

Галицький коледж імені В'ячеслава Чорновола
відділення комп'ютерних та видавничих технологій
циклова комісія інформатики та комп'ютерних дисциплін

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач відділенням

комп'ютерних та видавничих
технологій

Чубей О.О. / _____ /

підпис

«___» _____ 2020 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проєкту

освітньо-кваліфікаційного рівня «молодший спеціаліст»
зі спеціальності 122 «Комп'ютерні науки та інформаційні технології»

на тему: «Telegram бот для приймальної комісії Галицького коледжу
імені В'ячеслава Чорновола»

Студент групи К-47 Кузьменко К.О. _____
(підпис)

Керівник проєкту Чубей О.О. _____
(підпис)

Консультант:

з техніко-економічного Меленчук Л.І. _____
обґрунтування (підпис)

нормоконтролер Кульчинська Н.З. _____
(підпис)

Тернопіль - 2020

Галицький коледж імені В'ячеслава Чорновола
відділення комп'ютерних та видавничих технологій
циклова комісія інформатики та комп'ютерних дисциплін

ЗАТВЕРДЖУЮ
Завідувач відділенням
комп'ютерних та видавничих
технологій
Чубей О.О. /_____/
підпис

«__» _____ 2019 р.

ЗАВДАННЯ

на дипломне проєктування
на здобуття освітньо-кваліфікаційного рівня «молодший спеціаліст»
студенту Кузьменко Кирил Олександрович
(прізвище, ім'я та по-батькові студента)

1. Тема проєкту Telegram-bot для приймальної комісії Галицького коледжу імені
В'ячеслава Чорновола

затверджена наказом по коледжу від “20” листопада 2019 р., №_____

2. Термін здачі студентом завершеного проєкту “24” червня 2020 р.

3. Вихідні дані до проєкту _____

4. Перелік питань, які повинні бути розроблені в проєкті:

а) основна частина _____

б) техніко-економічне обґрунтування _____

5. Перелік графічного матеріалу _____

6. Консультанти проекту:

Розділ	Консультанти	Підпис, дата	
		Завдання видано	Завдання прийнято
з техніко-економічного обґрунтування	_____ (вчена ступень, звання П.І.Б. консультанта)		

КАЛЕНДАРНИЙ ПЛАН

дипломного проектування

№ п/п	Найменування етапу	Терміни	
		початку	завершення
1.	Вибір теми, ознайомлення з вимогами до дипломного проектування	04.11.19	18.11.19
2.	Огляд типових рішень та написання відповідного розділу ПЗ	20.11.20	27.01.20
3.	Дослідження технологій реалізації та написання відповідного розділу ПЗ	27.01.20	14.02.20
4.	Розробка функціональних вимог до проекту та робота над структурою програмного продукту. Написання відповідного розділу ПЗ	17.02.20	2.03.20
5.	Встановлення на налаштування середовища реалізації та написання відповідного розділу ПЗ	2.03.20	16.03.20
6.	Проектування програмного засобу (функціоналу, інтерфейсу, бази даних продукту) та написання відповідного розділу ПЗ	16.03.20	17.04.20
7.	Реалізація та налаштування програмного засобу та написання відповідного розділу ПЗ	17.04.20	4.05.20
8.	Доопрацювання модулів	5.05.20	18.05.20
9.	Тестування на налагодження програмного продукту та написання відповідного розділу ПЗ	18.05.20	19.06.20
10.	Опрацювання економічного розділу дипломного проекту та оформлення спеціального розділу	19.05.20	5.06.20
11.	Робота над оформленням пояснювальної записки	8.06.20	18.06.20
12.	Попередній захист дипломного проекту, доопрацювання	18.06.20	
13.	Підготовка до захисту дипломного проекту	23.06.20	25.06.20
14.	Захист дипломного проекту	26.06.20	26.06.20

7. Дата видачі завдання “_____” _____ 2019 р.

Керівник _____ / _____ /

Завдання прийняв до виконання _____ / _____ /

Реферат

Дипломний проєкт. Тема: «Telegram-bot». 60 сторінок, 41 рисунок, 2 таблиці. Галицький коледж імені В'ячеслава Чорновола.

Об'єкт дослідження – боти, автоматизація, штучний інтелект.

Предметом дослідження є месенджер телеграм, його оформлення, дизайн та функціонал і боти.

Метою дослідження є оцінка актуальності використання телеграм ботів в навчальних закладах.

Завданням проєкту є розробка бота, який зможе проконсультувати абітурієнтів.

Для розробки даного бота було використано мову Python та бібліотеку python-telegram-bot, тому що вже була використана ця бібліотека в інших проєктах. База даних PostgreSQL, з графічним інтерфейсом PgAdmin, в проєкті зв'язок з базою здійснювався за допомогою бібліотеки psycopg2.

Результат – розроблений бот, який може відповідати на запитання абітурієнтів, здійснювати розсилки та допомагати зв'язуватись з приймальною комісією користувачам.

МЕСЕНДЖЕР, ІНТЕРФЕЙС, КЛАВІАТУРА, БОТ, PYTHON.

Abstract

Diploma project. Subject: "Telegram-bot". 63 pages, 41 figures, 2 tables.

Galycky koledge name of Vyacheslav Chornovyl

Object of research - bots, automation, artificial intelligence.

The subject of the study is the telegram messenger, its design, design and functionality and bots.

The purpose of the study is to assess the relevance of the use of telegrams in educational institutions.

The task of the project is to develop a bot that will be able to advise applicants.

The Python language and the python-telegram-bot library were used to develop this bot, because this library has already been used in other projects. The PostgreSQL database, with the PgAdmin graphical interface, in the project communicated with the database using the psycopg2 library.

The result is a developed bot that can answer questions from applicants, make mailings and help users contact the admissions office.

ЗМІСТ

Вступ	7
1 Аналіз існуючих рішень та постановка завдання	8
1.1 Дослідження об'єкту інформатизації	8
1.2 Аналіз існуючих рішень	9
1.3 Постановка задачі	15
1.4 Обґрунтування доцільності створення бота	16
2 Проєктування бота	18
2.1 Проєктування структури бота	18
2.2 Проєктування інтерфейсу	20
2.3 Опис засобів реалізації	21
3 Реалізація та тестування	25
3.1 Реалізація основних модулів інформаційної системи	25
3.2 Типовий сценарій роботи з ботом	31
4 Техніко-економічне обґрунтування	47
4.1 Аналіз ринку	47
4.2 Розрахунок витрат на проєктування	47
4.3 Обґрунтування необхідності розробки	50
Висновки	51
Перелік джерел посилання	52
Додатки	53

					ДП.КН 20.408.03.000 ПЗ			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розробник		Кузьменко К.О.			Telegram-bot для приймальної комісії Галицького коледжу імені В'ячеслава Чорновола	Літ.	Арк.	Аркушів
Перевірено		Чубей О.О.					5	51
Реценз.		Кузик В.М.				ГК.ВКВТ. К - 47		
Н. Контр.		Кульчинська Н.З.						
Зав. відділ.		Чубей О.О.						

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

БД – база даних

СУБД – система управління базою даних

SQL – Structured Query Language

AES - Advanced Encryption Standard (симетричний алгоритм блочного шифрування)

ІТ – інформаційні технології

ПК – персональний комп'ютер

ПП – програмний продукт

ОС – операційна система

ІС – інформаційна система

ПЗ – програмне забезпечення

ЗП – заробітня плата

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		6

ВСТУП

На сьогодні галузь інформаційних технологій – це дев'яносто відсотків нашого життя, все що ми бачимо так чи інакше пов'язане з даною сферою. Більшість людей банально не розуміє, що вони є своєрідним носієм інформації, існують інформаційні гіганти такі як: Facebook, Twitter, Instagram, які кожен день збирають сотні мільйонів мегабайт даних про користувачів. Для запобігання несанкціонованого використання ваших особистих даних існує безліч методів їх захисту, але коли дані викрадає сама соціальна мережа захиститись не можливо, кожна фотографія яка була вивантажена в профіль містить інформацію про геолокацію, час здійснення фотографії і т. д. Тому використання соціальних мереж для спілкування супроводжується ризиком.

Сучасна молодь надає перевагу використанню месенджерів для спілкування між собою, на відміну від соціальних мереж вони надають кращий функціонал та захист, зокрема серед них виділяється популярний месенджер Telegram, він використовує 256-бітове симетричне AES-шифрування, 2048-бітове шифрування RSA і багато іншого, що забезпечує якісний захист даних.

Невід'ємною частиною любого месенджера є боти, кожен користувач може створити бота, для цього існують спеціальні платформи, якщо порівнювати з розробкою сайтів, ці платформи – це аналог використовуючи мови програмування такі як Python, php, Java і т. д.

Зважаючи на популярність месенджерів у дві тисячі двадцятому році, створення ботів це дуже перспективний напрямок, окрім складності, створення бота вимагає хороших знань алгоритмів, мови яка використовується для створення, також не менш важливим є вміння користуватись документацією та використовувати пошукові системи, але натомість виходить програмний продукт, який доволі захищений, може сам приймати рішення, дешевий в підтримці, простий при експорті і дуже корисний. Тому створення ботів і є трендом сьогодення.

					ДП.КН 20.408.03.000 ПЗ	Арк.
						7
Зм.	Арк.	№ докум	Підпис	Дата		

1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Дослідження об'єкту інформатизації

На сьогоднішній день використання месенджерів є актуальним, оскільки вони мають велику популярність, наприклад у WhatsUp зареєстровано понад 2 мільярди користувачів. Більшість інтернет магазинів пропонують відправляти дані про активність аккаунта через месенджери, Rozetka пропонує нам Viber, Allo – Telegram, Viber, Facebook Messenger.

Месенджер – служба для обміну повідомленнями між двома або декількома користувачами через інтернет. З початку, це були невеликі повідомлення і підтримувався лише текст. Але система розвивалась і зараз є можливість окрім тексту передавати файли, зображення, звукові сигнали та відео.

Відмінність месенджерів від, наприклад, електронної пошти в тому, що обмін повідомленнями здійснюється в реальному часі. При відправці повідомлення електронною поштою, воно зберігається в скриньці на сервері. Для того, щоб отримати повідомлення, користувач повинен сам перевірити свою скриньку і забрати повідомлення. У месенджерах відправлене повідомлення одразу передається отримувачу.

ІТ розвивається і з кожним днем з'являються нові програмні додатки, та тренди, деякі з них можуть вплинути на галузь досить серйозно, наприклад після появи соціальних мереж та месенджерів було поставлене досить серйозне питання захисту даних, та запобігання їх перехвату під час надсилання. Отож виникає запитання, чому месенджери на стільки популярні, адже можна спілкуватись через Skype, Instagram, це зумовлено тим що у соціальних мереж:

- складніший та довготриваліший процес реєстрації ніж у месенджерах;
- як правило додатки соціальних мереж потребують більше ресурсів пристрою, що відповідно спричиняє більший розхід заряду акумулятора;
- складний інтерфейс у порівнянні з месенджерами.

А також, одним з вирішальних факторів є розповсюдженість чат-ботів в месенджерах.

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		8

Що ж таке «чат-бот»? Це своєрідний додаток, який базується на платформі обміну повідомленнями, він надає користувачам взаємодіяти зі сторонніми сервісами, виконувати прості дії які потребують часу автоматично і швидко, а відігравати роль сайту, і все це через знайомий користувачу інтерфейс чату.

Тобто, для того, щоб отримати певну інформацію, людині не потрібно залишати межі месенджера, достатньо відправити йому команду, бот її обробить і дасть на неї відповідь, наприклад, можна швидко отримати інформацію про погоду, або дізнатися останні новини цікавого вам видання:

Особливу популярність боти отримали завдяки творцям месенджера Telegram, вони надали широкий спектр можливостей програмістам для розробки в цьому плані.

Однією з найприємніших особливостей ботів є те, що для взаємодії з ними не потрібно ніяких технічних знань або навичок програмування. Фактично, це нагадує листування з людиною, відмінність в тому, що бот розуміє тільки певний набір команд, або окремий набір тексту, все залежить від типу бота.

1.2 Аналіз існуючих рішень

Навчальні заклади рідко використовують телеграм-ботів і не зрозуміло чому, адже з їх допомогою можна отримувати інформацію про заклад, не звертаючись лишній раз до викладачів, особливо це буде корисно для початкових курсів.

Зважаючи на те, що кількість навчальних закладів, які використовують телеграм ботів для інформування студентів є невеликою, був знайдений тільки один подібний проєкт, де бот використовувався для допомоги вступникам «@vstup_knuba_bot», він простий у використанні, тому абітурієнт та його батьки можуть безперешкодно отримати всю потрібну для вступу інформацію.

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		9

Початок роботи з ботом зображений на рисунку 1.1

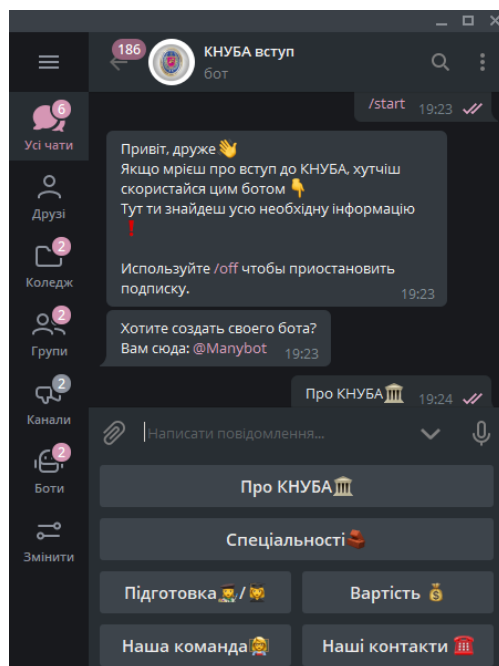


Рисунок 1.1 – Початок роботи з ботом «@vstup_knuba_bot»

Користувачу при початку роботи виводиться клавіатура бота, де розташовано 6 кнопок, 1 – «Про КНУБА», відправляє користувачу адрес, де розташована інформація про університет, цей підхід не є доцільним, оскільки об'єм інформації на сайті досить невеликий, і її можна відправити звичайним повідомленням, це буде швидше, та зручніше для користувача. Вміст сайту зображений на рисунку 1.2.

Київський національний університет будівництва і архітектури

February 27, 2020

Київський національний університет будівництва і архітектури (КНУБА) є провідним навчальним закладом в Україні з підготовки фахівців для будівельної галузі.

Заснований університет у 1930 році. За всі роки свого існування підготував понад **80 тисяч** інженерів і архітекторів, у тому числі близько **6 тисяч** спеціалістів для **120 країн світу**. В даний час в цнверситеті навчається близько **10000 студентів**.

На факультетах університету готують фахівців за **28 спеціальностями** і **45 освітніми програмами**.

Форми навчання: денна, заочна, програми подвійних дипломів.

Підготовка фахівців здійснюється за освітніми ступенями - бакалавр, магістр; освітньо-науковим - доктор філософії; науковим - доктор наук.

Рисунок 1.2 – Сайт з інформацією про КНУБА

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		10

Далі наведені плюси та мінуси даного бота:

Мінуси:

- використання не повноцінного домену telegra.ph/;
- відсутність реєстрації абітурієнтів.

Плюси:

- простота використання.

В зв'язку з відсутністю інформації на рахунок ботів навчальних закладів для прикладу було обрано бота, який завантажує відео з YouTube «@YTAudioBot» в комбінації з inline ботом, який шукає відео «@youtube» [1]. Пошук відео зображено на рисунку 1.3.

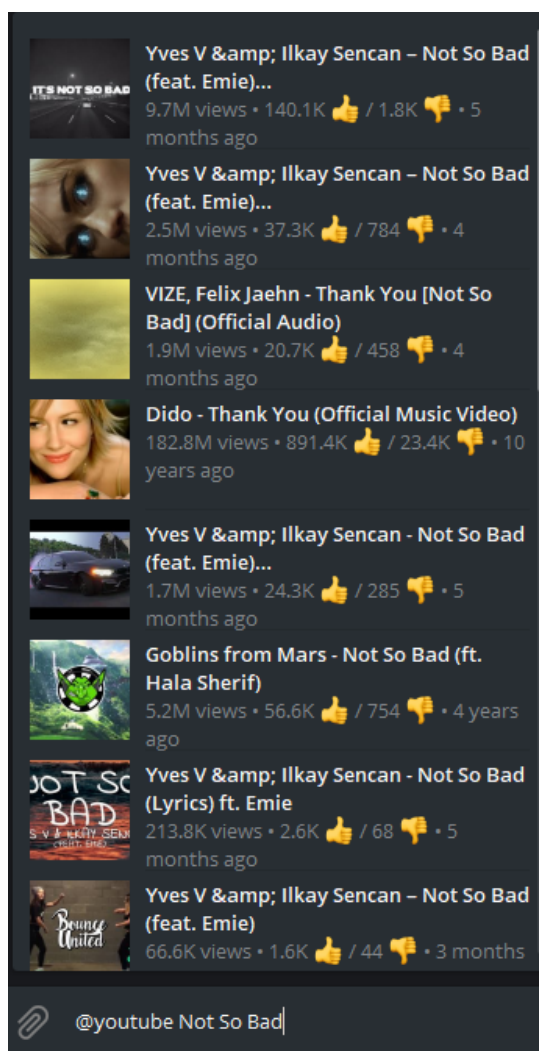


Рисунок 1.3 – Пошук музики з Youtube

Після того як ми знайшли потрібний нам відеозапис, він може бути відправлений в чат у якому було викликано inline бота, для прикладу було обрано чат бота, який конвертує відео в MP3. Відправлене inline-ботом відео містить у собі назву відео, у форматі посилання (на неї можна натиснути і воно відкриється браузером, або спеціалізованою програмою, яка обробляє посилання цього типу) бот якому було відправлено відео може обробляти тільки посилання і тільки ті, які адресовані на YouTube. Відправлене після пошуку відео зображене на рисунку 1.4.

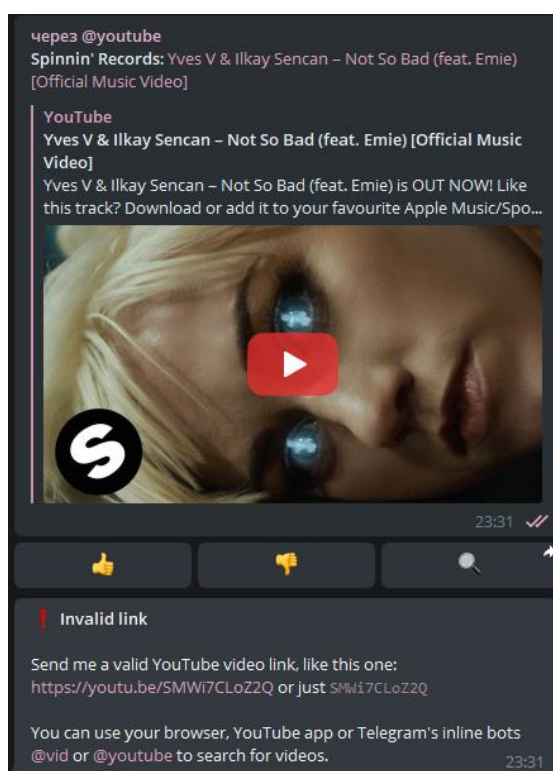


Рисунок 1.4 – Відправлений ботом відеозапис

Бот, якому було написано відповідь, що посилання не вірне, оскільки він не може прочитати посилання з тексту повідомлення, це власне і є мінус цього бота. Далі було скопійовано адресу з назви відео вручну, для цього на комп'ютерах потрібно натиснути правою клавішею мишки, в контекстному меню вибрати пункт «копіювати посилання», на мобільних пристроях роль правої кнопки миші відіграє затримка курсора на назві відео, після чого посилання було знову відправлено боту.

Відповідь бота зображено на рисунку 1.5.

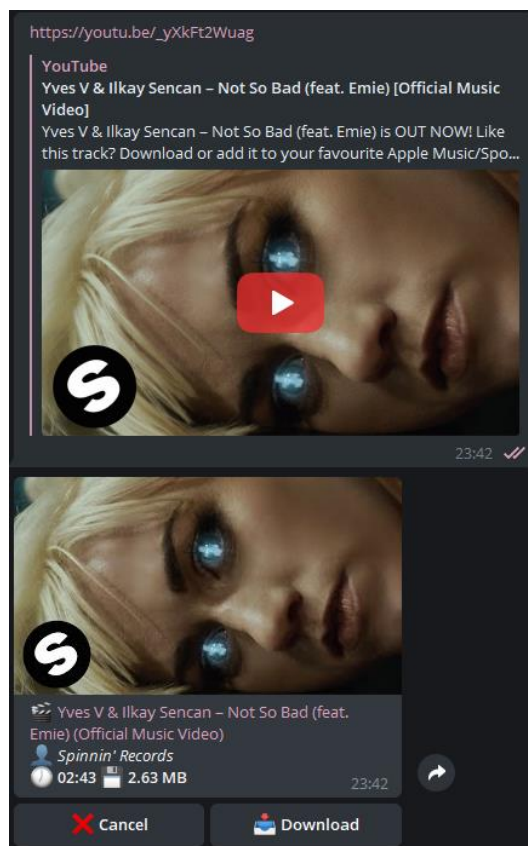


Рисунок 1.5 – Відправлене посилання та відповідь бота.

На рисунку 1.5 зображено відповідь бота, у ній вказано назву, тривалість, розмір та посилання на відео, також можливість видалити цей запис з чату, натиснувши на кнопку «Cancel», або завантажити в чат, натиснувши «Download». На рисунках 1.6, 1.7 зображено результат дії кнопок.

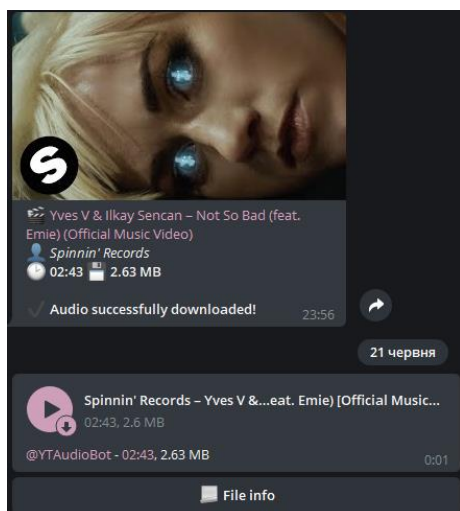


Рисунок 1.6 – Кнопка «Download»

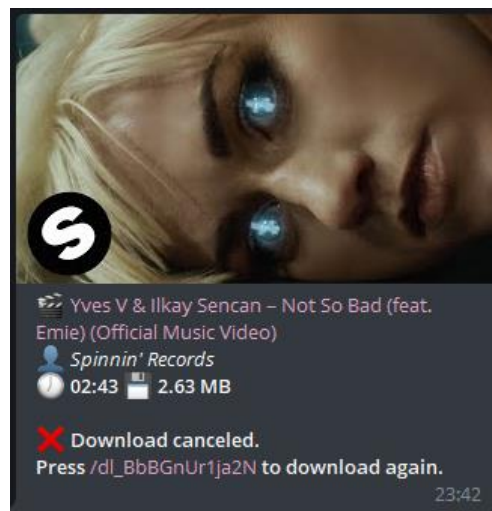


Рисунок 1.7 – Кнопка «Cancel»

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		13

Також даний бот (@YTAudioBot) надає простий інтерфейс у формі заміни клавіатури на кнопки. В інтерфейс включені чотири кнопки:

- «Donate» – надає посилання на рахунок;
- «History» – виводить кількість завантажених записів, а також кнопка «View», за допомогою якої можна переглянути записи;
- «Help» – виводить покрокову інструкцію з чотирьох пунктів, та кнопки щоб по них пересуватись;
- «Settings» – виводить меню налаштувань.

Інтерфейс зображений на рисунку 1.8

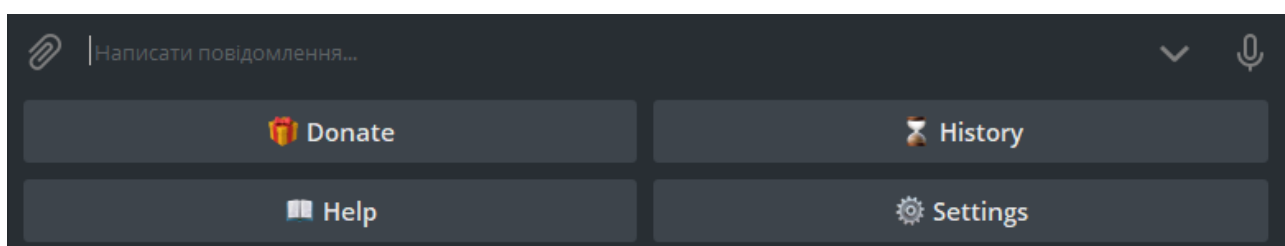


Рисунок 1.8 – Інтерфейс бота

Дане меню на відміну від попереднього виконано у формі inline клавіатури, вона дуже відрізняється від дії звичайної, звичайна клавіатура відправляє повідомлення боту, ніби його написав користувач, коли це inline клавіатура, вона прив'язується до конкретного повідомлення і це повідомлення не можливо підробити відправленим текстом, або відповіддю на це повідомлення, завдяки цьому можлива реєстрація, тому що вивід кнопок контролюється в залежності від дій користувача. На рисунку 1.9 зображено налаштування бота «@YTAudioBot», тут розташовано шість кнопок:

- «About» - виводить інформацію про бота;
- «FAQ» - виводить інформацію про загальні питання та відомі помилки;
- «Language» - виводить меню, де можна змінити мову інтерфейсу;
- «FeedBack» - надає можливість залишити відгук про користування ботом;
- «Donate» - дана кнопка вже була описана.

– «Translate» - надає посилання на проєкт, де можна допомогти розробникам бота у його перекладі, та інструкція для здійснення перекладу.

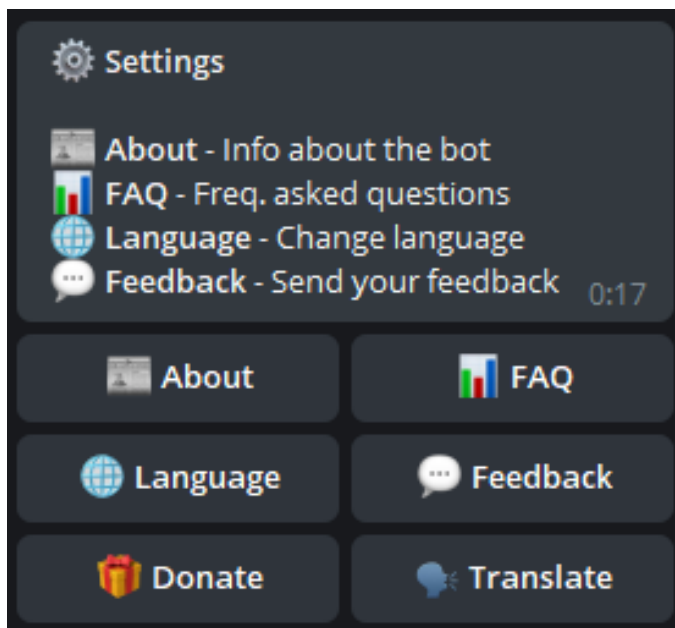


Рисунок 1.9 – Налаштування

1.3 Постановка задачі

Завданням дипломного проєкту – є створення телеграм-бота для примальної комісії Галицького коледжу імені В'ячеслава Чорновола.

Для створення був обраний месенджер «Telegram».

Месенджер Telegram був заснований Павлом Дуровим, який також є засновником мережі VK, обґрунтував він це тим, що коли дві тисячі одинадцятого року, до нього вірвався «спецназ» він усвідомив, що немає безпечного способу комунікації з братом [2].

Сам месенджер вперше з'явився на пристроях платформи iOS чотирнадцятого серпня дві тисячі тринадцятого року та двадцять другого серпня дві тисячі тринадцятого року на платформах Android. Telegram роздобув популярність завдяки високому рівню захисту даних, також він досить часто оновлюється, що робить його використання актуальним, з'являються все що є в тренді – є в Telegram.

Згідно статистики сайту statista.com починаючи з дві тисячі чотирнадцятого до дві тисячі вісімнадцятого року, кількість активних користувачів збільшилась з

					ДП.КН 20.408.03.000 ПЗ	Арк.
						15
Зм.	Арк.	№ докум	Підпис	Дата		

тридцяти п'яти мільйонів до двох сотень мільйонів, а за останні два роки вона ще збільшилась більше ніж в два рази, популярність месенджера росте в геометричній прогресії і це не можливо не замітити.

Дивлячись на статистику було важко не скористатись Telegram для написання бота, адже боти – це одна з сильних сторін даного месенджера. Серед інших додатків для обміну повідомленнями розробники Telegram одні з перших, хто виділив значну частину ресурсів для розвитку цієї сфери, тому процес розробки бота на даній платформі був максимально комфортний і зрозумілий.

1.4 Обґрунтування доцільності створення бота

На сьогодні існує безліч ботів для різних задач, одні використовуються для розвитку штучного інтелекту, інші використовуються для автоматизації певних функцій підприємства, боти – це стандарт дві тисячі двадцятого року і в майбутньому, вони використовуватимуться для автоматизації дій людини все частіше, прогнозується навіть написання елементарних програм ботами.

Часто боти використовуються для заміни вебсторінки, звісно це не є хорошим рішенням, боти актуальні для виконання певних кроків, які користувачу потрібно пройти на сайті, чи додатку, щоб отримати потрібний йому результат, боти повинні бути прості, але функціональні щоб допомагати користувачу використовувати час максимально продуктивно, також нижче наведено список переваг та недоліків бота в порівнянні з сайтом:

- бот простіший у використанні;
- не потрібно розбиратись з мапою сайту, привикати до його інтерфейсу, бот не вимагає залишати привичний користувачу месенджер і виводить інформацію у вигляді повідомлень;
- використання бота потребує менше ресурсів пристрою;
- бот здатний спілкуватись з людиною.

Великим плюсом у використанні бота є те, що йому не потрібен сервер, він використовує лише інтерпритатор Python, достатньо лиш запустити файл з ботом, що робить процес розгортання надзвичайно простим.

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		16

В Галицького коледжу є досить хороший сайт, з приємним та зручним інтерфейсом, але бракує доповнення в плані push – сповіщень, що супроводжується підключенням firebase, та значними затратами ресурсів, в порівнянні з ботом, де це вже реалізовано самим додатком Telegram.

У зв'язку з тим, що використані ресурси на розширення функціональності сайту з очікуваним результатом програє у пропорційному відношенні до кількості ресурсів та очікуваного результату розробки бота, було вирішено створити бота, який розширить функціонал сайту, а саме: надасть можливість організовувати розсилки повідомлень абітурієнтам, студентам, а також консультувати абітурієнтів, адже бот здатний швидше і конкретніше відповідати на прості питання, тим самим він економить час приймальній комісії.

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		17

2 ПРОЄКТУВАННЯ БОТА

2.1 Проєктування структури бота

Месенджери зараз є стандартом, їх використовують всі люди які мають доступ до інтернету, багато з них на стільки звикли до месенджерів, що застосовують їх замість веббраузера, в цьому їм допомагають боти, звісно функціонал слабший ніж повноцінне відвідування сайту, але для отримання конкретної інформації максимально швидко, це хороше рішення [3].

Існує багато різних типів ботів, наприклад:

- Welcome-бот – це свого роду візитна картка компанії, після початку спілкування з ботом, він відправляє характеристику компанії та описує свої основні функції.

- Боти-оператори – це боти, які певною мірою виконують роботу оператора. Вони допомагають користувачам вирішувати їхні проблеми, не залучаючи оператора, але якщо порад бота не достатньо, користувач завжди може попросити його викликати оператора.

- Пошукові боти – це боти, які дають відповідь користувачу, згідно відправленого користувачем повідомлення, також ці боти можуть реагувати на певний текст, наприклад «телефон», «рахунок», і виводить дані із бази компанії, для якої він створений.

- Inline-боти – це боти, які можна використовувати під час спілкування з людиною, або з групою людей (в чатах). Щоб написати такому боту в телеграм потрібно поставити символ «@», вказати псевдонім бота і текст чи команду, бот напише відповідь в чат, чи в переписку, з якої йому був адресований запит.

Для виконання дипломного проєкту був створений індивідуальний тип бота, він певною мірою виконати функції Welcome-бота, та бота-оператора, але він відрізняється від них можливістю здійснення розсилки вчителем повідомлень для студентів та абітурієнтів, це перекриває відсутність такої можливості для реалізації через веб сторінку коледжу.

Основна мета створення бота – це модернізація можливостей сайту, розширення методів можливого комунікування коледжу із студентами на надзвичайно популярний месенджер Telegram, додаткове інформування студентів та абітурієнтів.

Етапи створення бота:

- проєктування бота (постановка, аналіз та фіксація цілей);
- проєктування бази даних (створення бази даних користувачів);
- розробка типів профілю;
- розробка авторизації;
- розробка статусів зчитування;
- розробка розсилки повідомлень;
- розробка алгоритму виправлення помилок;
- проєктування загальних питань фбітурієнтів;
- тестування;
- запуск бота.

Отже потрібно створити систему для інформування абітурієнтів та збереження інформації про користувачів, можливість зв'язатись викладачу з конкретною групою, або абітурієнтами без пошуку кожного користувача в мережі Telegram.

Реалізація проєкту допоможе представникам приймальної комісії суттєво зекономити час, адже бот буде відповідати на більшість запитань абітурієнтів і тільки в крайньому випадку (коли бот не матиме відповіді на питання абітурієнта і відповідно не зможе його проконсультувати) повідомлення пуде відправлене в приймальну комісію, це зекономить надзвичайно багато часу, також реалізована масова розсилка повідомлень всім абітурієнтам, завдяки цьому не потрібно зв'язуватись з кожним студентом, достатньо буде лиш написати повідомлення боту, який в свою чергу розішле його всім потрібним користувачам.

Основні вимоги до функціоналу подані на діаграмі IDEF0 (рис 2.1).

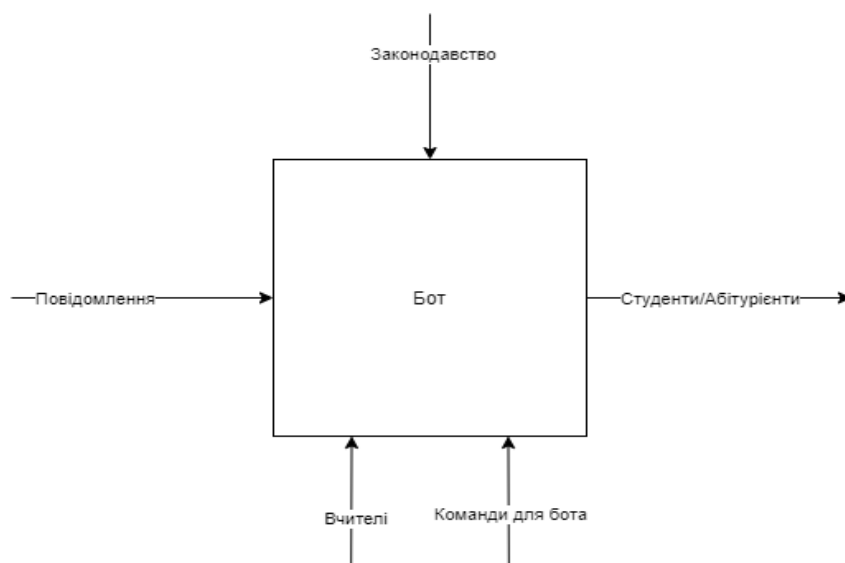


Рисунок 2.1 – Загальна схема діяльності бота

Робота внутрішньої інформаційної системи бота зображено на рисунку 2.2.



Рисунок 2.2 – Внутрішня структура інформаційної системи

2.2 Проєктування інтерфейсу

Інтерфейс повинен бути простим та зручний для користувачів, щоб користувач міг швидко та легко отримати потрібну йому інформацію.

В ботах використовуються кнопки управління, кожна кнопка відправляє боту команду, яку бот аналізує та дає відповідь. На відміну від сайтів де при

натисканні на кнопку ви можете перейти в інший розділ, міняються лише кнопки з командами для маніпулювання, а інформація виводиться в чат з ботом.

З сторони дизайну боти виділяються оформленням повідомлення та кнопок, але боти в загальному використовуються для швидкого та зручного отримання потрібної інформації, хороший бот – це бот де максимальна кількість інформації знаходиться в максимально простому інтерфейсі.

2.3 Опис засобів реалізації

Для реалізації бота був обраний месенджер Telegram у зв'язку з хорошим рівнем захисту даних. У створенні ботів допомагає велика кількість API, для кожної мови програмування використовуються свої, була обрана мова Python, у зв'язку з простотою, зручністю, та хорошими знаннями цієї мови, для Python існує декілька прикладних програмних інтерфейсів Telegram:

- python-telegram-bot;
- pyTelegramBotAPI;
- AIOGram;
- twx.botapi;
- telebot;
- Telepot;
- Telegram Bot Service;
- Pyrogram.

Для зручної розробки бота було використано бібліотеку python-telegram-bot, та програмне забезпечення:

- веббраузер «Chrome»;
- середовище програмування PyCharm;
- месенджер «Telegram» (для тестування).

python-telegram-bot – прикладний програмний інтерфейс для розробки Telegram-ботів на python [4],

PyCharm – середовище розробки створене спеціально для мови Python, воно вимагає більше ресурсів комп'ютера ніж текстові редактори такі як «Visual Studio

Code», «Sublime Text», але натомість надає більший функціонал. У мові програмування запроваджений стандарт «PEP8», згідно нього програміст повинен в першу чергу задумуватись над простотою коду, а потім над його чистотою, також в PEP прописані відступи, наприклад між об'єктами відступ 2 рядка і т. д. В PyCharm впроваджений модуль для рефакторингу коду, що оптимізує код згідно стандартів, це допомагає зробити код красивим та зрозумілим для інших всього за одну мить, важко уявити скільки часу процес рефакторингу займає у звичайному текстовому редакторі, особливо при великих об'ємах коду.

PostgreSQL – це система управління базами даних, яка на відміну від MySQL, Apache, FreeBSD не контролюється якоюсь однією компанією, її розробкою займається декілька компаній та людей, які співпрацюють, щоб розвивати і використовувати Postgres у власних цілях.

PostgreSQL оптимізована для роботи з Python і має власний інтерпретатор, що дозволяє маніпулювати даними не вдаючись до використання SQL [5].

PgAdmin – це інструмент управління базою даних, розроблений для бази PostgreSQL та похідних реляційних баз даних, таких як EDD Advanced Server, EnterpriseDB. Він може запускатися як вебсайт або як настільний додаток [6]. Головна сторінка PgAdmin зображена на рисунку 2.1.

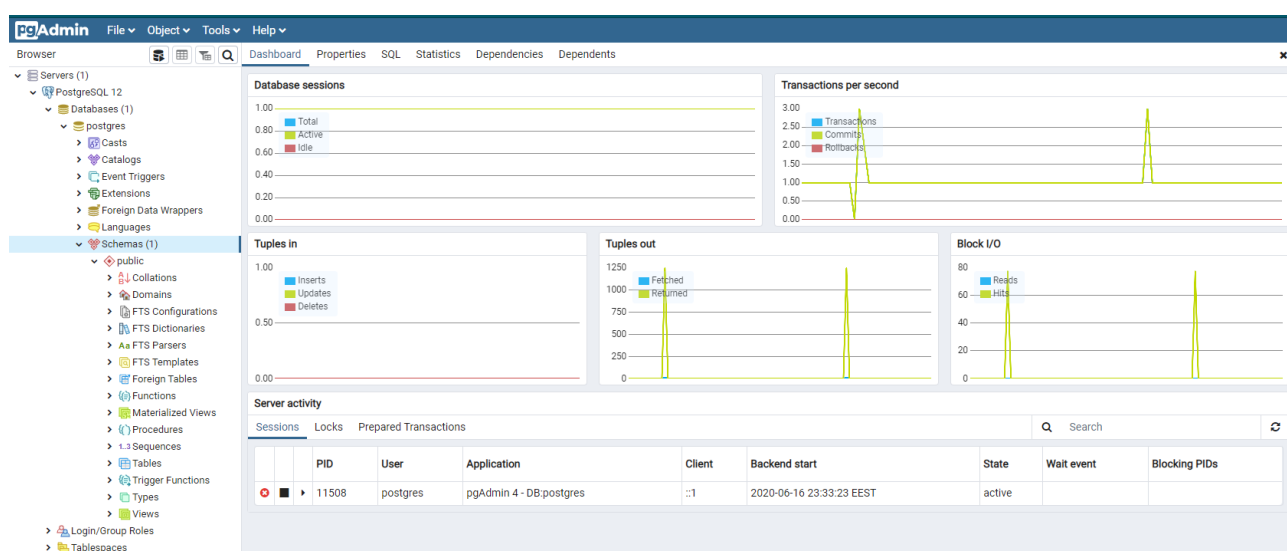


Рисунок 2.1 – Вебінтерфейс «PgAdmin»

Python – проста у використанні, мова програмування. Будучи мовою високого рівня, вона має вбудовані типи даних високого рівня, списки і дерева, ефективна реалізація яких на мовах низького рівня: С, С++, потребує значних витрат часу .

Розробку Python розпочав Гвідо ван Россумом, наприкінці 1980-х років, на той момент він був співробітником інституту в Голландії. Python розроблявся як скриптова мова з можливістю розширення, для операційної системи під назвою «Амоеба». Попри розробку Python, Гвідо також брав участь в розробці мови ABC (це мова, орієнтована на навчання програмування), тому деякі функції Python були запозичені з ABC.

Назва мови «Python» що на українському означає «Пітон», виникла далеко не з ряду лускатих. Автор вирішив назвати мову «Python» на честь комедійного серіалу 70-х років «Повітряний цирк Монті Пайтона». Втім, все одно назву мови частіше асоціюють саме з пітоном, ніж з серіалом, адже піктограми файлів і навіть емблема на сайті python.org зображують зміїну голову.

Перша версія Python вийшла 3 грудня 2008 року, під назвою «Python 3000» (або Python 3.0). Після тривалого тестування було усунено недоліки архітектури з максимально можливим збереженням сумісності з попередніми версіями мови. На сьогоднішній день актуальною версією Python є 3.9.

Python дозволяє розділяти програму на модулі, що можуть використовуватись в інших програмах, також він встановлюється з великою бібліотекою стандартних модулів, їх можна використовувати для полегшення роботи, або як приклади при вивченні мови. Стандартні модулі надають засоби для роботи з:

- файлами;
- системними викликами;
- мережними з'єднаннями;
- інтерфейсами до різних графічних бібліотек.

Python [7] – інтерпретована мова, що заощаджує масу часу, що була б витрачена на компіляцію. Інтерпретатор можна використовувати інтерактивно, що дозволяє дізнаватись про нові можливості мови, створювати шаблони програм, або відлагоджувати функції при розробці «знизу-вверх». Він також зручний як калькулятор. Python дозволяє писати дуже компактний, простий та зрозумілий код. Програми, написані мовою Python, як правило значно коротші подібних на C або C++ з декількох причин:

- типи даних високого рівня дозволять не зволікати ні типізацію методів, вхідних і вихідних типів даних;
- немає необхідності в оголошенні змінних

Python розширювана мова: оскільки виконання операцій через інтерпритатор є повільним відносно C, в python можна вставляти модулі, написані на C, щоб пришвидшити виконання ресурсномістких операцій, або написати інтерфейс для комерційних бібліотек, доступних тільки у двійковій формі. Інтерпретатор мови Python може бути вбудований у програму, написану на C, і використовуватись як розширення, або командну мову для цієї програми. Python використовується в даний час десятками тисяч програмістів в усьому світі і його популярність тільки росте [8].

Переваги Python:

- зручний синтаксис (блоки коду розподіляються відступами);
- безпроблемне транспортування коду, яке не залежить від операційної системи завдяки інтерпретатору;
- відсутня потреба у встановленні додаткових модулів, python включає багато стандартних модулів, функціонал яких є досить потужний, включено навіть модуль для роботи з графікою;
- можливість використання Python в консольному режимі (дуже корисне для тестування окремих частин програми та розв'язання простих завдань, можна використовувати як потужний калькулятор).

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

3.1 Реалізація основних модулів інформаційної системи

Під час розробки було виділено декілька основних завдань, які в подальшому було розподілено на модулі:

- створення бота;
- налаштування бота;
- розробка реєстрації в боті;
- розробка повідомлень бота;
- розробка різновидів облікових записів;
- передбачення зміни облікових даних;
- обробка відповіді користувача;

Створення бота в Telegram це доволі нескладний процес, який здійснюється за допомогою «батька ботів» (@BotFather), це бот створений телеграмом, у нього досить простий та зрозумілий інтерфейс, управління ботом здійснюється певним набором команд, вивести список команд можна просто ввівши боту любий текст, також цей список виводиться при початку спілкування з ботом. Короткий опис основних команд:

- «/newbot» - створення нового бота;
- «/mybots» - створені боти;
- «/setname» - зміна найменування бота;
- «/setdescription» - зміна опису бота;
- «/setabouttext» - зміна інформації про бота;
- «/setuserpic» - зміна фото профілю бота;
- «/setcommands» - зміна списку команд;
- «/deletebot» - видалення бота;
- «/token» - генерація нового токена для бота;
- «/setinline» - включити inline mode;
- «/setprivacy» - включити приватний статус бота.

Для створення бота достатньо ввести команду «/newbot», після чого придумати назву бота та псевдонім, який повинен закінчуватись надписом «bot». Алгоритм створення бота зображений на рисунку 3.1.

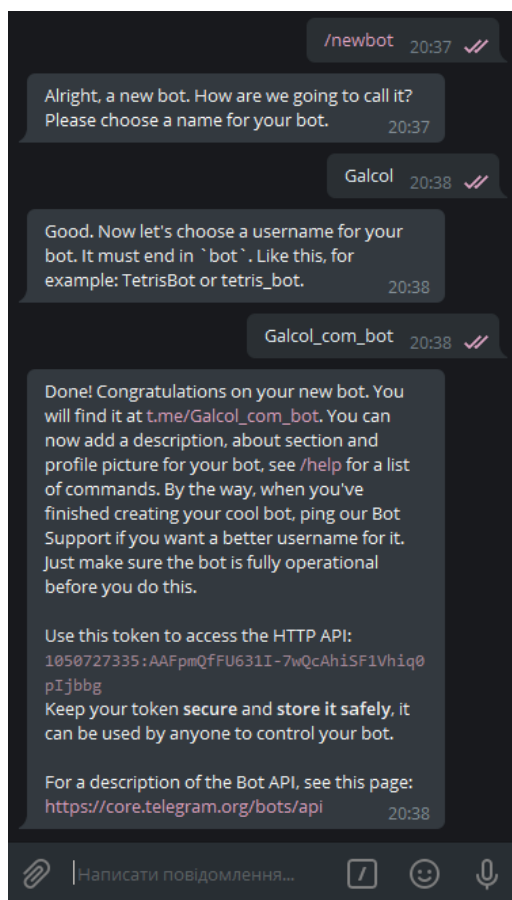


Рисунок 3.1 – Алгоритм створення бота

Налаштування бота є основною складовою розробки любого бота. В Telegram налаштування здійснюється через вище вказаного бота «@BotFather», потрібно відправити команду «/mybots», після чого вибрати потрібного бота, в inline клавіатурі, яка виведеться під повідомленням, після чого відкриється меню бота. В меню потрібно вибрати кнопку «Edit Bot», де ми можемо вказати короткий опис бота, додати команди, для цього потрібно відправити боту назву команди і через дефіс короткий опис бота, також можна змінити назву бота («Edit Name»), зображення профілю бота («Edit Botpic»), та інформацію про нього, яка виводитиметься під час пошуку бота («Edit About»), меню «Edit Bot» зображено на рисунку 3.2, також потрібно відкрити налаштування кнопкою «Bot Settings», де в нас є можливість встановити домен («Domain»), додати систему оплати

(«Payments»), додати включити, або відключити приватність в групах («Group Privacy»), якщо це включено використовувати бота в чатах зможуть тільки адміністратори, дозволити, або заборонити використання бота в групових чатах («Allow Groups?»), та налаштування inline стану («Inline Mode», «Inline Location Data», «Inline Feedback») меню «Bot Settings» зображено на рисунку 3.3.

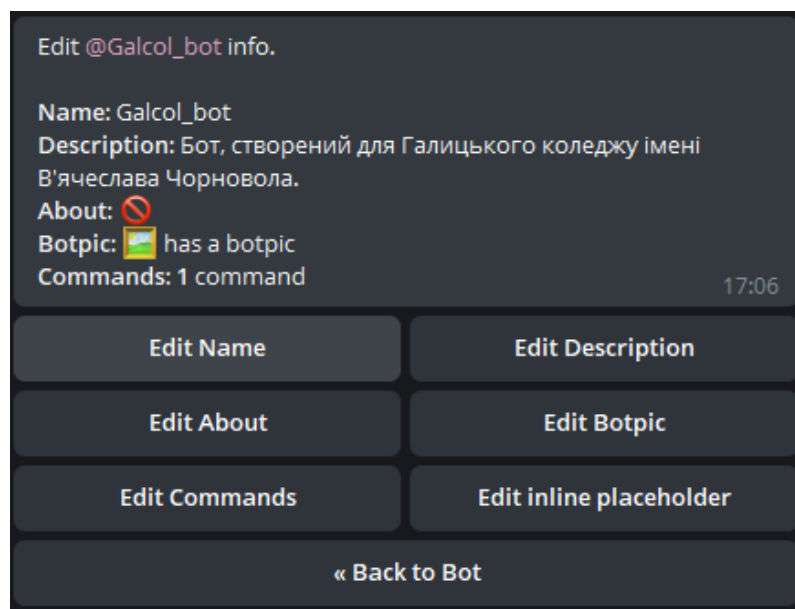


Рисунок 3.2 – Меню «Edit Bot»

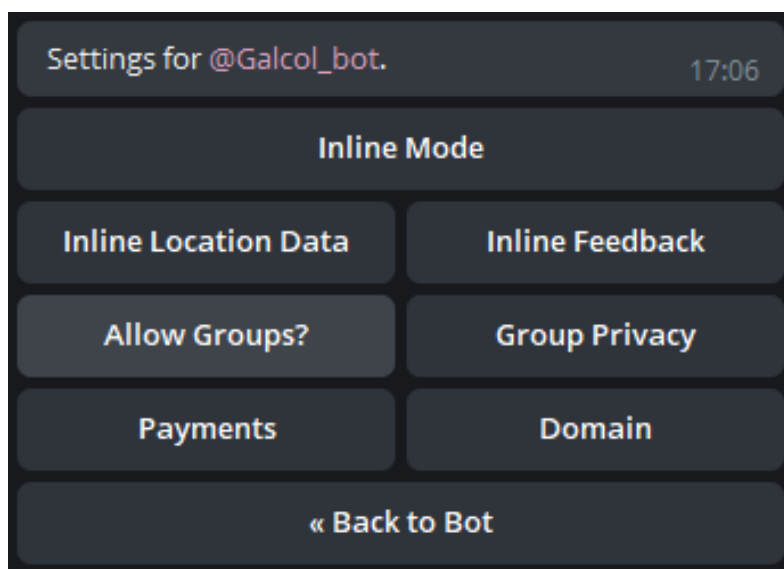


Рисунок 3.3 – Меню «Bot Settings»

Реєстрація – це збереження інформації про користувача, для подальшої її використання в цілях персоналізації контенту, ідентифікації та виводу.

В боті збереження інформації здійснюється після вибору користувачем типу профілю. Алгоритм реєстрації зображено на лістингу 3.1

Лістинг 3.1 – алгоритм реєстрації.

```
def register(user_id, group, fullname=None, link=None):
    conn = psycopg2.connect(config.DATABASE)
    cursor = conn.cursor()
    try:
        cursor.execute("INSERT INTO public.students(id, fullname,
link, stud_group) VALUES (%s, %s, %s, %s)",
                        (user_id, fullname, link, group))
    except UniqueViolation:
        cursor.close()
        conn.close()
        conn = psycopg2.connect(config.DATABASE)
        cursor = conn.cursor()
        cursor.execute("UPDATE public.students SET stud_group=%s,
link=%s, teacher=FALSE, entrant=FALSE WHERE id=%s",
                        (group, link, user_id))
        cursor.close()
        conn.commit()
```

Розробка повідомлень бота - це відповіді бота на активність користувача, бот повинен відповідати зрозуміло для користувача, тому дії описані максимально зрозуміло щоб бот міг використовуватись абсолютно кожним. Алгоритм обробки вибору користувачем статусу абітурієнта зображено на лістингу 3.2.

Лістинг 3.2 – Відповідь бота на вибір статусу абітурієнта

```
if data == ENTRANT:
    utils.create_entrant(user_id, fullname)
    query.edit_message_text(
        text='Привіт\n'
            'Які в тебе є запитання, %s???\n'
            'Я готовий відповісти 😊' % first_name,
        reply_markup=back_to_status())
```

При роботі з ботом реалізація декількох облікових записів є обов'язковою умовою, якщо є різні типи доступів і можливості користувачів, в боті існує всього три типа записів:

- абітурієнт;
- студент;
- представник приймальної комісії.

Абітурієнт – це базовий користувач, в нього є можливість вести переписку з ботом, запитувати в нього те, що його цікавить, якщо це стосується вступу, якщо бот не зрозуміє запитання він попередить про це користувача.

Студент – це профіль з можливістю перегляду інформації про свою групу.

Представник приймальної комісії – це обліковий запис з можливістю відправки повідомлень кожному з типів облікових записів.

Для вибору типу профілю реалізовано клавіатуру, яка зображена на лістингу 3.3.

Лістинг 3.3 – клавіатура вибору профілю.

```
def start_keyboard():
    keyboard = (
        InlineKeyboardButton(TITLES[ENTRANT],
callback_data=ENTRANT),
        InlineKeyboardButton(TITLES[STUDENT],
callback_data=STUDENT),
        InlineKeyboardButton(TITLES[TEACHER],
callback_data=TEACHER)
    ),
    )
    return InlineKeyboardMarkup(keyboard)
```

При виборі облікового запису існує вірогідність помилки, тому потрібно реалізувати можливість змінити облікові дані, для цього реалізовані UPDATE запити, які міняють статус профілю не замінюючи статичні поля, які для всіх користувачів однакові, на лістингу 3.4 зображена функція реєстрації студента.

Лістинг 3.4 – функція реєстрації студента.

```
def register(user_id, group, fullname=None, link=None):
    conn = psycopg2.connect(config.DATABASE)
    cursor = conn.cursor()
    try:
        cursor.execute("INSERT INTO public.students(id, fullname,
link, stud_group) VALUES (%s, %s, %s, %s)",
                        (user_id, fullname, link, group))
    except UniqueViolation:
        cursor.close()
        conn.close()
        conn = psycopg2.connect(config.DATABASE)
        cursor = conn.cursor()
        cursor.execute("UPDATE public.students SET stud_group=%s,
link=%s, teacher=FALSE, entrant=FALSE WHERE id=%s",
                        (group, link, user_id))
    cursor.close()
    conn.commit()
```

Обробка відповіді користувача при створенні бота є досить складний процес. Для обробки повідомлень передбачений один клас `MessageHandler`, за допомогою цього класу можна отримати вміст повідомлення і в залежності від змісту вивести користувачу відповідь, але коли користувач присилає текст, який після цього буде відправлятися, або зберігатись, то по тексту визначити яку дію потрібно виконати з повідомлення не можливо, було прийнято рішення створення додаткових полів в базі, кожне поле відповідає за статус користувача, і в залежності від статусу бот визначає, яке повідомлення буде приймати від користувача.

Обробка статусу відправки повідомлення студентам зображена на лістингу 3.5.

Лістинг 3.5 Обробка текстового повідомлення користувача.

```
if group_message_mode:
    text = update.message.text
    groups =
utils.get_bot_group_instance(update.effective_user['id']).rstrip().
split(' ', ' ')
    id_s = utils.select_students_id(groups)
    for id in id_s:
        context.bot.send_message(
            chat_id=id,
```

```

        text='Вам пише: ' +
update.effective_user.full_name + '\n' + update.effective_user.link
+ '\n' + text,
        reply_markup=back_keyboard()
    )
    context.bot.send_message(
        chat_id=update.effective_user['id'],
        text='Текст було успішно надіслано',
        reply_markup=teacher_keyboard()
    )

utils.delete_bot_group_instance(update.effective_user['id'])
utils.off_message_mode(update.effective_user['id'])

```

Вивід потрібної інформації здійснюється шляхом запитів до сервера телеграм, за допомогою прикладного програмного інтерфейсу python-telegram-bot, для конкретної інформації існують кнопки, яким присвоюються функції, реалізовані в боті, з їх допомогою інформація персоналізується при заповнених полях облікового запису користувача та виводиться в чаті.

3.2 Типовий сценарій роботи з ботом

Кожен бот в мережі Telegram має ім'я та псевдонім.

Ім'я бота – це надпис, який відображається при пошуку бота чи при спілкуванні у вигляді підпису контакту.

Псевдонім – це унікальний ідентифікатор користувачів в телеграмі, вони не повторюються, тому знайти свого знайомого або бота по псевдоніму вимагає менше зусиль ніж пошук по назві. Щоб розпочати пошук по псевдоніму потрібно ввести символ «@» і розпочати введення псевдоніму.

Назва та псевдонім бота, який був створений співпадають, тому не виникне путанини в процесі пошуку.

Пошук бота можна здійснити різними методами:

- перейти по посиланню;
- ввести псевдонім в пошуку;
- спробувати віднайти його по назві.

На рисунку 3.4 показано перший спосіб пушуку бота (по посиланню).



Рисунок 3.4 – Пошук бота по посиланню

На рисунку 3.5 зображено пошук бота за другим способом (за псевдонімом).

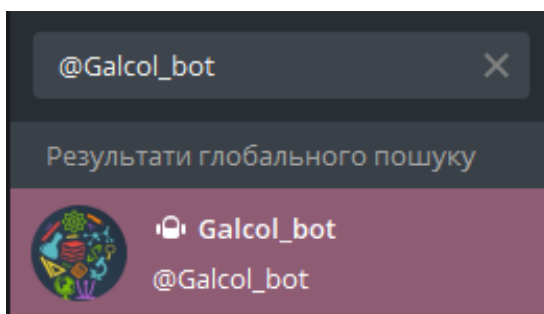


Рисунок 3.5 – Пошук за псевдонімом

На рисунку 3.6 зображено пошук бота за третім способом (по назві):

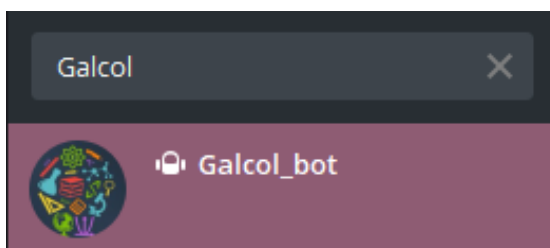


Рисунок 3.6 – Пошук по назві

Після відкриття розмови з ботом, внизу з'являється кнопка «Розпочати» і вікно з коротким описом бота. Вікно нерозпочатої розмови зображене на рисунку 3.7.

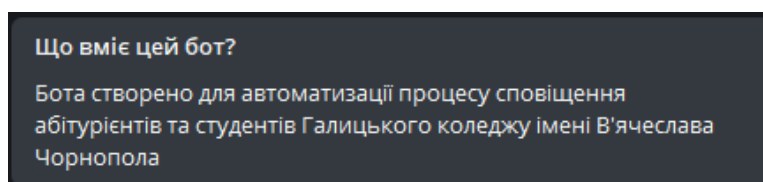


Рисунок 3.7 – Вікно бота

Після початку розмови бот нам відправляє вступне повідомлення з можливістю вибору статусу користувача, повідомлення зображене на рисунку 3.8.

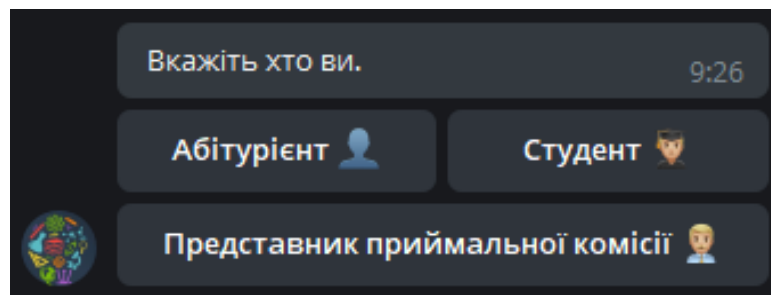


Рисунок 3.8 – Вступне повідомлення бота

Вибравши статус Абітурієнта, повідомлення від бота редагується, Клавіатура вибору статусу замінюється кнопками відображення FAQ і повернення назад у вибір статусу, яка переадресує абітурієнта назад у головне меню, це надасть йому можливість змінити статус на студента після всупу.

Екран абітурієнта зображено на рисунку 3.9.

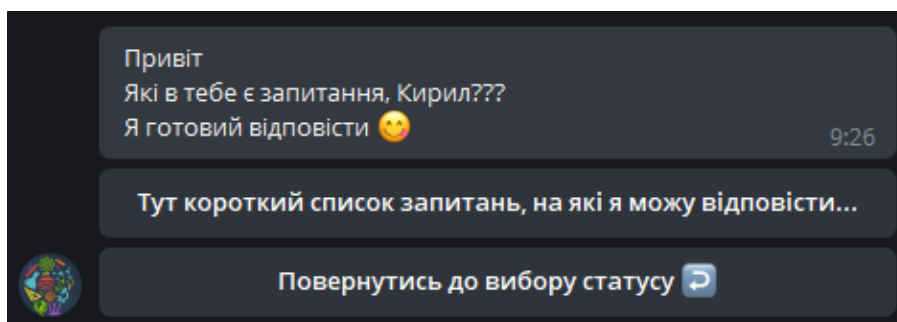


Рисунок 3.9 – Екран абітурієнта

Якщо, користувач вирішив переглянути список доступних команд, і натисне на першу кнопку, виведеться список з 3 пунктів, де коротко відображаються запитання які можна використати як приклад, щоб зрозуміти як працює бот.

Друга сторінка списку запитань зображена на рисунку 3.10

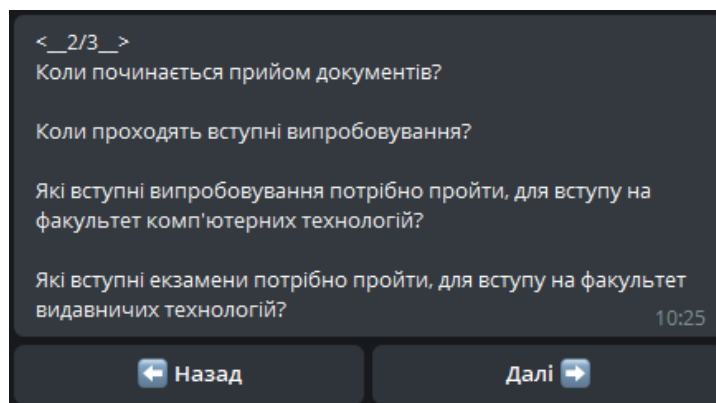


Рисунок 3.10 – Друга сторінка списку запитань

В боті реалізовано алгоритм «Відстань Дамерау - Левенштейна» [9] – це алгоритм пошуку найбільш схожого слова до того, яке ввів користувач шляхом визначення кількості операцій заміни щоб з повідомлення користувача вийшло повідомлення, яке записане в боті [10], завдяки цьому алгоритму користувач може дозволити собі робити помилки, або навіть використати інше слово замість вказаного в списку запитань і бот зрозуміє, що користувач мав на увазі, для прикладу було взято команду «Коли починається прийом документа?», в повідомлення боту було відправлено текст «Коли почнеться прийом доків?», на що бот відповів проханням уточнити пошукові дані, а саме: ввести ступінь про яку користувач хотів дізнатись, тому що дати і терміни вступної компанії відрізняються для вступників на Бакалаврат, та вступників до фахового молодшого бакалаврату. Для зручності уточнення виводиться дві кнопки, які будуть відправляти боту саме ту інформацію, яка записана в функціонал кнопки і користувач однозначно отримає очікувану відповідь.

Відповідь бота зображено на рисунку 3.11.

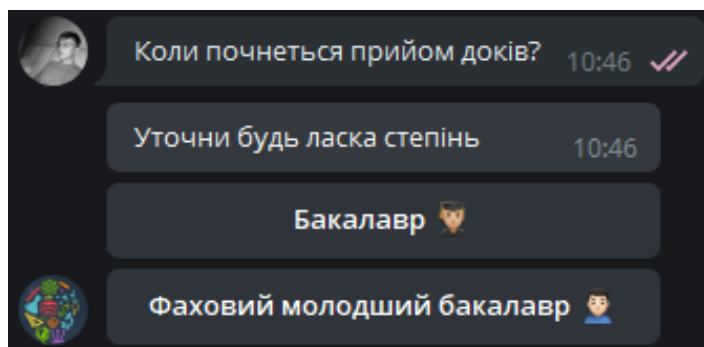


Рисунок 3.11 – Відповідь бота на запитання

Після відповіді було обрано перший варіант: «Бакалавр», після чого бот вивів всю потрібну користувачу інформацію.

Інформація зображена на рисунку 3.12.

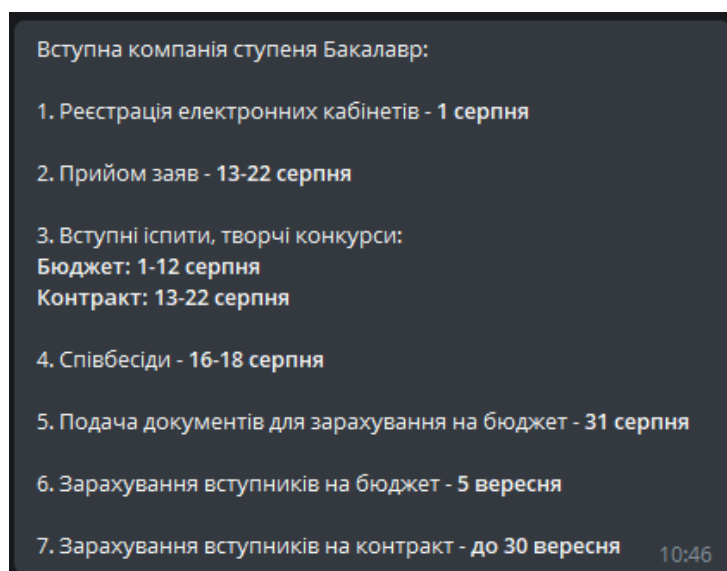


Рисунок 3.12 – Відповідь бота на вибраний користувачем ступінь

Якщо бот не розпізнає запитання, він попередить про це користувача та запитатиме, чи відправляти дане повідомлення в приймальну комісію, користувачу буде запропоновано лише два варіанти відповіді: «Так» і «Ні», якщо користувач вибрав «Так», то повідомлення буде переслано в приймальну комісію, таким чином представник комісії зможе перейти в особисту переписку з користувачем і проконсультувати його особисто, якщо вибрано «Ні», повідомлення видалиться з

буфера бота. Результат відповіді на некоректне запитання відображено на рисунку 3.13.

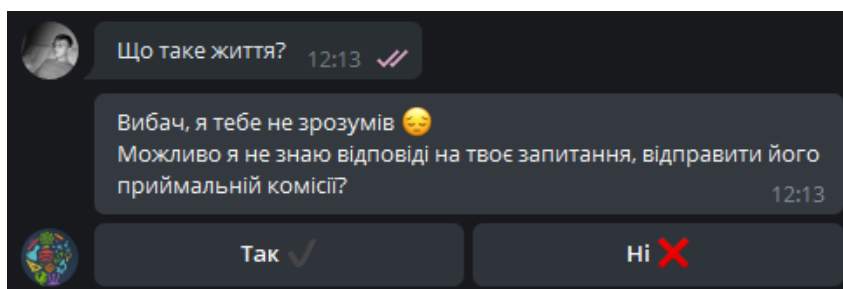


Рисунок 3.13 – Результат некоректного запитання

Результат відповіді бота на обраний користувачем варіант «Так» зображено на рисунку 3.14.

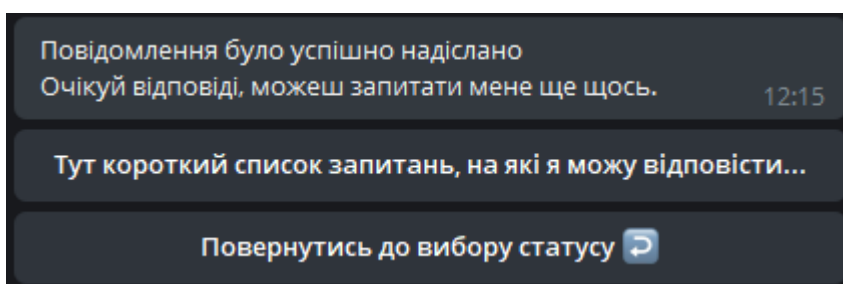


Рисунок 3.14 – Користувач обрав відповідь «Так»

Результат відповіді бота на обраний користувачем варіант «Ні» зображено на рисунку 3.15.

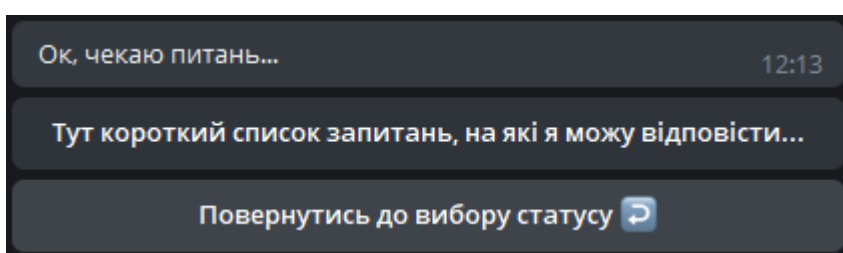


Рисунок 3.15 – Користувач обрав відповідь «Ні»

У Абітурієнта на відміну від інших користувачів є можливість вийти назад у меню вибору статусу, це зв'язано з тим, що окрім статусу «Абітурієнт», в боті передбачений статус «Студент», завдяки цьому першокурсники Галицького коледжу імені В'ячеслава Чорновола зможуть швидше дізнатись інформацію один про одного. Після вибору студентського статусу відобразиться клавіатура

вибору групи, в якій також присутня кнопка для повернення назад, на випадок якщо користувач помилився.

Клавіатуру вибору статусу зображено на рисунку 3.16.

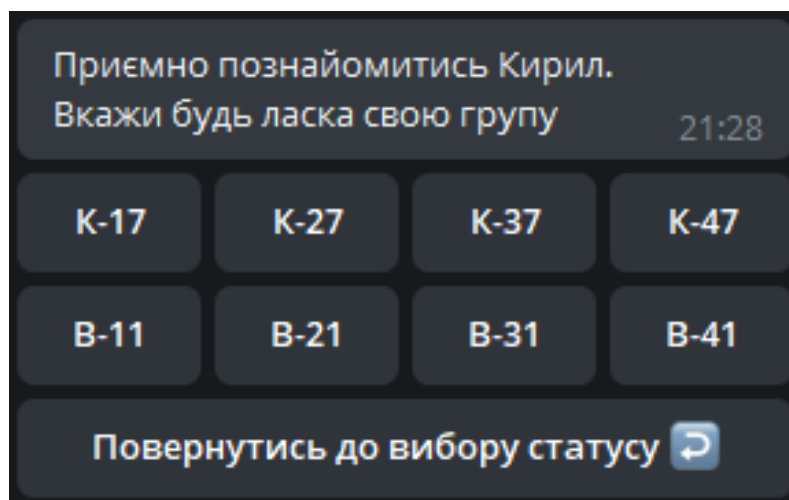


Рисунок 3.16 – Клавіатура вибору групи

Коли студент вибрав свою групу, повідомлення знову редагується і виводиться меню користувача. В меню розташовано три кнопки управління:

- «Налаштування»;
- «Група»;
- «Розклад занять».

Інтерфейс студента зображена на рисунку 3.17.

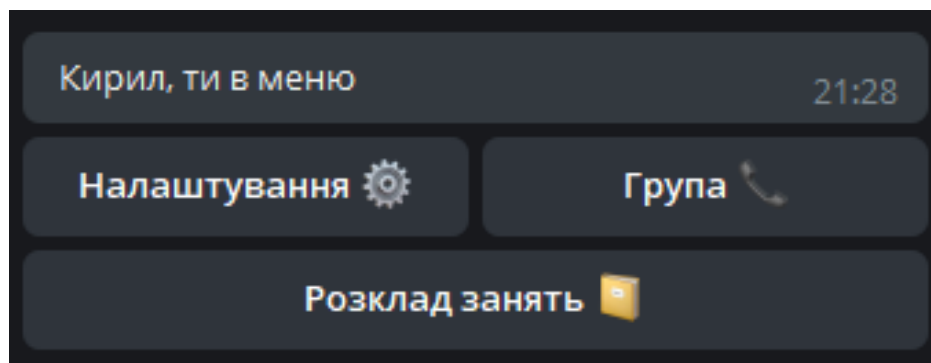


Рисунок 3.17 – Інтерфейс студента

Кнопка «Налаштування» дозволяє користувачу змінити свої облікові дані: додати опис профілю, номер телефону, змінити групу та повернутись назад.

Меню «Налаштування» зображено на рисунку 3.18.

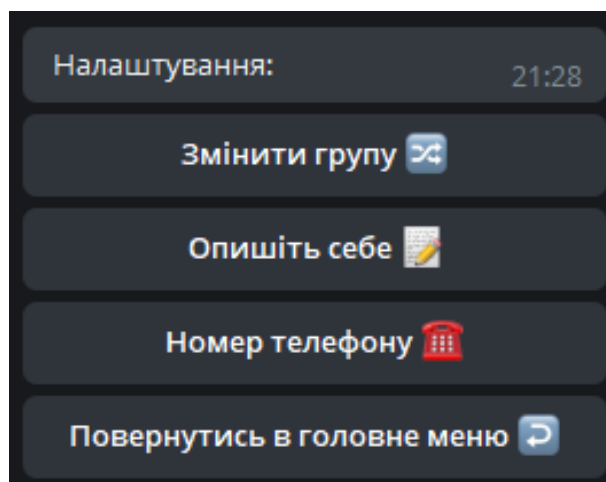


Рисунок 3.18 – Меню налаштувань

При виборі кнопки Змінити групу в клавіатурі налаштувань виводиться нашу поточну групу та повторно запитується чи користувач точно хоче її змінити, а також надано можливість повернутись до головного меню, або налаштувань.

Інтерфейс кнопки «Змінити групу» зображено на рисунку 3.19.

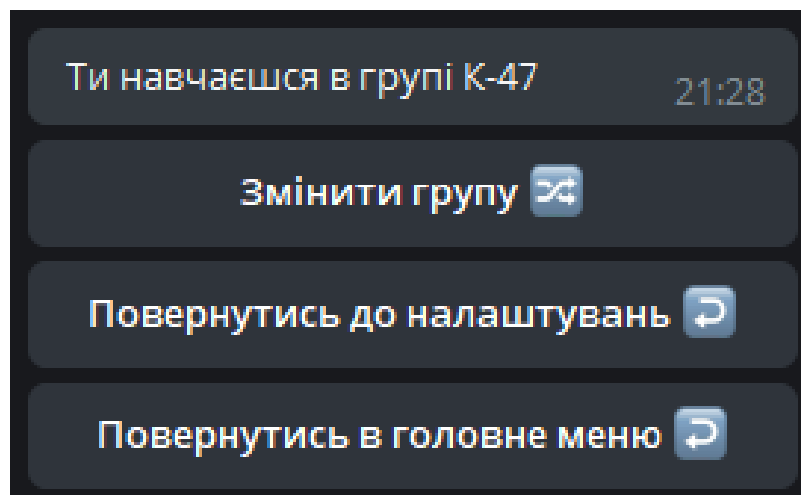


Рисунок 3.19 – Інтерфейс кнопки «Змінити групу»

При підтвердженні дії повторно натиснувши на цю кнопку, користувача переносить на етап повторного вибору групи, з цього меню також можна повернутись до вибору статусу. Меню вибору групи зображено на рисунку не відрізняється від попереднього, міняється тільки текст повідомлення.

Текст зображено на рисунку 3.20.

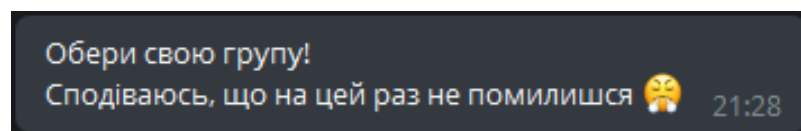


Рисунок 3.20 – Мтекстове наповнення повідомлення при повторному виборі групи

Також в налаштуваннях присутня кнопка «Опишіть себе», при виборі користувачим цієї кнопки бот попросить користувача коротко про себе розповісти, також зробить відповідь користувача відповіддю на своє запитання засобами телеграм.

Відповідь бота на вибір користувача зображено на рисунку 3.21.



Рисунок 3.21 – Бот очікує на опис

Після написання тексту він збережеться і бот сповістить про це користувача. Результат відповіді користувача зображено на рисунку 3.22.

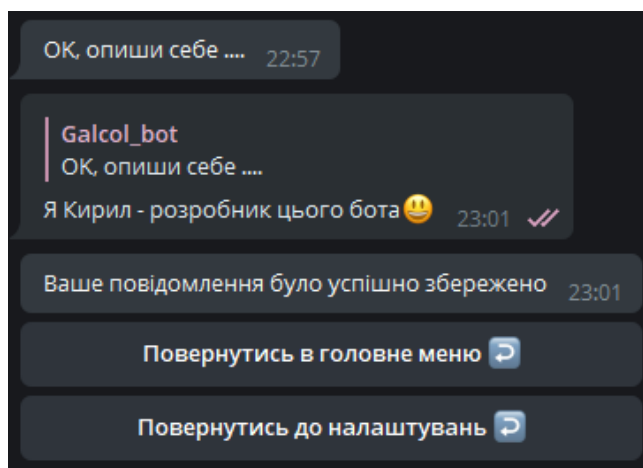


Рисунок 3.22 – Зображення відповіді бота

Також є можливість вказати номер телефону, процес збереження зображено на рисунку 3.23.

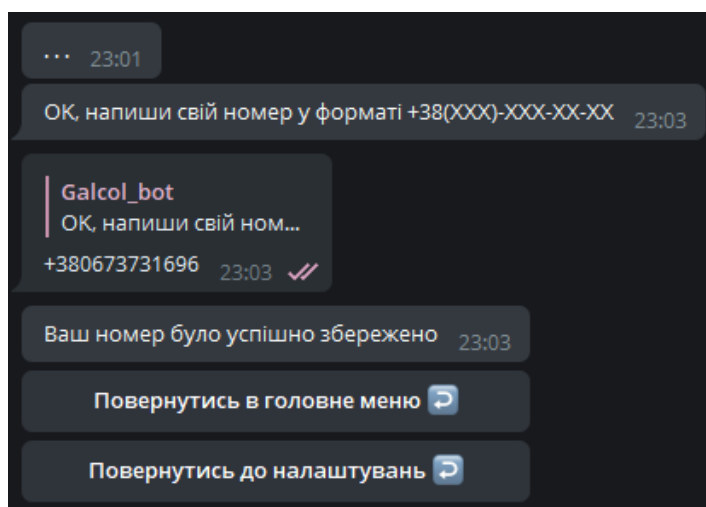


Рисунок 3.23 – Процес збереження номера телефону

При введенні неправильного номеру бот сповістить про це користувача та надасть можливість повторити спробу. На рисунку 3.24 зображено результат введення невірного номеру.

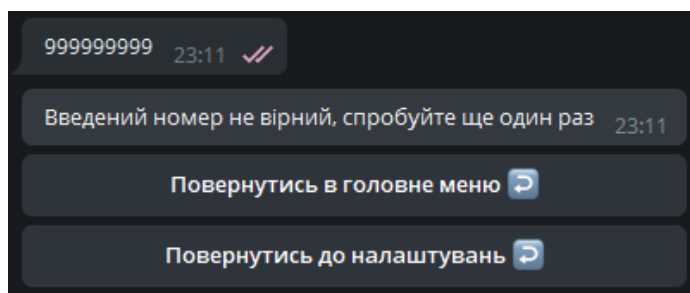


Рисунок 3.24 – Результат введення невірного номеру

В головному меню студента також є кнопки:

- «Група» - виводить інформацію про одногрупників (рис. 3.25);
- «Розклад занять» - виводить посилання на розклад занять, який розташований на сторінці Галицького коледжу (рис. 3.26).

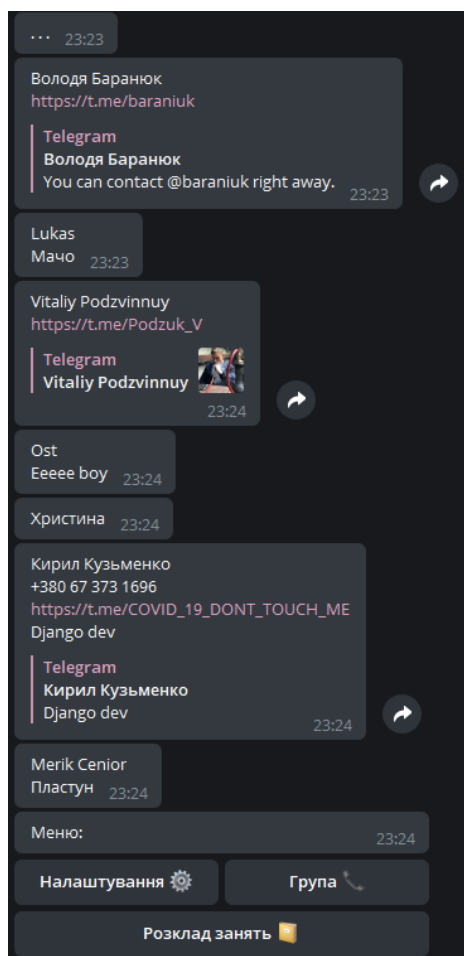


Рисунок 3.25 – Відображення інформації про одногрупників

Посилання на розклад занять зображено на рисунку 3.26.

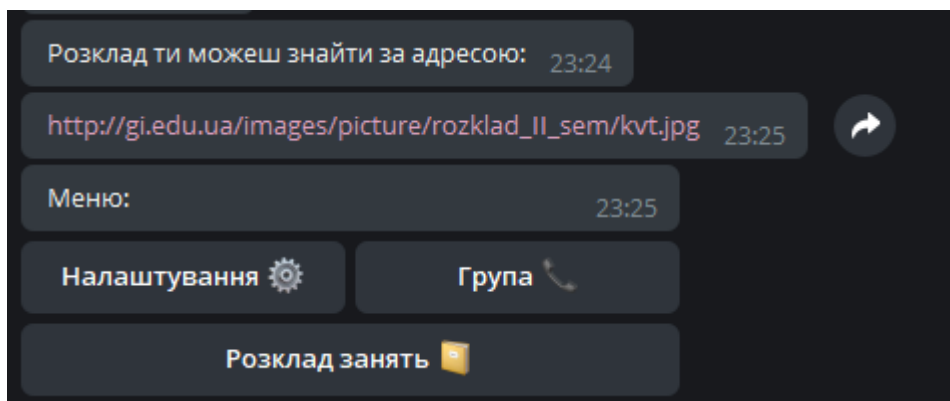


Рисунок 3.26 – Посилання на розклад занять

При виборі профілю представника приймальної комісії бот запитує користувача пароль до даного облікового запису в зв'язку з тим, що статус представника комісії надає можливість здійснювати розсилки всім групам відділення комп'ютерних та видавничих технологій і абітурієнтам, щоб запобігти дезінформування абітурієнтів та студентів було передбачено захист цього типу статусу. Запит пароллю зображено на рисунку 3.27.

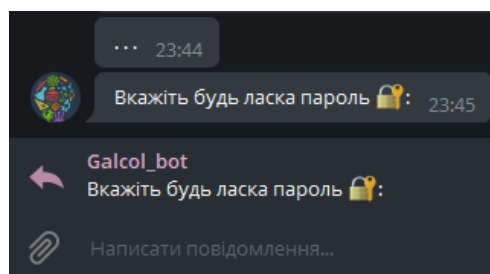


Рисунок 3.27 – Запит пароллю при спробі ввійти як викладач

Після введення не вірного пароллю бот надає можливість повернутись до вибору статусу, або спробувати ще один раз.

Результат введення невірного паролю зображено на рисунку 3.28.

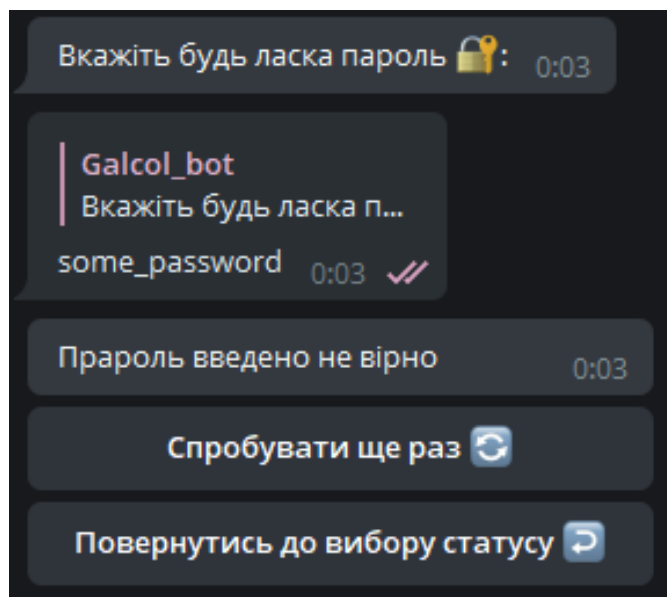


Рисунок 3.28 – Результат введення невірного паролю

Коли пароль введено вірно, відкривається інтерфейс представника комісії. В меню можна побачити три кнопки управління профілем:

- «Написати абітурієнтам» – кнопка створена для здійснення розсилок абітурієнтам;
- «Вибрати групи для розсилки» – кнопка створена для розсилки повідомлення групав, але спершу користувача чекає інтерфейс вибору груп для розсилки;
- «Приймати повідомлення» – кнопка створена для включення, або відключення режиму прийняття повідомлень.

Інтерфейс представника приймальної комісії зображено на рисунку 3.29.

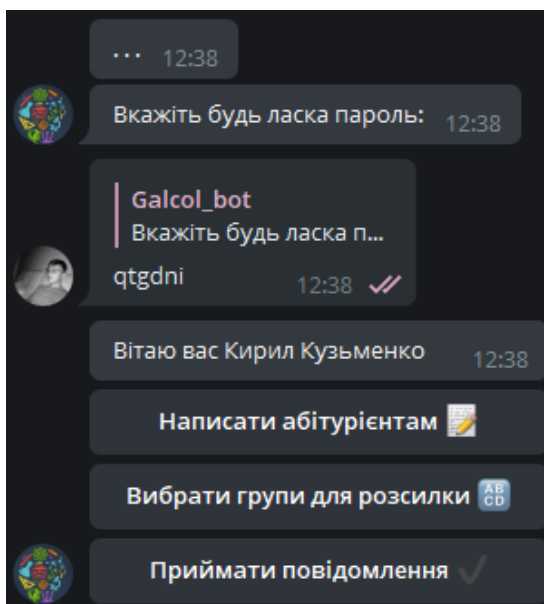


Рисунок 3.29 – Інтерфейс вчителя

У представника комісії є можливість написати студентам та абітурієнтам, на рисунку 3.30 зображений інтерфейс вибору груп студентів, яким буде відправлено майбутнє повідомлення.

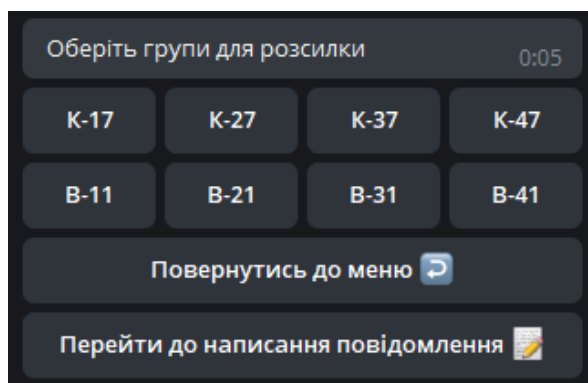


Рисунок 3.30 – Інтерфейс вибору груп для надсилання повідомлень

Для прикладу було вибрано групи К-47 та К-27, після вибору потрібних груп, ми можемо перейти до написання повідомлення, нажавши на кнопку «Перейти до написання повідомлення».

Результат вибору груп К-27 та К-47 зображено на рисунку 3.31.

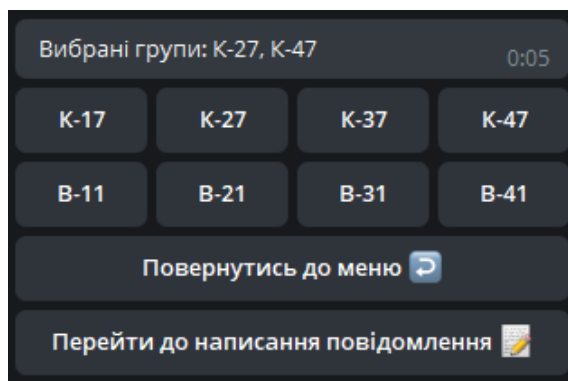


Рисунок 3.31 – Результат вибору користувачем груп К-27 та К-47

Після того як користувач вибрав потрібні групи та перейшов до написання повідомлення, бот попросить написати повідомлення студентам.

Результат відповіді бота на повідомлення зображено на рисунку 3.32.

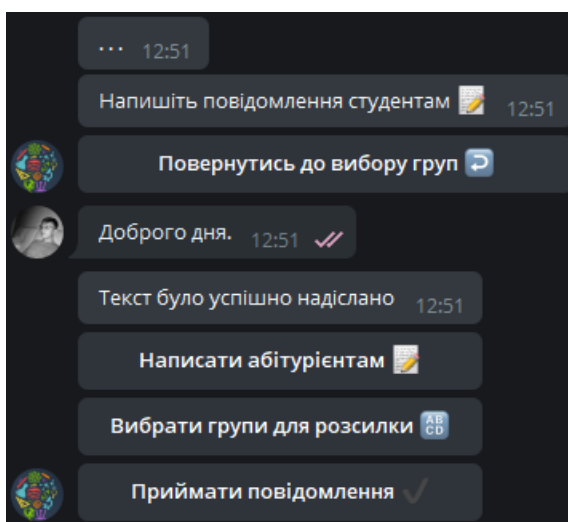


Рисунок 3.32 – Результат відповіді бота на повідомлення адресоване студентам

Також на рисунку 3.33 зображено повідомлення, яке прийшло студенту

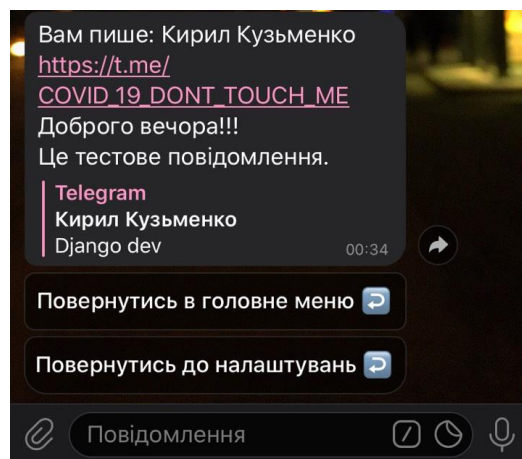


Рисунок 3.33 – Повідомлення, адресоване студенту

В майбутніх оновленнях буде додано систему авторизації для студентів, також розширені можливості для старости групи і можливість студентами здійснювати голосування. Бажано розширити функціонал вчительського облікового запису, додати можливість перегляду студентів окремих груп та модернізувати систему авторизації, додавши можливість зміни паролю після першого входу в обліковий запис. Також буде розширено список запитань на які бот зможе надати відповідь.

4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБҐРУНТУВАННЯ

4.1 Аналіз ринку

«Ера чат-ботів уже почалась, – переконаний Валерій Даниленко. – Вдосконалення технології штучного інтелекту й популярність месенджерів безпосередньо впливають на цей зростаючий тренд. Чат-боти стали для брендів сервісами додаткової вартості, які прямо не генерують дохід, а допомагають компаніям краще обслуговувати клієнтів та досягати своєї мети, а також підвищують продуктивність». [11]

На сьогоднішній день ступінь розвитку чат-ботів дозволяє їм значно зменшити операційні витрати, завдяки ботам оформлюються замовлення, проводиться консультування клієнтів, велика кількість користувачів готові отримувати інформацію через чат-ботів перед тим, як звертатися до операторів, тому що простіше буде отримати потрібну інформацію використовуючи привичний інтерфейс месенджера. Тренд все більше укріплюється в суспільстві, боти вже здійснюють більше вісімдесяти п'яти відсотків управління комунікаційними процесами з постачальниками послуг, без участі менеджерів.

На відміну від європейського ринку, де чат боти вже давно звичне явище, на українському ринку вони з'явилися відносно недавно: дві тисячі вісімнадцятого року, на сьогодні здійснення оплат через ботів, це вже норма навіть в Україні. Більшість українських банків мають своїх банків: ТАСКОМБАНК, ПРИВАТБАНК, МОНОБАНК та інші.

«Ми віримо, що майбутнє за платіжними чат-ботами й бачимо великий потенціал для розвитку цього напрямку», – підсумував Валерій Даниленко.

4.2 Розрахунок витрат на проектування

Законом України від «Про Державний бюджет України на 2020 рік» встановлено мінімальну заробітну плату: у місячному розмірі з 1 січня - 4723 гривні; у погодинному розмірі з 1 січня – 28,31 гривні, прожитковий мінімум для працездатних осіб: з 1 січня 2020 року - 2102 гривні, з 1 липня - 2197 гривня, з 1 грудня - 2270 гривня. Податкова соціальна пільга становить у 2018 році для

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		47

будь-якого платника 1051 грн. Граничний розмір зарплати, до якої застосовується податкова соціальна пільга, у 2020 році дорівнює 2940 грн.

Працівникові №1 нараховані за повний відпрацьований місяць 15 000 грн. Податкова соціальна пільга до такої заробітної плати не застосовується, оскільки вона більша за граничний розмір доходу, який дає право на податкову соціальну пільгу.

- Рахуємо податок на доходи фізичних осіб: $20\,000 * 18\%$ (ставка податку на доходи фізичних осіб) = 3 600 грн.
- Військовий збір: $20\,000 * 1,5\%$ (ставка військового збору) = 300 грн.
- Єдиний внесок: $20\,000 * 22\%$ (ставка ЄСВ) = 4 400 грн.
- Утримання – 3 900 грн. (3 600 грн. + 300 грн.)
- До виплати працівникові – 16 100 грн. (20 000 грн. – 3 600 грн. – 300 грн.)

Працівнику №2 нараховані за повний відпрацьований місяць 15 000 грн. Податкова соціальна пільга до такої заробітної плати не застосовується, оскільки вона більша за граничний розмір доходу, який дає право на податкову соціальну пільгу.

- Рахуємо податок на доходи фізичних осіб: $15\,000 * 18\%$ (ставка податку на доходи фізичних осіб) = 2 700 грн.
- Військовий збір: $15\,000 * 1,5\%$ (ставка військового збору) = 225 грн.
- Єдиний внесок: $15\,000 * 22\%$ (ставка ЄСВ) = 3 300 грн.
- Утримання – 2 925 грн. (2 700 грн. + 225 грн.)
- До виплати працівникові – 12 075 грн. (15 000 грн. – 2 700 грн. – 225 грн.)

Працівнику №3 нараховані за повний відпрацьований місяць 12 000 грн. Податкова соціальна пільга до такої заробітної плати не застосовується, оскільки вона більша за граничний розмір доходу, який дає право на податкову соціальну пільгу.

- Рахуємо податок на доходи фізичних осіб: $12\,000 * 18\%$ (ставка податку на доходи фізичних осіб) = 2 160 грн.

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		48

- Військовий збір: $12\,000 * 1,5\%$ (ставка військового збору) = 180 грн.
- Рахуємо єдиний внесок: $12\,000 * 22\%$ (ставка ЄСВ) = 2 640 грн.
- Утримання – 2 340 грн. (2 160 грн. + 180 грн.)
- До виплати працівникові – 9 660 грн. (12 000 грн. – 2 160 грн. – 180 грн.)

Заробітня плата учасників проєктування наведено у таблиці 4.1.

Таблиця 4.1 – Розрахування заробітної плати

	Посада	Оклад, грн./міс.	Відрахування, грн./міс.	Кількість		Сума, грн.
1	Керівник проєкту	20000	3600	чол. 1	міс. 1	16 100
2	Програміст	15000	2 925	чол. 1	міс. 1	12 075
3	Тестувальник	12000	2 340	чол. 1	міс. 1	9 660
		Усього зарплати:				37 835

Контрагентські роботи становлять 4 540,20 грн. ($37\,835 * 12\%$).
Співробітники не були у відрядженні, тому витрат на відрядження не має. Інші
прямі витрати за місяць становлять – 15 890,70 грн. ($37\,835 * 42\%$). Усього
прямих витрат за місяць 58 265,90 грн. ($37\,835\text{грн.} + 4\,540,20\text{ грн.} + 15\,890,70\text{ грн.}$)

Накладні витрати за місяць становлять 17 479,77 грн. ($58\,265,90\text{ грн.} * 30\%$).
Планові накопичення за місяць становлять 15 149,13 грн. ($(58\,265,90\text{ грн.} + 17\,479,77\text{ грн.}) * 20\%$). Усього кошторисна вартість проєкту 90 894,80 грн.
($58\,265,90\text{ грн.} + 17\,479,77\text{ грн.} + 15\,149,13\text{ грн.}$). Податок на додану вартість становить 18 178,96 грн. ($90\,894,80\text{ грн.} * 20\%$). Договірна ціна становить 109 073,76 грн ($90\,894,80\text{ грн.} + 18\,178,96\text{ грн.}$)

Загальний кошторис витрат протягом усього проєктування та по його завершенні наведено у таблиці 4.2.

Таблиця 4.2 - Кошторис витрат на проєктування

Найменування статей витрат	Сума, грн
1 Зарплата проєктувальників.	37 875
2. Відрахування на соціальні потреби.	0
3. Контрагентські роботи і послуги.	4 540,20
4. Витрати на відрядження.	0
5. Інші прямі витрати.	15 890,70
6. Усього прямих витрат.	58 265,90
7. Накладні витрати.	17 479,77
8. Планові накопичення.	15 149,13
9. Усього, кошторисна вартість проєкту.	90 894,80
10. Податок на додану вартість.	18 178,96
11. Загалом, договірна ціна розробки	109 073,76

4.3 Обґрунтування необхідності розробки

Зважаючи на високу популярність, створення бота для Галицького коледжу допоможе:

- Процесу привернення уваги вступників.
- Процесу впровадження інновацій.
- Йти коледжу в ногу з часом.

Ринок чат-ботів розвивається і вже скоро наявність чат-бота буде стандартом для кожної організації, створення бота для коледжу допоможе випередити ІТ становище в області, або навіть в цілій Україні, а також підніме шанс вибору коледжу для нових студентів.

ВИСНОВКИ

При виконанні дипломного проєкту було проаналізовано та узгоджено з приймальною комісією та вчителями загальні питання студентів, які часто виникають у більшості першокурсників.

Було проведено паралелі з сайтами, визначено плюси та мінуси створення дипломного проєкту, в результаті переваги в сторону створення бота було спроектовано структуру, інтерфейс бота, визначено засоби реалізації та чітко поставлено ціль яку, в результаті було виконано.

Боту було надано можливість консультувати абітурієнтів, майже не залучаючи до цього сторонніх осіб, також було надано функцію виведення додаткової інформації для студентів, а саме: інформацію про групу, також важливою частиною проєкту було надання членам приймальної комісії можливості здійснення розсилок повідомлень по окремим групам та абітурієнтам, на випадок, якщо потрібно буде максимально швидко сповістити студентам та/або абітурієнтам важливу інформацію.

Проведено аналіз ринку, визначено вартість витрат на розробку подібного бота, приблизну тривалість розробки, кількість залучених робітників, і ще один раз було доказано високу популярність, актуальність і перспективу використання ботів.

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		51

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ТОП-40 популярних телеграм-ботів в Україні. *PAYSPACE*: вебсайт. URL: <https://psm7.com/uk/news/top-40-boty-telegram-v-ukraine.html>
(дата звернення: 10.02.2020)
2. Огляд Telegram. *TexTerra*: веб-сайт. URL: <https://texterra.ru/blog/pochemu-stoit-poprobovat-telegram-obzor-messendzhera.html> (дата звернення: 15.03.2020)
3. Приклади ботів. *Telegram*: вебсайт. URL: <https://tlgrm.ru/docs/bots/samples>
(дата звернення: 16.03.2020)
4. python-telegram-bot. *GitHub*: вебсайт. URL: <https://github.com/python-telegram-bot/python-telegram-bot> (дата звернення: 18.04.2020)
5. PostgreSQL download. *PostgreSQL*: вебсайт. URL: <https://www.postgresql.org/>
(дата звернення: 19.04.2020)
6. PgAdmin download. *PgAdmin*: вебсайт. URL: <https://www.pgadmin.org/>
(дата звернення: 19.04.2020)
7. Python. *Wikipedia*: вебсайт. URL: <https://uk.wikipedia.org/wiki/Python>
(дата звернення: 20.05.2020)
8. Python 3.8.3 documentation. *Python*: вебсайт. URL: <https://docs.python.org/3/>
(дата звернення: 20.05.2020)
9. Розпізнавання тексту з помилками. *stackoverflow*: вебсайт. URL: <https://ru.stackoverflow.com/questions/859694/Розпізнавання-тексту-з-помилками>
(дата звернення: 25.05.2020)
10. Відстань Дамерау-Левенштейна. *Знайомо*: вебсайт. URL: http://znaimo.com.ua/Відстань_Левенштейна (дата звернення: 03.06.2020)
10. Цитата. *LEMARBET*: вебсайт. URL: <https://lemarbet.com/ua/razvitie-internet-magazina/boty-v-telegram/> (дата звернення: 04.06.2020)

ДОДАТКИ

Додаток А

Розпізнавання повідомлень

utils.py:

```
ENTRANT_COMMANDS = [  
    'Які документи потрібні для вступу',  
  
    'Адреса',  
    'Графік роботи',  
    'Графік роботи приймальної комісії',  
    'Хто має право на пільги',  
  
    'Коли починається прийом документів',  
    'Коли проходять вступні випробовування',  
    'Коли будуть вступні',  
  
    'Які екзамени потрібно пройти',  
    'Які вступні випробовування потрібно пройти',  
    'Які вступні випробовування потрібно пройти для вступу на  
факультет комп\'ютерних технологій',  
    'Які екзамени потрібно здати для вступу в комп\'ютерне  
відділення',  
    'Які вступні екзамени потрібно пройти для вступу на  
факультет видавничих технологій',  
  
    'Які спеціальності за ступенем фахового молодшого  
бакалавра',  
    'На яку спеціальність поступити щоб отримати ступінь  
бакалавра',  
    'Які є спеціальності з ступенем бакалавра',  
    'Чи є в комп\'ютерному відділенні бакалаврат',  
    'Я можу піти вчитись на бакалавра після коледжу?',  
    'Які спеціальності за ступенем кваліфікованого  
працівника?',  
    'Спеціальності з бакалавратом.',  
  
    'Чи потрібно подавати сертифікати ЗНО після 11 класу?',  
    'Потрібен сертифікат, якщо поступати після 11 класу?',  
    'Чи потрібна медична довідка в коледж?',  
    'Контакти',  
    'Контактні телефони',  
    'Мобільний телефон',  
    'Номера',  
    'Телефон',  
    'Мобільний',  
  
    'Скільки вартує навчання?',  
    'Скільки коштує навчання?',  
    'Скільки коштує навчання на видавника?',
```

					ДП.КН 20.408.03.000 ПЗ	Арк.
						53
Зм.	Арк.	№ докум	Підпис	Дата		

```

        'Скільки кштує навчання на журналіста?',
        'Скільки коштує навчання на видавничу справу та
редагування?',
        'Скільки коштує навчання на юриста?',
        'Скільки кштує навчання на правовому факультеті?',
        'Скільки коштує навчання на офіціанта?',
        'Скільки коштує навчання на фіца?',
        'Скільки коштує навчання на готельно-ресторанну справу?'
    ]

    STUDENT_COMMANDS = [
        ''
    ]

    def entrant_commands(text):
        import numpy as np
        array = np.array(ENTRANT_COMMANDS)
        from pyxdameraulevenshtein import
normalized_damerau_levenshtein_distance_ndarray
        result = list(zip(ENTRANT_COMMANDS,
list(normalized_damerau_levenshtein_distance_ndarray(text,
array))))
        command, rate = min(result, key=lambda x: x[1])
т.е. text среди all_commands нет
        if rate > 0.40:
            return

        return command

```

Додаток Б

Ініціалізація клавіатур бота

galcol_bot.py:

```

def degree_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[BACHELOR],
callback_data=BACHELOR),
        ),
        (
            InlineKeyboardButton(TITLES[JUNIOR_BACHELOR],
callback_data=JUNIOR_BACHELOR),
        )
    )
    return InlineKeyboardMarkup(keyboard)

def next_page_keyboard():
    keyboard = (

```

```

        InlineKeyboardButton(TITLES[NEXT_PAGE],
callback_data=NEXT_PAGE),
    ),
)
return InlineKeyboardMarkup(keyboard)

def payment_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[LAW_PAYMENT],
callback_data=LAW_PAYMENT),
        ),
        (
            InlineKeyboardButton(TITLES[JOURNALIST_PAYMENT],
callback_data=JOURNALIST_PAYMENT),
        ),
        (
            InlineKeyboardButton(TITLES[HOTEL_PAYMENT],
callback_data=HOTEL_PAYMENT),
        )
    )
    return InlineKeyboardMarkup(keyboard)

def vstup_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[COMPUTER],
callback_data=COMPUTER),
        ),
        (
            InlineKeyboardButton(TITLES[JOURNALIST],
callback_data=JOURNALIST),
        )
    )
    return InlineKeyboardMarkup(keyboard)

def previous_page_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[PREVIOUS_PAGE],
callback_data=PREVIOUS_PAGE),
        ),
    )
    return InlineKeyboardMarkup(keyboard)

def change_page_keyboard():
    keyboard = (

```



```

        InlineKeyboardButton(TITLES[PREVIOUS_PAGE],
callback_data=PREVIOUS_PAGE),
        InlineKeyboardButton(TITLES[NEXT_PAGE],
callback_data=NEXT_PAGE)
    ),
)
return InlineKeyboardMarkup(keyboard)

# 2 кнопки:
# 1 - повернутись до головного меню
# 2 - повернутись до налаштувань
def back_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[BACK_TO_BASE_MENU],
callback_data=BACK_TO_BASE_MENU),
        ),
        (
            InlineKeyboardButton(TITLES[BACK_TO_SETTINGS],
callback_data=BACK_TO_SETTINGS),
        )
    )
    return InlineKeyboardMarkup(keyboard)

def select_group_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[SELECT_K_17],
callback_data=SELECT_K_17),
            InlineKeyboardButton(TITLES[SELECT_K_27],
callback_data=SELECT_K_27),
            InlineKeyboardButton(TITLES[SELECT_K_37],
callback_data=SELECT_K_37),
            InlineKeyboardButton(TITLES[SELECT_K_47],
callback_data=SELECT_K_47)
        ),
        (
            InlineKeyboardButton(TITLES[SELECT_V_11],
callback_data=SELECT_V_11),
            InlineKeyboardButton(TITLES[SELECT_V_21],
callback_data=SELECT_V_21),
            InlineKeyboardButton(TITLES[SELECT_V_31],
callback_data=SELECT_V_31),
            InlineKeyboardButton(TITLES[SELECT_V_41],
callback_data=SELECT_V_41)
        ),
        (
            InlineKeyboardButton(TITLES[BACK_TO_TEACHER_MENU],
callback_data=BACK_TO_TEACHER_MENU),
        ),
    )

```

```

        InlineKeyboardButton(TITLES[CONFIRM],
callback_data=CONFIRM),
    )
)
return InlineKeyboardMarkup(keyboard)

def back_to_status():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[SHOW_QUESTION_EXAMPLE],
callback_data=SHOW_QUESTION_EXAMPLE),
        ),
        (
            InlineKeyboardButton(TITLES[BACK_TO_CHOSE_STATUS],
callback_data=BACK_TO_CHOSE_STATUS),
        )
    )
    return InlineKeyboardMarkup(keyboard)

# Клавіатура вибору групи
def set_group_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[SET_K_17],
callback_data=SET_K_17),
            InlineKeyboardButton(TITLES[SET_K_27],
callback_data=SET_K_27),
            InlineKeyboardButton(TITLES[SET_K_37],
callback_data=SET_K_37),
            InlineKeyboardButton(TITLES[SET_K_47],
callback_data=SET_K_47)
        ),
        (
            InlineKeyboardButton(TITLES[SET_V_11],
callback_data=SET_V_11),
            InlineKeyboardButton(TITLES[SET_V_21],
callback_data=SET_V_21),
            InlineKeyboardButton(TITLES[SET_V_31],
callback_data=SET_V_31),
            InlineKeyboardButton(TITLES[SET_V_41],
callback_data=SET_V_41)
        ),
        (
            InlineKeyboardButton(TITLES[BACK_TO_CHOSE_STATUS],
callback_data=BACK_TO_CHOSE_STATUS),
        )
    )
    return InlineKeyboardMarkup(keyboard)

# 3 клавіші
#     1 - змінити групу
#     2 - повернутися до налаштувань

```

					ДП.КН 20.408.03.000 ПЗ	Арк.
						57
Зм.	Арк.	№ докум	Підпис	Дата		

```

#      3 - повернутися до головного меню
def group_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[CHANGE_GROUP],
callback_data=CHANGE_GROUP),
        ),
        (
            InlineKeyboardButton(TITLES[BACK_TO_SETTINGS],
callback_data=BACK_TO_SETTINGS),
        ),
        (
            InlineKeyboardButton(TITLES[BACK_TO_BASE_MENU],
callback_data=BACK_TO_BASE_MENU),
        )
    )
    return InlineKeyboardMarkup(keyboard)

def invalid_teacher_auth():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[TRY_AGAIN],
callback_data=TRY_AGAIN),
        ),
        (
            InlineKeyboardButton(TITLES[BACK_TO_CHOSE_STATUS_FROM_TEACHER_AUTH],
callback_data=BACK_TO_CHOSE_STATUS_FROM_TEACHER_AUTH),
        )
    )
    return InlineKeyboardMarkup(keyboard)

# 3 кнопки:
#      1 - Абітурієнт
#      2 - Студент
#      3 - Вчитель
def start_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[ENTRANT],
callback_data=ENTRANT),
            InlineKeyboardButton(TITLES[STUDENT],
callback_data=STUDENT)
        ),
        (
            InlineKeyboardButton(TITLES[TEACHER],
callback_data=TEACHER),
        )
    )
    return InlineKeyboardMarkup(keyboard)

```

					ДП.КН 20.408.03.000 ПЗ	Арк.
Зм.	Арк.	№ докум	Підпис	Дата		58

```

# 1 - Група (зміна налаштувань групи)
# 2 - Опис профілю
# 3 - Телефон
# 4 - Повернутись до головного меню
def settings_keyboard():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[GET_GROUP_INFO],
callback_data=GET_GROUP_INFO),
        ),
        (
            InlineKeyboardButton(TITLES[DESC_PROFILE],
callback_data=DESC_PROFILE),
        ),
        (
            InlineKeyboardButton(TITLES[PHONE_NUMBER],
callback_data=PHONE_NUMBER),
        ),
        (
            InlineKeyboardButton(TITLES[BACK_TO_BASE_MENU],
callback_data=BACK_TO_BASE_MENU),
        )
    )
    return InlineKeyboardMarkup(keyboard)

# 1 - Налаштування
# 2 - Одногорупники
# 3 - Розклад
def base_keyboard():
    # Стандартна клавіатура

    keyboard = (
        (
            InlineKeyboardButton(TITLES[SETTINGS],
callback_data=SETTINGS),
            InlineKeyboardButton(TITLES[GET_GROUP_ACCOUNTS],
callback_data=GET_GROUP_ACCOUNTS),
        ),
        (
            InlineKeyboardButton(TITLES[GET_TIMETABLE],
callback_data=GET_TIMETABLE),
        )
    )
    return InlineKeyboardMarkup(keyboard)

def back_to_teacher_menu():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[BACK_TO_TEACHER_MENU],
callback_data=BACK_TO_TEACHER_MENU),
        ),
    )

```

```

    )
    return InlineKeyboardMarkup(keyboard)

def back_to_chose_group():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[BACK_TO_CHOSE_GROUPS],
callback_data=BACK_TO_CHOSE_GROUPS),
        ),
    )
    return InlineKeyboardMarkup(keyboard)

def teacher_keyboard(messaging_status):
    keyboard = (
        (
            InlineKeyboardButton(TITLES[SEND_TO_ENTRANTS],
callback_data=SEND_TO_ENTRANTS),
        ),
        (
            InlineKeyboardButton(TITLES[SEND_TO_GROUPS],
callback_data=SEND_TO_GROUPS),
        ),
        (
            InlineKeyboardButton(TITLES[ACCEPT],
callback_data=ACCEPT),
        )
    ) if messaging_status else (
        (
            InlineKeyboardButton(TITLES[SEND_TO_ENTRANTS],
callback_data=SEND_TO_ENTRANTS),
        ),
        (
            InlineKeyboardButton(TITLES[SEND_TO_GROUPS],
callback_data=SEND_TO_GROUPS),
        ),
        (
            InlineKeyboardButton(TITLES[REJECT],
callback_data=REJECT),
        )
    )
    return InlineKeyboardMarkup(keyboard)

def send_message_to_reception():
    keyboard = (
        (
            InlineKeyboardButton(TITLES[SEND_MESSAGE],
callback_data=SEND_MESSAGE),
            InlineKeyboardButton(TITLES[CANCEL_SENDING],
callback_data=CANCEL_SENDING)
        ),
    )
    return InlineKeyboardMarkup(keyboard)

```

РЕЦЕНЗІЯ
на дипломний проєкт
студентки відділення комп'ютерних та видавничих технологій
Галицького коледжу імені В'ячеслава Чорновола

студента IV курсу групи К-47

_____ Кузьменка Кирила Олександровича _____
(прізвище та ініціали)

Спеціальність 122 «Комп'ютерні науки та інформаційні технології»

Обсяг дипломного проєкту: 60 стор.

Кількість сторінок записки: 51 стор.

Тема: "Telegram бот для приймальної комісії Галицького коледжу імені
В'ячеслава Чорновола"

1. Актуальність теми: _____ зміст дипломного проєкту відповідає назві та поставленому завданню. Розкрито актуальність теми _____


2. Практична або теоретична цінність опрацьованих питань: _____
_____ у роботі показана практична значущість. Автор мотивує своє рішення щодо вибраної платформи розробки та реалізації інтерфейсної частини за допомогою мови Python та бібліотеки python-telegram-bot. Дана робота свідчить, що студент володіє практичними навичками сучасних підходів реалізації вказаних систем. Автор мотивує своє рішення щодо вибраної платформи _____

3. Недоліки роботи: _____ до недоліків можна віднести слабе посилання на джерела використаної літератури, мала кількість опрацьованих джерел посилання, в тексті зустрічаються неточності _____

4. Загальний висновок _____ дипломний проєкт виконаний згідно вимог і заслуговує на позитивну оцінку, а студент – присвоєння кваліфікації "техніка-програміста" _____

Рецензент _____ Кузик В.М., викладач комп'ютерних дисциплін ЦК Інформатики та комп'ютерних дисциплін _____

«25» червня 2020р.



(підпис)

ВІДГУК
на дипломний проект
студента відділення комп'ютерних та видавничих технологій
Галицького коледжу імені В'ячеслава Чорновола

студента IV курсу групи К-47

Кузьменко Керем
(прізвище та ініціали)

Спеціальність 122 „Комп'ютерні науки та інформаційні технології”

Керівник ДП Чубов О.О.

Тема: Телеграм - bot призначений ко
ллеґії Галицького коледжу імені
В'ячеслава Чорновола.

1. Загальна характеристика студента проявив себе грамот-
ним, кваліфікованим фахівцем, який зміг
на чистому використанні сервісів
сучасних інформаційних технологій.

2. Практична або теоретична цінність опрацьованих
питань тема є актуальною та цінною,
особливо в першій частині призначе-
ної коллеґії Галицького коледжу
імені В'ячеслава Чорновола.

3. Недоліки роботи _____

4. Загальний висновок

кваліфікаційна робота відповідає
вимогам поставленим вимогам, проте
є на поздовжній шкалі, рекомендується
до захисту.

Керівник дипломного проекту Чубов
(прізвище та ініціали)

Ім'я користувача:
Наталя Кульчинська

Дата перевірки:
23.06.2020 16:43:07 EEST

Дата звіту:
16.02.2021 10:08:28 EET

ID перевірки:
1004191087

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100004382

Назва документа: ДП_Кузьменко_K47_перевірка

Кількість сторінок: 51 Кількість слів: 6450 Кількість символів: 47176 Розмір файлу: 2.60 MB ID файлу: 1004203049

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

5.8% Схожість

Найбільша схожість: 1.58% з Інтернет-джерелом (<https://uteka.ua/ua/publication/news-14-ezhednevnyj-buxgalterskij-ob>

5.1% Джерела з Інтернету

55

Сторінка 53

1.83% Джерела з Бібліотеки

2

Сторінка 53

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

3.18% Вилучень

Деякі джерела вилучено автоматично (фільтри вилучення: кількість знайдених слів є меншою за 8 слів та 0%)

0.71% Вилучення з Інтернету

4

Сторінка 54

3.18% Вилученого тексту з Бібліотеки

18

Сторінка 54

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

11

Підозріле форматування

15
сторінок