

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
Факультет комп'ютерних наук, фізики та математики
Кафедра комп'ютерних наук та програмної інженерії

ВИКОРИСТАННЯ ТЕЛЕГРАМ БОТІВ У ШКОЛІ

Кваліфікаційна робота

на здобуття ступеня вищої освіти «бакалавр»

Виконав: Сушко Владислав Русланович

студент 4 курсу 432 групи

Спеціальності: 014.09 Середня освіта.

Інформатика

Освітньо-професійної програми:

Середня освіта. Інформатика

Корнієнко О.М.

Керівник: доктор пед.н., професор

Валько Н.В.

Рецензент:

Яковенко Євгенія Валеріївна,

учитель математики, Херсонська

спеціалізована школа із поглибленим

вивченням математики, фізики та

англійської мови №24

Херсон – 2022

Зміст	
ВСТУП.....	4
I. БОТИ ЯК ВАЖЛИВА ЧАСТИНА СПІЛКУВАННЯ У МЕРЕЖІ.....	4
1.1 Поняття чат-боту	4
1.2 Чат-боти в месенджері Telegram (створення та розробка).....	6
1.3 Чат-боти навчального характеру (для навчальних організацій, підприємств, шкіл, ВНЗ, тощо)	8
II. ВИБІР ЗАСОБІВ РЕАЛІЗАЦІЇ ЧАТ-БОТУ.....	10
2.1 Мова програмування Python	10
2.2 Мова програмування Java	15
III. ОПИС РОЗРОБКИ ЧАТ-БОТУ	17
3.1 Створення чат-боту	17
3.2 Тестування чат-боту	23
ВИСНОВКИ	26
Список використаних джерел.....	28

ВСТУП

I. БОТИ ЯК ВАЖЛИВА ЧАСТИНА СПІЛКУВАННЯ У МЕРЕЖІ

1.1 Поняття чат-боту

Програма чат-бот має заздалегідь прописаний алгоритм запитань і відповідей, який виводить текстові або голосові повідомлення. Це дозволяє брендам регулярно спілкуватися з клієнтами. Основна робота покладена на заздалегідь написані комунікаційні нотатки і робот відразу надає необхідну відповідь аудиторії в момент написання інформаційного запиту. Чат-боти можна використовувати безпосередньо на веб-сайтах, у месенджерах чи соціальних мережах. Вони часто замінюють комп'ютерні та мобільні додатки.

Чат-боти можуть бути легкими або важкими. У першому випадку справжній помічник працює лише за суворими інструкціями: він лише надсилає відповіді на запитання користувача у своїй системі. Якщо слова користувача не збігаються із заявленим у програмі, роботу буде важко отримати правильну відповідь. А тепер із замовником спілкується жива людина.

Деякі інтелектуальні чат-боти на основі штучного інтелекту знають команди будь-якою мовою. Такі роботи змінюються з часом: коригуючи інформацію, яку вони отримують, вони вчаться і стають розумнішими. Користувачам не потрібно вводити точні пошукові слова, щоб отримати відповідь від такого робота. Коротше кажучи, продуктивність таких розмовних ботів заснована на технології штучного інтелекту, обробки природної мови (NLP) і машинного навчання (ML).

Основним завданням чат-ботів, крім надання технічної допомоги, є збільшення аудиторії та утримання потенційних клієнтів у каналі продажів з мінімальними фінансовими внесками. Зосереджуючись на зростаючій

аудиторії, яка користується месенджерами, маркетологи все частіше використовують чат-ботів для вирішення комунікаційно-організаційних проблем.

Чат-боти можна використовуються практично у багатьох галузях. Розглянемо кілька прикладів того, як різні організації використовують роботів у своїй роботі.

Загалом, чат-боти можуть виконувати такі бізнес-функції:

- Надавати підтримку клієнтам щодо продукту (зазвичай відповідає на запитання щодо поширених запитань).
- Поширення важливої інформації (навчання, боти для розмови новин).
- Участь користувача в спілкуванні (та перенаправлення замовлення).
- Замовлення та оплата замовлень.
- Комунікації всередині компанії (збір та обробка інформації про співробітників, нагадування про роботу, отримання звітів і перезапуск).

З одного боку, чат-робот допомагає легко і швидко виконувати щоденні завдання, а з іншого – автоматично збирає та обробляє замовлення клієнтів. Месенджери більше не є засобом надсилання листів, вони використовували ділові листи для спілкування з колегами та співробітниками. На цій ідеї заснована концепція чат-ботів.

Зараз чат-боти доступні на таких платформах, як:

- Facebook Messenger
- Skype
- Viber
- Telegram
- Slack
- Discord

Кожен із них пропонує виробникам власні інструменти для взаємодії і має свій аспект у стилі та складі своєї аудиторії.

1.2 Чат-боти в месенджері Telegram (створення та розробка)

Telegram Messenger створив у червні 2015 року платформу для створення роботів, які реагують на команди користувача та взаємодіють з ними.

Відтоді було створено багато роботів. Навіть ЗМІ, які перевіряли платформу, швидко зацікавилися їх дизайном розробки чат-ботів, які все можуть залучати мобільні пристрої для контакту та взаємодії з ресурсами. Виконуючи рівномірну операцію, робот не тільки економить енергію та час людини, але і працює швидко. Чат-боти мають кращу продуктивність і довіру, ніж люди – їх використовують у комп'ютерних іграх, онлайн-аукціонах, маркетингу, електронній комерції тощо. Чат-боти використовуються для імітації діяльності людини, особливо в аудиторіях.

Вони можуть бути налаштовані для створення повних людських відповідей, допомагають робити різні речі: перекладати, говорити, вчити, грати.

Ця платформа пропонує багато можливостей для використання.

Чат-ботів найбільша допомога в тому, що користувачеві не потрібно виходити з месенджера, щоб робити щось інше.

Чат-боти можуть працювати разом. Їх можна налаштувати на виконання певних дій пізніше. Наприклад, принести останні новини, ранкову погоду, затори. А вночі - рекомендації для фільмів, анекдотів або щоденних рецептів.

Почати роботу з роботом досить легко: просто виберіть один довідник, дотримуйтесь інструкцій або шукайте за назвою та почніть роботу.

Розмова з ним у більшості випадків, якщо він увійшов в систему для запуску робота, виглядає як: команда / запуск

Чат-боти можуть бути дуже корисними у всіх сферах життя. Особливо високі очікування в плані розвитку Інтернету і інших аспектах життя.

Деякі люди використовують телеграфних роботів для організації взаємодії.

Наприклад, дозволити клієнту переглядати хід обслуговування та контролювати роботу

Використання обмежується лише фантазією користуючого та «батька».

За допомогою чат-боту можна робити дослідження, монтаж, нагадування.

Щоб освіжити пам'ять в діловому спілкуванні, ви можете запросити ботів, які переглядають граматику і вказують на помилки, або іноді запам'ятовують повідомлення людей і через деякий час надсилають їх іншим способом.

Бот має певні відмінності від звичайних облікових записів людей, що користуються месенджером Telegram, а саме [1]:

- У ботів немає статусів онлайн і «був в мережі», замість цього відображається назва бот. Це дозволяє користувачу зрозуміти, спілкується він з людиною чи з закодованою програмою;
- Для ботів виділено обмежене місце на серверах. Всі вони будуть видалені через певний термін після обробки;
- Боти не можуть самі почати спілкування з користувачем. Користувач повинен або додати робота в групу або першим почати з ним діалог;
- Ім'я бота має обов'язково закінчуватися на bot. Наприклад, @musicbot;

Для розроблення чат-ботів, необхідно дотримуватися деяких правил, а саме [2 Як створити чат-бот для Telegram-каналу – інструкція для адміністраторів. URL: <https://netpeak.net/ru/blog/kak-sozdat-chat-bot-dlyatelegraminstruktsiya-dlya-administratorov-kanalov/>];

1. Продумати і розпізнати, у чому саме бот стане допомагати організації та користувачам.
2. Написати весь план створення ботів.
3. Оформити докладну матрицю питань-відповідей взаємодії користувача з ботом.
4. Підготувати опис цілей і заходів для опції фахівці.
5. Підготувати технічне доручення для розробника з усіма важливими складовими частинами за планом.
6. Здійснити програмування, випробування, настроювання, налагодження, фінішний аналіз чат-ботів.
7. Запустити тестового бота.
8. Перевірити підсумки продуктивності роботи бота за сценарієм, спираючись на дані по фахівця і зворотному зв'язку за конкретний етап (наприклад, тиждень).
9. Доопрацювати бот у разі необхідності. Провести новий запуск бота, періодично перевіряти дані по фахівця і зворотному зв'язку.

1.3 Чат-боти навчального характеру (для навчальних організацій, підприємств, шкіл, ВНЗ, тощо)

Очікування покращення навчання за допомогою чат-ботів залежить від майстерності навчаючихся. Оскільки сучасні знання та навички часто знаходяться на найвищому рівні серед студентів, ми сприймаємо чат-боти як простий і швидкий інструмент, який не потребує великих коштів та ресурсів.

За існування великої кількості сервісів онлайн-навчання чат-боти є цікавим та багатообіцяючим інструментом, оскільки індивідуально використовуються будь-яким слухачем. Це залежить від рівня та швидкості навчання використовуючого. Зробіть навчання доступним для всіх, хто має доступ до Wi-Fi. Чат-боти не вимагають великих інвестицій і можуть принести користь мільйонам студентів у всьому світі.

Окрім відповідей на запитання, чат-боти також можуть заохочувати вас до вивчення, наприклад, чат-бот Norwegian Differ надсилає учням корисні публікації або запрошуюють їх приєднатися до чату, де обговорюють наукові питання .

Крім того, використання чат-ботів може допомогти вчителю оцінити доцільність навчання(провести оцінку знань), оскільки воно має особистий вплив на всіх учнів, оперативний зворотний зв'язок і створює згуртоване середовище. Найкращий варіант — створити чат-бота для Telegram самостійно (або за допомогою студентів), оскільки цей месенджер подобається молоді.

Популяризуються чат-боти для підготовки ЗНО, ДПА, тощо. Платформи, які існують в Україні [3. Чат-боти для навчання: огляд найпопулярніших та особливості використання URL: <https://teach-hub.com/chat-boty-dlia-navchannia-ohliad-naypopuliarnishykh-ta-osoblyvosti-vykorystannia/>]:

- Українська мова ZNOcloud – https://t.me/Ukr_mova_ZNOcloud.
- Українська література ZNOcloud – https://t.me/Ukr_lit_ZNOcloud.
- Історія ZNOcloud – https://t.me/ist_znocloud.
- Географія ZNOcloud – https://t.me/geographia_znocloud.
- Біологія ZNOcloud – https://t.me/biologia_znocloud.
- Математика ZNOcloud – https://t.me/math_znocloud.
- Фізика ZNOcloud – https://t.me/physics_znocloud.
- Хімія ZNOcloud – https://t.me/chemistry_znocloud.

Розумний чат-бот розробляється Університетом Дікіна у Вікторії, Австралія, щоб відповідати на запитання учнів про шкільне життя. Virtual Intelligence Assistant від IBM. Він допоможе вам знайти місце, запросити навчання, знайти місце для паркування тощо.

Сьогодні люди починають цінувати особистий спосіб підвищення самооцінки та професіоналізму в унікальній обстановці. Тому інтерес учнів

відіграє важливу роль у розробці навчальної програми. Зміст курсу адаптований до особистого характеру навчання, щоб усі студенти отримали досвід. Чат-боти та алгоритми штучного інтелекту допомагають зібрати та проаналізувати дані кожного учня.

II. ВИБІР ЗАСОБІВ РЕАЛІЗАЦІЇ ЧАТ-БОТУ

2.1 Мова програмування Python

Перш ніж вибрати інструмент для домашнього програмування та створення проекту чат-бота розберемося з деталями:

Інтерфейс прикладного програмування (API), який надає месенджер для розробників. Інтерфейс програмного забезпечення - набір термінів, які взаємодіють з різними типами програмного забезпечення. Це рівень рішення, необхідний для комунікації між кількома службами, де один або обидва забезпечують доступ до певних функцій, фокусуються на інших, не розкриваючи їх внутрішнього використання. У той же час програма, яка використовує API, має доступ до функціональних можливостей наданого їй сервісу, не потребуючи вмісту вивчення закономірності та відношення до понять. Інтерфейси прикладного програмування полегшують розробникам використання певних технологій під час створення додатків. API також дозволяють приховати внутрішні функції програми, щоб підвищити безпеку та надати можливість зовнішньої взаємодії із сервісом.

У попередньому розділі ми прийшли до висновку, що найкращим варіантом для створення чат-бота є Telegram Messenger. Telegram пропонує два типи API для розробників. Перший – це API Telegram, який призначений для створення сторонніх розробників для платформи Telegram. Він не буде використаний у цій роботі.

Іншим є Bot API, який дозволяє легко створювати програми, які використовують повідомлення Telegram як основу для спілкування з користувачами.

API бота – це те, що дозволяє легко та зручно створювати чат-ботів. Цей API дозволяє підключати роботів до системи Telegram. З точки зору системи Telegram, боти - це бонуси, які не потрібно вводити додаткові номери телефону. Ця сума відволікає цифри на нашому сервері. Вам не потрібно нічого знати про те, як працює процес шифрування MTProto, що використовується в системі Telegram - їх середній сервер підтримує все шифрування та зв'язок із Telegram API. Для зв'язку з цим сервером використовується простий інтерфейс HTTPS, який має просту структуру API Telegram.

Усі запити до API Telegram Bot мають надсилатися через HTTPS і мати таку форму: `https://api.telegram.org/bot <TOKEN> / <METHODNAME>`. Веб-сайт <TOKEN> повинен мати маркер авторизації роботу, відповідно і зміну методу <НАЗВА МЕТОДУ>, який використовується з API.

API підтримує HTTP-запити GET і POST. Додаток GET діє за необхідності отримати деяку інформацію з API і отримати запит - при необхідності відправити деяку інформацію з API. Відповідями на запитання є об'єкти JSON, які завжди добре продумані відповіддю «ok», що означає завершення запиту, і може мати опцію «description» з описом результату. Якщо для «ok» встановлено значення «true», запит завершено, і результати запиту можна побачити в полі «result». Якщо запит не вдався, "ok" має " false ", а помилка пояснюється в полі " description ".

Номер поля " error_code " вказує на помилку. Деякі помилки також можуть мати опцію " parameters ", що може допомогти виправити помилку. Є два спеціалізовані способи отримати оновлення завантаження - метод "getUpdates" і використання веб-хуків, обробників подій. Якщо ви використовуєте вебхуки, сервер Telegram, якщо є оновлення, ініціює подію, що вебхуки підключені, і відповідають, отримують оновлення. Оновлення можуть зберігатися на сервері, доки робот не отримає їх одним із зазначених вище способів, але не довше 24 годин. Тому зворотний зв'язок слід

отримувати щонайменше раз на 24 години. Це потрібно робити часто, оскільки частина цієї інформації є повідомленнями користувача.

Розглянемо основні типи які підтримує Bot API, та які можуть бути передані як у запиті бота, так і при відповіді на цей запит:

- а) User – цей об'єкт являє собою користувача або бота;
- б) Chat – цей об'єкт являє собою чат чи групу;
- в) Message – цей об'єкт являє собою повідомлення. Використовується як для надсилання інформації користувачу, так і для отримання від нього відповідей;
- г) MessageEntity – один із елементів повідомлення (наприклад тег, ім'я користувача чи url-посилання);
- д) PhotoSize – відображає розмір фотографії, ескізу файлу чи стікера;
- е) Audio – аудіо файл, який можна програвати за допомогою месенджера;
- ж) Document – файл будь-якого не описаного типу;
- з) Video – описує відео файл;
- и) Voice – голосове повідомлення надіслане користувачем;
- к) VideoNote – відео повідомлення надіслане користувачем;
- л) Contact – описує контакт телефонної книги;
- м) Location – відповідає певній точці на карті;
- н) Venue – описує певне місце проведення події;
- о) UserProfilePhotos – зображення профілю користувача;
- п) File – файл, який можна завантажити. Гарантовано, що посилання буде дійсним протягом щонайменше години. Коли термін дії посилання закінчиться, можна згенерувати нове за допомогою команди «getFile». Максимальний розмір файлу для завантаження - 20 Мб;
- р) А також – команди пов'язані зі створенням індивідуальної клавіатури, відповідями від такої клавіатури, надсиланням різноманітних медіафайлів, а

також пов'язані з роботою із елементами та користувачами чатів.

Відповідно до наведених вище типів існує спосіб їх надсилання та отримання. Існують також способи полегшити розмови, наприклад редагувати зображення, підписи, закривати або відкривати розмови користувачів тощо. Важливо, що Telegram як платформа дозволяє роботам виконувати різні текстові та чатові завдання, а також підтримувати та навіть заохочувати використання роботів для різноманітних комунікацій. Справа в тому, що для покращення спілкування з користувачем він вибирає інтерфейс як еталон для створення більшої кількості роботів, лише щоб визнати, що роботи сьогодні стали невід'ємною частиною спілкування. Крім як частина месенджера доступний бот для надсилання стікерів – спеціальних картинок, які замінюють емоїї.

Гнучкість за допомогою багатьох функцій дозволяє налаштувати ботів відповідно до потреб різних користувачів. Ви можете використовувати команди, слова, інтерактивну клавіатуру або навіть голос з роботом. Відповіді користувачів можуть бути подані у вигляді газети, фотографії, відео чи документа.

Ви можете вибрати мову програмування, але спочатку вам потрібно з'ясувати, чи стосується будь-якого використання API до мови програмування, а якщо ні – виберіть іншу мову або створіть власне використання API, що, звичайно, збільшує ціну.

Python — це всеосяжна мова програмування, яка спрямована насамперед на підвищення продуктивності самого розробника програмного забезпечення, а не на код, який він пише. Простіше кажучи, Python дозволяє писати про що завгодно: веб-додатки/додатки для настільних комп'ютерів, ігри, автоматизовані інформаційні системи, інтегровані системи, рятівні системи тощо. Крім того, поріг входу низький, а код переважно чіткий і зрозумілий навіть для тих, хто ніколи на ньому не писав. Завдяки простоті коду подальше обслуговування програм, написаних на Python, пов'язаних із

Java або C, буде легшим і веселішим. І з точки зору бізнесу, це знижує витрати та підвищує ефективність співробітників. Інтерпретатор Python реалізований практично на всіх системах і операційних системах, що є незаперечною перевагою. Першою такою мовою була C, але її типи даних на різних машинах часто займали різний обсяг пам'яті, і це є перешкодою для написання великих програм. Також важлива особливість - розширюваність мови, вона має велике значення і, як пише автор Гвідо ван Россум, мова була задумана як розширювана. Це означає, що є можливість покращити мову для всіх зацікавлених програмістів. Інтерпретатор розроблено на мові програмування C, і вихідний код доступний для будь-яких змін. Якщо необхідно, ви можете встановити його у свою програму та використовувати як власну вбудовану оболонку. Або, написавши розширення Python на C і створивши програму, ви отримаєте «розширений» інтерпретатор з новими функціями.

Наступна значна перевага – наявність великого числа модулів, які можна підключити, що забезпечують різні додаткові можливості. Такі модулі зазвичай пишуться на C, на самому Python і можуть бути створені більш досвідченими програмістами. Як приклад можна навести такі модулі:

а) Numerical Python – надає програмісту більш розширені математичні можливості, такі як маніпуляції з цілими векторами і матрицями;

б) Tkinter – дає змогу будувати додатки з використанням графічного інтерфейсу користувача (GUI);

в) OpenGL – це велика бібліотека графічного моделювання дво- та тривимірних об'єктів Open Graphics Library. Єдиним недоліком, на який звертав увагу сам автор є порівняно невисока швидкість виконання Python програми. Однак, це не відіграє велику роль у порівнянні з перевагами мови при написанні програм не дуже критичних до швидкості виконання.

2.2 Мова програмування Java

Java — це об'єктно-орієнтована мова програмування для написання невеликих програм, додатків і серверного програмного забезпечення. Програми Java можна перевести в байт-код, який працює на віртуальній машині Java (JVM) - програмі, яка обробляє код додаткового модуля та надсилає інструкції апаратному забезпеченню як інтерпретатор, що означає, що байт-код обробляється набагато швидше, ніж текст. Мова Java була створена в рамках проекту створення передового програмного забезпечення для різноманітної побутової техніки. Проект був запущений на C++, але незабаром виникли різні проблеми, які найкраще вирішувати шляхом модифікації самого інструменту - мови програмування. Було продемонстровано, що вбудовані мови програмування необхідні для створення додатків, які не потрібно компілювати спеціально для кожної архітектури і можуть використовуватися на різних процесорах під різними операційними системами. У технології Java поєднуються три ключові елементи:

- Java пропонує широко використовувані невеликі програми - невеликі, надійні, потужні, незалежні від сайту активні веб-додатки, вбудовані у веб-сторінки. Скрипти Java можна налаштувати та розповсюдити споживачам так само легко, як і будь-який HTML-документ;

- Java розкриває силу об'єктивної розробки додатків і поєднує простий і знайомий синтаксис з надійним і зручним середовищем розробки. Це дозволяє великій кількості розробників швидко створювати нові програми та нові віджети;

Java надає програмісту великий набір підкатегорій для пояснення вилучення багатьох системних функцій, які використовуються під час роботи з вікнами, мережами та вводом-виводом. Ключ до цих категорій полягає в тому, що вони пропонують незалежні від сайту витяги для різноманітних системних інтерфейсів.

Переваги Java як мови програмування: об'єктно-орієнтована: в Java все є об'єктом. Доповнення легко розширити, оскільки воно засноване на об'єктивній моделі. Крім того, він не залежить від сайту: на відміну від багатьох інших мов, у тому числі C і C ++, Java при створенні була скомпільована не на машинній платформі, а на незалежному від сайту байт-кодi. Цей код надбудови поширюється через Інтернет і інтерпретується у віртуальній машині Java (JVM), на якій він зараз працює. Java проста: процес навчання і написання на мові програмування Java все ще простий. Методи аутентифікації засновані на шифруванні з відкритим ключем. Компілятор створює структурно нейтральні файли у форматі файлу, що робить агрегований код виконуваним на кількох процесорах за допомогою Java Runtime. Переносний Java: архітектурно нейтральний і незалежний від реалізацій сценаріїв - все це робить Java портативною.

Компілятор в Java написаний на ANSI C з чистою переносимістю, який є підмножиною POSIX [4 Java. URL: <http://progopedia.ru/language/java/>. (дата звернення 15.03.2022).] . Java виконує зусилля, щоб усунути помилки в різних ситуаціях, спираючись в основному на час компіляції, перевірку помилок і перевірку під час виконання. Можна писати програми, які можуть виконувати безліч завдань одночасно.

Введення в мову Java цієї конструктивної особливості дозволяє розробникам створювати налагоджені інтерактивні додатки. Java байт-код переводиться на льоту в машинні інструкції та ніде не зберігається. Роблячи процес більш швидким і аналітичним, оскільки зв'язування відбувається як додаткове з невеликою вагою процесу. Введення Just-In-Time компілятора, дозволило отримати високу продуктивність. Програмування на Java вважається більш динамічним, ніж на C або C++, так як він призначений для адаптації до мінливих умов. Програми можуть виконувати велике кількість під час обробки інформації, яка може бути використана для перевірки і дозволу доступу до об'єктів на час виконання [4 Java. URL: <http://progopedia.ru/language/java/>. (дата звернення 15.03.2022).]

Java та Python – це популярні мови, тому при розробці програм не бракуватиме ресурсів. Для новачків у програмуванні, краще буде дотримуватися Python тому, що це достатньо проста мова з синтаксисом, схожим на англійську мову. Однак для створення додатків корпоративного рівня Java підходить більше.

III. ОПИС РОЗРОБКИ ЧАТ-БОТУ

3.1 Створення чат-боту

Ми розглянемо найпростіший спосіб створити бота власними руками та пояснимо, як він працює.

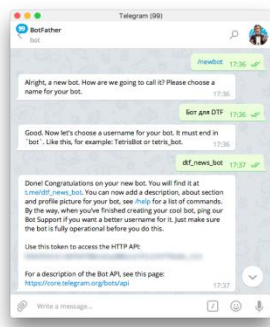
Після бурхливих обговорень в IT-пресі з приводу ефективності чат-ботів, вони зайняли свою нішу в екосистемі користувачів та компаній. Наприклад, часто проекти впроваджують ботів для оповіщення про якісь події, а служби підтримки використовують їх для того, щоб швидко відповідати на запитання клієнтів.

Ми розглянемо найпростіший спосіб створити бота власними руками та пояснимо, як він працює.

Почнемо з розробки бота, який зможе автоматично надсилати до Telegram-каналу новини компанії, опубліковані на сайті або у Facebook.

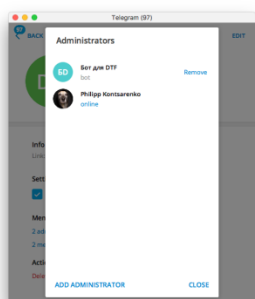
Бот в Telegram створюється за допомогою іншого бота під назвою BotFather. Відправляємо йому команду `/newbot`, вибираємо ім'я, яке відобразатиметься у списку контактів, та адресу. Наприклад, "Бот для DTF" з адресою `"dtf_news_bot"`.

Якщо адреса не зайнята, а ім'я введено правильно, BotFather надішле у відповідь повідомлення з токеном - "ключем" для доступу до створеного бота. Його треба зберегти та нікому не показувати.



Через BotFather також можна додати аватарку для бота, опис та інше.

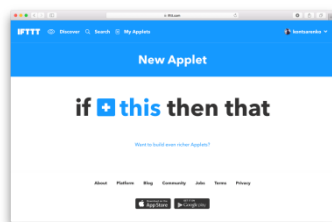
Тепер створюємо канал з будь-якою назвою та адресою, і переходимо до його налаштувань. Все, що потрібно - додати до списку адміністраторів створеного нами робота - саме він публікуватиме нотатки в канал.



Для пошуку робота можна використовувати його адресу. Наприклад, "dtf_news_bot".

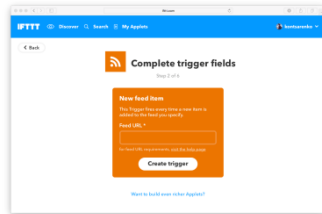
Наступний етап - навчити робота надсилати новини з сайту в створений канал. Для цього скористаємося популярним сервісом для автоматизації IFTTT.

З його допомогою можна створювати інструкції для роботи робота. У нашому випадку вона виглядає так: щоразу, коли в RSS-стрічці сайту з'являється новий запис, він має надсилати повідомлення до каналу Telegram.



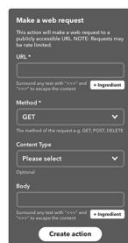
IFTTT розшифровується як If This Then That

Переходимо до розділу IFTTT "My Applets", натискаємо на кнопку "New Applet" і далі на посилання "This". Знаходимо у списку функцій тригер Feed, New feed item і вказуємо URL-адресу нашої RSS-стрічки. Наприклад, у Wordpress-сайті вона зазвичай розташована на адресі `example.com/feed/`.



Замість RSS-стрічки можна відстежувати появу нових записів у Twitter або Facebook-акаунті - для кожної функції в IFTTT передбачені окремі модулі.

Тепер переходимо до другого кроку - вибираємо дію, яка буде виконуватися при виявленні нового запису в RSS. Натискаємо на "That" і шукаємо Maker Webhook, "Make a web request" - за допомогою цього модуля можна надсилати запити до будь-яких сервісів. У нашому випадку до боту в Telegram.



У формі, що відкрилася в полі URL потрібно вказати посилання `https://api.telegram.org/botТOKEN/sendMessage`, підставивши в неї токен, згенерований на першому кроці. Метод: Post, тип контенту: `application/json`.

Body — поле для шаблону запиту, який надсилатиметься в Telegram. У ньому вказуємо, в який канал потрібно відправити повідомлення і що в ньому має бути написано:

```
{"chat_id":"@адреса_канала", "text":"{{EntryTitle}} {{FeedUrl}}"}

```

- *chat_id* — адреса каналу, до якого потрібно надіслати повідомлення. Таким чином, одного бота можна підключити відразу до кількох каналів. Як адресат можна вказати і конкретного користувача. У

такому разі замість адреси каналу необхідно вказати його ID (можна отримати за допомогою бот).

- *text* – зміст повідомлення. Наприклад, заголовок матеріалу з RSS (EntryTitle), його зміст (EntryContent) та посилання (FeedUrl).

Список доступних опцій можна переглянути за кнопкою Ingredient.

Якщо все налаштовано правильно, робота надішле повідомлення з RSS-стрічки в канал. Тригер в IFTTT має затримку, тому повідомлення, що з'явилося в RSS-стрічці, відправиться в Telegram не відразу, а через 30-60 хвилин.

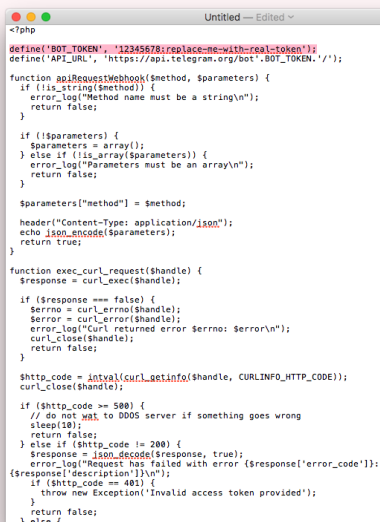
Як умову для надсилання повідомлення можна вибрати будь-який інший сценарій, доступний на IFTTT. Наприклад, Weather Underground вміє щодня надсилати повідомлення з прогнозом погоди на завтра. Тригер Stocks можна налаштувати на відправку вартості акцій під час закриття торгів.

Тепер вирішимо складніше завдання — навчимо створеного робота відповідати на повідомлення користувачів. Наприклад, надсилати по команді прайс-лист, контакти або відповідати на запитання клієнтів.

На цьому етапі знадобиться веб-хостинг та сертифікат SSL, який можна отримати безкоштовно за допомогою сервісу Let's Encrypt.

Найзручніше створити окремий піддомен для бота - наприклад, bot.example.com - і розмістити на ньому один файл index.php. У середині файлу розміщуємо код найпростішого робота з сайту Telegram.

Якщо у вас хостинг Vscale, то можна створити для бота окремий сервер. Це робиться за допомогою одного кліка на панелі керування.



```

<?php
define('BOT_TOKEN', '12345678:replace-me-with-real-token');
define('API_URL', 'https://api.telegram.org/bot'.BOT_TOKEN.'/');

function apiRequest($method, $parameters) {
    if (!is_string($method)) {
        error_log("Method name must be a string\n");
        return false;
    }

    if (!$parameters) {
        $parameters = array();
    } else if (!is_array($parameters)) {
        error_log("Parameters must be an array\n");
        return false;
    }

    $parameters["method"] = $method;
    header("Content-Type: application/json");
    echo json_encode($parameters);
    return true;
}

function exec_curl_request($handle) {
    $response = curl_exec($handle);

    if ($response === false) {
        $errno = curl_errno($handle);
        $error = curl_error($handle);
        error_log("Curl returned error $errno: $error\n");
        curl_close($handle);
        return false;
    }

    $http_code = intval(curl_getinfo($handle, CURLINFO_HTTP_CODE));
    curl_close($handle);

    if ($http_code >= 500) {
        // do not wait to DDOS server if something goes wrong
        sleep(5);
        return false;
    } else if ($http_code != 200) {
        $response = json_decode($response, true);
        error_log("Request has failed with error {$response['error_code']}:
    {$response['description']}\n");
    }
    if ($http_code == 401) {
        throw new Exception("Invalid access token provided");
    }
    return true;
} else {

```

У код бота потрібно внести лише дві зміни:

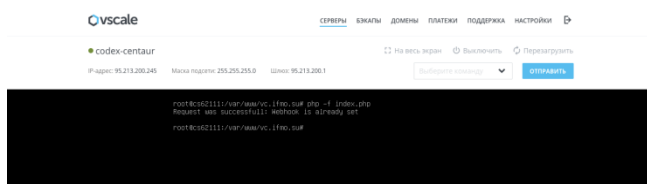
- у рядку `define('BOT_TOKEN', '12345678:replace-me-with-real-token');` замість `12345678:replace-me-with-real-token` написати токен, отриманий на першому кроці;
- у рядку `define('WEBHOOK_URL', 'https://my-site.example.com/secret-path-for-webhooks/');` замість `https://my-site.example.com/secret-path-for-webhooks/` вказати URL-адресу файлу з кодом для бота: `https://bot.example.com/index.php`.

Тепер необхідно зв'язати Telegram та файл на сервері, щоб запити, надіслані боту в месенджері, обробляв наш скрипт.

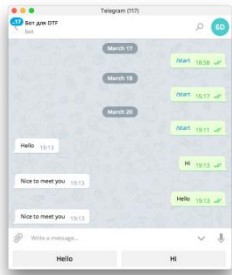
Для цього знадобиться консоль. У різних хостинг-провайдерів вона може бути в різних розділах інтерфейсу управління сайтом. Крім того, можна скористатися програмою Terminal на macOS, ввівши через неї команду `ssh імя_пользователя@адрес_домена`.

Після введення пароля, друкуємо для нашого піддомену просту команду:

```
php -f /var/www/bot.example.com/index.php
```



Якщо все зроблено правильно, бот відразу надішле в Telegram повідомлення "Hello".



Додамо різні сценарії відповідей бота. Відкриваємо створений раніше файл `index.php` і знаходимо у ньому рядок:

```
} else if ($text === "Hello" || $text === "Hi") {
    apiRequest("sendMessage", array('chat_id' => $chat_id, "text" => 'Nice to meet you'));
```

- Hello і Hi - текст користувача, який бот повинен відреагувати.
- Nice to meet you - відповідь бота на надіслане користувачем повідомлення.

Нижче у коді додаємо додаткові відповіді. Наприклад, щоб залежно від відправленого слова, бот надсилав необхідну інформацію користувачу:

```
} else if ($text === "Ціни") {
    apiRequest("sendMessage", array(' chat_id' => $chat_id, "text" => 'Прайс-лист можна завантажити за посиланням example.com/price'));
} else if ($text === "Контакти") {
    apiRequest("sendMessage", array('chat_id' => $chat_id, "text" => 'Контактна адреса для зв'язку — secret@vc.ru'));
```

Можемо вказати будь-яку кількість фраз та відповідей на них. Також можна змінити кнопки, доступні користувачеві під час роботи з роботом. Знаходимо рядок коду `'keyboard' => array(array('Hello', 'Hi'))` і замість *Hello* і *Hi* вказуємо «Ціни» та «Контакти».

Після збереження файлу зміни відразу набудуть чинності.

3.2 Тестування чат-боту

Оскільки чат-бот створено для месенджера Telegram, для тестування використовувались пристрої, що підтримують саме цей месенджер. Telegram можна використовувати на ПК, ноутбуках чи мобільних пристроях. Також, якщо користувач не бажає встановлювати додаток до себе на пристрій – у Telegram є офіційний сайт web.telegram.org, який має більшість функцій, як і в додатку.

Тестування проводилось за допомогою ноутбука, що має такі характеристики:

- Операційна система: Microsoft Windows 10;
- Оперативна пам'ять: 16 ГБ;
- Відеоадаптер: AMD Radeon R7 460;
- Процесор: Intel Core i7-8550U;
- Вбудована пам'ять: 1,8 ТБ;
- Telegram Desktop 2.1.6

Процедура тестування виконувалась наступним чином: спочатку знаходимо створений чат-бот. Для цього потрібно знайти обліковий запис, що називається_bot. Як тільки його знайдено, відкриваємо з ним діалог. Діалог буде пустим, і він не розпочнеться до того моменту, доки користувач не натисне кнопку “Запустити”. Після цього бот відправить повідомлення-привітання.

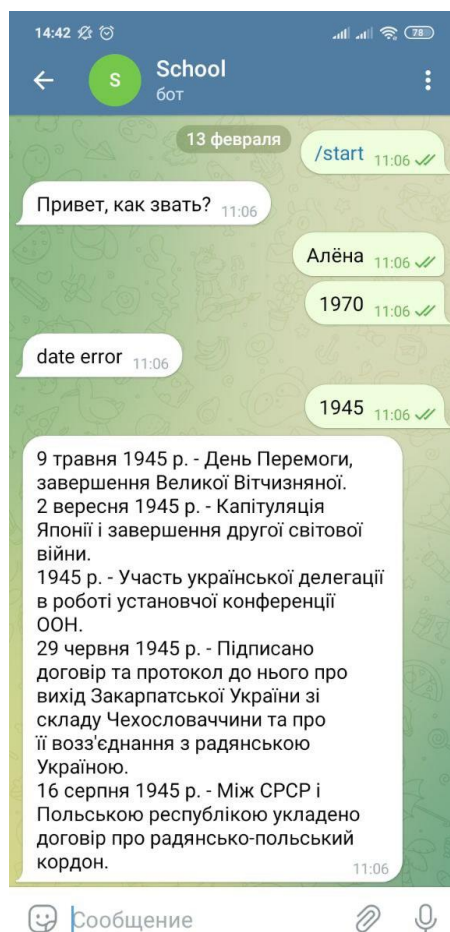
Далі треба написати в одному повідомленні команду /search і назву інформації, яку користувач бажає отримати.

Після того, як бот відправить користувачу результати пошуку, користувач повинен обрати один із наданих йому варіантів. При тестуванні бота на цьому прикладі було обрано перший варіант.

Отриманий при тестуванні розробленого бота результат доводить, що чат-бот виконує поставлені у дипломному проєкті задачі, а саме: здійснює

пошук за запитом користувача, надсилає результати пошуку користувачу, пропонує вибір варіанта відповідей користувачем та надає остаточний результат користувачу, розширивши відповідь додатковою інформацією.

Тестування чат-боту наглядно:



Вигляд бази даних для чат-боту(помічника) з історії України для

date_	history_date
860	860 р. - Перший похід русичів, очолюваних князем Аскольдом на Константинополь (Царгород).
882	882 р. - Убивство князя Аскольда та захоплення князівської влади в Києві Олегом. Виникнення держави Київська Русь.
907	907, 911 р. - Походи князя Олега на Царгород.
911	907, 911 р. - Походи князя Олега на Царгород.
941	941, 944 р. - Походи князя Ігоря на Візантію.
944	941, 944 р. - Походи князя Ігоря на Візантію.
945	945 р. - Повстання деревлян і вбивство князя Ігоря.
946	946 р. - Посольство княгині Ольги до Царгорода.
968	968 р. - Перший Балканський похід князя Святослава.
969	969–971 рр. - Другий Балканський похід князя Святослава.
971	969–971 рр. - Другий Балканський похід князя Святослава.
988	988 р. - Початок запровадження князем Володимиром християнства як державної релігії Київської Русі.
1036	1036 р. - Розгром князем Ярославом Мудрим печенігів під Києвом.
1054	1054 р. - Перша згадка в літописі Києво-Печерського монастиря.
1054	1054–1113 рр. - Співправління трьох братів Ярославичів – Ізяслава, Святослава та Всеволода.
1113	1054–1113 рр. - Співправління трьох братів Ярославичів – Ізяслава, Святослава та Всеволода.
1068	1068 р. - Повстання в Києві проти Ярославичів.
1097	1097 р. - Любецький з'їзд князів. 1097 р. - Народне повстання в Києві.

середньої школи.

В

ИГЛЯД

КОДУ:

```
import telebot
import sqlite3
import asyncio

bot = telebot.TeleBot('5232065310:AAGxgH290jghEyGk13-UycHKJFkcK7hNGws')
print ("Все ок")
sqlite_connection = sqlite3.connect('diplom.db', check_same_thread=False)
cursor = sqlite_connection.cursor()

@bot.message_handler(commands=['start'])
def start_message(message):
    bot.send_message(message.chat.id, 'Вітаю ' + '\n' + 'Вкажіть дату у форматі (Рік), я перевірю наявність важливих подій у історії України')
    try:

        @bot.message_handler(content_types=['text'])
        def get_text_messages(message):

            if "!" in message.text:
                book()
            if message.text.isdigit() == True:
                ret = 'SELECT rowid, * FROM history_list WHERE date_ = (?)'
                cursor.execute(ret, (message.text,))
                i = cursor.fetchall()
                for r in i:
                    bot.send_message(message.chat.id, r[2])
                return
            else:
                bot.send_message(message.chat.id, "date error")

    except sqlite3.Error as error:
        print ('Ошибка подключения: ', error)
        sqlite_connection.close()
        print ("Соединение с БД закрыто.")
bot.polling(none_stop=True, interval=2, timeout=9999)
```


ВИСНОВКИ

Технологічний прогрес сприяє виведенню кожної компанії на якісно новий рівень у комунікації з клієнтами. Одним із методів спілкування стають чат-боти, які набирають все більшої популярності серед власників та керівників бізнесу незалежно від сфери діяльності компанії. Це пояснюється тим, що віртуальні помічники забезпечують для компанії цілодобову безперебійну технічну підтримку, допомагають охопити більше покупців, збільшують швидкість відповіді на поширені запитання від клієнтів та надають можливість тісно співпрацювати з менеджерами компанії в разі необхідності, що в рази збільшує ефективність роботи працівників. Компаніям слід зосередитися на швидкості відповідей та встановленні «живого» спілкування, використовуючи чат-боти, адже клієнтам важливо відчувати значущість при обслуговуванні. Завдяки боту можливо в рази примножити позитивний відгук про сервіс бізнесу від його клієнтів.

В ході виконання курсового проєкту проведено аналіз предметної області та визначено головні принципи роботи чат-ботів, а також основні методики щодо їх створення.

Проаналізовано загальні принципи та засоби побудови чат-боту для месенджера Telegram та програмні засоби для його розроблення. Детально розглянуто ботів – програми які дозволяють автоматично спілкуватися з користувачами. Проаналізовано вже існуючих ботів та наведено приклади їх використання. В результаті, було виявлено необхідність популяризації використання розпізнавання команд із повідомлень користувачів, при використанні ботів як засобів роботи із клієнтами. Описано загальні принципи розроблення сервісу для платформи Телеграм та описано процес розробки бота.

Виконано порівняльний аналіз декількох мов програмування. У якості основних засобів розроблення обрано мову програмування Python, Telegram Bot API.

Принципова відмінність між розробкою чат-ботів за допомогою конструктора та мови програмування полягає у складності реалізації поставленої задачі. Створення віртуального помічника за допомогою Python потребує навичок програмування, а головне – проектування масштабованих систем задля подальшого розвитку програмного забезпечення. Це дає можливість впроваджувати нові структурні елементи незалежно від їх складності в реалізації. Проте використання веб-сервісів для реалізації чат-боту буде корисним для створення тестового варіанту, який можна використати для аналізу інтегрування даного підходу в бізнес-модель компанії

Як результат проектування, створено чат-бот для конференцій. Виконуючи запит користувача, чат-бот месенджера Telegram здійснює пошук потрібної композиції та детальної інформації про неї і надсилає отримані результати користувачу у зручному та наглядному вигляді.

Ця розробка є доказом концепції такого підходу до створення ботів для месенджерів та ілюструє можливу швидкість та гнучкість розробки

Список використаних джерел

1. Java. URL: <http://progopedia.ru/language/java/>. (дата звернення 20.03.2022).
2. Инструкция: как создавать ботов в Telegram. [Электронный ресурс] – Режим доступа: <https://habr.com/ru/post/262247/>
3. Чат-боти для навчання: огляд найпопулярніших та особливості використання URL: <https://teach-hub.com/chat-boty-dlia-navchannia-ohliad-na-populiarnishykh-ta-osoblyvosti-vykorystannia/>
4. Джек Траут «Дифференцируйся або вмирай! Вживання в епоху вбивчої конкуренції », 2010, ст 145-159
5. Telegram FAQ [Электронный ресурс]. – Режим доступа: <https://telegram.org/faq>
6. Акулич М. Чат-боты и маркетинг. URL <http://libs.ru/book/540177/read/>
7. Беловодская Е.А., Лагута К.А. Системное исследование использования чат-бота в коммуникации с клиентами. Формування ринкових відносин в Україні. №5 (228)/2020. – Режим доступа: https://er.knutd.edu.ua/bitstream/123456789/15953/3/%D0%91%D1%96%D0%B%D0%BE%D0%B2%D0%BE%D0%B4%D1%81%D1%8C%D0%BA%D0%B0_%D0%9B%D0%B0%D0%B3%D1%83%D1%82%D0%B0.pdf
8. Джек Траут «Дифференцируйся або вмирай! Вживання в епоху вбивчої конкуренції », 2010, ст 145-159
9. Инструкция: как создавать ботов в Telegram. [Электронный ресурс] – Режим доступа: <https://habr.com/ru/post/262247/>
10. Історія розвитку чат ботів. URL: <https://nv.ua/techno/technoblogs/chtotakoe-chat-boty-istoriya-razvitiya-50058587.html>
11. Исследование топ-50 чат-бот платформ и виртуальных ассистентов 2019 года. – Режим доступа: <https://habr.com/ru/company/mipt/blog/467183/>
12. Как создать чат-бота с нуля на Python: подробная инструкция [Электронный ресурс] – Режим доступа: <https://highload.today/kak-sozdat-telegram-bot-na-python-poshagovoe-rukovodstvo/>

13. Кузнецов В.В. Перспективы развития чат-ботов. Успехи современной науки. 2016. № 12, С. 16–19.
14. Официальный сайт Telegtam [Электронный ресурс]/ API – Режим доступа: <https://core.telegram.org/api>
15. Прикладний програмний інтерфейс [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Прикладний_програмний_інтерфейс.
16. Сріні Джанарсанам. Розробка чат-ботів і розмовних інтерфейсів. С. 250-300.
17. Технологии создания и применения чат ботов [Электронный ресурс]/ Новые технологии – Режим доступа: <https://cyberleninka.ru/article/v/tehnologii-sozdaniya-i-primeneniya-chatbotov>
18. ТОП-25 корисних чат-ботів для різних випадків. URL: <https://ilounge.ua/review/top-25-poleznyh-chat-botov-dlya-raznyh-sluchaev>
19. Як створити чат-бот для Telegram-каналу – інструкція для адміністраторів. URL: <https://netpeak.net/ru/blog/kak-sozdat-chat-bot-dlyatelegraminstruktsiya-dlya-administratorov-kanalov/>