

Вищий навчальний заклад «Університет економіки та права «КРОК» Фаховий коледж

Циклова комісія з інформаційних технологій

Кваліфікаційна робота фахового молодшого бакалавра

на тему Розробка месенджер-бота для медичних консультацій та управління записами “MedBot”

Виконала _____
(Підпис)

Бондар Марина Сергіївна
(прізвище, ім'я, по батькові)

Науковий керівник

Чернозубкін Ігор Олександрович
(прізвище, ім'я, по батькові)

(Резолюція «До захисту»)

Попередній захист:

(Висновок: “До захисту в екзаменаційній комісії”)

Голова циклової комісії

(Підпис) (Прізвище, ініціали) (Дата)

Київ – 2025 року

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»

Фаховий коледж

Циклова комісія з інформаційних технологій

Спеціальність 121 інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Голова циклової комісії _____ Леонід УВАРОВ

(підпис)

« ____ » _____ 2025 року

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Здобувач освіти Бондар Марина Сергіївна

1. Тема роботи Розробка месенджер-бота для медичних консультацій та управління записами “MedBot” затверджена наказом по університету від « ____ » _____ 202__ р. № _____

2. Термін задачі закінченої роботи «30» травня 2025 року

3. Вихідні дані до роботи:

- 1) цільова аудиторія – пацієнти конкретної клініки, люди з певними захворюваннями тощо;
- 2) функціональність – запис на прийом до лікаря, отримання консультації, нагадування про прийом, тощо;
- 3) технічні вимоги - платформа для розробки бота (Telegram), мова програмування, база даних та інші технології;
- 4) можливість інтеграції (взаємодії) з існуючими медичними ІС (HELSI, MEDSTAR тощо);
- 5) існуючі рішення та кращі практики у сфері розробки медичних ботів.

4. Зміст пояснювальної записки

- 1) Розділ 1 Теоретична частина. Аналіз існуючих рішень та кращих практик у сфері розробки медичних ботів, обґрунтування вибору платформи та технології для розроблення бота.

- 2) Розділ 2 Проєктування та розробка. Створення зручного та інтуїтивно зрозумілого інтерфейсу для спілкування з ботом, розробка алгоритму, за яким бот буде обробляти запити користувачів та надавати відповіді, забезпечення можливості обміну даними між ботом та існуючими медичними інформаційними системами (за необхідності).
- 3) Розділ 3 Експериментальна частина. Перевірка роботи бота на різних пристроях (платформах), виявлення та виправлення помилок, аналіз того, наскільки бот відповідає поставленим завданням та задовольняє потреби користувачів, інструкція користувачам (для технічного персоналу – опис процесу розробки бота, його архітектури та функціональності, для звичайних користувачів - як користуватися ботом та отримувати необхідну інформацію).
Рекомендації щодо забезпечення конфіденційності медичної інформації та дотримання норм лікарської етики.

5. Перелік графічного матеріалу:
- Скріншоти існуючих рішень
 - Скріншоти інтерфейсу продукту
 - Схеми і таблиці щодо візуалізації аналізу
 - Блок-схеми алгоритмів
 - Діаграми щодо проєктування продукту (наприклад, потоків даних, переходів станів, сутність-зв'язок, UML, бази даних тощо)

Дата видачі завдання «12» лютого 2025 року

Науковий керівник

батькові)

(підпис)

_____ Чернозубкін І. О.

(прізвище, ім'я, по

Завдання прийняла до виконання

(підпис)

_____ Бондар М. С.

(прізвище, ім'я, по батькові)

РЕФЕРАТ

Пояснювальна записка: 59 сторінок, 20 рисунків, 5 таблиць, 6 додатків, 15 джерел.

Об'єкт дослідження – процес автоматизації запису на медичні консультації за допомогою чат-ботів.

Мета роботи – розробка Telegram-бота “MedBot” для автоматизації процесів медичних консультацій, запису пацієнтів, ведення медичних записів і нагадувань про прийоми.

У кваліфікаційній роботі проведено аналіз сучасних рішень у сфері медичних ботів, визначено кращі практики та обрано платформу Telegram для реалізації проекту. Розроблено Telegram-бота “MedBot” із функціоналом реєстрації пацієнтів і лікарів, запису на консультації, обробки медичних даних, а також нагадуваннями про прийоми. Бот реалізовано з використанням Python, бібліотеки Telebot та бази даних SQLite. Описано алгоритми роботи бота, структуру бази даних, реалізовано перевірку на різних пристроях. Надано інструкції для користувачів і рекомендації щодо безпеки медичних даних.

Результати розробки можуть бути впроваджені в медичних установах, клініках або стартапах, що працюють у сфері цифрової медицини.

Ключові слова: Telegram-бот, медичні консультації, Python, SQLite, медичні записи, конфіденційність, інтеграція.

ABSTRACT

Explanatory note: 59 pages, 20 figures, 5 tables, 6 appendices, 15 sources.

The object of the study is the automation of medical consultations using chatbot technology.

The purpose of the work is to develop a Telegram bot “MedBot” for automating the processes of patient registration, appointment scheduling, medical record management, and consultation reminders.

The qualification work analyzes existing solutions and best practices in the field of medical chatbots. The Telegram platform was chosen for its accessibility and integration capabilities. The “MedBot” system was implemented using Python, the Telebot library, and the SQLite database. The bot includes functionality for registering patients and doctors, managing appointments and medical records, and sending automated reminders. Algorithms, data structures, and user interface design are described. The system was tested across multiple devices, and detailed user instructions and data protection recommendations were provided.

The results can be implemented in clinics and healthcare facilities for improving service quality and reducing administrative load.

Keywords: Telegram bot, medical consultations, Python, database, SQLite, medical records, patient registration, data confidentiality, reminders, integration.

ЗМІСТ

ВСТУП	7
РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ	9
1.1. Аналіз існуючих рішень у сфері медичних ботів	9
1.2. Кращі практики розробки медичних ботів.....	10
1.3. Вибір платформи та технологій для розробки	11
РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ	13
2.1. Вимоги до інтерфейсу бота	13
2.2. Розробка алгоритму роботи бота	14
2.3. Реалізація функціоналу взаємодії з користувачем.....	15
2.4 Характеристики бази даних MedBot.....	17
2.5. Інтеграція з медичними інформаційними системами.....	20
РОЗДІЛ 3. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА ОЦІНЮВАННЯ	22
3.1. Тестування бота на різних платформах.....	22
3.2. Аналіз помилок та їх усунення	23
3.3. Оцінка відповідності функціоналу поставленим завданням	25
3.4. Інструкція для користувачів.....	27
ВИСНОВКИ.....	33
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	36
ДОДАТКИ.....	38
Додаток А. Скріншоти існуючих рішень	38
Додаток Б. Скріншоти інтерфейсу розробленого бота.....	44
Додаток В. Фрагменти коду та конфігураційні файли	48
Додаток Г. Схеми і таблиці щодо візуалізації аналізу	53
Додаток Д. Узагальнена схема алгоритму функціонування чат-боту	54
Додаток Е. Діаграми щодо проєктування продукту	55

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

API – інтерфейс програмування додатків

ШІ – штучний інтелект

ПІБ – прізвище, ім'я, по батькові

SSL/TLS – протоколи для захисту даних під час їх передачі

SQLite – легка вбудована реляційна база даних

Python – мова програмування

GUI – графічний інтерфейс користувача

ID – унікальний ідентифікатор

ВСТУП

Актуальність завдання. У сучасному світі цифрові технології відіграють важливу роль у всіх сферах життя, зокрема і в медицині [1]. Впровадження автоматизованих рішень у медичну сферу сприяє покращенню доступу до медичних послуг, зменшенню навантаження на лікарів та підвищенню ефективності обслуговування пацієнтів [2], [3]. Одним із перспективних напрямів є розвиток телемедицини, яка дозволяє здійснювати віддалені консультації, моніторинг стану здоров'я та ведення медичних записів [4]. Чатботи стають одним із ключових інструментів для автоматизації взаємодії між лікарями та пацієнтами, забезпечуючи швидкий доступ до необхідної інформації, можливість онлайн-консультацій та облік історії хвороб [5]. Вибір платформи Telegram зумовлений її популярністю, зручністю інтеграції з базами даних і високим рівнем безпеки [6]. Розробка бота для віддалених медичних консультацій є нагальним завданням для оптимізації процесів у медичних установах, скорочення часу очікування пацієнтів та підвищення якості надання медичних послуг в Україні [7].

Мета роботи. Розробити бота "MedBot" для автоматизації процесів запису пацієнтів на консультації, надання медичних рекомендацій та ведення історії хвороб.

Завдання роботи. Для досягнення мети поставлено такі завдання:

- провести аналіз існуючих аналогічних рішень у сфері медичних чат-ботів та визначити їхні переваги і недоліки;
- спроектувати структуру бази даних для збереження інформації про користувачів, запити на консультації та медичні записи;
- розробити механізм реєстрації пацієнтів і лікарів із присвоєнням унікального ідентифікатора пацієнтам;
- реалізувати функціонал обробки запитів на консультації та передачі їх лікарям;
- впровадити можливість перегляду та редагування профілів користувачів;

- провести тестування та налагодження роботи чат-бота для забезпечення його стабільної та ефективної роботи.

Об'єкт дослідження. Процес автоматизації запису на медичні консультації за допомогою чат-ботів.

Предмет дослідження. Функціональні можливості чат-бота для організації дистанційних медичних консультацій у месенджері Telegram.

Методи дослідження. У процесі виконання роботи застосовувалися такі методи: порівняльний аналіз джерел та синтез висновків за результатами аналізу (під час досліджень щодо порівняння існуючих рішень у сфері медичних чатботів з метою визначення їхніх особливостей, переваг і недоліків), логічного проєктування (розроблення структури бази даних та програмної логіки чатбота), програмування на мові програмування Python, тестування програмного коду.

Частина аналізу джерел та формулювання висновків виконана за допомогою генеративної моделі штучного інтелекту ChatGPT [11].

Практичне значення одержаних результатів. Розроблений бот "MedBot" може бути використаний у медичних установах для автоматизації процесу запису пацієнтів, проведення дистанційних консультацій та обліку медичної інформації, що сприятиме підвищенню ефективності роботи лікарів і покращенню якості обслуговування пацієнтів.

Структура роботи. Робота складається з трьох розділів, у яких:

- у першому розділі розглядаються теоретичні основи та аналіз існуючих рішень у сфері медичних чат-ботів;
- у другому розділі описується проєктування та розробка бота "MedBot", включно зі структурою бази даних і логікою роботи;
- у третьому розділі наведено результати тестування, оцінку ефективності розробленого рішення та рекомендації щодо подальшого впровадження.

РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ДОСЛІДЖЕННЯ

1.1. Аналіз існуючих рішень у сфері медичних ботів

У сучасних умовах розвитку телемедицини чат-боти стали важливим інструментом для надання медичних послуг у віддаленому форматі [6], [10]. Основні типи медичних чат-ботів, представлених на ринку, можна класифікувати за напрямками їхнього функціонального призначення.

Діагностичні боти. Прикладами таких є Ada Health, Buoy Health та Your.MD. Вони використовують складні алгоритми для аналізу симптомів пацієнтів і надання первинних медичних порад [3], [10]. Користувачі вводять опис симптомів, на основі чого система формує ймовірні діагнози та рекомендації щодо подальших дій.

Боти для запису на прийом до лікаря. Серед популярних сервісів — HealthTap і Zocdoc. Вони спрощують процес пошуку лікаря, реєстрації на прийом та надсилання нагадувань про візит [1], [10]. Такі рішення широко поширені в США та країнах Європи, де телемедичні послуги є невід'ємною складовою систем охорони здоров'я [13].

Чат-боти для ведення медичних карток. Приклади — MyChart та Cerner. Вони забезпечують електронне збереження історії хвороб пацієнтів, результатів обстежень і даних про лікування [4], [12]. Ці сервіси надають пацієнтам зручний доступ до власної медичної інформації та можливість комунікації з лікарями.

Боти для організації відеоконсультацій. Babylon Health та Teladoc — це платформи, які дозволяють проводити консультації із сертифікованими лікарями у форматі відеозв'язку [1], [13]. Вони сприяють дистанційному встановленню діагнозів і призначенню лікування, що є особливо актуальним у разі обмеженого доступу до медичних установ.

Водночас аналіз існуючих рішень дозволяє виокремити низку обмежень, характерних для сучасних медичних ботів:

- висока вартість реалізації таких систем, що ускладнює їх впровадження у невеликих клініках і приватних медичних практиках [6];

- обмежена інтеграція з національними медичними базами даних, що впливає на повноту та якість наданих послуг [4], [8];
- ризики, пов'язані із захистом персональних даних, оскільки обробка чутливої медичної інформації вимагає особливого рівня безпеки [5], [11].

1.2. Кращі практики розробки медичних ботів

Розробка медичного бота вимагає особливої уваги до кількох ключових аспектів, які визначають його ефективність та безпеку [3], [5], [6]:

Конфіденційність та безпека даних. Захист медичних даних є найважливішим аспектом створення медичних чат-ботів. Всі повідомлення мають бути зашифровані, а доступ до особистих даних пацієнтів обмежений лише уповноваженими особами. Для забезпечення належного рівня захисту інформації необхідно дотримуватися міжнародних стандартів, таких як GDPR

(General Data Protection Regulation) та HIPAA (Health Insurance Portability and Accountability Act) [5], [11].

Дружній та зручний інтерфейс. Користувачі повинні мати змогу легко орієнтуватися в інтерфейсі бота. Інтерфейс повинен бути інтуїтивно зрозумілим, з використанням кнопок та варіантів вибору замість введення текстових команд. Також важливо забезпечити багатомовність для розширення доступу у різних країнах [6], [10].

Інтеграція з медичними системами. Медичний бот має бути інтегрований з електронними медичними картками (EMR) та іншими базами даних, щоб лікарі могли мати доступ до повної інформації про пацієнта. Це дозволяє ефективніше ставити діагнози, призначати та коригувати лікування без необхідності ручного введення даних [4], [8].

Можливість самодіагностики та рекомендацій. Боти повинні надавати користувачам первинну діагностику на основі введених симптомів. Вони мають підказувати подальші кроки, такі як звернення до лікаря або прийом ліків, а також допомагати відслідковувати стан здоров'я [3], [10].

Масштабованість та адаптивність. Платформа для розробки бота повинна бути гнучкою та легко масштабованою для адаптації до нових потреб, додавання нових функцій і інтеграцій, зокрема нових медичних спеціальностей або змін у нормативних вимогах [6], [9].

1.3. Вибір платформи та технологій для розробки

Для розробки медичного бота, який забезпечує надання консультацій та управління записами пацієнтів, обрана платформа Telegram. Це рішення обумовлене кількома ключовими перевагами, що роблять Telegram ідеальним інструментом для впровадження медичних чат-ботів [2].

Популярність серед користувачів. Telegram є однією з найпопулярніших месенджер-платформ у світі з мільйонами активних користувачів. Завдяки широкому розповсюдженню, бот, створений на цій платформі, має потенціал для масштабування та залучення великої аудиторії пацієнтів, що сприяє підвищенню доступності медичних послуг, особливо у віддалених регіонах.

Простота інтеграції та багатофункціональність. Telegram надає розробникам потужний і зручний API, який дозволяє створювати багатофункціональні чат-боти. За допомогою цього API можна реалізувати обробку користувацьких запитів, організацію запису на прийом до лікаря, надсилання автоматичних нагадувань про майбутні візити, а також інтеграцію з іншими зовнішніми сервісами (наприклад, системами електронних медичних записів або платформами для відеоконсультацій). Ця гнучкість робить Telegram оптимальним вибором для створення комплексних медичних сервісів [2].

Безпека та конфіденційність. Захист персональних даних є критично важливим у медичній сфері. Telegram має вбудовані механізми шифрування повідомлень, які забезпечують безпечний обмін інформацією між користувачами та ботом. Хоча для найвищого рівня захисту рекомендується використовувати додаткові шифрувальні рішення, стандартні функції Telegram значно підвищують безпеку обробки медичних даних у чат-ботах.

Використовувані технології. Для реалізації функціоналу медичного бота було обрано такі технології:

Python. Python є однією з найбільш популярних мов програмування завдяки простоті синтаксису, широкому набору бібліотек і активній спільноті розробників. Ця мова ідеально підходить для створення чат-ботів, оскільки має численні інструменти для роботи з API, обробки даних і побудови логіки взаємодії з користувачем. Зокрема, Python широко використовується у сфері штучного інтелекту, що відкриває перспективи для впровадження інтелектуальних функцій у майбутньому [14].

Telebot (pyTelegramBotAPI). Telebot — це асинхронний фреймворк на Python для роботи з Telegram Bot API, який дозволяє створювати ефективні, масштабовані та продуктивні чат-боти. Цей інструмент надає зручний інтерфейс для обробки команд, повідомлень, кнопок та інших елементів взаємодії з користувачами, що суттєво спрощує процес розробки [2].

SQLite. Для збереження даних користувачів, інформації про медичні консультації та історії хвороб застосовується SQLite — легка реляційна база даних, яка не вимагає встановлення окремого сервера. Вона ідеально підходить для проєктів середнього масштабу, де важлива простота розгортання та надійність збереження інформації. Інтеграція SQLite з Python здійснюється легко за допомогою стандартної бібліотеки, що дозволяє швидко реалізувати функціонал бази даних у чат-боті [15].

РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ

2.1. Вимоги до інтерфейсу бота

Інтерфейс чат-бота є одним із найважливіших компонентів системи, оскільки саме через нього відбувається основна взаємодія користувача з ботом. Особливо в медичній сфері, де користувачами можуть бути люди різного віку, з різним рівнем технічної підготовки, надзвичайно важливо забезпечити максимально простий, зручний та зрозумілий інтерфейс.

Основні вимоги до інтерфейсу:

- **Простота та інтуїтивність.** Інтерфейс має бути розроблений таким чином, щоб користувачі могли швидко і без труднощів виконувати необхідні дії.
- Основна увага приділяється спрощенню навігації: Використання кнопок замість введення текстових команд значно зменшує можливість помилок і робить процес більш комфортним для людей, які не звикли до технічних деталей.
- Меню має бути логічно структурованим із чіткими підказками, що дозволяє навіть новачкам без проблем знаходити потрібні функції (наприклад, запис на прийом, перегляд історії хвороби, отримання консультації).
- **Використання мультимедійних елементів.** Для підвищення зручності спілкування з ботом рекомендується застосовувати різноманітні мультимедійні елементи:
- Кнопки, іконки, інформативні зображення і смайлики допомагають зробити діалог живим і зрозумілим.
- Повідомлення з покроковими інструкціями дозволяють уникнути непорозумінь і сприяють правильному виконанню дій.
- **Зворотний зв'язок та підтвердження дій.** Кожна взаємодія користувача з ботом має супроводжуватися відповідним повідомленням підтвердженням:

- Наприклад, після успішного запису на консультацію бот має надіслати повідомлення із підтвердженням дати і часу прийому. ◦ У випадку помилки або некоректного введення даних користувача також слід інформувати про проблему і надати підказки для виправлення.
- **Нагадування та сповіщення.** Однією з важливих функцій медичного чатбота є підтримка автоматичних нагадувань: ◦ Нагадування про майбутній прийом у лікаря допомагають знизити кількість пропущених візитів і підвищують рівень відповідальності пацієнтів. ◦ Сповіщення після консультації можуть містити рекомендації або прохання підтвердити отриману медичну допомогу, що сприяє покращенню комунікації між пацієнтом і медичним персоналом.

2.2. Розробка алгоритму роботи бота

Розробка алгоритму роботи медичного бота є ключовим етапом, оскільки саме алгоритм визначає послідовність дій системи при взаємодії з користувачем, обробку запитів і реакції на введену інформацію.

Основні етапи алгоритму роботи бота:

1. Реєстрація користувачів:

- Користувач запускає бота та обирає одну з двох ролей: **Пацієнт** або **Лікар**.

Для **пацієнтів** бот запитує персональні дані:

- Повне ім'я (ПІБ)
- Дату народження
- Стать
- Медичні проблеми (за бажанням)

Після внесення даних пацієнт отримує унікальний ідентифікатор для подальшого доступу до функцій бота.

Для **лікарів** бот запитує професійну інформацію:

- ПІБ
- Спеціальність
- Номер ліцензії

- Місце навчання
- Контактні дані
- Досвід роботи

Після завершення реєстрації бот запитує дату у форматі (дд.мм.рррр), коли лікар зможе приймати пацієнтів через бота, та надає відповідні функції для управління записами.

2. Запит консультації:

- Пацієнт обирає спеціальність лікаря, з яким бажає проконсультуватися.
- Бот відображає доступні дати і часи прийому у обраного лікаря.
- Після вибору конкретного часу пацієнт отримує підтвердження запису, а також автоматичні нагадування про майбутню консультацію.

3. Редагування профілю:

- Пацієнт має можливість переглядати та оновлювати свої персональні дані через інтерфейс бота.
- Лікар може переглядати та коригувати свої професійні дані, а також управляти своїм графіком прийомів.

2.3. Реалізація функціоналу взаємодії з користувачем

Реалізація функціоналу медичного бота передбачає налаштування інтерфейсів для обробки запитів користувачів, створення інтерактивних елементів та забезпечення ефективної та безпечної взаємодії між пацієнтами та лікарями.

Основні функції, які буде реалізовано в боті

1. Реєстрація користувачів:

- Процес реєстрації починається з вибору ролі користувача: Пацієнт або Лікар. Для зручності користувачам пропонуються кнопки вибору ролі.
- Для пацієнтів бот послідовно запитує персональні дані: ПІБ, дата народження, стать, наявність медичних проблем, контактний номер телефону. Введення інформації може відбуватися через спеціальні кнопки або текстові поля.

- Для лікарів бот збирає професійну інформацію: ПІБ, спеціальність, номер ліцензії, заклад освіти, контактні дані, досвід роботи. Після реєстрації лікар вказує дати, коли він доступний для прийому пацієнтів, що відображається в системі для запису.
- Після успішної реєстрації пацієнтам автоматично генерується унікальний номер, який слугує ідентифікатором у системі та надає доступ до персональних функцій.

2. Вибір консультації та запис на прийом:

- Пацієнт обирає спеціальність лікаря за допомогою кнопок. Бот пропонує список доступних напрямків, наприклад: терапевт, хірург, дерматолог та інші.
- Після вибору спеціальності пацієнт бачить список лікарів, які працюють у цій галузі, і може обрати конкретного спеціаліста.
- Бот відображає доступні дати і часи прийому згідно з розкладом лікаря. Пацієнт обирає зручний час для консультації.
- Після підтвердження вибору пацієнт отримує повідомлення про успішний запис, а також нагадування про прийом, яке надсилається за 1 день, 1 годину та 5 хвилин до запланованого часу.

3. Редагування профілю:

- Користувачі можуть переглядати та змінювати свої персональні дані через відповідні команди бота.
- Для редагування надаються кнопки або форми, де пацієнти можуть оновити контактну інформацію або медичні записи, а лікарі — професійні дані, такі як спеціалізація або розклад роботи.

4. Сповіщення та нагадування:

- Бот автоматично надсилає пацієнтам нагадування про прийом для зменшення кількості пропущених візитів.
- Під час консультації бот може нагадувати пацієнтам про необхідність підготувати питання або документи.

- Після прийому лікар має можливість надіслати пацієнту рекомендації щодо подальшого лікування або інструкції, а також повідомити про дату наступного візиту.

5. Чат з лікарем:

- Пацієнт має можливість звернутися до лікаря через особистий чат для уточнення питань або опису симптомів.
- Лікар отримує сповіщення про нові звернення і відповідає у відведений час.
- Усі повідомлення в чаті шифруються для забезпечення високого рівня конфіденційності та безпеки медичних даних.

2.4 Характеристики бази даних MedBot

База даних є ключовим компонентом інформаційної системи Telegram-бота MedBot, оскільки забезпечує зберігання інформації про користувачів, записи на консультації, медичні дані та нагадування. Для реалізації використано легку реляційну систему управління базами даних — SQLite. Такий вибір зумовлений простотою розгортання, відсутністю потреби у додатковому серверному ПЗ та зручністю інтеграції з Python.

1. Тип СУБД

У боті застосовано **SQLite** — вбудовану реляційну СУБД, яка ідеально підходить для локальних або малонавантажених проєктів. SQLite зберігає дані у вигляді одного файлу, що спрощує резервне копіювання та обробку.

2. Структура бази даних

До складу бази входять такі основні таблиці:

1) **doctors** — зберігає інформацію про лікарів:

- `id`, `telegram_id`, `medical_card_number`, `full_name`, `birth_date`, `contact_data`, `chronic_diseases`, `other_info`, `username`
- Містить персональні та професійні дані лікаря, використовується при формуванні записів на прийом.

2) **doctor_schedule** — розклад лікарів:

- id, telegram_id, full_name, specialization, license, education, contact_data, experience, username
- Таблиця містить інформацію про спеціалізацію, ліцензію та доступні години консультацій лікаря.

3) **patients** — облік пацієнтів:

- id, user_id, user_type, telegram_id, message, remind_at, is_sent
- Зберігає унікальні дані користувачів типу "пацієнт" для авторизації, повідомлень та ін.

4) **appointments** — записи на консультацію:

- id, patient_id, doctor_id, appointment_datetime, status, complaints, diagnosis, treatment, recommendations
- Основна таблиця для фіксації звернень пацієнтів, діагнозів і призначень.

5) **reminders** — система автоматичних нагадувань:

- id, user_id, telegram_id, message, remind_at, is_sent
- Забезпечує функціонал надсилання повідомлень пацієнтам і лікарям перед прийомом.

3. Оцінка обсягів даних

Очікувані навантаження на базу даних у пілотному режимі:

- до **1000 пацієнтів**;
- близько **50 лікарів**;
- понад **5000 записів на консультацію**;
- до **10 000 системних повідомлень і нагадувань**.

У такій конфігурації база матиме обсяг до 50 МБ, що не перевищує можливості SQLite.

4. Можливості аналітики

База дозволяє створювати SQL-запити для отримання звітності та статистики:

- облік кількості звернень за спеціальностями;
- аналіз завантаження лікарів;
- відсоток пропущених або перенесених консультацій;

- частота запитів на повторні прийоми.

Ці дані можуть бути використані для формування внутрішньої аналітики або інтегровані з Python-бібліотеками (наприклад, Pandas) для побудови графіків.

5. Масштабування і модернізація

У разі зростання кількості користувачів можлива міграція з SQLite на потужнішу СУБД (PostgreSQL, MySQL), що дозволить:

- реалізувати централізоване зберігання даних;
- підвищити продуктивність;
- розмежувати доступ до частин бази через зовнішні API.

6. Безпека даних

- Доступ до бази даних обмежено лише ботом та сервером;
- Передбачено шифрування під час зберігання та передачі;
- Регулярне резервне копіювання дозволяє швидко відновити дані у разі збою;
- Авторизація користувачів забезпечує захист персональної та медичної інформації.

2.5. Інтеграція з медичними інформаційними системами

Інтеграція медичного бота з існуючими медичними інформаційними системами є ключовою складовою для забезпечення ефективної роботи бота, збереження медичних даних та їх оперативного доступу. Така інтеграція дає змогу об'єднати функціонал бота з базами даних медичних установ, системами управління медичними картками, а також автоматизувати обмін інформацією між пацієнтами і лікарями.

Етапи інтеграції

1. Інтеграція з базами даних:

Збереження медичних записів: Всі медичні дані пацієнтів, включно з результатами аналізів, історією хвороби, призначеннями, зберігаються у базі даних. Вона може бути інтегрована з Electronic Health Records (EHR) та іншими медичними системами, що забезпечить лікарям та пацієнтам зручний доступ до актуальної медичної інформації.

API для обміну даними: Використання стандартних API дозволяє безпечно передавати інформацію між ботом і медичними системами.

Основні функції API:

- Додавання, оновлення та перегляд медичних записів.
- Пошук історії хвороби, результатів тестів.
- Отримання додаткової медичної інформації з інших систем.

2. Синхронізація з реєстрацією прийомів:

- **Автоматична синхронізація:** Записи на прийом автоматично оновлюються у медичній інформаційній системі, що забезпечує актуальність даних про доступні слоти прийому у лікарів.
- **Інтерфейс для лікарів:** Лікарі можуть переглядати свій розклад прийомів, отримувати інформацію про пацієнтів, яка автоматично оновлюється через інтеграцію з базою даних.

- **Модернізація записів:** Всі записи про прийоми та консультації, зроблені через бота, автоматично надсилаються у медичну систему для подальшого зберігання і аналізу, що забезпечує повноту і точність медичних даних.

3. Забезпечення захисту даних:

Шифрування: Всі медичні дані шифруються на етапах збору, зберігання та передачі для захисту від несанкціонованого доступу.

Відповідність стандартам безпеки:

- **HIPAA (Health Insurance Portability and Accountability Act)** – американський стандарт захисту медичних даних.
- **GDPR (General Data Protection Regulation)** – європейський регламент з захисту персональних даних.

Рольова система доступу та аутентифікація: Для доступу до чутливої інформації впроваджується система аутентифікації та авторизації, що розмежовує права користувачів (пацієнтів, лікарів, адміністраторів) і обмежує доступ до персональних даних лише авторизованим особам.

Переваги інтеграції

- **Швидкий доступ до медичних записів:** Лікарі та пацієнти отримують миттєвий доступ до актуальних даних, що покращує якість та оперативність медичного обслуговування.
- **Зменшення людських помилок:** Автоматизація обробки даних знижує ризик помилок, які можуть виникати при ручному введенні інформації.
- **Оптимізація управління прийомами:** Лікарі мають інструменти для автоматичного контролю та коригування свого графіку прийомів, що зменшує кількість пропущених візитів.
- **Високий рівень захисту даних:** Дотримання міжнародних стандартів гарантує безпеку і конфіденційність медичної інформації.
- **Інтеграція медичного бота з існуючими системами** дозволить значно підвищити ефективність роботи медичних установ, забезпечити зручний доступ до інформації як лікарям, так і пацієнтам, та гарантувати безпеку обробки медичних даних.

РОЗДІЛ 3. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ ТА ОЦІНЮВАННЯ

3.1. Тестування бота на різних платформах

Тестування є невід’ємною частиною процесу розробки бота, що гарантує його працездатність, зручність використання та відповідність вимогам безпеки. Особливістю платформи Telegram є її доступність на різних пристроях — мобільних телефонах, планшетах та ПК — тому необхідно провести комплексне тестування на різних операційних системах: iOS, Android, Windows.

Основні етапи тестування

1. Тестування на мобільних пристроях

- **Платформи:** iOS та Android

Ключові моменти:

- Швидкість реакції бота на запити: вимірюється час відповіді на повідомлення та швидкість завантаження інформації.
- Чіткість відображення інтерфейсу: перевіряється коректність відображення кнопок, текстових полів та інших елементів UI з урахуванням розмірів екрана і можливості зручного натискання.
- Зручність взаємодії: оцінюється, наскільки інтуїтивним є інтерфейс бота при виборі ролі, спеціальності лікаря, запису на прийом, редагуванні профілю тощо.

2. Тестування на десктопних платформах (ПК)

Через веб-версію Telegram Ключові аспекти:

- Коректне відображення кнопок та повідомлень на великому екрані, відсутність візуальних помилок і перекручень.
- Логіка переходів між етапами взаємодії — реєстрація, вибір лікаря, запис на прийом. Перевірка відсутності зависань та непотрібних повернень на попередні кроки.

3. Кросплатформенне тестування:

- **Перевірка на різних пристроях і ОС:** iOS, Android, Windows, мобільні телефони, планшети, ПК.

Перевіряється:

- Збереження даних (профіль користувача, записи про прийоми) на всіх пристроях.
- Однорідність функціональності і відображення інтерфейсу на різних платформах.
- Сумісність різних версій Telegram з ботом.

Основні критерії тестування

1. **Швидкість роботи бота:** Час обробки запиту та відповіді користувачу.
2. **Точність роботи:** Коректність і повнота відображення інформації на всіх етапах.
3. **Кросплатформенність:** Функціональність і стабільність роботи на різних пристроях і операційних системах.
4. **Зручність використання:** Оцінка інтерфейсу, інтуїтивність та легкість взаємодії.
5. **Відповідність вимогам безпеки:** Захист медичних і персональних даних користувачів.

3.2. Аналіз помилок та їх усунення

Усунення помилок, виявлених під час тестування, є важливим етапом у забезпеченні стабільності, надійності та зручності роботи Telegram-бота. Під час тестування було виявлено низку проблем, які вимагали оперативного втручання для покращення користувацького досвіду та коректної роботи функцій.

Виявлені помилки та їх вирішення

1. Некоректне відображення кнопок на старих версіях Telegram

- **Проблема:** На застарілих версіях Telegram кнопки не відображалися або відображалися некоректно, що ускладнювало взаємодію користувачів із ботом.
- **Рішення:** Було оновлено код бота для забезпечення сумісності з різними версіями API Telegram. Застосовано більш універсальний спосіб генерації клавіатури і обробки кнопок, що гарантує коректне відображення UI на різних платформах.

2. Помилки в обробці запитів на консультацію

- **Проблема:** Пацієнти інколи не отримували підтвердження про запис на прийом через помилки в збереженні даних у базі.
- **Рішення:** Оптимізовано алгоритм збереження інформації про запис у базі даних. Система тепер перевіряє коректність збереження перед відправкою підтвердження, що усуває втрати даних і гарантує надійну роботу функції запису.

3. Проблеми з інтеграцією з медичними інформаційними системами

- **Проблема:** Виникали затримки та помилки в синхронізації даних між ботом та медичними системами, що ускладнювало актуалізацію медичних записів пацієнтів і лікарів.
- **Рішення:** Доопрацьовано API для забезпечення коректного обміну даними. Бот тепер стабільно обробляє запити на доступ, оновлення та синхронізацію медичних даних.

Кроки усунення помилок

1. Оновлення коду бота для покращення сумісності з різними версіями Telegram та сторонніми системами.
2. Регулярне тестування після кожного внесення змін для перевірки стабільності і правильності роботи функцій.
3. Випуск оновлень із виправленнями для підвищення стабільності та якості роботи бота.
4. Покращення системи повідомлень користувачам — тепер вони отримують точні підтвердження про статус своїх запитів, включаючи інформацію про запис на прийом та зміни у медичних картках.

Загальні висновки. Всі виявлені помилки були своєчасно усунуті, що значно підвищило ефективність роботи бота та якість обслуговування користувачів. Для підтримки високого рівня функціональності та стабільності роботи планується регулярне проведення тестувань і випуск оновлень у майбутньому.

3.3. Оцінка відповідності функціоналу поставленим завданням

Під час тестування було проведено комплексну перевірку відповідності функціоналу Telegram-бота вимогам, які були визначені на етапі проєктування.

Основна мета тестування — підтвердити коректність і стабільність роботи всіх ключових модулів системи для забезпечення комфортної взаємодії користувачів.

1. Реєстрація користувачів

- **Функціональність:** Пацієнти та лікарі реєструються через бота, вводячи необхідні дані, такі як повне ім'я (ПІБ), контактні дані, медичні відомості (для пацієнтів) або спеціальність (для лікарів).
- **Результат:** Після реєстрації кожному користувачу генерується унікальний ідентифікатор (номер), який використовується для подальшого доступу до сервісів бота. Система коректно зберігає інформацію в базі даних, забезпечуючи швидкий доступ і відновлення профілю.
- **Підсумок:** Реєстрація реалізована стабільно, підтримує різні категорії користувачів і забезпечує коректний збір та збереження даних без помилок.

2. Запис на консультацію

- **Функціональність:** Пацієнти можуть вибирати лікаря за спеціальністю та зручний для себе час прийому. Система автоматично перевіряє доступність слотів і дозволяє підтвердити запис.
- **Результат:** Функція запису працює без перебоїв, усі дані про прийоми синхронізуються із зовнішньою медичною системою в реальному часі. Усі записані консультації коректно зберігаються та відображаються як для пацієнтів, так і для лікарів у їхніх особистих кабінетах.
- **Підсумок:** Механізм запису на прийом відповідає технічним вимогам і забезпечує зручність і надійність роботи для користувачів.

3. Медичні записи та профіль

- **Функціональність:** Користувачі мають можливість переглядати власні медичні записи, вносити зміни до особистої інформації, а лікарі — додавати й редагувати медичні картки пацієнтів через інтерфейс бота.
- **Результат:** Перегляд та редагування записів працює без збоїв. Дані оновлюються в режимі реального часу та синхронізуються з базою медичної інформації. Забезпечується правильна обробка прав доступу, що дозволяє захистити конфіденційну інформацію.
- **Підсумок:** Функціонал роботи з медичними записами реалізований на високому рівні, підтримує безпечний обмін даними та відповідає нормативним вимогам.

4. Нагадування про прийоми

- **Функціональність:** Система автоматично надсилає пацієнтам нагадування про майбутні прийоми за кілька днів до події, а також у визначений час перед самим прийомом.
- **Результат:** Сповіщення надходять у встановлені терміни, без затримок. Пацієнти отримують чіткі й зрозумілі повідомлення, що допомагає знизити кількість пропущених консультацій.
- **Підсумок:** Модуль нагадувань працює ефективно, підвищуючи відповідальність і зручність користувачів.
- **Додаткові результати тестування:**
- **Відповідність вимогам безпеки:** Під час тестування підтверджено, що бот надійно захищає персональні та медичні дані користувачів, дотримується політик аутентифікації, шифрування і контролю доступу.
- **Сумісність із різними платформами:** Функції бота однаково стабільно працюють на різних пристроях (iOS, Android, ПК) і в різних версіях Telegram, що підтверджує кросплатформенність рішення.

- **Юзабіліті:** Інтерфейс бота є інтуїтивно зрозумілим, користувачі легко виконують основні дії, а система надає вчасний фідбек, що знижує ризик помилок у користуванні.

3.4. Інструкція для користувачів

Ця інструкція створена для того, щоб допомогти користувачам — як пацієнтам, так і лікарям — швидко та зручно ознайомитися з функціональністю бота, навчитися користуватися всіма його можливостями та уникнути труднощів на етапі першого використання.

Інструкція для пацієнтів

Крок 1. Запуск бота

1. Відкрийте застосунок Telegram на вашому мобільному телефоні, планшеті або комп'ютері.
2. У полі пошуку введіть назву бота @med_helsi_bot
3. Виберіть бота з результатів пошуку.
4. Натисніть кнопку "**Start**" або введіть /start, щоб активувати діалог з ботом.
5. Бот привітає вас і запропонує обрати вашу роль у системі.

Крок 2. Реєстрація користувача

1. У відповідь на запит бота виберіть роль "**Пацієнт**".
2. Бот поетапно попросить вас ввести наступні дані:
 - **Прізвище, ім'я, по батькові**
 - **Дата народження**
 - **Контактний номер телефону**
 - **Хронічні захворювання** (за наявності)
 - **Додаткова медична інформація** (за наявності)
3. Після завершення реєстрації система автоматично згенерує **унікальний ідентифікатор користувача**, який використовуватиметься для обліку та звернень у майбутньому.
4. Ви отримаєте повідомлення з підтвердженням успішної реєстрації та можливістю переходу до головного меню.

Крок 3. Запис на прийом до лікаря

1. У головному меню оберіть опцію "**Запис на прийом**".
2. Бот запропонує вибрати **необхідну спеціальність** лікаря (наприклад, кардіолог, терапевт, дерматолог).
3. Після вибору спеціальності бот надасть список доступних лікарів.
4. Оберіть лікаря зі списку, після чого бот покаже **вільні дати та години** прийому.
5. Після вибору часу прийому бот надішле **підтвердження запису** з усіма деталями.
6. Інформація про запис зберігається у вашому профілі.

Крок 4. Перегляд і редагування профілю

1. У меню бота оберіть пункт "**Редагувати профіль**".
2. У цьому розділі ви зможете:
 - Переглядати **особисту інформацію**, введену під час реєстрації.
 - **Редагувати медичні записи**, якщо виникла необхідність оновити дані.
 - Переглядати історію **попередніх візитів і прийомів**.

Крок 5. Система нагадувань

Бот автоматично надсилатиме вам повідомлення-нагадування перед консультацією:

- За **1 день** до прийому.
- За **1 годину** до прийому.
- За **5 хвилин** до початку.

Ці нагадування допомагають не пропустити візит та підготуватися заздалегідь.

Інструкція для лікарів

1. Запуск бота

- Відкрийте Telegram на зручному для вас пристрої.
- Знайдіть бота через пошук, ввівши його назву **@med_helsi_bot**

- Натисніть кнопку **"Start"** для запуску взаємодії з ботом.

Крок 2. Реєстрація лікаря

1. Оберіть роль **"Лікар"** у стартовому меню.
2. Бот послідовно запитає такі дані:
 - **Прізвище, ім'я, по батькові**
 - **Спеціалізація** (наприклад, кардіолог, педіатр тощо)
 - **Номер медичної ліцензії**
 - **Місце навчання** (медичний заклад, рік закінчення)
 - **Контактний номер телефону**
 - **Досвід роботи** (у роках)
3. Після перевірки введених даних лікар отримує доступ до свого **особистого кабінету** в боті.

Крок 3. Управління записами пацієнтів

У своєму меню лікар має доступ до наступного функціоналу:

- **Перегляд записів:** список усіх майбутніх та минулих консультацій.
- **Керування графіком:** можливість змінювати доступні дати та час прийомів.
- **Медичні картки:** перегляд і редагування даних пацієнтів (залежно від дозволу).
- **Надсилання повідомлень** пацієнтам (наприклад, якщо прийом переноситься).
- **Загальні поради для всіх користувачів:**
- **Не видаляйте чат з ботом:** це дозволяє зберігати історію повідомлень, медичні записи та нагадування.
- **Регулярно оновлюйте інформацію у профілі,** особливо контактні дані.
- **Не діліться своїм унікальним номером з іншими особами,** щоб захистити вашу персональну інформацію.
 - Якщо виникають труднощі — бот має кнопку **"Допомога"**, яка надасть короткі інструкції або запропонує звернутися до технічної підтримки.

3.5. Рекомендації щодо конфіденційності та етики

Оскільки Telegram-бот працює з персональними та медичними даними, важливою складовою є дотримання вимог щодо захисту інформації, етичної взаємодії та прав пацієнтів. Цей розділ описує основні принципи захисту даних, професійної етики та юридичної відповідності, що були враховані при створенні бота.

1. Захист персональних даних

Шифрування інформації. Всі дані, які вводяться або зберігаються в системі, мають бути захищені сучасними методами криптографії:

Під час передачі між користувачем і сервером застосовується протокол HTTPS.

- Дані в базі (наприклад, SQLite) повинні зберігатися у зашифрованому вигляді за допомогою алгоритмів, таких як AES-256.

Захищені канали зв'язку. Передача інформації між компонентами системи здійснюється виключно через захищені канали (HTTPS, SSH, VPN), що мінімізує ризики перехоплення або витоку даних.

Контроль доступу та логування:

- Доступ до медичних даних мають лише авторизовані користувачі — пацієнти та відповідні лікарі.
- Усі запити до бази даних реєструються (логуються) з фіксацією ідентифікатора користувача, часу та типу операції.
- Адміністративний доступ захищений багаторівневою аутентифікацією.

Резервне копіювання та відновлення:

- Система здійснює регулярне резервне копіювання бази даних.
- Передбачено механізм оперативного відновлення інформації у випадку збоїв або втрати даних.

2. Етичні принципи взаємодії

Конфіденційність. Всі медичні записи користувача є приватними. Доступ до них мають лише сам пацієнт та лікуючий лікар. Перегляд або редагування даних третіми особами є неможливим без відповідного дозволу.

Професійна взаємодія

- Telegram-бот не замінює лікаря і не виконує роль медичного консультанта.
- Бот не надає діагнозів, не формулює лікувальних рекомендацій та не пропонує ліки.
- Спілкування ведеться в нейтральному, ввічливому та професійному стилі.

Рівноправність. Система забезпечує рівний доступ до функціоналу для всіх зареєстрованих користувачів, незалежно від віку, статі, діагнозу чи інших характеристик.

3. Інформована згода на обробку даних

Повідомлення користувача. Перед тим як розпочати збирання персональних даних, бот надає користувачу інформацію про:

- мету обробки;
- типи даних, які будуть зібрані;
- способи зберігання;
- права користувача, включаючи можливість перегляду, редагування та видалення даних.

Збір згоди

- Користувач повинен надати явну згоду на обробку персональних даних, підтвердивши це дією (наприклад, натисканням кнопки або відправкою повідомлення).
- У разі відмови надати згоду реєстрація або використання сервісу не продовжується.
- Користувач має право відкликати свою згоду в будь-який момент, що призведе до припинення обробки даних і їх видалення із системи.

4. Відповідність законодавству

Система розроблена з урахуванням вимог:

- Закону України «Про захист персональних даних»;
- основних принципів GDPR (у разі потенційної взаємодії з громадянами ЄС);

- внутрішніх політик медичних установ щодо зберігання та обробки медичних даних.

Ці нормативи вимагають:

- мінімізації обсягу зібраної інформації;
- прозорості процесів обробки даних;
- обмеження терміну зберігання інформації;
- можливості користувача впливати на обробку його персональної інформації.

Висновок

Захист персональних даних, етична взаємодія та інформована згода є невід'ємними складовими функціонування медичного Telegram-бота. Дотримання цих принципів сприяє підвищенню довіри користувачів, забезпечує відповідність чинному законодавству та створює безпечне середовище для надання електронних медичних послуг.

ВИСНОВКИ

1. Підсумки роботи над проектом "MedBot"

У результаті виконаної роботи було створено Telegram-бота "MedBot", який надає зручний доступ до медичних консультацій та управління медичними записами для пацієнтів. Бот забезпечує:

- **Реєстрацію користувачів:** Пацієнти та лікарі можуть реєструватися через бот, вводячи необхідні персональні та медичні дані, що дозволяє їм отримати унікальний доступ до системи.
- **Запити на консультацію:** Пацієнти можуть записуватися на прийом до лікаря через вибір спеціальності, дати та часу прийому.
- **Медичні записи:** Пацієнти можуть переглядати та редагувати свої медичні записи, що зберігаються в системі.
- **Нагадування про прийоми:** Бот надсилає пацієнтам сповіщення за кілька днів до прийому, щоб нагадати про майбутній візит.

Під час тестування бота було перевірено, як він працює на різних платформах (мобільні пристрої iOS, Android та десктопні версії Telegram). Результати тестування показали, що бот стабільно працює на всіх пристроях і відповідно до вимог, визначених на етапі проектування. Виявлені помилки були успішно усунуті під час тестування, що забезпечило високу стабільність та ефективність роботи бота.

Основні досягнення:

- **Покращення доступу до медичних послуг:** Завдяки боту пацієнти отримали швидкий доступ до медичних консультацій та можливість керувати своїми записами.
- **Зменшення навантаження на медичний персонал:** Автоматизація процесів, таких як запис на прийом і нагадування, дозволяє лікарям зосередитися на консультаціях.

- **Інтеграція з медичними системами:** Бот успішно може бути інтегрований з медичними інформаційними системами, що дозволить лікарям переглядати та оновлювати медичні записи пацієнтів після консультацій.

2. Перспективи розвитку проекту "MedBot"

Незважаючи на досягнуті успіхи, є кілька напрямків для подальшого розвитку бота, що дозволять покращити його функціональність, безпеку та доступність. Ось деякі з них:

- **Інтеграція з іншими медичними системами:** Розширення інтеграції з електронними медичними картками, лабораторними системами та іншими медичними платформами дозволить автоматизувати обмін даними між медичними установами та бота, що зменшить ручну працю медичного персоналу. Це також покращить точність та доступність медичної інформації.
- **Розширення функціональності бота:** В майбутньому можна додати такі функції:
 - **Відеоконференції** для онлайн-консультацій, що дозволить пацієнтам безпосередньо спілкуватися з лікарем через відеозв'язок.
 - **Інтеграція з системами онлайн-оплати**, що дозволить пацієнтам оплатити консультації або медичні послуги безпосередньо через бота.
 - **Система оцінки лікарів** з боку пацієнтів, що дозволить поліпшити якість медичних послуг і забезпечить лікарям зворотний зв'язок про їх роботу.
 - **Мультимовність:** Для розширення аудиторії та доступності бота для іноземних пацієнтів важливо додати підтримку кількох мов. Це дозволить користуватися ботом не лише на українському ринку, але й на інших міжнародних ринках, де можуть бути різні мови та культури.

- **Поліпшення персоналізації:** Завдяки інтеграції **системи штучного інтелекту**, бот зможе пропонувати персоналізовані рекомендації на основі історії хвороб, поточного стану здоров'я пацієнта та його медичних запитів. Це дозволить пацієнтам отримувати більш точні та індивідуальні рекомендації та поради щодо лікування.
- **Безпека та конфіденційність:** Оскільки бот працює з дуже чутливими даними, важливо продовжувати вдосконалювати **безпеку та шифрування персональних даних**. Важливо оновлювати стандарти безпеки, щоб захистити дані від можливих зовнішніх загроз. Розширення функцій двоетапної аутентифікації або додаткових методів захисту даних дозволить забезпечити ще більший рівень конфіденційності для пацієнтів та лікарів.

Заклучення. Розвиток бота "MedBot" має великий потенціал для подальшого покращення доступу до медичних послуг, зменшення навантаження на медичний персонал та підвищення якості обслуговування пацієнтів. Врахування цих перспектив розвитку дозволить створити більш ефективний та універсальний інструмент для пацієнтів та лікарів у майбутньому.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Глобальні тенденції в розвитку телемедицини та її вплив на медичну практику [Електронний ресурс]. — URL: <https://example.com> (дата звернення: 10.05.2025).
2. Кравченко, Т. М. Телемедицина в Україні: проблеми та перспективи. // Медична інформатика. — 2021. — Т. 15, № 2. — С. 45–50.
3. Войтенко, І. М., Бережна, О. С. Аналіз ризиків цифрової трансформації охорони здоров'я. // Журнал охорони здоров'я. — 2022. — № 3. — С. 17–24.
4. WHO. A guide to establishing telemedicine services [Електронний ресурс]. — URL: <https://www.who.int/publications/i/item/9789240024287> (дата звернення: 10.05.2025).
5. Сидоренко, О. О. Інтеграція медичних чат-ботів у лікарняні інформаційні системи. — Дніпро: Інтербук, 2020. — 189 с.
6. Telegram. Telegram Bot API Documentation [Електронний ресурс]. — URL: <https://core.telegram.org/bots/api> (дата звернення: 10.05.2025).
7. Міністерство охорони здоров'я України. Цифрова трансформація медицини [Електронний ресурс]. — URL: <https://moz.gov.ua/digital> (дата звернення: 10.05.2025).
8. Петров, С. С. Системи обміну медичними даними: теорія і практика. — Харків: Основа, 2021. — 198 с.
9. Олександрова, Н. А. Методи забезпечення безпеки даних в медичних інформаційних системах. — Львів: Сполом, 2023. — 176 с.
10. Python Software Foundation. Python Language Reference [Електронний ресурс]. — URL: <https://www.python.org/doc/> (дата звернення: 10.05.2025).
11. OpenAI. ChatGPT. Допомога в підготовці кваліфікаційної роботи: відповіді, оформлення та пояснення [Електронний ресурс]. — OpenAI, 2025. — URL: <https://openai.com/chatgpt> (дата звернення: 10.05.2025).
12. Міністерство цифрової трансформації України. Національна стратегія розвитку ШІ [Електронний ресурс]. — URL: <https://thedigital.gov.ua/aistrategy> (дата звернення: 10.05.2025).

13. Kuhn, T., Basch, P., Barr, M. Clinical documentation in the 21st century. // Journal of Internal Medicine. — 2016. — Vol. 28(3). — P. 102–109.
14. Іванов, І. В. Штучний інтелект у медицині. — Київ: Наукова думка, 2022. — 212 с.
15. SQLite Documentation. About SQLite [Електронний ресурс]. — URL: <https://www.sqlite.org/about.html> (дата звернення: 10.05.2025).

ДОДАТКИ

Додаток А. Скріншоти існуючих рішень

Додаток А 1. «Запитай у Гриця»

Сервіс інформує, де отримати ліки, а також які препарати доступні за програмою «Доступні ліки».

- [Telega](#) [Viber-бот](#)



Додаток А 2. «Буду мамою»

Чат-бот інформує про безоплатні медичні послуги для вагітних жінок та новонароджених у межах Програми медичних гарантій.

- [Telegra](#)



Додаток А 3. «Календар здорових українців»

Інформує про безоплатну діагностику залежно від віку та показань, використовуючи актуальні дані НСЗУ.

- [Telega](#)



Додаток А 4. «Лікуйся»

Чат-бот для людей, які мають діагноз рак молочної залози. Допомагає дізнатися про доступні безкоштовні медичні послуги.



Додаток А 5. #АРТпоруч

Telegram-бот, що допомагає знайти місця надання антиретровірусної терапії (АРТ) в Україні та за кордоном.

- [Telega](#)



Додаток 6. «Турбота»

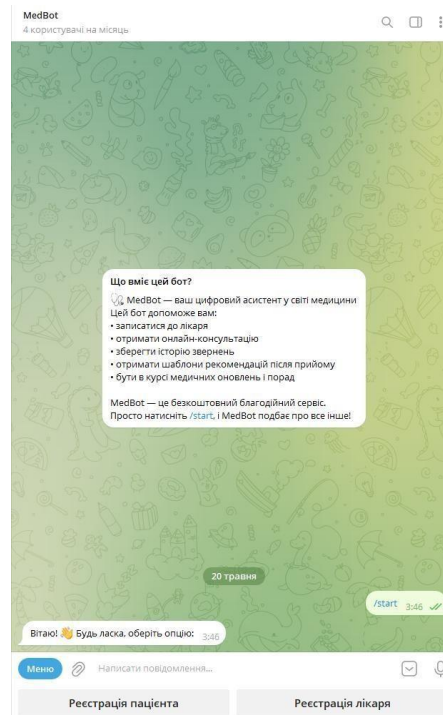
Онлайн-сервіс для безкоштовних медичних консультацій. Пацієнт обирає напрям і спілкується з лікарем через Telegram.

- [Telega](#)



Додаток Б. Скріншоти інтерфейсу розробленого бота

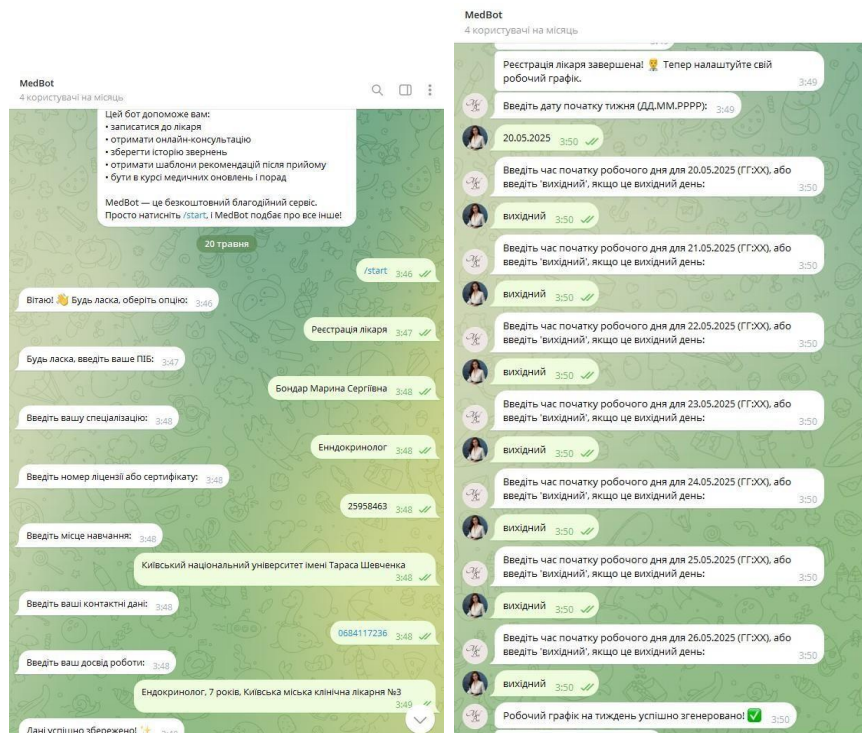
Додаток Б 1. Екран привітання та початку взаємодії з ботом



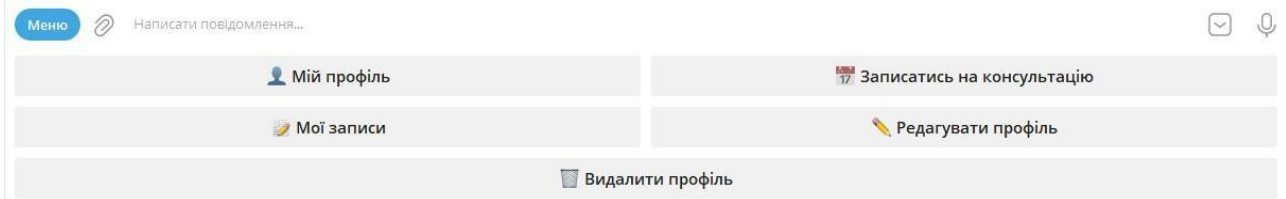
Додаток Б 2. Реєстрація користувача Пацієнт:



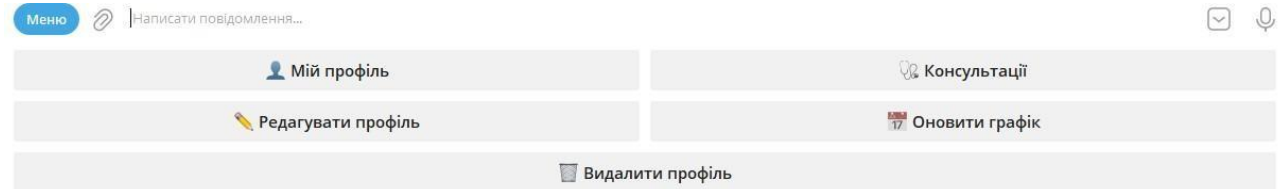
Лікар:



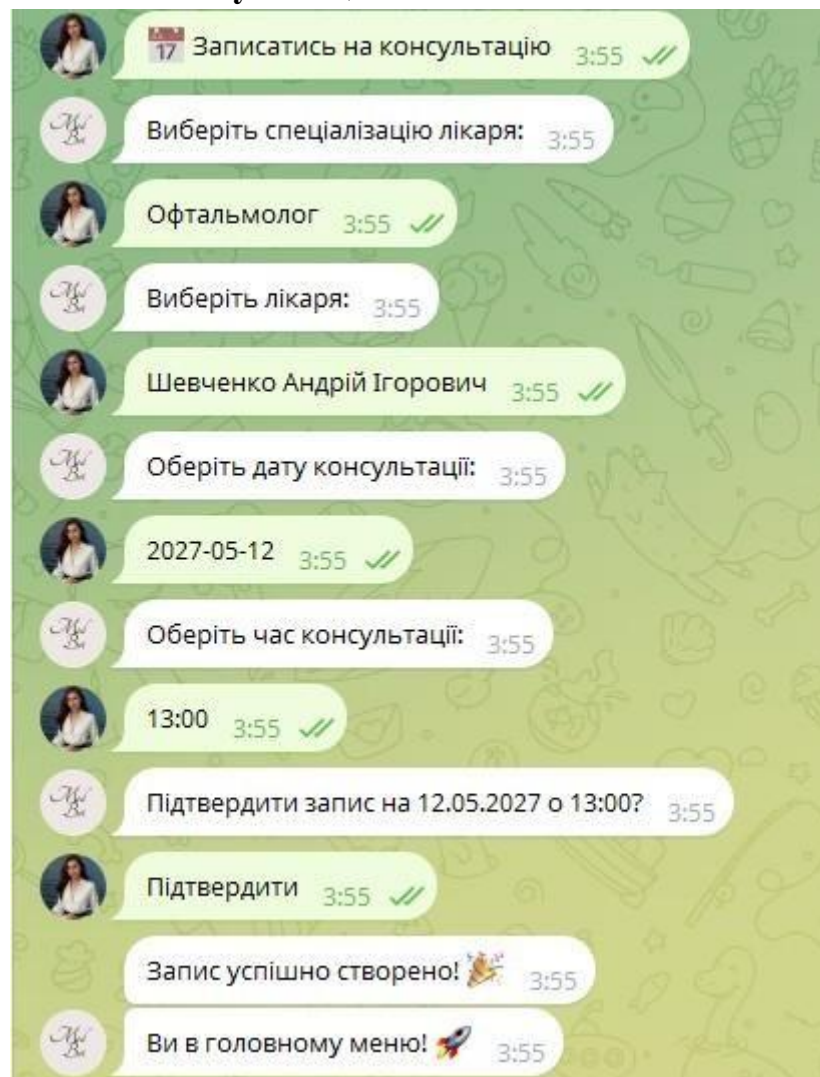
Додаток Б 3. Головне меню пацієнта



Додаток Б 4. Головне меню лікаря



Додаток Б 5. Запит на консультацію



Додаток Б 6. Список доступних лікарів (для пацієнта)

[Меню](#)[Написати повідомлення...](#)[Гінеколог](#)[Офтальмолог](#)[Пульмонологія, Терапія](#)[Назад](#)

Додаток В. Фрагменти коду та конфігураційні файли

Додаток В 1. Реєстрація користувачів

```
@bot.message_handler(func=lambda message: message.text in
    ["Реєстрація пацієнта", "Реєстрація лікаря"]) def registration_start(message):
    """Запускає процес реєстрації пацієнта або лікаря.""" role =
    "patient" if message.text == "Реєстрація пацієнта" else "doctor" global
    registration_data registration_data = {'role': role, 'telegram_id':
    message.from_user.id} bot.send_message(message.chat.id, "Будь ласка, введіть
    ваше ПІБ:") bot.register_next_step_handler(message, get_full_name)

def save_registration_data(message):
    """Зберігає дані реєстрації в базі даних.""" role = registration_data['role'] if
    role == "patient": table = "patients" data = {
        'telegram_id': registration_data['telegram_id'],
        'medical_card_number': registration_data['medical_card_number'],
        'full_name': registration_data['full_name'],
        'birth_date': registration_data['birth_date'],
        'contact_data': registration_data['phone_number'], # Зберігаємо номер
    телефону як контактні дані
        'chronic_diseases': registration_data['chronic_diseases'],
        'other_info': registration_data['other_info'],
        'username': registration_data['username'],
        'phone_number': registration_data['phone_number']
    }
    save_user_data(data, table) bot.send_message(message.chat.id,
    f"Реєстрація завершена! 📄 Ваш номер медичної картки:
    {registration_data['medical_card_number']}") show_menu(message, role) else:
    table = "doctors" data = {
        'telegram_id': registration_data['telegram_id'],
        'full_name': registration_data['full_name'],
        'specialization': registration_data['specialization'],
        'license': registration_data['license'],
        'education': registration_data['education'],
        'contact_data': registration_data['phone_number'], # Зберігаємо номер
    телефону як контактні дані
        'experience': registration_data['experience'],
        'username': registration_data['username'],
        'phone_number': registration_data['phone_number']
```

```

    }
    save_user_data(data, table)    bot.send_message(message.chat.id, "Реєстрація
лікаря завершена! 📄📅 Тепер налаштуйте свій робочий графік.")
setup_doctor_schedule(message)

```

Додаток В 2. Запис на прийом

```

@bot.message_handler(func=lambda message: message.text == "📅 Записатись на
консультацію") def appointment_request_start(message):
    """Запускає процес запису на консультацію."""    markup =
telebot.types.ReplyKeyboardMarkup(resize_keyboard=True)
    # Отримайте список спеціалізацій з бази даних    specializations =
get_all_specializations()    for specialization in specializations:
markup.add(telebot.types.KeyboardButton(specialization))
markup.add(telebot.types.KeyboardButton("Назад"))
bot.send_message(message.chat.id, "Виберіть спеціалізацію    лікаря:",
reply_markup=markup)    bot.register_next_step_handler(message,
appointment_select_specialization)

def appointment_save(message):
    """Зберігає запис на консультацію в базі даних."""    if message.text ==
"Підтвердити":
        appointment_datetime = datetime.combine(appointment_data['date'],
appointment_data['time'])    appointment_datetime_str =
appointment_datetime.strftime('%Y-%m-%d %H:%M:%S')

        conn =
sqlite3.connect(DATABASE_NAME)    cursor
= conn.cursor()    try:
            # Зберегти запис    cursor.execute("""
INSERT INTO appointments (patient_id, doctor_id, appointment_datetime)

```

```

VALUES (?, ?, ?)
"""', (appointment_data['patient_id'], appointment_data['doctor_id'],
appointment_datetime_str))

# Оновити статус у doctor_schedule cursor.execute("""
UPDATE doctor_schedule
SET is_available = 1
WHERE doctor_id = ? AND date_time = ?
"""', (appointment_data['doctor_id'], appointment_datetime_str))

conn.commit() bot.send_message(message.chat.id, "Запис
успішно створено! 🎉",
reply_markup=telebot.types.ReplyKeyboardRemove())

# Schedule reminders
schedule_reminders(appointment_data['patient_id'], appointment_data['doctor_id'],
appointment_datetime)

except Exception as e:
    print(f'Error saving appointment: {e}') conn.rollback()
    report_error_to_admin(f'Error saving appointment: {e}')
bot.send_message(message.chat.id, "Виникла помилка при збереженні запису.
Будь ласка, спробуйте пізніше. 😊",
reply_markup=telebot.types.ReplyKeyboardRemove()) finally:
conn.close() else:
    bot.send_message(message.chat.id, "Запис скасовано ",
reply_markup=telebot.types.ReplyKeyboardRemove())

show_menu(message, "patient")

```

Додаток В 3. Нагадування про прийом

```

def schedule_reminders(patient_id, doctor_id, appointment_datetime):
    """Планує нагадування для пацієнта та лікаря."""
    one_day_before = appointment_datetime - timedelta(days=1)
    five_minutes_before = appointment_datetime - timedelta(minutes=5)

    # Get patient and doctor telegram IDs
    patient = get_patient_by_id(patient_id)
    doctor = get_doctor_by_id(doctor_id)

    if patient and patient['telegram_id']:
        if one_day_before > datetime.now():
            bot.send_message(patient['telegram_id'],
                              f'Нагадування:                                Завтра
                              ({{appointment_datetime.strftime("%d.%m.%Y о %H:%M"}}) у вас консультація.")
        if five_minutes_before > datetime.now():
            bot.send_message(patient['telegram_id'],
                              f'Нагадування:                                Через                    5                                хвилин
                              ({{appointment_datetime.strftime("%d.%m.%Y о %H:%M"}}) у вас консультація.")
        else:
            report_error_to_admin(f'Не вдалося знайти Telegram ID пацієнта з ID
            {patient_id} для нагадування.")

    if doctor and doctor['telegram_id']:
        if one_day_before > datetime.now():
            bot.send_message(doctor['telegram_id'],
                              f'Нагадування:                                Завтра
                              ({{appointment_datetime.strftime("%d.%m.%Y о %H:%M"}}) у вас консультація з
                              пацієнтом.")

```

<pre>if five_minutes_before > datetime.now(): bot.send_message(doctor['telegram_id'],</pre>		
f"Нагадування:	Через	5
	ХВИЛИН	
<pre>({appointment_datetime.strftime('%d.%m.%Y о %H:%M')}) у вас консультація з пацієнтом.") else: report_error_to_admin(f"Не вдалося знайти Telegram ID лікаря з ID {doctor_id} для нагадування.")</pre>		

Додаток Г. Схеми і таблиці щодо візуалізації аналізу

Назва поля	Тип даних	Опис
id	INTEGER	Унікальний ідентифікатор запису
patient_id	INTEGER	Ідентифікатор пацієнта
doctor_id	INTEGER	Ідентифікатор лікаря
appointment_datetime	TEXT (DATETIME)	Дата та час прийому
status	TEXT	Статус запису (наприклад, confirmed)
complaints	TEXT	Скарги пацієнта
diagnosis	TEXT	Діагноз, встановлений лікарем
treatment	TEXT	Призначене лікування
recommendations	TEXT	Рекомендації лікаря

Таблиця Г.1 – Структура таблиці appointments

id	doctor_id	date_time	is_available
1	4	2025-05-20 14:00:00	true
2	4	2025-05-20 15:00:00	false

Таблиця Г.2 – Зразок даних таблиці doctor_schedule

i d	telegram _id	full_na me	specializat ion	license	educatio n	contact_dat a	experie nce	userna me
1	1234567 89	Іваненк о І.І.	Терапевт	АБ1234 56	НМУ ім. Богомол ьця	+380671112 233	10 років	drivanen ko

Таблиця Г.3 – Зразок даних таблиці doctors

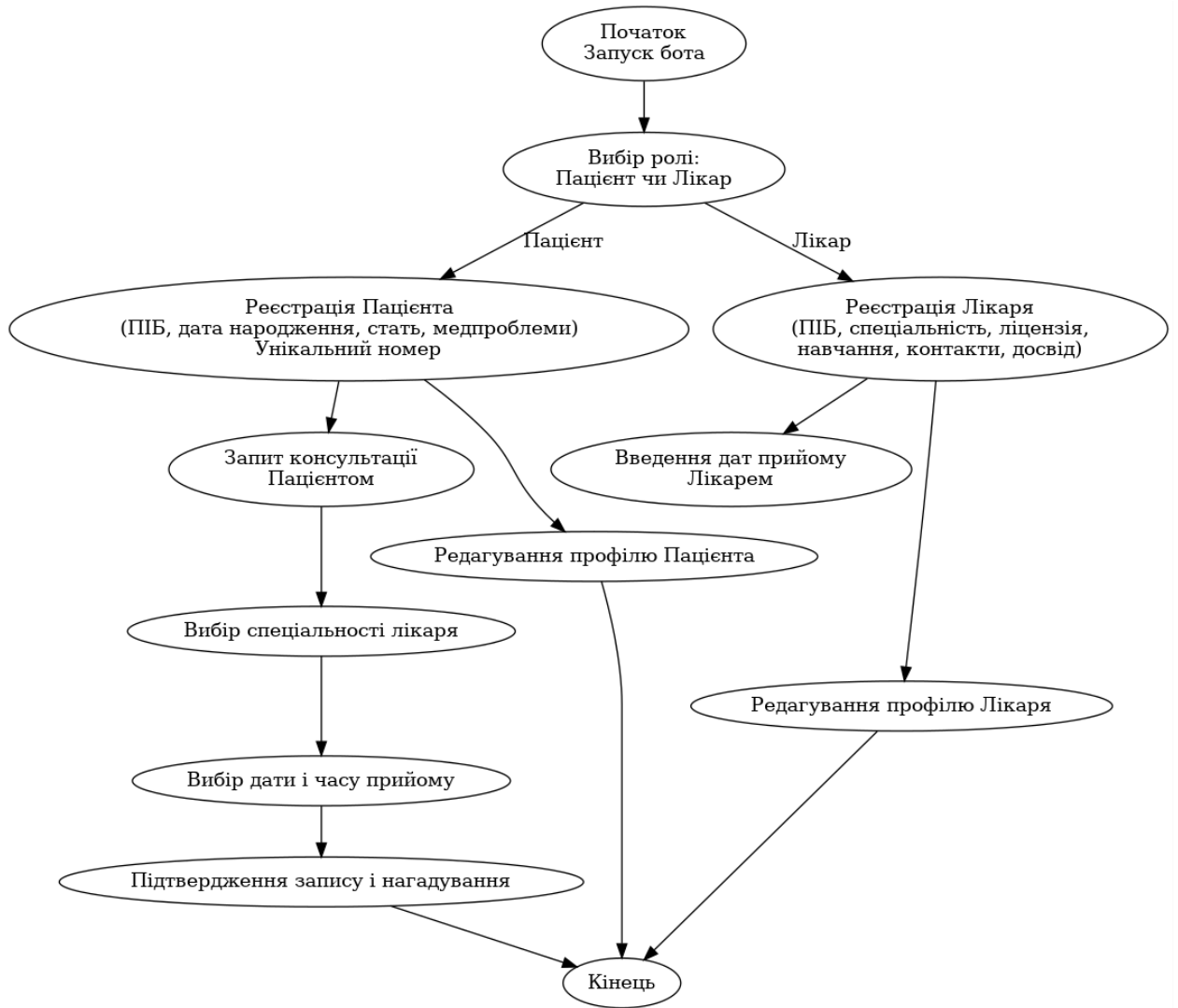
i d	telegra m_id	medical_card_ number	full_n ame	birth_ date	contact_d ata	chronic_di seases	other_ info	usern ame
1	9876543 21	МК-001234	Петре нко О.О.	12.04.2 000	+3805011 12233	Гіпертонія	алергія на пиллок	petro9 0

Таблиця Г.4 – Зразок даних таблиці patients

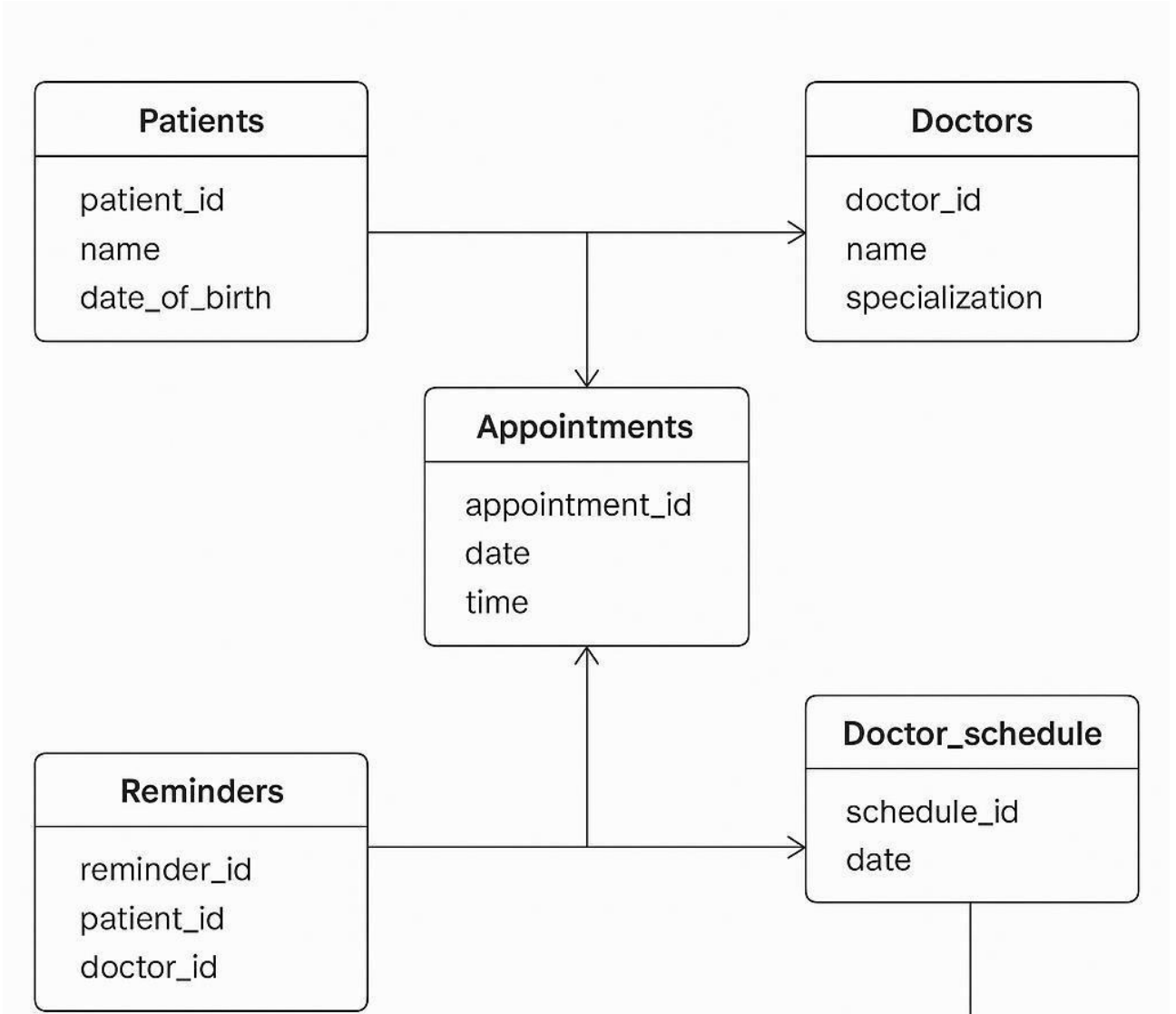
id	user_id	user_type	telegram_id	message	remind_at	is_sent
1	1	patient	987654321	Нагадування про прийом завтра	2025-05-19 09:00:00	1
2	1	patient	987654321	Нагадування за 1 годину	2025-05-20 13:00:00	0

Таблиця Г.5 – Зразок даних таблиці reminders

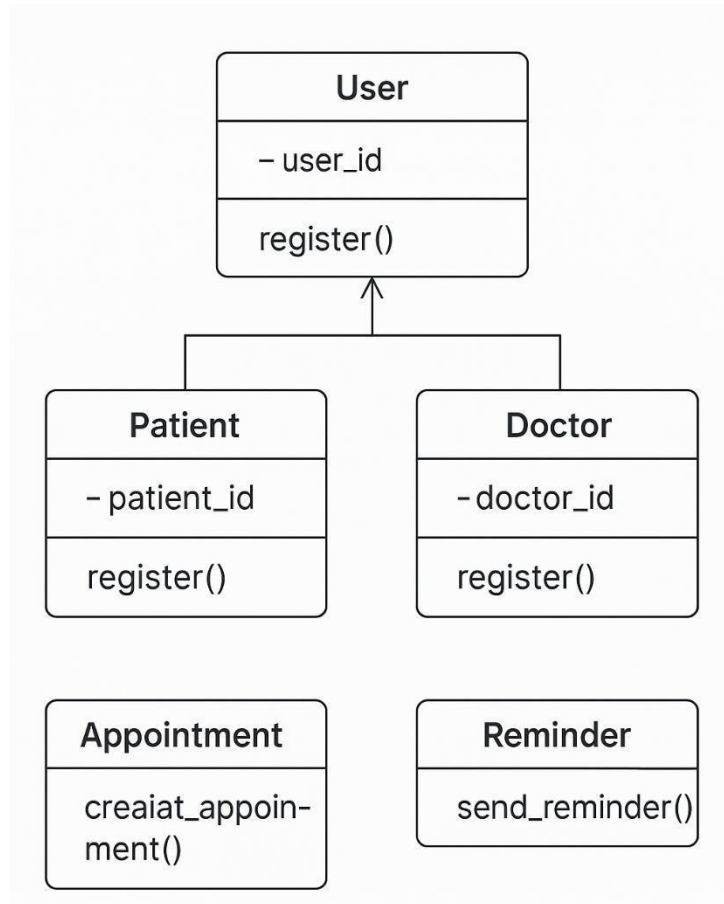
Додаток Д. Узагальнена схема алгоритму функціонування чат-боту



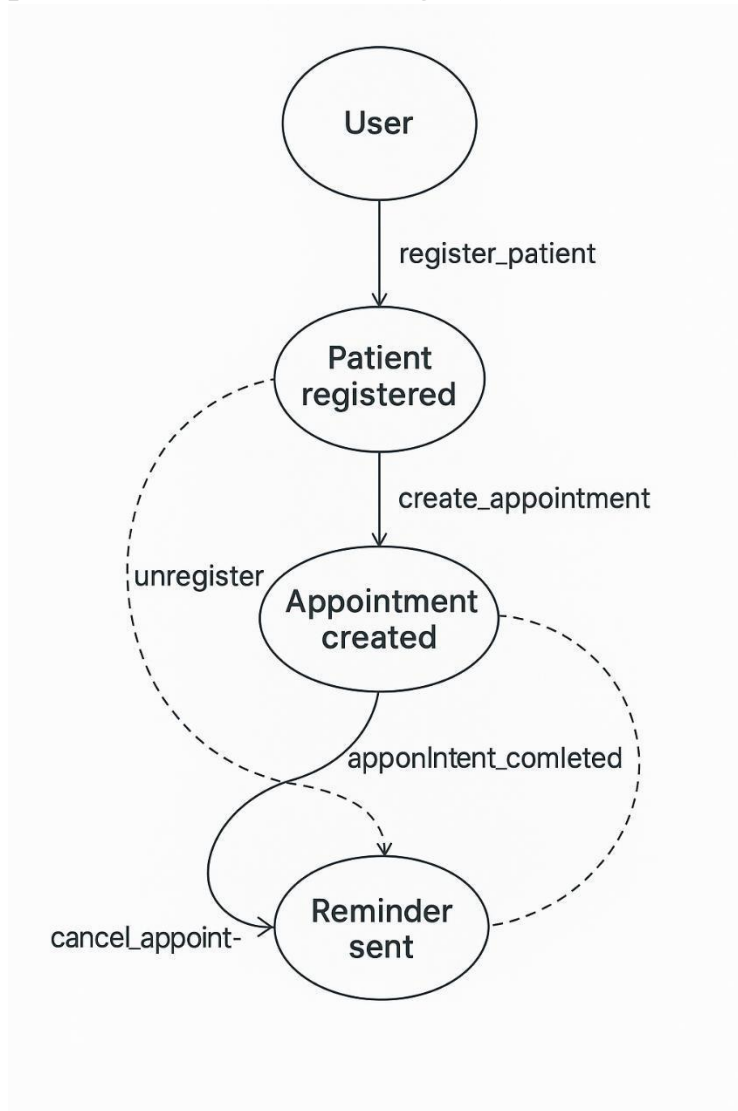
Додаток Е. Діаграми щодо проєктування продукту

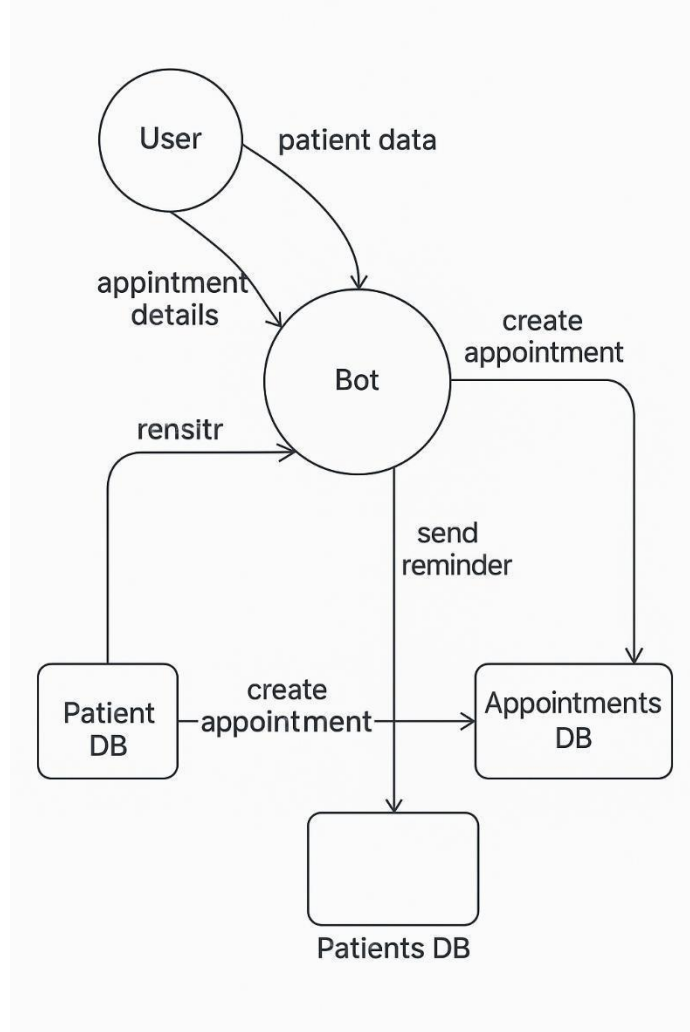


E.2. UML-діаграма класів



Е.3. Діаграма переходів станів (State Diagram)



Е.4. Діаграма потоків даних (DFD – Data Flow Diagram)

Е.5. Схема бази даних