Реплікація – забезпечує високу доступність за допомогою наборів реплік, що складаються з двох або більше копій вихідних даних.
- Агрегація – MapReduce можна використовувати для забезпечення пакетної обробки даних, а також для виконання операцій агрегації.
- Сховище файлів – MongoDB можна використовувати як файлову систему з використанням перерахованих вище функцій, і завдяки своїй чіткості вона може працювати розподіленим способом.

MongoDB реалізує асинхронну реплікацію в конфігурації master-slave на основі передачі журналів змін з головного. Підтримує автоматичне відновлення збоїв вузла хоста. Сервери, на яких виконується процес `mongod`, утворюють кворум для автоматичного виявлення нових первинних вузлів. Тому, якщо не використовується спеціальний процес кворуму (процес `mongod`, який бере участь лише в установці кворуму, але не зберігає жодних даних), кількість запущених реплік має бути непарним числом. MongoDB — це документно-орієнтована база даних NoSQL для зберігання великих обсягів даних. MongoDB використовує не таблиці та рядки, як традиційні реляційні бази даних, а колекції та документи. Документ складається з пар ключ-значення, які є основною одиницею даних у MongoDB. Колекції містять набори документів і функцій, які еквівалентні таблицям реляційної бази даних.

2.5. Telegram Bot Api

Bot API – це HTTP-інтерфейс для використання ботів у Telegram. Документація API Telegram Bot надає два способи отримання оновлень, які відрізняються один від одного:

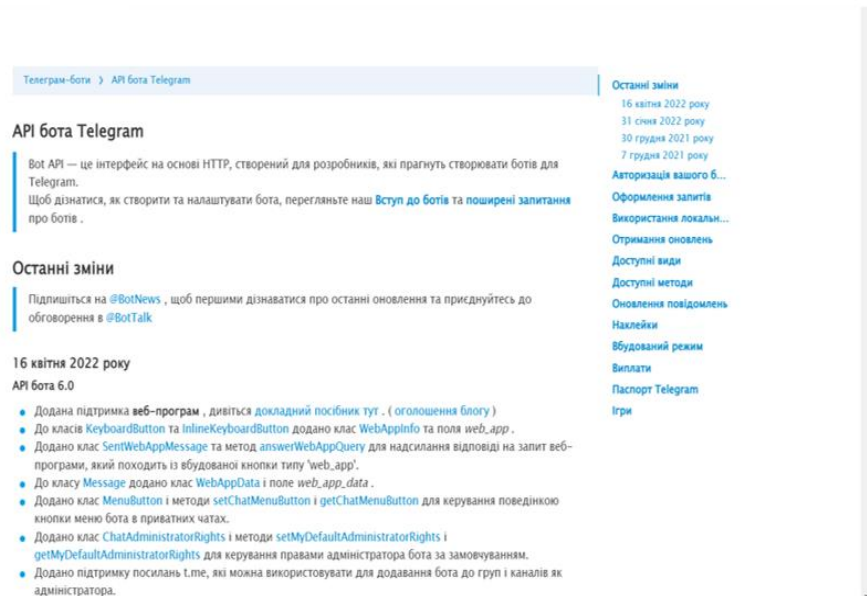
- Запити, що відбуваються періодично;
- Встановлення веб-хуків.

Вхідні оновлення зберігатимуться, поки сервер не обробить їх, але не більше 24 годин. У відповідь ми отримали об'єкт Update.

Цей об'єкт є вхідним оновленням. Оновлення – це дії, що виконуються над роботом, наприклад отримання Повідомлення від користувачів. У кожному оновленні може відобразитися лише один необов'язковий параметр.

Telegram API – це відкрита бібліотека для створення власних ботів Telegram. Він містить документацію для більш ніж 22 мов програмування, які є загальнодоступними. Він містить інструкції щодо створення та взаємодії з новим ботом через BotFather.

Є великий сайт з усією документацією, там можна знати все що потрібно для створення самого бота, на будь-якій мові програмування, на рисунок(Рисунок 2.5) це можна побачити.



Малюнок 2.5 - Telegram API.

Перший і один з найпростіших способів — періодично опитувати сервери Telegram на наявність нової інформації. З'єднання відкривається на короткий час і всі оновлення негайно надсилаються боту. Все це здійснюється за допомогою з'єднань Long Pooling. Цей спосіб простий, але не дуже надійний.

Веб-хуки працюють зовсім інакше. Якщо повідомлення потрапить в чат, Telegram сам розповість про нього. Це робота вебхуків. Тому немає необхідності періодично опитувати сервер, і причина помилок пошукової системи зникає. Однак для цього Вам потрібно заплатити за повноцінний веб-сервер на пристрої, на якому плануєте запуснути пошуковий бот.

API Telegram Bot надає кілька доступних методів:

- `sendMessage` - метод, який використовується для відправки повідомлень в текстовому виді;

- `getWebhookInfo` - метод для збору інформації про веб-хуки;
- `setWebhook` - метод, необхідний для визначення URL-адреси веб-хуку, на який бот надсилатиме оновлення;
- `getUpdates` – метод використовується для отримання оновлення за допомогою Long Pooling;

Метод `getUpdates` використовується для отримання оновлень із довгим опитуванням. Відповідь повертається у вигляді масиву об'єктів `Update`.

На малюнку цц можна побачити загальну схему роботи чат-бота месенджера Telegram (Рисунок 2.6).

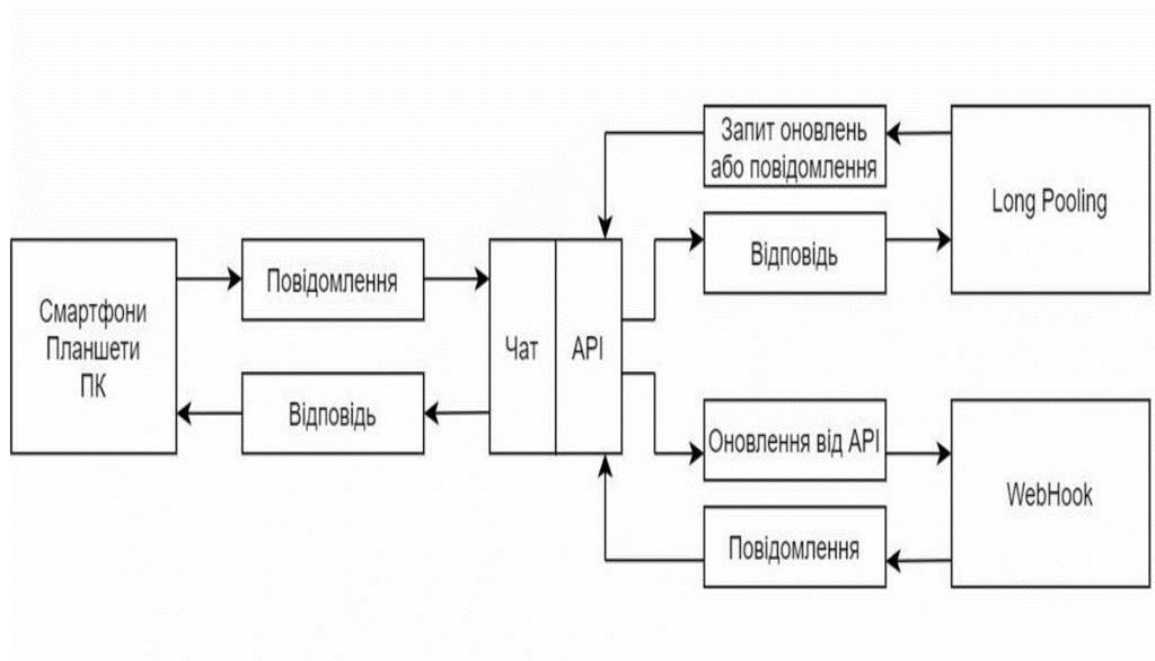


Рисунок 2.6 - Схема роботи чат-боту.

2.6. Діаграма прецедентів

Граф прецедентів — це граф, що складається з багатьох акторів, прецедентів (випадків використання), обмежених межами системи (прямокутниками), асоціацій між акторами та прецедентами, зв'язків між прецедентами та зв'язків узагальнення

між акторами. На діаграмі прецедентів показано елементи, які використовують модель.

Суть діаграми прецедентів полягає в тому, що розроблена система представлена у вигляді колекцій або акторів, які взаємодіють із системою за допомогою так званих випадків використання. Варіанти використання використовуються для опису послуг, які система надає акторам. Іншими словами, кожен варіант використання визначає набір дій, які система виконує під час розмови з актором. При цьому про те, як буде реалізована взаємодія учасників із системою, не йдеться.

Ключом до прецеденту є ціль користувача: прецедент — це набір сценаріїв, що складається з певної загальної мети користувача. Сценарій — це ряд кроків, які описують відносини між користувачем і системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

- Асоціації;
- Включення;
- Розширення;
- Узагальнення;

Тому загальні властивості використання варіантів можна представити трьома різними способами, а саме — через відношення стримування, розширення та узагальнення.

Асоціації є одним із фундаментальних понять в UML і використовуються певною мірою для побудови всіх графічних моделей системи у вигляді канонічних діаграм.

У UML міститься зв'язок між базовим варіантом використання та його окремими варіантами використання. У цьому контексті залежність — це зв'язок

між двома елементами моделі, де зміна одного елемента (незалежного) викликає зміну іншого (залежного).

Відношення розширення визначає зв'язок базового варіанту використання з іншим варіантом використання, і функціональна поведінка базового варіанту використання не завжди задіяна, а лише з додатковими умовами.

UML — це аббревіатура від Unified Modeling Language. Коротше кажучи, UML — це сучасний підхід до програмного забезпечення моделювання та документування. Насправді це один з найпопулярніших методів моделювання бізнес-процесів.

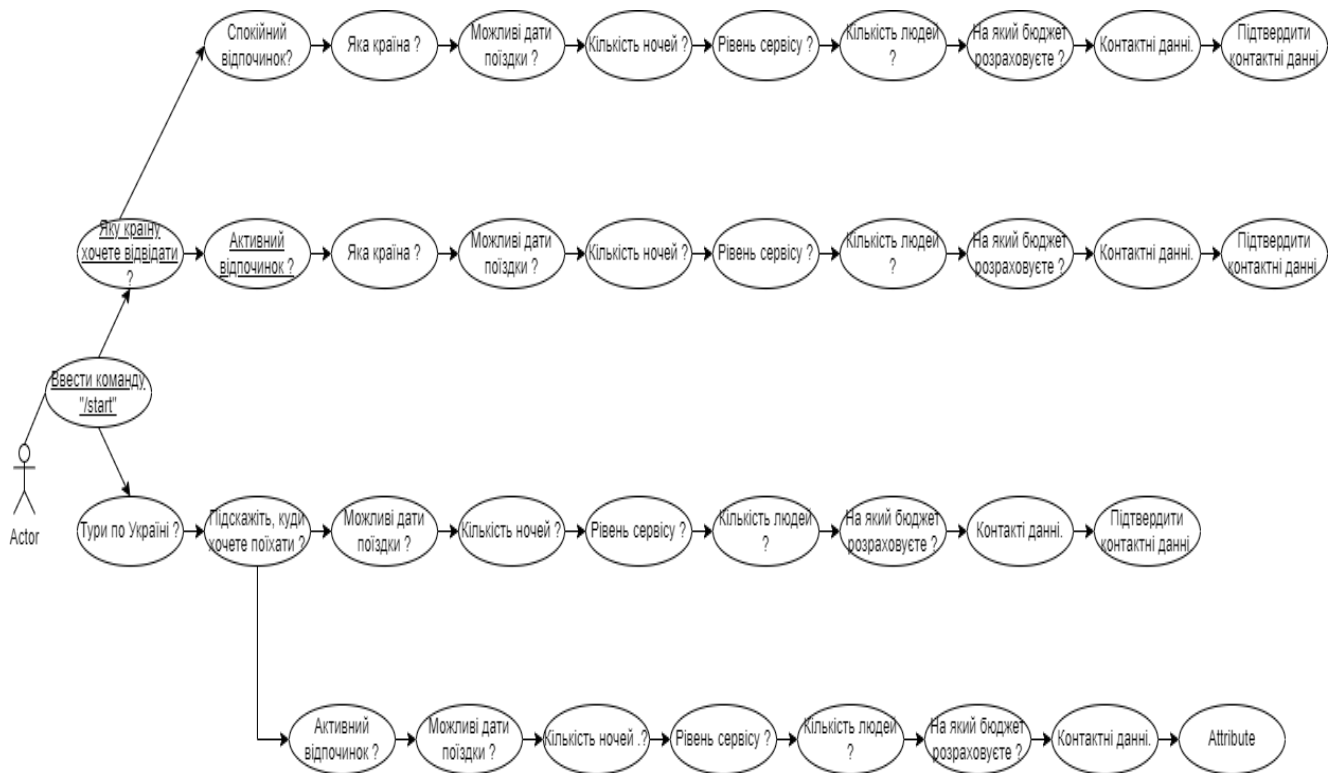
Він заснований на принципових схемах компонентів програмного забезпечення. Використовуючи візуальні уявлення, ми можемо краще зрозуміти можливі недоліки або помилки в програмному забезпеченні або бізнес-процесах. UML був створений із плутанини навколо розробки програмного забезпечення та документації. У 1990-х роках існувало кілька різних способів представлення та документування програмних систем. Потрібен був більш уніфікований спосіб візуалізації цих систем, тому між 1994-1996 роками UML був розроблений трьома інженерами-програмістами, які працювали в Rational Software. Пізніше він був прийнятий як стандарт у 1997 році і залишається ним донині.

Використовуйте діаграми, щоб описати функціональність високого рівня та область застосування системи. Ці діаграми також визначають взаємодії між системою та її учасниками. Використання та дійові особи на діаграмі використання описують, що робить система і як учасники її використовують, а не як система працює всередині(Малюнок 2.7).

Використовуйте діаграми, щоб проілюструвати та визначити контекст і вимоги всієї системи або значущих частин системи. Ви можете моделювати складні системи за допомогою однієї діаграми використання, або ви можете створити кілька діаграм використання для моделювання компонентів системи.

- Актори є користувачами.

- Прецедент - випадок використання, дія, зображений еліпсом.
- Обмеження системи – охоплює всі види використання в системі. Представлений прямокутником.



Малюнок 2.7 - Діаграма прецедентів.

У розробці програмного забезпечення діаграма класів у уніфікованій мові моделювання — це статична структурна діаграма, яка описує структуру системи, показуючи зв'язки між системними класами, їхніми властивостями, операціями та об'єктами.

Діаграми класів є основним будівельним блоком об'єктно-орієнтованого моделювання. Використовується як для загального концептуального моделювання програмних структур, так і для детального моделювання, перетворення моделей у програмний код. Діаграми класів також можна використовувати для моделювання даних. Класи на діаграмі класів представляють основні елементи, взаємодії в програмі та класи, які будуть запрограмовані.

Для подальшого опису поведінки системи ці діаграми класів можуть бути доповнені діаграмами станів або автоматами станів UML.

Діаграми класів є основним будівельним блоком будь-якого об'єктно-орієнтованого рішення. Він показує класи в системі, властивості та операції кожного класу, а також відносини між кожним класом.

У більшості інструментів моделювання цей клас складається з трьох частин. Ім'я знаходиться вгорі, властивість – посередині, а дія чи метод – унизу. У великих системах з багатьма пов'язаними класами класи групуються разом, щоб створити діаграми класів. Різні відносини між класами зображуються різними типами стрілок. Нижче наведено зображення діаграми класів. Натисніть на посилання нижче, щоб отримати більше прикладів діаграм класу або розпочніть роботу з нашим шаблоном діаграми класу вже сьогодні.

При розробці системи кілька класів ідентифікуються та групуються разом у діаграмі класів, що допомагає визначити статичні зв'язки між ними. При детальнішому моделюванні ці класи концептуального проектування дуже часто їх поділяють на як кажуть під класи.

Class - Абстрактний опис багатьох однорідних об'єктів, які мають такі ж властивості, операції та зв'язки, що й об'єкти інших класів. Графічно клас у нотації UML представляється у вигляді прямокутника, який також може бути розбиті на відрізки або відрізки горизонтальними лініями. У цих розділах можна вказати ім'я класу, властивості та операції класу.

На початкових етапах розробки діаграми один клас може бути представлений простим прямокутником, який має вказувати назву класу. При роботі з окремими компонентами діаграми опис класу доповнюється властивостями та операціями. Використовуючи методи, класи взаємодіють один з одним. сума всіх Методи класу становлять інтерфейс системи.

2.7. Діаграма станів

Діаграма стану - це особливий тип діаграми, що представляє деякий автомат. Вершинами графа є можливі стани автомата, які зображуються відповідними графічними символами, а дуги — його переходи із стану в стан. Діаграми станів можуть бути вкладені для більш детального представлення окремих елементів моделі.

Діаграми станів застосовуються для того, щоб пояснити, яким чином працюють складні об'єкти. Діаграма станів показує, як об'єкт переходить з одного стану в інший. Очевидно, що діаграми станів служать для моделювання динамічних аспектів системи.

Діаграма станів корисна при моделюванні життєвого циклу об'єкта. Від інших діаграм діаграма станів відрізняється тим, що описує процес зміни станів тільки одного примірника певного класу - одного об'єкта, причому об'єкта реактивного, тобто об'єкта, поведінка якого характеризується його реакцією на зовнішні події. Поняття життєвого циклу вдаються якраз до реактивних об'єктів, даний стан яких обумовлено їх минулим станом. Але діаграми станів важливі не тільки для опису динаміки окремого об'єкта. Вони можуть використовуватися для конструювання виконуваних систем шляхом прямого і зворотного проектування.

Діаграми станів найчастіше використовуються для опису поведінки окремого об'єкта, але також можуть використовуватися для визначення функціональності інших компонентів моделі, таких як варіанти використання, дійові особи, підсистеми, операції та методи.

На Рисунку 2.8 можна побачити діаграму станів.



Рисунок 2.8 - Діаграма станів.

Діаграму стану системи можна описати так. Співробітник магазину чекає на клієнтів. Якщо замовник подає заявку, працівник проконсультує його, а потім оформить замовлення. Після отримання працівником запчастини, замовник зобов'язаний оплатити при оформленні. Потім працівник передає замовнику запчастину.

2.8. Діаграма діяльності

Діаграма діяльності - Візуальне представлення діаграми діяльності. Діаграма діяльності — це діаграма стану кінцевого автомата, вершинами якого є певні дії, а переходи відбуваються після завершення дій.

Дії є основними одиницями, які визначають поведінку в специфікації. Дії беруть набір вхідних сигналів і перетворюють їх у набір вихідних сигналів. Будь-яка або обидві з цих колекцій можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Аналогічно, виконання діяльності – це виконання окремої діяльності, буквально, включаючи виконання тих дій, які містяться в цій діяльності. Кожну дію в діяльності можна виконати один, два або кілька разів протягом дії. Принаймні, дії мають отримати, трансформувати та перевірити дані, а деякі дії можуть потребувати певної послідовності. Специфікація діяльності (на вищому рівні сумісності) може дозволити виконувати декілька (логічних) потоків, а також

існують механізми синхронізації, які забезпечують виконання операцій у правильному порядку.

Принципову схему системи можна описати так. Менеджер магазину консультує покупців. Якщо клієнт здійснить покупку туру, він буде знайдений в базі даних або внесений в базу як новий клієнт. Після цього працівник розмістить замовлення та перевірить наявність туру. Тож за наявності туру клієнти оплачуватимуть і отримають свій відпочинок. Якщо на даний момент турів немає, замовник повинен оплатити передоплату, після чого менеджер замовить її у постачальника. Після того, як менеджер отримає угоду на тур, розрахуйтеся із замовником і віддайте йому тур.

3. РЕАЛІЗАЦІЯ ТА СТВОРЕННЯ ЧАТ-БОТА

3.1. Основні задачі чат-боту

Після інтерв'ю с власницею компанії, було проаналізовано сферу та взагалі для чого потрібен бот. Основні задачами боту, що розробляються, мають бути такими:

- Вступне повідомлення-привітання до користувача, що коротко описує бота та його функції;
- Повідомлення про те, що нам потрібно дізнатися від клієнта інформацію що до туру;
- Запитання для збору інформації;
- База даних для менеджера, котрий зможе подивитися всі відповіді клієнта;
- Зібрати контактні данні клієнта.
- Почати спочатку якщо десь помилився

Саме це і потрібно для компанії, при аналізі сфери директор компанії розповіла мені, що саме така структура допоможе. Менеджер с самого початку діалога з клієнтом буде розуміти що клієнту потрібно, та зможе показати високий та швидкий сервіс, По-друге менеджер не буде витратити час на стандартні питання, та ще відповідь на них можна почути не одразу, а це робить процес повільним. По-третє це економить кошти компанії.

3.2. Створення чат-боту

Для початку потрібно надіслати повідомлення боту @BotFather і запустити його.

BothFather — це спеціальна служба для реєстрації, налаштування та керування створеними вами чат-ботами.

Щоб знайти цього бота серед усіх інших ботів, нам потрібно ввести пошуковий запит @botfather у рядок пошуку та почати з ним розмову(рисунок 3.1)

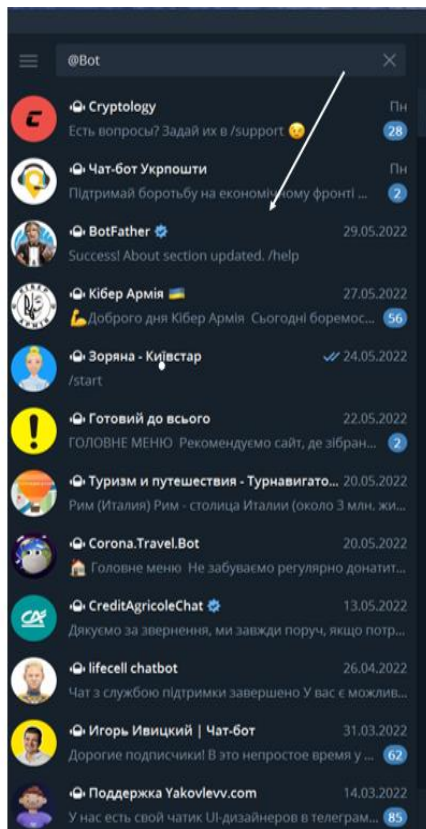


Рисунок 3.1 - пошук @BotFather

Далі клікаємо на іконку бота, після чого у нас з ним відкривається чат(Рисунок 3.2).

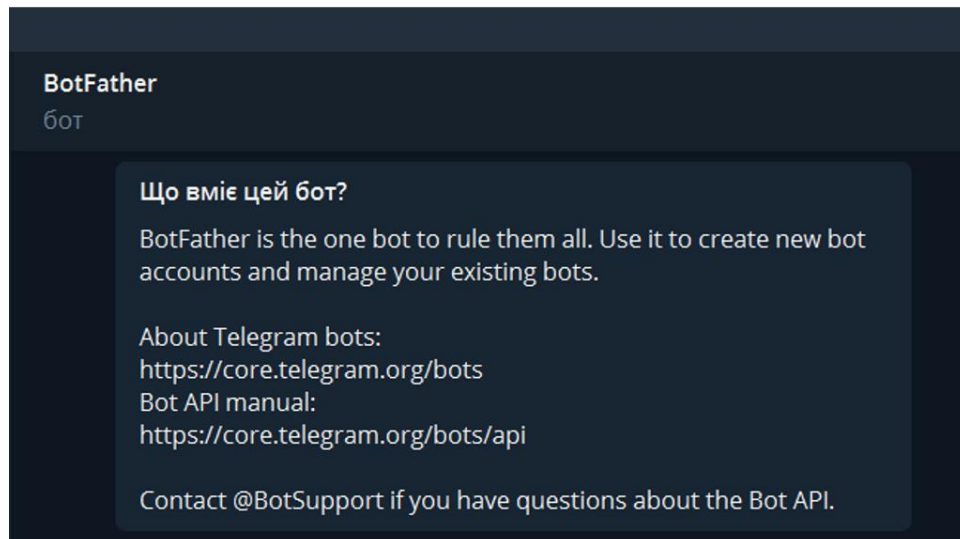


Рисунок 3.3 - чат с BotFather

Після, натискаємо кнопку “/start” розпочинаємо створення боту(рисунок 3.3). Після кнопки старт бот нам у відповідь надсилає свої команди списком, за допомогою цього списку ми зможемо керувати ним(рисунок 3.4)

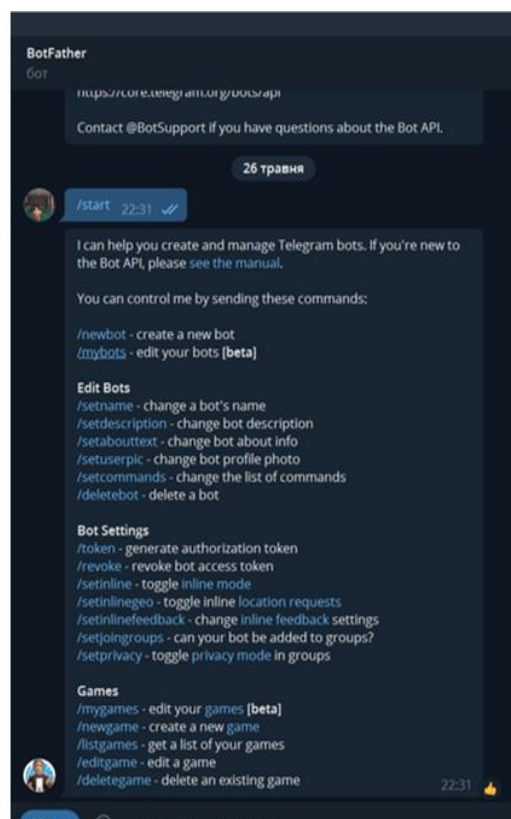


Рисунок 3.4 - Список команд бота

Ці команди дуже важливі для створення боту, вони ідуть як основні, і ось що вони означають :

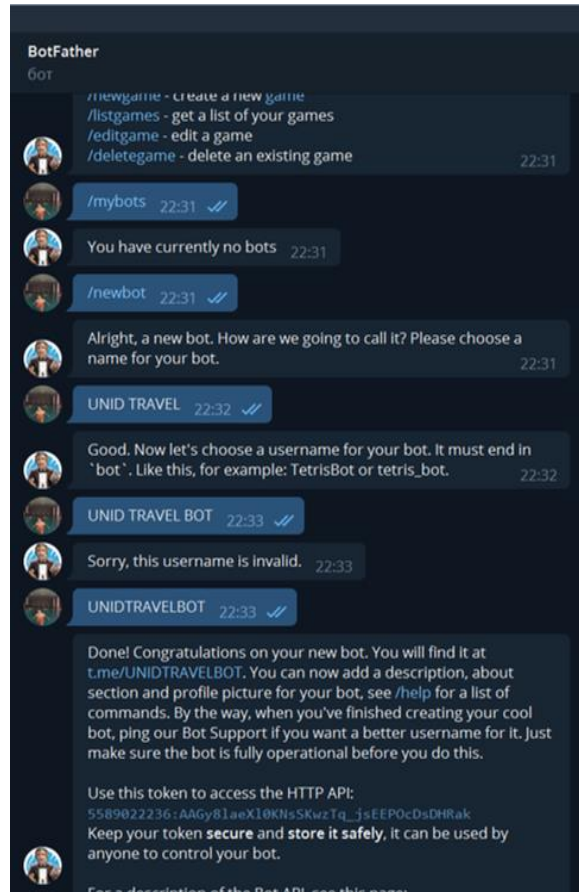
- /setname - за допомогою цієї команди можна змінити ім'я;
- /setdescription - створює текст, який буде відображатись при першому відкритті чат-бота;
- /setabouttext - додає текст в розділі про чат-боти;
- /setuserpic - рисунок для бота;
- /setcomands - можна створити список команд;
- /deletebot - можна видалити нашого бота.

Також є ще пакет функцій, він зосереджений на можливостях токена. Токен – це рядок із символів, який потрібний для того, щоб встановити справжність нашого бота в системі, він діє як ключ. Цей токен треба зберегти нікому не показувати, він дуже важливий як пароль наприклад від почти. Нижче показано команди які зв'язані з токеном:

- /token - повертає отриманий токен у бота;
- /revoke - анулює токен;
- /setinline - включає та виключає можливість викликати бота з інших чатів;
- /setinlinegeo - включає або виключає можливість передавати місцеположення бота з іншого чату;
- /setinlinefeedback - дає змогу отримати інформацію про кількість вибраних користувачами команд;
- /setjoingroup - визначає чи є можливість додати в групові діалоги бота;
- /setprivcy - включає режим конфіденційності.

Є ще команди для ігор але вони нам в нашому проєкті непотрібні.

Після ознайомлення с BotFather переходимо до назви боту (малюнок 3.5) та отримуємо наш токен.



Малюнок 3.5 - назва боту та токен

Після Отримання токену, ми зможемо побачити, що наш бот готов та нижче (рисунок 3.6) можна побачити, що вже прописано в боті, та що можна змінити.

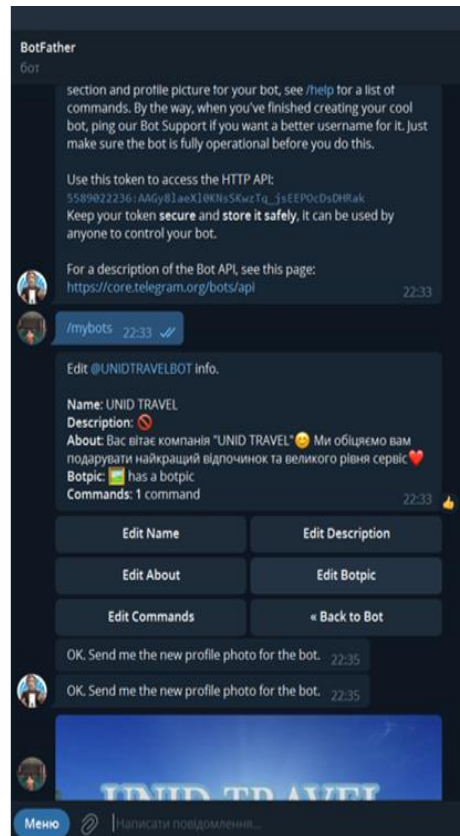


Рисунок 3.6 - зміни вигляду бота

Отримавши токен та поверхне налаштування можна перейти до розробки програмної частини.

3.2. Розробка на мові програмування Python

В дипломній роботі ми використали дві бібліотеки(рисунок 3.7). Бібліотека telebot — це спеціально розроблена бібліотека Python для написання чат-ботів. Ця бібліотека містить різні модулі для керування чат-ботами. Це дозволяє налаштувати його під себе. Telebot — дуже ефективний інструмент для написання ботів, завдяки наданим в ньому модулям ви можете змінити стандартну поведінку об'єкта, яка попередньо зареєстрована в телеграмі. Та бібліотека pymongo, вона використовується в Python для використання MongoDB. Формат файлів конфігурації config.py.



Рисунок 3.7 - Бібліотеки

Ми підключаємо токен, імпортуємо з `config.py` і беремо змінну Токен. Client-якщо простими словами, це підключення до `mongodb`, для цього ми використовуємо стандартний хостінг, для того щоб підключити `mongodb` використовуємо `// localhost : 27017/`. Далі можна побачити як ми використовуємо взаємозв'язок з базою даних(рисунок 3.8).



Рисунок 3.8 - підключення Токену

Далі можемо побачити як тільки ми створили змінну `Bot` ми будемо відловлювати повідомлення користувача. `Commands` в розробці буде нам показувати яку команду виконує користувач, якщо користувач виконав правильно команду ми будемо звертатися до функції `def`. За допомогою `Markup` ми створюємо дві кнопки, користувач зможе натиснути одну на вибір, та далі показано як буде реагувати сам бот і що буде відповідати(рисунок 3.9).

```

@bot.message_handler(commands=['start'])
def start(m):
    arrResult.clear()
    markup = telebot.types.ReplyKeyboardMarkup(resize_keyboard=True, one_time_keyboard=True)
    abroad = telebot.types.KeyboardButton("Тури за кордоном")
    tourUkraine = telebot.types.KeyboardButton("Тури по Україні")
    bot.send_message(m.chat.id, config.HELLO_MESS_START)
    markup.add(abroad, tourUkraine)
    bot.send_message(m.chat.id, config.HELLO_MESS_START2, reply_markup=markup)

@bot.message_handler(content_types=["text"])
def handle_text(m):
    if m.text.strip() == 'Тури за кордоном':
        arrResult.append(m.text)
        markup = telebot.types.ReplyKeyboardMarkup(resize_keyboard=True, one_time_keyboard=True)
        active = telebot.types.KeyboardButton("Активний відпочинок")
        passive = telebot.types.KeyboardButton("Спокійний відпочинок")
        markup.add(active, passive)
        msg = bot.send_message(m.chat.id, config.ABROADTYPE, reply_markup=markup)
        bot.register_next_step_handler(msg, handle_text_WhatCountry)
    elif m.text.strip() == 'Тури по Україні':
        arrResult.append(m.text)
        msg = bot.send_message(m.chat.id, config.TOURUKRAINEQUESSONE)
        bot.register_next_step_handler(msg, handle_text_ChooseType)

```

Рисунок 3.9 - код програми

На цьому рисунку можна побачити як ми налаштуємо бота, до процесу який був обговорений раніше. Як зберігаємо відповіді, як додаємо текст, за допомогою чого робимо панель кнопок, як відправляються значення після натискання кнопок користувачем, які ставимо умови для якісної роботи боту та як робимо опитувальник(рисунок 3.10).

```

@bot.message_handler(commands=['start'])
def start(m):
    arrResult.clear()
    markup = telebot.types.ReplyKeyboardMarkup(resize_keyboard=True, one_time_keyboard=True)
    abroad = telebot.types.KeyboardButton("Тури за кордоном")
    tourUkraine = telebot.types.KeyboardButton("Тури по Україні")
    bot.send_message(m.chat.id, config.HELLO_MESS_START)
    markup.add(abroad, tourUkraine)
    bot.send_message(m.chat.id, config.HELLO_MESS_START2, reply_markup=markup)

@bot.message_handler(content_types=["text"])
def handle_text(m):
    if m.text.strip() == 'Тури за кордоном':
        arrResult.append(m.text)
        markup = telebot.types.ReplyKeyboardMarkup(resize_keyboard=True, one_time_keyboard=True)
        active = telebot.types.KeyboardButton("Активний відпочинок")
        passive = telebot.types.KeyboardButton("Спокійний відпочинок")
        markup.add(active, passive)
        msg = bot.send_message(m.chat.id, config.ABROADTYPE, reply_markup=markup)
        bot.register_next_step_handler(msg, handle_text_WhatCountry)
    elif m.text.strip() == 'Тури по Україні':
        arrResult.append(m.text)
        msg = bot.send_message(m.chat.id, config.TOURUKRAINEQUESSONE)
        bot.register_next_step_handler(msg, handle_text_ChooseType)

def handle_text_WhatCountry(m):
    arrResult.append(m.text)

```

Рисунок 3.10 - код програми

На рисунках ми можемо побачити функцію, за допомогою якої ми бачимо запитання користувачеві(рисунки 3.11).

```

36
37
38 def handle_text_WhatCountry(m):
39     arrResult.append(m.text)
40     msg = bot.send_message(m.chat.id, config.WHATCOUNTRY)
41     bot.register_next_step_handler(msg, handle_text_Date)
42
43 def handle_text_Date(m):
44     arrResult.append(m.text)
45     msg = bot.send_message(m.chat.id, config.DATASET)
46     bot.register_next_step_handler(msg, handle_text_HowMuchDay)
47
48 def handle_text_HowMuchDay(m):
49     arrResult.append(m.text)
50     msg = bot.send_message(m.chat.id, config.DAYSET)
51     bot.register_next_step_handler(msg, handle_text_LevelService)
52
53 def handle_text_LevelService(m):
54     arrResult.append(m.text)
55     msg = bot.send_message(m.chat.id, config.LEVELSERVICE)
56     bot.register_next_step_handler(msg, handle_text_CountPeople)
57
58
59
60 def handle_text_CountPeople(m):
61     arrResult.append(m.text)
62     msg = bot.send_message(m.chat.id, config.COUNTPEOPLE)

```

Рисунок 3.11.1 - код програми

```

msg = bot.send_message(m.chat.id, config.WHATCOUNTRY)
bot.register_next_step_handler(msg, handle_text_Date)

def handle_text_Date(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.DATASET)
    bot.register_next_step_handler(msg, handle_text_HowMuchDay)

def handle_text_HowMuchDay(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.DAYSET)
    bot.register_next_step_handler(msg, handle_text_LevelService)

def handle_text_LevelService(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.LEVELSERVICE)
    bot.register_next_step_handler(msg, handle_text_CountPeople)

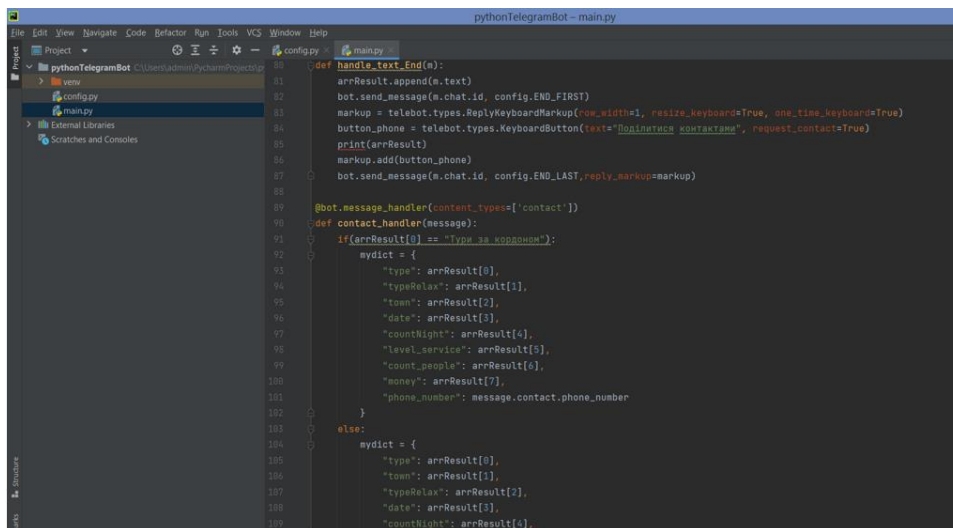
def handle_text_CountPeople(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.COUNTPEOPLE)
    bot.register_next_step_handler(msg, handle_text_HowMuchPay)

def handle_text_HowMuchPay(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.HOWMUCHPAY)
    bot.register_next_step_handler(msg, handle_text_End)

```

Рисунок 3.11.2 - код програми

Тут можна побачити як ми зможемо взяти контактні данні у клієнта і якщо він погодиться їх надати, то всі значення котрі ми набрали в масив будуть збережені в базу даних. В кінці бачимо як користувач зможе пройти ще раз опитування, якщо він щось не правильно відповів(рисунки 3.12).



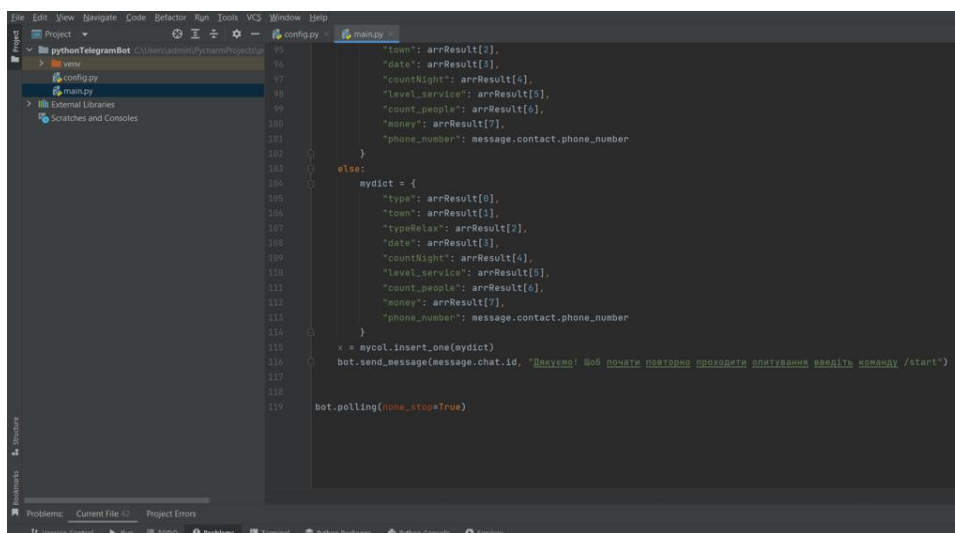
```

pythonTelegramBot - main.py
def handle_text_End(m):
    arrResult.append(m.text)
    bot.send_message(m.chat.id, config_END_FIRST)
    markup = telebot.types.ReplyKeyboardMarkup(row_width=1, resize_keyboard=True, one_time_keyboard=True)
    button_phone = telebot.types.KeyboardButton(text="@UnidTravelBot", request_contact=True)
    print(arrResult)
    markup.add(button_phone)
    bot.send_message(m.chat.id, config_END_LAST, reply_markup=markup)

@bot.message_handler(content_types=['contact'])
def contact_handler(message):
    if(arrResult[0] == "Типу за координат"):
        mydict = {
            "type": arrResult[0],
            "typeRelax": arrResult[1],
            "town": arrResult[2],
            "date": arrResult[3],
            "countNight": arrResult[4],
            "level_service": arrResult[5],
            "count_people": arrResult[6],
            "money": arrResult[7],
            "phone_number": message.contact.phone_number
        }
    else:
        mydict = {
            "type": arrResult[0],
            "town": arrResult[1],
            "typeRelax": arrResult[2],
            "date": arrResult[3],
            "countNight": arrResult[4],

```

Рисунок 3.12.1 - код програми



```

            "town": arrResult[2],
            "date": arrResult[3],
            "countNight": arrResult[4],
            "level_service": arrResult[5],
            "count_people": arrResult[6],
            "money": arrResult[7],
            "phone_number": message.contact.phone_number
        }
    else:
        mydict = {
            "type": arrResult[0],
            "town": arrResult[1],
            "typeRelax": arrResult[2],
            "date": arrResult[3],
            "countNight": arrResult[4],
            "level_service": arrResult[5],
            "count_people": arrResult[6],
            "money": arrResult[7],
            "phone_number": message.contact.phone_number
        }
    x = mydict.insert_one(mydict)
    bot.send_message(message.chat.id, "Дякуємо! Щоб почати повторно проходите опитування введіть команду /start")

bot.polling(none_stop=True)

```

Рисунок 3.12.2 - код програми

3.3. Тестування чат боту

Процес тестування здійснюється наступним чином: Спочатку Знайдіть створеного чат-бота(рисунок 3.12). Для цього потрібно знайти обліковий запис Називається @UNID TRAVELBOT. Як тільки ми його знайдемо, ми почнемо з ним розмову. Діалогове вікно буде порожнім і не запуститься, поки користувач не натисне кнопку «/start».

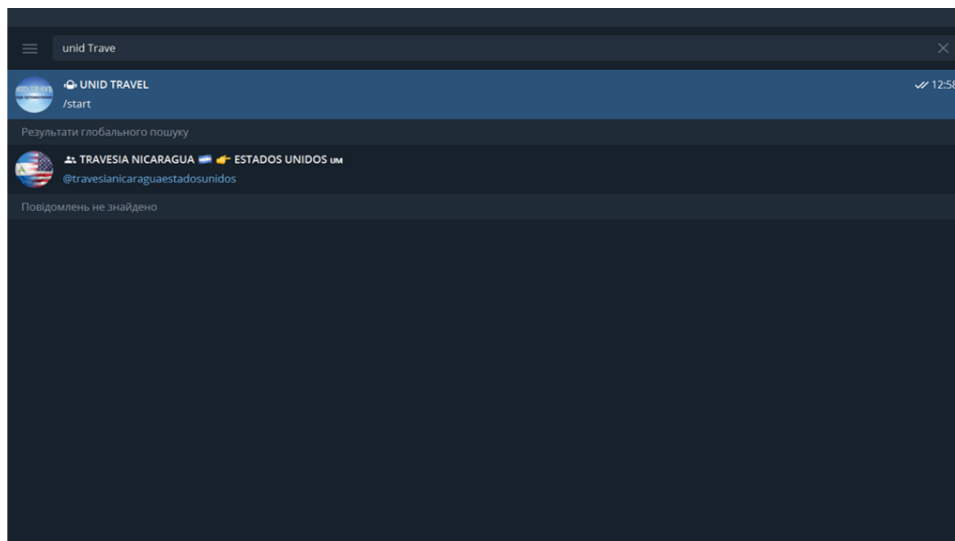


Рисунок 3.12 - бот UNID TRAVEL

Сам Бот, логотип та коротенька інформація про бот(рисунок 3.13)

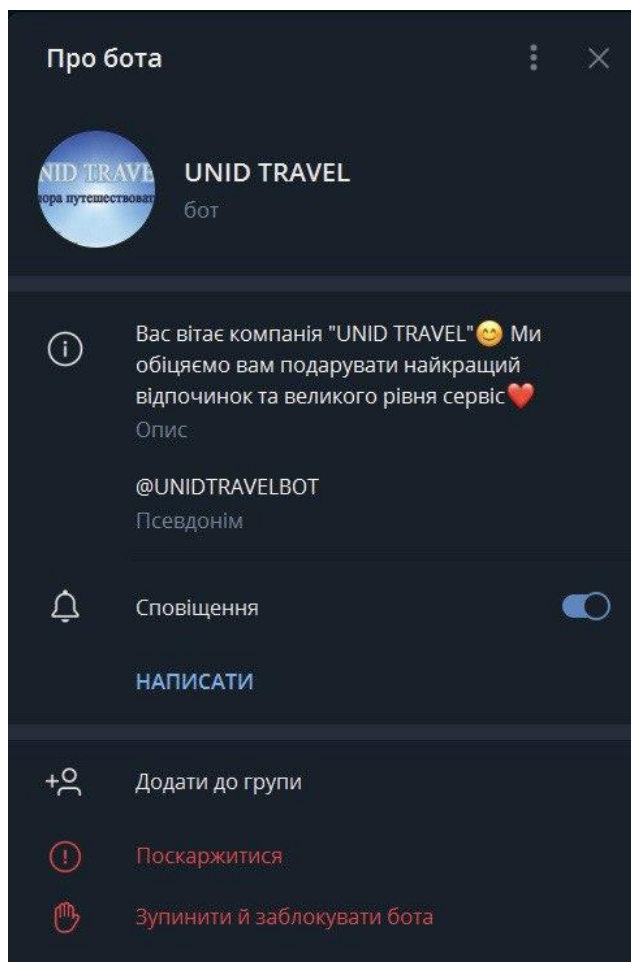


Рисунок 3.13 - Інформація про чат-бота

Після цього бот надішле вітальне повідомлення(рисунок 3.14)



Рисунок 3.14 - Вітальне повідомлення

Після цього бот запропонує на вибір два варіанта відповіді для уточнюючої інформації (рисунок 3.15).

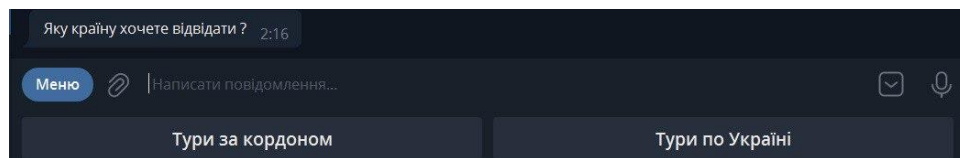


Рисунок 3.15 - варіанти відповіді

Далі бот пропонує знову два варіанти відповіді для уточнення інформації (рисунок 3.16)

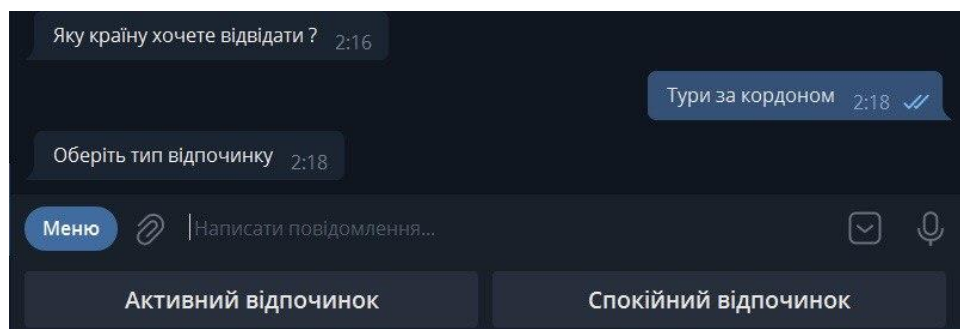


Рисунок 3.16 - варіанти відповідей

На наступному рисунку можна вибрати країну або її прапор (рисунок 3.17)

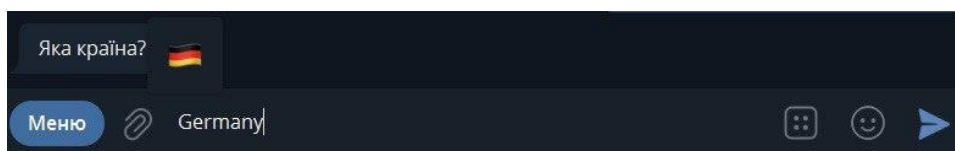


Рисунок 3.18 - рисунок чат-боту

Можна побачити уточнюючі питання для користувача (рисунок 3.19)

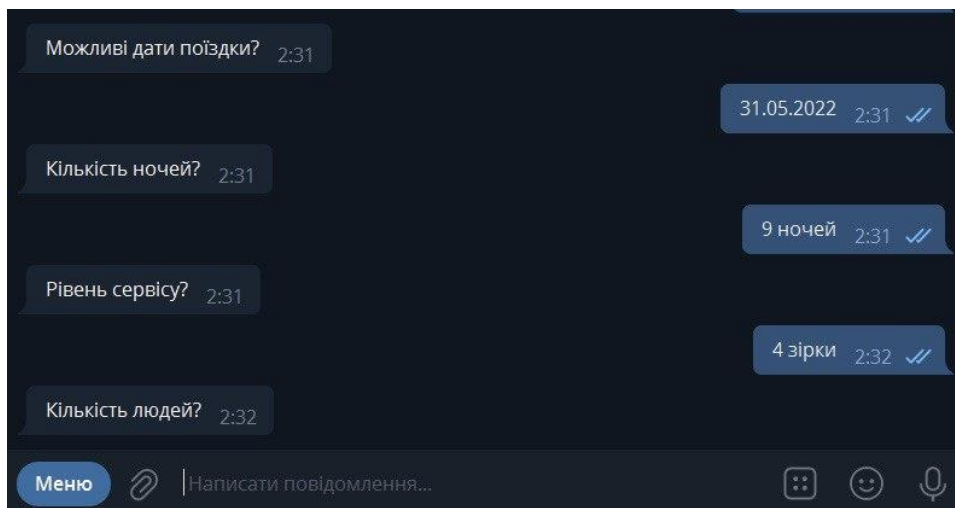


Рисунок 3.19 - Уточнюючі питання

Далі ми можемо побачити питання, та подяка за відповіді. Пропонується залишити контактні данні клієнта та уточнити час, коли можна бути зателефонувати(рисунок 3.20).

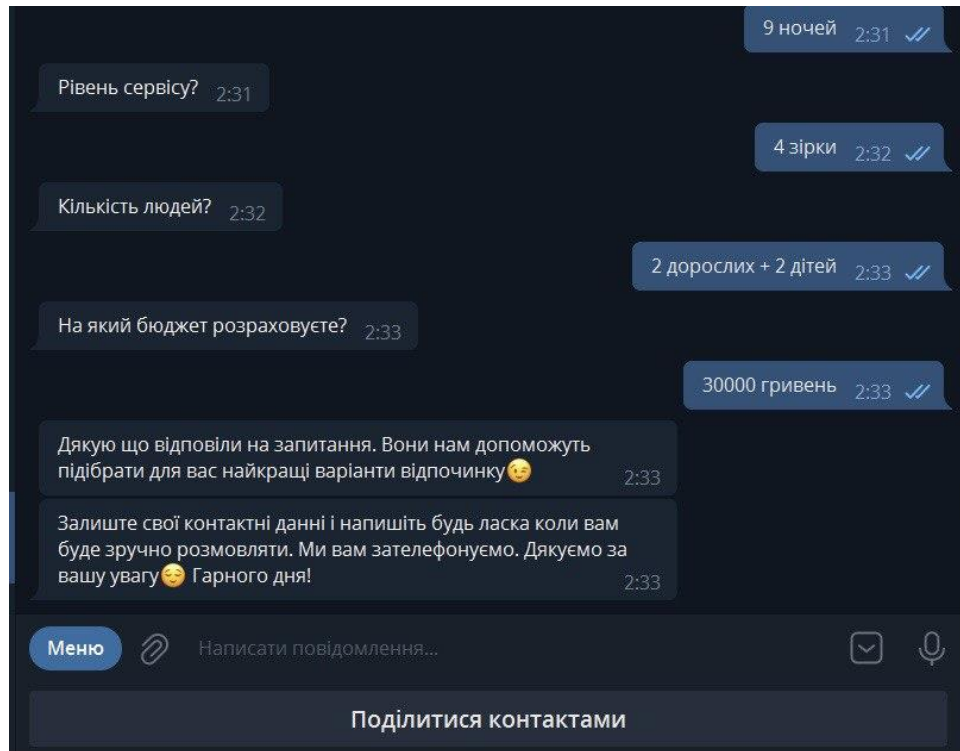


Рисунок 3.20 - Частина чат-боту

Клієнту не треба писати контактні дані, йому впливає вікно де він зможе поділитися своїми контактами (рисунок 3.21).

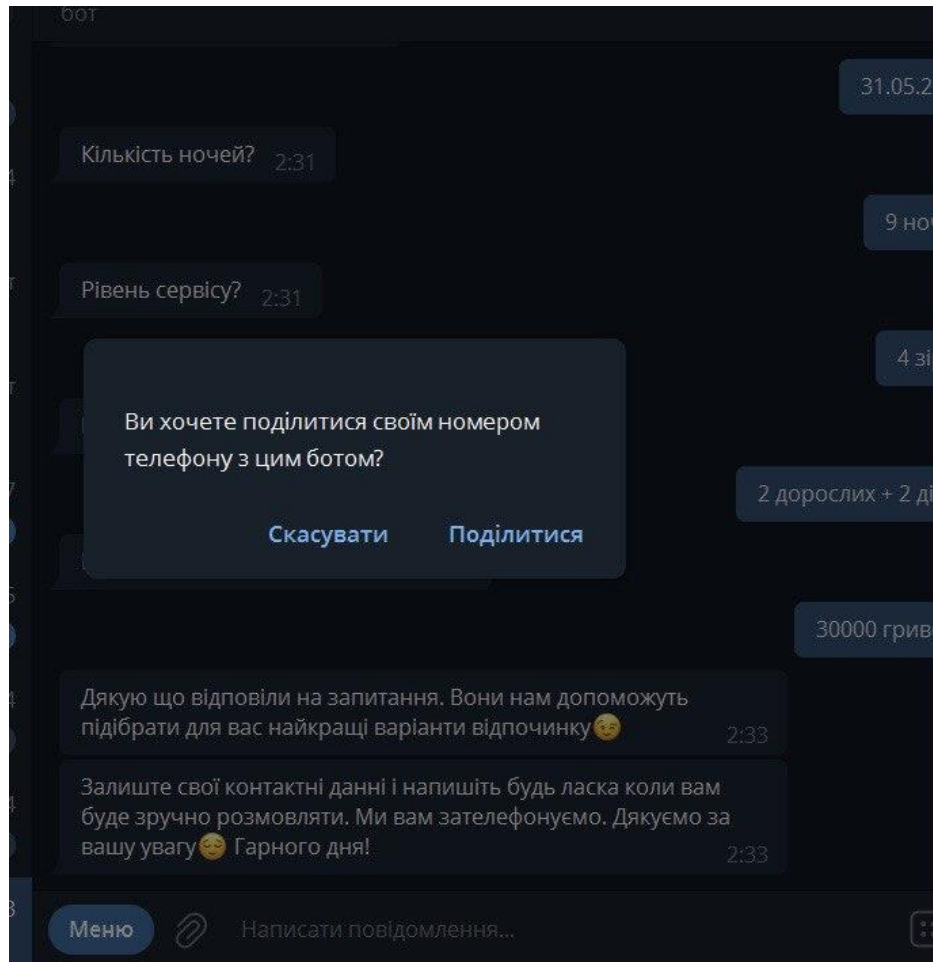


Рисунок 3.21 - запит на контактні дані

Звісно, в кінці бот дякує за відповіді на питання та якщо користувач помилився він зможе почати заново(рисунок 3.22).



Рисунок 3.22 - Кінцева частина чат боту

І одне з найголовніших питань яке вирішувалось, це збір даних та можливість менеджером легко та зручно знати дані користувача та проаналізувати, який тур потрібен, для цього менеджер зможе зайти в Mongodb Compass customers(рисунок 3.23).

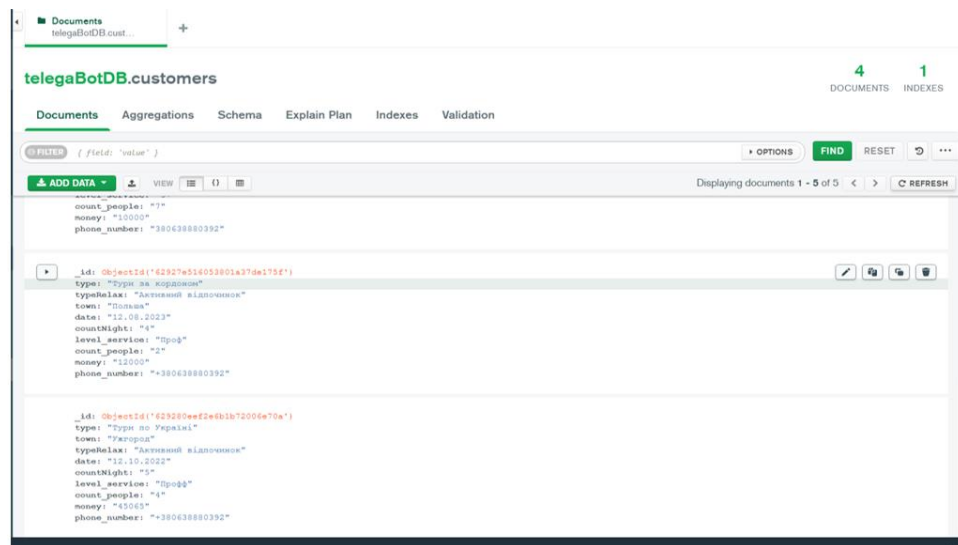


Рисунок 3.23 - Дані клієнтів

Таким чином, маємо поноційний бот, який збирає інформацію та дає змогу швидко та якісно подивитись дані які залишив клієнт.

Висновки

У даній роботі було розроблено чат-бот зброу інформації клієнта, за допомогою якого менеджер туристичного агенства зможе пришвидшити свою роботу, та компанія зможе заощаджувати кошти. За допомогою програмних засобів реалізауїї.

В першому розділі було розглянуто що таке взагалі чат-боти, та що в них корисного, і які є варіанти використання. Був розглянутий опис предметного середовища. Було розглянуто та проаналізовано платформи які підтримують чат-боти. В ході аналізу аналогів було проаналізовано які є мінуси в них, в основному мінуси були пов'язані з некоректною інформацією та з тим що не було української мови.

В другому розділі після постановки задач, було вибрано мову програмування Python, у цієї мови легкий синтаксис та її бібліотеки для чат- боту дуже легкі та практичні у використанні. Як середовища роробка був вибраний PyCharm. MongoDB була вибрана як база даних, тому що вона легка у використанні для менеджера компанії. Та було розроблено діаграми за допомогою яких можна побачити принцип роботи чат-боту.

В третьому розділі було розглянуто більш детально як працює сам бот, як працюють бібліотеки, як менеджер зможе легко використовувати базу даних. Був проведений огляд програмного коду, з описом функцій та методів. Було проведено тестування самого чат-боту, та показано як бот обмінюється повідомленнями з клієнтом та збирає цінну інформацію для менеджера.

Отже, при виконанні цієї роботи виконуються всі ключові етапи розробки програмного забезпечення. Від аналізу існуючих програм і виявлення проблем до вибору правильних інструментів і методів для вирішення проблем, розробки додатків і їх тестування.

Перелік посилань

1. Python programming language [Електронний ресурс]-Режим доступу до ресурсу:

<https://www.python.org/doc/essays/blurb/>

2. Telegram Api [Електронний ресурс] -Режим доступу до ресурсу:

<https://core.telegram.org/>

3. Bot [Електронний ресурс] -Режим доступу до ресурсу:

<https://www.actualidadgadget.com/uk/>

4. Viber telgram wats app [Електронний ресурс] -Режим доступу до ресурсу:

<https://www.semanticscholar.org/>

5. Java [Електронний ресурс] -Режим доступу до ресурсу:

<https://tebapit.com/>

6. Чат-боти [Електронний ресурс] -Режим доступу до ресурсу:

<https://web-promo.ua/ua/blog/vse-o-chat-botah-tipy-i-primery-kakomu-biznesu-podojdet-spisok-konstruktorov-dlya-sozdaniya/>

7. Бот платформи [Електронний ресурс] -Режим доступу до ресурсу:

<https://creativesmm.com.ua/shho-take-chatbot-ta-komu-vonu-potribni/>

8. A comparative study: MongoDB vs. MySQL [Електронний ресурс] - Режим доступу до ресурсу:

<https://ieeexplore.ieee.org/abstract/document/7158433>

9. Що таке чат-боти та кому вони потрібні [Електронний ресурс] – Режим доступу до ресурсу:

<https://creativesmm.com.ua/shho-take-chatbot-ta-komu-vonu-potribni/>

10. PyCharm Community Edition [Електронний ресурс] – Режим доступу до ресурсу:

<https://github.com/JetBrains/intellij-community/tree/master/python>

11. PyCharm Homepage [Електронний ресурс] – Режим доступу до ресурсу:

<https://www.jetbrains.com/pycharm/>

12. Инструкция: как создавать ботов в Telegram [Електронний ресурс] – Режим доступу до ресурсу:

<https://habr.com/ru/post/262247/>

13. Телеграм бот за допомогою TeleBot [Електронний ресурс] - Режим доступу до ресурсу:

<https://habr.com/ru/post/448310/>

14. Топ-10 корисних Telegram-ботів для українських користувачів[Електронний ресурс] - Режим доступу до ресурсу:

<https://www.epravda.com.ua/publications/2017/08/7/627822>

Додаток А

```

import telebot
import config
import pymongo

bot = telebot.TeleBot(config.TOKEN)
client = pymongo.MongoClient("mongodb://localhost:27017/")
mydb = client["telegaBotDB"]
mycol = mydb["customers"]

arrResult = []

@bot.message_handler(commands=['start'])
def start(m):
    arrResult.clear()
    markup = telebot.types.ReplyKeyboardMarkup(resize_keyboard=True,
one_time_keyboard=True)
    abroad = telebot.types.KeyboardButton("Тури за кордоном")
    tourUkraine = telebot.types.KeyboardButton("Тури по Україні")
    bot.send_message(m.chat.id, config.HELLO_MESS_START)
    markup.add(abroad,tourUkraine)
    bot.send_message(m.chat.id, config.HELLO_MESS_START2,
reply_markup=markup)

```

```

@bot.message_handler(content_types=["text"])
def handle_text(m):
    if m.text.strip() == 'Тури за кордоном':
        arrResult.append(m.text)
        markup = telebot.types.ReplyKeyboardMarkup(resize_keyboard=True,
one_time_keyboard=True)
        active = telebot.types.KeyboardButton("Активний відпочинок")
        passive = telebot.types.KeyboardButton("Спокійний відпочинок")
        markup.add(active, passive)
        msg = bot.send_message(m.chat.id, config.ABROADTYPE,
reply_markup=markup)
        bot.register_next_step_handler(msg, handle_text_WhatCountry)
    elif m.text.strip() == 'Тури по Україні':
        arrResult.append(m.text)
        msg = bot.send_message(m.chat.id, config.TOURUKRAINEQUESSONE)
        bot.register_next_step_handler(msg, handle_text_ChooseType)

def handle_text_WhatCountry(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.WHATCOUNTRY)
    bot.register_next_step_handler(msg, handle_text_Date)

def handle_text_Date(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.DATASET)
    bot.register_next_step_handler(msg, handle_text_HowMuchDay)

def handle_text_HowMuchDay(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.DAYSET)
    bot.register_next_step_handler(msg, handle_text_LevelService)

def handle_text_LevelService(m):
    arrResult.append(m.text)
    msg = bot.send_message(m.chat.id, config.LEVELSERVICE)
    bot.register_next_step_handler(msg, handle_text_CountPeople)

def handle_text_CountPeople(m):
    arrResult.append(m.text)

```

```

msg          =          bot.send_message(m.chat.id,          config.COUNTPEOPLE)
bot.register_next_step_handler(msg,          handle_text_HowMuchPay)

def          handle_text_HowMuchPay(m):
    arrResult.append(m.text)
    msg          =          bot.send_message(m.chat.id,          config.HOWMUCHPAY)
    bot.register_next_step_handler(msg,          handle_text_End)

def          handle_text_ChooseType(m):
    arrResult.append(m.text)
    markup          =          telebot.types.ReplyKeyboardMarkup(resize_keyboard=True,
one_time_keyboard=True)
    active          =          telebot.types.KeyboardButton("Активний          відпочинок")
    passive          =          telebot.types.KeyboardButton("Спокійний          відпочинок")
    markup.add(active,          passive)
    msg          =          bot.send_message(m.chat.id,          config.CHOOSETYPEUA,
reply_markup=markup)
    bot.register_next_step_handler(msg,          handle_text_Date)

def          handle_text_End(m):
    arrResult.append(m.text)
    bot.send_message(m.chat.id,          config.END_FIRST)
    markup = telebot.types.ReplyKeyboardMarkup(row_width=1, resize_keyboard=True,
one_time_keyboard=True)
    button_phone = telebot.types.KeyboardButton(text="Поділитися          контактами",
request_contact=True)
    print(arrResult)
    markup.add(button_phone)
    bot.send_message(m.chat.id,          config.END_LAST,reply_markup=markup)

@bot.message_handler(content_types=['contact'])
def          contact_handler(message):
    if(arrResult[0]          ==          "Тури          за          кордоном"):
        mydict          =          {
            "type":          arrResult[0],
            "typeRelax":          arrResult[1],
            "town":          arrResult[2],
            "date":          arrResult[3],
            "countNight":          arrResult[4],
            "level_service":          arrResult[5],
            "count_people":          arrResult[6],

```

```

        "money": arrResult[7],
        "phone_number": message.contact.phone_number
    }
else:
    mydict = {
        "type": arrResult[0],
        "town": arrResult[1],
        "typeRelax": arrResult[2],
        "date": arrResult[3],
        "countNight": arrResult[4],
        "level_service": arrResult[5],
        "count_people": arrResult[6],
        "money": arrResult[7],
        "phone_number": message.contact.phone_number
    }
x = mycol.insert_one(mydict)
bot.send_message(message.chat.id, "Дякуємо! Щоб почати повторно проходити
опитування введіть команду /start")

bot.polling(none_stop=True)

```

Додаток Б



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
 НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
 ТЕХНОЛОГІЙ
 КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Розробка чат-боту для туристичного агентства UNID TRAVEL мовою Python

Виконав студент 5 курсу

групи ППЗ-51

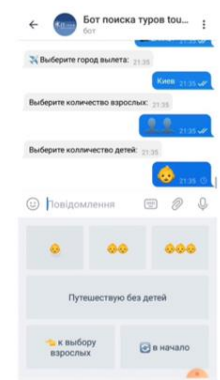
Струпінський Ілля Юрійович

Керівник роботи

Старший викладач Коба Андрій Борисович

Київ – 2022

Аналоги



Toursitebot



Eddy Travels

Порівняльна таблиця аналогів

Назва	Плюси	Мінуси
Toursitebot	1. Надає змогу відповідати на все варіантами кнопок. 2. Дає змогу після надання інформації боту, відразу надати посилання на сайт де під критерії вже буде підібрані готелі. 3. Зібрання даних клієнтів для подальшої роботи з ними.	1. Мало країн для вибору клієнта. 2. Не має Української мови. 3. Не має турів по Україні. 4. Сайт іноді не коректно підбирає тури а іноді взагалі не дає варіантів і не має куди подзвонити та клієнт не залишає номер для роботи менеджера.
Eddy Travels	1. Підтримка 24/7. 2. Можна відповідати голосовими повідомленнями. 3. Дозволяє відстежувати ціни на квитки або готелі. 4. Можна підібрати для себе готель.	1. Не можна зв'язатися з менеджером, тому є трішки не довіра бронювання або покупки білетів. 2. Є мови на вибір але все одно всі відповіді на англійській мові. 3. Потрібна підписка щоб користуватися ним весь час.

3

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Мета роботи** – підвищення якості роботи менеджера туристичних компаній за рахунок розробленого чат-бота для збирання інформації про тур та отримання контактних даних від клієнта.
- **Об'єкт дослідження** – процес роботи менеджера туристичних компаній.
- **Предмет дослідження** – телеграм чат-бот для збирання інформації та контактних даних.


4


ТЕХНІЧНІ ЗАВДАННЯ


1. Провести аналіз предметної області
2. Можливість отримувати результат опитування
3. Розробити діаграму прецедентів
4. Розробити можливість користувача залишати данні
5. Обґрунтувати вибране середовища реалізації додатку
6. Розробити елементи чату-боту


5

ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ

- 

Telegram - месенджер в якому реалізован бот
- 

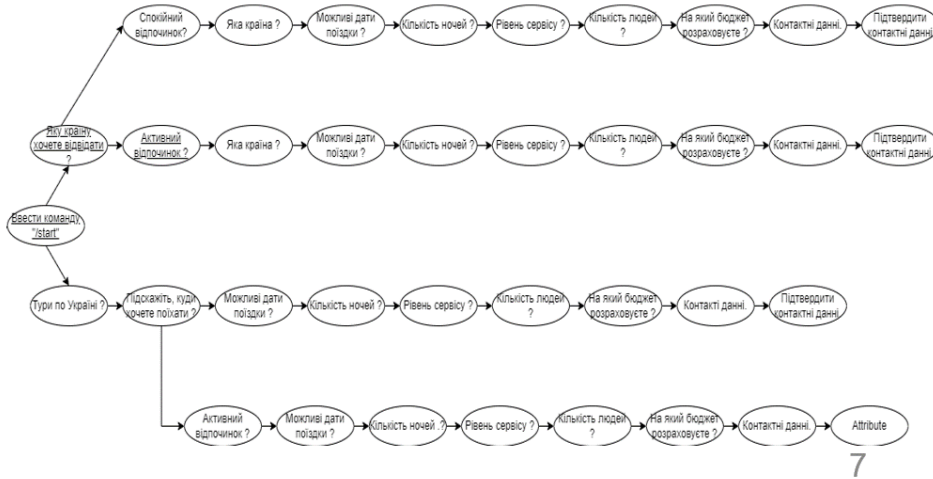
Pycharm - середовища для написання коду
- 

Mongodb - база даних яка використовується в проєкті
- 

Python - мова програмування, за допомогою якої створюються телеграм-боти

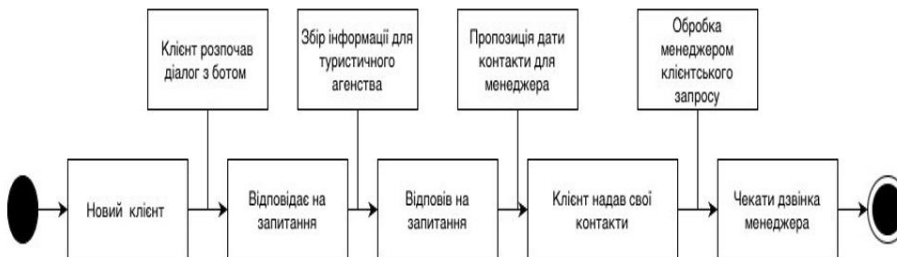
6

Схема роботи бота



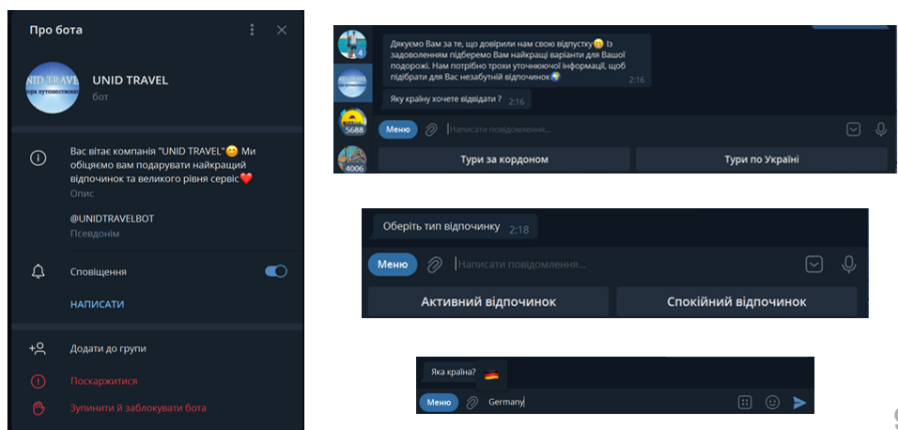
7

Діаграма станів



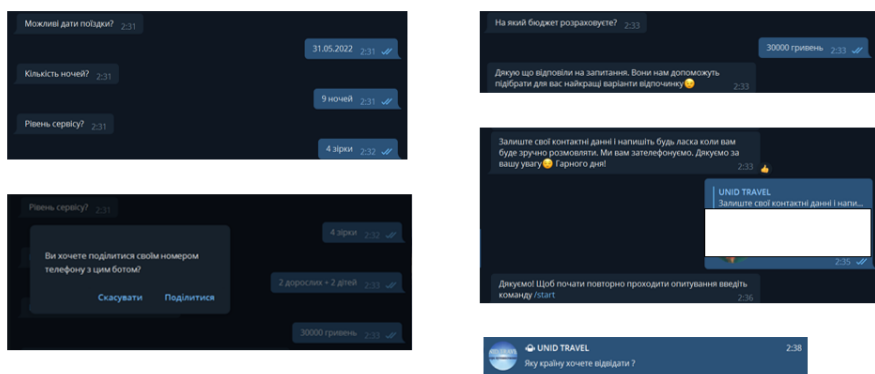
8

Екранні форми



9

Екранні форми



10

АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

- Струпінський І.Ю Розробка чат боту для туристичного агенства UNID TRAVEL мовою Python/ Міжнародна студентська конференція Наука сьогодні: від досліджень до стратегічних рішень" 17.06.2022, м.Івано-Франківськ

ВИСНОВКИ

- 1.Проведений аналіз предметної області.
- 2.Розроблена діаграма прецедентів.
- 3.Розроблена можливість отримувати результати дослідження.
- 4.Розроблена можливість користувача залишати контактні данні для подальшої роботи з менеджером.
- 5.Обґрунтоване вибране середовища реалізації додатку.
- 6.Розроблено елементи чат-боту.

12

ДЯКУЮ ЗА УВАГУ!