

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»

|  |  |
| --- | --- |
| Перевірив:  Викл. Альбрехт Й.О. | Виконав:  Студент ІП-15 Мєшков 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 aoigram 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="ІТ-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: Мєшков 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 = "ІТ-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())

# Отримані результати

На рис 3.1-3.6 можна побачити результат роботи боту.

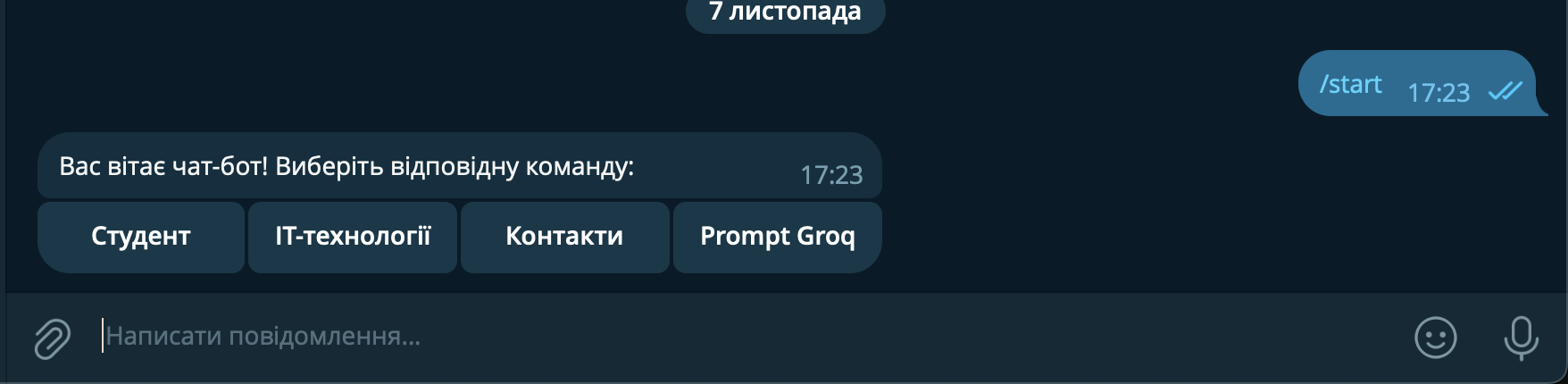


Рис. 3.1. Початок роботи

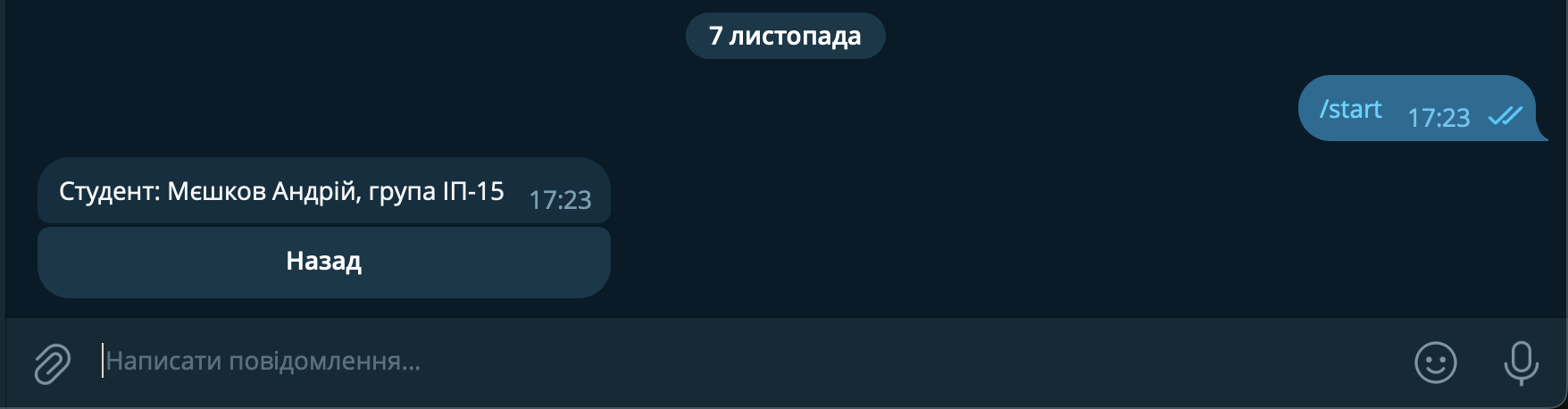


Рис. 3.2. Команда студент

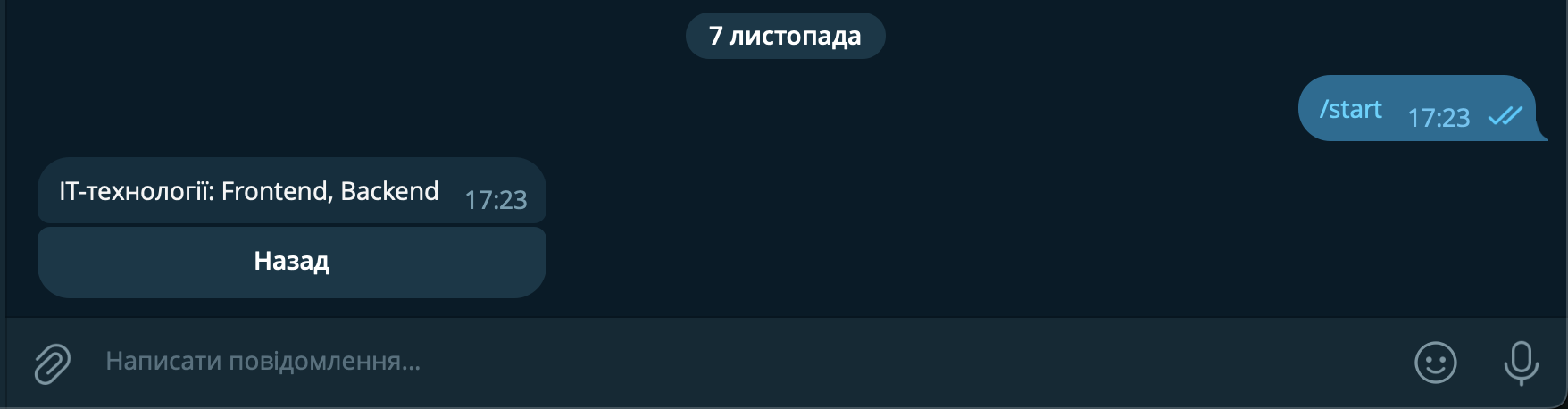


Рис. 3.3. Команда IT-технології

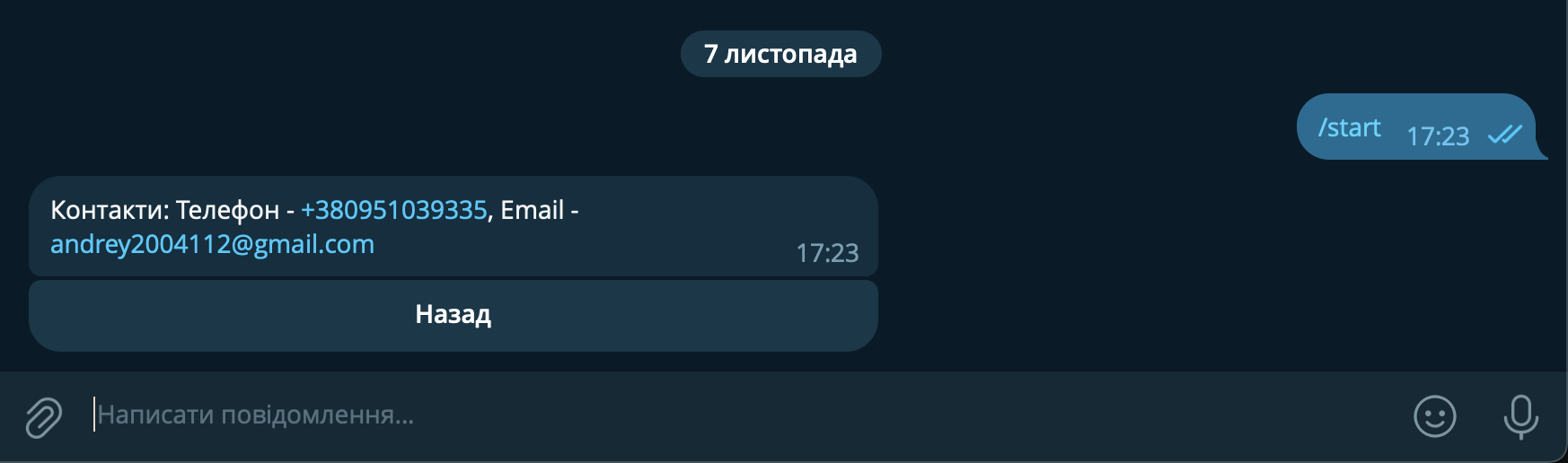


Рис. 3.4. Команда контакти

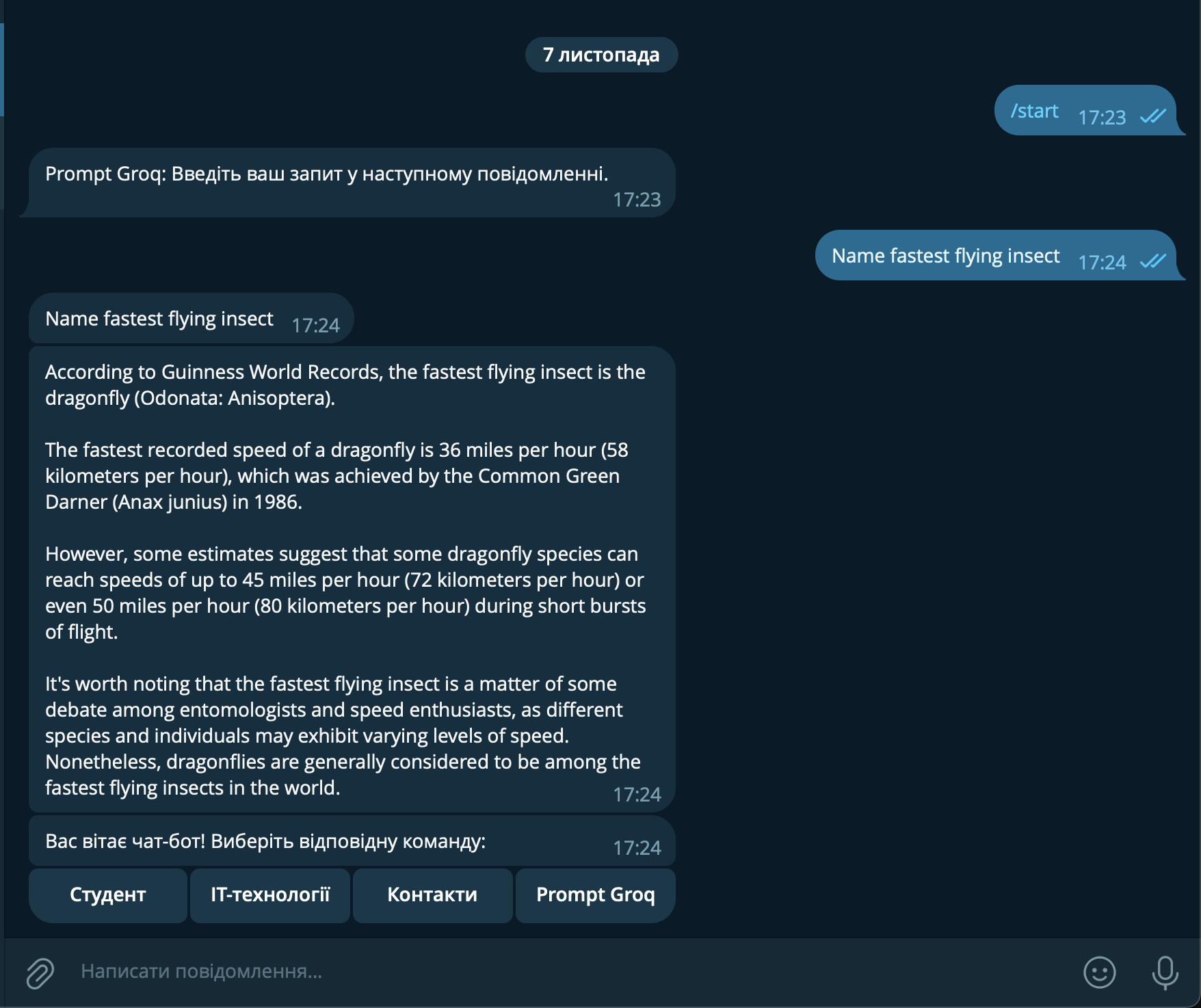


Рис. 3.5. Запит Groq

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

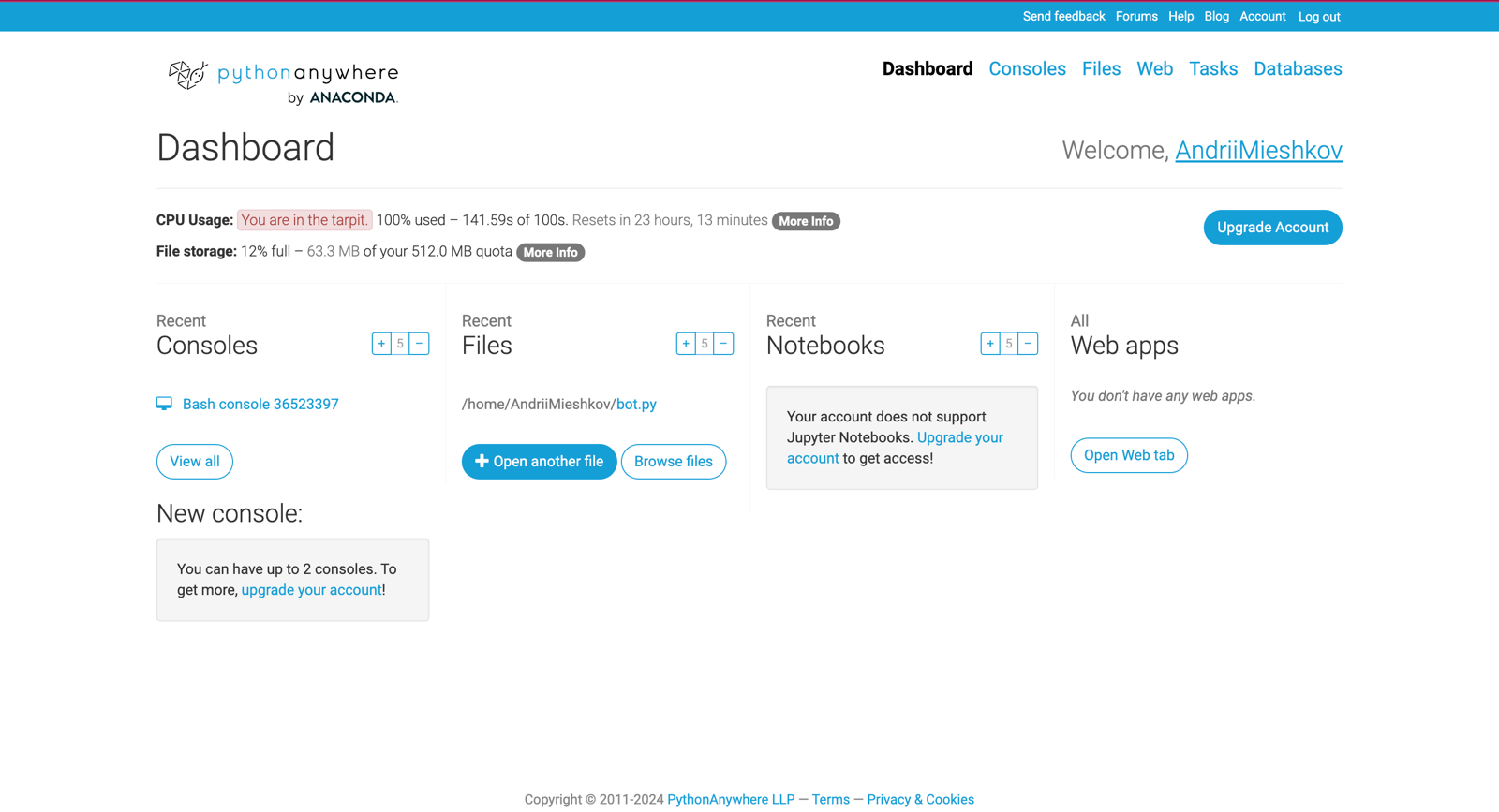


Рис. 3.6. Задеплоєна робота

# 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.