

Міністерство освіти і науки України
Національний технічний університет України „КПІ
імені Ігоря Сікорського ”
Факультет інформатики та обчислювальної техніки
Кафедра інформатики і програмної інженерії

ЗВІТ

лабораторної роботи № 3
з курсу «Основи WEB - технологій»

Тема: «Створення telegram-боту з меню та запитом ChatGPT»

Перевірів:

Викл. Альбрехт Й.О.

Виконав:

Студент ІІІ-15
Мешков А.І

Київ 2024

Завдання

1. Створити telegram-бот з меню та задеплоїти його на сервісі <https://pythonanywhere.com/>.

Хід роботи

Було створено бот, який використовує бібліотеку aiogram для API телеграму, та groq для AI запитів замість ChadGPT.

bot.py

```
import asyncio
import logging
import sys
import os

from aiogram import Bot, Dispatcher, Router
from aiogram.client.default import DefaultBotProperties
from aiogram.enums import ParseMode
from aiogram.filters import CommandStart
from aiogram.types import Message, InlineKeyboardButton, InlineKeyboardMarkup, CallbackQuery

from groq import Groq

TOKEN = os.getenv("TELEGRAM_TOKEN")
dp = Dispatcher()
router = Router()

client = Groq(
    api_key=os.getenv("GROQ_API_KEY"),
)

async def send_main_menu(message):
    keyboard = InlineKeyboardMarkup(inline_keyboard=[
        [
            InlineKeyboardButton(text="Студент", callback_data="student"),
            InlineKeyboardButton(text="IT-технології", callback_data="it_tech"),
            InlineKeyboardButton(text="Контакти", callback_data="contacts"),
            InlineKeyboardButton(text="Prompt Groq", callback_data="prompt_groq"),
        ]
    ])

    if isinstance(message, Message):
        await message.answer("Виберіть розділ:", reply_markup=keyboard)
    elif isinstance(message, CallbackQuery):
        await message.message.edit_text("Виберіть розділ:", reply_markup=keyboard)

@dp.message(CommandStart())
async def command_start_handler(message: Message) -> None:
    await send_main_menu(message)
```

```

@dp.callback_query()
async def callback_query_handler(callback_query: CallbackQuery):

    keyboard = InlineKeyboardMarkup(inline_keyboard=[
        [
            InlineKeyboardButton(text="Назад", callback_data="back"),
        ]
    ])

    if callback_query.data == "student":
        response_text = "Студент: Мешков Андрій, група ІП-15"
        await callback_query.message.edit_text(response_text, reply_markup=keyboard)
        await callback_query.answer()

    elif callback_query.data == "it_tech":
        response_text = "IT-технології: Frontend, Backend"
        await callback_query.message.edit_text(response_text, reply_markup=keyboard)
        await callback_query.answer()

    elif callback_query.data == "contacts":
        response_text = "Контакти: Телефон – +380951039335, Email – andrey2004112@gmail.com"
        await callback_query.message.edit_text(response_text, reply_markup=keyboard)
        await callback_query.answer()

    elif callback_query.data == "prompt_groq":
        response_text = "Prompt Groq: Введіть ваш запит у наступному повідомленні."
        await callback_query.message.edit_text(response_text)
        await callback_query.answer()
    elif callback_query.data == "back":
        await send_main_menu(callback_query)

@dp.message()
async def echo_handler(message: Message) -> None:
    try:
        await message.send_copy(chat_id=message.chat.id)

        reply = client.chat.completions.create(
            messages=[
                {
                    "role": "user",
                    "content": message.text,
                }
            ],
            model="llama3-8b-8192",
        )
        response_text = reply.choices[0].message.content
    except Exception as e:
        response_text = f"Error: {e}"

```

```
    await message.answer(response_text)
    await send_main_menu(message)

async def main() -> None:

    print(os.getenv("GROQ_API_KEY"))
    print(os.getenv("TELEGRAM_TOKEN") )
    bot = Bot(token=TOKEN, default=DefaultBotProperties(parse_mode=ParseMode.HTML))

    await dp.start_polling(bot)

if __name__ == "__main__":
    logging.basicConfig(level=logging.INFO, stream=sys.stdout)
    asyncio.run(main())
```

1. Отримані результати

На рис 3.1-3.6 можна побачити результат роботи боту.

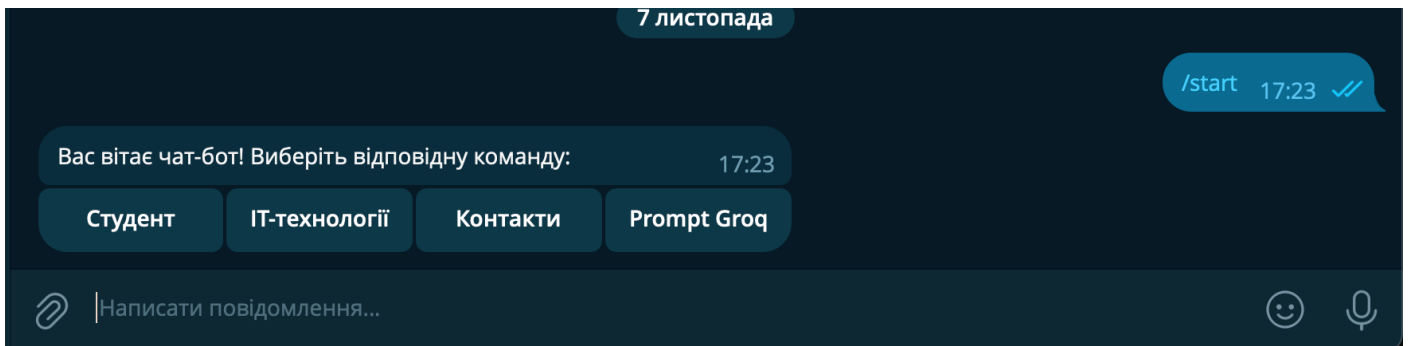


Рис. 3.1. Початок роботи

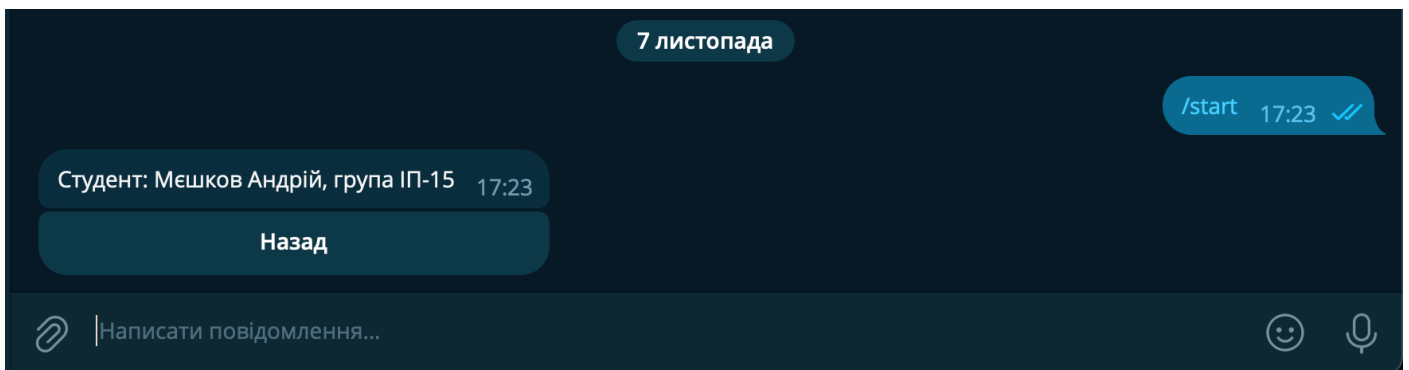


Рис. 3.2. Команда студент

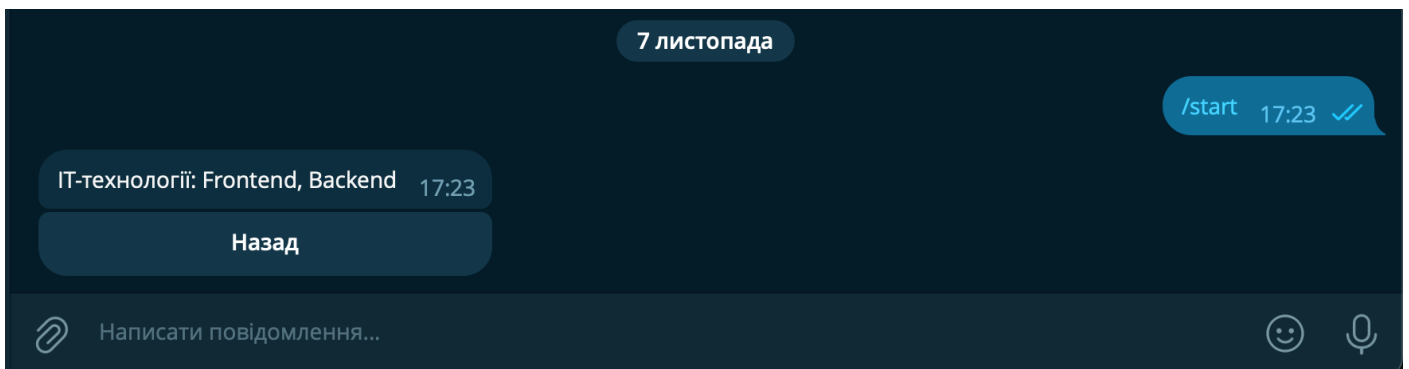


Рис. 3.3. Команда ІТ-технології

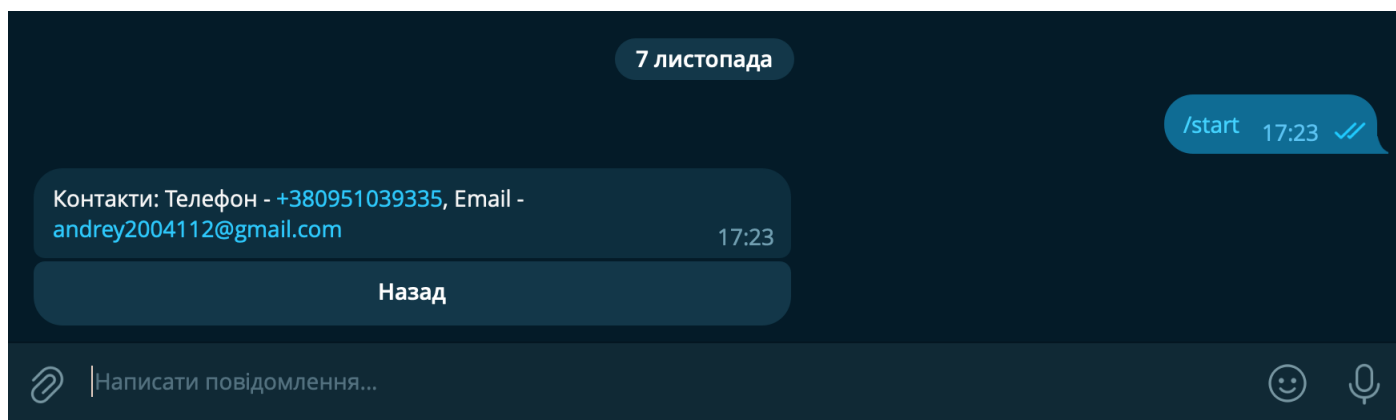


Рис. 3.4. Команда контакти

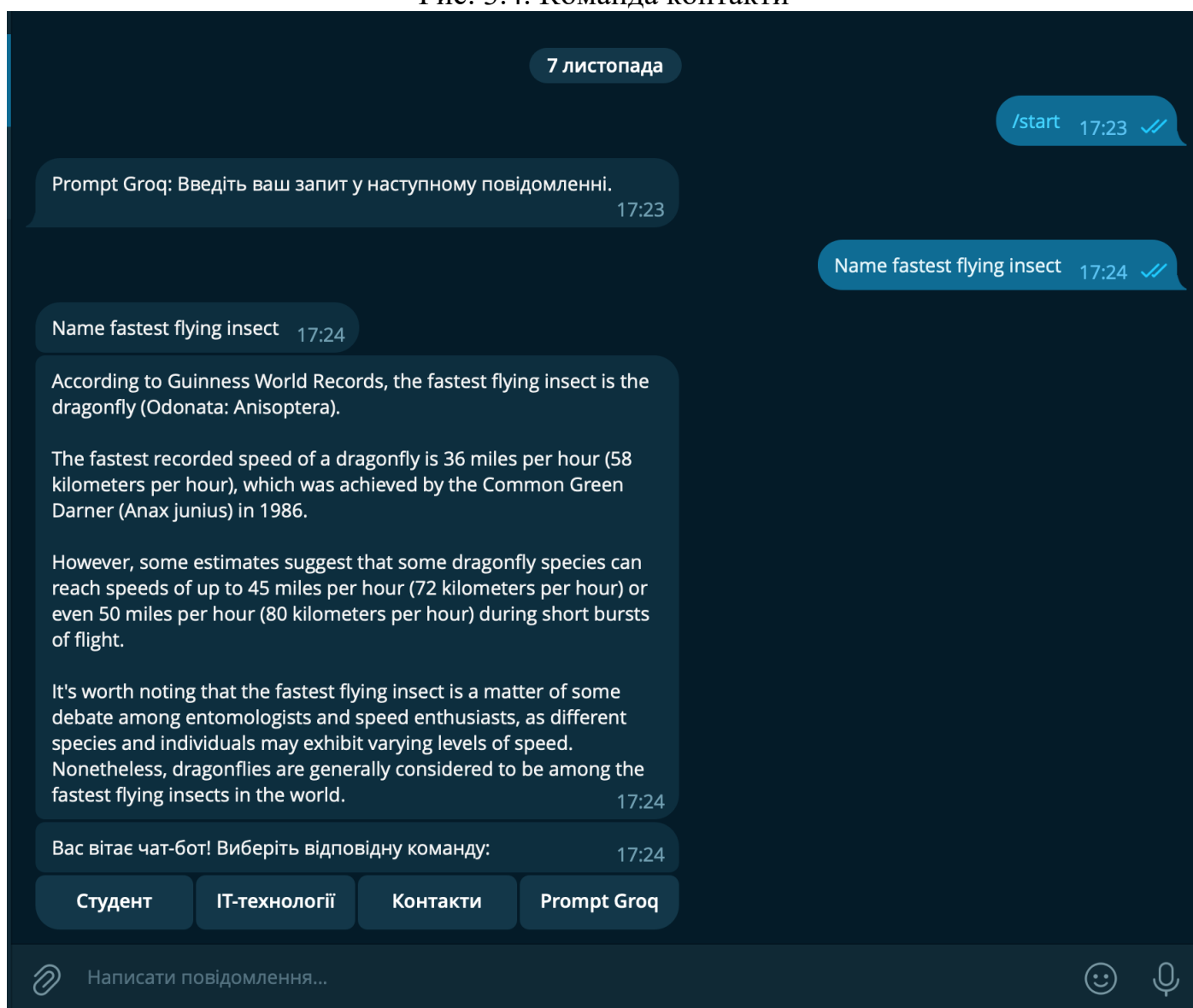


Рис. 3.5. Запит Groq

На рисунку 3.6 можна побачити сторінку pythonanywhere.com з задеплоїним проєктом.

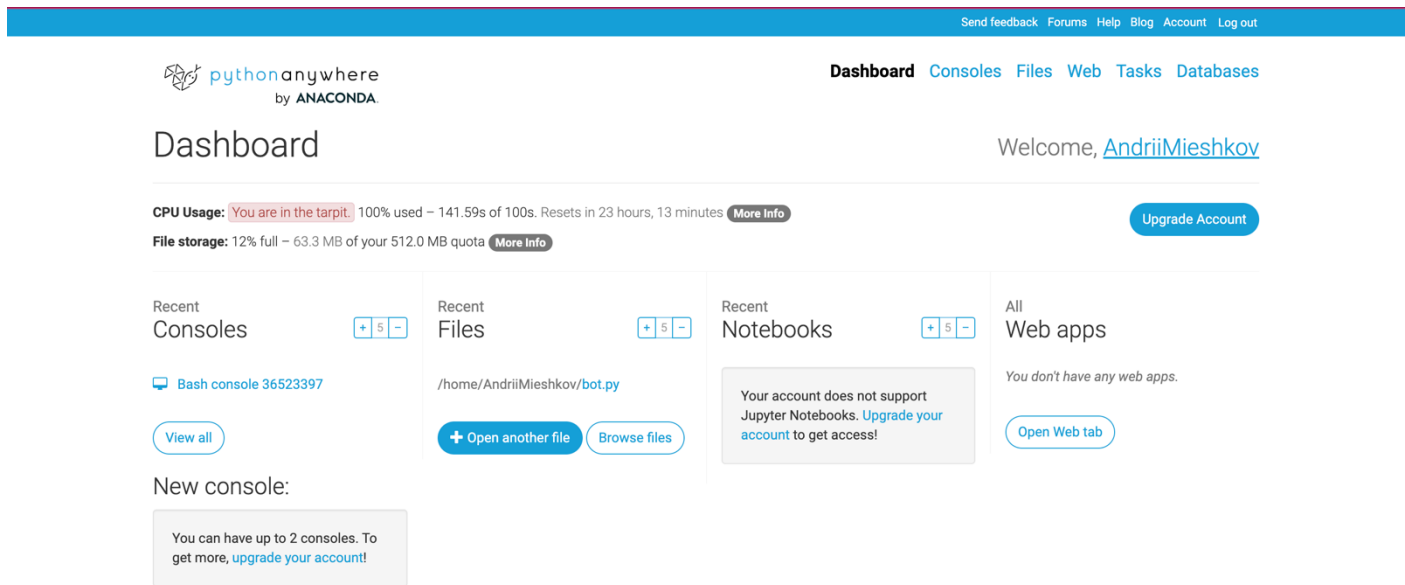


Рис. 3.6. Задеплоєна робота

ВИСНОВОК

Під час виконання даної лабораторної роботи було створено Telegram-бота з використанням бібліотеки `aiogram` для інтеграції з API Telegram та бібліотеки `groq` для виконання запитів до штучного інтелекту. Основні етапи роботи включали:

1. Створення меню для взаємодії користувача з ботом за допомогою кнопок Inline Keyboard.
2. Налаштування обробки команд та callback-запитів для забезпечення відповідей бота на запити користувачів.
3. Впровадження можливості введення запитів до моделі AI через Groq API для отримання відповідей.
4. Деплой та тестування бота на хостингу PythonAnywhere.

У результаті лабораторної роботи ми отримали практичний досвід:

- налаштування асинхронного обробника подій на базі бібліотеки `aiogram`;
- роботи з клавіатурами Telegram бота;
- інтеграції зовнішнього сервісу для обробки AI запитів;
- деплою проекту на сервері, що включає налаштування проксі для стабільної роботи в мережі.

Завдання виконано успішно, і бот працює згідно з поставленими вимогами, дозволяючи користувачеві обирати різні розділи меню, а також робити запити до штучного інтелекту.