

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТА
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра комп'ютерних наук

ПРОЄКТ ДРУГОГО РІВНЯ

НАЗВА:

«БІЗНЕС-ЧАТБОТ ДЛЯ БІЗНЕС ШКОЛИ КРОК»

Ступінь вищої освіти – бакалавр

Спеціальність – 122 «Комп'ютерні науки»

Освітня програма «Комп'ютерні науки»

ЗВІТ З ПРОЄКТУ

Команда проєкту:

Марина ЛАЗАРЕНКО

Катерина СИТНИК

Керівник проєкту:

Олег МУШИНСЬКИЙ

ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«УНІВЕРСИТЕТ ЕКОНОМІКИ ТА ПРАВА «КРОК»»

ЗАТВЕРДЖУЮ:
 завідувач кафедри комп'ютерних наук
 _____ **Сергій МІЧКІВСЬКИЙ**
 «28» квітня 2025 р.

ЗАВДАННЯ НА ПРОЄКТ ДРУГОГО РІВНЯ

Назва проєкту	Бізнес-чатбот для Бізнес Школи КРОК.
Мета проєкту	Автоматизувати відповіді на типові запитання потенційних клієнтів, впорядкувати інформацію та дати можливість клієнтам зв'язуватись із менеджером в реальному часі.
Опис проєкту	Надає у короткий час основну інформацію, яка цікавить клієнта; оптимізує роботу менеджерів зарахунок того що загальний час на консультацію зменшився; збільшує надходження заявок на програми та збільшує кількість відвідувачів на сайт; оптимізує знаходження інформації для клієнта.
Очікувані результати	Створити Чат-бот який в короткі терміни надасть основну інформацію про програми і зможе підключити до чату клієнта менеджера у реальному часі.
Ролі та відповідальність виконавців	Марина Лазаренко – менеджер проєкту, Frontend-розробник. Катерина Ситник – Fullstack-розробник.
Критерії успішності проєкту	Інтерфейс зрозуміли, адаптивний та зручний для клієнта. Інформація подана коротко та зрозуміло, клієнти швидко отримують необхідну інформацію. Бот працює швидко та стабільно, не виникає багів та критичних помилок. Проєкт протестований і його можна підключити до сайту Бізнес Школи.
<i>Технологічний стек, що використовується, та його обґрунтування</i>	
Технологія/Інструмент	JavaScript, HTML, CSS, JSON, Python, Flask, WebSocket
Опис	HTML – основна сторінка сайту CSS – для стилізації чат-боту JavaScript – прописаний основний функціонал роботи чат-боту

	<p>JSON - формат обміну даними між клієнтською частиною чат-бота і сервером</p> <p>Python - використовується для створення сервера чат-бота, обробки повідомлень, роботи із WebSocket, надсилання листів на пошту, збереження даних</p> <p>Flask - використовується як вебсервер для обробки REST-запитів</p> <p>WebSocket - забезпечує двосторонній зв'язок між клієнтом і сервером у реальному часі</p>
Причини вибору	Ми обрали ці стеки, оскільки вони направлені на розробку сайтів і їх функціоналу.

Дата видачі завдання 28 квітня 2025 р.

Керівник команди

Керівник проєкту

Марина ЛАЗАРЕНКО

Олег МУШИНСЬКИЙ

АНОТАЦІЯ

У межах проєкту було розроблено інтерактивного чат-бота для сайту Бізнес Школи КРОК з метою покращення взаємодії користувачів із інформаційними матеріалами про освітні програми. Бот поєднує клієнтську та серверну логіку, підтримує сценарії швидкої навігації за категоріями, а також функціонал звернення до менеджера в реальному часі через WebSocket-з'єднання. Передбачено збереження даних у формі звіту і відправка на електронну пошту. Дизайн чат-бота виконано у корпоративних кольорах, із урахуванням принципів адаптивного інтерфейсу та зручності для користувача. Продукт є гнучким, масштабованим та готовим до інтеграції на сайт <https://bs.krok.edu.ua>.

Ключові слова: чат-бот, UI/UX, WebSocket, HTML, інтерактивна взаємодія, освітній сайт

Посилання на Github проєкту: <https://github.com/smdxxxx/FAQ-bot.git>

ЗМІСТ

ВСТУП	6
РОЗДІЛ 1. КОНЦЕПТУАЛІЗАЦІЯ ПРОДУКТУ	8
1.1 Опис ідеї продукту	8
1.2 Аналіз предметної області та конкурентів	10
1.3 Опис продукту	13
РОЗДІЛ 2. ПРОЄКТУВАННЯ ПРОДУКТУ	15
2.1 Моделювання поведінки продукту	15
2.2 Моделювання структури продукту	16
2.3 Дизайн продукту	17
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОДУКТУ	19
3.1 Особливості реалізації	19
3.2 Тестування	21
3.3 Документація продукту	22
ВИСНОВКИ	24
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	25
ДОДАТОК А	26
ДОДАТОК Б	27

ВСТУП

Працюючи в компанії чи будь якому іншому підприємстві ти сам не вільно помічаєш елементи, які можеш покращити, оптимізувати, структурувати, з метою продуктивнішої роботи. Не існує ідеальної компанії, де вже впровадили все, що тільки можливо для комфортної і швидкої роботи й не залишилося нових фішок, котрі можна зробити. Завжди є можливість розвиватися і вдосконалюватися.

Бізнес Школу КРОК можна віднести до тих компаній, які переходять на цифровізацію, але все одно більшість процесів зосередженні й зберігаються в документах і папках на столі. Тому для практики, впровадження нових систем, поступової цифровізації робочого процесу, розробка та впровадження нових систем/програм – Бізнес Школа КРОК одне із найкращих площадок розвитку.

І почати цифровізацію Бізнес Школи КРОК вирішили з оптимізації надання інформації клієнтам. Зараз Бізнес Школа працює за даною схемою: Клієнт заходить на сайт -> Поверхнево вивчає наявну інформацію -> Заповнює форму зв'язку -> Менеджери зв'язуються з клієнтом і стисло розповідають інформацію на сайті -> Налагоджується контакт з клієнтом -> Пропонуються додаткові/цікавіші послуги. Основна проблема даної схеми – затребуваний час. На один телефонний дзвінок з клієнтом, менеджер може витратити від 10 до 40 хвилин, якщо не більше. І здебільшого розмова йде лише про загальну інформацію: що це за програма, що вивчають, який формат навчання, тривалість, навчальні матеріали, вартість тощо. Так, вся загальна інформація знаходиться на сайті, але сторінки дуже перенасичені і стратегічно головні компоненти для клієнтів губляться серед маси другорядної інформації, через що не досягається мети – дізнатися більше про програму.

З цієї проблеми і з'явилася ідея рішення цієї проблеми. Розробка чат-боту, який одразу зможе задовільнити головні питання клієнта, зменшить час розмови і пояснення всіх деталей і швидше налагодиться контакт з клієнтом.

Для втілення ідеї потрібно провести аналіз існуючих аналогів чат-ботів, зрозуміти плюси і мінуси, які інструменти краще обрати, скласти етапи

проєктування, розробки та впровадження проєкту: зібрати базу даних по освітнім програмам, створити макет чат-боту, розробити продукт, протестувати та підготувати продукт до впровадження.

РОЗДІЛ 1. КОНЦЕПТУАЛІЗАЦІЯ ПРОДУКТУ

1.1 Опис ідеї продукту

Ідея створення клієнтського чат-бота для Бізнес Школи КРОК з'явилася як відповідь на одну з ключових проблем сучасного споживання інформації – люди постійно сприймають велику кількість інформації і вони дуже швидко перенасичуються нею. Тому наповненість сайту всією існуючою інформацією призводить до перевантаженості сайту і, відповідно, повільного й складнішого сприйняття інформації для потенційних клієнтів.

У сучасному світі інформаційного перевантаження користувачі прагнуть отримати необхідну інформацію швидко, стисло і в зручному форматі. Водночас існуючий сайт Бізнес Школи містить велику кількість, беззаперечно, корисної та необхідної інформації: де є і детальний опис програм, і особливості навчання, і паралельні курси/навички, котрі можна отримати і багато-багато різного. Однак через перенасичення контентом та відсутності певної логіки в розміщені блоків, клієнти губляться в обсязі інформації, витрачають багато часу на пошук та аналіз, а зрештою вони втомлюються і втрачають інтерес до програм.

Особливо критично це для першого враження – коли потенційний клієнт не знаходить одразу потрібну інформацію, він шукає найпростіші та швидші методи їх отримання. І не знайшовши кращого варіанта, клієнт телефонує менеджеру і задає базові питання. Це чудово, коли здобувачі до нас самостійно телефонують, але тут існує 2 глобальні проблеми: 1) за розмовою менеджер може витратити на одне таке звернення багато часу, а це в першу чергу гроші; 2) не всі люди телефонують чи подають заявки, бо їм було не зручно чи вони не зрозуміли як це зробити.

Тому, щоб вирішити ці проблеми, була проявлена ініціатива по створенню чат-бота, який виконує функцію швидкого навігаційного помічника для клієнта. Основна ідея – надати клієнту можливість отримати необхідний контент у вигляді простого дерева рішень, де зібрано та описано

головну інформацію, стисло, коротко і без води. Замість перегляду десятків сторінок, відвідувач ввівши буквально декілька ключових слів проходить через логічні блоки та швидко отримує відповідь на бажанні питання. А в разі, якщо клієнт не знайшов відповідь на своє питання, він зможе у реальному часі надіслати запит менеджеру і отримати від відповідь.

Для підтвердження власних суджень було проведено опитування і зібрано низку найчастіше заданих питань клієнтами менеджером Бізнес Школи (опитування проходило в усній формі у вільному форматі). Серед цих питань були:

1. Розкажіть, будь ласка, про програму.
2. В якому форматі проходить навчання? Онлайн чи офлайн?
3. В які дні навчання? Скільки триває по часу день навчання?
4. Яка вартість програми? Чи можлива щомісячна оплата?
5. Які документи потрібно подати і до коли?
6. Мені обов'язково складати ЄВІ/ЄФВВ?
7. Які курси проходять?
8. А як будуть проходити заняття? Будуть практичні кейси чи більше лекційної інформації?

На частину цих питань можна отримати інформацію із сайту, але її потрібно знайти. Деякі знаходяться дуже легко, деякі навпаки – складно. Також серед менеджерів запитали, на які питання найбільше витрачається часу? І звичайно – це найбільш обширні, як питання 1, 5, 6, 7 та 8. Ці питання мають як великий обсяг інформації так і багато нюансів, які впливають лише під час розмови з клієнтом.

Завдяки цьому опитуванню та аналізу вдалося сформувати головну структуру чат-бота, яка містить основний контент про програми, структурували та дали відповіді на найчастіше задані питання, а у разі нестандартної ситуації, клієнт міг зв'язатися із менеджером і отримати відповідь на своє питання.

Підводячи підсумок вдалося сформулювати чітку ідею чат-боту, його концепцію: простого, легкого, клієнтоорієнтованого інструменту, який автоматизує базову комунікацію з клієнтом і в разі його зацікавленості можна поспілкуватися напряду з менеджером.

1.2 Аналіз предметної області та конкурентів

Перед проектуванням чат-бота потрібно проаналізувати конкурентів, освітні заклади, які також мають освітній-допоміжний чат-бот. Зрозуміти, яка їх основна концепція, особливості реалізації, зручність та актуальність. Для зручності аналіз проведемо в таблиці 1.1.

Таблиця 1.1 - Аналіз освітніх чат-ботів конкурентів

Назва вищого закладу	Вид	Основні функції	Надана інформація	Переваги	Недоліки
Полтавська політехніка	Telegram-бот	Меню-команди, кнопки	Зосереджена на інформації про вступ	Легкий доступ у Telegram	Вузкий функціонал, не завжди працює
Херсонський державний університет	AI-бот	AI-асистент здобувача по навчанню	Персональна допомога в навчанні	Індивідуальний підхід	Лише для студентів університету
Дніпровський нац. ун-т імені О. Гончара	Telegram-бот	Меню-команди, кнопки	Зосереджена на інформації про вступ	Велика кількість вже існуючих питань	не завжди працює
Луцький національний технічний університет	Telegram-бот	Меню-команди, кнопки	Зосереджена на інформації про вступ	Зосереджений фокус на вступ	Інформація лише за вступ

Отже, обрані конкуренти найчастіше маю Telegram-бот, але, на жаль, в Університеті КРОК заборонено використовувати застосунок Telegram у роботі, тому його використання потрібно максимально зменшити. У більшості випадків чат-боти сфокусовані саме на умовах вступу, а не на освітніх

послугах, які вони можуть пропонувати. Як виявилось чат-ботів, які саме зосереджені на продажу освітніх послуг досить мало, їх фактично немає, тому схема побудови освітніх ботів буде сильно відрізнятися від чат-бота для Бізнес Школи. Також деякі боти надавали контакти приймальних комісій для зв'язку, але з менеджерами не можна було поспілкуватися в самому боті.

У підсумку можна сказати, що чат-бот повинен відповідати політиці Бізнес Школи КРОК, зручній взаємодії і мав можливість зв'язуватися із менеджером для детальнішого спілкування. Наприклад створити браузерний чат бот і безпосередньо підключити його до сайту Бізнес Школи.

Після аналізу конкурентів, варто глибше виконати аналіз нашого продукту. Для цього найкраще підходить метод аналізу *SWOT* (Strengths, Weaknesses, Opportunities, Threats) (табл. 1.2). Завдяки цьому аналізу, можна швидко зрозуміти всі сильні та слабкі сторони продукту.

Таблиця 1.2 - *SWOT*-аналіз

<i>Сильні сторони</i>	<i>Слабкі сторони</i>
<ul style="list-style-type: none"> - повністю стилізований інтерфейс під сайт Бізнес Школи; - основна робота боту не залежить від серверної частини; - зручне редагування вмісту й швидке оновлення інформації; - комунікація з клієнтом в реальному часі за допомогою серверного підключення. 	<ul style="list-style-type: none"> - відсутність глибокої аналітики поведінки користувачів. - затримка у відправці актуальних звітів при збою або відключені серверу. - менеджер може довго підключатися до чату з клієнтом (потрібно постійно моніторити).
<i>Можливості</i>	<i>Загрози</i>
<ul style="list-style-type: none"> - легка інтеграція з CRM або Google Sheets у майбутньому; - можливість масштабування на інші освітні сайти; - додавання мультимовності, сценаріїв або персоналізації. 	<ul style="list-style-type: none"> - зміни в структурі сайту можуть потребувати оновлення логіки/посилань; - збій у серверній і стирання даних всіх чатів.

Проведений SWOT-аналіз дозволив виявити ключові переваги та обмеження запропонованого чат-бота, а також окреслити потенційні можливості розвитку та ризики при впровадженні бота. Загалом, аналіз підтвердив, що обрана модель реалізації чат-бота є оптимальною для потреб Бізнес Школи КРОК, з можливістю поступового розширення та удосконалення.

Після повного аналізу проблем клієнта, вивчення конкурентів, дослідження SWOT-аналізу та формуванні ідеї продукту, потрібно структурувати основні бізнес-аспекти в одну цілу картину.

Зараз найчастіше використовується методика *Lean Canvas* - інструмент для візуалізації ключових елементів продукту на ранніх етапах проектування. Ця модель дозволяє комплексно розглянути проблеми клієнтів, запропоноване рішення, канали просування, витрати, вигоди та інші важливі фактори.

У фокусі *Lean Canvas* (рис. 1.1) – розробка інтерактивного чат-бота для сайту Бізнес Школи КРОК, що не лише забезпечує швидкий доступ до ключової інформації для користувача, а й включає повноцінну серверну логіку для живого спілкування з менеджером, збереження історії діалогу та автоматичного надсилання звітів. Завдяки структурованому підходу було чітко окреслено основні проблеми взаємодії відвідувачів з сайтом, визначено цільову аудиторію, сформовано унікальну ціннісну пропозицію – поєднання запрограмованого та живого спілкування, – а також виокремлено ключові ресурси, витрати, канали та метрики ефективності.

В такому вигляді вже наглядно видно не лише що повинен робити бот, а й сформувалися можливості, в яких його можна розвивати далі, підтримувати та вдосконалювати.

БІЗНЕС-ЧАТБОТ ДЛЯ БІЗНЕС ШКОЛИ КРОК Lean Canvas



Рисунок. 1.1. - Чат-бот Lean Canvas

Джерело: розроблено авторами

1.3 Опис продукту

Метою створення чат-бота є впровадження зручного, інтерактивного інструменту комунікації між потенційними клієнтами та менеджерами Бізнес Школи КРОК, який дозволяє суттєво скоротити час на пошук базової інформації про освітні програми та зменшити навантаження на менеджерів. Продукт вирішує поширену проблему – перевантаженість сайту деталями та складність швидкої орієнтації в інформації, через що користувачі витрачають багато часу на пошук або взагалі залишають сайт, не знайшовши потрібного. Чат-бот дозволяє за введенням ключових слів отримати всю основну інформацію про програми, формат, вартість навчання, умови вступу, та, за потреби, зв'язатися з менеджером у реальному часі.

Функціонально бот складається з клієнтської та серверної частин. Клієнтська частина реалізована за допомогою HTML, CSS і JavaScript — саме вона відповідає за відображення вікна чату, логіку взаємодії з кнопками та текстом введення та виведення відповідей. Структура сценаріїв збережена у

вигляді JSON-подібних даних у спеціальному JavaScript-файлі, що забезпечує швидке редагування та масштабування інформації без втручання у основний код.

Серверна частина побудована з використанням Python, Flask і WebSocket. Вона відповідає за обробку звернень користувачів, які надсилають запит через кнопку “Зв’язатися з менеджером”. У цей момент між клієнтом і менеджером створюється WebSocket-з’єднання, яке дає змогу вести синхронну текстову комунікацію прямо у чаті, не перезавантажуючи сторінку. Крім того, сервер формує текстовий звіт усієї сесії користувача –збереження історії діалогу, який надсилається менеджеру на електронну пошту. Якщо на момент надсилання сервер тимчасово не працює, звіт зберігається локально і буде надісланий автоматично, щойно сервер відновить з’єднання. Це гарантує безперервність комунікації навіть у випадках збоїв.

Таким чином, чат-бот поєднує швидкодію клієнтського інтерфейсу з функціональністю серверної взаємодії, забезпечуючи як автономне користування, так і живий діалог з менеджером у реальному часі. Його можна легко інтегрувати в будь-яку HTML-сторінку сайту без необхідності підключення сторонніх бібліотек або платформ. Підтримується робота в усіх сучасних браузерах та на мобільних пристроях. Така гібридна архітектура робить продукт зручним, гнучким та надійним інструментом цифрової комунікації для освітнього закладу.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ПРОДУКТУ

2.1 Моделювання поведінки продукту

Після аналізу та детального опису чат-бота, його функціонал та особливості, та причини. Наступним етапом стає проєктування бота і створення основної заготовки, умовно кажучи, чернетки за якою схемою має працювати бот, яку модель структури баз даних міститиме і як він має виглядати в результаті. І першим етапом в проєктуванні це як між собою основні компоненти мають взаємодіяти.

Для кращого візуального розуміння логіки взаємодії між клієнтом, чат-ботом та менеджером, найзручніше побудувати Use Case діаграму, що наглядно відображає взаємозв'язок всіх компонентів та їх роботу між собою (рис. 2.1). Діаграма ілюструє типову поведінку клієнта, послідовну роботу чат-бота та функції, які виконує система в процесі обробки запитів.

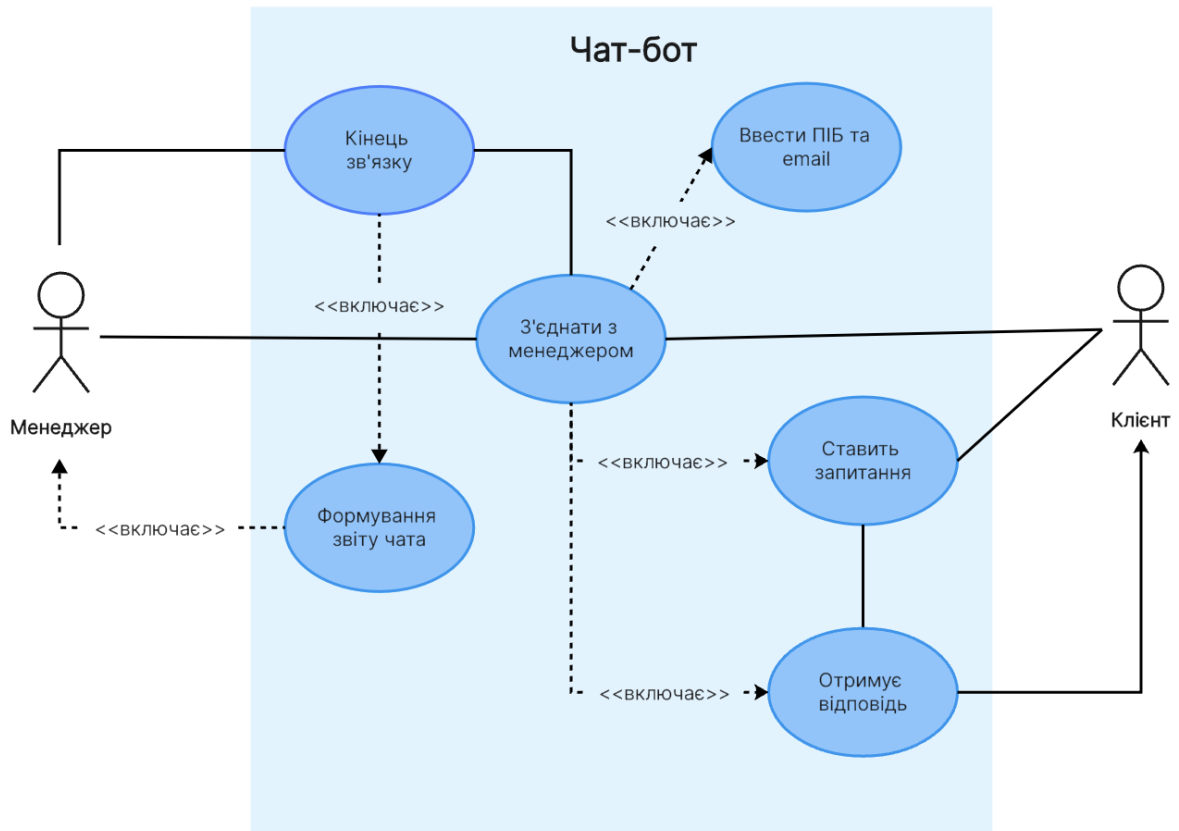


Рисунок 2.1 - Use Case діаграма взаємодії клієнта з чат-ботом

Джерело: розроблено авторами

На діаграмі видно, що актором являються Менеджер та Клієнт. Основним актором чат-боту являється **Клієнт**. Він ініціює взаємодію з ботом і може обрати два наступні сценарії: **поставити запитання** або **зв'язатися з менеджером**.

За першим сценарієм Клієнт ставить питання боту і він **Надсилає відповідь** із зарання підготовлених блоків. Цей процес може йти постійно.

За другим сценарієм Клієнт зв'язується з менеджером, що автоматично включає в себе заповнити форму і **Ввести ПІБ та email**. Після відправки запиту із Клієнтом зв'язується Менеджер і їх зв'язок автоматично включає процес ставлення питання та надання відповіді. Як на всі питання Клієнта Менеджер відповів, він **Завершує зв'язок**, що автоматично включає в себе **Формування звіту чата** і відправка звіту Менеджеру.

Завдяки поданій моделі видно, що бот поєднує автономну навігацію по готовим відповідям із можливістю гнучкої взаємодії в режимі реального часу. Система підтримує як автоматизовану, так і живу комунікацію, дозволяючи максимально охопити різні сценарії поведінки користувачів і забезпечити швидку відповідь на індивідуальні запити.

2.2 Моделювання структури продукту

Такого плану чат-бот не може міститися без бази даних. Від клієнтів буде поступати 2 типи даних: ті хто взаємодіяв із чат-ботом, але не залишив контактні дані, і ті хто і взаємодіяв із чат-ботом і залишив контактні дані. У такому випадку потрібно спроектувати двокомпонентну структуру збереження даних. Основна ідея полягає у розділенні бази даних на дві логічні категорії: персоналізовану (користувацьку або клієнтську) базу та загальну (анонімну) базу чату.

Користувацька база даних повинна створюватися для кожного клієнта, який під час взаємодії з ботом ініціює зв'язок з менеджером і вводить свої контактні дані – зокрема, прізвище, ім'я, по батькові та електронну пошту. Для таких сесій створюється унікальний запис, що включає повну історію

переписки в чаті, дані користувача, а також маркування про те, чи була розмова завершена, чи ні. Ці дані в подальшому будуть відправлені менеджеру у формі звіту одразу після завершення діалогу.

Загальна база чату повинна зберігати дані тих клієнтів, які не ідентифікували себе – тобто не залишили контактних даних. Незважаючи на це, система має фіксувати чат: запити клієнта, відповіді бота, переходи між розділами. Ця інформація повинна тимчасово зберігатися локально до моменту завершення розмови або сплину періоду неактивності. Після завершення також формується звіт і надсилається менеджеру.

Якщо на момент закриття часу зв'язок із сервером відсутній, система зберігатиме звіт локально і автоматично відправити його після відновлення.

Таким чином, структура продукту повинна гарантувати збереження, цілісність та своєчасну обробку усіх сесій взаємодії з клієнтом – незалежно від того, залишає він контактні дані чи ні. Це дозволить покращити якість обслуговування, зберегти історії комунікацій і забезпечити прозорість внутрішніх процесів взаємодії між клієнтом і менеджером.

2.3 Дизайн продукту

І на останок найсмачніше: дизайн чат-боту. UI/UX-дизайн орієнтований на те, щоб зробити взаємодію інтуїтивно зрозумілою, візуально привабливою і логічно структурованою. Зовнішній вигляд бота виконано у корпоративному стилі Бізнес Школи КРОК: сірого (rgb 217, 217, 214), синього (rgb 0, 85, 135) і та зеленого (rgb 206, 220, 0). Ці кольори застосовано як для фону, так і для елементів навігації, кнопок і виділення ключової інформації. Так для не перенасичення використовуватиметься білий колір.

Інтерфейс чат-бота з боку користувача має вигляд невеликого віджета, закріпленого у правому нижньому куті сторінки. При натисканні відкривається компактне діалогове вікно з розділеннями на зони: область повідомлень, поле для введення питання, кнопка Надіслати та кнопка

Зв'язатися з менеджером. Всі елементи розташовані таким чином, щоб не перевантажувати екран і не блокувати основний зміст сайту (рис 2.2.).



Рисунок 2.2 - Чат-ботом початкове вікно

Джерело: <https://www.canva.com/>

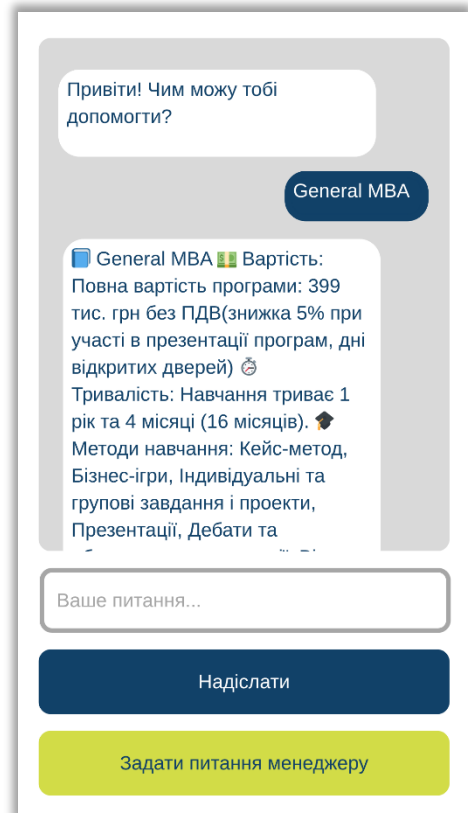


Рисунок 2.3 - Чат-ботом при введенні ключового слова

Джерело: <https://www.canva.com/>

При веденні запитання або ключового слова приходила відповідь від бота (рис. 2.3). Якщо неправильно або не зрозуміло введено слово приходиться помилка. Вигляд форми зв'язку із менеджером та екран чату зі сторони менеджера представлені в додатку А (ст. 27).

Усі елементи дизайну підпорядковані принципам UX-простоти: клієнт не вводить текст вручну, поки не ініціює звернення; усі повідомлення чітко відокремлені; кнопки не дублюються й мають логічну назву. Такий підхід дозволяє не лише знизити поріг входу для нових користувачів, а й забезпечує зручність використання з мобільних пристроїв.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ПРОДУКТУ

3.1 Особливості реалізації

Реалізація чат-бота базується на взаємодії між клієнтською частиною, яка створена за допомогою HTML, CSS і JavaScript, та серверною частиною, написаною на Python із використанням фреймворку Flask та WebSocket-з'єднання для обміну повідомленнями в реальному часі.

Користувач взаємодіє з інтерфейсом чату, введене повідомлення обробляється JavaScript-кодом і надсилається на сервер через WebSocket-з'єднання. Сервер Python приймає це повідомлення, обробляє його та надсилає у відповідь.

Клієнтська частина:

- HTML створює структуру чату (форма введення, поле діалогу, список користувачів, сповіщення);
- CSS відповідає за оформлення та адаптивний вигляд інтерфейсу;
- JavaScript керує логікою відображення повідомлень, з'єднанням WebSocket, прийманням та надсиланням даних, генерацією повідомлень на екрані;
- для зручності менеджера реалізовано сповіщення про нові запити (toast-повідомлення) та вибір активного чату.

Серверна частина реалізована на Python з використанням:

- Flask –для обробки REST-запитів, зокрема, надсилання заявки на email після завершення чату;
- websockets –для підтримки постійного з'єднання між клієнтом і сервером у реальному часі;
- dotenv –для зберігання конфіденційних змінних середовища (пароль, порти тощо);
- email.mime –для форматування і надсилання листів менеджеру з історією діалогу.

Сервер підтримує:

- ідентифікацію ролі користувача (user / manager),
- логіку підключення, прийняття чату,
- обробку текстових повідомлень,
- передачу повідомлень між сторонами,
- завершення діалогу з автоматичним надсиланням листа з історією

переписки на email.

Архітектура:

- один WebSocket сервер (ws server.py) обробляє весь діалог у реальному часі;
- окремий HTTP-сервер (server.py) використовується для надсилання історії чату на email;
- замість бази даних інформація тимчасово зберігається в оперативній пам'яті сервера у вигляді словників (clients, history);
- у випадку відключення –дані видаляються, що є оптимальним рішенням для простих MVP-реалізацій або тестових версій.

Нижче наведено приклад коду обробки повідомлення на клієнті:

```
ws.onmessage = (event) => {
const data = JSON.parse(event.data);
if (data.type === "message") {
const userId = connectedUserId;
chatHistories[userId].messages.push({ role: "user", text: data.text });
displayChat(userId);
}
};
```

А на сервері:

```
elif msg["type"] == "message":
partner_email = sender.get("partnerId")
if partner_email and partner_email in clients:
```

```
await clients[partner_email]["socket"].send(json.dumps({
    "type": "message",
    "text": msg["text"]
}))
```

Таким чином, реалізація поєднує простоту та гнучкість, забезпечуючи мінімальну затримку обміну повідомленнями між користувачем і менеджером.

3.2 Тестування

Після завершення основного етапу розробки чат-бота потрібно провести комплексне тестування з метою перевірки коректності його роботи, стабільності та відповідності заявленому функціоналу. Тестування охоплює як технічну частину (окремі логічні блоки), так і поведінку системи під час повноцінної взаємодії з клієнтами.

Модульне тестування дозволило перевірити ключові елементи чат-бота у відриві від загальної системи. Зокрема, були протестовані механізми взаємодії з кнопками, генерація відповідей на запити, відображення повідомлень у чаті, запуск і обробка форми зворотного зв'язку, передача даних на сервер і створення звітів. Особливу увагу приділяли перевірці ситуацій, коли сервер тимчасово недоступний. У таких випадках чат-бот зберігав історію локально, а після відновлення з'єднання – коректно надсилав звіти без втрати інформації. Усі перевірені компоненти працювали стабільно й без помітних збоїв.

Функціональне тестування охоплювало повний життєвий цикл взаємодії між користувачем і системою: від відкриття чату на сайті й ознайомлення з тематичними категоріями до відправлення повідомлення менеджеру та завершення сесії. Перевірка охоплювала поведінку бота на різних типах пристроїв, коректність відображення інтерфейсу, здатність системи реагувати на помилки введення, а також підтримку живого чату через WebSocket-

підключення. Навіть при навмисному порушенні сценарію, як-от введення порожніх даних або відключення від мережі, бот реагував передбачувано та стійко, забезпечуючи збереження інформації і коректне завершення взаємодії.

У результаті тестування було підтверджено, що реалізована система повністю відповідає технічному завданню, задовольняє вимоги щодо зручності та інтуїтивності інтерфейсу й демонструє надійну роботу в стандартних і виняткових сценаріях.

3.3 Документація продукту

Щоб розпочати взаємодію з чат-ботом, клієнту необхідно натиснути на іконку бота, що розташована у правому нижньому куті сторінки. Після цього відкриється діалогове вікно, де автоматично з'явиться привітальне повідомлення, поле введення та 2 кнопки: Надіслати та Зв'язатись з менеджером.

У поле введення клієнт може вести питання або ключове слово щоб отримати відповідь на питання.

У разі, якщо жодна з відповідей не підходить, передбачена можливість звернутися до менеджера. Для цього необхідно скористатися кнопкою "Зв'язатися з менеджером", після чого відкривається форма для введення ПІБ та електронної пошти.

Після заповнення форми користувач може надіслати запит до менеджера. Якщо менеджер перебуває онлайн, він приєднається до сесії в реальному часі й розпочне живе спілкування через той самий інтерфейс чату. Клієнт може ставити додаткові запитання, а менеджер відповідатиме безпосередньо в діалоговому вікні. По завершенню розмови менеджер завершити сесію.

Після завершення діалогу система автоматично формує звіт про всю історію спілкування, який потім одразу відправляється менеджеру. Якщо під час надсилання виникають технічні труднощі або сервер тимчасово

недоступний, система зберігає звернення локально і відправляє його автоматично після відновлення.

Чат-бот працює цілодобово, дозволяючи клієнтам отримувати відповіді навіть у неробочий час. Менеджери приєднуються до діалогів у визначений період – з 9:00 до 16:00. Такий підхід дозволяє забезпечити безперервність комунікації, зручність користування та ефективну обробку запитів з боку адміністрації освітнього закладу.

ВИСНОВКИ

У межах даного проєкту було реалізовано функціонального чат-бота для сайту Бізнес Школи КРОК, який виконує роль цифрового консультанта для потенційних клієнтів. Його головне призначення –автоматизувати відповіді на типові запитання, зменшити навантаження на менеджерів та скоротити час, необхідний для первинної комунікації з відвідувачами. Основною метою було усунути залежність від сторонніх сервісів на кшталт Chatra Brevo і створити незалежне рішення, повністю адаптоване до потреб школи.


Команда проєкту обрала технологічний стек, до якого увійшли JavaScript, HTML, CSS, JSON, Python, Flask і WebSocket. Це дозволило реалізувати бот, що поєднує швидкодію фронтенду з динамічною логікою на сервері. Чат-бот побудований за принципом дерева рішень із багаторівневою навігацією, що дозволяє користувачу швидко орієнтуватися у великому обсязі інформації без необхідності звертатися до сторонніх каналів.

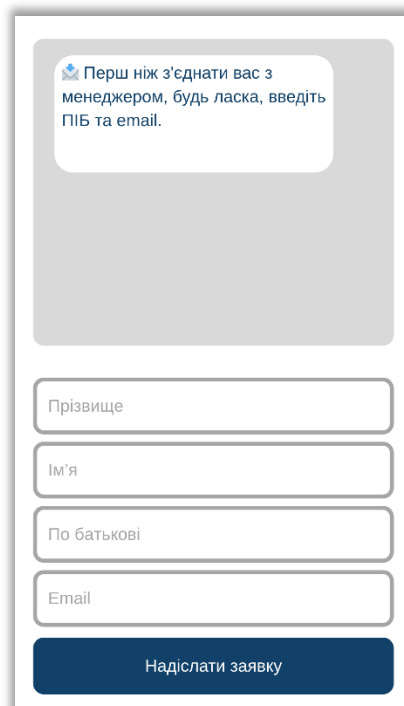
Серверна частина дозволяє забезпечити синхронну комунікацію з менеджером у реальному часі, автоматичне формування та надсилання звітів, а також збереження історії діалогів. У випадках перебоїв із сервером чат-бот зберігає дані локально й надсилає їх автоматично після відновлення з'єднання, що підвищує надійність та безперервність взаємодії.

Важливо зазначити, що реалізований інтерфейс дотримується корпоративного стилю школи, є інтуїтивно зрозумілим, адаптивним та зручним як на ПК, так і на мобільних пристроях. Проєкт пройшов етапи планування, аналізу, проєктування, розробки, тестування та документування, а також показав високу стабільність у роботі під час перевірки різних сценаріїв поведінки користувачів.

У результаті реалізації цього проєкту Бізнес Школа КРОК отримала сучасний, ефективний інструмент цифрової комунікації, який не тільки покращує перший контакт із клієнтами, а й створює основу для подальшої цифровізації освітніх послуг..

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Що таке use CASE [Електронний ресурс]. URL: https://dist.krok.edu.ua/pluginfile.php/198860/mod_resource/content/1/%D0%A9%D0%BE%20%D1%82%D0%B0%D0%BA%D0%B5%20use%20CASE.pdf
2. Boardmix [Електронний ресурс]. URL: <https://boardmix.com/>
3. Canva [Електронний ресурс]. URL: <https://www.canva.com/>
4. Чит-кіт для вступників by LNTU  [Електронний ресурс]. URL: https://t.me/VseProVstup_bot
5. DNU Info Bot [Електронний ресурс]. URL: https://t.me/info_dnu_bot
6. WebSockets: навіщо потрібні та як з ними працювати [Електронний ресурс]. URL: <https://proit.ua/websockets-navishcho-potribni-ta-iaak-z-nimi-pratsiuvati-2/>
7. Бізнес Школа КРОК [Електронний ресурс]. URL: <https://bs.krok.edu.ua/>
8. Що таке JSON і як з ним працювати [Електронний ресурс]. URL: <https://foxminded.ua/json/>
9. JSON серіалізація в Python за допомогою серпу [Електронний ресурс]. URL: <https://devzone.org.ua/post/json-serializatsiia-v-python-za-dopomohoiu-serpy>



Перш ніж з'єднати вас з менеджером, будь ласка, введіть ПІБ та email.

Прізвище

Ім'я

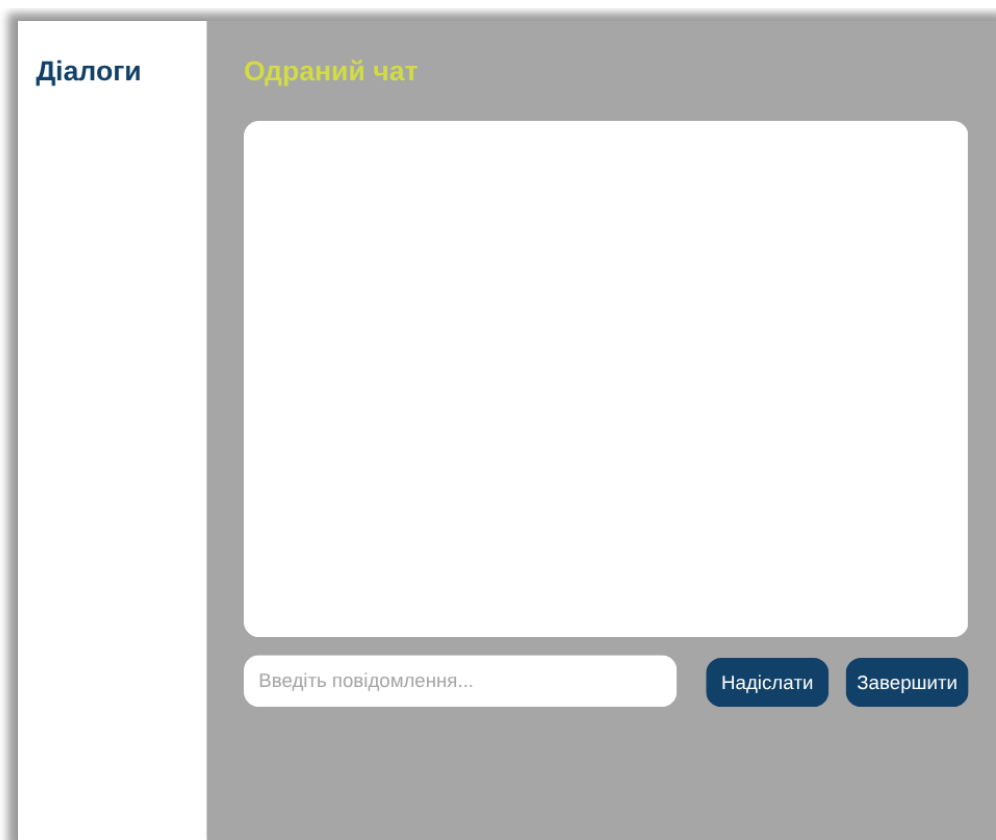
По батькові

Email

Надіслати заявку

Рисунок А.1. - Чат-ботом форма зв'язку

Джерело: <https://www.canva.com/>



Діалоги

Одраний чат

Введіть повідомлення...

Надіслати

Завершити

Рисунок А.2. - Чат-ботом з екрану менеджера

Джерело: <https://www.canva.com/>

ДОДАТОК Б

Програмний код.

Ключові слова та Питання

```
const keywordsMap = {
  програми: ["програма", "програми", "програму", "програм",
"програмами"],
  академічні: ["академічні"],
  модульні: ["модульні"],
  адміністрування: ["адміністрування"],
  тривалість: ["тривалість", "довжина", "скільки триває"],
  вартість: ["ціна", "вартість", "коштує", "оплата"],
  заявка: ["заявка", "подати", "реєстрація", "реєструватись", "заявку"],
  початок: ["початок", "старт", "стартує"],
  місцезнаходження: ["місцезнаходження", "де ви", "адреса",
"знаходиться"],
  контакти: ["контакт", "контакти", "зв'язатись", "телефон", "номер"],
};
```

```
const questions = {
  програми:
    "Ми надаємо різні освітні послуги. Ви зможете знайти, все що вас цікавить:\n☐ Академічні програми\n☐ Модульні та сертифікаційні програми\n☐ Тренінги та семінари\n☐ Корпоративні програми\nОберіть напрям, який вас цікавить.",
```

академічні:

```
"Ми пропонуємо на вибір 4 магістерські програми, які гарантовано приємно Вас здивують, бо вони створені саме для працюючих та дорослих людей!\n☐ Бізнес адміністрування або MBA - для керівників, менеджерів вищої і середньої ланки, власникам бізнесу, які прагнуть розвинути власний
```

бізнес, прокачати бізнес-процеси, професійного керувати людьми і бажають тільки більшого розвитку!\n ☐ Медіація та менеджмент конфліктів - для менеджерів, юристів, суддів, адвокатів, психологів, які кожного дня працюють з людьми та вирішують складні справи. Це можливість зрозуміти співрозмовника та досягти вирішення спору мирним шляхом\n ☐ Менеджмент проєктів та процесів - для менеджерів, котрим потрібні розширенні знання в управлінні проєктами, зв'язку з командою, побудовою плану, грамотне розподілення ресурсів й досягнення стабільності і якості\n ☐ Глобальний бізнес менеджмент - для іноземців та українців, які хочуть прокачати свій англійський та дізнатися основи керування та глобалізації бізнесу!",

адміністрування:

"Ось що можемо запропонувати: \n☐ General MBA\n ☐International MBA\n ☐MBA IT\n ☐Міні MBA",

модульні:

"Ось що можемо запропонувати: \n☐ Креативне мислення в бізнесі\n☐ Майстерність закупівель\n☐ Управління продажами\n☐ Закупівлі та управління ланцюгом постачання\n☐ Executive HRM\n☐ Маркетинг для немаркетологів\n☐ Міні MBA",

заявка: "Подати заявку можна через сайт або тут, залишивши контактні дані.",

місцезнаходження: "Ми в Києві, вул. Табірна, 30-32",

контакти:

"☎ +38 044 339 90 63\n☎ +38 097 601 61 21\n☎ +38 050 446 16

94\n✉ office@krok.edu.ua",

};

Обробка повідомлень від клієнта

```

function handleSend() {
  const userMsg = input.value.trim();
  if (!userMsg) return;
  addMessage("user", userMsg);

  if (isLive) {
    sendLiveMessage(userMsg);
    input.value = "";
    return;
  }

  const normMsg = normalize(userMsg);
  let found = null;
  let showContact = false;

  const keyHandlers = {
    вартість: (p) => (p?.cost ? `💰 Вартість: ${p.cost}` : ""),
    тривалість: (p) => (p?.duration ? `🕒 Тривалість: ${p.duration}` : ""),
    методи: (p) => (p?.methods ? `🎓 Методи навчання: ${p.methods}` :
    ""),
    структура: (p) =>
      p?.structure ? `📖 Структура: ${p.structure.join(" ")}` : "",
    умови: (p) => (p?.admission ? `🎓 Умови вступу: ${p.admission}` : ""),
    формат: (p) => (p?.format ? `🎓 Формат навчання: ${p.format}` : ""),
  };

  const programsData = window.programsData || [];
  for (const key in keywordsMap) {

```

```

const synonyms = keywordsMap[key];
const matchedPrograms = findMatchingPrograms(normMsg,
programsData);

if (synonyms.some((word) => normMsg.includes(word))) {
  if (key in keyHandlers) {
    const list = (matchedPrograms.length ? matchedPrograms :
programsData)
      .map((p) => {
        return `☐ ${p.name}\n${keyHandlers[key](p)}`;
      })
      .join("\n\n");
    found = list;
  } else {
    found = questions[key];
  }

  if (key === "заявка") {
    showContact = true;
    contactReason = "application";
  }

  break;
}

if (matchedPrograms.length) {
  found = matchedPrograms
    .map(
      (p) =>
        `☐ ${p.name}\n${Object.keys(keyHandlers)

```

```

        .reduce((acc, key) => {
            const str = keyHandlers[key](p);
            if (str) acc.push(str);
            return acc;
        }, [])
        .join("\n")}`
    )
    .join("\n\n");
}
}

const botReply =
    found ||
    "🤔 Вибачте, я не маю відповіді на це запитання. Спробуйте інакше.";
setTimeout(() => {
    addMessage("bot", botReply);
    if (showContact) {
        contactForm.style.display = "block";
        questionBlock.style.display = "none";
        addMessage(
            "bot",
            "✉️ Будь ласка, залиште свої ПІБ та email, і ми зв'яжемося з вами."
        );
    }
}, 500);
input.value = "";
}

```