

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіoeлектроніки,
Факультет комп'ютерних наук і технологій
(повне найменування інституту, назва факультету)

Кафедра програмних засобів
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ БОТА-
КОНСУЛЬТАНТА ДЛЯ ПЕРСОНАЛЬНОГО ПІДБОРУ КОМПЛЕКСУ
ВІТАМІНІВ
RESEARCH AND SOFTWARE IMPLEMENTATION OF THE BOT
CONSULTANT FOR INDIVIDUAL SELECTION OF A VITAMIN COMPLEX

Виконав: студент(ка) 2 курсу, групи КНТ-219м

Спеціальності 122 Комп'ютерні науки

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Системи штучного інтелекту

Фоменко Ю.О.

(прізвище та ініціали)

Керівник Зеленьова І.Я.

(прізвище та ініціали)

Рецензент Терещенко Е.В.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Інститут, факультет ПРЕ, ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти магістр
 Спеціальність 122 Комп'ютерні науки
(код і найменування)
 Освітня програма (спеціалізація) Системи штучного інтелекту
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
С.О. Субботін
 “ ” 2020 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

Фоменка Юрія Олександровича

(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна реалізація бота-консультанта для персонального підбору комплексу вітамінів. Research and Software Implementation of the Bot Consultant for Individual Selection of a Vitamin Complex.

керівник проєкту (роботи) Зеленьова Ірина Яківна, к.т.н., доцент,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “03” листопада 2020 року № 301

2. Строк подання студентом проєкту (роботи) 5 грудня 2020 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Огляд аналогів та аналіз основних засобів створення чат-ботів. 2. Розробка основних рішень щодо реалізації компонентів системи. 3. Розробка програмного комплексу чат-бота. 4. Результати тестування та застосування чат-боту. 5. Економіко-організаційна частина. 6. Охорона праці та безпека в надзвичайних ситуаціях.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-5 Основна частина	Зеленьова І.Я., к.т.н. доцент		
Економіко-організаційна частина	Пожуєва Т.О., професор		
Охорона праці та безпека в надзвичайних ситуаціях	Коробко О.В., ст. викладач		
Нормоконтролер	Камінська Ж.К., асистент		

7. Дата видачі завдання “04” вересня 2020 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	2-3 тижні	Розділ 1
3	Розробка та удосконалення методів, моделей й алгоритмів вирішення задачі.	4-5 тижні	Розділ 2
4	Розробка архітектури програми.	6 тиждень	Розділ 2
5	Розробка програми.	7-8 тижні	Розділ 3
6	Тестування та експериментальне дослідження програмного забезпечення.	9 тиждень	Розділ 4
7	Розробка економіко-організаційної частини.	10 тиждень	Розділ 5
8	Розробка заходів з охорони праці та безпеки в надзвичайних ситуаціях.	11 тиждень	Розділ 6
9	Оформлення пояснювальної записки та документів до неї.	12-13 тиждень	Додатки
10	Нормоконтроль та рецензування.	14-15 тижні	
11	Захист роботи.	16 тиждень	

Студент _____ Фоменко Ю.О.
(підпис) (прізвище та ініціали)

Керівник проєкту (роботи) _____ Зеленьова І.Я.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
137 с., 15 рис., 13 табл., 4 додатки, 43 джерел.

ТЕЛЕГРАМ-БОТ, ОНЛАЙН, КОРИСТУВАЧ, ІНТЕРФЕЙС,
МЕСЕНДЖЕРИ, СИСТЕМИ МИТТЄВОГО ОБМІНУ ПОВІДОМЛЕННЯМИ,
ЧАТ-БОТ, PYTHON, SQLITE.

Об'єкт дослідження – засоби обміну онлайн повідомленнями.

Предмет дослідження – можливості залучення уваги користувачів соціальної мережі до проєкту за допомогою програмного продукту автоматизації взаємодії користувачів.

Мета дипломної роботи – розробка програмного забезпечення «Health Care», що дозволяє автоматизувати завдання, пов'язані з опитуванням користувача щодо його стану здоров'я, способу життя, харчування тощо, та на основі відповідей обирає комплекс вітамінів чи певних рекомендацій що стосовно здоров'я користувача.

Наукова новизна полягає в тому, що проаналізовано велику кількість програмних засобів для реалізації проєкту, продемонстровано велика кількість застосувань чат-ботів в бізнесі, а також розроблений унікальний чат-бот, безпосередньо для громадян щодо зміцнення здоров'я, профілактики захворювань та подолання їх наслідків, щоб люди не залишалися наодинці зі своїми недугами.

Практична значимість полягає в тому, що даний чат-бот можна застосувати в будь-якій медичній компанії, в якій є бізнес-процес обробки звернень клієнтів в чат.

В результаті було розроблено продукт, повністю відповідний до технічного завдання та сучасних потреб до подібних інтернет ботів. Був досягнутий симбіоз зручного, красивого інтерфейсу та інтуїтивно зрозумілого необхідного функціоналу.

ABSTRACT

Explanatory note to the master's thesis: 137 p., 15 figures, 13 tables, 4 appendices, 43 sources.

TELEGRAM BOT, ONLINE, USER, INTERFACE, MESSAGERS, INSTANT MESSAGE EXCHANGE SYSTEMS, CHAT BOT, PYTHON, SQLITE.

Object of study - the means of online messaging.

Subject of the research is the possibility of attracting the attention of social network users to the project with the help of the software product of automation of user interaction.

The purpose of the thesis is to develop software "Health Care", which allows you to automate tasks related to the survey of the user about his health, lifestyle, diet, etc., and based on the answers selects a set of vitamins or certain recommendations for health. I am a user.

The scientific novelty is that a large number of software tools for the project have been analyzed, a large number of applications of chatbots in business have been demonstrated, and a unique chatbot has been developed directly for citizens to promote health, prevent diseases and overcome their consequences. so that people are not left alone with their illnesses.

The practical significance is that this chatbot can be used in any medical company that has a business process for processing customer requests to chat.

As a result, a product was developed that fully meets the technical task and modern needs for such chatbots. A symbiosis of a convenient, beautiful interface and intuitively necessary functionality was achieved.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ.....	8
ВСТУП	9
1 ОГЛЯД АНАЛОГІВ ТА АНАЛІЗ ОСНОВНИХ ЗАСОБІВ СТВОРЕННЯ ЧАТ-БОТІВ.....	12
1.1 Дослідження чат-ботів	12
1.2 Огляд і аналіз сучасних бот-платформ.....	19
1.3 Аналіз теми «зв'язок здоров'я та вітамінів»	24
1.3.1 Охоплення напрямів пріоритетів.....	27
1.4 Огляд складності створення чат-бота.....	34
1.5 Аналіз недоліків аналогів чат-ботів.....	37
1.6 Постановка задачі дослідження	38
2 РОЗРОБКА ОСНОВНИХ РІШЕНЬ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ.....	40
2.1 Інструменти використані для розробки продукту.....	40
2.2 Розробка способу організації даних та структура таблиць	48
2.3 Розробка проєкту в обраному середовищі	49
2.4 Висновки до розділу 2	51
3 РОЗРОБКА ПРОГРАМНОГО КОМПЛЕКСУ ЧАТ-БОТА	52
3.1 Основні розроблені алгоритми.....	52
3.2 Процес створення бота та огляд Telegram Bot API	53
3.3 Модуль БД	55
3.4 Модуль обробки нових користувачів	57
3.5 Висновки до розділу	58
4 РЕЗУЛЬТАТИ ТЕСТУВАННЯ ТА ЗАСТОСУВАННЯ ЧАТ-БОТУ.....	60
5 ЕКОНОМІКО-ОРГАНІЗАЦІЙНА ЧАСТИНА	66
5.1 Альтернативи ринкової поведінки.....	68
5.2 Базова стратегія розвитку	70

5.3 SWOT-аналіз та визначення пріоритетних груп зацікавлених осіб	71
5.4 Визначення команди, та життєвого циклу проєкту	72
5.5 Визначення витрат на розробку	73
5.5.1 Розрахунок основної та додаткової заробітної плати	73
5.5.2 Визначення витрат на матеріали	75
5.5.3 Витрати на обладнання.....	76
5.5.4 Накладні витрати.....	78
5.6 Розрахунок економічної ефективності програмного продукту	79
5.7 Оцінка економічної ефективності.....	81
6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ	82
6.1 Аналіз потенційних небезпек	82
6.2 Заходи з забезпечення безпеки	83
6.3 Заходи з виробничої санітарії та гігієни праці	86
6.4.1 Заходи з пожежної безпеки.....	88
6.4.2 Заходи з цивільного захисту	89
ВИСНОВКИ.....	92
ПЕРЕЛІК ПОСИЛАНЬ.....	94
ДОДАТОК А - Технічне завдання	98
ДОДАТОК Б - Опис програми.....	101
ДОДАТОК В - Текст програми	104
ДОДАТОК Г - Слайди презентації.....	127

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Бот – Автоматизована програма на сервері

БД – База даних.

Графічний інтерфейс – тип інтерфейсу, який дозволяє користувачам взаємодіяти з електронними пристроями через графічні зображення та візуальні вказівки.

Онлайн – суб'єкт який знаходиться в мережі, здійснює дії в інтернеті в режимі реального часу.

ОС – Операційна система;

ПЗ – Програмне забезпечення;

ППІ – Прикладний програмний інтерфейс;

ПК – Програмний комплекс;

СУБД – Система управління базою даних;

СМОП – Система миттєвого обміну повідомленнями;

ЧБ – Чат-бот;

API – Application Programming Interface;

CRUD – Create, Read, Update, Delete;

GUI – Graphical User Interface;

HTML – HyperText Markup Language;

HTTP – HyperText Transfer Protocol.

ВСТУП

У сучасних умовах інтернет є універсальним середовищем для спілкування, розваг та навчання. Сьогодні у світі існує дуже багато засобів комунікації, і багато з них так чи інакше пов'язані з сучасними технічними можливостями, представленими, зокрема, глобальною комп'ютерною мережею. Окрім надання різноманітної та корисної інформації для користувачів, інтернет є основною формою віртуального спілкування. Відносини з друзями та родичами, стосунки з колегами, нові знайомства - все це є важливим елементом повсякденного життя сучасної людини, а для користувача вибір найбільш зручних способів спілкування в Інтернеті досить великий.

Тоді чому боти актуальні і навіть необхідні в наш час? По-перше, це дуже вигідно для бізнесу. Завдяки роботам ви можете оптимізувати витрати та зменшити навантаження на працівників, їх здатність виконувати понад вісімдесят відсотків завдань у взаємодії між організацією та замовником або всередині компанії для підвищення якості послуг. У міру того, як роботи стають розумнішими, завдання можуть ускладнюватися, і боти мають можливість вчитися на більш складній роботі.

У магістерській роботі описані загальні принципи створення бота для месенджера, його можлива архітектура та можливість введення функціональних можливостей нейронної мережі для поліпшення взаємодії з користувачем при використанні бота.

Особливо це стосується українського ринку, оскільки компанії лише починають освоювати цю технологію, тому попит на професіоналів для розвитку таких послуг швидко зростає.

Програма буде виконувати наступні функції:

- опитування користувача;
- запис даних до бази;
- створення рекомендації за характеристиками користувача.

Задачі дослідження: описати ключові принципи створення ботів для месенджерів. Розглянути можливості застосування засобів нейронних мереж для покращення досвіду користувача при використанні бота. На основі описаних принципів розробити сервіс для платформи Телеграм.

Метою даної магістерської роботи є розробка програмного забезпечення «Health Care», що дозволяє автоматизувати завдання, пов'язані з опитуванням користувача щодо його стану здоров'я, способу життя, харчування тощо, та на основі відповідей обирає комплекс вітамінів чи певних рекомендацій що стосовно здоров'я користувача.

Об'єкт дослідження – засоби обміну он-лайн повідомленнями.

Предмет дослідження – можливості залучення уваги користувачів соціальної мережі до проєкту за допомогою програмного продукту автоматизації взаємодії користувачів.

Таким чином, рішення розробки чат-боту було продиктовано бажанням нормалізувати, покращити стан здоров'я людей, який буде надавати практичні рекомендації громадянам по зміцненню здоров'я, профілактиці захворювань і подолання їх наслідків.

Для досягнення певної мети в атестаційній роботі магістра рекомендуємо вирішити такі завдання:

- дослідити сучасних підходів до розробки ботів;
- дослідити наявні проблеми з чат-ботами;
- визначення програмної платформи, а саме каркасу та інтерфейсу прикладних рішень та бібліотек API з низкою доступних інструментів: функцій, класів, структур, які підходять для сучасної розробки ботів;
- дослідити методи машинного навчання для розробки роботів та їх взаємодії з користувачем;
- розробка прикладної програми;
- впровадження розробленого програмного забезпечення.

У роботі було досліджено та описано загальні принципи створення бота для месенджера, на прикладі декількох ботів, код яких знаходяться у

загальному доступі. За результатами дослідження було виділено можливу архітектуру сервісу. Також було досліджено можливість введення функціоналу бази даних на прикладі застосування SQLite для покращення виведення інформації ботом.

Практична значимість розробленого чат-боту - це те що проєкт може бути застосований у медичній сфері, відповідаючи на питання людини, без присутності медичного спеціаліста.

1 ОГЛЯД АНАЛОГІВ ТА АНАЛІЗ ОСНОВНИХ ЗАСОБІВ СТВОРЕННЯ ЧАТ-БОТІВ

1.1 Дослідження чат-ботів

Принципи та технології машинного навчання, що лежать в основі голосових асистентів, більше не є прерогативою компаній. За їх допомогою класичні роботи-чати стали набагато розумнішими, досить надійними для комерційного використання та популярними серед користувачів.

Говорячи про актуальність, нам потрібно лише поглянути на статистику за останні 5 років (рисунок 1.1) [1].

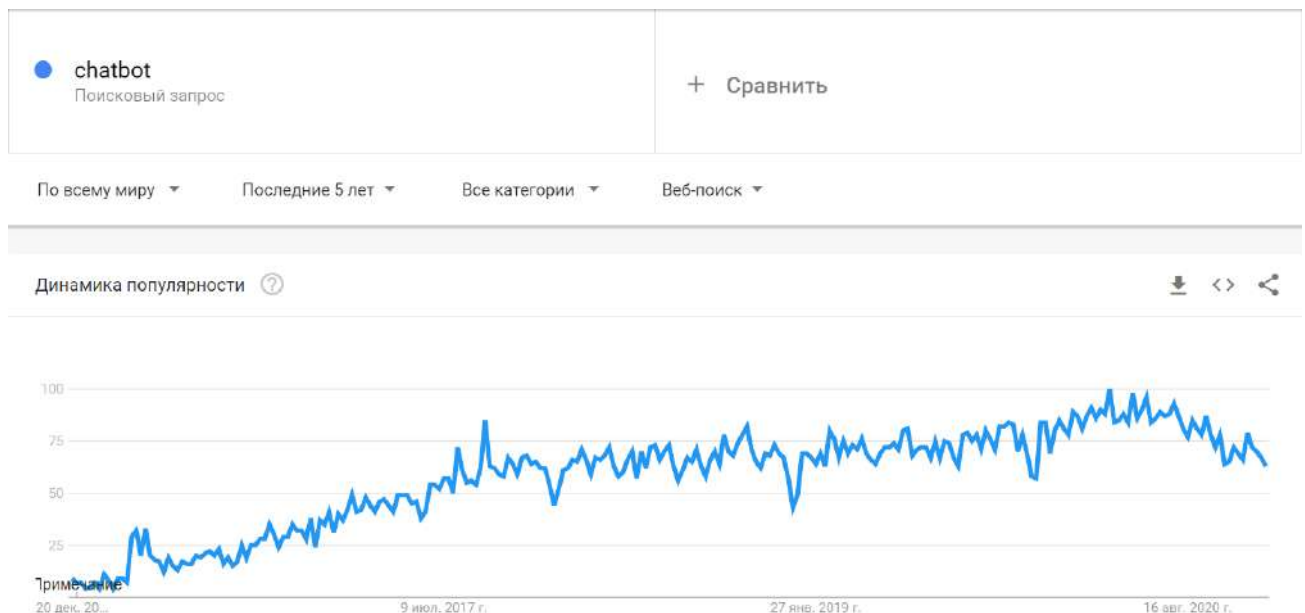


Рисунок 1.1 – Динаміка популярності чат-ботів на сьогодні [1]

Сьогодні боти стають актуальним стандартом інтерфейсу для взаємодії з програмними послугами. Це пов'язано з широким використанням платформ обміну повідомленнями (таких як Facebook Messenger для користувачів соціальних мереж та Slack для розробників) і частково завдяки розробці багатьох пристроїв, що підтримуються ботами. Ще одним рушієм є широке впровадження великих даних та алгоритмів машинного навчання: боти мають

зручний інтерфейс для взаємодії з системами, які надають відповіді на основі результатів аналізу величезної кількості інформації.

Великі програмні компанії визнають зручність робіт в інтеграції послуг, каналів зв'язку та асоціацій користувачів. Наприклад, Facebook хоче поступово замінити всі програми ботами платформи обміну повідомленнями Messenger, а Microsoft каже, що операційна система майбутнього - це "діалог як платформа".

Про зростання популярності ботів можна судити за розвитком таких сервісів, як Alexa, Siri, IBM Watson та Google Now. На платформах також є багато роботів, з якими розробники програмного забезпечення спілкуються зі своїми колегами та такими службами, як Slack, Microsoft Teams та HipChat.

Згідно з глобальним опитуванням statista 2017 року, 34% респондентів сказали, що вони могли б краще відповідати на питання ШІ за допомогою чат-бота або віртуального помічника при використанні електронної комерції (рисунок 1.2) [1].

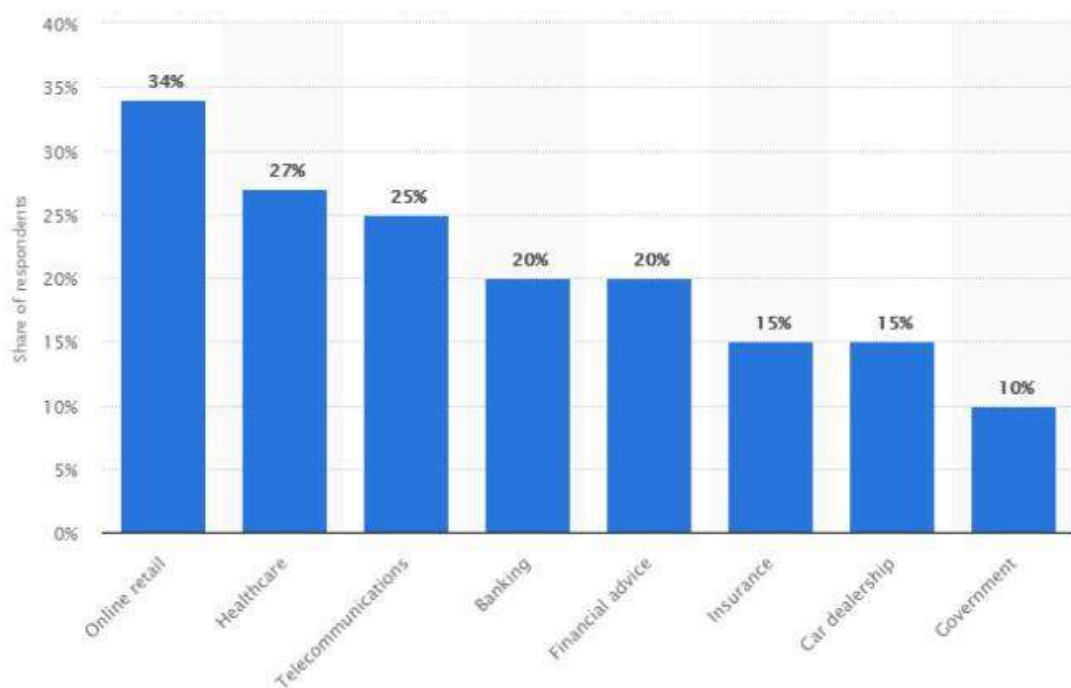


Рисунок 1.2 – Данні опитування Statista на 2017 рік [1]

У наш час боти активно створюються для масових користувачів, багато часу приділяє програмам обміну повідомленнями. Такі боти часто стають альтернативою традиційним мобільним додаткам.

Боти стають все більш поширеними - ми взаємодіємо з ними в машині, вдома, на роботі та за її межами. Крім того, вони відіграють складну і все більш помітну роль у проєктах з розробки програмного забезпечення. Досвід, накопичений дотепер у використанні ботів, дозволяє оцінити як їх реальні практичні переваги, так і можливі недоліки. Дотримуйтесь цих принципів під час створення та використання пошукових систем.

Прості боти можна створювати з нуля і розміщувати на власних ресурсах, але коли справа доходить до спрощення розробки та розповсюдження, вони використовують зовнішні платформи. Кількість інструментів для розробки роботів швидко зростає, але є різниця між платформами для створення програм-роботів та платформами розповсюдження, якими вони керують.

Такі компанії, як Microsoft, Facebook, Telegram та інші постачальники, надають комплексні інструменти для розробки та розповсюдження ботів. Інші компанії пропонують спеціалізовані ресурси для конкретних завдань створення та розповсюдження.

Платформи для розробки ботів надають різноманітні програмні засоби, фреймворки, інструменти, інтерфейси програмування та додаткові послуги (обробка природної мови, послуги пошуку та аналізу зображень тощо). Вони можуть бути націлені на певну платформу розповсюдження або призначені для створення платформ, які можуть бути розгорнуті на багатьох платформах, включаючи Microsoft Bot Framework, Botkit та Pandorabots. Надані послуги варіюються від документації та шаблонів коду до інтерфейсів, що не програмують (наприклад, Chatfuel). Кілька популярних платформ для розробників створили активні спільноти розробників, члени яких можуть спілкуватися у відповідних онлайн-групах, обмінюватися досвідом та

посиланнями на тематичні публікації, отримувати поради та відповіді на запитання та брати участь у дискусіях. Деякі із цих спільнот, такі як Botmaker Slack Group та Chatbot Magazine, були в центрі дискусій з широкого кола тем, пов'язаних з робототехнікою [2].

Незважаючи на величезну кількість існуючих чат-ботів, вони поділяються на дві широкі категорії, залежно від технології, що використовується для інтелектуального діалогу: обробка текстів на природній мові за допомогою NLP (Natural Language Processing) та програм на основі міркувань. CBR, що ґрунтується на справах (міркування на основі справ).

Чат-боти, що працюють на технологіях обробки текстів на природній мові, починають аналіз користувацьких реплік з аналізу, в результаті чого речення діляться на нетермінальні та кінцеві символи відомої програми. Чат-боти, що використовують технологію NLP, не мають бази даних з чітко визначеними шаблонами відповідей, відповідь на введення користувачем терміна щоразу синтезується на основі правил використаної граматики, структури введеного раніше тексту користувача та ключових слів, які він містить. Деякі з цих роботів не мають оригінальної граматики, але використовують індуктивний метод виведення граматики, що дозволяє їм навчитися і “налаштуватися” на свої манери під час діалогу з користувачем [3].

Боти телеграм - це програмні засоби, що реалізують певні функції в телеграфному чаті, тобто отримують повідомлення та команди від користувача і відповідають у вигляді обробленої інформації, тобто інструменти, які можна формалізувати для команд та обробити, прикладом такої функціональності є чат. Палиця компанії для надання інформації про замовлення, паличка для моніторингу або реагування на зміни даних, інструкції до технічного інструменту, який автоматично відповідає на популярні запитання про його використання, пошук інформації про підтримку через мережу, пошук інформації та надання її для завантаження, обробка зображень, виклик таксі та порівняння цін, отримання та обробка платежів.

Список доступних послуг обмежений лише інтерфейсом чату та його частотою оновлення.

Ряд можливостей ботів приведено нижче:

- можливість обробки платежів користувачів;
- створення інструментів;
- інтеграція з зовнішніми сервісами, наприклад поштовою скринькою;
- отримувати відредаговані новини або оновлення;
- створення соціальних сервісів;
- одно-користувацьких та багато-користувацьких ігор;
- інші віртуальні можливості.

Кожен бот має атрибут `username`, за допомогою якого ви можете знайти його в пошуку. Існує два типи ботів, роботів-чатів та вбудованих роботів, які працюють з ними в групах чату або каналах. Ви можете використовувати ботів і керувати ними із зарезервованими командами, текстовими повідомленнями або надсиланням повідомлення відповідного типу. Звичайно включати команди в опис в телеграмі бота, і перший раз користувач запускає команду `"/start"`.

Тобто, Telegram - це прогресивний додаток, який може створювати власні бот-програми на своїй платформі. Телеграм-бот здатний реалізувати більшість функцій Telegram. Це дозволяє створювати повноцінні чат-боти на своїй платформі, що вже включає основні взаємодії клієнт-сервер, оптимізовані клієнти в популярних операційних системах та потужні функції створення чат-ботів.

В даний час спостерігається тенденція використання інтерфейсів природної мови на основі технологій чат-ботів для практичних застосувань: у телефонних центрах обслуговування; для автоматичної відповіді на запити електронною поштою; доступ до баз даних та випуск запитуваної інформації віддаленим користувачам; надавати послуги телефонного банкінгу тощо. Однак використання цих програм у системах електронної комерції

залишається вкрай обмеженим, і на цьому етапі ці програми використовуються лише як прототипи, які не мають широкого поширення.

Що до видів ботів, то на даний час є не офіційна класифікація:

- бот-автоматизатор;
- бот-помічник;
- вбудований бот;
- бот який допомагає оформити замовлення;
- бот-збирач лідів.

Бот-автоматизатор - виконує кілька рутинних дій для користувача в автоматичному режимі, економлячи йому час. Корисно для таких завдань, як відстеження пакунків або графіків доставки, замовлення таксі, автоматична розсилка електронних листів за результатами CRM-подій. [4-8].

Бот-помічник - здатний розпізнавати запити клієнтів та надавати відповіді на них, не залучаючи до цього особу. Коли він відпочиває на стелі своїх знань, він підключає клієнта до менеджера. З іншого боку, менеджер отримує повну історію управління - з персональними даними, питаннями та історією спілкування, озвученими клієнтом. Цей бот "розвантажує" менеджерів, позбавляючись від рутини. Це самоучка, тому з часом ви можете виконувати дедалі складніші завдання. Наприклад, служба підтримки може відповісти на найпопулярніші запитання [4-8].

Вбудований бот – бот з елементами ІІ, йому можна «згодувати» великий обсяг даних, в якому він буде шукати інформацію краще і швидше людини. Корисний там, де потрібно орієнтуватися у величезних обсягах даних, наприклад, в документах страхових або юридичних фірм [4-5].

Бот, який допомагає оформити замовлення – цей бот може відшукати потрібний товар і оформити замовлення через інтерфейс месенджера швидше, ніж це можна було б зробити через сайт. Економить час користувача за рахунок того, що виконує максимальну кількість завдань через інтерфейс чату, і перекидає в браузер для завершення оформлення замовлення (авторизації або оплати) автоматично, позбавляючи від необхідності щось шукати вручну [4-8].

Бот-збирач лідів – збирає дані про клієнтів, зацікавлених в замовленні, уточнює у них персональні дані та інформацію про цікавить товар / послугу. Також береже менеджерів компанії від рутинної роботи. Корисний всюди, де потрібно збирати досить однотипні дані в користувачів [4-8].

Боти часто використовуються в робочих групах для автоматизації завдань, які зазвичай вимагають втручання людини. Однак повна відмова від взаємодії може обмежити творчі здібності та продуктивність людей. Тому ботів найкраще використовувати для полегшення зв'язків та координації роботи. Наприклад, роботи можуть інформувати про хід виконання завдання, сповіщати всю групу, зв'язувати експертів з новачками та слугувати зміцненню довіри та командного духу. Попри цьому неправильне виконання або надмірне використання роботів може призвести до перевантаження інформацією або надмірних перебоїв.

В ідеалі користувач повинен взаємодіяти з роботом комфортно і без запинок. Цього можна досягти, ретельно спланувавши схему розмови, особливо для шукачів діалогових вікон. Зокрема, бот міг повторити введені команди, щоб користувач знав, що він «прослухав», і правильно їх розумів. Боти також повинні мати можливість визначити, чи діалог зайшов у глухий кут, і надати підказки щодо того, чи може взаємодія продовжуватися.

Боти можуть мати елементи графічного інтерфейсу, які прискорюють взаємодію. Забезпечуючи глобальний контроль над ключовими словами, пов'язаними з навігацією («довідка», «назад», «видалити», «перезапустити», «вихід» тощо), ви можете уникнути створення надто «дурного» бота. Такі інструменти, як BotMock, допомагають змодельовати взаємодію користувача зі своїм ботом. Крім того, на багатьох платформах є готові комунікаційні протоколи, яких повинні виконувати боти, наприклад, Facebook має прості поради щодо організації розмов. Роботи в чаті відрізняються залежно від завдань, які потрібно виконати.

Комерційний чат-бот, який працює в певному діалоговому вікні, темі чи зоні відповідальності чат-бота, не потребує технології машинного

навчання. Використовуйте основні технології обробки мови для цього типу роботів-чатів:

- вибір ключових слів або іменованих сутностей;
- сегментація (це поділ на речення);
- стовбур / лематизація.

Технології для створення регулярних граматик для типу вхідної батареї також є шаблонами для відповідної реакції, приклад з AIML:

- масив (змінні) для вивчення або запам'ятовування контексту;
- регулярні вирази;
- рекурсія;
- цикли та багато іншого.

Якщо є статистика, краще створити більш досконалий робот-чат про технологію машинного навчання, а саме метод глибоких нейронних мереж[4-8].

1.2 Огляд і аналіз сучасних бот-платформ

Якщо говорити про платформи, які використовують ботів, то більша частка належить саме месенджерам.

Що таке месенджер? Найчастіше термін месенджер позначає програму або мобільний додаток, в якому ви можете обмінюватися повідомленнями з іншими користувачами. Перші месенджери з'явилися в 1990-х роках, але більшу популярність вони почали набувати лише в епоху пост-ПК. Швидкий розвиток мобільних пристроїв зробив їх доступними для людей з різним рівнем доходу. Зараз майже у кожного є під рукою смартфон. Найпопулярніші месенджери в Україні - WhatsApp, Viber та Telegram. Розробка ботів у месенджерах почалася після того, як Telegram відкрив платформу для створення спеціальних акаунтів, для обробки та відправки повідомлень. В даний час половина створених ботів базується на Telegram. Він простий у використанні, доступний насамперед у месенджері та соціальних мережах. Це

означає, що на мобільних пристроях вони на 1 крок ближче до користувача, ніж мобільний додаток, і на 2 кроки ближче до вебсайту. Для вирішення елементарної проблеми більше не потрібно звертатися до різних джерел даних [9-13].

Використовувати бота на цих платформах досить просто, вам просто потрібно вибрати його з каталогу, перейти за посиланням або шукати його за іменем та шукати за ним. Для запуску бота введіть / запустіть (запустити) або запропонуйте натиснути кнопку «старт» на віртуальній клавіатурі. Програмовані клавіші на віртуальній клавіатурі можна використовувати для роботи, роблячи чат-бота міні-додатком з інтуїтивно зрозумілим інтерфейсом. Потім бот надсилає інформацію про себе, інструкції та список або відображає доступні команди або кнопки на екрані пристрою.

Що стосується безпеки, боти в самому месенджері не можуть зробити нічого поганого, оскільки це, по суті, однакові облікові записи користувачів, лише алгоритмічна обробка. Як і звичайний користувач, нудного бота можна видалити і заблокувати. Важливо, що бот не є ініціатором розмови - ви першими надсилаєте йому запит, тому бот не може заважати вам без вашого дозволу.

Люди проводять більшу частину часу в месенджерах, про що свідчить активна статистика користувачів за липень 2019 року на сайті [statista.com](https://www.statista.com) (рисунок 1.3) [1].

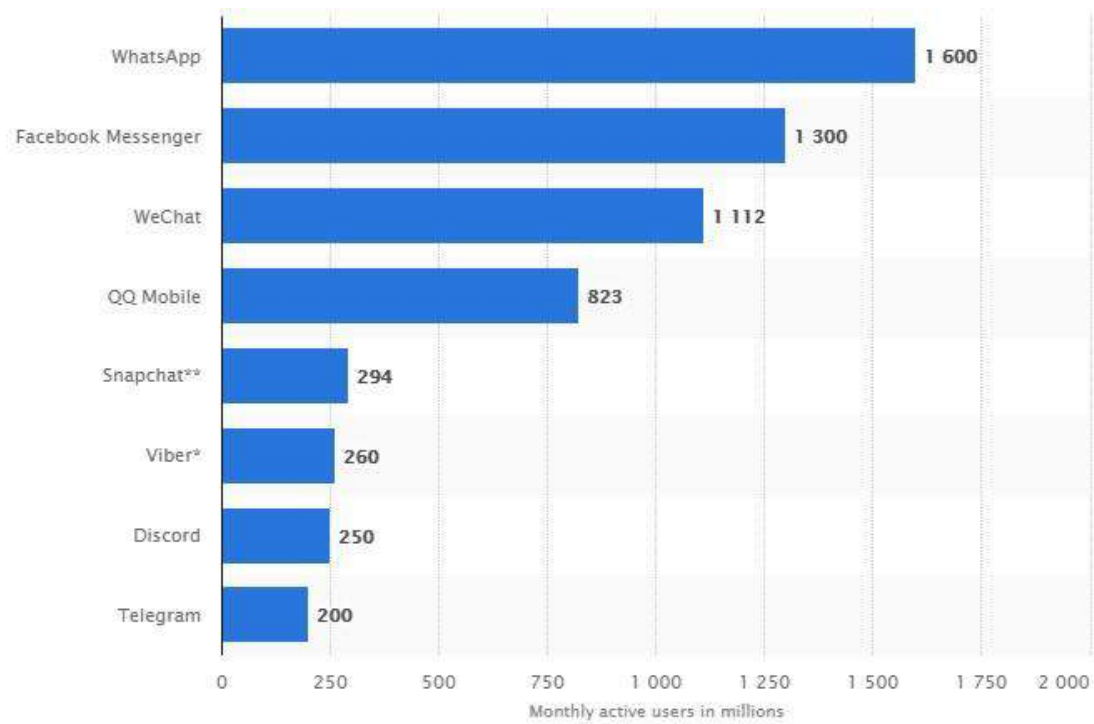


Рисунок 1.3 – Кількість користувачів в месенджерах за місяць [1].

Звичайно, потрібно розуміти, що ці платформи абсолютно різні, але вони мають щось спільне. Наприклад, досить легкий та інтуїтивно зрозумілий інтерфейс. У кожному зі списків кур'єри можуть створювати групові розмови, здійснювати голосові та відеодзвінки та надсилати голосові повідомлення.

Згідно з цією статистикою можна впевнено стверджувати, що ці послуги є важливим елементом у взаємодії користувачів інтернету.

Telegram Messenger має власний протокол шифрування MTProto, який використовується для шифрування пошти користувачів. Розроблено власником компанії. Протокол заснований на поєднанні протоколу обміну ключами Diffie-Hellman RSA-2048, який працює за принципом відповідності ключів, та симетричного алгоритму AES з набором унікальних хеш-функцій для шифрування та зберігання окремих повідомлень користувачів на сервері.

На рисунку 1.4 наведено ілюстрацію схеми шифрування MTProto[13].

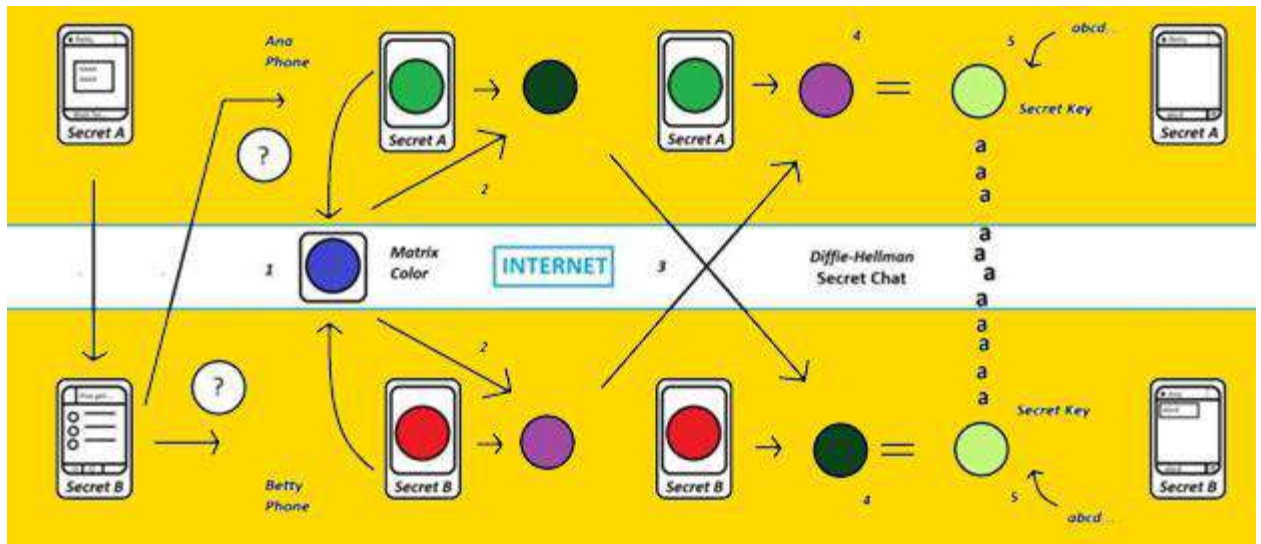


Рис. 1.4. Ілюстрація схеми шифрування MTProto [13]

Функція секретного чату дозволяє зберігати всі повідомлення з таких чатів лише на пристроях користувачів і використовувати алгоритм шифрування AES-256 в режимі IGE (наскрізне мармурове розширення) із наскрізним шифруванням. Різниця між такими чатами полягає в тому, що сервер не декодує такі повідомлення, а лише надсилає їх кінцевим користувачам, а вся історія пошти зберігається лише на їх пристроях, що дозволяє гарантовано видаляти повідомлення після закриття чату та при спробі зробити знімок екрана. можна спілкуватися з іншими пристроями в такому чаті, оскільки повідомлення зберігаються локально, ви також можете встановити таймер для видалення повідомлень або файлів через певний час після їх прочитання або відкриття. Разом це дозволяє створити високий рівень безпеки для вашої програми. Тобто повідомлення, надіслані телеграмою, гарантовано не будуть прочитані іншими [9-13].

Також можливо створювати групи за допомогою телеграм, об'єднувати користувачів у групові чати та спілкуватися з великою кількістю користувачів одночасно, швидко та зручно. Все, що вам потрібно зробити, це ввести назву чату та додати користувачів до групи. Творці чату мають можливість керувати користувачами, обмежуючи доступ членів до можливостей чату. Такі групи

можуть одночасно включати до п'яти тисяч людей, і такі чати є більш зручною альтернативою корпоративним та Skype-чатам. Ви можете зв'язатися з певним користувачем у чаті, ввівши символ "@" та вибравши потрібного користувача. Ви також можете додати посилання на чати та групи з однаковим символом, створити пошукові теги із символом "#" та ввести потрібне слово. Ви можете натиснути лише на ряд повідомлень, у яких його використовувати. Усі користувачі можуть створювати групи. Для кращого управління групою творець може призначати адміністраторів та визначати їх функціональні можливості. Також можна змінити формат тексту, зробивши його жирним, курсивом та одностороннім.

Канали - це групи, що дозволяють запускати медіа-канали, де лише один користувач може надсилати повідомлення, а всі інші є передплатниками цих повідомлень. Існують публічні та приватні канали, які лише невидимі для цього пошукового каналу. Ці канали можуть просувати та монетизувати інформаційний контент. Усі канали каналу можна передавати або копіювати.

У Telegram можна підключити групу до каналу для обговорення публікацій, після підключення внизу каналу можна обговорити або провести анонімне опитування, що дозволяє негайно продовжувати обговорення статті.

Telegram дозволяє зберігати та обмінюватися файлами розміром до 1,5 гігабайт/повідомлення, безкоштовне зберігання у необмеженому сховищі на серверах. Програма використовує систему завантаження файлів після переривання мобільного зв'язку. Деякі файли зберігаються в локальній пам'яті та видаляються, щоб звільнити пам'ять. Також можна зберігати дані у збережених повідомленнях, доступних лише користувачеві. Ви можете зберегти всі необхідні програми в окремому діалоговому вікні. Клієнт-користувач також має вбудований редактор фотографій для редагування зображень із програм та редактор анімації GIF [9-13].

Основними перевагами месенджера є:

- крос-платформеність;
- оптимізація трафіку, при низькій швидкості передачі даних;

- швидкість роботи;
- надійність та безпека;
- відсутність сторонньої реклами, банерів;
- відсутність спаму та неперевіреної інформації;
- різноманітність;
- популярність;
- доступність передачі файлів рядом форматів: RAR, Excel, PDF, MP3, MP4, AVI. З обмеженням по об'єму до 1,5 гігабайт;
- хмарне збереження даних.

Отже, Телеграм – це прогресивний месенджер що набирає популярність та має великий потенціал для створення та розвитку, чому сприяє високий рівень захисту, технологічно опрацьований дизайн, широкий функціональний діапазон та відкритість до сторонніх розробок чому сприяє функціонал телеграм ботів.

1.3 Аналіз теми «зв'язок здоров'я та вітамінів»

Повноцінне харчування визначається не тільки енергетичною цінністю їжі, збалансованістю раціону за білками, жирами та вуглеводами, а й забезпеченістю вітамінами, мікроелементами і мінералами. За даними Всесвітньої організації охорони здоров'я (ВООЗ), стан здоров'я людини на 15% залежить від генетичних особливостей, на стільки ж - від стану медичної служби. А ось лівова частка, 70%, залежить від способу життя і харчування.

Збалансованості харчування і наявності в ньому вітамінів і мікроелементів були присвячені популяційні дослідження, проведені Інститутом харчування РАМН. Результати цих досліджень тривожні: у значної частини населення нашої країни відзначається вкрай недостатнє споживання і все більш наростаючий дефіцит вітамінів (А, групи В, С, Е) і ряду мікроелементів (залізо, цинк, йод). Брак вітамінів групи В у 30-40% українців, бета-каротину - більш ніж у 40%, а дефіцит вітаміну С - у 70-90%[14-17]!

Більшість вітамінів не синтезуються в організмі людини. Тому вони повинні регулярно і в достатній кількості надходити в організм з їжею або у вигляді вітамінно-мінеральних комплексів і харчових добавок.

Захворювання, що виникають в результаті відсутності вітамінів в організмі, отримали назву авітамінозів. Авітаміноз - важкі захворювання, які за відсутності лікування можуть привести до смертельного результату. Кожен авітаміноз може бути попереджений або вилікуваний тільки прийомами відповідного вітаміну. Захворювання, що виникають від недостатнього надходження в організм певних вітамінів, отримали назву гіповітамінозів. Гіповітаміноз розпізнаються важче, ніж авітамінози, так як характер захворювання має менш виражену, стерту картину. Гіповітаміноз знижують працездатність і повертають до інфекційних захворювань. Гіповітаміноз широко поширені в періоди, пов'язані з обмеженим харчуванням (стихійні лиха, війни, неврожаї). Вони часто виявляються в весняні періоди, коли обмежено споживання рослинної їжі, основного джерела багатьох вітамінів. Деякі вітаміни руйнуються при тривалій термічній обробці та консервуванні харчових продуктів, що веде до істотного зниження їх вітамінної цінності. Гіповітаміноз можуть виникати внаслідок гострих або хронічних розладів шлунково-кишкового тракту, що призводять до зниження всмоктування вітамінів в кишечнику. І, нарешті, гіповітаміноз можуть виникати за певних станів людини, пов'язаних з підвищеною потребою організму у вітамінах. Це - періоди активного росту, вагітність, великі фізичні навантаження, серйозні інфекційні захворювання та ін[14-17].

У ряду вітамінів існують антагоністи, що перешкоджають їх всмоктуванню і обміну - антивітаміни. Вони виявлені в ряді харчових продуктів. Так, в яєчному білку міститься авидин - речовина, що пов'язує вітамін Н, а в багатьох сортах сирої риби є фермент тіаміназа, що руйнує вітамін В1. Іноді в лікувальних цілях використовують штучні антивітаміни. Так, похідні кумарину (антивітамін К) перешкоджають згортанню крові.

Більшість вітамінів швидко руйнується в організмі, і тому необхідно постійне їх надходження ззовні. Кількість вітамінів, щоденний прийом яких необхідний для нормального розвитку організму і попередження гіпо- і авітамінозів, називається профілактичною дозою. Більша кількість вітаміну необхідно для лікування вже розвиненого авітамінозу. Ця кількість називається лікувальною дозою.

Деякі люди, припускаючи, що вітаміни «не зашкодять», приймають їх в надмірних кількостях. Стану, при яких спостерігаються передозування вітамінів, називаються гіпервітаміноз. У своїй більшості вітаміни швидко виводяться з організму, але такі вітаміни, як А, В1, D, РР, утримуються в організмі більш тривалий час. Тому використання високих доз вітамінів може привести до передозування - викликати головні болі, порушення травлення, зміни шкіри, слизових, кісток і т. Д. Однак токсичні дози, що ведуть до передозуванні, для цих вітамінів у багато разів перевищують звичайну для них добову потребу[14-17].

Добова потреба людини у вітамінах вимірюється в міліграмах, мікрограмах, все залежить від того, який це вітамін. Наприклад, встановлено, що людині необхідно трохи більше 1 г вітаміну В12. Але дорогоцінні молекули цього вітаміну повинні надходити в організм протягом усього життя!

Навіть цих, абсолютно мікроскопічних, на наш погляд, доз вистачає, щоб забезпечити необхідним кожен зі ста трильйонів клітин нашого тіла. Вчені пояснюють це високою біологічною активністю, якою володіють вітаміни. При цьому повинні враховуватися індивідуальні фізіологічні потреби кожної людини, залежні від статі, віку, характеру і інтенсивності його праці, сезонних факторів.

Всі тварини і рослини потребують майже у всіх відомих вітамінах, і тому рослини, а також деякі тварини мають здатність синтезувати ті чи інші вітаміни. Однак людина і ряд тварин, мабуть, в процесі еволюції втратили цю здібність. Джерелом вітамінів для людини є харчові продукти рослинного і

тваринного походження. Вони надходять в організм або в готовому вигляді, або у формі провітамінів, з яких потім ферментативним шляхом утворюються вітаміни. Деякі вітаміни у людини синтезуються мікробної флорою кишечника [14-17].

1.3.1 Охоплення напрямів пріоритетів

В магістерській роботі було досліджено і використано дванадцять тем, за якими користувач обирає певну проблему чи ціль.

Пріоритет «імунітет» охоплює такі питання:

- "За останні 6 місяців ви приймали антибіотики?", з варіантами відповідей "так" чи "ні";
- "Ви часто хворієте на застуди та ГРВІ?", з варіантами відповідей "так" чи "ні".

Імунній системі постійно доводиться боротися з вірусами і бактеріями, які бажають «проникнути» в організм людини і спровокувати різні захворювання. Щоб підняти імунітет, необхідно дотримуватися режиму дня, займатися спортом, забезпечити збалансоване харчування, більше бувати на свіжому повітрі. Якісної і надійної підживленням є також хороші вітаміни для імунітету. Саме персональні комплекси вітамінів підвищують захисні сили організму, як дорослих, так і дітей, запобігаючи, тим самим, швидку стомлюваність, сонливість, сприйнятливість до вірусних інфекцій, алергічні прояви та інші проблеми[14-17].

Пріоритет «Сон» охоплює такі питання:

- "Чи відчуваєте труднощі при засипанні?", з варіантами відповідей "так" чи "ні";
- "Як ви себе почуваєте після нічного сну?", з варіантами відповідей: "Бадьоро і з запасом сил", "Втомлені і розбиті", "Інший варіант".

Систематичне порушення сну відбувається у 40% населення планети. Порушується він через тривалих перельотів, занадто виснажливого графіка роботи, перевтоми і т.д.

Бадьорість духу і позитивний настрій, без сумніву, це добре. Крім цього, вітаміни для сну дорослим впливають на ряд моментів.

Такі вітаміни для здорового сну сприяють профілактиці діабету, допоможуть уникнути проблем з серцем.

Висипатися - значить бути щасливим. Вчені підраховали, що завдяки нічному відпочинку виробляється гормон щастя. Вітаміни від стресу допомагають виконувати багато завдань кожен день легко, не відчуючи ваги і втоми.

Якісні вітаміни для гарного сну допомагають боротися із зайвою вагою. Мозок повинен підживлюватися енергією, а для цього важливо добре спати.

Безсоння - сама несподівана в будинку «подруга». Не секрет, що вітаміни від безсоння останнім часом відрізняються високим попитом. Тривала робота до ночі або навіть в нічний час виснажує організм, веде до порушення біоритму, страждає якість сну і, як наслідок, погіршується самопочуття.

Якщо не допомогти організму, може трапитися важка депресія, яку лікувати вкрай важко. У зв'язку з цим у ряді випадків лікарі рекомендують приймати вітаміни для сну[14-17].

Пріоритет «Стрес чи настрій» охоплює такі питання:

– "Чи відчуваєте тривожність, невпевненість в собі, пригнічений настрій?", з варіантами відповідей: "Так, часто", "Час від часу", "Ні, Рідко або ніколи";

– "Часто відчуваєте стрес, емоційні навантаження?", з варіантами відповідей: "Ні", "Рідко", "В постійному стресі".

У сучасному світі стреси і депресія - це часто зустрічається явище. Численні проблеми, активний спосіб життя, робота і недостатня кількість відпочинку призводить до поступового виснаження нервової системи. Тому

варто приділяти увагу не тільки своїм способом життя, але ще і додатково приймати спеціальні вітаміни від стресу, які будуть позитивно впливати не тільки на нервову, але і на інші системи.

Коли людина досить довгий час відчуває напругу, то в кінцевому результаті це може обернутися не просто стресом, але і депресією. У такому випадку варто приймати вітаміни від втоми і стресу, які допоможуть вирішити цю проблему. Тому, якщо у вас з'явилися такі симптоми, не варто відкладати на потім можливість допомогти собі[14-17].

Пріоритет «Шкіра» охоплює такі питання:

- "Останнім часом ваша шкіра?", з варіантами відповідей: "Досить зволожена", "Суха";
- "Чи відзначаєте, що ваша шкіра?", з варіантами відповідей: "Еластична", "Втратила пружність".

Шкірний покрив здатний сигналізувати про нестачу важливих компонентів, що проявляється його сухістю, роздратуванням, почервонінням, блідістю, різними висипаннями, передчасним старінням. Отримати необхідні вітаміни для шкіри тільки з їжею не просто. Підходити до такої проблеми необхідно комплексно, переглянувши своє харчування, використовуючи косметичні засоби, а також беручи персональні комплекси вітамінів і мінералів для шкіри[14-17].

Пріоритет «Волосся і нігті» охоплює такі питання:

- "Ваше волосся?", з варіантами відповідей: "Сухе", "Ламке", "Схильне до випадання", "В хорошому стані", "Інша";
- "Ваші нігті?", з варіантами відповідей: "Ламкі і розшаровуються", "Виражена лейконихія (білі плями на поверхні)", "Інша", "В хорошому стані".

Для того, щоб волосся було здоровим, блискучим і шовковистим, дуже важливо, щоб вони отримували різноманітні вітаміни та мікроелементи в достатній кількості. Крім того, необхідно усунути причини, які спровокували погіршення стану локонів. Наприклад, це може бути наслідком стресу або

простудних захворювань. Варто звернути увагу і на вітаміни для шкіри, адже на зовнішній вигляд безпосередньо впливає внутрішній стан організму.

Сучасні вітамінні комплекси містять в собі все необхідне і для шкіри голови, що допомагає волоссю стати набагато красивіше і міцніше. Наприклад, вітаміни від випадіння волосся, містять цинк, кальцій, натуральні масла і вітаміни різних груп. З їх допомогою можна в короткі терміни оживити навіть самі безнадійні локони і зробити їх не тільки зовні привабливими, але ще і здоровими[14-17].

Пріоритет «Фітнес та спорт» охоплює такі питання:

- "Скільки разів на тиждень ви активно займаєтеся фізичним вправами?", з варіантами відповідей: "1-2", "3-4", "5 і більше", "Чи не займаюся";

- "Під час занять ви даєте собі кардіо навантаження?", з варіантами відповідей: "Постійно", "Час від часу", "Ні".

Вітаміни для енергії і бадьорості активізують біохімічні реакції, які заповнюють енергетичний запас в тканинах. Така властивість стимулює нервову, імунну та ендокринну систему. Комплекси для відновлення сил мають певний вектор дії, який залежить від складу. Окремі компоненти безпосередньо збуджують нервову систему, інші стимулюють гіпофіз, підшлункову залозу і кору надниркових залоз.

Знизити рівень керуючої енергії можуть захворювання, стреси, недолік мікро і макроелементів і гіповітаміноз. Вітаміни для сил і енергії здатні ефективно боротися з усіма негативними факторами, підвищувати витривалість і заповнювати запаси необхідних речовин[14-17].

Пріоритет «Енергія» охоплює такі питання:

- "При звичайному графіку дня, ви відчуваєте втому і брак сил вже до обіду?", з варіантами відповідей: "так" чи "ні";

- "Прокинувшись вранці, вам необхідно зарядитися бадьорістю на весь день за допомогою енергетиків? [Наприклад, кава, міцний чай]", з варіантами відповідей: "так" чи "ні";

Щоб розібратися в питанні, які чинники впливають на життєву енергію, потрібно розібратися які в принципі існують види енергії:

- Фізіологічна енергія, джерелом якої є глюкоза. Дає організму необхідний запас сил для всіх рухових функцій: скорочення м'язів, перистальтика і так далі;
- Психічна, або керуюча енергія, яка забезпечує виконання головних функцій головного мозку: розумові процеси, організація і управління метаболізмом, організація роботи систем органів і так далі [36-38].

Кістки і суглоби. Пріоритет «Кістки і суглоби» охоплює такі питання:

- "Чи турбують вас суглоби в даний період життя?", з варіантами відповідей: "так" чи "ні";
- "Чи є у вашій родині спадкові проблеми кістками?", з варіантами відповідей: "так" чи "ні(не знаю)".

Вітаміни для кісток - необхідна складова правильного раціону, особливо в зимовий час. Підживлюючи організм корисними елементами, що містяться у відповідних вітамінних комплексах, ви поліпшите своє самопочуття. Міцні кістки гарантують спокійні заняття спортом, звивши нанівець всілякі зміщення і переломи.

Вітаміни для зміцнення кісток корисні всім, але особливо в них потребує зростаючий організм. Адже кісткова і м'язова маса швидко збільшується і організм повинен отримувати всі елементи, необхідні для цього процесу.

Регулярно вживаючи вітаміни для зміцнення суглобів і кісток, ви однозначно зробите кістки міцними і зможете запобігти переломи або вивихи, а також поліпшите власне самопочуття. Препарати кальцію з вітамінів D3 і K2 допомагають тілу краще засвоювати кальцій, а також мінімізувати наслідки сезонного дефіциту вітаміну D. Вітамін K2, що входить в вітамінні комплекси для кісток, не тільки підтримує скелет, але і зміцнює судини. Це особливо актуальне властивість для професійних спортсменів. Магній, також міститься

в вітамінах для зміцнення кісток і суглобів, допомагає нормалізувати м'язову функцію і мінералізацію скелета[14-17].

Пріоритет «Пам'ять та увага» охоплює такі питання:

- "Забуваєте давно отриману інформацію, маєте проблеми з короткостроковою пам'яттю або труднощі з концентрацією уваги?", з варіантами відповідей: "так" чи "ні";
- "Зараз у вас період високих інтелектуальних навантажень? Вимагають підвищеної концентрації уваги?", з варіантами відповідей: "так", "ні", "планую".

В наш час саме мозок піддається найбільшому навантаженню з усіх внутрішніх органів. Щоденне рішення складних професійних і особистих завдань, на жаль, може викликати проблеми з розумовою активністю і пам'яттю. Лякатися не варто - швидше за все, причина занепаду продуктивності криється в нестачі вітамінів.

Щодо частоти застосування вітамінів для пам'яті і роботи мозку все однозначно - дотримуйтесь інструкцій. Споживайте не більше 2 капсул в день, тільки так можна правильно підтримувати організм і стимулювати природні процеси оздоровлення в тілі. Вітаміни повинні лише обережно доповнювати ваш раціон і виступати в якості приємного бонусу для тіла. Нерозумно заміщати правильне харчування, здоровий сон і активний спосіб життя вітамінами. Все описане має бути присутнім в рівних частках, тоді здоров'я стане вашим кращим другом[14-17].

Пріоритет «Серце і судини» охоплює такі питання:

- "Чи вважаєте ви за необхідне профілактично підтримувати роботу серця і стану судин?", з варіантами відповідей: "так" чи "ні";
- "У вашій сім'ї є спадкові проблеми з серцево-судинною системою?", з варіантами відповідей: "так" чи "ні".

Корисні вітаміни для підтримки серця здатні запобігти розвитку захворювань пов'язаних з ураженням перикарда, міокарда, ендокардам, клапанного апарату серця, судин серця. Вітаміни для зміцнення судин

допомагають попередити атеросклероз, варикозне розширення вен, ішемічну хворобу серця та багато інших захворювань. Комплексні препарати для серця і судин, вітаміни в яких містять в оптимальній добовій дозуванні, допомагають підтримувати необхідний рівень мікро і мікроелементів в організмі для здорової роботи серцево-судинної системи.

Сильне серце і здорові кровоносні судини - запорука довголіття та повноцінного життя. Поки ми молоді, думка стежити за своїм серцем не приходить в голову, хоча передумови для розвитку хвороби серцево-судинної системи починають формуватися задовго до настання старості. Серце людини кожен день справляється з колосальними навантаженнями, а скажений ритм життя, стресові стани і мізерне харчування призводить до ослаблення кровоносної системи, її заповнення шлаками і отруйними речовинами. Вітаміни для серця в комплексі з вітамінами від стресу допоможуть компенсувати щоденний шкоду, що наноситься кровоносній системі, значно продовжити життя і поліпшити її якість[14-17].

Пріоритет «Корекція ваги» охоплює такі питання:

- "Найближчим часом ви хочете?", з варіантами відповідей: "Скинути вагу", "Набрати вагу", "Не змінювати вагу";
- "У своєму раціоні ви віддасте перевагу", з варіантами відповідей: "Борошняні вироби", "Білкова їжа", "Рослинну їжу", "Різноманітну".

Будь-які дієти, спрямовані на зниження ваги, часто є стресовими для організму, провокуючи авітаміноз. Таку ж дію надають виснажливі тренування. У підсумку це призводить до зниження працездатності, поганого самопочуття, частої зміни настрою, погіршення сну, виникнення проблем зі шкірою і волоссям. Ефективно боротися з подібними проявами допомагають хороші вітаміни для схуднення[14-17].

Основні причини, які провокують порушення метаболізму, полягають у:

- неправильному раціоні харчування;
- відсутності фізичної активності;

- спадкових факторів;
- частих стресових ситуаціях;
- нестачі калорій, харчових зриви;
- збої в роботі щитовидної залози.

Пріоритет «Травлення» охоплює таке питання:

- "Чи є у вас проблеми з шлунково-кишкового тракту?", з варіантами відповідей: "Дисбактеріоз", "Здуття", "Тяжкість", Нема проблем".

Вітаміни для печінки, жовчного міхура, шлунка, кишечника допомагають впоратися з такими потужними факторами впливу, як зловживання алкоголем, неправильне харчування, вірусні атаки. Вітаміни для шлунково-кишкового тракту здатні мінімізувати вплив інших сильнодіючих лікарських засобів на всіх адміністративних органів системи.

Завдати шкоди функціонуванню шлунково-кишкового тракту може надмірна вага, тому людям, що страждають від нього, бажано придбати вітаміни для схуднення[14-17].

1.4 Огляд складності створення чат-бота

Складність створення полягає в тому, що не дуже просто налагодити пошук виданої користувачем фрази, на основі термінів та слів, які є у бота. Для чіткого мовлення потрібна спеціальна семантична пошукова машина для загальних шаблонів реплік, яка здатна розпізнавати найбільш подібний шаблон, навіть якщо деякі слова в запиті пропущені або замінені іншими словами. Цей пошуковий механізм повинен мати вбудовану алгоритмічну морфологію, також відому як редагування тексту, а також аналіз, словники синонімів тощо.

У діалоговому вікні слід шукати відповідний сигнал, пам'ятаючи контекст попередньої бесіди. Але головна проблема полягає в тому, щоб охопити різноманітність вхідних сигналів, тому що можна запитати що завгодно і в будь-якій формі.

Справа в тому, що будь-яка пошукова система має дві характеристики: точність і повноту. Звичайні пошукові системи потребують точності: важливо, щоб список з перших десяти результатів був правильно структурований, а повнота полягає в тому, що безліч відповідних документів, вилучених пошуковою системою з Інтернету, нікому не заважають. У випадку семантичного пошуку все навпаки. Точність більшості робіт завжди висока - якщо ви впізнаєте копію, відповідь буде точною. Що стосується повноти, то для такої пошукової машини важко гарантувати визнання більшості введених термінів, оскільки людина, яка вільно володіє мовою, може вигадати будь-який введений термін, їх різноманітність настільки велике, що дуже важко на все відповісти. Також майте на увазі, що з часом бот зіткнеться з «хвилюю» нових подій, заснованих на окремих подіях. У деяких випадках потік сигналів можна передбачити, але потрібно швидко реагувати [4-7].

Можливості робіт:

- можливість збору зворотного зв'язку;
- сучасні технології мовного процесингу дозволяють вести складно організований діалог з людиною;
- інтеграція з зовнішніми джерелами даних;
- впровадження в мобільні платформи;
- зручний і доброзичливий інтерфейс діалогових систем.
- рішення бізнес задач, таких як технічна підтримка, довідка, просування.

Відмінності автоматизованої комунікації від природної полягають:

- в рівні та методі розуміння віртуальним співрозмовником репліки співрозмовника;
- в загальній схемі і глибині всього діалогу;
- в характері, стилістиці і інші особливості формування відповіді [4 - 7].

Ці відмінності є наслідком природної обмеженості технологій NLP в порівнянні з мовною компетенцією людини наявністю у віртуального співрозмовника можливостей, які людині не доступні.

Обмеження на рівні аналізу вхідного тексту:

- семантично-лексичні дані;
- тематична специфіка робота;
- жаргон, неологізми і т.д.;
- помилки.

Важливий аспект, що потребує уваги підчас розробки:

- тестування, що призводить до збою комунікації;
- діалогова поведінка віртуального співрозмовника;
- набір цілей співбесіди будь то розвага, чи консультація, чи рекомендація або навіть реклама;
- глибина володіння той чи іншою темою (предметною областю);
- варіативність, стилістична структура і інші мовні вимоги до відповідей віртуального співрозмовника;
- комунікативна організація діалогу.

Віртуальний співрозмовник повинен завжди швидко реагувати, володіти великою кількістю даних, не повинен відволікатися, використовувати повторювані формулювання, замінювати дані, взяті з баз даних та служб, мати правильне позиціонування в спілкуванні: не потрібно намагатися наслідувати людину, підказки та запитання активне використання, корекція проблем з реакцією на спілкування: переформулюйте, будь ласка, питання про те, чи відступати від теми, структуру бесіди відповідно до функціонального призначення даного стику[4].

1.5 Аналіз недоліків аналогів чат-ботів

Складно уявити технологію, у якій немає недоліків. Основні мінуси чат-ботів:

- багато хто не знає, хто такі віртуальні помічники;
- немає усталених майданчиків для розміщення;
- мало можливостей;
- простому користувачеві недоступні потужні боти;
- недосконалий інтерфейс.

Нерозуміння сарказму. Чат-боти призначені для взаємодії з людьми, але останні, на відміну від ботів, схильні говорити неоднозначно і часто надають прихованого значення їх словам. Хоча роботи-чати чудово справляються з простими питаннями, дають швидкі та точні відповіді, але навіть ботам зі штучним інтелектом важко розпізнати сарказм, іронію та гумор. Вони сприймають кожне слово як належне і не можуть читати між рядків. Вчені постійно працюють над розвитком штучного інтелекту, намагаючись гуманізувати комп'ютерні програми, але на даний момент їх здібності залишають бажати кращого в цьому плані[4-7].

Погана імпровізація. Роботи в чаті працюють добре, доки розмова не виходить за межі запрограмованого алгоритму. Однак розробники не можуть передбачити один мільярд можливих сценаріїв, тому не модель поведінки користувачів турбує бота. Чат-бот задає навідні запитання, щоб з'ясувати, що означає партнер у чаті. Якщо подальші відповіді замовника все ще незрозумілі, програма продовжить повторювати ті самі уточнюючі запитання. Як результат, клієнт дратується і незадоволений послугою і залишає сторінку або заявку, не надаючи необхідної інформації[4-7]

Не для всіх видів бізнесу. Чат-бот не є універсальним рішенням і підходить не для всіх підприємств. Його використання може бути успішним для компаній, які надають такі послуги, як таксі, харчування, бронювання номерів у готелях, продаж квитків, роздрібна торгівля та інші, де запити

клієнтів здебільшого дуже передбачувані та шаблонні. Однак багатьом компаніям, що займаються бізнесом (B2B), потрібно надати ретельну консультацію з індивідуальним підходом та відповідати на ситуацію на місцях, щоб задовольнити споживачів та завоювати їхню довіру. На жаль, чат-боти не можуть надавати якісне обслуговування клієнтів у таких випадках і не можуть стати повноцінною заміною цього типу бізнес-співробітників [11-13].

Слід взяти до уваги такі нюанси, як:

- є складні системи фільтрів та налаштувань;
- є досить багато різноманітних сценаріїв використання продукту, з якими один бот не впорається;
- в складних питаннях все одно необхідно підключення до діалогу оператора;
- не використовує методи самонавчання;
- є сфери, в яких програма не впорається із запитом (наприклад, складна бізнес послуга).

Аналогами виступають такі сервіси:

- VivAlternativa;
- Vitagen.

Поданню розробку планується застосовувати для продажу та підігріву лідів. Користувачів відправляють не на сайт, а в месенджер, де його зустрічає бот. Користувачеві доступні інтерактивні команди, за допомогою яких він може отримати потрібну інформацію і зробити замовлення або запитати зворотний дзвінок. Електронний продавець-консультант - це один з найпопулярніших сценаріїв[11-13].

1.6 Постановка задачі дослідження

Об'єктом дослідження є методи розробки бот-систем.

Метою роботи є проведення дослідження популярності сучасних платформ, месенджерів, визначення програмної платформи (фреймворк) та

інтерфейс прикладних рішень і бібліотек API з великою кількістю доступних інструментів (функцій, класів, структур), які підходять для сучасної розробки бота.

В ході дослідження отримані такі результати – була проведена постановка задачі, що включає в себе:

- дослідити існуючі підходи для розробки бота;
- дослідити існуючі проблеми чат-ботів;
- визначення програмної платформи, а саме фреймворків та інтерфейс прикладних рішень і бібліотек API з великою кількістю доступних інструментів: функцій, класів, структур, які підходять для сучасної розробки бота;
- дослідити методи машинного навчання для вдосконалення ботів, та їх взаємодію з користувачем;
- розробка прикладної програми.

2 РОЗРОБКА ОСНОВНИХ РІШЕНЬ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ

2.1 Інструменти використані для розробки продукту

Щоб зробити вибір мови програмування для чат-бота, потрібно зрозуміти, а що таке чат-бот. Чат-бот - це програма, яка з'ясовує потреби користувачів, а потім допомагає задовольнити їх. За статистикою і мою особисту аналізу і пошуку в інтернеті, я з'ясовано, що гущавину всього в якості мови програмування для чат-бота вибирають Python.

Наведемо кілька прикладів переваг створення бота на Python:

- ваші можливості практично безмежні;
- економія грошей;
- отримання навичок, які можуть стати в нагоді;
- можете заробляти на створенні «роботів».

Недоліки:

- витрата часу і сил на вивчення мови програмування.

За підсумками, у кожної мови програмування є свої плюси і мінуси, кожна мова хороший для конкретних цілей, наприклад якщо створювати сайт, то за описаним критерієм потрібно вибрати JavaScript, а для написання чатбот в месенджері Telegram більше підійде Python так як він більш простий в написанні, має багато документації і має великий вибір бібліотек, які допоможуть створити хороший сервіс[18-22].

Python - це високорівнева мова програмування загального призначення, який також використовується для розробки веб-додатків. Мова націлений на підвищення продуктивності праці розробників і читабельності коду[19].

Python підтримує різні парадигми програмування: структурну, об'єктно-орієнтовану, функціональну, імперативну і аспектно-орієнтовану. Мова включає в себе динамічну типізацію, автоматичне керування пам'яттю, повне самоспостереження, механізм обробки виключень, підтримку

багатопотокових обчислень і практичні структури даних високого рівня [18-22].

Переваги Python [19]:

- відкрита розробка;
- досить простий у вивченні, особливо на початковому етапі;
- особливості синтаксису стимулюють програміста писати добре читається код;
- надає кошти швидкого прототипування і динамічної семантики;
- має велике співтовариство, позитивно налаштоване по відношенню до новачків;
- безліч корисних бібліотек і розширень мови можна легко використовувати в своїх проектах завдяки гранично уніфікованого механізму імпорту та програмним інтерфейсами;
- механізми модульності добре продумані і можуть бути легко використані;
- абсолютно все в Python є об'єктами в сенсі ООП, але при цьому об'єктний підхід не нав'язується програмісту.

Недоліки Python:

- не дуже вдала підтримка багатопоточності;
- на Python створено не так вже й багато якісних програмних проектів у порівнянні з іншими універсальними мовами програмування, наприклад, з Java;
- відсутність комерційної підтримки засобів розробки (хоча ця ситуація з часом змінюється);
- початкова обмеженість коштів для роботи з базами даних;
- бенчмарки показують меншу продуктивність Python в порівнянні з основними Java VM, що створює цій мові репутацію повільного.

У Python існує реалізація PyPy, яка за швидкістю близька до Java, але не має всіх функцій оригінальної мови. Python не підходить для завдань, які потребують багато пам'яті - їх краще вбудовувати у формат C або C++.

Іншим недоліком є сильна залежність мови від системних бібліотек, що ускладнює перехід до інших систем. Для цих цілей існує інструмент Virtualenv, але він також має недоліки: надмірність методів повної ізоляції, милиць, дублювання системних бібліотек.

Інша проблема полягає в тому, що Global Interpreter Lock (GIL) не дозволяє одночасно запускати декілька потоків Python у реалізації CPython.

Однак GIL можна відключити на деякий час, як це відбувається з математичним пакетом NumPy.

Мова програмування Python була обрана через високу швидкість розробки та ізоляції кожного модуля, оскільки кожен програмний файл Python може запускатися незалежно від зовнішніх модулів і підтримує всі функції модульного програмування. Крім того, Python має велику кількість справді безкоштовних бібліотек, які не вимагають оплати навіть після тривалого використання, мова дуже зручна для швидкого проектування та розробки програм, які легко підтримувати на ранніх стадіях розробки проектів. Ідеально підходить для концептуальних проектів та реальних випробувань [18-22].

Функціонал Python полегшує додавання зовнішніх бібліотек для швидкого впровадження необхідної функціональності та приділяє більше уваги архітектурним особливостям програми.

Основними функціями чат-бота є:

- функція перегляду та додавання правил від простих до складних із загальної основи правил;
- переклад слів з більшості природних мов на іспанську, що дозволяє зберегти слова у словнику користувача та підтримувати загальний словник;
- функція перегляду та додавання нових слів з онлайн-словника;
- підтримка можливостей авторизації користувачів та їх

ідентифікація системою.

Тобто збереження нових користувачів до бази даних, підтримка відповідних словників залежно від власника та збереження окремих слів, правил для подальшого відображення користувачів, повторного застосування автентифікації користувачів або необхідності вибору дозволити дані

- функція ведення користувацького словника, тобто можливість редагування словника на основі користувача та його знань;
- можливість тестування вивчених слів та поданого матеріалу методом тестування;
- можливість перегляду статистики результатів тесту;
- має функцію ведення журналу консолі для вхідних повідомлень користувачів, яка містить інформацію про ім'я користувача, текст надісланого повідомлення тощо. З'являється додавання нових користувачів та інші запити до бази даних [18-22].

Кожна функція відповідає гілці її функцій у системі та пов'язаному з нею програмному коду, до кожної функції можна отримати доступ за допомогою набору програмних команд, що утворюють інтерфейс із користувачем, що відображає порядок користувацьких команд.

Модуль TeleBot - це оболонка для запитів до TelegramBotAPI, яка використовується для спрощення та мінімізації написаного коду. Усі типи включені до `tips.py`. Усі вони повністю відповідають визначенню типів API Telegraph, за винятком поля Повідомлення, яке перейменовано: `_user` (оскільки: зарезервований маркер Python). До таких атрибутів, як `message_id`, можна безпосередньо отримати доступ, наприклад: `message.message_id`. При написанні програм, які не дуже вимогливі з точки зору швидкості виконання, слід зазначити, що атрибут `message.chat` може належати конкретному користувачеві та груповому чату, це окупається перевагами мови [23-24] .

Всі методи API належать до класу TeleBot. Вони були перейменовані відповідно до загальних конвенцій про іменування Python. Наприклад: `sendMessage` `send_message`, `editMessageText`, `edit_message_text`. Послуга,

прикрашена декоратором екземплярів TeleBot, є менеджером повідомлень. Обробники повідомлень складаються з одного або декількох фільтрів. Кожен фільтр повертає справжнє або хибне значення для даного повідомлення, і оператор має дозвіл обробляти повідомлення, якщо воно повертає True [23-24].

Засоби Python для роботи з БД. Для комфортної роботи з SQL потрібно використовувати мову Python для організації відповідного інтерфейсу та протоколів для взаємодії з БД. Для цього SQLite має відповідний PPI через драйвер Connector для роботи із зовнішніми пристроями SQL.

З'єднувач є автономним драйвером для використання бази даних з іншими інструментами, включаючи мову програмування Python. Драйвер дозволяє налаштовувати підключення та запити до бази даних за допомогою стандартного каталогу "sqlite.connector", а також основних функцій для роботи з базою даних [18].

Підключіться до бази даних за допомогою функції `sqlite.connector.connect`, основними параметрами якої є: «host» - розташування домену, «database» - назва бази даних, необхідна для підключення, «user» - користувач та «password» - пароль для з'єднання. Для перевірки підключення до бази даних існує метод `"is_connected ()"`, який повертає `"true"` при успішному підключенні. Використовуйте метод `"close ()"` [18], щоб закрити з'єднання.

Він може виконувати запити до бази даних за допомогою об'єкта `"cursor ()"`, що включає метод `"cursor.execute (query, args)"`, який приймає вираз сформованого запиту та зазначені в ньому параметри як аргументи. Метод повертає масив об'єктів до курсору з набором полів, які можна прочитати методом `"fetchone ()"`, `"fetchall ()"` або `"fetchmany ()"` і програмно взаємодіяти з набором об'єктів. Методи читання відрізняються тим, як дані повертаються з основного об'єкта курсору [18].

Можливість запиту додати один або кілька наборів даних до бази даних за допомогою об'єкта `"cursor ()"`, використовуючи методи `"execute (query,`

args)" і "execute many (query, all_arguments)", які успішно повертають успішно введений номер. додано рядки та приймає запит та параметри. Крім того, метод "обязательства ()" використовується для підтвердження додавання полів до бази даних [18].

Тобто ви можете вибрати основні сутності, що працюють з базою даних: підключення, від'єднання бази даних, курсор для читання, виконання запитів через суть запиту та аргументів. Це мінімальний набір методів, який слід використовувати при роботі з базою даних через каталог Connector [18].

Мова структурованих запитів SQL. SQL - це декларативна мова програмування, призначена для обробки великих обсягів даних за допомогою структурованих запитів. Він використовується при роботі з реляційними базами даних, що передбачає три завдання: робота з базою даних, зокрема створення та модифікація структури бази даних, маніпулювання та обробка даних для їх належного забезпечення. Запит - це сформульований набір параметрів для вибірки рядків з бази даних [18].

Для зберігання даних, які необхідні для коректної роботи чат-боту була обрана база даних SQLite.

SQLite - це компактна інтегрована реляційна база даних. Вихідний код бібліотеки було передано в суспільне надбання. Це чисто реляційна база даних [19].

Підтримувані типи даних [18]:

- NULL: значення NULL;
- INTEGER: ціле число зі знаком, що зберігається в 1, 2, 3, 4, 6 або 8 байтах;
- REAL: число з плаваючою комою, що зберігається в 8-байтовому форматі IEEE;
- TEXT: текстовий рядок, закодована UTF-8, UTF-16BE або UTF-16LE;
- BLOB: тип даних, що зберігаються в тій же формі, в якій вони були отримані.

Переваги:

- файлова: вся база даних зберігається в одному файлі, що полегшує переміщення;
- стандартизована: SQLite використовує SQL;
- дуже добре підходить для розробки і навіть тестування: на етапі розробки більшості потрібно масштабується. SQLite з його багатим набором функцій може надати більш ніж достатню функціональність, а також бути досить простим для роботи з одним файлом і пов'язаної бібліотекою.

Недоліки:

- відсутність призначеного для користувача управління: просунуті БД надають користувачам можливість управляти зв'язками в таблицях відповідно до привілеями, але у SQLite такої функції немає;
- неможливість додаткової настройки: SQLite не можна зробити більш продуктивною.

В даному підрозділі була представлена база даних, необхідна для роботи чат-бота, було розглянуто її поняття, переваги і недоліки.

СУБД використовуються в сучасних ІС для роботи з базами даних. Система складається з набору апаратного та програмного забезпечення, що забезпечує створення, модифікацію, управління, контроль та використання бази даних. СУБД може бути встановлена на додаток до роботи зі структурою бази даних для маніпулювання даними, контролю доступу до них, контролю роботи процесів СУБД, а також може бути обладнана системами відновлення, але існують окремі версії з усіченою функціональністю. Найпопулярніші бази даних включають MySQL, Oracle, PostgreSQL, Interbase, Sybase, Microsoft SQL Server, IBM DB2 та Firebird [18].

Архітектура СУБД має три організаційні рівні, які називаються ANSI-SPARC:

- зовнішній - представлення бази даних на рівні користувача;
- концептуальний - подання, що описує дані та їх зв'язок у базі даних у загальному вигляді;

- внутрішній - відображення бази даних на фізичному носії, тобто в системі.

Системи управління базами даних побудовані на архітектурі клієнт-сервер, де єдиний сервер з базою даних обслуговує клієнтів, користувачів через запити або через PPI, але існує архітектура файлового сервера, де відповідний файл надсилається клієнту, і його обробка включено. сторона клієнта.

На рівнях організації існують логіни, концептуальна та фізична моделі бази даних, що використовуються для проектування бази даних, і вони мають належне призначення.

Інтерфейс програмного забезпечення SQLite схожий на MySQL та PostgreSQL, хоча дещо відрізняється. Основна різниця між SQLite та іншими базами даних SQL полягає в тому, що в SQLite немає типів даних. Усі поля зберігаються, наприклад, із роздільником нульового рядка, замість двійкового подання цілого числа. Ви можете використовувати будь-який тип даних при створенні таблиці - VARCHAR, TEXT, INTEGER, FLOAT тощо, але вони зазвичай ігноруються механізмом [27-35].

Ключові особливості SQLite Server:

- одночасна робота необмеженої кількості користувачів бази даних;
- просте управління та регулювання;
- кількість полів не може перевищувати п'ятдесяти мільйонів;
- швидке виконання команд;
- ефективне та просте адміністрування та безпека.

База даних SQLite забезпечує інформаційну підтримку бази даних чат-ботів шляхом вибору вітамінів, а також дозволяє чат-боту зручно зберігати та обробляти інформацію [18].

2.2 Розробка способу організації даних та структура таблиць

Для керування даними в проєкті використовується модуль SQLite3. Він відповідає за отримання та передачу даних між сайтом та таблицями бази даних SQLite. В базі даних знаходиться таблиця, які створюються автоматично при старті проєкту.

В таблиці 2.1 та на рисунку 2.1 описані поля таблиці “testdb”. Вона призначена для зберігання даних користувача, персонального телеграм ID, ім’я та запиту.

Таблиця 2.1 — Структура таблиці «testdb» бази даних

№ з/п	Назва поля	Тип поля	Опис поля
1	ID	Int(11), AUTOINCREMENT	Ідентифікатор запиту
2	gender	VarChar(100)	Юзернейм користувача
3	Age	VarChar(2)	Ідентифікатор користувача телеграм
4	taking	VarChar(100)	Обробка несумісності
5	priorities	VarChar(100)	Пріоритет користувача
6	Date	DATE(30)	Дата подання запиту

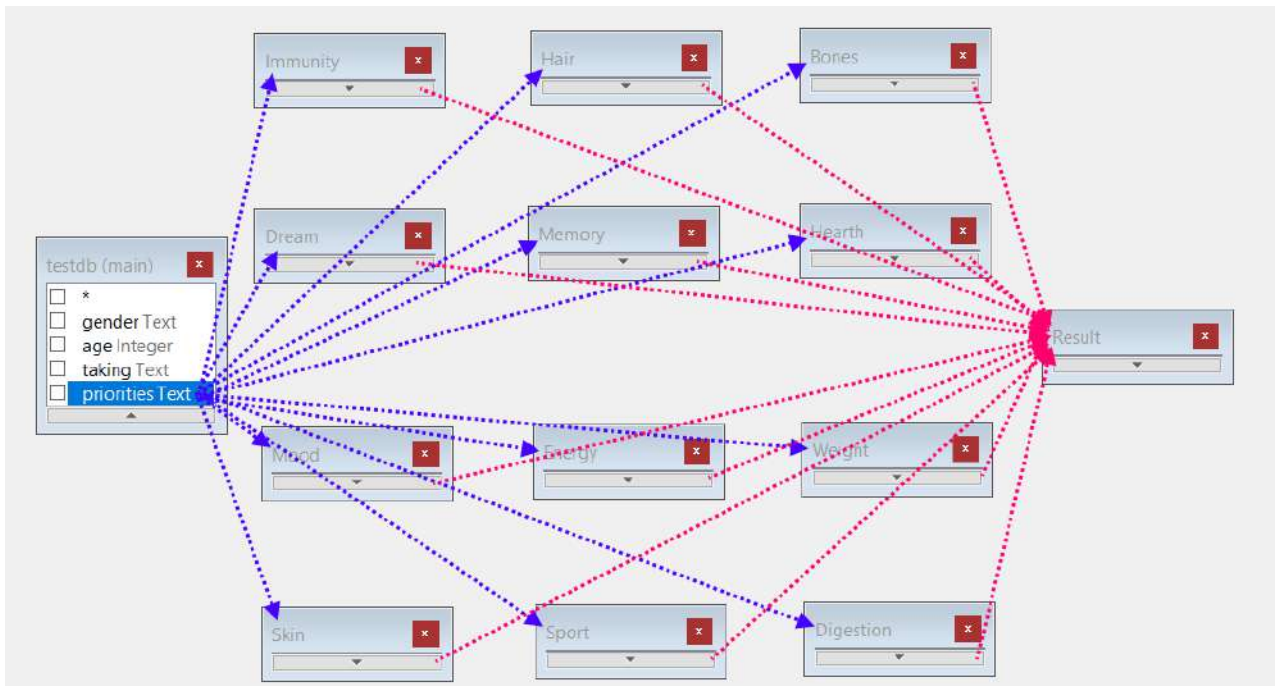


Рисунок 2.1 – Структура таблиці «testdb» бази даних

2.3 Розробка проєкту в обраному середовищі

PyCharm — інтегроване середовище розробки для мови програмування Python. Надає засоби для аналізу коду, графічний зневаджувач, інструмент для запуску юніт-тестів і підтримує веброботку на Django. PyCharm розроблена чеською компанією JetBrains в 2010 році.

PyCharm працює під операційними системами Windows, Mac OS X і Linux.

Огляд інтерфейсу програмного середовища наведено на рисунку 2.2.

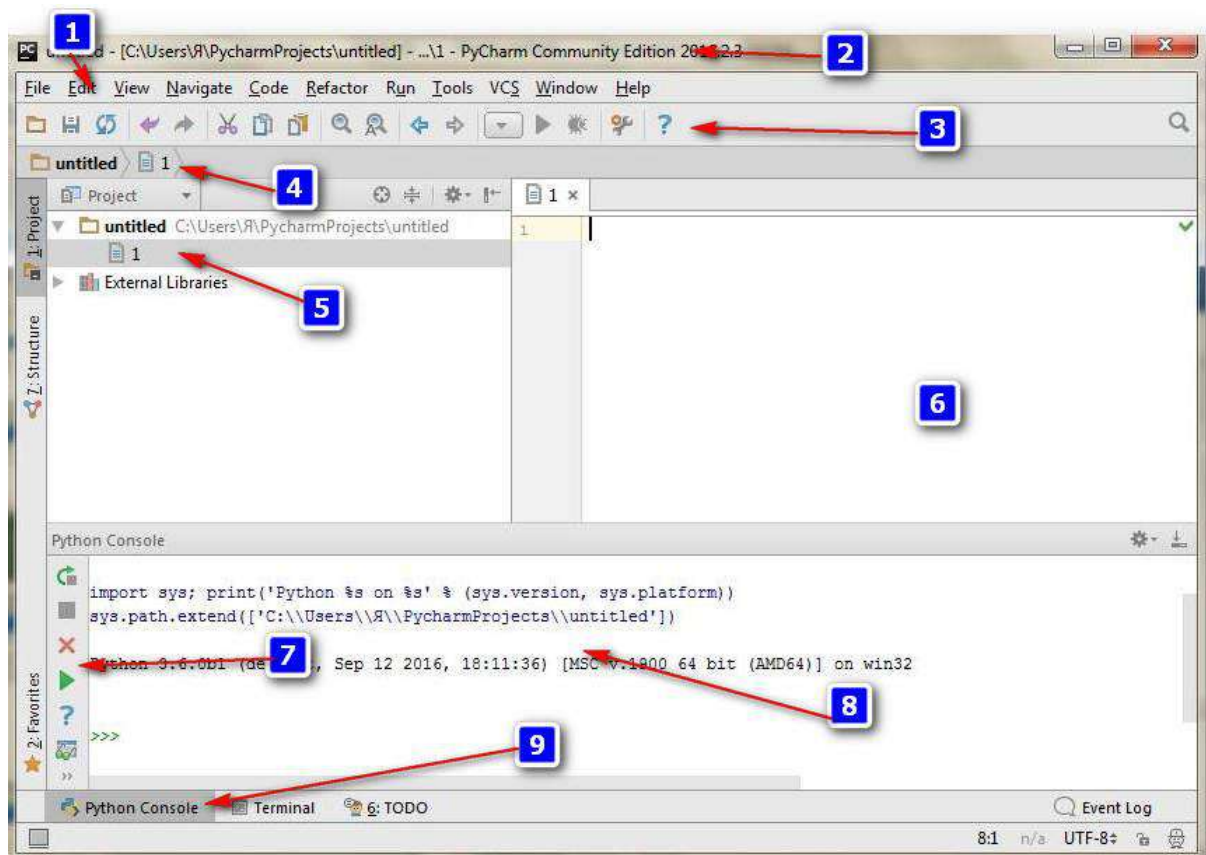


Рисунок 2.2 – Огляд програмного середовища

Позначка 1. Головне меню та панель інструментів - містять команди, які впливають на весь проєкт або його великі частини, такі як відкриття, створення проєкту, рефакторинг коду, запуск та налагодження програм, зберігання файлів під керуванням версії та багато іншого;

Позначка 2. Заголовок діалогового вікна;

Позначка 3. Основна панель інструментів дублює важливі команди головного меню для швидшого доступу. За замовчуванням основна панель інструментів прихована;

Позначка 4. Панель навігації - швидка альтернатива перегляду проєкту. Використовуйте його для навігації по проєкту та відкривайте файли для редагування;

Позначка 5. Інспектор об'єктів;

Позначка 6. Редактор - тут ви можете читати, створювати та змінювати свій код;

Позначка 7. Вікно інструментів - вторинні вікна, які забезпечують доступ до різних конкретних завдань (управління проєктами, пошук, запуск і налагодження, інтеграція з системами контролю версій тощо);

Позначка 8. Стрічка стану - вказує на статус вашого проєкту, всю IDE, а також показує різні попереджувальні та інформаційні повідомлення;

Позначка 9. Режим роботи середовища. Додаткові опції.

2.4 Висновки до розділу 2

В цьому розділі було проаналізовано використані при розробці інструменти, саму середу розробки, реалізовані алгоритми та модулі, відповідні за реалізацію. Була представлена інформація про базу даних та продемонстровані всі її таблиці.

Як результат, було розроблено програмний продукт, який повністю реалізує поставлену задачу.

Програмні модулі архітектурно розроблені та розроблені, щоб протистояти модульній архітектурі. Вони розподіляються відповідно до основних технічних та програмних функціональних можливостей, тобто кожен файл виконує окреме завдання, що викликається загальним сценарієм, є файл `main.py`, який керує основною роботою програми, приймає всі команди в систему і відповідає для зовнішніх для всіх функцій чат-бота, який імпортує файли, розширює та дозволяє використовувати реалізовані функції з інших програмних модулів. У свою чергу, дочірні програмні модулі можуть бути розширені за межами програмних модулів або інших системних модулів.

3 РОЗРОБКА ПРОГРАМНОГО КОМПЛЕКСУ ЧАТ-БОТА

Програма призначена для підбору комплексу вітамінів за характеристиками користувача.

Чат-бот містить в собі наступні функції:

- опитування користувача;
- вибір інтересуючого пріоритету;
- обробка опитування та отримання результату.

3.1 Основні розроблені алгоритми

Під час розробки програмного продукту було розроблено такі алгоритми:

- старт або привітання;
- заповнення персональних даних(вік та стать);
- вибір пріоритету;
- опитування за пріоритетом;
- отримання результату рекомендації;

Алгоритми проєкту.

Крок 1. Старт або привітання. Для користування ботом, користувачеві одразу треба зв'язатися з телеграм-ботом. Для цього в пошуку меседжера телеграм треба ввести username бота. У кожного бота є унікальний username. У нашому випадку це @te555tz_bot. На головному екрані бота, буде запропоновано натиснути Start, або ввести команду /start, після чого з'явиться вікно привітання бота з користувачем, та відкрита клавіатура навігацій, яка надає доступ до функцій бота;

Крок 2. Заповнення персональних даних(вік та стать). При старті, користувач отримує вказівку на вибір статі. Після чого бот посилає запит на ведення поточного віку. Після того, як користувач послав запит статі та віку, бот приймає запит і опрацьовує за допомогою бази даних;

Крок 3. Вибір пріоритету. Далі користувач отримує можливість вибору пріоритету. Бот охоплює дванадцять пріоритетів, які стосуються здоров'я людини.

Крок 4. Опитування за пріоритетом. Після того як користувач вибрав інтересуючу йому тему, бот починає опитування за вибраним пріоритетом.

Крок 5. Отримання результату рекомендації. База даних аналізує відповіді за опитуванням та видає певну рекомендацію в залежності від відповідей користувача.

3.2 Процес створення бота та огляд Telegram Bot API

Щоб створити бота, ви повинні бути користувачем Telegram і отримати маркер бота від робота, який надає ключ ППІ, будь-який користувач може підключити програму управління через Telegram ППІ як інтерфейс. ППІ забезпечує доступ до повної функціональності телеграм, тому клієнти будь-якої складності можуть створювати телеграми з можливістю розширення стандартних функціональних можливостей і дозволяють створювати інші інструменти для телеграм як боти або власні текстові описи для посилань або додатків. створювати пакети обробки інформації з телеграм. Це відкриває широкий спектр технічних можливостей для розробки технологій та написання, з інтерфейсом роботів-чатів за допомогою телеграм ППІ [23-29].

На мові програмування Python присутня бібліотека для зручної роботи з ППІ Телеграму, під назвою “pyTelegramBotAPI”. Вона дозволяє використовувати ППІ в лаконічній та спрощеній формі за допомогою декораторів та відповідних їм методам.

Для роботи з бібліотекою її необхідно встановити та імпортувати командами:

```
>pip install pyTelegramBotAPI
>import telebot
```

Після чого достатньо ініціалізувати сутність бота через виклик відповідного методу сутності telebot:

```
>bot = telebot.TeleBot("TOKEN")
```

Щоб перевірити наявність нового повідомлення чи команди, використовуйте метод `message_handler`, який отримує значення команд

відповісти на бота, а потім виконати функцію, яку він буде виконувати після отримання відповідного повідомлення [23]

Після ініціалізації всіх необхідних команд все, що вам потрібно зробити, це запустити бот-запит на нові повідомлення за допомогою цього методу:

```
>bot.polling()
```

Функція `message_handler` приймає всі доступні типи сутності, регулярні вирази, команди або лямбда-функції в ППІ Telegram.

Приклад дизайну:

```
>@bot.message_handler()
```

Ви можете налаштувати власну клавіатуру за допомогою методів `ReplyKeyboardMarkup`, щоб вказати розкладку клавіатури з параметром `row_width` та додати будь-які нові методи до методу `KeyboardButton` об'єкта типу.

Можна надіслати за допомогою методу `send_message` суб'єкта телеботу:

```
>bot.send_message(chat_id, "Choose one letter:", reply_markup=markup)
```

В комплексі набір функцій для обробки користувацьких повідомлень має вигляд:

```
>@bot.message_handler(func=lambda message: True)
```

```
>def echoFunction_all(message):
```

```
>results = operations(message.text)
```

```
>bot.reply_to(message, "results")
```

`pyTelegramBotAPI` має лаконічну структуру, організований функціонал та надає всі необхідні можливості для розробки Телеграм ботів [23].

3.3 Модуль БД

Модуль бази даних розміщується у файлі на основі модульної архітектури, що містить функції для виконання певних запитів.

Окремий програмний модуль описує функціонал для створення різних типів запитів, залежно від потреби, розробляється нова функція для обробки запиту та передаються значення, необхідні для обробки запиту. Потім запит виконується на базовому наборі послідовностей для додавання, видалення та редагування необхідного запиту. Нижче наведено ряд функцій для виконання деяких запитів у базі даних [18].

Функція з додавання нового користувача до БД:

```
>def addUserIntoUsers(chat_id, user_id, first_name, last_name, username):
>sql = "INSERT INTO users (`chat_id`,`user_id`,`first_name`,`last_name`,
`username`)"
VALUES (%s, %s, %s, %s, %s)"
>val = (chat_id, user_id, first_name, last_name, username) >try:
>    mycursor.execute(sql, val)
>    db.commit()
>    print("1 record inserted, ID:", mycursor.lastrowid)
>    return 1 >except:
>    print("Record is NOT inserted. User exists")
>    return 0
```

Функція видалення користувача з БД:

```
>def addUserIntoUsers(field_rule, parametr):
>sql = "DELETE FROM user WHERE %s = %s;"
>val = (field_rule, parametr) >try:
>mycursor.execute(sql, val)
>db.commit()
>print("1 record deleted, ID:", mycursor.lastrowid)
```

```
>return 1 >except Error as err:
```

```
>print("Record is NOT deleted. \n Something went wrong: {}".format(err))
```

```
>return 0
```

Функція додавання слова до словника користувача:

```
>def wordIntoDictionary(id_word , word , transcription , translation ,
meaning , example , image, id_user_dictionary , id_user , id_dictionary , add_date ,
level):
```

```
>sql = "INSERT INTO dictionary (`id_word` , `word` , `transcription`
, `translation`
```

```
 , `meaning` , `example` , `image`) VALUES (%s , %s , %s , %s , %s , %s ,
%s);" >val = (id_word , word , transcription , translation , meaning , example ,
image) >try:
```

```
>mycursor.execute(sql, val)
```

```
>db.commit()
```

```
>print("1 record inserted, ID:", mycursor.lastrowid) >except Error as err:
```

```
>print("Record is NOT inserted. In to 'dictionary'. \n Something went wrong:
{}".format(err))
```

```
>sql = "INSERT INTO user_dictionary (`id_user_dictionary` , `id_user` ,
`id_dictionary` , `add_date` , `level`) VALUES (%s , %s , %s , %s , %s);" >val = (id_user_dictionary , id_user , id_dictionary , add_date , level)
```

```
>try:
```

```
>try:
```

```
>mycursor.execute(sql, val)
```

```
>db.commit()
```

```
>print("1 record inserted, ID:", mycursor.lastrowid)
```

```
>return 1
```

```
>except Error as err:
```

```
>print("Record is NOT inserted. In to 'user_dictionary'. \n Something went
wrong: {}".format(err))
```

```
>return 0
```

3.4 Модуль обробки нових користувачів

Модуль для обробки нових користувачів призначений для обробки вводу користувача та початкових знань про роботу та функціональність бота, алгоритм його роботи показано на рисунку 3.1 у вигляді блок-схеми.



Рисунок 3.1 – Алгоритм роботи модуля обробки нового користувача

Після отримання команди модуль викликає відповідний під-модуль функції, який отримує відповідні дані з бази даних і пересилає їх користувачеві.

Під час роботи з системою кожен новий користувач починає з надсилання команди `"/start"`, яка може обробити бота для отримання загальнодоступної інформації про користувача та збереження її для повторного використання або авторизації, а після відправлення команди бот надсилає базовий інтерфейс налаштування у формі командної клавіатури та швидкого набору основних команд; і початкова текстова інструкція з набором основних команд для доступу до основних функцій бота.

Після отримання команди створюється запит на додавання користувача до бази даних та відповідь на успішну реєстрацію із пропозицією користувачеві завантажити командну клавіатуру для взаємодії з ботом. Не вдалося додати користувача до бази даних, користувач не матиме доступу до функцій збереження слів у словнику користувача та збереження результатів тесту до статистики продуктивності. І вам буде запропоновано ввести команду пізніше, щоб зареєструвати профіль користувача. Після завершення первинної реєстрації особа повинна додати принаймні п'ять слів до словника, щоб отримати доступ до варіантів тестування.

Основні команди також включають команду `"/stop"`, яка дозволяє зупинити бота і при необхідності видалити обліковий запис користувача.

3.5 Висновки до розділу

Вирішуючи, який чат-бот підходить саме вам, спочатку уявіть себе користувачем і подумайте про значення, яке вони намагаються отримати від вашого бота. Чи істотно впливає контекст розмови на це значення? Якщо ні, сміливо використовуйте будь-яку структуру палички.

Іншим аспектом, який слід врахувати, є цільова база користувачів та їх переваги для UX. Деякі користувачі можуть віддавати перевагу боту чату з

візуальними кнопками меню, а не відкритим діалоговим вікном, тоді як інші повинні запитувати бота безпосередньо.

Правильний чат - це той, який найкраще відповідає цінній пропозиції, яку ви намагаєтесь отримати для користувачів. Що стосується поєднання підходів чат-ботів, можна сказати, що система підвищила точність і швидкість відповіді на запитання користувачів.

4 РЕЗУЛЬТАТИ ТЕСТУВАННЯ ТА ЗАСТОСУВАННЯ ЧАТ-БОТУ

Тестування телеграм палички проводили вручну. Після кожного етапу розробки проводився тест продуктивності заздалегідь підготовлених тестових кейсів, які складались для визначення цілей розвитку та використовуваних інструментів [30]. Робота з Telegram-ботом:

- швидкість, з якою бот відповідає на повідомлення про запуск;
- правильна обробка різних типів повідомлень від користувача.
- правильно відобразити діалогове вікно;
- обробка натискання кнопки для відправки повідомлення;
- швидкість, з якою відображається відповідь після того, як користувач написав повідомлення;
- швидкість бази даних;
- оцінка правильності відповідей.

Чат-бот був протестований на мобільному пристрої Apple iPhone 8 Plus із наступними характеристиками.

- екран: 5,5 ", IPS LCD, 1920x1080, мультитач;
- процесор: Apple A10 Fusion, 4x1;
- операційна система: iOS 12.2.4;
- оперативна пам'ять: 3 ГБ;
- вбудована пам'ять: 64 ГБ;
- навігація: GPS;
- Telegram: Telegram v 6.1.

Після приєднання до чату користувачу доступна тільки одна команда – /start. Відправивши цю команду, в відповідь отримуємо основну інформацію щодо опитування. Початок роботи з чат-ботом ілюстровано на рисунку 4.1



Рисунок 4.1 – Привітання та початок опитування

Після заповнення персональних даних, чат-бот аналізує доцільність рекомендації. Наступним кроком є вибір цілі користувача (рисунок 4.2)

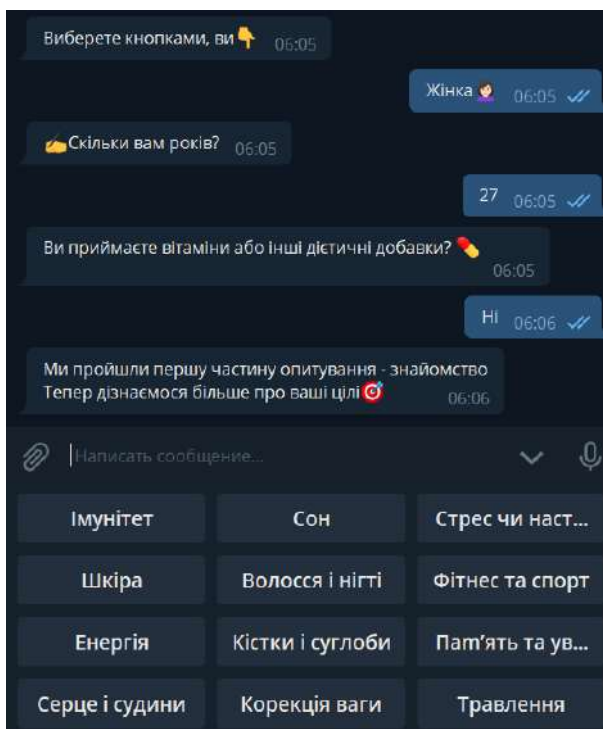


Рисунок 4.2 – Введення персональних даних та вибір пріоритету

Чат-бот обробив пріоритет користувача, та продовжує опитування, починаючи з ключових питань за пріоритетом (рисунок 4.3-4.5)



Рисунок 4.3 – Опитування

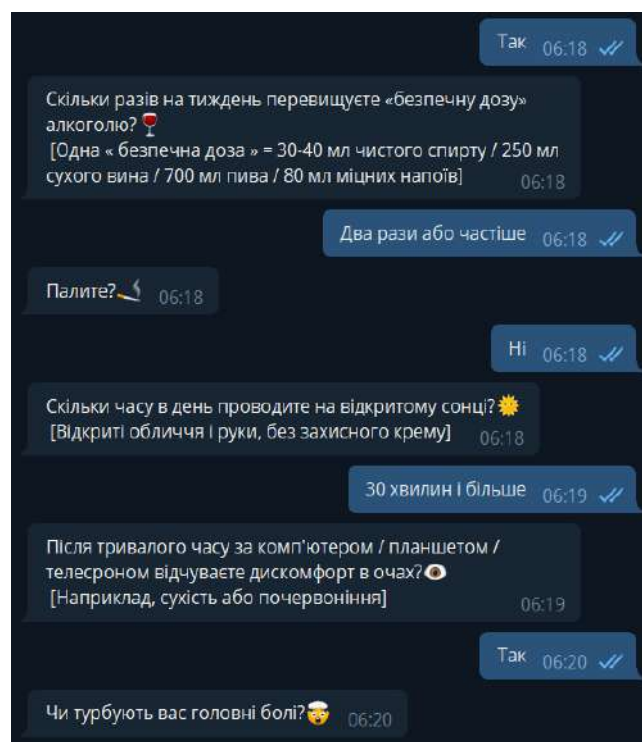


Рисунок 4.4 – Опитування

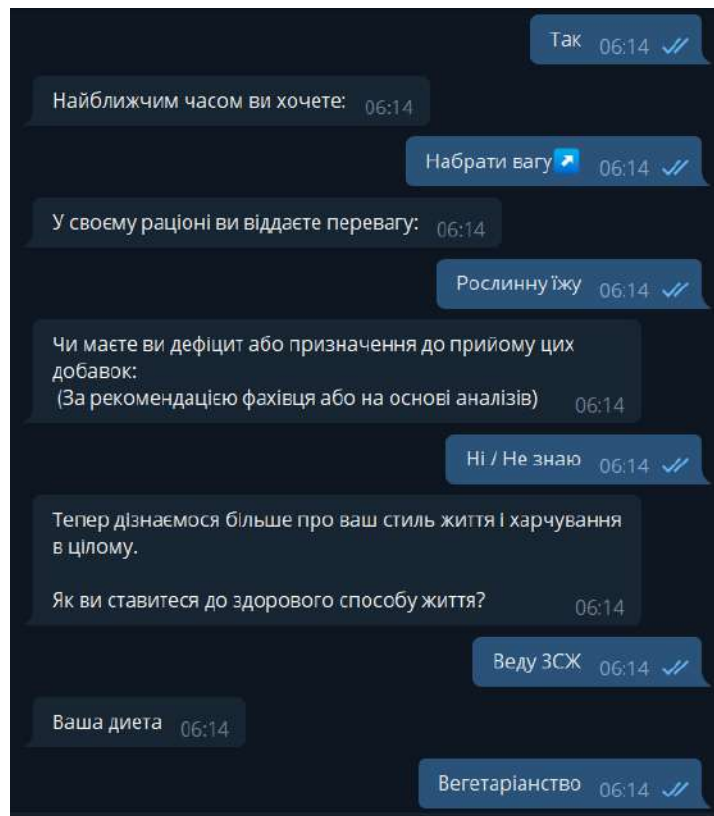


Рисунок 4.5 – Опитування

Після проходження опитування, користувач отримує повідомлення, яке радить отримати консультацію з лікарем, щоб уникнути негативної реакції на організм (рисунок 4.6)

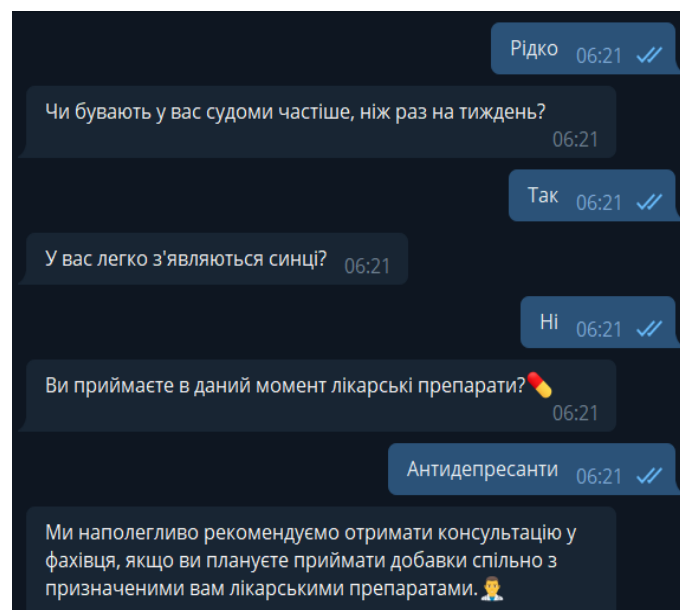


Рисунок 4.6 – Кінець опитування та попередження

Чат-бот записує данні в базу даних, аналізуючи відповіді користувача, щоб надати доцільну рекомендацію персонального комплексу вітамінів (рисунок 4.7)

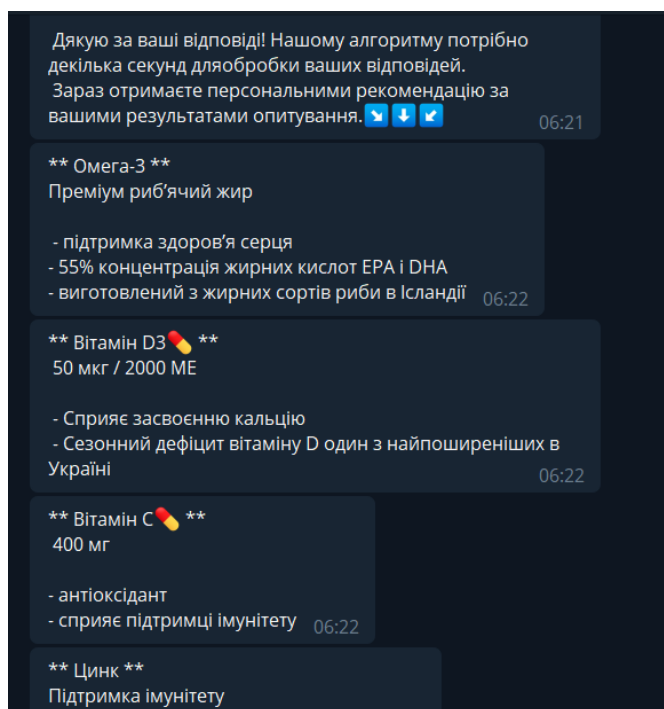


Рисунок 4.7 Обробка відповідей та отримання рекомендації

Отримавши рекомендацію, бот пропонує пройти опитування знову, вибравши інший пріоритет (рисунок 4.8)

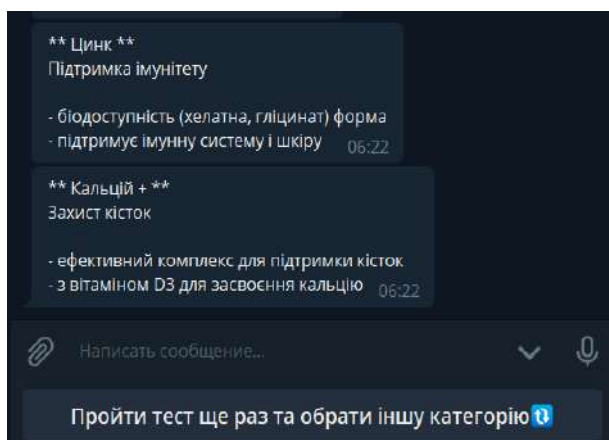


Рисунок 4.8 Отримання рекомендацій з можливістю повторного опитування

В результаті використання чат-бота було встановлено, що всі заплановані функціональні можливості реалізовані відповідно до сучасних методів відбору вітамінного комплексу, і ці можливості реалізовані на платформі Telegram, що дозволяє швидко і легко працювати з багатьма програмами, не вимагаючи реєстрації в всі та зручні для користувача ви можете отримати доступ до навчального процесу в одному місці. А архітектурні особливості чат-бота відповідають реалізації конкретних вимог та функцій міні-додатку, які дозволяють реалізацію та розширення функціональних можливостей при необхідності.

Хоча чат-бот має ряд переваг, до недоліків можна віднести залежність від платформи Telegram та обмеження інтерфейсу у вигляді повідомлень. Але ці недоліки не впливають на процес навчання користувача та роботу основних функцій чат-бота.

Після завершення процесу опитування користувач може продовжити вивчати наступний елемент завдання або виконати іншу команду.

5 ЕКОНОМІКО-ОРГАНІЗАЦІЙНА ЧАСТИНА

Основна мета роботи - розробити робота для платформи Telegram, який надаватиме зручні, персоналізовані рекомендації користувачеві. Додаток здатний надати інформацію про необхідні вітаміни, корисні дієтичні добавки тощо. Дослідження в цій роботі були зосереджені на персоналізації робота та адаптації його до кожного користувача, а також правильній відповіді на його повідомлення. Опис ідеї проєкту наведено у таблиці 5.1.

Таблиця 5.1 – Опис ідеї проєкту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Отримання чіткої рекомендації за характеристиками користувача, які оброблюються опитуванням, зв'язаними з комплексом вітамінів	Пошук користувачем корисної інформації	Зручний інтерфейс отримання інформації
	Аналіз повідомлення користувача надання індивідуально підібраної інформації	Відповідно до запиту та до попередніх знань про користувача буде надано індивідуально підібрану інформацію
	Адаптація під конкретного користувача з часом	Чим частіше користувач використовує бота, тим краще його відповіді

Аналоги:

- VivAlternativa;
- Vitagen.

Порівняння проєкту з аналогами враховуючи характеристики наведено у таблиці 5.2.

Таблиця 5.2 – Визначення сильних, слабких та нейтральних характеристик ідеї проєкту

№ п/ п	Технікоекономічні характеристики ідеї	(потенційні) товари/концепції аналогів			N (нейтральна сторона)	S (сильна сторона)
		Мій проєкт	VivAlternat iva	Vitagen		
1.	Аудиторія, вік	3 – ∞	18 – 60	3 – 80+		Немає вікових обмежень
3.	Перспективи росту	Високі	Низькі	Середні		Хороші перспективи зростання
4.	Функціональність	Без обмежень	Обмежена	Частково обмежена	Хороша функціональн ість	
5.	Сприйняття команд	+	+	+	Як і в інших	
6.	Виділення команд з повідомлення	+	–	–		Присутня можливість
7.	Розуміння контексту	У перспекти ві	–	–		Перспектива удосконален ня

Відповідно до таблиці 5.2 можна виділити наступні сильні сторони проєкту:

- високі перспективи для росту аудиторії, особливо якщо урахувати можливе поширення сервісу;
- можливість виділення команд із повідомлення;
- в перспективі – можливість розуміти контекст діалогу.

Для покращення конкурентоспроможності товару послуга повинна бути безкоштовно надана всім користувачам.

Незважаючи на те, що на ринку багато конкурентів, привабливим є вхід, оскільки користувачі хочуть отримати більше рефералів, а витрати на вихід на ринок не великі.

Характеристика потенційних клієнтів стартап-проєкту наведено у таблиці 5.3.

Таблиця 5.3 – Характеристика потенційних клієнтів стартап-проєкту

Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп	Вимоги споживачів до товару
Зручний інструмент тематичного пошуку	Молодь, люди похилого віку, користувачі месенджерів	Зручний інтерфейс. Підтримка задання команд повідомленнями.	Зручний інтерфейс. Красивий дизайн.

5.1 Альтернативи ринкової поведінки

Як базові необхідні функції визначимо можливість надання інформації щодо основних принципів здоров'я та отримання рекомендацій за відповідями опитувальника. Цих функцій достатньо для отримання рекомендацій, але є перспектива додати продаж необхідних користувачу, вітамінів. Якщо базові функції реалізувати відносно просто, то впровадження магазину значно ускладнює розробку, проте переваги від її реалізації значно перевищують затрати ресурсів. Саме ця функція слугуватиме основним інструментом утримання користувача. На таблиці 5.4 наведено альтернативи ринкового впровадження стартап-проєкту [31].

Таблиця 5.4 – Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Реалізація максимальної кількості функцій	Низька	6 місяців
2	Часткова реалізація функцій з поступовим релізом нових	Висока	4 місяці
3	Відмова від деяких функцій	Низька	3 місяці

Проаналізувавши таблицю 5.4, ми обираємо можливість часткової реалізації функціоналу з подальшим поступовим випуском нових функцій.

Стратегія розширення первинного попиту в цьому випадку доречна, оскільки як лідер нашої компанії не доцільно торгувати для дрібних конкурентів. Розширення рівня первинного попиту може мати великі економічні вигоди. У цьому випадку компанія вживає заходів для формування попиту (навчання споживачів користуванню товаром, розвиток регулярного попиту, збільшення одноразового споживання) та просування нових застосувань існуючих товару (бота), шляхом введення нових можливостей. У таблиці 5.5 наведено вибір цільових груп потенційних споживачів.

Таблиця 5.5 – Вибір цільових груп потенційних споживачів

№ п/п	Цільова група потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Похилого віку (45+)	Висока	Високий	Висока	Низька
2	Середнього віку (25 - 45)	Середня	Середній	Висока	Середня

За результатами аналізу потенційних груп споживачів обрані обидві цільові групи, для яких пропонуватиметься товар

Тому оберемо стратегію концентрованого маркетингу, як стратегію охоплення ринку (таблиця 5.6).

5.2 Базова стратегія розвитку

Таблиця 5.6 – Визначення базової стратегії розвитку

Обрана альтернатива розвитку проєкту	Стратегія охоплення ринку	Ключові конкурентоспромо жні позиції	Базова стратегія розвитку
Часткова реалізація функцій з поступовим релізом нових	Стратегія концентровано го маркетингу	Виділення ключових слів із повідомлення; Функціонал	Стратегія диференціації

5.3 SWOT-аналіз та визначення пріоритетних груп зацікавлених осіб

Був здійснений SWOT-аналіз, який представлений в таблиці 5.7.

Таблиця 5.7 – SWOT-аналіз

Сильні сторони:	Слабкі сторони:
- немає вікових обмежень - хороші перспективи росту - сукупність функцій - перспектива удосконалення	- організація та оптимізація коду - не містить адмін-панелі - не має можливості придбання товару - відсутність аудиторії
Можливості:	Загрози:
- високий попит через пандемію; - висока швидкість розробки - отримання рекомендацій щодо необхідних вітамінів, добавок, тощо. - обробка результату за можливими алергіями, способу життя чи раціоном користувача	- велика кількість конкурентів - мінливі вимоги користувачів

Зацікавленими особами мого проєкту є такі групи:

- ініціатор та розробник проєкту;
- інвестори;
- споживачі.

Ініціатор та розробник проєкту підпадає під повноваження та відповідальність, має прямий вплив на весь процес проектування та реалізації проєкту та має найбільший інтерес до проєкту.

Інвестори - це потенційні зацікавлені сторони, на яких впливає безпосередньо і які інвестують у розвиток свого капіталу з певним ступенем ризику з метою отримання доходу, мають економічний інтерес.

Споживачі перебувають у сфері непрямого впливу, зацікавлені в стабільності програми як наслідку фінансової поваги розробника та зацікавлені в максимальній технологічній якості, економічних та технологічних інтересах проекту.

За результатами аналізу всіх зацікавлених сторін було обрано ключових споживачів, які найкраще відповідають основним параметрам, визначеним у попередньому розділі.

5.4 Визначення команди, та життєвого циклу проекту

Команда складається з одної людини, яка є ініціатором та розробником проекту. На основі цього була створена схема життєвого циклу проекту (рисунок 5.1).

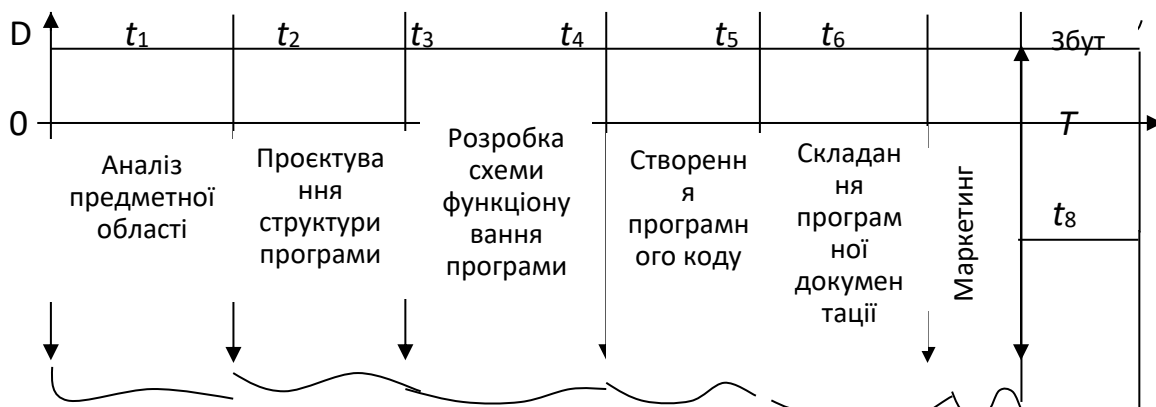


Рисунок 5.1 – Схема життєвого циклу проекту

Умовні позначки

Тривалість життєвого циклу проекту вираховується за формулою(5.1):

$$T = t_1 + t_2 + t_3 + t_4 + t_5 + t_6 \quad (5.1)$$

де t – це тривалість віхи (етапу) життєвого циклу інноваційного проекту,

T – це тривалість життєвого циклу інноваційного проєкту;

t_1 – це аналіз предметної області = 3 дні;

t_2 – це проєктування структури застосунка = 7 днів;

t_3 – це розробка схеми функціонування застосунка = 6 днів;

t_4 – це створення програмного коду = 7 днів;

t_5 – це складання програмної документації = 4 днів;

t_6 – це маркетинг = 3 дні.

Після підстановки всіх величин тривалості окремої віхи, отримуємо тривалість всього життєвого циклу:

$$3 + 7 + 6 + 7 + 4 + 3 = 30 \text{ днів.}$$

5.5 Визначення витрат на розробку

5.5.1 Розрахунок основної та додаткової заробітної плати

Проводиться розрахунок основної та додаткової заробітної плати, який наведено у таблиці 5.8. Вона заснована на місячному окладі працівників та тривалості, яка біла відпрацьована за планом.

Таблиця 5.8 – Склад, чисельність та фонд заробітної плати виробничих працівників

Категорії працівників	Наявна чисельність, осіб		Тарифна ставка за розрядом виконуваних робіт, грн / годину	Ефективний фонд робочого часу, годин	Тарифний заробіток, грн.	Преміальний відсоток до тарифного заробітку	Розмір премії, грн.	Річний фонд заробітної плати, грн.	ЄСВ, грн.
	за	на добу							
1	2	3	4	5	6	7	8	9	10
Виробничі працівники, в тому числі основний працівник	1	1	35	160	6000	10	600	6000	1220
Разом виробничих працівників	1	1	x	x	x	x	600	6000	1220

Додаткова заробітна плата приймається рівною 10% від основної заробітної плати працівника і розраховується за формулою (5.2):

$$ЗП_{\text{дод}} = ЗП_{\text{осн}} \cdot 0,1. \quad (5.2)$$

Після підстановки величини основної заробітної плати в дану формулу, отримується:

$$ЗП_{\text{дод}} = 6000 \cdot 0,1 = 600 \text{ грн.}$$

Відрахування на єдиний соціальний внесок становлять 22% і беруться від основної та додаткової заробітної плати. Вони розраховуються за формулою (5.3):

$$\text{ВІД} = (\text{ЗП}_{\text{осн}} + \text{ЗП}_{\text{дод}}) \cdot 0,22, \quad (5.3)$$

де $\text{ЗП}_{\text{осн}}$ – основна заробітна плата (грн.);

$\text{ЗП}_{\text{дод}}$ – додаткова заробітна плата (грн.).

Розрахуємо:

$$\text{ВІД} = (6000 + 600) \cdot 0,22 = 1452 \text{ грн.}$$

Таким чином отримано, що додаткові соціальні відрахування становлять 1452 грн.

5.5.2 Визначення витрат на матеріали

Витрати на матеріали розраховуються за формулою (5.4):

$$M = \sum_{i=1}^n (\text{Ц}_i N_i (1 + K_{\text{т.з.}}) - \text{Ц}_{io} N_{io}), \quad (5.4)$$

де M – витрати на матеріали, покупні напівфабрикати та комплектуючі(грн.);

$K_{\text{т.з.}}$ – коефіцієнт, що враховує транспортно-заготівельні витрати;

Ц_i – ціна i -го найменування чи комплектуючого (грн.);

N_i – потреба в i -му матеріалі, напівфабрикаті, комплектуючих;

Ц_{io} – ціна зворотних відходів i -го найменування матеріалу (грн.);

N_{io} – кількість зворотних відходів i -го найменування.

n – кількість найменувань матеріалів, напівфабрикатів та комплектуючих.

Додатковими матеріалами, необхідними для розробки, є папір для оформлення документації, який коштує 100 грн.

5.5.3 Витрати на обладнання

Цей пункт покриває витрати на придбання, доставку, встановлення та налаштування нестандартного обладнання. Також необхідно розрахувати витрати на оплату часу роботи комп'ютерної машини (амортизаційні витрати на обслуговування та вартість спожитої електроенергії).

Розрахуємо витрати на обслуговування ПК. Амортизаційні відрахування розраховуються за формулою (5.5):

$$A = \Phi_B \frac{H_a}{100}, \quad (5.5)$$

де Φ_B – балансова вартість обчислювальної техніки (грн.);

H_a – норма амортизаційних відрахувань на повне відновлення обчислювальної техніки (%).

$$A = 13200 \cdot 0,25 = 3300 \text{ грн.}$$

Амортизаційні відрахування становлять 3300 грн.

Для розрахунку вартості експлуатації обладнання використаємо формулу (5.6):

$$C_e = N_n \cdot \Phi_{\text{еф}} \cdot K_{\text{зч}} \cdot K_{\text{зп}} \cdot C_e, \quad (5.6)$$

де N_n – номінальна потужність ПК, кВт;

$\Phi_{\text{еф}}$ – річний ефективний фонд часу роботи ЕОМ;

$K_{\text{зч}}$ – середній коефіцієнт завантаження за часом;

$K_{\text{зп}}$ – коефіцієнт завантаження за потужністю;

C_e – ціна одного кВт • год електроенергії, грн/(кВт • год.).

Річний фонд часу роботи ПК становить 1000 годин. Середні коефіцієнти завантаження за часом та за потужністю становлять 0.6 та 0.9.

$$C_9 = 0,2 \cdot 1000 \cdot 0,9 \cdot 0,6 \cdot 2,11 = 227,88 \text{ грн.}$$

Стаття «Поточний ремонт обладнання» приймається рівною 3% від балансової вартості обладнання і складає 380 грн.

Стаття «Інші витрати» відповідає п'яти відсоткам загальних попередніх витрат, пов'язаних з обслуговуванням та експлуатацією обладнання. Сума всіх попередніх статей становить 3777,9 грн., 5% від суми становлять 188,89 грн.

Кошторис витрат на утримання та експлуатацію обладнання наведено у таблиці 5.9.

Таблиця 5.9 – Кошторис витрат на утримання та експлуатацію обладнання

Найменування статей витрат	Сума, грн.
Амортизація обладнання	3300
Експлуатація обладнання (окрім витрат на поточний ремонт)	227,88
Поточний ремонт обладнання	380
Інші витрати	190,54
Разом	4098,42

Витрати на оплату машинного часу ПК для написання й відлагодження застосунка вираховується за формулою (5.7):

$$C_{mc} = P_{екс} t_{мч}, \quad (5.7)$$

де C_{mc} – витрати на сплату машинного часу (грн);

$P_{екс}$ – експлуатаційні витрати на один час машинного часу ПК;

$t_{мч}$ – машинний час ПК для написання та відлагодження програмного засобу.

Розрахуємо:

$$P_{\text{екс}} = \frac{3865,98}{900} = 4,09 \text{ грн/машино-час.}$$

ПК експлуатується 30 днів в одну зміну, що становить у сумі 240 годин. Таким чином витрати на сплату машинного часу становлять:

$$C_{\text{мс}} = 4,09 \cdot 240 = 981,6 \text{ грн.}$$

5.5.4 Накладні витрати

Накладні витрати - це витрати на загальне управління, загальні економічні потреби (зарплата виконавчого персоналу, витрати на офіс тощо), утримання та експлуатацію будівель. Накладні витрати опосередковано включаються у вартість розробки додатків як відсоток від базової заробітної плати розробників. При цьому накладні витрати становлять 40% від базової зарплати забудовників, що становить 2000 грн.

Потім розраховується прибутковість програмного забезпечення, яка становить 30% від суми минулих витрат та ПДВ, 20% від загальної суми витрат, враховуючи прибутковість.

Результати визначення витрат на розробку застосунка приведені в таблиці 5.10.

Таблиця 5.10 – Вартість розробки застосунка

Стаття витрат	Сума (грн.)	Питома вага до підсумку (%)
Основна заробітна плата	6000	33,35
Додаткова заробітна плата	600	3,33
ЄСВ	1452	8,07
Вартість матеріалів	100	0,56
Витрати на обладнання	981,6	5,46
Накладні витрати	2400	13,34
Разом	11533,6	64,1
Рентабельність	3460,08	19,23
Разом	14993,68	83,33
ПДВ	2998,736	16,67
Всього	17993,416	100

5.6 Розрахунок економічної ефективності програмного продукту

В разі створення програмної системи економічний ефект визначається за формулою (5.8):

$$\mathcal{E}_{\phi} = \mathcal{E}_{\Gamma} - E_{\text{н}} \cdot K, \quad (5.8)$$

де $\mathcal{E}_{\phi}$ – річна економія поточних витрат, грн.;

K – капітальні витрати на створення програмного виробу, грн.;

$E_{\text{н}}$ – нормативний коефіцієнт економічної ефективності капіталовкладень, долі;

$E_{\text{н}}$ залежить від особливостей вживання засобів автоматизації в різних галузях; для даного випадку він дорівнює 0,6.

Розрахуємо $\mathcal{E}_{\phi}$:

$$\mathcal{E}_{\phi} = 17993,416 - 19506,58 \cdot 0,6 = 6\,289,46 \text{ грн.}$$

Коефіцієнт економічної ефективності капіталовкладень показує величину річного приросту прибутку. Коефіцієнт економічної ефективності капіталовкладень розраховується за формулою (5.9):

$$E_p = \mathcal{E}_T / K, \quad (5.9)$$

Розрахуємо:

$$E_p = 17993,416/19506,58 = 0,92$$

Розроблена програма є економічно ефективною, оскільки виконується нерівність:

$$E_p \geq E_n,$$

$$0,92 \geq 0,6.$$

Термін окупності капіталовкладень – період часу, протягом якого окупаються витрати на ПМК. Знаходимо за формулою (5.10):

$$T_p = \frac{1}{E_p}, \quad (5.10)$$

Отримаємо, $T_p = 1,08 < T_n = 2,4$ року (термін окупності капіталовкладень менше нормативного), то можна стверджувати, що капіталовкладення використовуються ефективно. Техніко-економічні показники наведено на таблиці 5.11.

Таблиця 5.11 – Техніко-економічні показники

Показники	Сума	Одиниця виміру
Витрати на розробку програми	17993,416	грн.
Річна економія	6 289,46	грн.
Строк окупності програми	1,08	рік

Таким чином, впровадження розробленого програмного забезпечення на виробничому заводі дозволить зменшити кількість працівників, відповідальних за обробку великих масивів даних, з щомісячною зарплатою 6000 гривень. Економічні розрахунки показали, що термін окупності програми становить 1,08 року, що набагато менше нормативного періоду окупності (5 років). Крім того, щорічна економія при використанні програми становить 6289,46 грн. На підставі розрахунків можна зробити висновок, що розробка програмного продукту є економічно доцільною.

5.7 Оцінка економічної ефективності

Створення програми спрямоване на отримання більших доходів, залучення нових клієнтів та збільшення обсягу послуг.

Вартість розробки додатків становить 17 993 416 гривень. Очікується продати 500 примірників за мінімальною ціною 150 грн / примірник. Враховуючи 50% ризик, запланований дохід від реалізації складе 37 500 грн. Враховуючи, що доходи, враховуючи ризик, перевищують ціну програмного забезпечення (на 19 506,58 гривень), що вже врахувало рентабельність (30%), можна зробити висновок, що розробка цього програмного забезпечення є економічно доцільною.

Є перспективи для реалізації щодо потенційних груп користувачів. Вхідні бар'єри відносно високі через високу конкуренцію майже у всіх віках, але застосування стратегії диференціації привертає більше користувачів. Конкурентоспроможність проекту висока.

Доцільно обирати реалізацію з частковим виконанням функцій проекту та поступовим впровадженням нових.

Рекомендується подальша реалізація проекту.

6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА У НАДЗВИЧАЙНИХ СИТУАЦІЯХ

Дисертація присвячена темі «Дослідження та програмна реалізація бота-консультанта для персонального підбору комплексу вітамінів», ця робота включає наявність робочого місця в лабораторії програмування.

Обладнання, що використовується в робочому процесі, можна розділити на дві групи:

- обладнання, що безпосередньо підтримує робочий процес (комп'ютер та різноманітна периферія, якщо працює в лабораторії програмування);
- обладнання для підтримки комфортних умов праці (обігрівачі, кондиціонери, іонізатори повітря, що забезпечують оптимальні параметри мікроклімату приміщення).

6.1 Аналіз потенційних небезпек

Потенційні ризики роботи в офісі:

- порушення опорно-рухового апарату внаслідок тривалих статичних навантажень при роботі з ПК;
- нервово-психічні перевантаження внаслідок постійних контактів з клієнтами, колегами, керівництвом, що може бути суперечливим і викликати емоційний дискомфорт, внутрішнє роздратування, емоційну нестабільність та захворювання нервової системи;
- неадекватні ергономічні особливості робочого місця через нераціональне оформлення робочого місця, що може призвести до механічних травм, уражень електричним струмом та розладів опорно-рухового апарату;
- негативний вплив недостатнього освітлення робочої зони на зір і продуктивність праці працівника через несправність світильників або неправильну конструкцію системи освітлення;
- негативний вплив незадовільних параметрів повітря в робочій зоні

на здоров'я працівника через неправильну конструкцію або несправність вентиляційної системи;

- негативний вплив підвищеного рівня шуму на психоемоційний стан працівника, який пов'язаний із використанням застарілої периферії, кондиціонерів, ксероксів, світильників;
- ризик пожежі внаслідок виходу з ладу електрообладнання, недотримання правил протипожежного захисту або порушення правил протипожежного захисту, що може призвести до пожежі;
- неправильні дії персоналу в надзвичайних ситуаціях.

6.2 Заходи з забезпечення безпеки

Статус розробки програмного забезпечення згідно "Правил розташування електричного обладнання" (ПУЕ) відноситься до категорії відсутність підвищеного ризику ураження електричним струмом. Площа приміщення - 10,91 м².

Обладнання в приміщенні споживає 220 В змінного струму від заземленої нейтральної мережі і застосовується до закритого електрообладнання до 1000 В. За способом захисту людини від ураження електричним струмом відповідає згідно з ГОСТ 12.2.007.0-75* (2001) «ССБТ. Изделия электротехнические. Общие требования безопасности» I (стаціонарні комп'ютери,) та II (освітлювальні прилади, кондиціонери, опалювальні пристрої, ноутбуки, сканери) класу захисту [32].

Відповідно до «Правил улаштування електрообладнання» («ПУЕ») були реалізовані наступні групи заходів з електробезпеки.

Конструктивні заходи забезпечують захист від випадкового контакту з живими частинами через їх ізоляцію та захисну оболонку. Згідно з ГОСТ 12.1.009-76 (1999) «ССБТ. Электробезопасность. Термины и определения» у приладах II класу захисту використовується подвійна ізоляція - електрична ізоляція, що складається з робочої і додаткової ізоляції.

Згідно з НПАОП 40.1-1.32-01 «Правила устройства электроустановок. Электрооборудование специальных установок» кімнатні приміщення у більшості своїй відносяться до класу пожежебезпечної зони.

П-Па (приміщення, в якому входять тверді речовини, що заготовляють речовини), тому передбачена ступінь захисту ізоляції

Шемно-конструктивні заходи призначені для забезпечення захисту від ураження електричним струмом при дотику до металевих оболонок, які можуть описувати під напругою в результаті аварії [33].

Згідно з ГОСТ 12.1.030-81 (2001) «ССБТ. Электробезопасность. Защитное заземление, зануление» у приміщенні де я працював, було влаштоване занулення.

Експлуатація електроустановок і електроустаткування проводиться відповідно до НПАОП 40.1-1.01-97 «Правила безпечної експлуатації електроустановок» (далі «ПБЕЕ») та НПАОП 40.1-1.21-98 «Правила безпечної експлуатації електроустановок споживачів» (далі «ПБЕЕС»).

Щоб запобігти статистичному навантаженню, під час використання ПК робили перерви на 15 хвилин. кожні дві години. Зап'ястково-тунельний синдром, або тунельний зап'ястково-зап'ястковий синдром, який може бути наслідком хронічної травми, виникає у людини в результаті тривалої роботи з мишею: це призводить до постійних стресів і дискомфорту, мікротравм, здавлення нервів навколишніх тканин і набряки [33].

Щоб мене не турбував тунельний синдром, я дотримувався деяких правил щодо організації робочого місця:

- висота клавіатури від підлоги - 60-70 см;
- ергономічна та зручна наявність миші та клавіатури;
- регулюйте нахил і висоту клавіатури (відстань від поверхні столу до центру клавіатури - до 35 мм, кут нахилу клавіатури - від 2 градусів до 17 градусів);
- підставка для клавіатури для рук;
- використання подушечки для миші із захистом від тунельного

синдрому (спеціальний виступ забезпечує правильне положення кисті);

- наявність стільця з підлокітниками.

Працюючи з мишею та клавіатурою, я дотримувався цих правил. Коли я набирав текст, кисть була зігнута під прямим кутом (90 градусів) в лікті, а під час роботи з мишею кисть була прямою і якомога далі від краю. Комп'ютерний час обмежувався тим, що було насправді [32-36].

Щоб запобігти тунельному синдрому, якомога частіше виконували спеціальні вправи для рук. Ці вправи допомогли поліпшити і циркулювати м'язи в крові. Серію вправ повторювали приблизно кожні 40 хвилин, тривалість однієї вправи становила 1-3 хвилини.

Нервова напруга впливає на серцево-судинну систему, підвищуючи артеріальний тиск і частоту серцевих скорочень, а також на терморегуляцію та емоційний стан організму. Організація робочого місця, правильне робоче положення, ритм роботи, оптимізація робочого процесу, використання емоційних стимулів, впровадження раціональних способів праці та відпочинку тощо відіграли особливу роль у запобіганні втоми. Боротьба з втомою полягає насамперед у покращенні оздоровчих та гігієнічних умов виробничого середовища (усунення забруднення повітря, шуму, вібрації, нормалізація мікроклімату, раціональне освітлення тощо).

Для профілактики втоми використовувались специфічні методи, які включають відновлення функціонального стану зорово-рухової системи, зменшення гіподинамії, посилення мозкового кровообігу та оптимізацію розумової діяльності [34 - 35].

Загальні ергономічні вимоги встановлено ДСТУ ISO 9241-1:2003. «Ергономічні вимоги до роботи з відеотерміналами в приміщенні. Частина 1. Загальні положення». Організація робочого місця включає: правильне розміщення робочого місця у виробничому приміщенні; ергономічно виправдане робоче положення, вибір виробничих меблів з урахуванням антропометричних характеристик людини; розумне розташування обладнання на робочому місці; з урахуванням характеру та особливостей зайнятості [35].

«Правила охорони праці під час роботи електронно-обчислювальних машин» Рекомендовані розміри верстака з ВДТ для установки: висота - 725 мм, ширина - 600-1400 мм, глибина - 800-1000 мм. Розмір смуги становить близько трьох міліметрів: видимість - 725 мм, чорна - 1200 мм, гладіолуси - 800 мм.

6.3 Заходи з виробничої санітарії та гігієни праці

Згідно ДБН В.2.5-28:2018 «Природне і штучне освітлення» [37-38] в приміщенні використовувалось природне та штучне освітлення. Природне світло забезпечувалось світловими отворами, які забезпечували коефіцієнт природного освітлення (КПО) щонайменше 1,5%. Проти прямих сонячних променів, які спричиняють прямі та відбивні відбиття на поверхнях екранів та клавіатур, використовували сонцезахисні пристрої, а на вікнах розміщували жалюзі. Штучне освітлення в кімнаті забезпечується системою загального освітлення. Величина освітленості на поверхні робочого столу в області документів становить 300-500 люкс [31].

Показники мікроклімату приміщення відповідають встановленим санітарно-гігієнічним вимогам ДСН 3.3.6-042-99 «Санітарні норми мікроклімату виробничих приміщень», ГОСТ 12.1.005-88 (1991) [39-40].

«ССБТ. Общие санитарно-гигиенические требования к воздуху рабочей зоны» і ГН 2152-80 «Санітарно-гігієнічні норми допустимих рівнів іонізації повітря виробничих та громадських приміщень».

Були встановлені наступні оптимальні значення параметрів мікроклімату (табл. 6.1):

- в холодну пору року: температура 22-24 °С; відносна вологість: 40-60%; швидкість повітря: 0,2 м / с;
- у теплу пору року: температура 23 °С; відносна вологість: 40-60%; швидкість повітря: 0,2 м / с.

Пора року	Температура, °C	Відносна вологість, %	Швидкість руху повітря, м/с
Холодна	22-24	40-60	0,2
Тепла	23-25	40-60	0,2

Таблиця 6.1 - Оптимальні значення параметрів мікроклімату

Рівні оптимальних позитивних ($n +$) та негативних ($n -$) іонів у повітрі приміщення нормалізували за допомогою ПК відповідно до ГН 2152-80 "Стандарти здоров'я та гігієни для допустимих рівнів іонізації промислового та громадського повітря". становили: $n + = 1500-30000$ (шт. на 1см^3); $n - = 3000-5000$ (шт на 1см^3). Підтримка оптимальних рівнів світлових позитивних та негативних аероіонів у повітрі на робочому місці забезпечується за допомогою біполярних іонізаторів повітря з короною [38].

Відповідність вимогам цих документів було досягнуто шляхом забезпечення приміщення кондиціонуванням та вентиляцією, дезодорацією, опаленням відповідно до вимог ДБН В.2.5-67: 2013 «Опалення, вентиляція та кондиціонування повітря».

Рівні звукового тиску в октавних діапазонах, рівні шуму та еквівалентні рівні шуму на робочому місці кімнати нормуються згідно ДСанПіН 3.3.2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» та ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку».

"Нормативи охорони здоров'я для промислового шуму, ультразвуку та інфразвуку".

Рівень шуму в кімнаті знижується на:

- використання найсучаснішого обладнання;
- перевести жорсткий диск у режим сну (режим очікування), якщо комп'ютер деякий час не працює;
- використання охолоджених джерел живлення для гумових

підвісок.

6.4 Заходи безпеки у надзвичайних ситуаціях

6.4.1 Заходи з пожежної безпеки

Приміщення лабораторії програмування, де проводяться роботи, класифікуються за класом пожежі А - переважно тверді речовини органічного походження, горіння яких супроводжується тлінням (дерев'яні меблі, документація).

Крім визначених ГОСТ 27331-87 класів пожеж, згідно НАПБ Б.03.001-2004 «Про затвердження Типових норм належності вогнегасників» офісні приміщення також належать до класу горіння вогню.

В офісних приміщеннях класу пожежі класу А (Е) категорія приміщень забезпечує основний засіб гасіння пожежі, а саме встановлення двох вогнегасників з вуглекислим газом ємністю 5 літрів.

Лабораторні приміщення належать до адміністративних приміщень, тому з метою ліквідації пожежі на кожному поверсі, щонайменше, два переносні (пил, вода, вода - хай буде - слід зазначити, що використання води та водяних вогнегасників заборонено через класифікацію лабораторних кабінетів у цьому класі протипожежного захисту).) з вакциною щонайменше 5 кг або більше.

Таким чином, присутні два вуглекислотні вогнегасники ВВК-3,5 (ВВК-2) та 2 порошкові вогнегасники ВП-5, що зберігаються у приміщенні відповідно до ДСТУ 3734-98.

— Всі наявні електроприлади та вогнегасники у справному стані, систематично перевіряються на наявність несправностей, а в приміщенні для повідомлення про пожежу встановлені автоматична пожежна сигналізація та автоматична система пожежогасіння, до якої завжди мають доступ усі працівники.

Крім того, застосовуються такі заходи протипожежного захисту:

- підлога комп'ютерної підлоги та облицювання стелі та стін кімнати виконано згідно ДБН В. 1.1-7-2002 із матеріалів групи Г1 (низької горючості);
- наявні евакуаційні виходи з приміщення та інструкції щодо організованої евакуації з приміщення, журнал інструкцій;
- автоматична система пожежної сигналізації (АПС) використовується для виявлення початкової стадії пожежі та оповіщення пожежної охорони; працівники підприємства можуть експлуатувати сигналізацію самостійно, якщо пожежа ще не досягла значних розмірів;
- куріння заборонено на робочих місцях, а на території є спеціальна зона для паління з урнами для недопалків з написом «Місце для паління»;
- папір, що використовується для роботи, тримається на достатній відстані від електричних ламп та електрощитів, а побутові балони, що знаходяться під тиском, зберігаються в шафах, де сонце не уникне запалення [41-42].

Усі працівники можуть користуватися засобами індивідуального захисту, первинними засобами пожежогасіння та інформуються про свої обов'язки у разі надзвичайної ситуації.

Цей пакет організаційно-технічних заходів щодо запобігання пожежам дає можливість запобігати пожежі та, якщо вона виникає, забезпечувати безпеку людей, зупиняти розповсюдження вогню та створювати умови для успішного пожежогасіння.

6.4.2 Заходи з цивільного захисту

Промислові будівлі та приміщення можна розділити на 5 категорій залежно від розміщеного в них виробництва та вибухонебезпечних властивостей матеріалів на них: А, В, С, D, D [42-43].

Приміщення, в якому проводились науково-дослідні роботи, відноситься до категорії D - негорючі матеріали та матеріали в холодному стані.

З метою характеристики умов експлуатації електрообладнання та правильного вибору електрообладнання, правила облаштування електрообладнання визначають класифікацію зон вибуху та пожежі.

Для запобігання короткого замикання та пожеж із перевантаженням слід належним чином вибрати, встановити та дотримуватись розробленого режиму роботи електричних мереж, дисплеїв та іншої електричної автоматики.

Тому в технічному, оперативному, організаційному плані необхідно передбачити ряд профілактичних заходів.

План евакуації з (будівельного) поверху розміщується на видному місці на кожному поверсі будівлі. У плані евакуації крім виходів (стрілки) вказано місце розташування вогнегасників та телефонів.

Щоб запобігти пожежі, слід дотримуватись наступних правил протипожежного захисту:

- виключення горючого середовища (відключення обладнання, контроль повітряного середовища, робота та аварійна вентиляція);
- використання при будівництві та обробці будівель, які не горять або не містять антипіренів [36].

У офісі необхідно вжити таких заходів протипожежного захисту.

Організаційна діяльність:

- навчання гасіння пожеж для обслуговуючого персоналу;
- навчання персоналу правилам безпеки;
- публікація інструкцій, плакатів, планів евакуації.

Оперативні заходи:

- відповідність нормам експлуатації обладнання;
- забезпечення вільного доступу до обладнання.
- зручність використання ізоляції провідних провідників.

Технічні заходи:

- дотримання протипожежних заходів при влаштуванні електропроводки, обладнання, систем опалення, вентиляції та освітлення. В аудиторії є вогнегасник типу ОУ-5, встановлений автоматичний вимикач, вся кімната знеструмлена, на дверях аудиторії на випадок пожежі видно план евакуації, а також є пожежний щит. Якщо спалах спрацював в електрообладнанні, для його усунення слід використовувати вуглекислий газ або порошковий вогнегасник;
- профілактичний огляд, ремонт та випробування обладнання [37].

Окрім ліквідації джерела пожежі, евакуація людей повинна бути організована вчасно.

Комфорт та безпечні умови праці є головним фактором, що впливає на продуктивність людей, які беруть участь у розробці програмного забезпечення. У робочому кабінеті умови праці характеризуються можливістю впливу на працівників таких факторів виробництва: шуму, психофізіологічних факторів, впливу мікроклімату, параметрів технологічного обладнання та робочого місця. У цьому розділі аналізуються шкідливі та небезпечні фактори, що впливають на здоров'я людини, основні заходи щодо їх запобігання та усунення. Відзначено джерела негативного впливу, колективні та індивідуальні заходи захисту.

ВИСНОВКИ

Чат-боти захопили світ. Їх успішно використовують в самих різних бізнес-нішах, тим самим дозволяючи економити і вирішувати величезну кількість задач. За даними Oracle 36% компаній вже використовують цю технологію, а 44% планують впровадити її в найближчому році.

Спілкування з чат-ботом - це імітація спілкування з живою людиною. Ви ведете діалог, задаєте питання і отримуєте відповіді, відчуваючи, що по той бік екрану не машина, а реальний співрозмовник.

В ході виконання магістерського проєкту було розглянуто різні типи ботів і їх функціонал. Було впроваджено аналіз їх функціоналу, інтерфейсу і систем безпеки. Виходячи з вироблених аналізів, розроблений бот, який дозволяє автоматизувати процес збору персональних даних. Бот працює з відкритими джерелами, як відкритих джерел маються на увазі соціальні мережі. Потрібно відзначити те, що для роботи з ботом користувач зобов'язаний отримати дозвіл на збір інформації шуканого людини.

Проаналізовано вже існуючих ботів і приведено приклади їх використання. Як результат, виникла необхідність сприяти використанню розпізнавання команд з повідомлень користувачів, коли боти використовуються як засіб взаємодії з клієнтами. Алгоритм вибору засобів розробки формалізований.

Для розробки була використана мова програмування Python. Дана мова була обрана, оскільки містить всі необхідні бібліотеки для роботи з Telegram ботами.

В результаті виконаної роботи було проведено аналіз методів персональних рекомендацій та виділено з них ефективні та відповідні задачі роботи.

Метою роботи є розробка бота, що дозволяє автоматизувати процес збору інформації.

Однією із головних задач було дослідження методів машинного навчання при створенні чат-бота та розробка моделі HEALTH CARE чат-бота на основі розпізнавання кнопок меню та контексту.

Перевагою теми автора полягає в тому, що розроблена автором програма об'єднує в собі безліч позитивних критерій, які були виявлені в процесі формування різних спамерських та пошукових роботів, це дозволило розробити якісного, надійного, корисного і безпечного з точки зору інформаційної безпеки бота. Бот включає в себе зручний і інтуїтивно зрозумілий інтерфейс.

Весь чат-бот реалізований з використанням поділу одного великого завдання на більш маленькі, це дозволило з легкістю змінювати функціонал бота, що не переписуючи його повністю. Це так само дозволяє впроваджувати в код бота будь нові модулі без численних змін.

Чат-бот було розроблено за каркасної архітектурою, що дозволяє не легше швидко та легко розширювати базовий функціонал і доповнювати його завданнями інших типів в рамках роботи чат-боту.

ПЕРЕЛІК ПОСИЛАНЬ

1. 10 Best Chatbot Development Frameworks to Build Powerful Bots [Electronic resource]. / Chatbot Report 2018: Global Trends and Analysis – Access mode: <https://geekflare.com/chatbot-development-frameworks/>
2. Chatbot Report 2018: Global Trends and Analysis [Electronic resource]. – Access mode: <https://chatbotsmagazine.com/chatbot-report-2018-global-trends-and-analysis-4d8bbe4d924b>
3. Рейтинг мессенджеров. Сервис сравнения и выбора приложений и сервисов в сфере IT. [Электрон. ресурс] – Режим доступа: <https://coba.tools/compilation/reiting-messendzherov-2018>
4. Чат-бот - что это такое, как работает, виды, преимущества и недостатки, как создать [Электрон. ресурс] – Режим доступа: <http://kak-bog.ru/chat-bot-chto-eto-takoe-kak-rabotaet-vidy-preimushchestva-i-nedostatki-kak-sozdat>
5. Все о чат-ботах: типы и примеры, какому бизнесу подойдет, список конструкторов для создания [Электрон. ресурс] – Режим доступа: <https://web-promo.ua/blog/vse-o-chat-botah-tipy-i-primery-kakomu-biznesu-podojdet-spisok-konstruktorov-dlya-sozdaniya/>
6. Чат-боты: хайп или эффективный бизнес инструмент [Электрон. ресурс] – Режим доступа: <https://jam.in.ua/blog/chat-boty-khajp-abo-efektyvnyj-biznes-instrument/>
7. Что такое боты и как они работают [Электрон. ресурс] – Режим доступа: <https://ru.epicstars.com/boty-i-telegram/>
8. Виды ботов [Электрон. ресурс] – Режим доступа: <https://psycho.ru/library/3791>
9. Hackernoon.com [Electronic resource]. – Access mode: <https://hackernoon.com/chatbots-in-2021-what-you-need-to-know-7f123482>
10. Анализ безопасности Telegram [Электрон. ресурс] – Режим доступа: <https://www.securitylab.ru/analytics/490726.php>

11. Telegram FAQ [Electronic resource]. – Access mode: <https://telegram.org/faq>
12. Бийбосунов Б. И. Описание концепции Telegram ботов и их разработка [Текст] / Б. И. Бийбосунов, С. К. Бийбосунова, Н. Ж. Жолочубеков // Colloquium-journal. – 2020. – №7. – С. 1–4.
13. Положий Г. А. Сравнительный анализ систем мгновенного обмена сообщениями [Текст] / Г. А. Положий А. Р. Тосунова, О. А. Сафарьян, Л. В. Черкесова // Молодой исследователь Дона/ – 2020. – №4. – С. 59–63.
14. Витамины от А до Я [Электронный ресурс] – Режим доступа: <https://www.kp.ru/putevoditel/zdorove/vitaminy/>
15. Как правильно выбрать витаминный комплекс [Электрон. ресурс] – Режим доступа: <http://mpgu.su/novosti/kakie-vitaminy-luchshe-prinimat-dlja-immuniteta-vzroslym/>
16. Витамины – незаменимые питательные микроэлементы. [Электрон. ресурс] – Режим доступа: <https://toitumine.ee/ru/energiya-i-potrebnost-v-pitatelnyh-veshhestvah/vitaminy>
17. Витаминный минимум: без каких важных веществ нам не обойтись [Электронный ресурс] – Режим доступа: <https://www.championat.com/lifestyle/article-4099551-kakie-vitaminy-neobhodimy-cheloveku-sovety-nutriciologa.html>
18. Прохоренок Н. А. Python 3. Самое необходимое [Текст] / Н.А. Прохоренок, В.А. Дронов. – Санкт-Петербург: БХВ, 2016 – 442 с.
19. Официальный сайт Python [Electronic resource]. – Access mode: <https://www.python.org/>
20. Welcome to Python Telegram Bot’s documentation [Electronic resource]. – Access mode: <https://python-telegram-bot.readthedocs.io/en/stable>
21. Python Requests [Electronic resource]. – Access mode: <https://2.python-requests.org/en/master/>
22. Самоучитель Python [Электрон. ресурс] – Режим доступа: <https://pythonworld.ru/samouchitel-python>

23. Telegram APIs [Electronic resource]. – Access mode: <https://core.telegram.org/api>
24. Общая информация про Telegram [Электрон. ресурс] – Режим доступа: <https://en.wikipedia.org/wiki/Telegram>
25. Створюємо Telegram бота на Python. Частина 1 [Електронний ресурс] - Режим доступу: <https://codeguida.com/post/410>
26. Learn to build your first bot in Telegram with Python [Electronic resource]. – Access mode: <https://medium.com/free-code-camp/learn-to-build-your-first-bot-in-telegram-with-python-4c99526765e4>
27. Инструкция: Как создавать ботов в Telegram [Электрон. ресурс] - Режим доступа: <https://habr.com/ru/post/262247>
28. Простой Telegram-бот на Python за 30 минут [Электрон. ресурс] – Режим доступа: <https://habr.com/ru/post/442800/>
29. Телеграм бот за допомогою TeleBot [Электрон. ресурс] – Режим доступа: <https://habr.com/ru/post/448310/>
30. Тестування телеграм-бота [Электрон. ресурс] – Режим доступа: <https://habr.com/ru/post/322816/>
31. Остапенко, В.В. Методичні вказівки по економічному обґрунтуванню дипломних проєктів для студентів спеціальностей 7.080403 «Програмне забезпечення автоматизованих систем», 7.080402 «Інформаційні технології проєктування» і 7.091501 «Комп'ютерні та інтелектуальні системи і мережі» всіх форм навчання [Текст] / В.В. Остапенко, Е.М. Касьян, Є.М. Сердюк. – Запоріжжя : ЗНТУ, 2009. – 22 с.
32. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями. НПАОП 0.00-7.15-18. Державні санітарні правила і норми [Текст]. – Харків : Форт, 2018. – 48 с.
33. Шмирко, В.І. Методичні вказівки до лабораторного заняття «Розрахунок загального рівномірного штучного освітлення виробничих приміщень» з дисципліни «Цивільний захист і охорона праці в галузі» [Текст] : для студентів усіх спеціальностей та форм навчання / В.І. Шмирко, О.В.

Коробко, Ю.І. Троян. – Запоріжжя : каф. ОПІНС. НУ «Запорізька політехніка», 2020. – 36 с.

34. ДСТУ 8604:2015 Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги [Текст]. – Київ : Держспоживстандарт України, 2015. – 15 с.

35. Правила улаштування електроустановок [Текст] : ПУЕ-2017. – На заміну ПУЕ-86 ; чинний з 2017-08-21. – К. : Міненерговугілля України, 2017. – 617 с.

36. ДСТУ 7237:2011 Система стандартів безпеки праці. Електробезпека. Загальні вимоги та номенклатура видів захисту [Текст]. – Київ : Держспоживстандарт України, 2011. – 10 с.

37. Охорона праці в галузі. Навчальний посібник [Текст] / П. Атаманчук, В. Мендерецький, О. Панчук, Р. Билык. – К. : Центр навчальної літератури, 2017. – 322 с.

38. ДБН В.2.5-28:2018. Природне та штучне освітлення. Посібник [Текст] : проблеми природного і штучного освітлення. – К. : Етін, 2019. – 180 с.

39. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень [Текст]. – Харків : Форт, 2000. – 24 с.

40. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку [Текст]. – К. : МОЗ України, 1999. – 24 с.

41. ДСТУ Б В.1.1-36:2016 Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою [Текст]. – Київ : Мінрегіонбуд України, 2016. – 66 с.

42. ДБН В.1.1-7:2016 Пожежна безпека об'єктів будівництва. Загальні вимоги [Текст]. – Київ : Міністерство регіонального розвитку, будівництва та житлово-комунального господарства України, 2017. – 39 с.

43. Стеблюк, М. І. Цивільна оборона та цивільний захист [Текст] : навч. посіб. для вузів / М. І. Стеблюк. – К. : Знання, 2013. – 487 с.

ДОДАТОК А
ТЕХНІЧНЕ ЗАВДАННЯ

A.1 Вступ

Актуальність теми полягає в необхідності описати загальні принципи створення роботів для сучасних месенджерів, оскільки вони широко використовуються різними компаніями для розвитку свого бізнесу. Особливо це стосується українського ринку, оскільки компанії лише починають освоювати цю технологію, тож попит на професіоналів для розвитку таких послуг швидко зростає.

Сучасна тенденція полягає в автоматизації всіх можливих процесів. Для розробленого продукту це працює в першу чергу. Потрібно мати можливість користуватися послугами якомога швидше, незалежно від людського фактора, за який згодом платять гроші.

A.2 Підстави для розробки

Підставою для розробки програмного продукту став наказ закладу вищої освіти від “03” листопада 2020 року № 301 на затвердження теми дипломної роботи: “Дослідження та програмна реалізація бота-консультанта для персонального підбору комплексу вітамінів.” відповідно до наказу

A.3 Призначення розробки

Чат-бот містить в собі наступні функції:

- опитування користувача;
- вибір інтересуючого пріоритету;
- обробка опитування та отримання результату.

Всі необхідні дані будуть зберігатися у базі даних, що надасть чат-боту оброблювати запити та надавати відповідь враховуючи запитами користувачів.

A.4 Вимоги до програми

Вимоги до функціональних характеристик.

Розроблений продукт має забезпечувати наступні можливості:

- зручне та інтуїтивне користування чат-ботом;
- можливість ознайомитися з поданою інформацією;
- можливість описати характеристику;
- отримання результату рекомендацій за характеристикою.

Продукт має відповідати наступним технічним вимогам:

- компактність ;
- адаптивність – коректне відображення на будь-яких екранах;
- легке керування змістом чат-бота та його функціями;
- дотримання структури написання програми для легкості

подальшого редагування, або оновлення;

A.5 Етапи розробки

Розробка чат-боту починається з визначення приблизної концепції, дизайну та функціоналу. Після цього формується технічне завдання, де вже детально описується вище викладене. Етап розробки технічного завдання розбивається на наступні пункти:

- постановка задачі;
- визначення та уточнення вимог до технічних засобів;
- визначення вимог до програми;
- визначення стадій, етапів і термінів розробки програми і документації на неї;
- погодження та затвердження технічного завдання.

Потім настає етап проектування, спочатку інтерфейсу, а після і функціональної частини. Після завершення проектування, починається етап тестування розробленого продукту і редагування невірно працюючих частин.

ДОДАТОК Б
ОПИС ПРОГРАМИ

Б.1 Загальні відомості

В якості теми для дипломної роботи була розробка чат-боту для меседжеру “Telegram”.

Бот розроблений за ідеєю отримання чіткої рекомендації за характеристиками користувача, які оброблюються опитуванням, зв’язаними з комплексом вітамінів. Було створено такий продукт, яким користуватись та керувати можна будо б маючи мінімальні навички роботи з комп’ютером.

Б.2 Функціональне призначення

Продукт також дає можливість максимально швидко інформувати користувача про створення діалогу за новим запитом юзера чат-боту.

Зручне та інтуїтивне користування чат-ботом:

- можливість ознайомитися з поданою інформацією;
- можливість пройти опитування;
- отримання результату за опитуванням.

Б.3 Виклик і завантаження

Доступ до чат-боту можна отримати маючи будь-який гаджет під рукою, інтернет-підключення та месенджер Telegram.

Б.4 Вхідні дані

Вхідними даними для програми є ознаки, які надає програма в контексті, для зручного проходження опитування.

Б.5 Вихідні дані

Вихідними даними для програми є рекомендація щодо комплексу вітамінів за характеристиками користувача.

ДОДАТОК В
ТЕКСТ ПРОГРАМИ

```

import telebot
import time
import pypyodbc
import datetime
from telebot.types import Message
from telebot import types
from telebot.types import InputTextMessageContent
import sqlite3
token = "752538997:AAF1nsGJI9IS-Uxit_oJOqHN_dPfaIF8Jlw"
bot = telebot.TeleBot(token)

conn = sqlite3.connect("testdb.db", check_same_thread=False) #З'єднання з
базою даних
c = conn.cursor()

gender = ''
surname = ''
age = 0
choose = ''
end = ''
hide_markup = telebot.types.ReplyKeyboardRemove()

@bot.message_handler(commands=['start', 'help'])
def send_welcome(message):
    username = message.from_user.first_name
    user_markup = telebot.types.ReplyKeyboardMarkup(True, False)
    user_markup.row('✔Почати тестування', '✖Потім')
    bot.reply_to(message, 'Привіт, '+str(username)+'👋\n\n Цей тест про те, які
вітаміни і мікроелементи будуть корисні вашому організму \n\nВін займе
близько 3-х хвилин\n\nПочати? ☐', reply_markup=user_markup)

@bot.message_handler(func=lambda m: True)
def echo_all(message):
    if message.text == '✔Почати тестування':
        go(message)

@bot.message_handler(func=lambda m: True)
def go(message):
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(
        *[types.KeyboardButton(name) for name in ['Хлопець🧑♂️', 'Жінка🧑♀️']])
    bot.send_message(message.from_user.id, 'Виберете кнопками, ви👉',

```

```

reply_markup=keyboard)
    bot.register_next_step_handler(message, gender)

def gender(message):
    global gender
    gender = message.text
    bot.send_message(message.from_user.id, "👉Скільки вам
років?",reply_markup=hide_markup)
    bot.register_next_step_handler(message, taking)

def taking(message):
    global age
    age = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in ['Hi', 'Так, 1-3
виду', 'Так, більше 3-ох видів']])
    bot.send_message(message.from_user.id, 'Ви приймаєте вітаміни або інші
дієтичні добавки? 🍷',
                    reply_markup=keyboard)
    bot.register_next_step_handler(message, priorities)

@bot.message_handler(content_types=['text'])
def priorities(message):
    global taking
    taking = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=3)
    keyboard.add(*[types.KeyboardButton(name) for name in
['Імунітет', 'Сон', 'Стрес і настрій', 'Шкіра', 'Волосся і нігті',
'Фітнес і спорт', 'Енергія', 'Кістки і суглоби', 'Пам'ять і увагу', 'Серце і
судини', 'Корекція ваги', 'Травлення']])
    msg = bot.send_message(message.from_user.id,
                    'Ми пройшли першу частину опитування -
знайомство\nТепер дізнаємося більше про ваші цілі🎯',
                    reply_markup=keyboard)
    bot.register_next_step_handler(message, choose)

@bot.message_handler(content_types=['text'])
def choose(message):
    global bomb
    bomb = message.text
    if bomb == 'Імунітет':

```

```

    immunity(message)
elif bomb == 'Сон':
    dream(message)
elif bomb == 'Стрес чи настрій':
    mood(message)
elif bomb == 'Шкіра':
    skin(message)
elif bomb == 'Волосся і нігті':
    hair(message)
elif bomb == 'Фітнес та спорт':
    sport(message)
elif bomb == 'Енергія':
    energy(message)
elif bomb == 'Кістки і суглоби':
    bones(message)
elif bomb == 'Пам'ять та увага':
    memory(message)
elif bomb == 'Серце і судини':
    hearth(message)
elif bomb == 'Корекція ваги':
    weight(message)
elif bomb == 'Травлення':
    digestion(message)
return

def immunity(message):
    global immunity
    immunity = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in ['Так', 'Ні']])
    bot.send_message(message.from_user.id, "Ви часто хворієте на застуди та
ГРПІ? □", reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity1)

def immunity1(message):
    global immunity1
    immunity1 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in ['Так', 'Ні']])
    bot.send_message(message.from_user.id,
        "Ви регулярно використовуєте громадський транспорт? 🚗
Піддаючи себе ризику респіраторних захворювань?",
        reply_markup=keyboard)

```

```
bot.register_next_step_handler(message, immunity2)
```

```
def immunity2(message):
    global immunity2
    immunity2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in ['Так', 'Ні']])
    bot.send_message(message.from_user.id,
        "За останні 6 місяців ви приймали антибіотики? 🩺",
        reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity3)
```

```
def immunity3(message):
    global immunity3
    immunity3 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in ['Так', 'Ні']])
    bot.send_message(message.from_user.id,
        "Чи вважаєте ви за необхідне профілактично підтримувати
роботу серця і стану судин? ♥",
        reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity4)
```

```
def immunity4(message):
    global immunity4
    immunity4 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in ['Так', 'Ні']])
    bot.send_message(message.from_user.id,
        "У вашій сім'ї є спадкові проблеми з серцево-судинною
системою? 👤👤👤",
        reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity5)
```

```
def immunity5(message):
    global immunity5
    immunity5 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in ['Скинути вагу↘',
'Набрати вагу↗', 'Не змінювати вагу👌']])
```

```

bot.send_message(message.from_user.id,
                  "Найближчим часом ви хочете:",
                  reply_markup=keyboard)
bot.register_next_step_handler(message, immunity6)

def immunity6(message):
    global immunity6
    immunity6 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in ['Борошняні
вироби', 'Білкова їжа', 'Рослинну їжу', 'Різноманітну']])
    bot.send_message(message.from_user.id,
                      "У своєму раціоні ви віддаєте перевагу:",
                      reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

def immunity7(message):
    global immunity7
    immunity7 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                  ['Залізо', '💧Вітамін B12', '💧Вітамін E', 'Кальцій', '💧Вітамін
Д3', 'Ні / Не знаю']])
    bot.send_message(message.from_user.id,
                      "Чи маєте ви дефіцит або призначення до прийому цих
добавок: \n (За рекомендацією фахівця або на основі аналізів)",
                      reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle)

def lifestyle(message):
    global lifestyle
    lifestyle = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                  ['Веду ЗСЖ', 'Намагаюся дотримуватися', 'Хочу почати',
'Інше']])
    bot.send_message(message.from_user.id,
                      "Тепер дізнаємося більше про ваш стиль життя і харчування
в цілому.\n\nЯк ви ставитеся до здорового способу життя?",
                      reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle1)

```

```

def lifestyle1(message):
    global lifestyle1
    lifestyle1 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Їм усе', 'Вегетаріанство', 'Веганство', 'Інша']])
    bot.send_message(message.from_user.id, "Ваша дієта",
reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle2)

def lifestyle2(message):
    global lifestyle2
    lifestyle2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Глютен', 'Гриби', 'Лактоза', 'Цукор', 'Морепродукти', 'Інша',
'Нічого не виключаю']])
    bot.send_message(message.from_user.id, "Ви виключаєте зі свого раціону
(через алергію і непереносимості, наприклад.):", reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle3)

def lifestyle3(message):
    global lifestyle3
    lifestyle3 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['1-2 або менше', 'Три-вісім', 'Більше восьми']])
    bot.send_message(message.from_user.id, "Скільки порцій овочів і фруктів
їсте в день? 🍎 \n [Одна порція = 50 грам] ", reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle4)

def lifestyle4(message):
    global lifestyle4
    lifestyle4 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Рідко або не їм', 'Два-три рази на тиждень або частіше']])
    bot.send_message(message.from_user.id,
        "Як часто їсте продукти, багаті протеїном \n [М'ясо, риба,
яйця, сир, бобові] ",
        reply_markup=keyboard)

```

```

bot.register_next_step_handler(message, lifestyle5)

def lifestyle5(message):
    global lifestyle5
    lifestyle5 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Не їм або дуже Рідко', 'Два-три рази на тиждень і частіше']])
    bot.send_message(message.from_user.id,
        "Як часто їсте горіхи і насіння? \n [Наприклад, волоські
горіхи або насіння льону☐]",
        reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle6)

def lifestyle6(message):
    global lifestyle6
    lifestyle6 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Так', 'Ні']])
    bot.send_message(message.from_user.id,
        "Чи присутні у вашому щоденному раціоні йодовмісні
продукти? \n Наприклад, йодована сіль, морська капуста, морська риба і
морепродукти☐",
        reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle7)

def lifestyle7(message):
    global lifestyle7
    lifestyle7 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Не п'ю або що не перевищує «безпечну дозу»', ' Один раз ',
Два рази або частіше ']])
    bot.send_message(message.from_user.id,
        "Скільки разів на тиждень перевищуєте« безпечну дозу
»алкоголю? 🍷 \n [Одна « безпечна доза » = 30-40 мл чистого спирту / 250
мл сухого вина / 700 мл пива / 80 мл міцних напоїв]",
        reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle8)

def lifestyle8(message):

```

```

global lifestyle8
lifestyle8 = message.text
keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
keyboard.add(*[types.KeyboardButton(name) for name in
                ['Так', 'Ні']])

bot.send_message(message.from_user.id, "Палите? 🚬", reply_markup=keyboard)
bot.register_next_step_handler(message, lifestyle9)

def lifestyle9(message):
    global lifestyle9
    lifestyle9 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['30 хвилин і більше ', ' Менше 30 хвилин ']])
    bot.send_message(message.from_user.id, "Скільки часу в день проводите
на відкритому сонці? ☀️ \n [Відкриті обличчя і руки, без захисного
крему]", reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle10)

def lifestyle10(message):
    global lifestyle10
    lifestyle10 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['Так', 'Ні']])
    bot.send_message(message.from_user.id, "Після тривалого часу за
комп'ютером / планшетом / телесроном відчуваєте дискомфорт в очах? 👁
\n [Наприклад, сухість або почервоніння]", reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle11)

def lifestyle11(message):
    global lifestyle11
    lifestyle11 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['Так', 'Рідко', 'Ні']])
    bot.send_message(message.from_user.id, "Чи турбують вас головні
болі? ☐", reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle12)

```

```

def lifestyle12(message):
    global lifestyle12
    lifestyle12 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Так', 'Ні']])
    bot.send_message(message.from_user.id, "Чи бувають у вас судоми
частіше, ніж раз на тиждень?", reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle13)

def lifestyle13(message):
    global lifestyle13
    lifestyle13 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Так', 'Ні']])
    bot.send_message(message.from_user.id, "У вас легко з'являються
синці?",
        reply_markup=keyboard)
    bot.register_next_step_handler(message, lifestyle14)

def lifestyle14(message):
    global lifestyle14
    lifestyle14 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Ні', 'Антибіотики', 'Антидепресанти', 'Курс лікування від
хронічних хвороб', 'Інші']])
    chat_id = message.chat.id
    msg = bot.send_message(chat_id, "Ви приймаєте в даний момент
лікарські препарати?💊",
        reply_markup=keyboard)
    bot.register_next_step_handler(msg, choise)

def dream(message):
    global dream
    dream = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in

```

```

        ['Так', 'Ні']])
    bot.send_message(message.from_user.id, "Чи відчуваєте труднощі при
засипанні?☹️", reply_markup=keyboard)
    bot.register_next_step_handler(message, dream2)

```

```

def dream2(message):
    global dream2
    dream2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Бадьоро і з запасом сил', 'Втомлені і розбиті', 'Інший
варіант']])
    bot.send_message(message.from_user.id, "Як ви себе почуваєте після
нічного сну?", reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

```

```

def mood(message):
    global mood
    mood = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Так, часто', 'Час від часу', 'Ні, Рідко або ніколи']])
    bot.send_message(message.from_user.id, "Чи відчуваєте тривожність,
непевненість в собі, пригнічений настрій?", reply_markup=keyboard)
    bot.register_next_step_handler(message, mood2)

```

```

def mood2(message):
    global mood2
    mood2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
        ['Ні', 'Рідко', 'В постійному стресі']])
    bot.send_message(message.from_user.id, "Часто відчуваєте стрес, емоційні
навантаження?", reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

```

```

def skin(message):
    global skin
    skin = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in

```

```

        ['Досить зволожена', 'Суха']])
    bot.send_message(message.from_user.id, "Останнім часом ваша шкіра?",
                      reply_markup=keyboard)
    bot.register_next_step_handler(message, skin2)

def skin2(message):
    global skin2
    skin2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['Еластична', 'Втратила пружність']])
    bot.send_message(message.from_user.id, "Чи відзначаєте, що ваша
шкіра",reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

def hair(message):
    global hair
    hair = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['Сухі', 'Ламкі', 'Чи схильні до випадання', 'В хорошому стані',
'Інша']])
    bot.send_message(message.from_user.id, "Ваше
волосся",reply_markup=keyboard)
    bot.register_next_step_handler(message, hair2)

def hair2(message):
    global hair2
    hair2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['Ламкі і розшаровуються', 'Виражена лейконихия (білі плями
на поверхні)', 'Інша', 'В хорошому стані']])
    bot.send_message(message.from_user.id, "Ваші
нігті👉",reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

def sport(message):
    global sport
    sport = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)

```

```

keyboard.add(*[types.KeyboardButton(name) for name in
               ['1-2', '3-4', '5 і більше', 'Чи не займаюся']])
bot.send_message(message.from_user.id, "Скільки разів на тиждень ви
активно займаєтеся фізичним вправами? 📊", reply_markup=keyboard)
bot.register_next_step_handler(message, sport2)

```

```

def sport2(message):
    global sport2
    sport2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                   ['Постійно', 'Час від часу', 'Ні']])
    bot.send_message(message.from_user.id, "Під час занять ви даєте собі
кардіо навантаження? 🏃📊 \n [Біг, ігрові види спорту, плавання і тд]",
                    reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

```

```

def energy(message):
    global energy
    energy = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                   ['Так', 'Ні']])
    bot.send_message(message.from_user.id, "При звичайному графіку дня, ви
відчуваєте втому і брак сил вже до обіду?",
                    reply_markup=keyboard)
    bot.register_next_step_handler(message, energy2)

```

```

def energy2(message):
    global energy2
    energy2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                   ['Так', 'Ні']])
    bot.send_message(message.from_user.id, "Прокинувшись вранці, вам
необхідно зарядитися бадьорістю на весь день за"
                    "допомогою енергетиків? [Наприклад, кава,
міцний чай☕]", reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

```

```

def bones(message):

```

```

global bones
bones = message.text
keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
keyboard.add(*[types.KeyboardButton(name) for name in
                ['Так', 'Ні(не знаю)'])
bot.send_message(message.from_user.id,
                "Чи є у вашій родині спадкові проблеми кістками? □",
                reply_markup=keyboard)
bot.register_next_step_handler(message, bones2)

```

```

def bones2(message):
    global bones2
    bones2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['Так', 'Ні'])
    bot.send_message(message.from_user.id,
                    "Чи турбують вас суглоби в даний період життя? ",
                    reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

```

```

def memory(message):
    global memory
    memory = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['Так', 'Ні'])
    bot.send_message(message.from_user.id,
                    "Забуваєте давно отриману інформацію, маєте проблеми з
короткостроковою пам'яттю або труднощі з концентрацією уваги?",
                    reply_markup=keyboard)
    bot.register_next_step_handler(message, memory2)

```

```

def memory2(message):
    global memory2
    memory2 = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                    ['Так', 'Ні', 'Планую'])
    bot.send_message(message.from_user.id,
                    "Зараз у вас період високих інтелектуальних навантажень?

```

Вимагають підвищеної концентрації уваги?",

reply_markup=keyboard)

bot.register_next_step_handler(message, immunity7)

def hearth(message):

global hearth

hearth = message.text

keyboard = types.ReplyKeyboardMarkup(**resize_keyboard=True,**
row_width=2)

keyboard.add(*[types.KeyboardButton(name) **for** name **in** ['Так', 'Ні']])

bot.send_message(message.from_user.id,

"Чи вважаєте ви за необхідне профілактично підтримувати роботу серця і стану судин?♥",

reply_markup=keyboard)

bot.register_next_step_handler(message, hearth2)

def hearth2(message):

global hearth2

hearth2 = message.text

keyboard = types.ReplyKeyboardMarkup(**resize_keyboard=True,**
row_width=2)

keyboard.add(*[types.KeyboardButton(name) **for** name **in** ['Так', 'Ні']])

bot.send_message(message.from_user.id,

"У вашій сім'ї є спадкові проблеми з серцево-судинною системою?👤👤👤",

reply_markup=keyboard)

bot.register_next_step_handler(message, immunity7)

def weight(message):

global weight

weight = message.text

keyboard = types.ReplyKeyboardMarkup(**resize_keyboard=True,**
row_width=2)

keyboard.add(*[types.KeyboardButton(name) **for** name **in** ['Скинути вагу↘',
'Набрати вагу↗', 'Не змінювати вагу👌']])

bot.send_message(message.from_user.id,

"Найближчим часом ви хочете:",

reply_markup=keyboard)

bot.register_next_step_handler(message, weight2)

def weight2(message):

global weight2

weight2 = message.text

```

keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
keyboard.add(*[types.KeyboardButton(name) for name in ['Борошняні
вироби', 'Білкова їжа', 'Рослинну їжу', 'Різноманітну']])
bot.send_message(message.from_user.id,
                "У своєму раціоні ви віддаєте перевагу:",
                reply_markup=keyboard)
bot.register_next_step_handler(message, immunity7)

```

```

def digestion(message):
    global digestion
    digestion = message.text
    keyboard = types.ReplyKeyboardMarkup(resize_keyboard=True,
row_width=2)
    keyboard.add(*[types.KeyboardButton(name) for name in
                ['Дисбактеріоз', 'Здуття', 'Тяжкість', 'Нема проблем']])
    bot.send_message(message.from_user.id,
                "Чи є у вас проблеми з шлунково-кишкового тракту?",
                reply_markup=keyboard)
    bot.register_next_step_handler(message, immunity7)

```

```

def choise(message):
    if bomb == 'Імунітет':
        done(message)
        immunity_result(message)
    elif bomb == 'Сон':
        done(message)
        dream_result(message)
    elif bomb == 'Стрес чи настрій':
        done(message)
        mood_result(message)
    elif bomb == 'Шкіра':
        done(message)
        skin_result(message)
    elif bomb == 'Волосся і нігті':
        done(message)
        hair_result(message)
    elif bomb == 'Фітнес та спорт':
        done(message)
        sport_result(message)
    elif bomb == 'Енергія':
        done(message)
        energy_result(message)
    elif bomb == 'Кістки і суглоби':
        done(message)

```

```

bones_result(message)
elif bomb == 'Пам'ять та увага':
    done (message)
    memory_result(message)
elif bomb == 'Серце і судини':
    done (message)
    hearth_result(message)
elif bomb == 'Корекція ваги':
    done (message)
    weight_result(message)
elif bomb == 'Травлення':
    done (message)
    digestion_result(message)

def immunity_result(message):
    bot.send_message(message.from_user.id, '** Омега-3 ** \nПреміум
риб'ячий жир \n\n - підтримка здоров'я серця \n- 55% концентрація
жирних кислот ЕРА і ДНА \n- виготовлений з жирних сортів риби в
Ісландії')
    bot.send_message(message.from_user.id, '** Вітамін D3💊 ** \n 50 мкг / 2000
МЕ \n\n - Сприяє засвоєнню кальцію\n - Сезонний дефіцит вітаміну D
один з найпоширеніших в Україні')
    bot.send_message(message.from_user.id, '** Вітамін C💊 ** \n 400 мг \n\n-
антиоксидант \n- сприяє підтримці імунітету')
    bot.send_message(message.from_user.id, '** Цинк ** \nПідтримка імунітету
\n\n- біодоступність (хелатна, гліцинат) форма \n- підтримує імунну
систему і шкіру')
    msg = bot.send_message(message.from_user.id, '** Кальцій + ** \nЗахист
кісток \n\n- ефективний комплекс для підтримки кісток \n- з вітаміном
D3 для засвоєння кальцію')
    if message.text == "Пройти тест ще раз та обрати іншу категорію🔄":
        bot.register_next_step_handler (msg, go)

def dream_result(message):
    bot.send_message(message.from_user.id, '** 5-htp **\nНастрій та сон\n\n—
Попередник серотоніну з натурального екстракту гриффоні\n-
Підтримка здорового сну і настрою \n- Підтримка при корекції ваги ')
    bot.send_message(message.from_user.id, '** L-Теанін **\nРелакс и
концентрація\n\n- Підтримка здорового протистояння стресу \n-
Підтримка здорового сну \n- Підтримка когнітивних функцій:
концентрація, пам'ять і увага ')
    msg = bot.send_message(message.from_user.id, '** L-Тріптофан **\n400 мг
+ B6\n\n- сприяє підвищенню настрою \n- сприяє поліпшенню якості сну
')
```

```

if message.text == "Пройти тест ще раз та обрати іншу категорію🌀":
    bot.register_next_step_handler (msg, go)

def mood_result(message):
    bot.send_message(message.from_user.id,
        '** В-Комплекс🍷 **\n7 вітамінів групи В\n\n- підтримує метаболізм \n- підтримує нервову систему \n- заповнює дефіцит при веганської \n вегетаріанської дієти ')
    bot.send_message(message.from_user.id,
        '** Магній **\nАнтистресс\n\n- Морський гідроксид магнію з високою біодоступністю \n- Видобувається в чистих водах Ірландського моря \n- Природно містить 72 мікроелемента \n- 3 вітаміном В6 для підтримки нервової системи ')
    bot.send_message(message.from_user.id,
        '** 5-htp **\nНастрій та сон\n\n- Попередник серотоніну з натурального екстракту грифоні\n- Підтримка здорового сну і настрою \n- Підтримка при корекції ваги ')
    msg = bot.send_message(message.from_user.id,
        '** L-Теанін **\nРелакс и концентрація\n\n- Підтримка здорового протистояння стресу \n- Підтримка здорового сну \n- Підтримка когнітивних функцій: концентрація, пам'ять і увага ')
    if message.text == "Пройти тест ще раз та обрати іншу категорію🌀":
        bot.register_next_step_handler (msg, go)

def skin_result(message):
    bot.send_message (message.from_user.id,
        '** Омега-3 ** \nПреміум риб'ячий жир \n\n - підтримка здоров'я серця \n- 55% концентрація жирних кислот ЕРА і DHA \n- виготовлений з жирних сортів риби в Ісландії')
    bot.send_message (message.from_user.id,
        '** Біотин + ** \nКожа, волосся, нігті \n\n - Підтримка здоров'я волосся, шкіри, нігтів. \n- Підтримка метаболізму')
    bot.send_message (message.from_user.id,
        '** Вітамін Е🍷 ** \n100 мг / 136 МЕ \n\n- забезпечує захист шкіри \n- бере участь в біосинтезі колагену')
    bot.send_message (message.from_user.id,
        '** Вітамін С🍷 ** \n400 мг \n\n- антиоксидант \n- сприяє підтримці імунітету')
    msg = bot.send_message (message.from_user.id,
        '** Цинк ** \nПідтримка імунітету \n\n- біодоступність (хелатна, гліцинат) форма \n- підтримує імунну систему і шкіру')
    if message.text == "Пройти тест ще раз та обрати іншу категорію🌀":
        bot.register_next_step_handler (msg, go)

```

```

def hair_result(message):
    bot.send_message (message.from_user.id,
        '** Біотин + ** \nКожа, волосся, нігті \n\n - Підтримка
здоров'я волосся, шкіри, нігтів. \n- Підтримка метаболізму')
    bot.send_message(message.from_user.id,
        '** В-Комплекс **\n7 вітамінів групи В\n\n- підтримує
метаболізм \n- підтримує нервову систему \n- заповнює дефіцит при
веганської \ вегетаріанської дієти ')
    bot.send_message (message.from_user.id,
        '** Вітамін Е ** \n100 мг / 136 МЕ \n\n- забезпечує захист
шкіри \n- бере участь в біосинтезі колагену')
    msg = bot.send_message (message.from_user.id,
        '** Вітамін С ** \n 400 мг \n\n- антиоксидант \n- сприяє
підтримці імунітету')
    if message.text == "Пройти тест ще раз та обрати іншу категорію":
        bot.register_next_step_handler (msg, go)

def sport_result(message):
    bot.send_message(message.from_user.id,
        '** В-Комплекс **\n7 вітамінів групи В\n\n- підтримує
метаболізм \n- підтримує нервову систему \n- заповнює дефіцит при
веганської \ вегетаріанської дієти ')
    bot.send_message (message.from_user.id,
        '** Вітамін В-12 ** \n100 мкг \n\n- незамінний вітамін для
енергообміну в організмі; \n- бере участь у формуванні нервових
волокон; \n- забезпечує роботу центральної нервової системи;')
    bot.send_message (message.from_user.id,
        '** Метілфолат (5-MTHF) ** \nЗдорова вагітність \n\n-
Кращий в світі фолат від італійської компанії Gnosis \n- 100%
засвоюваність \n- Підтримка здоров'я при плануванні і під час
вагітності')
    msg = bot.send_message (message.from_user.id,
        '** Коензим Q10 ** \n Енергетик для клітин \n\n- підтримка
здоров'я серця \n- антиоксидант \n- бере участь у виробництві клітинної
енергії')
    if message.text == "Пройти тест ще раз та обрати іншу категорію":
        bot.register_next_step_handler (msg, go)

def energy_result(message):
    bot.send_message(message.from_user.id,
        '** В-Комплекс **\n7 вітамінів групи В\n\n- підтримує
метаболізм \n- підтримує нервову систему \n- заповнює дефіцит при
веганської \ вегетаріанської дієти ')
    bot.send_message(message.from_user.id,

```

```

        *** Вітамін B-12💊 *** \n100 мкг \n\n- незамінний вітамін для
енергообміну в організмі; \n- бере участь у формуванні нервових
волокон; \n- забезпечує роботу центральної нервової системи;')
        bot.send_message(message.from_user.id,
            *** Метілфолат (5-MTHF) *** \nЗдорова вагітність \n\n-
Кращий в світі фолат від італійської компанії Gnosis \n- 100%
засвоюваність \n- Підтримка здоров'я при плануванні і під час
вагітності')
        bot.send_message (message.from_user.id,
            *** Коензим Q10 *** \ Енергетик для клітин \n\n- підтримка
здоров'я серця \n- антиоксидант \n- бере участь у виробництві клітинної
енергії')
        msg = bot.send_message(message.from_user.id,
            *** Залізо *** \n 18 мг \n\n- біодоступність хелатна форма
(гліцинат) \n- У поєднанні з вітаміном С \n- Підтримка при
профілактиці анемії')
        if message.text == "Пройти тест ще раз та обрати іншу категорію🔁":
            bot.register_next_step_handler (msg, go)

def bones_result(message):
    bot.send_message(message.from_user.id,
        *** Вітамін D3💊 *** \n 50 мкг / 2000 ME \n\n - Сприяє
засвоєнню кальцію\n - Сезонний дефіцит вітаміну D один з
найпоширеніших в Україні')
    bot.send_message (message.from_user.id,
        *** Кальцій + *** \nЗахист кісток \n\n- ефективний комплекс
для підтримки кісток \n- з вітаміном D3 для засвоєння кальцію')
    bot.send_message(message.from_user.id,
        *** Магній ***\nАнтистресс\n\n- Морський гідроксид магнію з
високою біодоступністю \n- Видобувається в чистих водах Ірландського
моря \n- Природно містить 72 мікроелемента \n- З вітаміном B6 для
підтримки нервової системи ')
    msg = bot.send_message(message.from_user.id,
        *** Хондроитин + *** \nC глюкозаміном і вітаміном D \n\n-
захист від остеопорозу \n- знеболюючий ефект \n- відновлення після
травм')
    if message.text == "Пройти тест ще раз та обрати іншу категорію🔁":
        bot.register_next_step_handler (msg, go)

def memory_result(message):
    bot.send_message (message.from_user.id,
        *** Омега-3 *** \nПреміум риб'ячий жир \n\n - підтримка
здоров'я серця \n- 55% концентрація жирних кислот EPA і DHA \n-
виготовлений з жирних сортів риби в Ісландії')

```

```

bot.send_message(message.from_user.id,
                  '** Вітамін В-12💊 ** \n100 мкг \n\n- незамінний вітамін для
енергообміну в організмі; \n- бере участь у формуванні нервових
волокон; \n- забезпечує роботу центральної нервової системи;')
bot.send_message(message.from_user.id,
                  '** Метілфолат (5-MTHF) ** \nЗдорова вагітність \n\n-
Кращий в світі фолат від італійської компанії Gnosis \n- 100%
засвоюваність \n- Підтримка здоров'я при плануванні і під час
вагітності')
bot.send_message (message.from_user.id,
                  '** Куркумін ** \nЗахист суглобів \n\n- Стандартизований
екстракт \n- У комплексі з екстрактом чорного перцю для кращого
засвоєння \n- Підтримка здоров'я суглобів і когнітивних функцій')
msg = bot.send_message (message.from_user.id,
                        '** L-Теанін **\nРелакс и концентрація\n\n- Підтримка
здорового протистояння стресу \n- Підтримка здорового сну \n-
Підтримка когнітивних функцій: концентрація, пам'ять і увага ')
if message.text == "Пройти тест ще раз та обрати іншу категорію🔄":
    bot.register_next_step_handler (msg, go)

def hearth_result(message):
    keyboard = types.ReplyKeyboardMarkup (resize_keyboard=True,
row_width=2)
    keyboard.add (*[types.KeyboardButton (name) for name in
                    ['Спробую✔', 'Пройти тест ще раз та обрати іншу
категорію🔄']])
    bot.send_message (message.from_user.id,
                      '** Омега-3 ** \nПреміум риб'ячий жир \n\n - підтримка
здоров'я серця \n- 55% концентрація жирних кислот ЕРА і ДНА \n-
виготовлений з жирних сортів риби в Ісландії')
    bot.send_message (message.from_user.id,
                      '** Коензим Q10 ** \nЕнергетик для клітин \n\n- підтримка
здоров'я серця \n- антиоксидант \n- бере участь у виробництві клітинної
енергії')
    bot.send_message (message.from_user.id,
                      '** Магній **\nАнтистресс\n\n- Морський гідроксид магнію з
високою біодоступністю \n- Видобувається в чистих водах Ірландського
моря \n- Природно містить 72 мікроелемента \n- 3 вітаміном В6 для
підтримки нервової системи ')
    msg = bot.send_message (message.from_user.id,
                            '** Селен ** \n55 мкг \n\n- антиоксидант \n- підтримує роботу
щитовидної залози \n- сприяє підтримці імунітету')
    if message.text == "Пройти тест ще раз та обрати іншу категорію🔄":
        bot.register_next_step_handler (msg, go)

```

```

def weight_result(message):
    keyboard = types.ReplyKeyboardMarkup (resize_keyboard=True,
row_width=2)
    keyboard.add (*[types.KeyboardButton (name) for name in
        ['Спробую✔', 'Пройти тест ще раз та обрати іншу
категорію🔄']])
    bot.send_message (message.from_user.id,
        '** Селен ** \n55 мкг \n\n- антиоксидант \n- підтримує роботу
щитовидної залози \n- сприяє підтримці імунітету')
    bot.send_message(message.from_user.id,
        '** 5-htp **\nНастрій та сон\n\n– Попередник серотоніну з
натурального екстракту гріфони\n- Підтримка здорового сну і настрою
\n- Підтримка при корекції ваги ')
    bot.send_message (message.from_user.id,
        '** Куркумін ** \n Захист суглобів \n\n- Стандартизований
екстракт \n- У комплексі з екстрактом чорного перцю для кращого
засвоєння \n- Підтримка здоров'я суглобів і когнітивних функцій')
    bot.send_message (message.from_user.id,
        '** Органічна спіруліна ** \n Суперфуд \n\n- Рослинний
джерело білка \n- Підтримка загального стану здоров'я')
    bot.send_message (message.from_user.id,
        '** Екстракт гарцинии **\nКорректор ваги\n\n- природне
джерело гідроксилимонной кислоти\n- сприяє корекції
ваги',reply_markup=keyboard)
    if message.text == "Пройти тест ще раз та обрати іншу категорію🔄":
        bot.register_next_step_handler(message, restart)

```

```

def digestion_result(message):
    keyboard = types.ReplyKeyboardMarkup (resize_keyboard=True,
row_width=2)
    keyboard.add (*[types.KeyboardButton (name) for name in
        ['Спробую✔', 'Пройти тест ще раз та обрати іншу
категорію🔄']])
    bot.send_message (message.from_user.id,
        '** Органічна спіруліна ** \n Суперфуд \n\n- Рослинний
джерело білка \n- Підтримка загального стану здоров'я')
    bot.send_message (message.from_user.id,
        '** Силімарин Максі ** \nГепатопротектор \n\n- природна
підтримка печінки \n- 80% зміст силімарину')
    bot.send_message (message.from_user.id,
        '** Пробиотик ** \n Корисні бактерії \n\n- Містить культури
Lactobacillus Plantarum, Streptosoccus thermophilus. \n- Забезпечує

```

```

відновлення нормальної мікрофлори кишечника')
    msg = bot.send_message (message.from_user.id,
        '** Екстракт гарцини **\nКорректор ваги\n\n- природне
джерело гідроксилимонної кислоти\n- сприяє корекції
ваги',reply_markup=keyboard)
    if message.text == "Пройти тест ще раз та обрати іншу категорію ↻":
        bot.register_next_step_handler (msg, restart)

def restart(message):
    go(message)

@bot.message_handler(func=lambda m: True)
def done(message):
    keyboard = types.ReplyKeyboardMarkup (resize_keyboard=True,
row_width=2)
    keyboard.add (*[types.KeyboardButton (name) for name in
        ['Спробую✔', 'Пройти тест ще раз та обрати іншу
категорію ↻']])
    msg = bot.send_message(message.from_user.id, "Ми наполегливо
рекомендуємо отримати консультацію у фахівця, якщо ви плануєте
приймати добавки спільно з призначеними вам лікарськими
препаратами.💊💰"
        "\n\n Дякую за ваші відповіді! Нашому
алгоритму потрібно декілька секунд для"
        "обробки ваших відповідей. \n Зараз отримаєте
персональними рекомендацію за вашими результатами
опитування.↘↓✔",reply_markup=keyboard)

bot.polling(none_stop=True,interval=0)

```

ДОДАТОК Г
СЛАЙДИ ПРЕЗЕНТАЦІЇ

Запорізький національний технічний університет
Кафедра програмних засобів

Дипломна кваліфікаційна робота ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ БОТА-КОНСУЛЬТАНТА ДЛЯ ПЕРСОНАЛЬНОГО ПІДБОРУ КОМПЛЕКСУ ВІТАМІНІВ

Виконав
ст.гр. КНТ-219м

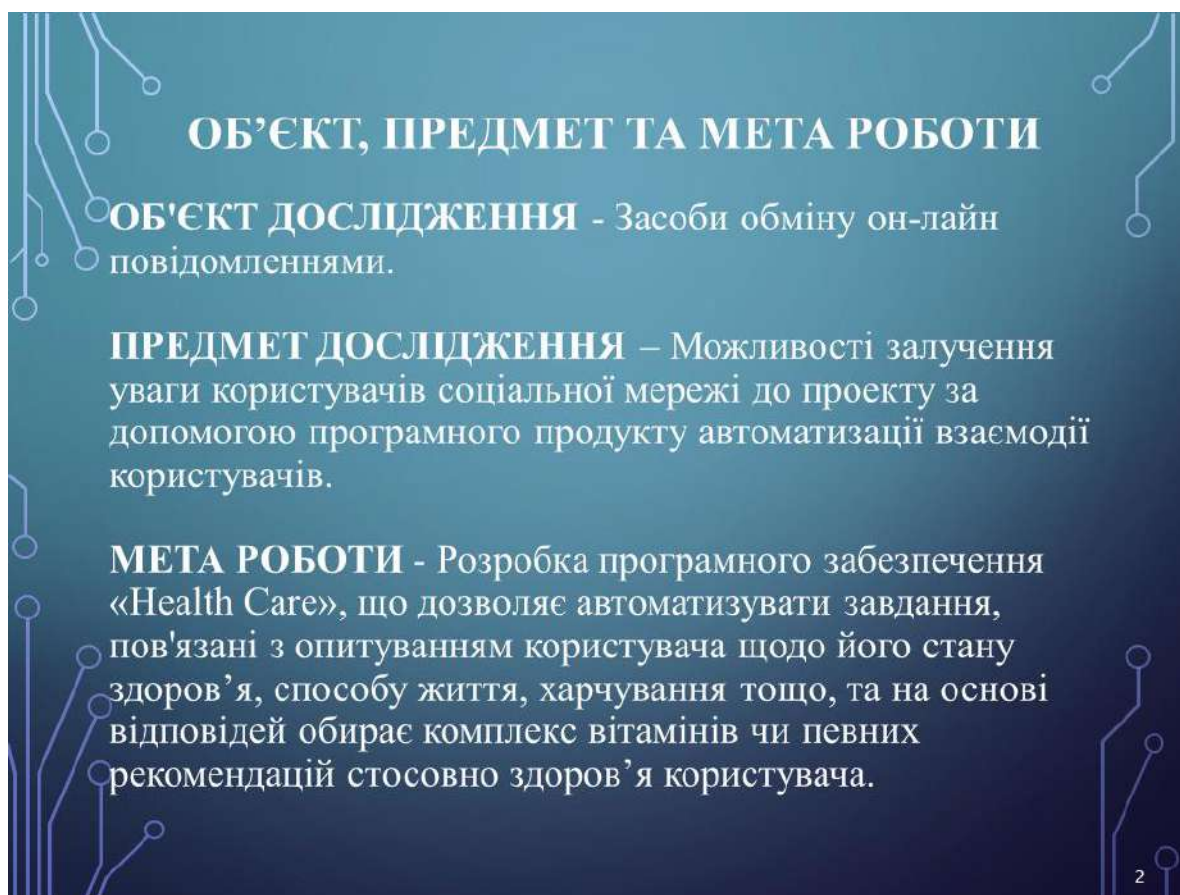
Ю.О.Фоменко

Керівник
доцент,к.т.н.

І.Я.Зеленьова

Запоріжжя, 2020

Рисунок Д.1 – Слайд №1



ОБ'ЄКТ, ПРЕДМЕТ ТА МЕТА РОБОТИ

ОБ'ЄКТ ДОСЛІДЖЕННЯ - Засоби обміну он-лайн повідомленнями.

ПРЕДМЕТ ДОСЛІДЖЕННЯ – Можливості залучення уваги користувачів соціальної мережі до проекту за допомогою програмного продукту автоматизації взаємодії користувачів.

МЕТА РОБОТИ - Розробка програмного забезпечення «Health Care», що дозволяє автоматизувати завдання, пов'язані з опитуванням користувача щодо його стану здоров'я, способу життя, харчування тощо, та на основі відповідей обирає комплекс вітамінів чи певних рекомендацій стосовно здоров'я користувача.

2

Рисунок Г.2 – Слайд №2

НАУКОВА НОВИЗНА ТА ПРАКТИЧНА ЦІННІСТЬ

Наукова новизна полягає в тому, що проаналізовано велику кількість програмних засобів для реалізації проекту, продемонстровано велика кількість застосувань чат-ботів в бізнесі, а також розроблений унікальний чат-бот, безпосередньо для громадян щодо зміцнення здоров'я, профілактики захворювань та подолання їх наслідків, щоб люди не залишалися наодинці зі своїми недугами.

Практична значимість полягає в тому, що даний чат-бот можна застосувати в будь-якій медичній компанії, в якій є бізнес-процес обробки звернень клієнтів в чат.

3

Рисунок Г.3 – Слайд №3

ЗАВДАННЯ РОБОТИ:

- дослідити існуючі підходи для розробки бота;
- дослідити існуючі проблеми чат-ботів;
- визначення програмної платформи, а саме фреймворків та інтерфейс прикладних рішень і бібліотек API з великою кількістю доступних інструментів: функцій, класів, структур, які підходять для сучасної розробки бота;
- дослідити методи машинного навчання для вдосконалення ботів, та їх взаємодію з користувачем;
- розробка прикладної програми;
- впровадження розробленого програмного забезпечення.

4

Рисунок Г.4 – Слайд №4



Рисунок Г.5 – Слайд №5



Рисунок Г.6 – Слайд №6

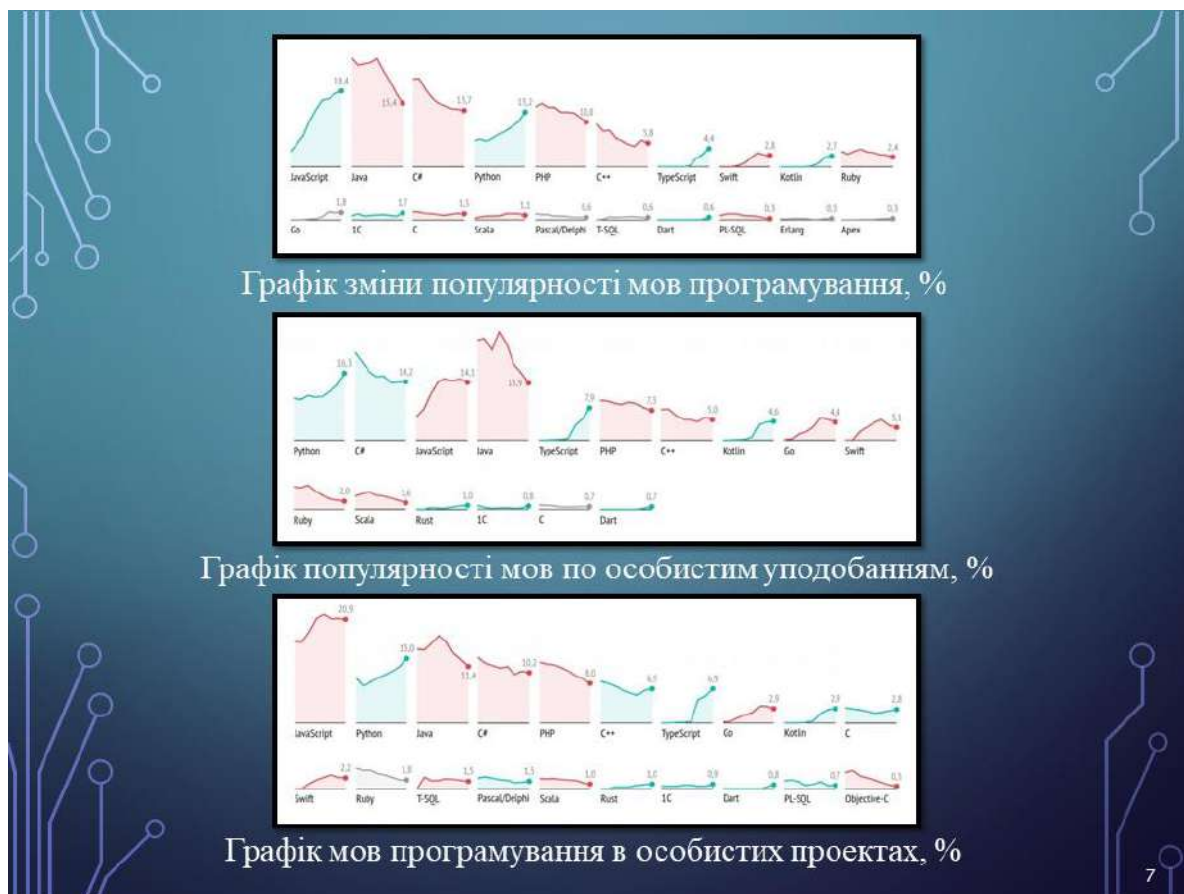


Рисунок Г.7 – Слайд №7

ВИБІР ІНСТРУМЕНТАРІЮ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕСПЕЧЕННЯ

Python – об'єктно-орієнтована мова програмування

SQLite3 – реляційна СУБД

Telegram API – Прикладний програмний інтерфейс

8

Рисунок Г.8 – Слайд №8

ПОРІВНЯЛЬНА ХАРАКТЕРИСТИКА МОВ ПРОГРАМУВАННЯ			
Критерій порівняння	C#	Python	Microsoft Visual Basic
Базова мова програмування	C	Python	Basic
Клас мови	об'єктно-орієнтована	об'єктно-орієнтована	об'єктно-орієнтована
Синтаксис	Простий, C-подібний, статична типізація	простий, дозволяє дуже швидко освоїти мову	простий
Компілятор	проводить статичну перевірку типів, але на додаток до цього робить перевірку коректності під час завантаження	високопродуктивний, найшвидший	повноцінний
Робота з базами даних	взаємодіє з базами даних через мову SQL	взаємодіє з базами даних через мову SQL	набір об'єктних бібліотек, що входять в.NET Framework
Середовище розробки додатків	інтегроване	інтегроване	інтегроване
Сприйняття коду	гарне	гарне	погане

Рисунок Г.9 – Слайд №9



Рисунок Г.10 – Слайд №10

СТРУКТУРА ТАБЛИЦІ БАЗИ ДАНИХ SQLITE

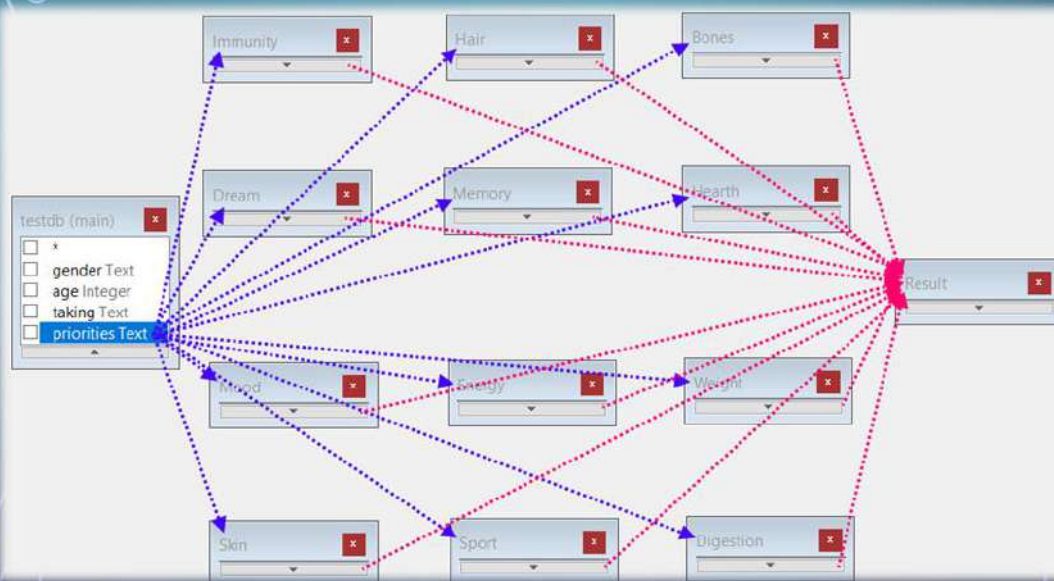
Таблиця test.db

№	Назва	Тип поля	Опис поля
№ з/п	Назва поля	Тип поля	Опис поля
1	ID	Int(11), AUTOINCREMENT	Ідентифікатор запиту
2	gender	VarChar(100)	Юзернейм користувача
3	age	VarChar(100)	Ідентифікатор користувача телеграм
4	taking	VarChar(100)	Обробка несумісності
5	priorities	VarChar(100)	Пріоритет користувача
6	Date	DATE(100)	Дата подання запиту

11

Рисунок Г.11 – Слайд №11

ГРАФІК ТАБЛИЦІ «TESTDB» БАЗИ ДАНИХ



12

Рисунок Г.12 – Слайд №12



Рисунок Г.13 – Слайд №13

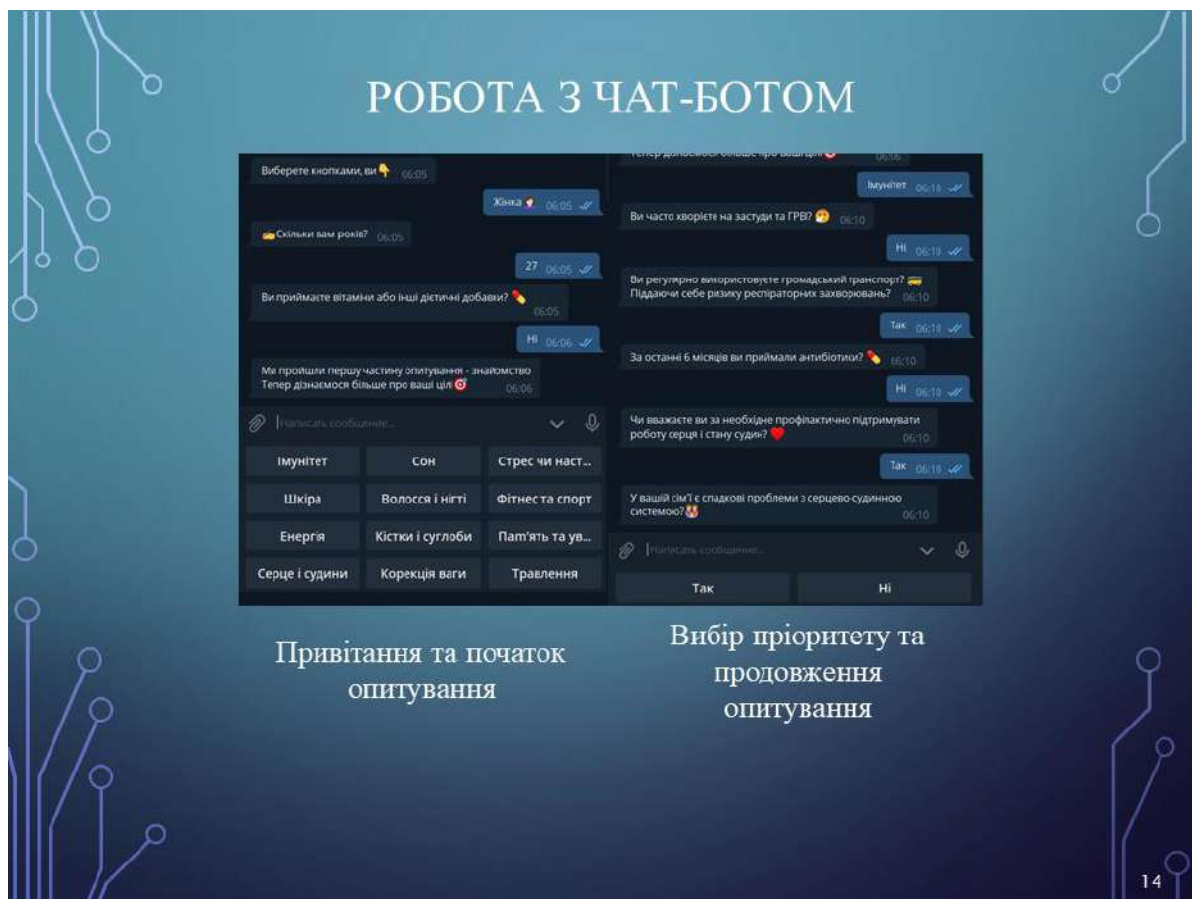


Рисунок Г.14 – Слайд №14

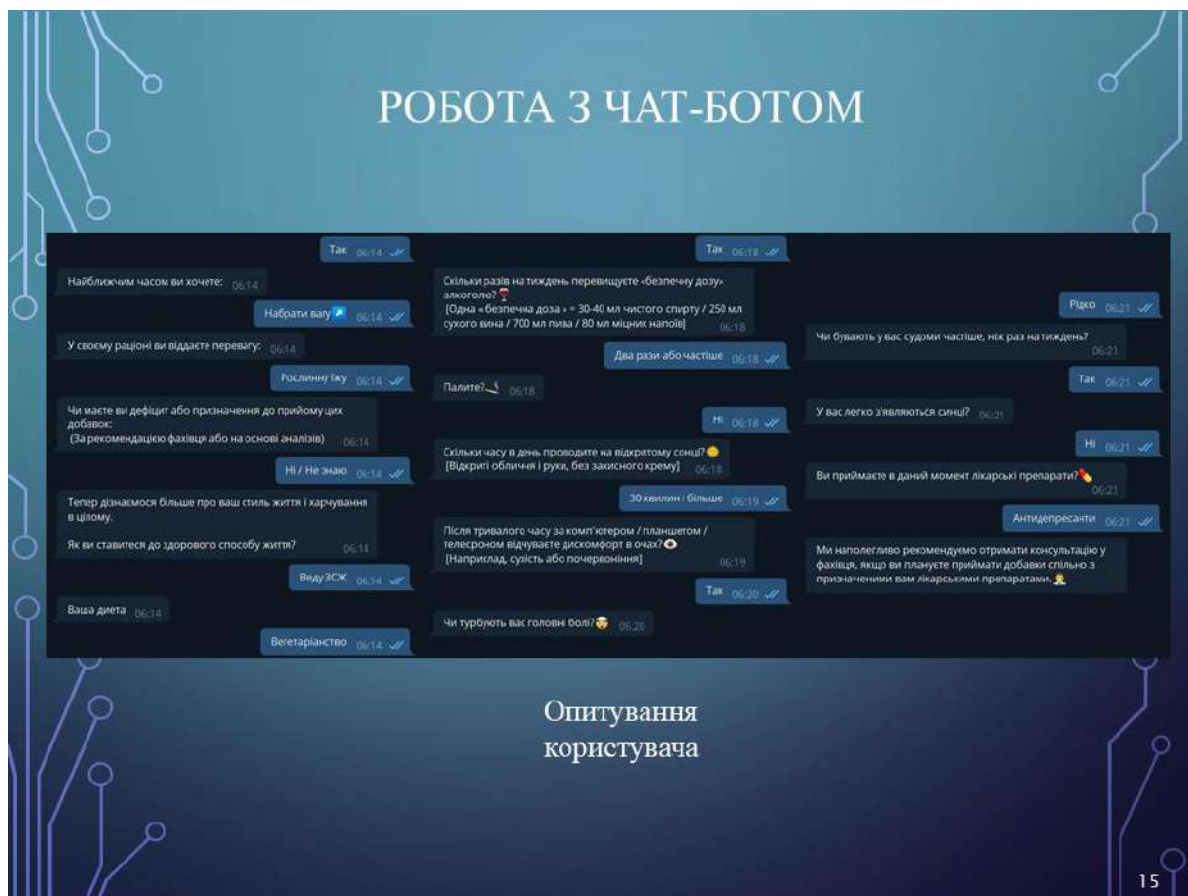


Рисунок Г.15 – Слайд №15

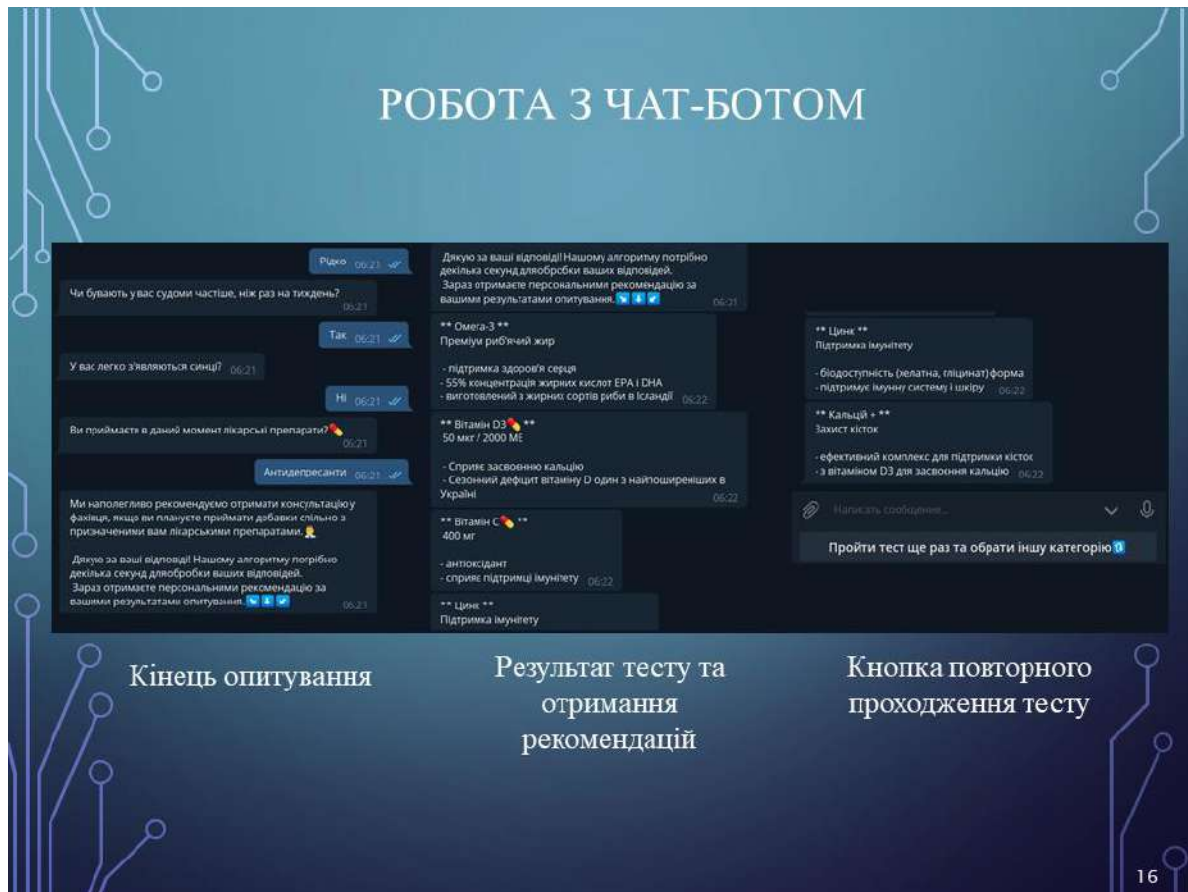


Рисунок Г.16 – Слайд №16

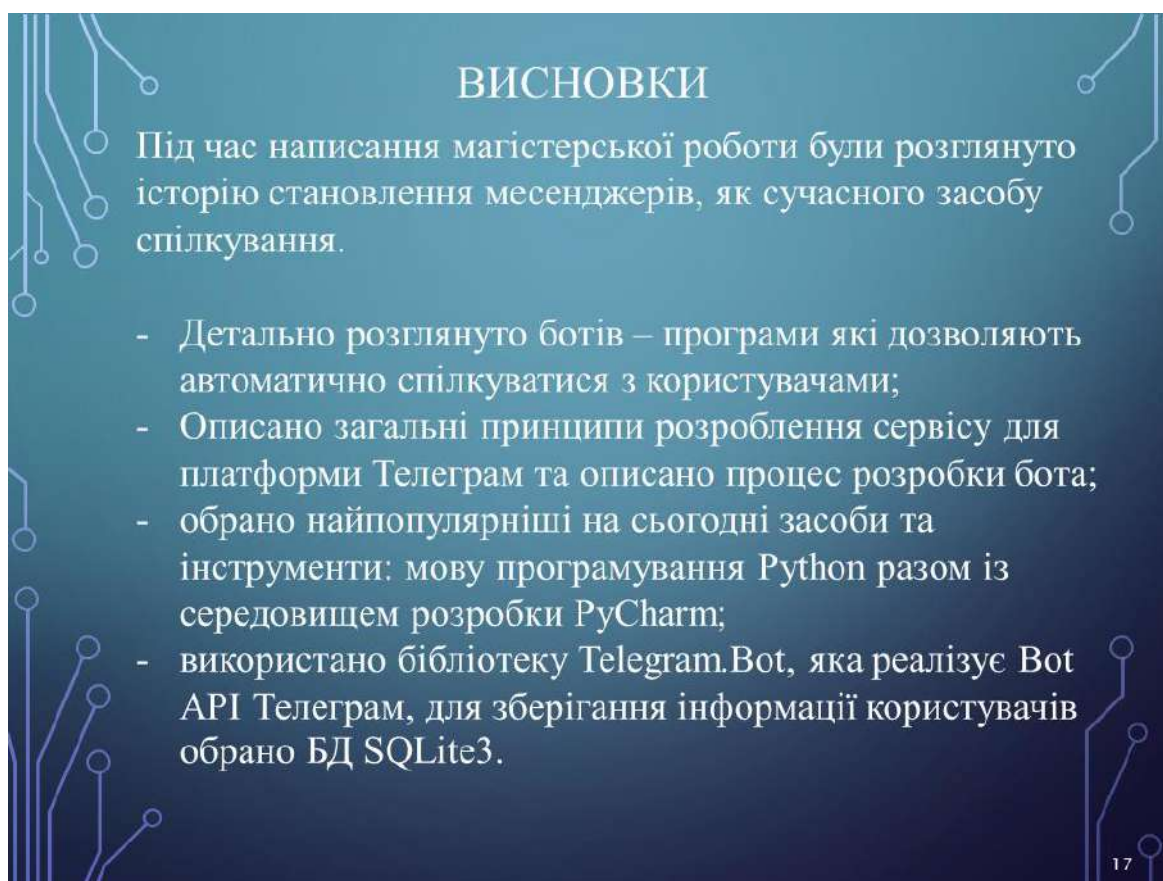


Рисунок Г.17 – Слайд №17

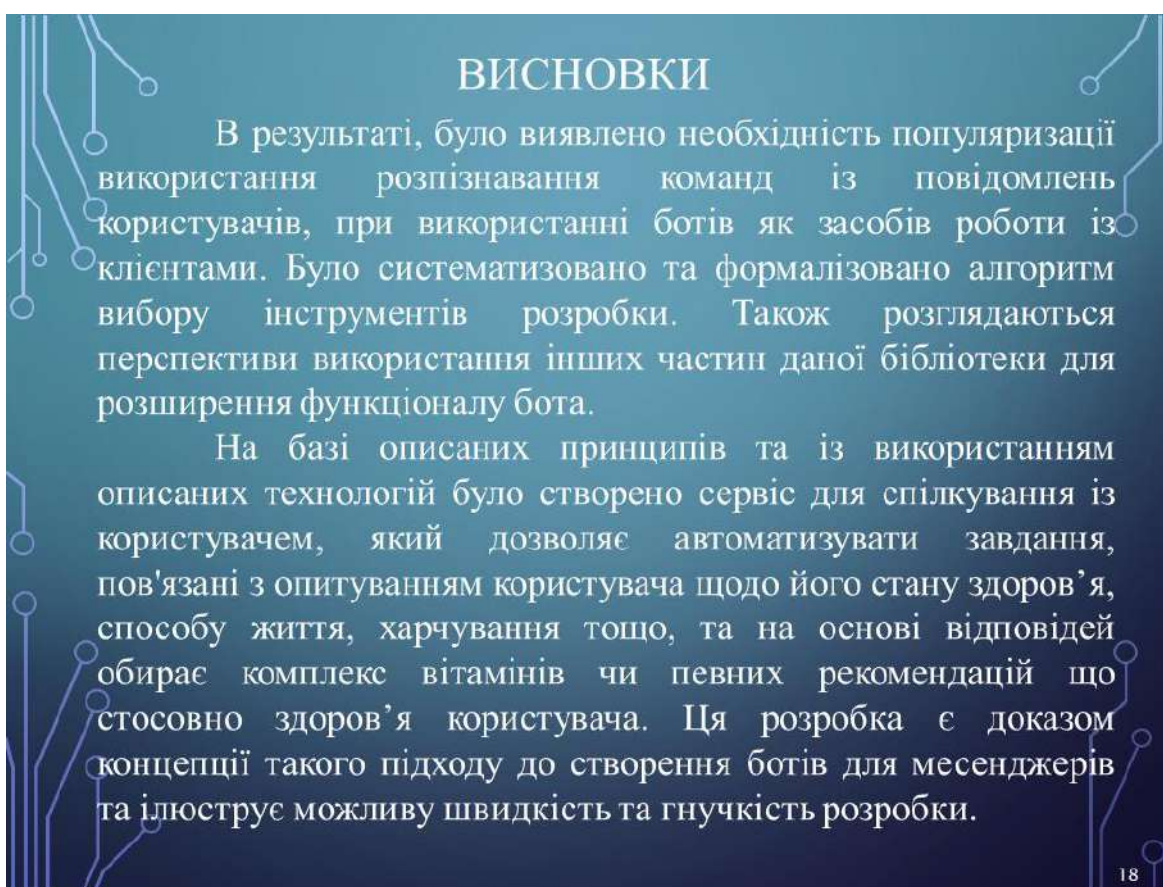


Рисунок Г.18 – Слайд №18



Рисунок Г.19 – Слайд №19