

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
Факультет кібербезпеки, комп'ютерної та програмної інженерії
Кафедра комп'ютерних інформаційних технологій

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач кафедри

_____ (Савченко А.С.)
« _____ » _____ 2021 р.

ДИПЛОМНИЙ ПРОЕКТ
(ПОЯСНЮВАЛЬНА ЗАПИСКА)
ВИПУСКНИКА ОСВІТНЬОГО СТУПЕНЯ
“БАКАЛАВР”

Тема: Telegram-бот для гуртожитку на мові високого рівня програмування

Виконавець: Гасанова Аліса Мавсумівна

Керівник: професор Зіатдінов Юрій Кашафович

Нормоконтролер: ст. викл. Шевченко Олександр Петрович

Київ 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра Комп'ютерних інформаційних технологій

Освітня ступінь: Бакалавр

Галузь знань, спеціальність, спеціалізація: 12 "Інформаційні технології",
122 "Комп'ютерні науки", "Інформаційні управляючі системи та технології"

ЗАТВЕРДЖУЮ
Завідувач кафедри

_____ (Савченко А. С.)
« _____ » _____ 2021 р.

ЗАВДАННЯ

на виконання дипломного проекту студентки

Гасанової Аліси Мавсумівни

1. Тема дипломного проекту: «Телеграм-бот для гуртожитку на мові високого рівня програмування»
затверджена наказом ректора від 22.04.2021 р. №636/ст.
2. Термін виконання роботи: з 11.05.21 по 14.06.21 року.
3. Вихідні дані до роботи: розробка і створення Телеграм-боту на мові високого рівня програмування.
4. Зміст пояснювальної записки: вступ, Телеграм-бот, як сучасний спосіб поширення інформації, технічні засоби для реалізації продукту, розробка продукту, висновки.
5. Перелік обов'язкового графічного матеріалу: кодування боту, приклад роботи.

КАЛЕНДАРНИЙ ПЛАН-ГРАФІК

№ п/п	Етапи виконання дипломної роботи	Термін виконання етапів	Примітка
1.	Збір матеріалу, що стосується теми дипломної роботи.	11.05.2021 14.05.2021	
2.	Розписано постановку задачі та методи її реалізації.	15.05.2021 17.05.2021	
3.	Приступлено до виконання індивідуального завдання. Завершено збір матеріалів.	18.05.2021 19.05.2021	
4.	Аналіз зібраних матеріалів, написано перший розділ.	20.05.2021 23.05.2021	
5.	Підготовка технологій перед розробкою продукту. Написано другий розділ.	24.05.2021 29.05.2021	
6.	Реалізація даного продукту за допомогою мови високого рівня програмування. Написано третій розділ.	30.05.2021 04.06.2021	
7.	Оформлення та редагування дипломного проекту, написано висновки.	05.06.2021 07.06.2021	

Студентка

(Гасанова А.М.)

Керівник дипломної роботи

(Зіатдінов Ю.К.)

РЕФЕРАТ

Пояснювальна записка до дипломної роботи «Телеграм-бот для гуртожитку на мові високого рівня програмування»: 54 с., 47 рис., 7 літературних джерел.

Об'єкт дослідження: Телеграм-бот для гуртожитку на мові високого рівня програмування.

Мета роботи: створення продукту, що допоможе мешканцям гуртожитку швидко знайти відповіді на поширені питання.

Методи дослідження: аналіз написання Телеграм-боту.

Результат проекту: Телеграм-бот для гуртожитку.

ТЕЛЕГРАМ, ТЕЛЕГРАМ-БОТ, PYTHON, БД, API, VS CODE.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ.....	6
ВСТУП.....	7
РОЗДІЛ 1. ТЕЛЕГРАМ-БОТ, ЯК СУЧАСНИЙ СПОСІБ ПОШИРЕННЯ ІНФОРМАЦІЇ.....	8
1.1. Особливість використання месенджера Телеграм	8
1.2. Переваги та недоліки месенджеру Телеграм	15
1.3. Сутність Телеграм-боту	16
1.4. Види Телеграм-ботів	20
1.5. Взаємодія між користувачем і ботом.....	23
1.6. Python, як мова для Телеграм-боту.....	24
Висновки до першого розділу.....	25
РОЗДІЛ 2. ТЕХНІЧНІ ЗАСОБИ ДЛЯ РЕАЛІЗАЦІЇ ПРОДУКТУ.....	26
2.1. Утиліта Botfather	26
2.2. Середовище розробки.....	29
2.3. Telegram Bot API.....	30
2.4. SQLite.....	31
Висновки до другого розділу	32
РОЗДІЛ 3. РОЗРОБКА ПРОДУКТУ	33
3.1. Реєстрація Телеграм-боту	33
3.2. Тестування програми написаної на мові високого рівня програмування Python.....	34
Висновки до третього розділу.....	52
ВИСНОВКИ.....	53
СПИСОК БІБЛОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	54

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

HTML – Hyper Text Markup Language

VS Code – Visual Studio Code

БД - база даних

API – Application programming interface

SQLite - Structured Query Language

ВСТУП

Месенджер Телеграм відрізняється від своїх конкурентів, таких як WhatsApp та Co, завдяки можливості створення ботів та каналів. Безкоштовно доступні інтерфейси програмування програм (API) полегшують життя, збиранням даних або автоматизацією речей. Таким чином, Телеграм-бот - це хороший спосіб доставити інформацію майже з усіх можливих областей, коли це потрібно.

Телеграм-бот - це програма, створена за допомогою штучного інтелекту, яка може виконувати безліч функцій: надсилати відповідну інформацію про погоду чи корисні статті новин, планувати нагадування, відтворювати мелодії, створювати списки справ та багато іншого.

РОЗДІЛ 1

ТЕЛЕГРАМ-БОТ, ЯК СУЧАСНИЙ СПОСІБ ПОШИРЕННЯ ІНФОРМАЦІЇ

1.1. Особливість використання месенджера Телеграм

Телеграм - це мультиплатформна служба обміну повідомленнями, заснована російським підприємцем Павлом Дуровим. Вперше він вийшов на IOS та Android наприкінці 2013 року. У січні 2021 року було оголошено, що Телеграм має 500 мільйонів активних користувачів щомісяця, які щодня відправляють десятки мільярдів повідомлень. Код програмування клієнтів Телеграму переважно вільно доступний. Для сервера пропонується відкритий інтерфейс програмування (API).



Рис 1.1. Логотип Телеграму

Кафедра КІТ (47)				НАУ 21 04 86 000 ПЗ			
Виконав	Гасанова А.М.			Телеграм-бот, як сучасний засіб поширення інформації	Літера	Аркуш	Аркушів
Керівник	Зіатдінов Ю.К.					8	54
Консульт.					411 122		
Н-котрол.	Шевченко О.П.						
Зав. каф.	Савченко А.С.						

Telegram поєднує швидкість WhatsApp із швидкоплинністю Snapchat. Як і WhatsApp, Telegram має можливість показувати статус партнера по спілкуванню в Інтернеті та передавати тексти, фотографії, відео, аудіофайли, інформацію про місцезнаходження, контакти та документи. Але є також можливість надати цю інформацію із датою закінчення часу і таким чином її автоматично видалити.

Основна функціональність Телеграму така ж, як і у більшості інших програм обміну повідомленнями: можливість надсилати повідомлення іншим користувачам Телеграму, створювати групові розмови, дзвонити контактам та надсилати файли та наліпки. Однак є кілька специфічних функцій, які змушують його працювати не так, як інші програми чату.

Основний список функцій Телеграму можуть перетинатися з іншими програмами, але між ним та його конкурентами є багато конкретних відмінностей. Зручність користування додатком Telegram наведена у наступних абзацах.

Telegram унікальний, він сфокусований на конфіденційності, шифруванні та API з відкритим кодом. Існує незліченна кількість неофіційних клієнтів, які підходять до офіційних програм Телеграму та веб-інтерфейсу. Це також дозволяє декільком пристроям використовувати один і той же обліковий запис (перевірено за допомогою SMS) та декілька облікових записів на одному пристрої. Усі заходи, включаючи чати, групи та засоби масової інформації, якими діляться учасники, шифруються. Це означає, що їх не видно без попередньої розшифровки.

Telegram пропонує наскрізне шифрування між партнерами в чаті за допомогою функції «секретні чати». Це те, що заважає людям поза двосторонньою розмовою - будь то компанія, уряд, хакери чи хтось інший - бачити надіслане. Крім того, користувачі не можуть пересилати або знімати скріншоти повідомлень у «секретних чатах», а повідомлення можна запрограмувати на самознищення. Видалення повідомлення також видаляє його для всіх у службі, і користувачі мають можливість видалити не лише власні повідомлення, але й повідомлення інших користувачів.

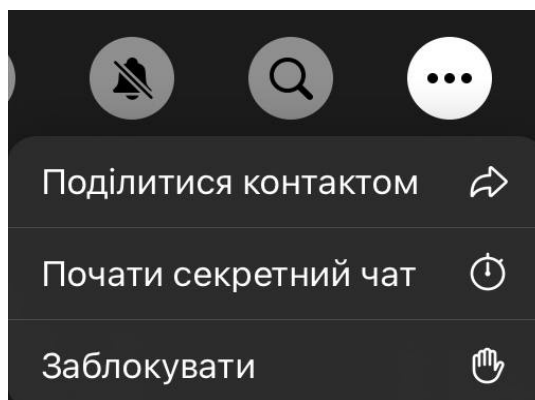


Рис.1.2. Кнопка «секретний чат»

Послуга пропонує наскрізне шифрування для голосових дзвінків та необов'язкові наскрізні зашифровані секретні чати між двома онлайн-користувачами, але не для груп чи каналів. Він має режим самознищення, що дозволяє всім чатам працювати бездоганно, Він працює з дуже стабільною та надійною роботою, і повідомлення доставляються без затримки, тому це одна з найшвидших доступних програм обміну повідомленнями. Це захищає ваші повідомлення від хакерських атак. Ви можете підключитися з більшості віддалених місць, ви можете координувати групи до 5000 членів, а також синхронізувати свої чати на всіх своїх пристроях.

Вищезазначені секретні чати - це місце, де можна взяти участь у наскрізних зашифрованих повідомленнях із контактом. Звичайно, хтось може сфотографувати екран за допомогою іншого пристрою, але це все одно не рекомендується, і це підкріплюється ще однією функцією: таймерами самознищення.

Telegram дозволяє встановити таймери самознищення, щоб назавжди їх видалити. Після отримання повідомлення воно залишається в чаті на заздалегідь визначений період - можна вибрати час від однієї секунди до одного тижня.

Telegram - це все про конфіденційність та безпеку. Причиною цього є розширене використання Telegram-хмари. Він зберігає всі ваші повідомлення та фотографії на захищеному сервері. Це означає, що ви можете отримати до них доступ з будь-якого підключеного пристрою, що робить Telegram набагато зручнішим для різних платформ. [1]

Telegram надає необмежену пам'ять. Це означає, що всі текстові повідомлення, медіа-файли зображень та документи будуть збережені в їхній хмарі. Можливо вийти та увійти будь-яку кількість разів із будь-якої кількості пристроїв одночасно, не втрачаючи даних, не потрібно турбуватися про резервне копіювання та відновлення. Можливо переглянути активні сеанси та побачити, на яких пристроях ввійшли зараз. Тим самим можна будь-коли завантажити будь-який файл де завгодно. На сьогодні можливо завантажити один файл максимального розміру до 2 Гб. Немає обмежень на загальну кількість файлів, які можна завантажити.

Ще однією функцією безпеки є теги користувачів. Замість того, щоб надавати людям свій номер телефону, ви можете просто дати свій тег користувача. Це дає кращий контроль над наявною інформацією та тим, як люди можуть зв'язатися з вами в майбутньому.

Telegram - це безкоштовний додаток без реклами, який у майбутньому фінансуватиметься за рахунок пожертв. Послуга доступна для всіх поширених платформ і конкурує з іншими мобільними месенджерами. Компанія почне застосовувати нові методи монетизації у 2021 році, але основна діяльність чату залишиться безкоштовною для всіх.

Ці нові методи монетизації включають рекламу на загальнодоступних каналах "один до багатьох", преміум-функції для бізнес-команд та досвідчених користувачів, а також платні пакети наліпок. Частина цих коштів повернеться користувачам у вигляді покращеної видимості каналів або платежів для творців унікальних пакетів наліпок.

Telegram поставляється з деякими параметрами налаштування, відсутніми у багатьох конкурентів, де можна вибрати домінуючий колір додатка, спосіб відкриття посилань, незалежно від того, чи відображається анімація в інтерфейсі тощо. Існує також інтеграція чат-ботів, де можливо використовувати ботів для покращення досвіду. Можна створювати власні теми, ботів та повнофункціональні програми за допомогою Telegram API.

Папки чату - одна з найкращих хитрощів Телеграму, яку повинні знати та використовувати з максимальною перевагою. Стеження за такою кількістю різних

типів каналів та груп у Телеграмі, не кажучи вже про окремі чати, які входять до складу. Наявність усіх цих чатів на одній панелі робить речі набагато складнішими для того, щоб розібратися та зрозуміти. Тож для вирішення цього питання Телеграм запропонував щось під назвою «Папки чату» (рис.1.3, 1.4). Це дозволяє класифікувати чати за різними мітками, і Телеграм сортує речі за вас. Як результат, Телеграм стає надзвичайно чистим і набагато простішим в управлінні.

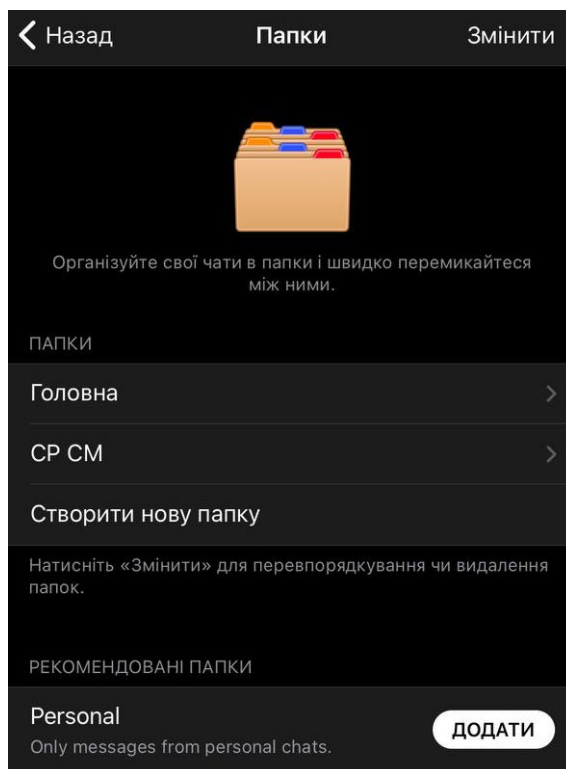


Рис.1.3. Створення «папки чату»

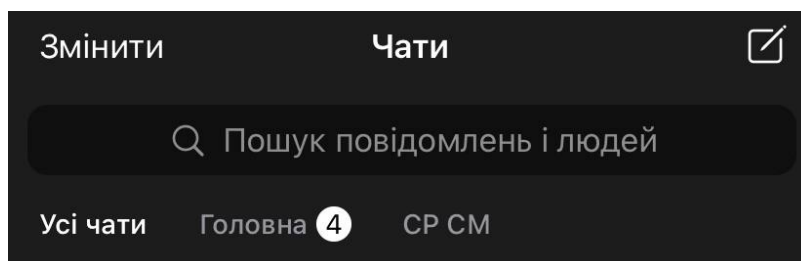


Рис.1.4. Вигляд «папки чату»

Версія Телеграму для ПК або ще не повністю розроблена, або її розвиток зупинено, оскільки в ньому немає секретних чатів. Ця функція доступна виключно

на мобільні пристрої.

Минулого року Телеграм представив унікальну функцію, яка дозволяє виявляти користувачів і групи Телеграм залежно від вашого місцезнаходження (рис.1.5). Це дуже зручно, коли є можливість додати когось до контакту в Телеграмі, не повідомляючи номер мобільного телефону. Це зручно для конференцій, зборів, тощо.

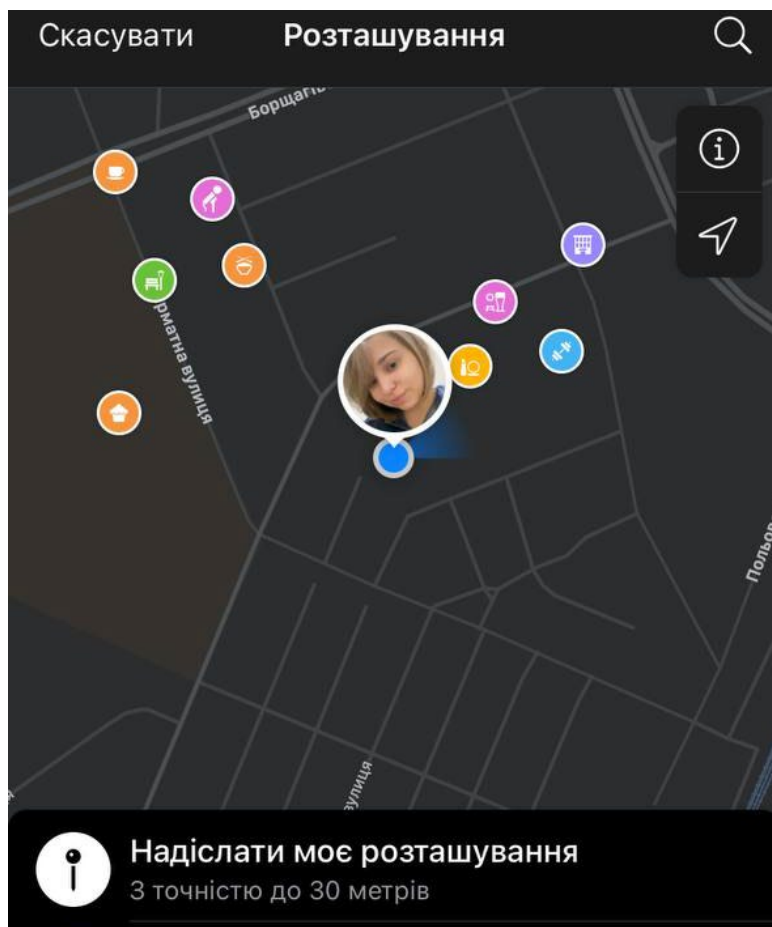


Рис.1.5. Місцезнаходження в додатку Телеграм

Кнопка редагування (рис.1.6) допомагає редагувати уже надіслане повідомлення .

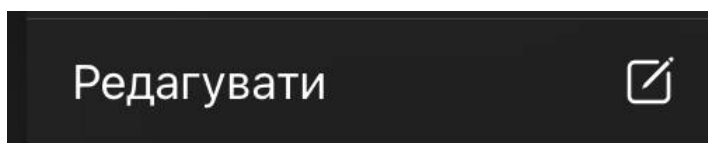


Рис.1.6. Кнопка «редагувати»

Користувачі можуть вибрати, стискати зображення та відео або надсилати нестиснуту версію. Наприклад, можливо надіслати вихідне зображення у форматі jpeg або як вихідне, або стиснене. Щоб зберегти якість, виберіть варіант надіслати без стиснення. [2]

Канал чимось схожий на групи, які можуть вмістити необмежену кількість учасників. Адміністратор каналу може вирішити, хто може розміщувати повідомлення, а інші учасники можуть переглядати публікації.

Telegram доступний для Android, iOS, Windows Phone, Windows PC, Mac OS, OS Linux і навіть через будь-який браузер.

Telegram пропонує надзвичайно швидку послугу обміну повідомленнями, вона має миттєву підтримку, оскільки підтримка в режимі реального часу доступна, щоб користувачі могли зв'язуватися з компанією з будь-яких питань, а в налаштуваннях є опція «Задати запитання», щоб зробити запити.

Тепер користувачі можуть надсилати фотографії, аудіо та відео як частину своїх повідомлень, це дуже надійно, і не втрачається величезна кількість даних, оскільки в чаті потрібні лише байти даних.

Telegram може подбати про вашу безпеку даних. Як показують усі останні події, він не видасть ваш IP або номер телефону, Telegram все ще може видавати ваші дані, якщо є офіційне рішення суду, і лише в європейських країнах.

Telegram постачається з каналами, де людина може під'єднуватися до інформації, не перебуваючи в груповому чаті. Ця функція недоступна в інших додатках. Це дуже важлива функція, яка робить Telegram різною та чіткою, канали можуть розміщувати інформацію без будь-яких коментарів та аналізу.

Telegram дозволяє передавати важкі файли на своїй платформі. Це робить Telegram хорошим місцем для обміну файлами. Є більше шансів познайомитись з новими людьми за допомогою групових чатів. Групові чати не схожі на груповий чат Whatsapp, оскільки ви можете приєднатися просто перейшовши за посиланням.

Прив'язуючись до телефонного номера, даючи можливість дізнатися всю інформацію про ваші контакти та надіслати їх на сервер. Основним недоліком є не

просто показ ваших контактів, майже будь-яка служба державної безпеки може додати всі телефонні номери у країні до своїх контактів, отже, можливо дізнатись, які люди зареєстровані в соціальній мережі, і коли вони востаннє її відвідували.

Коли повідомлення надсилаються за допомогою звичайного, а не секретного чату, то формально вони не зашифровані, не слід думати, що ваш провайдер може легко прочитати будь-яке з ваших повідомлень, оскільки всі діалоги мають закритий протокол, хоча, усі повідомлення знаходяться на спеціальному сервері. Коли повторно вводиться рахунок, вони переходять туди.

1.2. Переваги та недоліки месенджера Телеграм

Переваги додатку Телеграм:

- Висока швидкість роботи - не дивлячись на розташування серверів в різних частинах планети, доставка повідомлення займе частки секунди;
- Зрозумілий класичний інтерфейс програми, який можна оформити в різних стилях: вибрати з готових варіантів або застосувати власний;
- Автоматичне додавання списку користувачів сервісу з телефонної книги.
- Просте початок роботи. Для реєстрації потрібен тільки номер телефону.
- Всі функції і можливості безкоштовні;
- Інноваційність - сервіс постійно оновлюється. Його можливості розширюються як з урахуванням пропозицій конкурентів, так і власними розробками, які не мають аналогів;
- Високий рівень безпеки і захисту від злому аккаунта і листування.
- Безліміт на все: кількість відправлених в день повідомлень, обсяги завантажуваних в хмару файлів, часу використання;
- Сумісність з усіма основними операційними системами, включаючи мобільні.

Недоліки додатку Телеграм

- З основних недоліків, які виявлені і поки не усунуто, одним з головних

можна вважати те, що комунікація зі службою технічної підтримки ведеться тільки англійською мовою. До недавнього часу інтерфейс не мав підтримки російської мови;

- Виняткова захист листування і відмова надавати владі і правоохоронним органам змісту бесід, служить причиною того, що через цей комунікатор відбувається спілкування терористичних угруповань, збут наркотиків та інші дії, заборонені законодавчо;
- Неможливість проконтролювати і зупинити розсилку незаконної інформації, призводить до блокування роботи програми в деяких країнах;
- Незважаючи на високі заходи безпеки, частина інформації про користувача можна отримати - його номер телефону. Це не сприяє повній анонімності профілю.

1.3. Сутність Телеграм-боту

Телеграм-бот - це програма, створена за допомогою штучного інтелекту, яка може виконувати безліч функцій: надсилати відповідну інформацію про погоду чи корисні статті новин, планувати нагадування, відтворювати мелодії, створювати списки справ та багато іншого.

Створення Телеграм-боту почалось після того, як Телеграм оголосив про новий API ботів Телеграм у 2015 році. Він дозволяє третім сторонам створювати ботів, які використовуватимуть цей додаток для обміну повідомленнями як основний інтерфейс. Список ботів Телеграм довгий: багато розробників експериментували із наданим API для створення унікальних ботів для різних цілей.

Розвиток Телеграм не стоїть на місці, технологія бот-шеринга розвивається набуває все більшої популярності. Це найбільш зручний і масовий спосіб виконувати широкий спектр завдань - від піару до управління домашньою технікою. Уже зараз коло завдань, нездійсненний для бота стрімко звужується, а в

найближчому майбутньому таких «неможливих» доручень не стане зовсім.

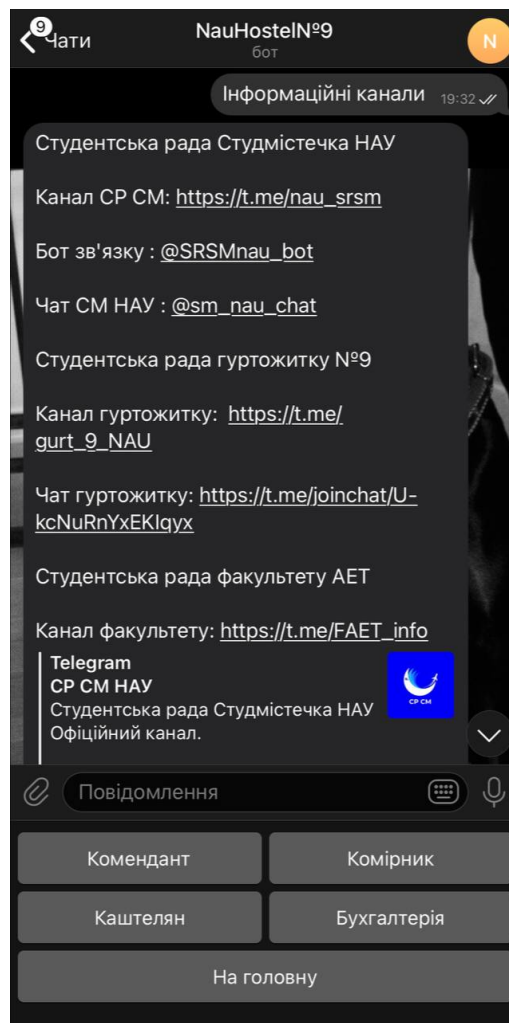


Рис.1.7. Приклад Телеграм-бота

Підсумовуючи, Телеграм-боти - це програми, призначені для виконання певних функцій, дотримання інструкцій та взаємодії з користувачами. Боти працюють у середовищі Телеграм і не потребують додаткової процедури встановлення.

За допомогою ботів люди можуть покращити функціональність Телеграму. Наприклад, можна щодня надсилати сповіщення своїм друзям, встановлювати собі нагадування, перекладати повідомлення, отримувати попередження про погоду, грати в ігри тощо.

Телеграм-бот може у визначених вами середовищах комфортно інформувати

та реагувати. Телеграм може не тільки забезпечити спілкування між людьми, але за допомогою Телеграм-боту можна забезпечити спілкування між людьми та програмованими машинами. Таким чином, ми можемо пересилати повідомлення на кілька пристроїв одним клацанням миші або отримувати їх.

Телеграм-бот дозволяють легко надсилати новини, оновлення та рекламний контент як автоматизовані інформаційні бюлетені. Тому, Телеграм-боти ідеально підходять для автоматичного просування через існуючі системи.

Крім того, компанії можуть створювати чат-боти, які, наприклад, попередньо кваліфікують запити (чат-боти підтримки першого рівня), автоматично відповідають на повторювані запитання (FAQ) або надають персональну інформацію для користувачів за допомогою підключених баз даних.

Бот може виступати розумною газетою та надсилати відповідний вміст відразу після його публікації. Завдяки інтеграції зовнішніх служб, таких як YouTube або бот GIF, інформаційний бюлетень можна збагатити різноманітним вмістом. Бот може працювати як віртуальна вітрина, пропонувати платні послуги або надсилати сповіщення, прогнози погоди, форматування та переклади.

Телеграм-бот може в основному робити все, що робить людина-чат-партнер. Автоматично або на запит він може надіслати вам таку інформацію:

- текстові повідомлення;
- зображення;
- відео;
- файли будь-якого іншого виду.

Важливою функцією Телеграм-боту є можливість виконувати команди в чаті, які потім безпосередньо запускають дії або запитують інформацію. Наприклад, можна надіслати боту команду “/help”, яка потім виводить команди, можливі для цього бота в чаті, як текстовий зворотний зв’язок. Це може бути такий список команд:

- /status;
- /temperature;

- /last alarm;
- /stop.

Телеграм зберігає всі дані, тобто всі чати, а також боти, зашифровані в хмарі. Таким чином, зовнішнє резервне копіювання даних Телеграму не є абсолютно необхідним, і всі особисті налаштування доступні для вхідних користувачів на різних платформах у будь-який час і скрізь. Однак конфіденційні дані та команди, до яких бот потім звертається зовні, можуть бути доступними поза хмарою, наприклад на серверах місцевих компаній у власних базах даних.

Звичайні Телеграм-боти не є приватними. Будь-хто може знайти і використовувати їх. Однак Телеграм-боти також можуть бути доступними для певних користувачів, лише якщо встановлений окремий канал зв'язку з ботом. Цей канал є "приватною" групою з ботом, до якої належить лише ви, схвалені вами користувачі та фактичний бот. Таким чином, Телеграм-бот може користуватися лише контрольована вами група користувачів.

Все ж таки Телеграм-боти є безпечні для користувача. Теоретично боти можуть запропонувати вийти на зовнішні ресурси за посиланням, де буде чекати небезпечний контент або віруси. І це не буде відрізнятися від того, що випадково або спеціально можуть надіслати в повідомленні інші користувачі.

Однак в рамках самого Телеграм-бота, він не може зробити нічого поганого, тому що по своїй суті це ті ж акаунти користувачів, тільки управляються алгоритмічно. Як і звичайного користувача, набридлого бота можна видалити і заблокувати.

Важливо, що бот не виступає ініціатором бесіди - це ви перший відправляєте йому запит, тому без вашого дозволу бот не може вас потурбувати. Крім того, навіть бот, доданий до вас в групу, за замовчуванням налаштований так, щоб не «бачити» ваших переговорів. Тобто його творець не зможе шпигувати і перехоплювати повідомлення з чату, куди доданий бот.

Якщо розглядати Телеграм-бота в контексті джерела даних, фактичні дані зазвичай залишаються на локальних серверах. Там бот отримує доступ до необхідних наборів даних, а потім готує їх до виведення повідомлень за запитом на

кінцевому пристрої. У хмарі на серверах постійно знаходиться лише фактична структура програми.

Фактична програма Телеграм-бот із відповідними даними зберігається на власних комп'ютерах та базах даних. Тут безпека програми залежить від адміністратора локального середовища ПК на сайті. Бот доставляє лише відповідні вихідні повідомлення, визначені в додатку, на відповідні кінцеві пристрої. Потім лише ці вихідні повідомлення розміщуються у хмарі Телеграму.

Щоб створити Телеграм-бота самостійно, потрібно лише встановити додаток Телеграм на пристрій і увійти в нього. Можливо знайти додаток Телеграм для мобільних пристроїв у магазинах програм (наприклад, Google Play Store або Apple App Store), а також для інших пристроїв, наприклад на офіційному веб-сайті.

Справжнє створення власного бота, як правило, дуже просте. Вас підтримує віртуальний користувач "Botfather", який є центральним інструментом розробки Телеграм-ботів.

Телеграм-бот завжди можна чітко ідентифікувати за допомогою суфікса "бот". Наприклад, "@weatherbot", "@ARD_tagesschau_Bot" або "@mein_erster_test_bot".

1.4. Види Телеграм-ботів

Самі ж боти діляться на кілька напрямків:

Чат-боти. Являють собою найпростіший чат, що імітує спілкування на задану користувачем тематику.

Чат-боти допомагають скоротити видання на клієнтському сервісі - вони можуть задати питання та відповісти на них, знайти та підбирати необхідний продукт, проінформувати про діючі акції. Зазвичай їх додають у чат-сайт або в месенджери.

Чатові боти часто рекламуються як революція у взаємодії користувачів з технологіями та бізнесом. Вони мають досить простий інтерфейс у порівнянні з традиційними програмами, оскільки вони вимагають від користувачів лише спілкування, а чат-боти повинні розуміти і робити все, що від них вимагає

користувач, принаймні теоретично. [3]

Багато галузей перекладають обслуговування клієнтів на системи чат-ботів. Це пов'язано з величезним падінням вартості порівняно з реальними людьми, а також завдяки надійності та постійній доступності. Чат-боти забезпечують певний рівень підтримки користувачів без істотних додаткових витрат. Сьогодні чат-боти використовуються в багатьох сценаріях, починаючи від таких легких завдань, як відображення даних про час і погоду, і закінчуючи більш складними операціями, такими як елементарна медична діагностика та спілкування / підтримка клієнтів.



Рис.1.8. Статистичні дані застосування чат-ботів

Боти-інформатори. Окремий вид ботів, головна мета яких - інформування користувача про ті чи інші події (новини, заходи, публікації тощо).

Боти-інформатори допомагають доносити останні події різних категорій до користувача.

Ігрові боти. Боти, в яких можна пограти в різні ігри. Здебільшого, це текстові версії різних ігор.

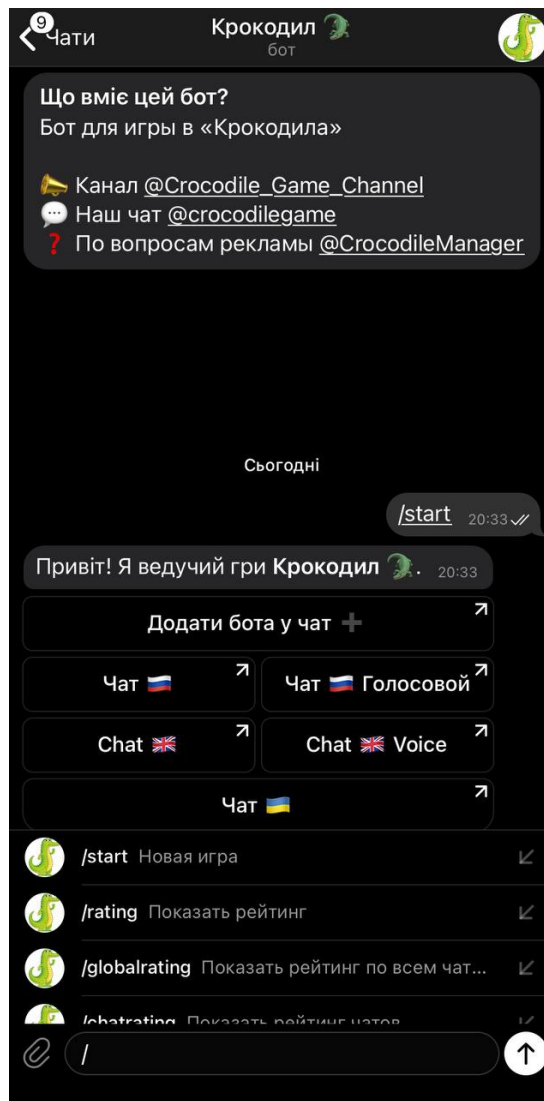


Рис.1.9. Приклад ігрового бота

Боти-асистенти. Боти, розроблені різними онлайн-сервісами як доповнення до основної веб-версії.

Насправді, чіткого поділу немає, тому що деякі боти містять відразу кілька механік і успішно виконують безліч призначених для користувача завдань. З їх допомогою можна перекладати, навчатися, тестувати, шукати інформацію, грати в ігри і навіть користуватися іншими сервісами і взаємодіяти з речами, що мають вихід в глобальну мережу. Усі боти в Телеграма є безкоштовними, але в 2017 році Павлом Дуровим була анонсована можливість налаштувати і платне користування такими програмами.

Завдяки цьому, боти стали кишеньковими помічниками, користуватися якими

можна навіть не покидаючи месенджер. Вони надають можливість вирішувати елементарні завдання за допомогою миттєвих команд, усі ці програми не потребують встановлення і не займають окреме місце в пам'яті девайса.

1.5. Взаємодія між користувачем і ботом

Взаємодія між користувачем і ботом виглядає наступним чином:

Користувач бота віддає йому команду -> Бот передає команду на сервер -> Програма на сервері обробляє отриманий від бота запит -> Сервер віддає відповідь боту -> Бот виводить відповідь на вікні керування користувачеві.

І цей цикл повторюється раз за разом, коли ви натискаєте на кнопки і взаємодієте з будь-яким телеграм-ботом.

Спілкування проходить з серверами за допомогою простого HTTPS-інтерфейсу, який є спрощеною версією API Телеграму. Інакше цей інтерфейс можна назвати програмним каталогом або бот-алгоритмом. Детальніше про те, як працює бот в телеграм можна дізнатися, ставши розробником програмного забезпечення. Нові бот-утиліти створюються за допомогою спеціальної утиліти @BotFather, який значно спрощує процес розробки.

Користувачі можуть взаємодіяти з програмами, відправляючи їм повідомлення, команди і запити. У їх числі можуть бути зазначені ключові слова, голосові повідомлення, геолокації. Для початку спілкування з ботом є два способи:

Відправляти команди, відкриваючи чат з ними або додаючи їх в групи. Це корисно для чат-ботів або новинних ботів.

Відправляти запити безпосередньо з поля введення, ввівши ім'я користувача @bot і запит. Це дозволяє відправляти контент з вбудованих ботів безпосередньо в будь-який чат, групу або канал.

На сьогоднішній день навряд чи знайдеться бот, який викличе труднощі навіть у самих недосвідчених користувачів програми. Вони стають простіше, доступніше і швидше.

1.6. Python, як мова для Телеграм-боту

Python - інтерпретована, об'єктно-орієнтована мова програмування високого рівня з динамічною семантикою. Високорівневі вбудовані структури даних у поєднанні з динамічним набором тексту та динамічним прив'язуванням роблять його дуже привабливим для швидкої розробки додатків, а також для використання в якості мови сценаріїв або клею для з'єднання існуючих компонентів. Простий, легкий у вивченні синтаксис Python підкреслює читабельність і, отже, зменшує витрати на обслуговування програми. Python підтримує модулі та пакети, що заохочує модульність програми та повторне використання коду. Інтерпретатор Python та велика стандартна бібліотека доступні у вихідній або двійковій формі безкоштовно для всіх основних платформ і можуть вільно розповсюджуватися.

Часто програмісти закохуються в Python через підвищену продуктивність, яку він забезпечує. Оскільки кроку компіляції не існує, цикл редагування-тестування-налагодження наймовірно швидкий. Налагодження програми на Python дуже просто: помилка або неправильний ввід ніколи не спричинить помилку сегментації. Натомість, коли інтерпретатор виявляє помилку, виникає виняток. Коли програма не вловлює виняток, інтерпретатор друкує трасування стека. Налагоджувач рівня джерела дозволяє перевіряти локальні та глобальні змінні, оцінювати довільні вирази, встановлювати точки зупинки. Налагоджувач написаний на самому Python, що свідчить про інтроспективну силу Python. З іншого боку, часто найшвидшим способом налагодження програми є додавання кількох операторів друку до джерела: швидкий цикл редагування-тестування-налагодження робить цей простий підхід дуже ефективним.

Python - єдина мова програмування з великим набором бібліотек підтримки та розумних об'єктно-орієнтованих конструкцій, що підвищує продуктивність програміста до десяти разів. [4]

Ще одним фактором, який сприяє підвищенню ефективності, швидкості та продуктивності розробників Python, є його надійна функція інтеграції процесів. Це дає можливість модульного тестування та вдосконаленого управління, що не тільки

додає темпу розробки додатків, але й полегшує розробникам програмного забезпечення створення багатогранних масштабованих програм.

Саме ці переваги роблять Python найкращою мовою для написання Телеграм-ботів.

Висновки до першого розділу

Отже, Телеграм - це популярний крос-платформний додаток для обміну повідомленнями, який широко використовується, оскільки він пропонує деякі розширені функції конфіденційності та шифрування, а також підтримку функцій великого групового чату. Головна його перевага – це безпека. Телеграм пропонує наскрізне шифрування між партнерами в чаті за допомогою функції «секретні чати».

Також, однією з переваг Телеграму можна виділити створення Телеграм-ботів, які покращують життя користувачу. Телеграм-боти дозволяють легко надсилати новини, оновлення та рекламний контент як автоматизовані інформаційні бюлетені. Тому, Телеграм-боти ідеально підходять для автоматичного просування через існуючі системи.

Зроблені висновки, що Python, як мова високого рівня програмування найзручніша для написання Телеграм-ботів.

РОЗДІЛ 2

ТЕХНІЧНІ ЗАСОБИ ДЛЯ РЕАЛІЗАЦІЇ ПРОДУКТУ

2.1. Утиліта Botfather

Створення нового бота відбувається за допомогою команди /newbot. BotFather запитає у ім'я та ім'я користувача, а потім створить маркер авторизації для вашого нового бота. Ім'я бота відображається в контактних даних та інших місцях.

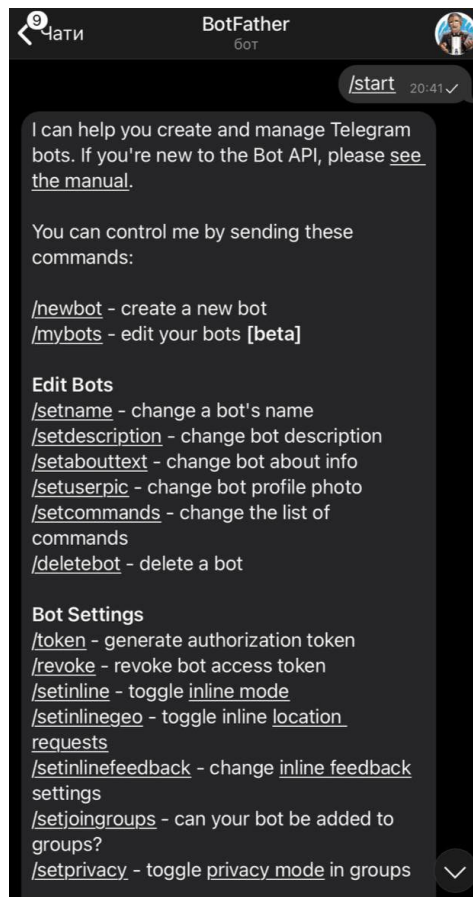


Рис.2.1. Утиліта Botfather

Кафедра КІТ (47)				НАУ 21 04 86 000 ПЗ			
Виконав	Гасанова А.М.			Технічні засоби для реалізації продукту	<i>Літера</i>	<i>Аркуш</i>	<i>Аркушів</i>
Керівник	Зіатдінов Ю.К.					26	54
Консульт.					411 122		
Н-котрол.	Шевченко О.П.						
Зав. каф.	Савченко А.С.						

Ім'я користувача - це коротка назва, яка використовуватиметься у згадках та посиланнях t.me. Імена користувачів мають довжину 5–32 символи та не чутливі до регістру, але можуть містити лише латинські символи, цифри та підкреслення. Ім'я користувача вашого бота має закінчуватися на "бот", напр. 'tetris_bot' або 'TetrisBot'.

Маркер - це рядок, який необхідний для авторизації бота та надсилання запитів до API бота. Маркер зберігається у безпеці, ним може користуватися кожен, хто керує вашим ботом.

Якщо існуючий маркер порушено або його з якоїсь причини втратили, скористайтеся командою / token, щоб створити новий.

Команди Botfather:

- / mybots - повертає список ботів із зручними елементами управління для редагування їх налаштувань;
- / mygames - робить те саме для ваших ігор;
- / setname - змінить ім'я бота;
- / setdescription - змінить опис бота, короткий текст довжиною до 512 символів, що описує бота. Користувачі побачать цей текст на початку розмови з ботом під назвою "Що може зробити цей бот?";
- / setabouttext - змінить інформацію про бота про інформацію, ще короткий текст довжиною до 120 символів. Користувачі побачать цей текст на сторінці профілю бота. Коли вони діляться ботом з кимось, цей текст надсилається разом із посиланням;
- / setuserpic - змінить зображення профілю бота;
- / setcommands - змінить список команд, що підтримуються вашим ботом. Користувачі бачитимуть ці команди як пропозиції, коли вони вводять / у чаті з вашим ботом. Кожна команда має назву (повинна починатися з косої риски «/»), буквено-цифрові та підкреслення, не більше 32 символів, без урахування регістру), параметри та текстовий опис. Користувачі бачитимуть список команд кожного разу, коли вони наберуть '/' у розмові з вашим ботом;

- / deletebot – видалить бота та звільнить його ім'я користувача;
- / setinline - перемикання вбудованого режиму для вашого бота;
- / setinlinegeo - запитувати дані про місцезнаходження для надання вбудованих результатів на основі розташування;
- / setjoininggroups - перемикає, чи можна бота додавати до груп чи ні. Будь-який бот повинен мати можливість обробляти приватні повідомлення, але якщо ваш бот не був розроблений для роботи в групах, можливо вимкнути це;
- / setprivacy - встановить, які повідомлення бот буде отримувати при додаванні до групи. При вимкненому режимі конфіденційності бот отримуватиме всі повідомлення. Рекомендуємо залишити ввімкнений режим конфіденційності. Вам потрібно буде повторно додати бота до існуючих груп, щоб ця зміна набрала чинності;
- / newgame - створити нову гру;
- / listgames - отримати список своїх ігор;
- / editgame - редагування гри;
- / deletegame - видалити існуючу гру.

Мільйони вибирають Телеграм за швидкість. Щоб залишатися конкурентоспроможним у цьому середовищі, бот також повинен бути чуйним. Щоб допомогти розробникам підтримувати своїх ботів у формі, Botfather надсилатиме сповіщення про стан, якщо побачить, що щось не так.

За замовчуванням ви отримуватимете лише одне сповіщення на кожного бота на годину. Кожне попередження має такі кнопки:

- **Виправлено.** Використовуйте це, якщо виявляється проблема з ботом і виправили її. Якщо натиснути кнопку виправлення, відновиться надсилання сповіщень регулярним способом, щоб можна побачити, чи спрацювало виправлення протягом 5-10 хвилин, замість того, щоб чекати годину;
- **Підтримка.** Використовуємо це, щоб відкрити чат з @BotSupport, якщо

немає проблем із ботом;

- Вимкнення звуку протягом 8 годин. Використовуємо це, якщо на даний момент не можна виправити свого бота. Це вимкне всі сповіщення для відповідного бота на вказаний проміжок часу. Ця процедура не є рекомендованою.

2.2. Середина розробки

Visual Studio Code - повнофункціональний редактор коду (рис.2.2) , доступний на Windows, Linux і Mac OS X. VS Code є розширюваним open-source редактором, який можна налаштувати під будь-яке завдання. Як і Atom, VS Code побудований на Electron, тому у нього є ті ж переваги і недоліки.

Додати підтримку Python в VS Code простіше простого - знайдіть за запитом «Python» в Marketplace, натисніть «Install» і запустіть редактор, якщо буде потрібно. VS Code автоматично визначить інтерпретатор Python і встановлені бібліотеки.

Редактор містить вбудований зневаджувач, інструменти для роботи з Git і засоби рефакторингу, навігації по коду, автодоповнення типових конструкцій і контекстні підказки. Продукт підтримує розробку для платформ ASP.NET і Node.js, і позиціюється як легковагове рішення, що дозволяє обійтися без повного інтегрованого середовища розробки. Серед підтримуваних мов і технологій: JavaScript, C++, C#, TypeScript, jade, PHP, Python, XML, Batch, F#, Docker File, Coffee Script, Java, HandleBars, R, Objective-C, PowerShell, Luna, Visual Basic, Markdown, JSON, HTML, CSS, LESS і SASS, Naxe. [5]

Працювати з Python у Visual Studio Code за допомогою розширення Microsoft Python - це просто, весело та продуктивно. Розширення робить VS Code чудовим редактором Python і працює в будь-якій операційній системі з різноманітними інтерпретаторами Python. Він використовує всі можливості VS Code для забезпечення автоматичного заповнення та IntelliSense, підключення, налагодження та модульного тестування, а також можливість легкого переключення між середовищами Python, включаючи віртуальне середовище та середовище conda.

Розширення Python підтримує тестування з модулями unittest, pytest та nose test.

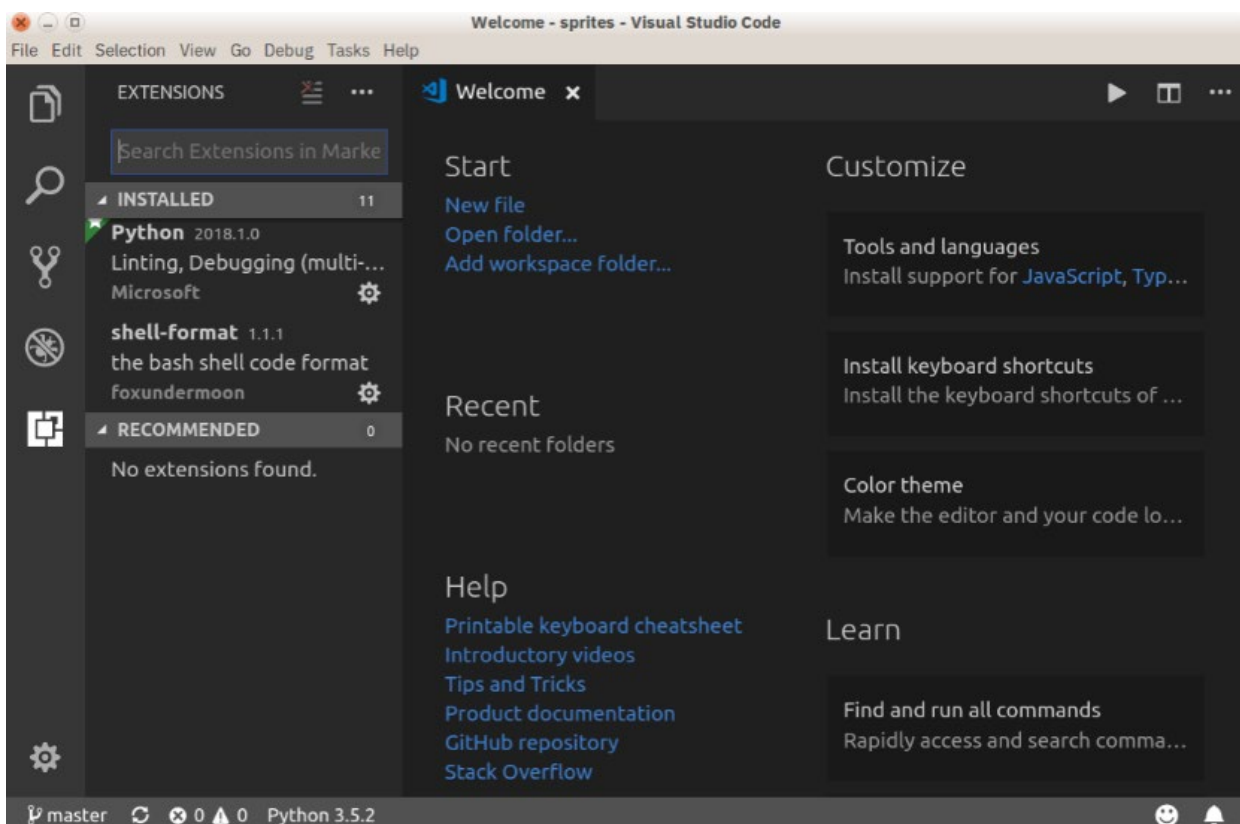


Рис.2.2. Вигляд VS Code

Переваги: завдяки Electron, VS Code доступний на всіх платформах з дивно великою функціональністю. Крім того, вихідні коди можна знайти у відкритому доступі.

Недоліки: раз замішаний Electron, значить VS Code НЕ нативний додаток.

2.3. Telegram Bot API

Цей API дозволяє підключати ботів до системи. Боти Телеграм - це спеціальні облікові записи, для налаштування яких не потрібен додатковий номер телефону. Ці облікові записи служать інтерфейсом для коду, який працює десь на вашому сервері.

Щоб використовувати це, не потрібно нічого знати про те, як працює

протокол шифрування MTProto - посередницький сервер буде обробляти для все шифрування та зв'язок з API Телеграм. Спілкування із цим сервером через простий HTTPS-інтерфейс, який пропонує спрощену версію API Телеграм. [6]

Vot API - це інтерфейс на основі HTTP, створений для розробників.

Ця бібліотека забезпечує чистий інтерфейс Python для API Telegram Bot. Він сумісний з версіями Python 3.6+. РТВ також може працювати на PyPy (інтерпретатор мови програмування), хоча раніше було багато проблем. Отже, PyPy офіційно не підтримується.

На додаток до чистої реалізації API, ця бібліотека має ряд класів високого рівня, що роблять розробку ботів простою та зрозумілою. Ці класи містяться в підмодулі telegram.ext.

2.4. SQLite

SQLite — полегшена реляційна система керування базами даних (рис.2.3). Втілена у вигляді бібліотеки, де реалізовано багато зі стандарту SQL-92. Сирцевий код SQLite поширюється як суспільне надбання (англ. public domain), тобто може використовуватися без обмежень та безоплатно з будь-якою метою. [7]

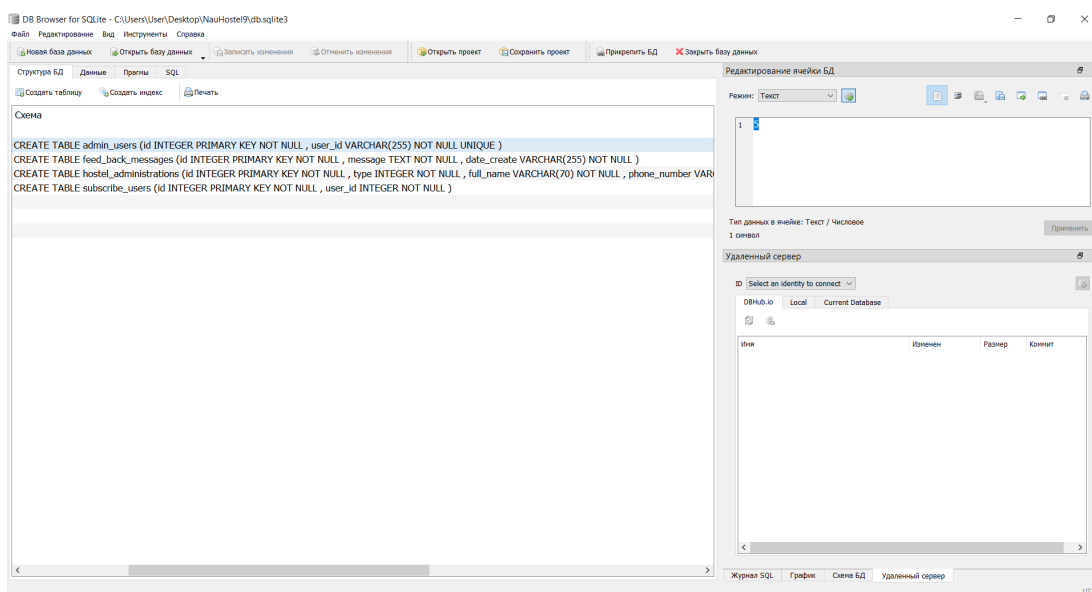


Рис.2.3. Вигляд SQLite

Висновки до другого розділу

Отже в даному розділі описано програмне забезпечення, що використовується для реалізації проекту.

BotFather - найпростіший спосіб для реєстрації, налаштування та управління іншими Telegram-ботами. Робота з ним проста і не вимагає специфічних навичок. За допомогою BotFather можна зареєструвати необмежену кількість нових спамерських пошукових роботів. Єдиною умовою для реєстрації нового бота - є його унікальним ім'ям користувача.

Visual Studio Code - повнофункціональний редактор коду, доступний на Windows, Linux і Mac OS X. VS Code є розширюваним open-source редактором, який можна налаштувати під будь-яке завдання.

SQLite — полегшена реляційна система керування базами даних. Втілена у вигляді бібліотеки, де реалізовано багато зі стандарту SQL-92.

РОЗДІЛ 3. РОЗРОБКА ПРОДУКТУ

3.1. Реєстрація Телеграм-боту

Інструкція для створення Телеграм-боту для гуртожитку:

- Реєстрація бота за допомогою утиліти Botfather. Вигляд утиліти має бути саме такою як на рис.3.1;



Рис. 3.1. Вигляд утиліти Botfather

- Придумуємо ім'я та тег, який обов'язково має містити вираз bot. У даному випадку бот має назву @NauHostel9_bot;
- Бот успішно зареєстрований, далі отримуємо повідомлення з токеном. Токен – це секретний код, який дозволяє зв'язати сервіси з ботом у Телеграмі.

Кафедра КІТ (47)				НАУ 21 04 86 000 ПЗ			
Виконав	Гасанова А.М.			Розробка продукту	Літера	Аркуш	Аркушів
Керівник	Зіатдінов Ю.К.					33	54
Консульт.							
Н-котрол.	Шевченко О.П.				411 122		
Зав. каф.	Савченко А.С.						

3.2. Тестування програми написаної на мові високого рівня програмування Python

Telegram-бот розпочинає свою роботу із питання (рис.3.2, 3.3).

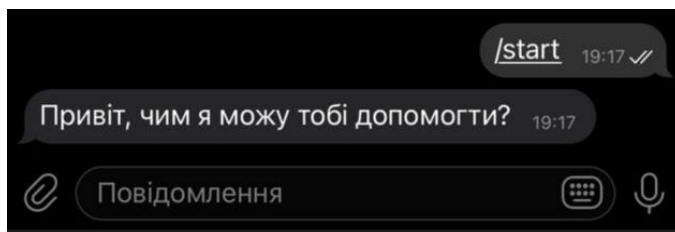


Рис. 3.2. Запуск Телеграм-боту

```
1 from loader import dp, bot
2 from constants import Constant
3 from app.menu.keyboards import menu_keyboard
4
5
6 async def start_command(message):
7     await message.answer('Привіт, чим я можу тобі допомогти?', reply_markup=menu_keyboard)
8
```

3.3. Реалізація за допомогою коду

Головне меню складається з основних розділів боту (рис.3.4,3.5):

- Студрада гуртожитку;
- Адміністрація;
- Заяви;
- Поселення;
- Інформаційні канали;
- Вахта;
- Скарги та пропозиції.

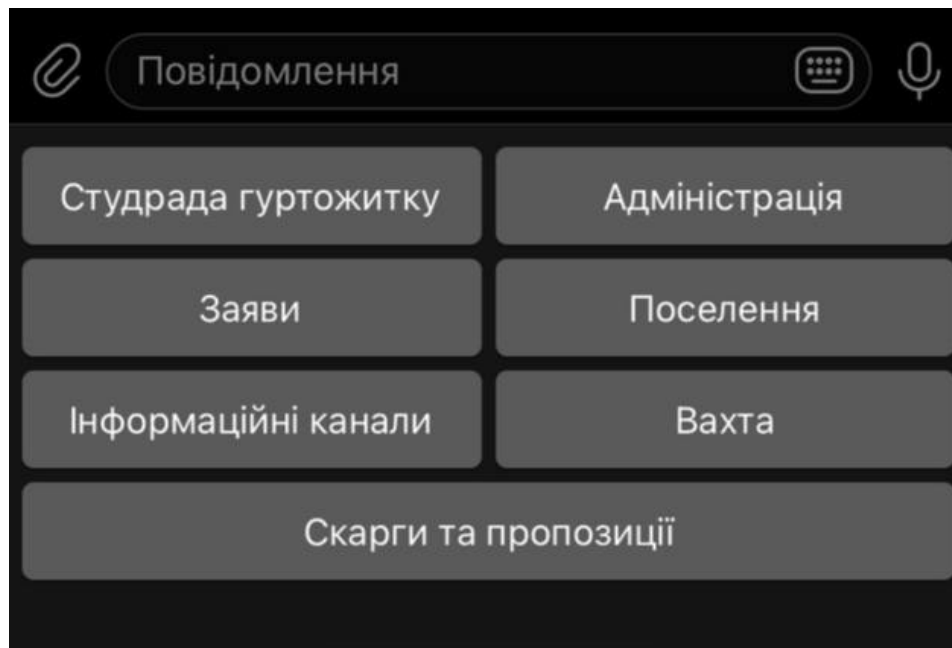


Рис. 3.4. Головне меню

```

1  from app.common.keyboards import BaseReplyKeyboardMarkup
2
3
4  class MenuKeyboard(BaseReplyKeyboardMarkup):
5      rows = [
6          ['Студрада гуртожитку', 'Адміністрація'],
7          ['Заяви', 'Поселення'],
8          ['Інформаційні канали', 'Вахта'],
9          ['Скарги та пропозиції']
10     ]
11
12
13  menu_keyboard = MenuKeyboard().get_keyboard()
14

```

Рис. 3.5. Реалізація головного меню

У кнопці «Студрада гуртожитку» (рис.3.6, 3.7) прописані члени Студентської ради гуртожитку. Студентська рада розбита на відділи, що займаються своїми обов'язками:

- Голова СРСМ;
- Голова гуртожитку №9;

- Зами;
- Старости;
- СБ;
- Сан.стан;
- Контроль відпрацювань;
- СММ;
- Івенти;
- Еко НАУ.

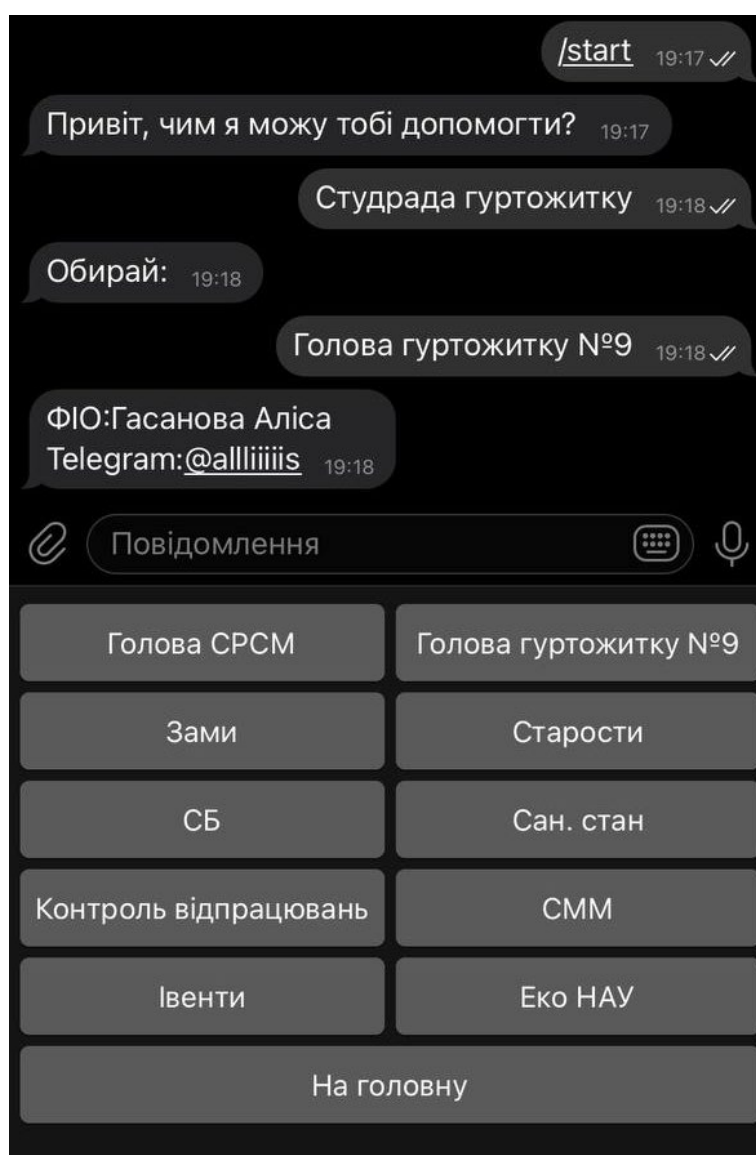


Рис. 3.6. Тестування кнопки «Студентська рада»

```

1  from app.common.keyboards import BaseReplyKeyboardMarkup
2
3
4  class StudentHouseKeyboard(BaseReplyKeyboardMarkup):
5      rows = [
6          ['Голова СРСМ', 'Голова гуртожитку №9'],
7          ['Зами', 'Старости'],
8          ['СБ', 'Сан. стан'],
9          ['Контроль відпрацювань', 'СММ'],
10         ['Івенти', 'Еко НАУ'],
11     ]
12     main_page_button = True
13

```

Рис. 3.7. Реалізація «Студ.ради»

Обробка команди при натиску кнопки «Голова гуртожитку №9», відповідь рис. 3.8.

```

async def head_hostel_info(message):
    head = await dp.db.query_manager(HostelAdministration).get_or_none(
        type=HostelAdministration.AdministrationType.HEAD_HOSTEL)
    if head:
        await message.answer(f'''
ФІО:{head.full_name}
Telegram:{head.telegram_username}
''')
    else:
        await message.answer('Дані відсутні')

```

Рис. 3.8. Кодова реалізація кнопок «Студ.ради»

Переходячи у розділ «Старости» (рис. 3.9,3.10) , ми можемо побачити:

- Ліве крило;
- Праве крило;
- Старости лівого крила;
- Старости правого крила.

Таки поділ кнопок зручний для користувача в знаходженні шуканого старости.

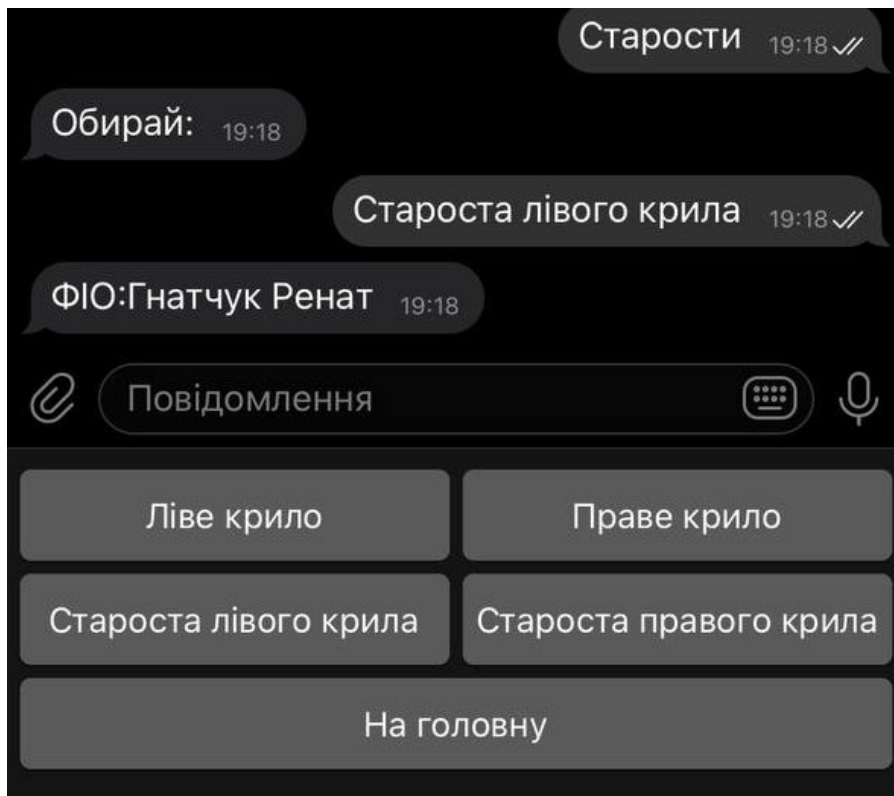


Рис. 3.9. Тестування кнопки «Старости»

```
class HeadmanKeyboard(BaseReplyKeyboardMarkup):
    rows = [
        ['Ліве крило', 'Праве крило'],
        ['Староста лівого крила', 'Староста правого крила']
    ]
    main_page_button = True
```

Рис. 3.10. Написання кнопок «Старости»

Для прикладу обираємо «Ліве крило» і «Староста 7» отримуємо відповідь (рис. 3.11, 3.12). У випадку, якщо старости немає чи не ведені дані отримуємо відповідь «Дані відсутні».

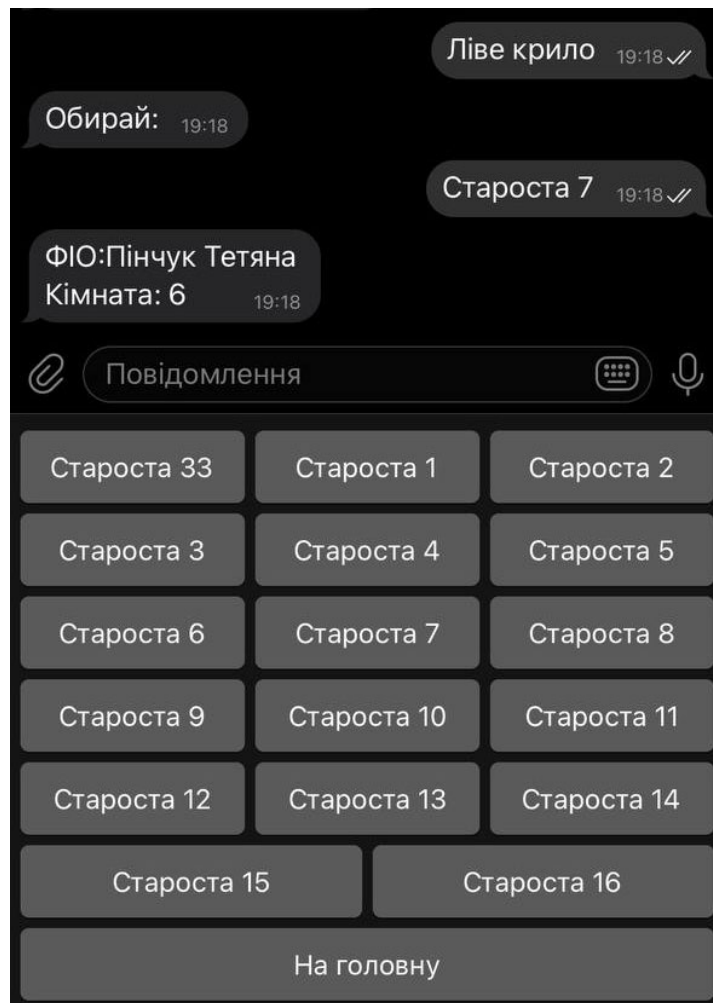


Рис. 3.11. Тестування кнопки «Староста 7»

```

async def left_block_head_info(message):
    head = await dp.db.query_manager(HostelAdministration).get_or_none(
        type=HostelAdministration.AdministrationType.HEAD_LEFT_SIDE)
    if head:
        await message.answer(f'''
        ФІО:{head.full_name}

        ''')
    else:
        await message.answer('Дані відсутні')

```

Рис. 3.12. Реалізація за допомогою коду

Повертаючись до Головного меню переходимо на кнопку «Адміністрація» (рис.3.13,3.14):

- Бачимо початкове питання;
- Натискаємо кнопку «Каштелян» відповідь ФІО, кабінет та робочі години.

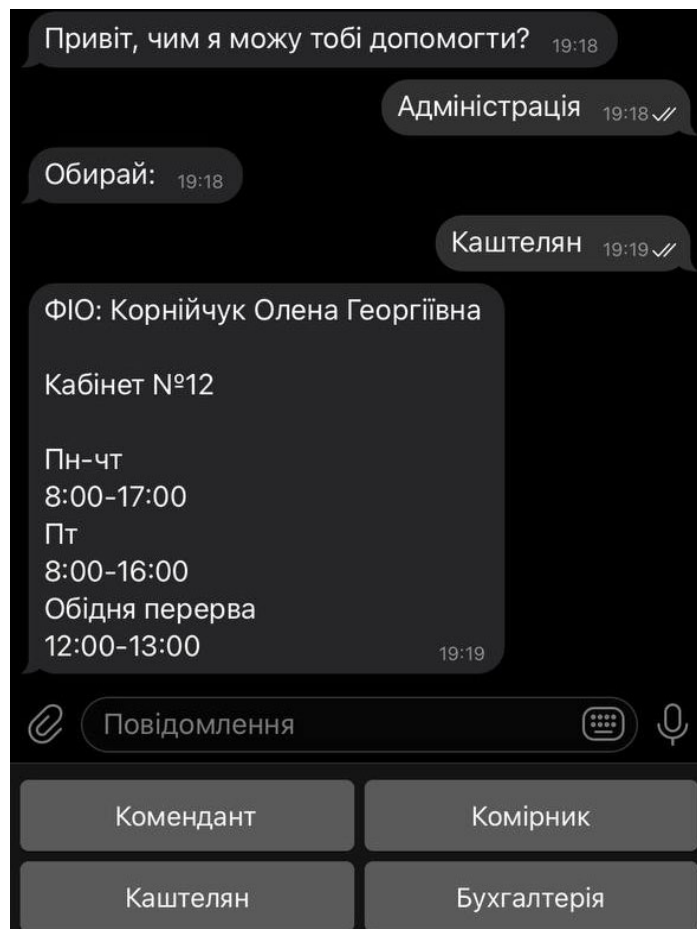


Рис. 3.13. Тестування кнопки «Адміністрація» і «Каштелян»

Кабінет, робочі години прописані відразу, так як вони незмінні.

```

async def storekeeper_info(message):
    template = """
    ФІО: {fullname}
    {data}

    """
    storekeeper = await dp.db.query_manager(HostelAdministration).get_or_none(
        type=HostelAdministration.AdministrationType.STOREKEEPER)
    await message.answer(template.format(fullname=storekeeper.full_name, data='''
    Кабінет №4

    Пн-чт
    8:00-17:00
    Пт
    8:00-16:00
    Обідня перерва
    12:00-13:00
    '''))
  
```

Рис. 3.14. Реалізація за допомогою коду обробку запиту «Каштелян»

Переходимо на головне меню, натискаємо кнопку «Заяви» (рис. 3.15, 3.16).

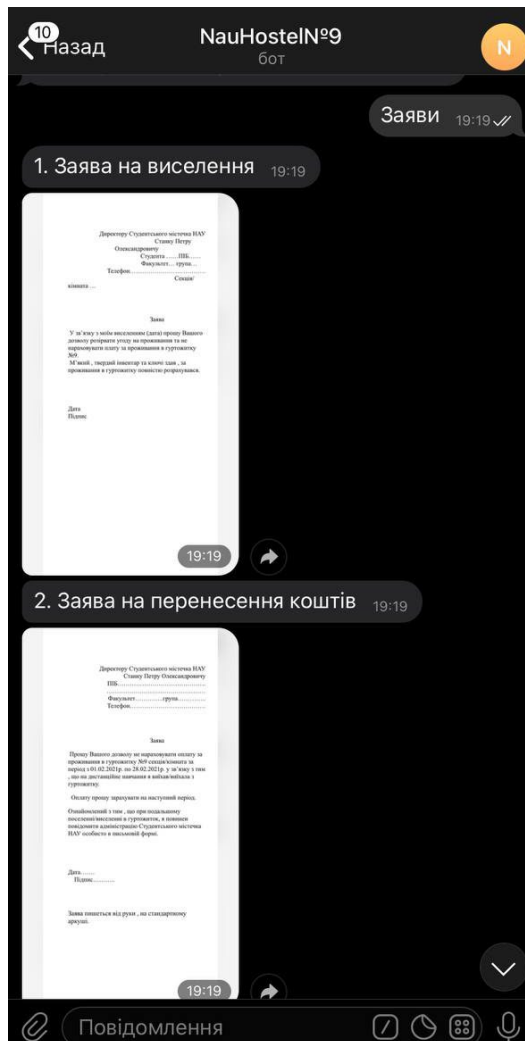


Рис. 3.15. Тестування кнопки «Заяви»

```
async def declarations(message):
    await message.answer("1. Заява на виселення")
    await bot.send_photo(message.chat.id, Constant.EVICTION_REQUEST_FILE_ID)
    await message.answer("2. Заява на перенесення коштів")
    await bot.send_photo(message.chat.id, Constant.MONEY_TRANSFER_FILE_ID)
    await message.answer("3. Заява на тимчасове проживання в гуртожитку")
    await bot.send_photo(message.chat.id, Constant.TEMPORARY_RESIDENCE)
    await message.answer("4. Заява на поселення 2021-2022 навчальний рік")
    await bot.send_photo(message.chat.id, Constant.RESIDENCE_DECLARATION)
    await message.answer("5. Гугл-форма для заповнення на поселення 2021-2022 навчальний рік")
    await message.answer(Constant.GOOGLE_FORM_RESIDENCE_DECLARATION)

async def settlements(message):
    await message.answer(Constant.SETTLEMENT)
```

Рис. 3.16. Реалізація за допомогою коду відповіді на запит «Заяви»

Фотокартки завантажені на хмарний сервер Телеграму, там вони мають унікальний ід (рис. 3.17), це зроблено для того, щоб не зберігати всі медіа файли локально.

```
EVICTIION_REQUEST_FILE_ID = "AgACAgIAAxkBAAEJj0tghvWn4WmQVLedw1twoefDjYcQ2wACs7QxGxUaEEg0WvVhH4NA3XgLCJ8uAAMBAAMCAAN5AA0xagE
MONEY_TRANSFER_FILE_ID = "AgACAgIAAxkBAAEJj09ghvXel9nuFII9yV8yVWmmyaZnQAACtLQxGxUaEEI2NJuWvST91Okzo6IuAAMBAAMCAAN5AANmdgEAA
TEMPORARY_RESIDENCE = "AgACAgIAAxkBAAEJj1NghvZq_Ad6wCNfuz9IYV8S4VakawACTbQxGxUaEEhogoWdXbuPIxTuzaIuAAMBAAMCAAN5AAPqWgEAAR8E
RESIDENCE_DECLARATION = "AgACAgIAAxkBAAEJj1VghvaxPuYq6Q3pcNZwzQc8V8zTTQACtrQxGxUaEEjozEiISC9RF-LSKaQuAAMBAAMCAAN5AAMnOQACHw
GOOGLE_FORM_RESIDENCE_DECLARATION = "https://docs.google.com/forms/d/1_5hwouT7P5R9BBGDSmZIUxWnbGs-kz4NzAcujAiEnCQ/edit"
SETTLEMENT = "https://telegra.ph/Algoritm-poselennya-pershokursnik%D1%96v-v-gurtozhitki-studentskogo-m%D1%96stechka-NAU-08-
```

Рис. 3.17. Унікальний ід фотокарток

Натискаємо кнопку «Поселення», відповідь є посилання на детальну інструкцію для поселення (рис. 3.18).

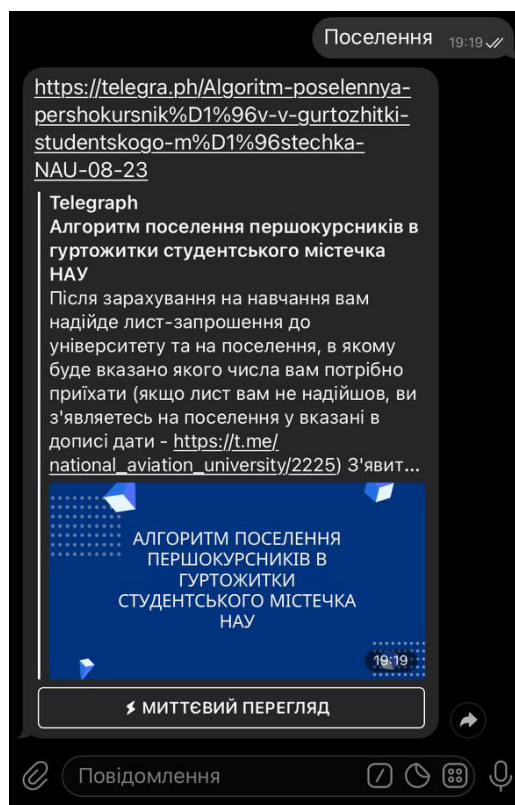


Рис. 3.18. Тестування кнопки «Поселення»

Натискаємо кнопку «Інформаційні канали» (рис. 3.19, 3.20), де зібрані всі

необхідні служби для допомоги та ознайомлення ситуації в гуртожитку.

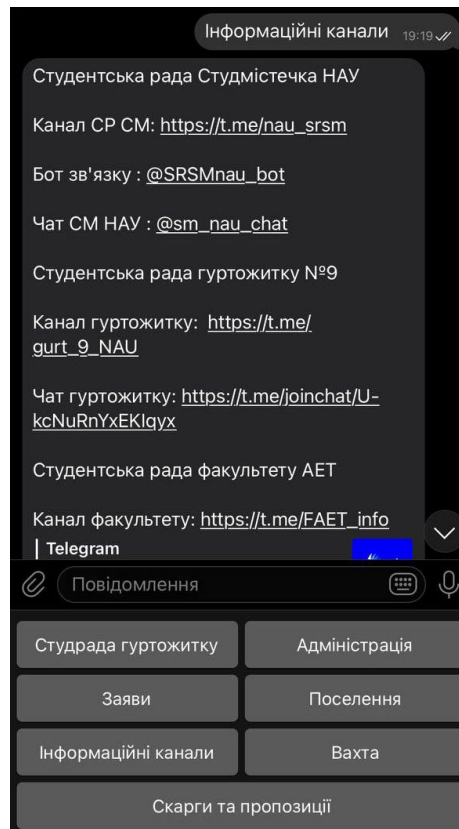


Рис. 3.19. Тестування кнопки «Інформаційні канали»

```
async def information_channels(message):  
    await message.answer("""  
Студентська рада Студмістечка НАУ  
  
Канал СР СМ: https://t.me/nau\_srsм  
  
Бот зв'язку : @SRSMnau_bot  
  
Чат СМ НАУ : @sm_nau_chat  
  
Студентська рада гуртожитку №9  
  
Канал гуртожитку: https://t.me/gurt\_9\_NAU  
  
Чат гуртожитку: https://t.me/joinchat/U-kcNuRnYxEKlqyx  
  
Студентська рада факультету АЕТ  
  
Канал факультету: https://t.me/FAET\_info  
  
""")
```

Рис. 3.20. Реалізація за допомогою коду відповіді на запит «Інформаційні канали»

Допоміжна кнопка «Скарги/пропозиції» (рис. 3.21), яка передає інформацію до адміністратора боту. Мешканець гуртожитку має можливість виразити свої думки анонімно.

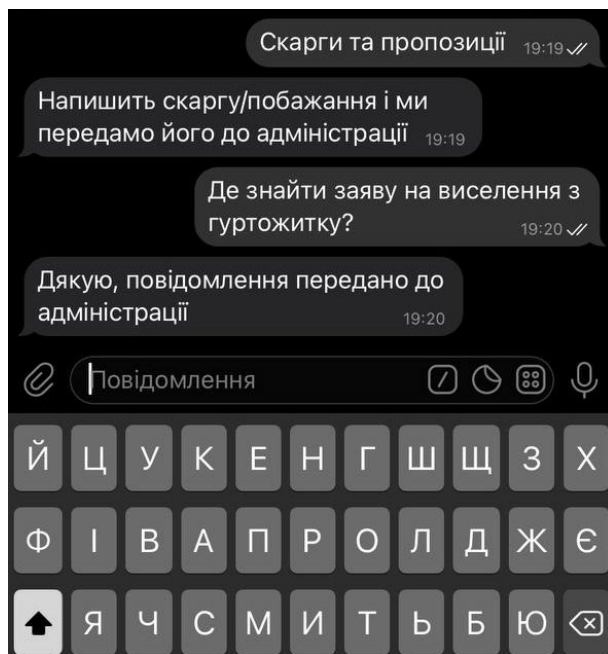


Рис. 3.21. Тестування кнопки «Скарги та пропозиції»

Студентська рада гуртожитку та адміністрація може змінюватися, для полегшення видалення та додавання нових членів студради чи адміністрації є `/admin_panel` (рис. 3.22). Вона створена для спрощення коду та миттєвої зміни студради чи адміністрації, також там знаходяться скарги чи пропозиції від мешканців гуртожитку.

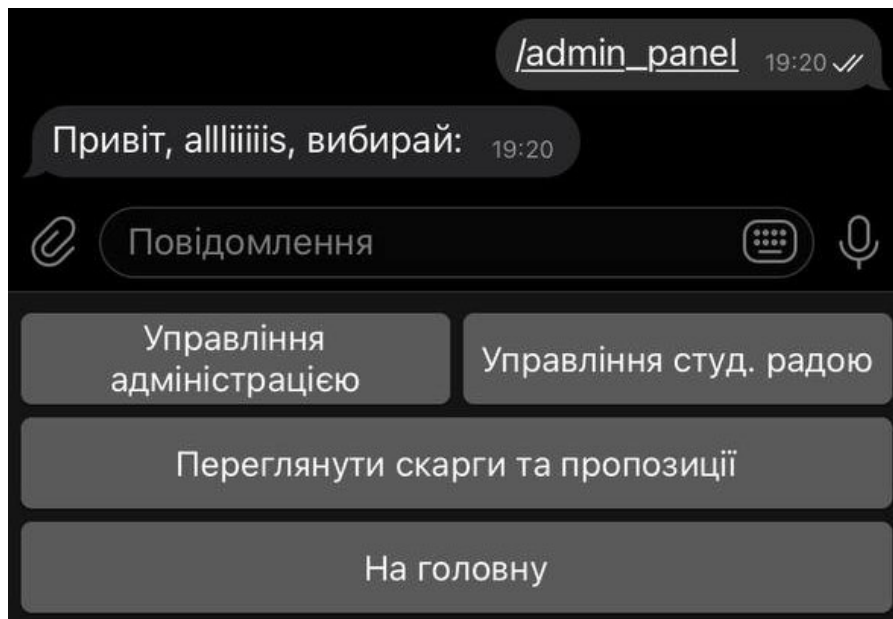


Рис. 3.22. Вхід в admin_panel

Містить кнопки (рис. 3.23):

- Управління адміністрацією;
- Управління студ. Радою;
- Переглянути скарги та пропозиції.

```
from app.common.keyboards import BaseReplyKeyboardMarkup

class AdminMenuKeyboard(BaseReplyKeyboardMarkup):
    rows = [
        ['Управління адміністрацією', 'Управління студ. радою'],
        ['Переглянути скарги та пропозиції']
    ]
    main_page_button = True
```

Рис. 3.23. Реалізація кнопок за допомогою коду

Вибираємо «Управління адміністрацією», далі «Змінити каштеляна» (рис. 3.24), записуємо ФІО, отримуємо «Дані збережено» (рис. 3.25, 3.26) результат

зберігається у БД, переходимо в головне меню, натискаємо кнопку «Адміністрація», «Каштелян» та бачимо відповідь рис. 3.13.

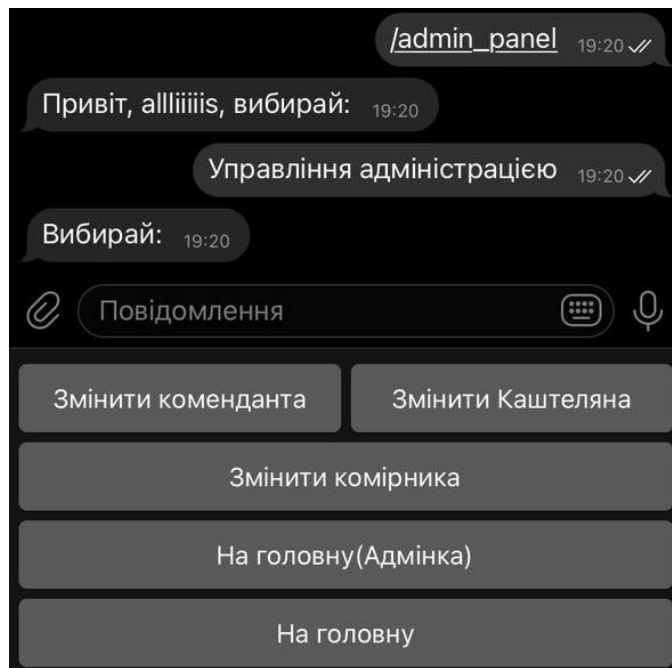


Рис. 3.24. Тестування кнопки «Управління адміністрацією»

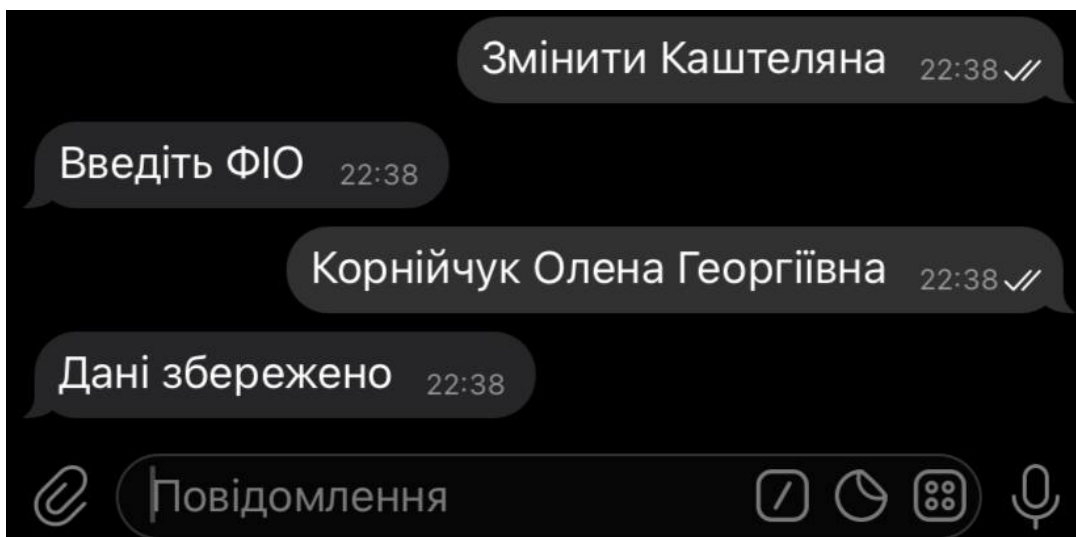


Рис. 3.25. Зміна Каштеляна

```

async def set_castellane(message):
    await SetCastellaneForm.fullname.set()
    await message.answer('Введіть ФІО')

async def process_fio_castellane(message, state):
    async with state.proxy() as data:
        data['full_name'] = message.text
        await dp.db.query_manager(HostelAdministration).filter(
            type=HostelAdministration.AdministrationType.CASTELLANE).delete()
        await dp.db.query_manager(HostelAdministration).create(type=HostelAdministration.AdministrationType.CASTELLANE,
                                                                **data)

    await state.finish()
    await message.answer('Дані збережено')

```

Рис. 3.26. Реалізація за допомогою коду зміну Каштеляна

Переходимо в головне меню (адміністратора), натискаємо кнопку «Управління студ. Радою» (рис. 3.27, 3.28) «Додати старосту секції» та вводимо дані (рис. 3.29, 3.30). Після чого перевіряємо правильність даних у головному меню рис. 3.11.

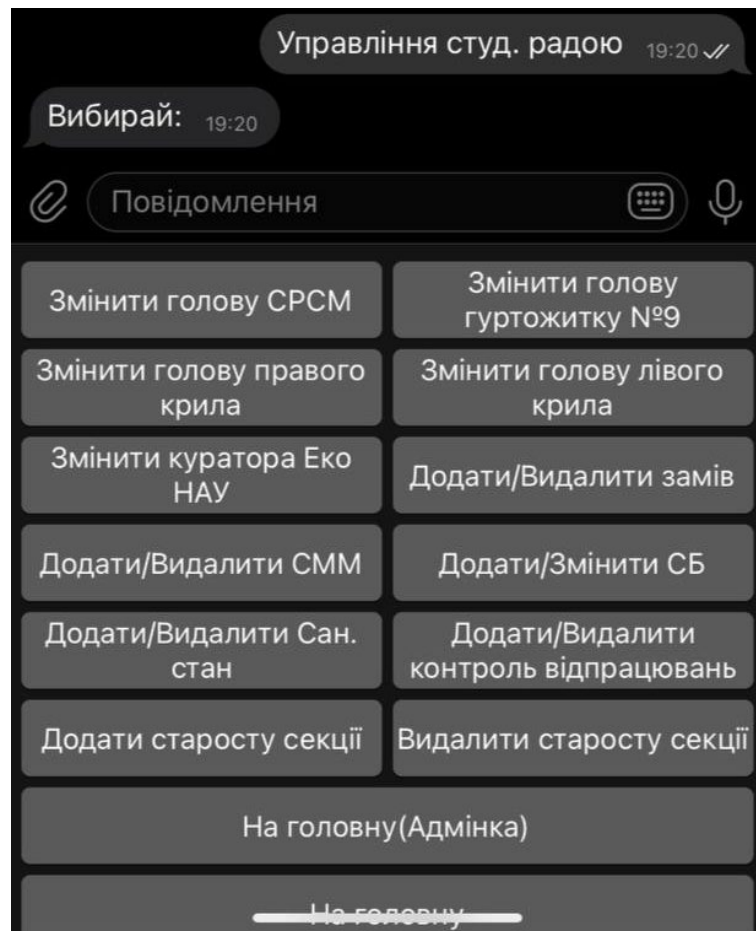


Рис. 3.27. Тестування кнопки « Управління студ.радою»

```

class AdminManageStudentHouseKeyboard(BaseReplyKeyboardMarkup):
    rows = [
        ['Змінити голову СРСМ', 'Змінити голову гуртожитку №9'],
        ['Змінити голову правого крила', 'Змінити голову лівого крила'],
        ['Змінити куратора Еко НАУ', 'Додати/Видалити замів'],
        ['Додати/Видалити СММ', 'Додати/Змінити СБ'],
        ['Додати/Видалити Сан. стан', 'Додати/Видалити контроль відпрацювань'],
        ['Додати старосту секції', 'Видалити старосту секції'],
    ]
    main_page_button = True
    main_admin_button = True

```

Рис. 3.28. Реалізація кнопок за допомогою коду

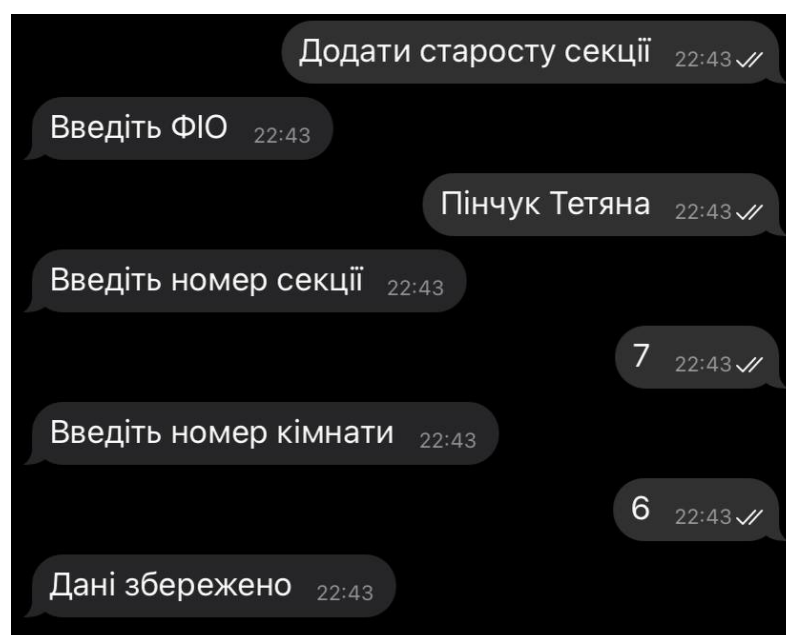


Рис. 3.29. Додавання старости секції


```

async def add_section_head(message):
    await AddHeadSectionForm.fullname.set()
    await message.answer('Введіть ФІО')

async def process_fio_add_section_head(message, state):
    async with state.proxy() as data:
        data['full_name'] = message.text

    await AddHeadSectionForm.next()
    await message.answer('Введіть номер секції')

async def process_section_number_add_section_head(message, state):
    async with state.proxy() as data:
        data['section_number'] = message.text

    await AddHeadSectionForm.next()
    await message.answer('Введіть номер кімнати')

async def process_room_number_add_section_head(message, state):
    async with state.proxy() as data:
        data['room_number'] = message.text
        await dp.db.query_manager(HostelAdministration).filter(
            type=HostelAdministration.AdministrationType.HEAD_SECTION, section_number=data['section_number']).delete()
        await dp.db.query_manager(HostelAdministration).create(
            type=HostelAdministration.AdministrationType.HEAD_SECTION, **data)

    await state.finish()
    await message.answer('Дані збережено')

```

Рис. 3.30. Реалізація за допомогою коду обробку «Додавання старости секції»

Відкриваємо кнопку «Переглянути скарги та пропозиції», бачимо останні 10 скарг/пропозицій. Усе, що пишеться в даній кнопці зберігається у БД, а потім відображається при запиті (рис. 3.31, 3.32).

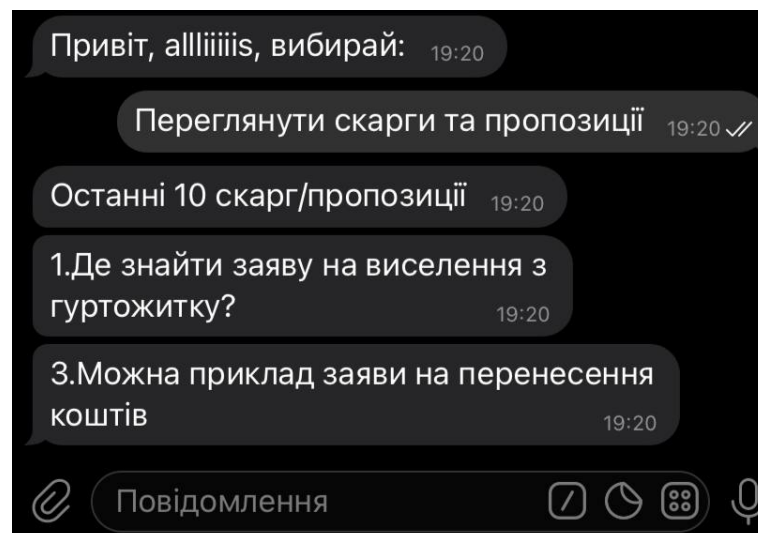


Рис. 3.31. Перегляд «скарг/пропозицій»

```

async def all_feedback_message(message):
    messages = await dp.db.query_manager(FeedBackMessage).order_by('-id').limit(10).execute()
    if messages:
        await message.answer('Останні 10 скарг/пропозиції')
        for row_num, msg in enumerate(messages, 1):
            await message.answer(f"{row_num}.{msg.message}")
    else:
        await message.answer('Дані Відсутні')

```

Рис. 3.32. Реалізація за допомогою коду додавання старости секції

Для зручності керування можна долучити ще одного адміністратора. Для цього потрібно Телеграм id користувача (рис. 3.33).

```

1  from asynormlite.engine import SQLiteEngine
2  from app.admin.models import AdminUser
3  from loader import config
4
5  async def create_admin():
6      user_id = input('Введіть ваше ID в телеграмме:')
7
8
9      db = await SQLiteEngine(config['DB_NAME'])
10     await db.create_table(AdminUser)
11     await db.query_manager(AdminUser).create(user_id=user_id)
12     print('Новий користувач успішно створено')
13     await db.close_connection()

```

Рис. 3.33. Реалізація за допомогою коду, як долучити ще одного адміністратора

Як вище зазначалось, усі зміни та скарги/пропозиції зберігаються у БД. З БД можливо також видаляти дані, що вже неактуальні (рис. 3.34, 3.35).

DB Browser for SQLite - C:\Users\User\Desktop\NauHostel9\db.sqlite3

Файл Редактирование Вид Инструменты Справка

Новая база данных Открыть базу данных Записать изменения Отменить изменения Открыть проект Сохранить проект Прикрепить БД Закрывать

Структура БД Данные Праги SQL

Таблица: hostel_administrations Filter in any column

	id	type	full_name	phone_number	telegram_username	email	section	section_number	room_number
	Фи...	Фи...	Фильтр	Фильтр	Фильтр	Фил...	Фильтр	Фильтр	Фильтр
1	5	12	Петраков Василь	NULL	NULL	NULL	NULL	NULL	
2	6	8	Гнатчук Ренат	NULL	NULL	NULL	NULL	NULL	
3	11	9	Іващенко Дмитро	NULL	NULL	NULL	NULL	NULL	
4	12	10	Даніч Максим	NULL	NULL	NULL	NULL	33	
5	13	10	Полякова Олександра	NULL	NULL	NULL	NULL	1	
6	14	10	Сахно Михайло	NULL	NULL	NULL	NULL	2	
7	16	10	Разицкий Денис	NULL	NULL	NULL	NULL	4	
8	17	10	Артеменко Діана	NULL	NULL	NULL	NULL	5	
9	18	10	Різник Сергій	NULL	NULL	NULL	NULL	6	
10	20	10	Аліфіренко Тетяна	NULL	NULL	NULL	NULL	8	
11	21	10	Старожитник Дмитро	NULL	NULL	NULL	NULL	9	
12	22	10	Сава Максим	NULL	NULL	NULL	NULL	10	
13	23	10	Вакансія відкрита	NULL	NULL	NULL	NULL	11	Будь-яка з 11 с
14	24	10	Янченко Михайло	NULL	NULL	NULL	NULL	12	
15	25	10	Гнатчук Ренат	NULL	NULL	NULL	NULL	13	
16	26	10	Білан Євгеній	NULL	NULL	NULL	NULL	14	
17	27	10	Гондарук Тетяна	NULL	NULL	NULL	NULL	16	
18	28	10	Додати старосту секції	NULL	NULL	NULL	NULL	Додати старосту секції	Додати старосту с
19	29	10	Космірак Валерія	NULL	NULL	NULL	NULL	17	
20	31	10	Ломаченко Анжела	NULL	NULL	NULL	NULL	19	
21	32	10	Попенко Владислав	NULL	NULL	NULL	NULL	20	
22	33	10	Іващенко Дмитро	NULL	NULL	NULL	NULL	21	

Рис. 3.34. Вигляд БД

```

async def init_db(dp):
    dp.db = await SQLiteEngine(config['DB_NAME'])

async def init_tables(dp):
    await dp.db.create_tables(TABLES)

```

Рис. 3.35. Підключення БД до Телеграм-боту

Висновки до третього розділу

У даному розділі було спроектовано Телеграм-бот на мові високого рівня програмування. Відбулося тестування програми, опис її реалізації за допомогою коду.

На рисунках показана реалізація кнопок, відтворення кнопок у роботі з Телеграм-ботом, написання коду за допомогою мови Python та БД, де зберігаються дані.

Також, розроблена `/admin_panel`, що використовується у керуванні та доступна тільки адміністраторам.

ВИСНОВКИ

У даному дипломному проєкті розглядається актуальна тема для мешканці гуртожитку. Кожен з студентів стикається з проблемою інформативності. Для продуктивності та спрощення обов'язків Студентської ради гуртожитку було створено Телеграм-бот, який включає в себе актуальні дані.

За період розробки Телеграм-боту було проаналізовано актуальність Телеграму та його властивості. Показані навички користування програмним забезпеченням та написання Телеграм-боту на мові високого рівня програмування.

Розроблено Телеграм-бот, що повністю оновлений поточною інформацією і готовий до опублікування. Використання даного боту зрозуміле та немає викликати проблеми у пересічного користувача.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Який меседжер безпечніший? – [Електронний ресурс] Режим доступу: <https://yak.dslua.org/articles/yakyy-mesendzher-naybezpechnishyy/>(дата звернення 14.05.2021) – Назва з екрана.
2. Function of Telegram – [Електронний ресурс] Режим доступу: <https://studway.com.ua/pro-telegram/>(дата звернення 19.05.2021) – Назва з екрана.
3. Telegram bot – [Електронний ресурс] Режим доступу: <https://evergreens.com.ua/ru/articles/bot-reviews.html/>(дата звернення 22.05.2021) – Назва з екрана.
4. Python – [Електронний ресурс] Режим доступу: <http://www.plug.org.ua/documentation/about-python/>(дата звернення 27.05.2021) – Назва з екрана.
5. VS Code – [Електронний ресурс] Режим доступу: https://uk.wikipedia.org/wiki/Visual_Studio_Code/(дата звернення 31.05.2021) – Назва з екрана.
6. Telegram Bot API – [Електронний ресурс] Режим доступу: <https://core.telegram.org/bots/api/>(дата звернення 03.06.2021) – Назва з екрана.
7. SQLite – [Електронний ресурс] Режим доступу: <https://uk.wikipedia.org/wiki/SQLite/>(дата звернення 05.06.2021) – Назва з екрана.