

Міністерство освіти і науки України
Чернівецький національний університету імені Юрія Федьковича»
Відокремлений структурний підрозділ «Фаховий коледж Чернівецького
національного університету імені Юрія Федьковича»

Природничче відділення
Циклова комісія комп'ютерної інженерії

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітньо-професійного ступеня «фаховий молодший бакалавр»
зі спеціальності 123 Комп'ютерна інженерія
підготовки за освітньо-професійною програмою «Комп'ютерна інженерія»
на тему: **«Розробка інтелектуального чат-боту»**

Виконав (ла):

студент (ка) 4-го курсу Страдецький Роман Юрійович
(прізвище, ім'я та по батькові) (підпис)

Керівник:

Мельничук А.Ю.
(науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Рецензент:

Букурос О.В.
(науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

До захисту допущено:

Протокол засідання циклової комісії № _

від „__” _____ 2024 р.

Голова циклової комісії _____ Олександр ТАЩУК

Чернівці, 2024

Завдання та календарний план роботи

Міністерство освіти і науки України
Чернівецький національний університету імені Юрія Федьковича»
Відокремлений структурний підрозділ «Фаховий коледж
Чернівецького національного університету імені Юрія Федьковича»

Затверджую
Голова циклової комісії,
Олександр ТАЩУК.

«___» _____ 2023 р.

Завдання

На кваліфікаційну роботу	Страдецький Роман Юрійович
Тема кваліфікаційної роботи	Розробка інтелектуального чат-боту
Постановка завдання в короткій формі	Розробка та програмна реалізація інтелектуального чат-боту для ВСП «Фаховий коледж ЧНУ ім. Ю.Федьковича».
Вихідні дані (2-3 адреси сайтів, матеріали яких рекомендує керівник кваліфікаційної роботи)	https://pytba.readthedocs.io/en/latest/ https://core.telegram.org/bots/api http://college-chnu.cv.ua/

Календарний план роботи

Дата отримання версії звіту керівником	Етап виконання роботи	Виконання (зазначає керівник)
10.10.23	Отримання завдання до кваліфікаційної роботи.	
06.11.23	Огляд літератури, присвяченої розробці інтелектуального чат-боту.	
30.11.23	Аналіз існуючих чат-ботів для освітніх закладів. Оформлення розділу 1.	
15.12.23	Вибір технологій та інструментів для розробки.	
29.12.23	Написання теоретичного матеріалу по використовуваних модулях.	
15.01.24	Проектування та розробка телеграм-боту. Оформлення розділу 2.	
16.02.24	Практична реалізація функцій та формування вмісту бота. Оформлення розділу 3.	
18.03.24	Тестування та перевірка працездатності бота. Зворотній зв'язок. Оформлення розділу 4.	
05.04.24	Перспективи впровадження та розвитку бота.	

	Оформлення розділу 5.	
26.04.24	Оформлення та редагування кваліфікаційної роботи.	
<i>за графіком</i>	Подання роботи на перевірку Інтернет-сервісом Unichesk.	
<i>за графіком</i>	Подання роботи на рецензію	
<i>за графіком</i>	Представлення роботи на засіданні Циклової комісії	
<i>за графіком</i>	Захист кваліфікаційної роботи	

Дата видачі завдання _____.

Студент

Роман СТРАДЕЦЬКИЙ

Керівник (П.І.Б)

Христина МЕЛЬНИЧУК

Андрій МЕЛЬНИЧУК

АНОТАЦІЯ

В цій роботі проведено розробку та програмну реалізацію телеграм-бота для коледжу з метою покращення комунікації зі студентами, абітурієнтами та викладачами. Робота актуальна у зв'язку з необхідністю забезпечення швидкого та зручного доступу до інформації про навчальний процес, події та ресурси коледжу.

У роботі використано метод аналізу існуючих телеграм-ботів для освітніх закладів для визначення оптимального функціоналу. На основі цього було сформульовано завдання та визначено функціональні вимоги до розробленого бота.

Для розробки бота були використані Python та бібліотека Telebot для взаємодії з API Telegram. Основні функції бота включають відправлення користувачам інформації про навчальний процес, оголошення, розклад занять, інформацію про вступ та інші корисні сервіси.

Розроблений бот пройшов успішне тестування функціоналу, включаючи взаємодію з користувачами та обробку їхніх запитів. Зібрано та проаналізовано відгуки користувачів для подальшого вдосконалення бота.

Робота складається з 5 розділів та містить загалом 42 сторінки, 9 рисунків та 7 посилань на використані джерела.

Ця робота відображає процес розробки та реалізації телеграм-бота для коледжу з урахуванням сучасних стандартів та потреб користувачів, а також може бути використана в якості основи для подальшого розвитку та вдосконалення подібних проектів.

Основні ключові слова: інтелектуальний чат-бот, чат-бот, телеграм-бот, освітній заклад, комунікація, Python, Telebot.

ЗМІСТ

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ	7
ВСТУП.....	8
1. ОГЛЯД ІСНУЮЧИХ ТЕЛЕГРАМ БОТІВ ДЛЯ ОСВІТНІХ ЗАКЛАДІВ.....	9
1.1 Аналіз функціональності та можливостей існуючих телеграм-ботів для коледжів та університетів	10
Висновки з розділу 1:	11
2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ТЕЛЕГРАМ БОТА ДЛЯ КОЛЕДЖУ	12
2.1. Постановка завдання та визначення функціональних вимог	12
2.1.1. Аналіз потреб користувачів	12
2.1.2. Визначення основних функцій бота	12
2.1.3. Створення специфікації вимог	13
2.2. Вибір технологій та інструментів для розробки	13
2.2.1. Аналіз доступних технологій	13
2.2.2. Вибір мови програмування	13
2.2.3. Вибір фреймворку та інструментів	14
2.3. Архітектура системи та структура бота.....	14
2.3.1. Компоненти системи	14
2.3.2. Зв'язки між компонентами	15
2.3.3. Структура бота.....	15
Висновки до розділу 2:	16
3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ФУНКЦІЙ ТА ФОРМУВАННЯ ВМІСТУ БОТА.....	17
3.1. Реалізація команд та відповідей на них	18
3.1.1. Визначення списку команд.....	18
3.1.2. Програмування логіки відповідей.....	19
3.1.3. Обробка варіантів запитань та сценаріїв	19
3.1.4. Тестування та вдосконалення.....	20
3.2. Формування вмісту для різних категорій користувачів (абітурієнти, студенти)	20
3.2.1. Абітурієнти.....	20
3.2.2. Студенти	21
3.2.4. Формат відправки інформації.....	21
3.2.5. Адаптація контенту	22
Висновки з розділу 3:.....	22
4. ТЕСТУВАННЯ ТА ОТРИМАННЯ ЗВОТНОГО ЗВ'ЯЗКУ	23
4.1. Планування тестування	23
Висновки з розділу 4:.....	25
5. ПЕРСПЕКТИВИ ВПРОВАДЖЕННЯ ТА РОЗВИТКУ	26
5.1. Можливості подальшого розвитку та удосконалення бота.....	27

5.2. Плани щодо впровадження бота в роботу коледжу.....	28
Висновки з розділу 5:.....	29
ВИСНОВКИ	31
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	32
ДОДАТКИ	33
Додаток А. Лістинг боту.....	33
Додаток Б. Тестування боту	41
Додаток В. Результати тестування боту	42

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

1. Telegram - Месенджер, який використовується для обміну повідомленнями та мультимедійними файлами, створення чатів, каналів та телеграм ботів.
2. Телеграм бот - Автоматизований обліковий запис у месенджері Telegram, що надає різноманітні сервіси через інтерактивні функції.
3. API (Application Programming Interface) - Інтерфейс програмування застосунків, який дозволяє різним програмам взаємодіяти між собою.
4. Chat ID - Унікальний ідентифікатор чату у Telegram, використовується для надсилання повідомлень у певний чат.
5. Python - Високорівнева мова програмування, що використовується для написання коду телеграм бота.
6. Бібліотека - Набір готових модулів або функцій, які можуть бути використані в програмуванні.
7. Telegram Bot API - Інтерфейс для створення телеграм ботів, який надає доступ до функцій Telegram через програмні засоби.
8. Inline Keyboard - Вбудована клавіатура у Telegram, яка дозволяє створювати інтерактивні кнопки у повідомленнях.
- 9 Token - Унікальний ключ, що використовується для автентифікації та доступу до Telegram Bot API.
10. IDE (Integrated Development Environment) - Інтегроване середовище розробки програмного забезпечення.
11. Фреймворк - Набір інструментів та бібліотек для спрощення розробки програмного забезпечення.
12. Система контролю версій - Інструмент для відстеження змін у коді та управління різними версіями проекту.

ВСТУП

У сучасному цифровому світі використання технологій для автоматизації та поліпшення комунікаційних процесів стає не лише розумним кроком, але й необхідністю. В контексті освіти та навчання, відповідно, з'являється потреба в інструментах, які забезпечують ефективну взаємодію між студентами, викладачами та адміністрацією навчального закладу.

Використання месенджерів, таких як Telegram, для створення телеграм ботів, є одним із перспективних напрямків у цьому контексті. Телеграм бот - це автоматизований обліковий запис, який здатний надавати різноманітні сервіси через месенджер Telegram. Це може бути все: від надання інформації та відповідей на запитання до проведення опитувань та розсилки повідомлень.

Метою даної кваліфікаційної роботи є розробка та реалізація телеграм бота для коледжу з метою спрощення комунікації між учасниками навчального процесу та забезпечення доступу до актуальної інформації про навчальний заклад. Основні завдання включають в себе аналіз існуючих рішень у сфері телеграм ботів для освітніх закладів, проектування та розробку функціоналу бота, реалізацію програмного забезпечення та тестування його працездатності.

В рамках роботи буде розглянуто процес проектування та реалізації функціоналу бота, зокрема створення архітектури, вибір технологій та інструментів програмування, а також оцінка перспектив впровадження та розвитку розробленого рішення.

Ця кваліфікаційна робота має на меті допомогти у вдосконаленні комунікаційних процесів в коледжі, зробивши їх більш зручними та ефективними для всіх учасників навчального процесу.

1. ОГЛЯД ІСНУЮЧИХ ТЕЛЕГРАМ БОТІВ ДЛЯ ОСВІТНІХ ЗАКЛАДІВ

У сучасному освітньому середовищі використання телеграм ботів набуває все більшої популярності як ефективний інструмент для поліпшення комунікаційних процесів та надання різноманітних сервісів студентам, викладачам та адміністрації навчального закладу. Перш ніж розпочати проектування та реалізацію власного телеграм бота для коледжу, важливо ознайомитися з існуючими рішеннями у цій сфері та зрозуміти їхні можливості та обмеження.

Існують різноманітні телеграм боти для освітніх закладів, які надають різноманітні сервіси, такі як:

1. Інформаційна підтримка: Деякі боти просто надають інформацію про розклад занять, важливі події, оголошення та інші актуальні питання навчального процесу.

2. Онлайн-консультації: Деякі боти дозволяють студентам та викладачам отримувати консультації з певних предметів чи проблем через чат-бота.

3. Проведення тестів та опитувань: Деякі боти можуть створювати та проводити тести або опитування серед студентів.

4. Нагадування та планування: Деякі боти дозволяють створювати нагадування про важливі події, зустрічі чи завдання.

5. Звітність та аналітика: Деякі боти можуть забезпечувати звітність про активність користувачів, їхні досягнення та іншу статистику.

Даний огляд існуючих телеграм ботів для освітніх закладів допоможе зрозуміти, які можливості вже існують та які сервіси можуть бути корисними для реалізації власного проекту для коледжу. Такий аналіз допоможе вибрати найбільш ефективні рішення для досягнення поставлених цілей та врахувати найкращі практики при проектуванні та розробці власного телеграм бота для коледжу.

Опрацьований мною телеграм-бот належить Київському столичному

університету імені Бориса Грінченка. (рис. 1.1)

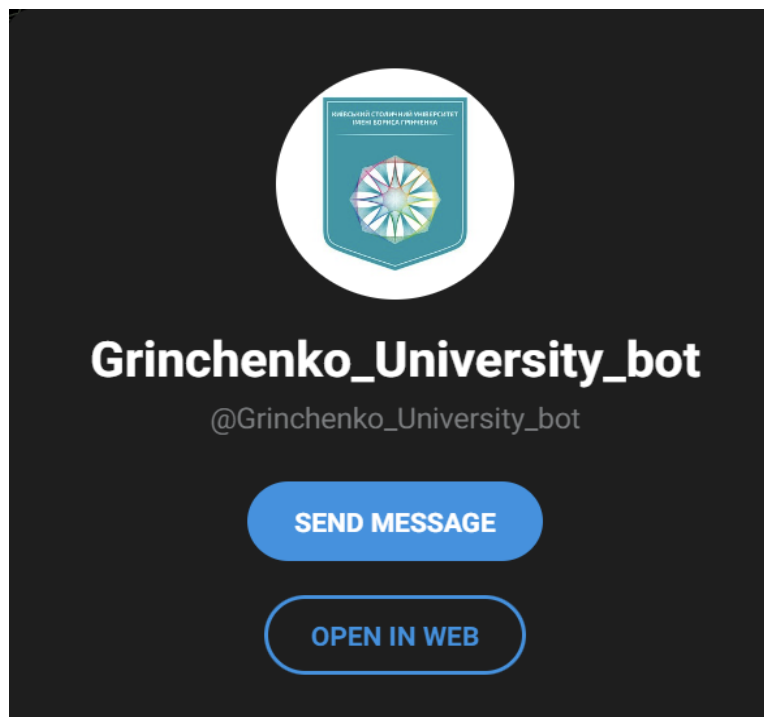


Рис.1.1. Телеграм-бот КСУ ім. Б.Грінченка

1.1 Аналіз функціональності та можливостей існуючих телеграм-ботів для коледжів та університетів

Нижче наведено функціональність знайденого телеграм-бота для університету (див. рис. 1.2).

Даний телеграм бот для університету надає широкий спектр сервісів та можливостей, спрямованих на полегшення комунікації та забезпечення зручності для учасників навчального процесу.

1. Інформаційне сповіщення:

- Бот регулярно надсилає оновлення стосовно розкладу занять, змін у графіку, важливих подій та оголошень коледжу.

2. Консультації та підтримка:

- Студенти та викладачі можуть отримати консультації з різних питань, звернувшись до бота через чат. Бот надає відповіді на питання щодо навчального процесу, правил та положень коледжу.

3. Опитування та тести:

- Бот проводить опитування серед студентів з метою збору думок,

відгуків та інших даних, а також може надавати можливість проходження тестів для перевірки знань.

4. Нагадування та планування:

- Користувачі можуть створювати нагадування про важливі події, завдання та інші справи. Бот надсилає сповіщення згідно встановленого розкладу.

5. Звітність та аналітика:

- Бот забезпечує звітність про активність користувачів, їхні досягнення та іншу статистику, що дозволяє адміністрації коледжу аналізувати та вдосконалювати навчальний процес.

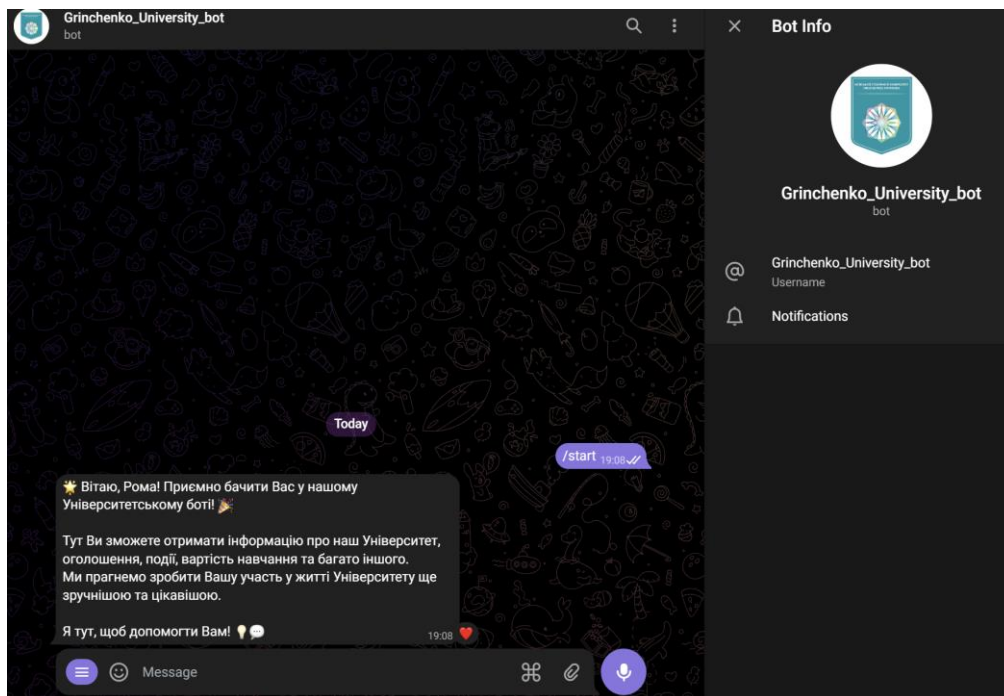


Рис.1.2. Початок роботи телеграм-бота КСУ ім. Б.Грінченка

Висновки з розділу 1:

Знайдений телеграм-бот Київського столичного університету імені Бориса Грінченка може слугувати важливим джерелом інформації та вдосконалення комунікації в коледжі. Ми можемо використати його як зразок для проектування та реалізації власного бота, враховуючи його функціональність та особливості взаємодії з користувачами.

2. ПРОЕКТУВАННЯ ТА РОЗРОБКА ТЕЛЕГРАМ БОТА ДЛЯ КОЛЕДЖУ

2.1. Постановка завдання та визначення функціональних вимог

На етапі постановки завдання та визначення функціональних вимог ми глибоко досліджуємо потреби користувачів та створюємо детальний перелік функцій, які повинен виконувати телеграм бот для коледжу. Цей етап є критично важливим, оскільки правильне визначення вимог визначає успішність та ефективність роботи бота.

2.1.1. Аналіз потреб користувачів

Починаючи з аналізу потреб користувачів, ми проводимо широке дослідження, спрямоване на з'ясування очікувань та вимог різних груп користувачів: студентів, викладачів та адміністрації коледжу. Ми вивчаємо їхні потреби, проблеми та очікування щодо функціоналу бота. Для цього ми можемо проводити опитування, інтерв'ю з представниками цих груп або аналізувати вже існуючі системи комунікації та інформаційного обміну.

2.1.2. Визначення основних функцій бота

На основі результатів аналізу ми визначаємо основні функції, які має виконувати телеграм бот для коледжу. Це може включати такі можливості, як:

- Надання розкладу занять для студентів та викладачів.
- Оповіщення про важливі події, зміни в розкладі або інші оголошення.
- Надання інформації про викладачів, їхні контактні дані та консультаційні години.
- Доступ до загальної інформації про коледж: адреса, телефон, електронна пошта тощо.
- Можливість проведення опитувань, збору відгуків та зворотного зв'язку від користувачів.

2.1.3. Створення специфікації вимог

На цьому етапі ми створюємо детальну специфікацію вимог, в якій описується кожна функція бота, її параметри, взаємозв'язки та особливості реалізації. У специфікації також визначаються вимоги до інтерфейсу користувача, методи взаємодії з системою, обмеження та вимоги до безпеки та конфіденційності даних. Цей документ є основним засобом комунікації між командою розробників та замовником і гарантує однозначне розуміння вимог до системи.

2.2. Вибір технологій та інструментів для розробки

При виборі технологій та інструментів для розробки телеграм бота для коледжу необхідно ретельно врахувати ряд факторів, що включають потреби користувачів, складність функціоналу, швидкість реакції, масштаби проекту та інші вимоги. Детально розглянемо кожен аспект вибору технологій та інструментів.

2.2.1. Аналіз доступних технологій

Перший етап включає дослідження різноманітних технологічних стеків, що можуть бути використані для створення телеграм бота. Це включає аналіз мов програмування, фреймворків, бібліотек та інших інструментів, доступних для розробки ботів. Ми оцінюємо їхні переваги та недоліки з точки зору виконання завдань бота, його масштабу та інтеграції з іншими системами.

2.2.2. Вибір мови програмування

Важливим аспектом є вибір мови програмування, яка найбільш підходить для реалізації функціоналу бота. Ми розглядаємо такі мови, як Python, JavaScript, Java або C#, оцінюючи їхню ефективність, швидкодію, розширюваність та підтримку спільноти. Наприклад, Python є популярним

вибором завдяки широкій підтримці та наявності різноманітних бібліотек для роботи з Telegram API[3,4].

2.2.3. Вибір фреймворку та інструментів

Після вибору мови програмування ми обираємо відповідні фреймворки та інструменти для розробки. Наприклад, для Python можуть використовуватися фреймворки, такі як Telegraf або Aiogram, що дозволяють зручно працювати з Telegram API та реалізовувати різноманітний функціонал[4]. Для JavaScript можна використовувати Telegraf.js. Крім того, ми оцінюємо можливість використання спеціалізованих платформ або сервісів для швидкого розгортання та хостингу бота, таких як Heroku, AWS або Google Cloud Platform[3].

2.3. Архітектура системи та структура бота

В процесі проектування архітектури системи та структури телеграм бота для коледжу важливо ретельно визначити й організувати компоненти, що складатимуть систему, а також забезпечити їх взаємодію та взаємодію з Telegram API[5]. Розглянемо детально кожен аспект:

2.3.1. Компоненти системи

Першим кроком є визначення основних компонентів системи. Це можуть бути, наприклад:

- *Telegram Bot API Interface:*

Цей компонент відповідає за взаємодію з Telegram API. Він отримує вхідні повідомлення від користувачів, обробляє їх та відправляє відповіді назад.

- *Бізнес-логіка бота:*

Цей компонент містить логіку, яка визначає, які дії повинен виконувати бот відповідно до вхідних повідомлень. Він аналізує текст повідомлень,

виконує необхідні операції (наприклад, відправлення інформації або запитів до бази даних) та генерує відповіді.

- *База даних (якщо необхідно):*

Якщо для роботи бота потрібно зберігати та обробляти дані (наприклад, розклад занять, контактні дані користувачів тощо), то вона використовується для зберігання цих даних та доступу до них.

- *Інтерфейси взаємодії з іншими системами (опціонально):*

Якщо бот повинен отримувати або відправляти дані іншим системам (наприклад, системі управління навчальним процесом), то ці інтерфейси також реалізуються.

2.3.2. Зв'язки між компонентами

Після визначення компонентів системи важливо встановити зв'язки між ними. Наприклад [5]:

- *Telegram Bot API Interface* може викликати методи бізнес-логіки бота для обробки вхідних повідомлень та генерації відповідей.
- *Бізнес-логіка бота* може використовувати базу даних для зберігання та отримання даних, необхідних для виконання різних завдань.

2.3.3. Структура бота

На останньому етапі визначається структура самого бота, тобто його функціональність та логічне поділ на модулі або команди. Наприклад:

- *Реєстрація та авторизація користувачів.*
- *Відправлення розкладу занять або інших корисних даних.*
- *Відповіді на загальні запитання користувачів.*
- *Опитування та збір зворотного зв'язку від користувачів.*

Кожен з цих модулів або команд виконує конкретну функцію і має відповідати на певний тип запиту користувача. Ретельно продумана структура дозволить зробити бота зручним та ефективним у використанні.

Висновки до розділу 2:

В розділі 2 моєї дипломної роботи було визначено вимоги до телеграм бота для коледжу, вибрано технології для розробки, розроблено архітектуру системи та структуру бота. Це дозволить ефективно реалізувати функціонал, спрямований на задоволення потреб користувачів.

3. ПРАКТИЧНА РЕАЛІЗАЦІЯ ФУНКЦІЙ ТА ФОРМУВАННЯ ВМІСТУ БОТА

У цьому розділі розглянуто процес реалізації функцій та формування вмісту телеграм бота для коледжу. Це включає в себе визначення конкретних функціональних можливостей бота, програмування його логіки, розробку відповідей на типові запитання, а також створення та наповнення вмістом різних команд та модулів.

Визначення функціональних можливостей

Першим кроком у реалізації бота є визначення його функціональних можливостей. Це можуть бути:

- Авторизація та реєстрація:

Реалізація можливості авторизації користувачів за допомогою їхніх студентських облікових записів або інших ідентифікаторів.

- Відправлення розкладу занять:

Бот може надавати користувачам актуальну інформацію про розклад занять, дати та час проведення лекцій, практичних занять тощо.

- Відповіді на загальні запитання:

Бот може мати базу знань, що містить відповіді на часті запитання студентів щодо навчального процесу, правил і політики коледжу тощо.

- Опитування та збір зворотного зв'язку:

Бот може проводити опитування серед студентів, викладачів та інших учасників навчального процесу для збору думок, відгуків та пропозицій щодо поліпшення роботи коледжу.

Програмування логіки бота

Після визначення функціональних можливостей бота розробляється його логіка. Це включає в себе написання коду, який обробляє вхідні повідомлення від користувачів та відповідає на них відповідно до встановлених правил. Наприклад, для виконання функції відправлення розкладу занять бот може звертатися до бази даних, отримувати потрібну інформацію та формувати текстову відповідь для користувача.

Розробка відповідей на типові запитання

Для забезпечення швидкої та ефективної комунікації з користувачами бота важливо розробити відповіді на типові запитання. Це може включати в себе текстові відповіді на питання про розклад занять, контактну інформацію коледжу, правила та політику навчального закладу тощо.

Створення та наповнення вмістом різних команд та модулів

Крім основних функціональних можливостей, бот може мати різні команди та модулі для виконання конкретних завдань. Наприклад, команда для відправлення розкладу, команда для отримання інформації про викладачів, модуль для проведення опитувань тощо. Важливо створити ці команди та модулі і наповнити їх відповідним вмістом, щоб зробити бота максимально корисним та зручним для користувачів.

Цей етап роботи дозволить налагодити роботу бота та забезпечити його готовність до подальшого тестування та використання в реальних умовах.

3.1. Реалізація команд та відповідей на них

У цьому розділі ми розглянемо процес реалізації команд та відповідей на них для телеграм бота коледжу. Реалізація команд полягає у програмуванні бота таким чином, щоб він міг розпізнавати вхідні команди від користувачів і надавати відповіді або виконувати відповідні дії залежно від цих команд.

3.1.1. Визначення списку команд

Першим кроком у реалізації команд є визначення списку функцій, які має виконувати бот. Це може включати в себе такі команди, як [1]:

- /start - початок взаємодії з ботом;
- /schedule - відправлення розкладу занять;
- /news - отримання останніх новин або оголошень коледжу;
- /contacts - надання контактної інформації коледжу;
- /feedback - можливість залишити відгук або запитання адміністрації.

- /site - посилання на сайт коледжу

Ці команди, та багато інших реалізовані в форматі клавіатури (рис. 3.1, 3.2).

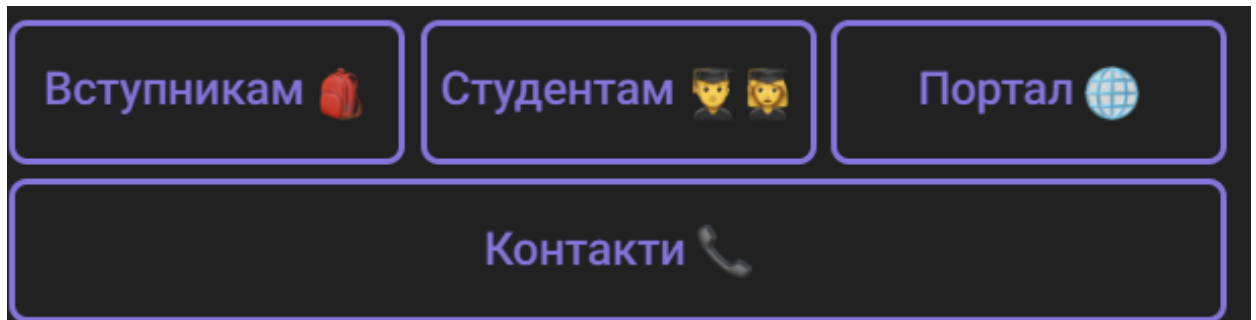


Рис. 3.1. Вигляд меню

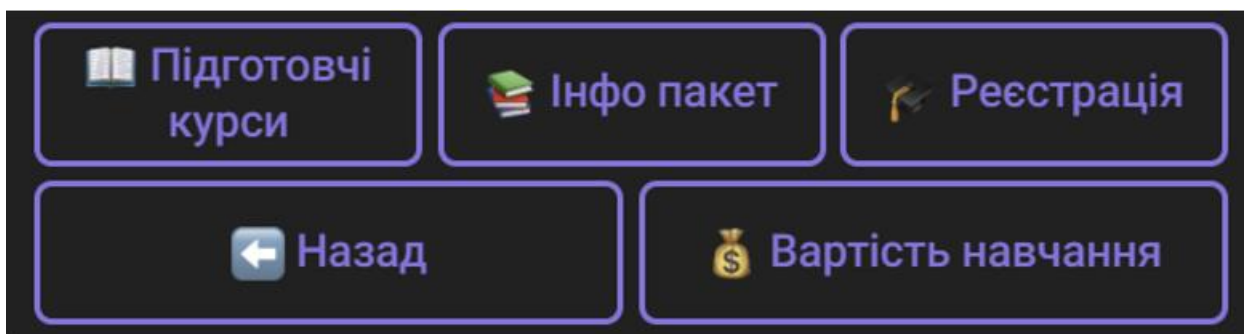


Рис. 3.2. Вигляд меню для Вступників

3.1.2. Програмування логіки відповідей

Після визначення списку команд потрібно реалізувати логіку відповідей на ці команди. Це означає написання коду, який буде обробляти отримані команди та надавати відповіді. Наприклад, як видно з рис. 3.3, для команди /schedule бот надсилає офіційний сайт де ви можете переглянути розклад пар.

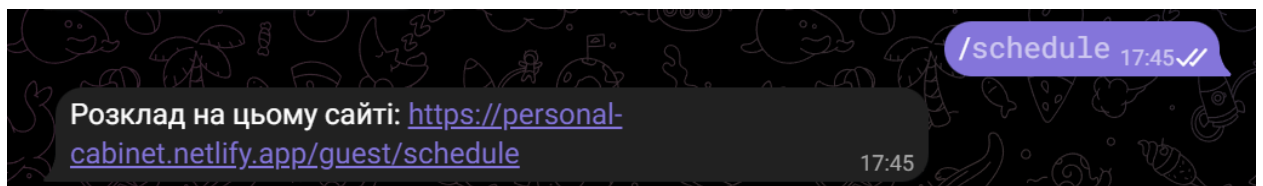


Рис. 3.3. Результат виконання команди schedule

3.1.3. Обробка варіантів запитань та сценаріїв

Під час програмування логіки відповідей важливо передбачити різні

варіанти запитань та сценаріїв взаємодії з користувачами. Наприклад, якщо користувач введе команду, яку бот не розпізнає, то потрібно надати відповідне повідомлення про помилку або запропонувати варіанти дій (див. рис. 3.4).

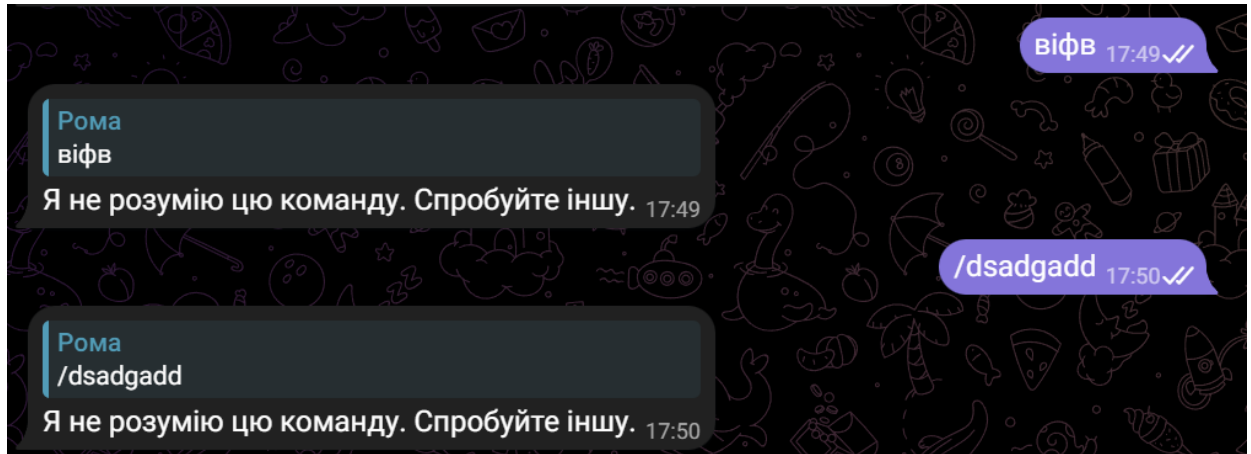


Рис. 3.4. Повідомлення про обробку невідомої команди

3.1.4. Тестування та вдосконалення

Після написання коду для реалізації команд важливо провести тестування, щоб переконатися, що бот коректно розпізнає команди та надає правильні відповіді. Якщо виявляються помилки або неочікувані сценарії, необхідно внести відповідні зміни в код для поліпшення роботи бота.

Цей етап роботи є ключовим для створення функціонального та користувацькі дружнього телеграм бота, який може задовольнити потреби користувачів коледжу.

3.2. Формування вмісту для різних категорій користувачів (абітурієнти, студенти)

У цьому розділі ми розглянемо процес формування вмісту для різних категорій користувачів телеграм бота коледжу, зокрема для абітурієнтів, студентів та викладачів.

3.2.1. Абітурієнти

Для абітурієнтів бот може надавати наступну інформацію (див. рис. 3.5)

[2]:

- Інформацію про вступну кампанію: строки подачі документів, умови вступу, перелік спеціальностей та кількість бюджетних місць.
- Розклад відкритих днів та екскурсій для ознайомлення з коледжем.
- Контактну інформацію приймальної комісії та можливість записатися на консультацію.

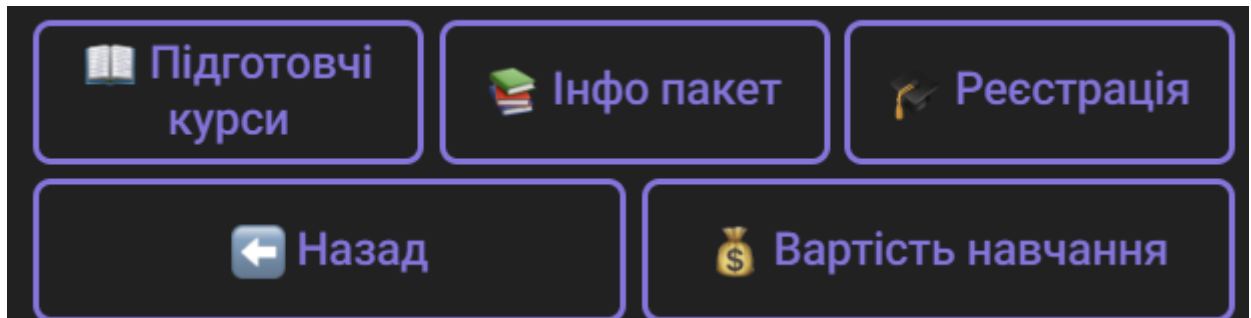


Рис. 3.5. Меню для абітурієнтів

3.2.2. Студенти

Для студентів бот може надавати наступну інформацію (рис. 3.6) [2]:

- Розклад занять та іспитів на поточний тиждень.
- Оголошення та новини коледжу: важливі події, конкурси, стипендії.
- Можливість перегляду власних оцінок та звітів.

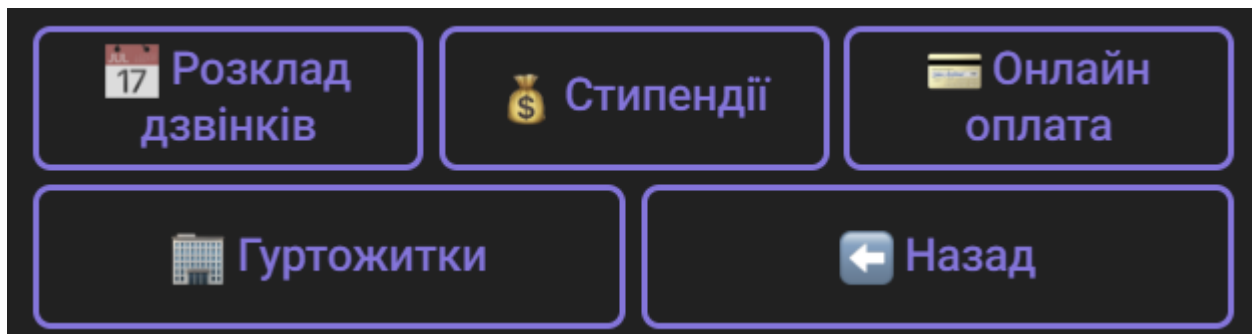


Рис. 3.6. Меню для студентів

3.2.4. Формат відправки інформації

Важливо врахувати, що інформація для різних категорій користувачів може відрізнятися за форматом та змістом. Наприклад, для абітурієнтів може бути корисними фотоальбоми та відеоролики з оглядом коледжу, в той час як

для студентів і викладачів більш актуальною буде текстова інформація про розклад та події.

3.2.5. Адаптація контенту

При формуванні вмісту для різних категорій користувачів важливо також врахувати їхні потреби та інтереси. Наприклад, для студентів можуть бути корисними поради з навчання, матеріали для самостійного вивчення та можливість обміну досвідом, в той час як для викладачів актуальними можуть бути матеріали з методики викладання та педагогічні ресурси.

Цей підхід дозволить забезпечити користувачів бота актуальною та цікавою інформацією, яка відповідає їхнім потребам та очікуванням.

Висновки з розділу 3:

Розділ 3 уточнює процес створення функціоналу та вмісту для телеграм бота коледжу. Він описує програмування бота для реакції на команди користувачів, включаючи визначення команд, програмування логіки відповідей, обробку варіантів запитань та тестування. Також розглядається формування вмісту для абітурієнтів, студентів та викладачів, включаючи визначення інформації та формату відправки.

4. ТЕСТУВАННЯ ТА ОТРИМАННЯ ЗВОРОТНОГО ЗВ'ЯЗКУ

У цьому розділі ми розглянемо процес тестування та отримання зворотного зв'язку щодо розробленого телеграм бота для коледжу.

4.1. Планування тестування

Для планування тестування телеграм бота для коледжу був розроблений детальний план, що включав наступні аспекти:

- Цілі тестування: Основною метою було перевірити функціональність бота, його відповідність вимогам та коректність реакції на різні команди.
- Ресурси тестування: Для проведення тестування були використані тестові середовища з реальними даними, а також тестові дані, що відображали реальні сценарії використання бота.

Цей план допоміг забезпечити систематичне та ефективне проведення тестів та забезпечити якість розробленого продукту.

Виконання тестів

Після розробки плану тестування розпочинається фаза виконання тестів, в якій проводяться різноманітні тести для перевірки функціональності, стабільності та безпеки бота. Це включало в себе тестування взаємодії з користувачем, перевірку правильності обробки введених даних, тестування реакції на помилкові вхідні дані тощо. Звичайно! Для тестування бота ми використовували наступні методи:

1. Функціональне тестування: Перевірка кожної функції бота згідно з його вимогами та очікуваннями користувачів. Ми переконалися, що кожна команда бота працює правильно та повертає очікуваний результат.

2. Тестування зручності використання: Я провів опитування, наскільки легко користувачі можуть взаємодіяти з ботом, включаючи зрозумілість команд, швидкість реакції та загальний інтерфейс.

3. Тестування на завантаження: Випробування бота на великі навантаження, щоб переконатися, що він залишається стійким та швидким під час активного використання.

Ці методи допомогли нам забезпечити високу якість бота та готовність до впровадження в роботу.

Збір зворотного зв'язку

У телеграм боті була розроблена та інтегрована функція збору зворотного зв'язку, яка дозволяє користувачам залишати свої відгуки та коментарі щодо роботи бота. Ця функція була реалізована через використання спеціальної команди (/feedback) у меню бота, яка спрямовувала користувачів на форму для залишення відгуку.

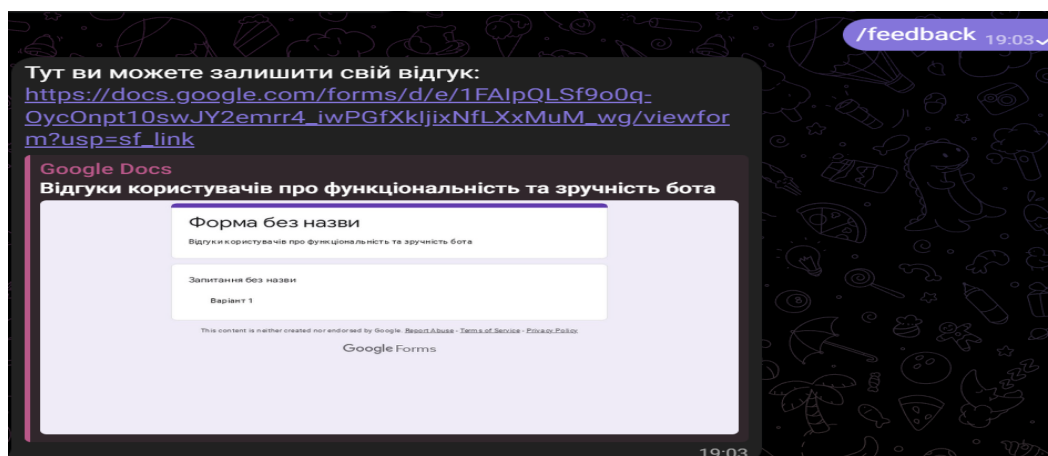


Рис. 4.1. Збір зворотного зв'язку

На формі зворотного зв'язку (див. додаток Б) користувачі могли надати свої враження щодо роботи бота, висловити пропозиції щодо його вдосконалення, а також залишити будь-які коментарі чи побажання. Ці відгуки автоматично збиралися та зберігалися в системі для подальшого аналізу та обробки адміністраторами бота.

Інтеграція функції збору зворотного зв'язку в телеграм бота дозволила залучити користувачів до активного взаємодії з ботом та надала можливість збирати цінні відгуки для поліпшення функціональності та якості обслуговування.

Аналіз результатів тестування

Зібрані дані про результати тестування та зворотний зв'язок були проаналізовані з метою ідентифікації потенційних проблем, помилок або недоліків у роботі бота.

Проведемо аналіз отриманих даних в процесі збору відгуків користувачів (див. додаток В). В опитуванні приймало участь 29 студентів. Більшій частині (48.3%) було дуже зручно користуватися ботом, 37.9% - зручно, решті незручно або нейтрально. Що стосується задоволеності користувачів якістю наданих послуг ботом – то 44.8% користувачів повністю задоволені, 44.8% частково задоволені та 10.3% не задоволені.

На основі цього аналізу були виправлені помилки, які містилися в боті.

Внесення змін та покращень

За результатами аналізу тестування та отриманого зворотного зв'язку проводиться внесення необхідних змін та покращень до розробленого телеграм бота. Це включало в себе виправлення помилок, додавання нового функціоналу, оптимізацію швидкодії та покращення користувацького досвіду.

Таким чином, процес тестування та отримання зворотного зв'язку є важливим етапом у розробці та вдосконаленні телеграм бота для коледжу, що дозволяє забезпечити його якість та відповідність потребам користувачів.

Висновки з розділу 4:

У розділі про тестування телеграм бота для коледжу було використано методи тестування, що допомогли перевірити функціональність та стабільність бота перед впровадженням. Планування тестування включало визначення цілей, методів, розподіл обов'язків та ресурсів. Завдяки інтегрованій функції збору зворотного зв'язку користувачі мали можливість залишати свої відгуки та пропозиції для подальшого вдосконалення бота. Тестування дозволило виявити та виправити можливі проблеми перед впровадженням бота в повний масштаб, забезпечуючи його більшу ефективність та надійність.

5. ПЕРСПЕКТИВИ ВПРОВАДЖЕННЯ ТА РОЗВИТКУ

Після успішного впровадження та реалізації телеграм бота для коледжу виникають перспективи подальшого розвитку та вдосконалення системи. Нижче наведено кілька перспективних напрямків, які можна розглядати для подальшого розвитку бота:

Розширення функціональності бота:

Однією з основних перспектив є постійне розширення функціональності бота, щоб він надавав більше корисних сервісів та можливостей для користувачів. Нові функції можуть включати в себе можливість перегляду розкладу занять, відправлення сповіщень про події в коледжі, надання консультацій з питань навчання тощо.

Інтеграція з іншими інформаційними системами:

Ще однією перспективою є інтеграція телеграм бота з іншими інформаційними системами, що використовуються в коледжі. Це може включати інтеграцію з системами управління навчальним процесом, бібліотечною системою, системою обліку успішності студентів тощо. Така інтеграція дозволить забезпечити ще більшу зручність та доступність інформації для користувачів.

Підтримка мультимовності:

Оскільки в коледжі можуть навчатися студенти з різних країн, важливо розглянути можливість підтримки мультимовності в телеграм боті. Це означає розробку бота з можливістю обслуговування користувачів різних мовних груп, надання інформації та відповідей у різних мовах.

Вдосконалення інтерфейсу та дизайну:

Постійне вдосконалення інтерфейсу та дизайну бота також є важливим напрямком розвитку. Зручний та привабливий інтерфейс сприятиме збільшенню зручності використання бота та підвищенню задоволеності користувачів.

Використання штучного інтелекту:

Використання штучного інтелекту для покращення функціональності та

реакції бота на запити користувачів може значно підвищити ефективність та рівень обслуговування. Використання алгоритмів машинного навчання дозволить боту ставати більш "розумним" та адаптивним до потреб користувачів.

Отже, реалізація цих перспективних напрямків дозволить постійно вдосконалювати та розширювати можливості телеграм бота для коледжу, забезпечуючи користувачам ще більш зручний та ефективний інструмент для отримання необхідної інформації та спілкування з навчальним закладом.

5.1. Можливості подальшого розвитку та удосконалення бота

Можливості подальшого розвитку та удосконалення бота для коледжу є безмежними і включають різноманітні аспекти, спрямовані на покращення функціональності, зручності використання та підвищення задоволеності користувачів. Деякі з потенційних напрямків розвитку включають:

- **Розширення функціональності:**

Однією з ключових можливостей є постійне розширення функціональних можливостей бота. Це включає в себе додавання нових сервісів та інструментів, таких як можливість перегляду розкладу занять, запитання про дати та події, відправлення нагадувань про важливі події та інше.

- **Впровадження інтерактивних функцій:**

Додавання інтерактивних елементів, таких як кнопки, меню та елементи вибору, може значно полегшити взаємодію з користувачами. Це дозволить швидше та ефективніше надавати необхідну інформацію та послуги.

- **Підтримка багатьох мов:**

У зв'язку з різноманітністю студентів, які можуть вивчати мови різних країн, важливо розглянути можливість підтримки багатьох мов в боті. Це дозволить користувачам обирати мову, яка їм зручна для спілкування з ботом.

- **Використання штучного інтелекту:**

Впровадження штучного інтелекту, такого як системи розпізнавання мови та аналізу настроїв, може підвищити рівень автоматизації та інтелектуальності бота. Це дозволить боту краще розуміти потреби користувачів та надавати більш персоналізовані та ефективні відповіді.

- Розширення інтеграцій:

Важливим напрямком розвитку є розширення можливостей інтеграції з іншими системами та сервісами. Це може включати в себе інтеграцію з системами управління навчальним процесом, бібліотечною системою, електронними журналами та іншими інформаційними ресурсами коледжу.

- Удосконалення аналітики та звітності:

Розвиток бота також включає покращення системи аналітики та звітності, що дозволить адміністраторам більш ефективно відслідковувати використання та задоволеність користувачів. Це допоможе зробити бота більш адаптивним та спрямованим на потреби користувачів.

Отже, подальший розвиток та удосконалення бота для коледжу має великий потенціал для покращення якості навчання та комунікації в навчальному закладі. Реалізація перерахованих можливостей може сприяти підвищенню ефективності та задоволеності всіх учасників навчального процесу.

5.2. Плани щодо впровадження бота в роботу коледжу

Плани щодо впровадження бота в роботу коледжу мають визначити стратегію впровадження, встановлення пріоритетів та кроки, необхідні для успішної інтеграції бота в навчальний процес. Деякі з ключових планів можуть включати:

- Розробка плану впровадження:

Перший крок - розробка докладного плану впровадження, який включатиме в себе етапи реалізації, визначення відповідальних осіб та встановлення часових рамок.

- Тестування та відладка:

Перед впровадженням бота в повний масштаб необхідно провести тестування та відладку. Це дозволить виявити та виправити будь-які можливі проблеми перед тим, як бот буде доступний для широкого користування.

- Підготовка персоналу:

Важливо забезпечити, щоб персонал коледжу був готовий до впровадження нового інструменту. Це може включати навчання адміністраторів та викладачів, а також створення навчальних матеріалів для користувачів.

- Підтримка користувачів:

Необхідно забезпечити механізми підтримки користувачів після впровадження бота. Це може включати створення системи зворотного зв'язку, де користувачі можуть подавати свої запити та отримувати допомогу з використанням бота.

- Моніторинг та оновлення:

Після впровадження необхідно проводити моніторинг використання бота та збирати відгуки користувачів для подальшого вдосконалення. Також важливо вчасно вносити оновлення та виправлення для забезпечення найвищої якості роботи.

- Маркетингові заходи:

Для успішного впровадження бота необхідно також провести маркетингові заходи серед студентів, викладачів та адміністрації коледжу, щоб підвищити обізнаність про новий інструмент та його переваги.

Отже, плани щодо впровадження бота в роботу коледжу повинні бути детально виконаними та орієнтованими на забезпечення успішного виконання цієї стратегічної ініціативи.

Висновки з розділу 5:

Після успішного впровадження та реалізації телеграм бота для коледжу виникають перспективи подальшого розвитку та вдосконалення системи. Основні напрямки розвитку включають розширення функціональності,

інтеграцію з іншими інформаційними системами, підтримку мультимовності, вдосконалення інтерфейсу та дизайну, а також використання штучного інтелекту. Ці можливості дозволять постійно вдосконалювати та розширювати можливості бота, забезпечуючи користувачам ще більш зручний та ефективний інструмент для отримання необхідної інформації та спілкування з навчальним закладом. Для успішного впровадження бота в роботу коледжу важливо розробити докладний план впровадження, провести тестування та відладку, підготувати персонал та забезпечити підтримку користувачів.

ВИСНОВКИ

У результаті виконання проекту був розроблений телеграм-бот для ВСП «Фахового коледжу ЧНУ ім. Юрія Федьковича», що має на меті полегшити доступ користувачів до інформації про навчальний заклад та його сервіси.

Проведений аналіз сучасних телеграм-ботів для освітніх закладів підтвердив потребу у створенні зручного та функціонального інструменту для спілкування зі студентами, абітурієнтами та викладачами. Бот надає інформацію про розклад занять, вартість навчання, актуальні оголошення.

Після аналізу різних платформ для розробки телеграм-ботів та їх можливостей, було обрано бібліотеку Telebot для Python як оптимальний інструмент для створення бота. Telebot забезпечує широкі можливості для реалізації функціональності бота та має зручний інтерфейс для програмістів.

Структура бота спроектована таким чином, щоб забезпечити зручну навігацію для користувачів. Розроблено окремі розділи для абітурієнтів, студентів та викладачів, а також включено можливість доступу до інформаційних ресурсів коледжу, таких як сайт та контактна інформація.

У результаті тестування функціональності бота було виявлено його високу ефективність та зручність використання для користувачів. Отримано позитивний зворотний зв'язок від користувачів, які висловили задоволення від можливостей бота та швидкості отримання інформації.

На основі вищевикладеного, можна зробити висновок, що розроблений телеграм-бот для ВСП «Фахового коледжу ЧНУ ім. Юрія Федьковича» є ефективним та корисним інструментом для спілкування зі студентами та іншими користувачами. Рекомендується використання бота в повсякденній роботі коледжу та подальший розвиток його функціоналу для забезпечення ще більшого комфорту користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Документація Telebot. URL: <https://pytba.readthedocs.io/en/latest/> (дата звернення: 01.11.2023)
2. Сайт ВСП «Фаховий коледж ЧНУ ім. Юрія Федьковича». URL: <http://college-chnu.cv.ua/> (дата звернення: 5.04.2024)
3. GitHub. Telegram Bot for College. URL: <https://github.com/chinmaywali/TelegramBotforCollege> (дата звернення: 08.01.2024)
4. Офіційна документація Python. URL: <https://docs.python.org/uk/3/> (дата звернення: 15.01.2024)
5. Офіційна документація Telegram API. URL: <https://core.telegram.org/bots/api> (дата звернення: 10.01.2024)
6. Joe Mayo. Programming the Microsoft Bot Framework. A Multiplatform Approach to Building Chatbots: Microsoft Press, 2017. 400p.
7. Р.А Мельник. Програмування веб-застосунків (фронт-енд та бек-енд). Львів: Львівська політехніка, 2018. 248 с.

ДОДАТКИ

Додаток А. Лістинг боту

```
import telebot
from telebot import types

# Створюємо бота, використовуючи токен
bot = telebot.TeleBot('6301468865:AAERyanAM5A9queOu749m7BX9J0emNO5-JI')

# Обробник команди /start
@bot.message_handler(commands=['start'])
def handle_start(message):
    # Отримуємо ім'я користувача
    user_name = message.from_user.first_name

    # Приємне привітання з ім'ям користувача
    welcome_message = f"🌟 Вітаю, {user_name}! Приємно бачити Вас у телеграм боті Коледжа ЧНУ! 📣\n\n" \
        "Тут Ви зможете отримати інформацію про наш Коледж, оголошення, події, " \
        "вартість навчання та багато іншого.\nМи прагнемо зробити Вашу участь у житті Коледжу " \
        "ще зручнішою та цікавішою.\n\nЯ тут, щоб допомогти Вам! 💡💬"

    # Відправляємо привітання
    bot.send_message(message.chat.id, welcome_message)
```

```

# Виводимо клавіатуру
    show_menu(message)

# Обробник команди /site
@bot.message_handler(commands=['site'])
def handle_site(message):
    # Відправляємо посилання на сайт
    site_url = "http://college-chnu.cv.ua/"
    bot.send_message(message.chat.id, f"Ось посилання на сайт:
{site_url}")

@bot.message_handler(commands=['schedule'])
def handle_site(message):
    # Відправляємо посилання на сайт
    site_url = "https://personal-
cabinet.netlify.app/guest/schedule"
    bot.send_message(message.chat.id, f"Розклад на цьому сайті:
{site_url}")

@bot.message_handler(commands=['contacts'])
def handle_contacts(message):
    contact_info = (
        "58002, м. Чернівці\n"
        "вул. Банкова, 1\n"
        "Навчальний корпус №20.\n"
        "Тел/факс: (0372) 55-38-26\n"
        "E-Mail: college@chnu.edu.ua"
    )
    bot.send_message(message.chat.id, contact_info)

```

```

@bot.message_handler(commands=['feedback'])
def handle_site(message):
    # Відправляємо посилання на сайт
    site_url = "https://docs.google.com/forms/d/e/1FAIpQLSf9o0q-
OycOnpt10swJY2emrr4_iwPGfXkIjixNfLXxMuM_wg/viewform?usp=sf_link"
    bot.send_message(message.chat.id, f"Тут ви можете залишити свій
відгук: {site_url}")

# Функція для виведення клавіатури
def show_menu(message):
    # Створюємо об'єкт клавіатури
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True)
    # Додаємо кнопки до клавіатури
    button_vstup = types.KeyboardButton("Вступникам 🎓")
    button_stud = types.KeyboardButton("Студентам 👨👩👧👦")
    button_portal = types.KeyboardButton("Портал 🌐")
    button_contacts = types.KeyboardButton("Контакти ☎")
    keyboard.add(button_vstup, button_stud, button_portal,
button_contacts)
    # Виводимо клавіатуру разом з повідомленням
    bot.send_message(message.chat.id, "Оберіть пункт меню:",
reply_markup=keyboard)

```

```

# Обробник для кнопки "Інформація"
@bot.message_handler(func=lambda message: message.text == "Вступникам
📖")
def handle_vstup(message):
    vstup_menu(message)
@bot.message_handler(func=lambda message: message.text == "Студентам
👤👤👤👤")
def handle_vstup(message):
    stud_menu(message)
@bot.message_handler(func=lambda message: message.text == "Портал 🌐")
def handle_portal(message):
    inline_keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton("Портал коледжу",
url="http://college-chnu.cv.ua/")
    inline_keyboard.add(url_button)
    bot.send_message(message.chat.id, "Натисніть кнопку нижче, щоб
перейти до Порталу Університету:", reply_markup=inline_keyboard)
# Обробник для кнопки "Контакти 📞"
@bot.message_handler(func=lambda message: message.text == "Контакти 📞")
def handle_contacts(message):
    contact_info = (
        "58002, м. Чернівці\n"
        "вул. Банкова, 1\n"
        "Навчальний корпус №20.\n"
        "Тел/факс: (0372) 55-38-26\n"
        "E-Mail: college@chnu.edu.ua"
    )

```

```

bot.send_message(message.chat.id, contact_info)

def vstup_menu(message):
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True)
    button_pidgot_kurs = types.KeyboardButton('📖 Підготовчі курси')
    button_infopacket = types.KeyboardButton('📁 Інфо пакет')
    button_registration = types.KeyboardButton('📝 Реєстрація')
    button_vartis = types.KeyboardButton('💰 Вартість навчання')
    button_nazad = types.KeyboardButton('⬅ Назад')
    keyboard.add(button_pidgot_kurs, button_infopacket,
button_registration, button_nazad, button_vartis)
    bot.send_message(message.chat.id, "Оберіть потрібний пункт:",
reply_markup=keyboard)

@bot.message_handler(func=lambda message: message.text == "📖
Підготовчі курси")
def handle_pidgot_kurs(message):
    inline_keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton("Переглянути підготовчі
курси", url="http://college-
chnu.cv.ua/article/5c5c7a988f13050d955f21c4")
    inline_keyboard.add(url_button)
    bot.send_message(message.chat.id, "Якщо Ви зацікавлені у
підготовчих курсах, можете переглянути інформацію на нашому веб-
сайті:", reply_markup=inline_keyboard)

```

```

@bot.message_handler(func=lambda message: message.text == "📁 Інфо пакет")
def handle_infopacket(message):
    inline_keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton("Переглянути інфо пакет",
url="http://college-chnu.cv.ua/pages/files/920b4ba6950f/College_2022_2.0.pdf")
    inline_keyboard.add(url_button)
    bot.send_message(message.chat.id, "Якщо вас цікавить інформаційний пакет, можете переглянути його, скориставшись посиланням:",
reply_markup=inline_keyboard)
@bot.message_handler(func=lambda message: message.text == "📝 Реєстрація")
def handle_registration(message):
    inline_keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton("Реєстрація",
url="https://vstup.edbo.gov.ua/")
    inline_keyboard.add(url_button)
    bot.send_message(message.chat.id, "Якщо вас цікавить реєстрація, можете виконати її, скориставшись посиланням:",
reply_markup=inline_keyboard)
@bot.message_handler(func=lambda message: message.text == "🎓 Вартість навчання")
def handle_vartis(message):
    inline_keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton("Вартість навчання",
url="http://college-chnu.cv.ua/article/62b4355c440ea40c926611ec")
    inline_keyboard.add(url_button)
    bot.send_message(message.chat.id, "Якщо вас цікавить вартість навчання, можете дізнатися її, скориставшись посиланням:",
reply_markup=inline_keyboard)

```

```

@bot.message_handler(func=lambda message: message.text == "← Назад")
def handle_nazad(message):
    show_menu(message)

def stud_menu(message):
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True)
    button_rozklad = types.KeyboardButton('📅 Розклад дзвінків')
    button_stupendii = types.KeyboardButton('💰 Стипендії')
    button_online_onpata = types.KeyboardButton('💳 Онлайн оплата')
    button_gurtojitok = types.KeyboardButton('🏠 Гуртожитки')
    button_nazad = types.KeyboardButton('← Назад')
    keyboard.add(button_rozklad, button_stupendii, button_online_onpata,
button_gurtojitok, button_nazad)
    bot.send_message(message.chat.id, "Оберіть потрібний пункт:",
reply_markup=keyboard)
@bot.message_handler(func=lambda message: message.text == "📅 Розклад
дзвінків")
def handle_rozklad(message):
    with open('Screenshot_4.png', 'rb') as photo:
        bot.send_photo(message.chat.id, photo, caption="Розклад
дзвінків")

@bot.message_handler(func=lambda message: message.text == "💰
Стипендії")
def handle_stupendii(message):
    inline_keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton("Стипендії",
url="http://college-chnu.cv.ua/article/scholarship")
    inline_keyboard.add(url_button)
    bot.send_message(message.chat.id, "Якщо вас цікавить чи ви
претендують на призначення академічної стипендії, можете дізнатися,
скориставшись посиланням:", reply_markup=inline_keyboard)

```

```

@bot.message_handler(func=lambda message: message.text == "🖨️ Онлайн  
оплата")
def handle_online_onpata(message):
    inline_keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton("Оплата онлайн",
url="https://next.privat24.ua/payments/form/%7B%22token%22:%2287e4d66b  
-ef80-4a0f-a0e6-32668f9f1ae4%22%7D")
    inline_keyboard.add(url_button)
    bot.send_message(message.chat.id, "Якщо ви б хотіли оплатити  
навчання онлайн, скористуйтеся посиланням:",
reply_markup=inline_keyboard)
@bot.message_handler(func=lambda message: message.text == "🏠  
Гуртожитки")
def handle_gurtojitok(message):
    inline_keyboard = types.InlineKeyboardMarkup()
    url_button = types.InlineKeyboardButton("Гуртожиток",
url="http://college-chnu.cv.ua/article/HOSTEL")
    inline_keyboard.add(url_button)
    bot.send_message(message.chat.id, "Вся інформація щодо гуртожитка  
знаходиться за посилання нижче:", reply_markup=inline_keyboard)

# Обробник для всіх повідомлень, які не є командами
@bot.message_handler(func=lambda message: True, content_types=['text'])
def handle_messages(message):
    bot.reply_to(message, "Я не розумію цю команду. Спробуйте іншу.")

# Запуск бота
bot.polling(none_stop=True)

```

Додаток Б. Тестування боту

Відгуки користувачів про функціональність та зручність бота

hzhshz795@gmail.com

Змінити обліковий запис

📧

Спільно не використовується

Як ви оцінюєте зручність використання бота

☐ Дуже зручно

☐ Зручно

☐ Нейтрально

☐ Незручно

Чи задоволені ви якістю наданих послуг ботом?

☐ Так, повністю

☐ Так, частково

☐ Ні, не задоволені

Чи хочете ви щось додати або виправити у нашому боті?

Ваша відповідь

Надіслати

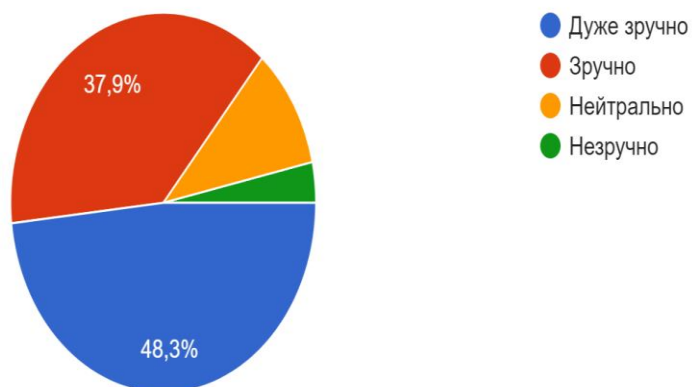
Очистити форму

					Розробка інтелектуального телеграм-боту			Лім.		Маса		Масштаб	
Змн.	Арк.	№ докум.	Підпис	Дата									
Розроб.		Страдецький Р.Ю.											
Перевір.		Мельничук А.Ю.											
Т. Контр.								Арк. 1			Аркушів		
Реценз.								Циклова комісія Комп'ютерної інженерії					
Н. Контр.													
Затверд.		Ташук О.Ю.											

Додаток В. Результати тестування бота

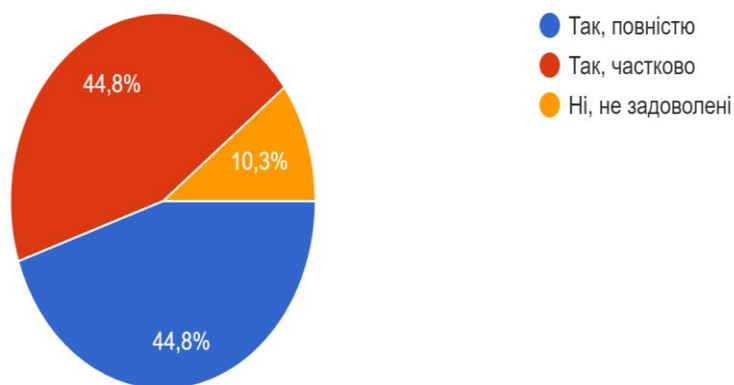
Як ви оцінюєте зручність використання бота

29 відповідей



Чи задоволені ви якістю наданих послуг ботом?

29 відповідей



Змн.	Арк.	№ докум.	Підпис	Дата	Розробка інтелектуального телеграм-боту	Лім.			Маса		Масштаб	
Розроб.		Страдецький Р.Ю.										
Перевір.		Мельничук А.Ю.										
Т. Контр.						Арк. 1			Аркушів			
Реценз.												
Н. Контр.						Циклова комісія Комп'ютерної інженерії						
Затверд.		Ташук О.Ю.										