

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Харківський національний університет радіоелектроніки  
Центр післядипломної освіти  
Кафедра Програмної інженерії

**КВАЛІФІКАЦІЙНА РОБОТА**  
**Пояснювальна записка**

\_\_\_\_\_ другий (магістерський) \_\_\_\_\_

(рівень вищої освіти)

Дослідження методів прогнозування при створенні Телеграм-бота  
для допомоги абітурієнтам при вступі до університету

Виконав:

студент \_\_\_\_\_ 2 \_\_\_\_\_ курсу групи ППЗдм-21-1

\_\_\_\_\_ Денисов М.І. \_\_\_\_\_

(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного забезпечення

Тип програми \_\_\_\_\_ Освітньо-наукова \_\_\_\_\_

Керівник \_\_\_\_\_ доц. Кириченко І.В. \_\_\_\_\_

(посада, прізвище, ініціали)

Допускається до захисту

Зав. Кафедри \_\_\_\_\_

З.В. Дудар

2023 р.

Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ Центр післядипломної освіти \_\_\_\_\_  
Кафедра \_\_\_\_\_ Програмної Інженерії \_\_\_\_\_  
Рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_  
Спеціальність \_\_\_\_\_ 121 – Інженерія програмного забезпечення \_\_\_\_\_  
(код і повна назва)  
Тип програми \_\_\_\_\_ освітньо-наукова програма \_\_\_\_\_  
Освітня програма \_\_\_\_\_ Інженерія програмного забезпечення \_\_\_\_\_

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ

студента \_\_\_\_\_ Денисова Миколи Ігоровича \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи: Дослідження методів прогнозування при створенні Телеграм-бота для допомоги абітурієнтам при вступі до університету  
затверджена наказом університету від «03» 04 2023 р. № 83Стз
2. Термін подання студентом роботи до екзаменаційної комісії «01» 05 2023 р.
3. Вихідні дані до роботи встановлений календарний план роботи, методичні вказівки до оформлення пояснювальної записки, методи формування гібридної архітектури програмного продукту.
4. Перелік питань, що потрібно опрацювати в роботі вивчення предметної галузі, проектування програмного забезпечення телеграм бот для допомоги абітурієнтам під час вступу до університету, реалізація проектного програмного забезпечення.

## КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Теоретичний аналіз предметної області та постановка задачі	01.04.2023 - 05.04.2023	виконано
2	Проектування програмного продукту	06.04.2023 - 10.04.2023	виконано
3	Розробка програмного продукту	11.04.2023 - 15.04.2023	виконано
4	Тестування програмного продукту	16.04.2023 - 20.04.2023	виконано
5	Підготовка пояснювальної записки	21.04.2023 - 25.04.2023	виконано
6	Підготовка презентації та доповіді	25.04.2023 - 30.04.2023	виконано
7	Перевірка роботи на плагіат та нормоконтроль	01.05.2023	виконано
8	Захист кваліфікаційної роботи	10.05.2023	виконано

Дата видачі завдання 01 квітня 2023 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_ доц. Кириченко І.В.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота магістра містить: 75 с., 29 рис., 5 табл., 16 джер., 5 додатків.

ТЕЛЕГРАМ БОТ, ПРОЕКТУВАННЯ, РОЗРОБКА, ІНТЕРНЕТ, YII2, ТЕЛЕГРАМ, ХНУРЕ, API.

Метою дослідження є комп'ютерна програма, яка допоможе абітурієнтам подавати документи до університетів.

Метою є розробка телеграм бота та навіть модуля управління для управління вмістом телеграм бота.

У дослідженні проаналізовано вимоги до програмного забезпечення телеграм бота, розглянуто архітектури та інструменти для реалізації програмного продукту.

Результатом дослідження є проектування та розробка програмного забезпечення телеграм-бота та модуля управління.

Рівень застосування: модуль наразі використовується та тестується; програмне забезпечення може використовуватися скрізь, де є мережа, і не потребує додаткового обладнання. Його мета - полегшити доступ для кандидатів. В основному воно орієнтоване на абітурієнтів та людей, які щойно вирішили навчатися і потребують короткої інформації про університет.

Функціональність програмного забезпечення може бути розширена за допомогою додаткових модулів або шляхом підключення інтелектуальних систем.

TELEGRAM BOT, DESIGN, DEVELOPMENT, INTERNET, YII2, TELEGRAM, KHNURE, API.

The purpose of the study is a computer program that will help applicants submit documents to universities.

The goal is to develop a Telegram bot and even a management module to manage Telegram bot content.

The research analyzed the requirements for Telegram bot software, considered architectures and tools for implementing the software product.

The result of the research is the design and development of the telegram bot software and the control module.

Application Level: The module is currently being used and tested. Application level: The software can be used anywhere there is a network and does not require additional hardware. Its purpose is to facilitate access for candidates. It is mainly aimed at applicants and people who have just decided to study and need brief information about the university.

The functionality of the software can be expanded with the help of additional modules or by connecting intelligent systems.

## Умови публікації пояснювальної записки

Я, Денисов Микола Ігорович  
(прізвище, ім'я, по батькові)  
студент групи ППЗдм-21-1 здобувач вищої освіти на другому (магістерському)  
рівні

кафедра програмної інженерії  
(повна назва кафедри)

заявляю: моя кваліфікаційна робота на тему «Дослідження методів прогнозування при створенні Телеграм-бота для допомоги абітурієнтам при вступі до університету», що буде представлена до ЕК для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу EIArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

## ЗМІСТ

Вступ	8
1 Теоретичний аналіз предметної області та постановка задачі	10
1.1 Аналіз літератури, яка використана у роботі	10
1.2 Аналіз методів взаємодії приймальної комісії університету з абітурієнтами	11
1.3 Аналіз аудиторії користувачів месенджерів	13
1.4 Огляд існуючих рішень	15
2 Проектування програмного продукту	18
2.1 Концептуальна модель програмного продукту	18
2.2 Компонентна архітектура	21
2.3 Визначення загальних вимог до програмного продукту	25
2.4 Архітектура програмного забезпечення	26
2.5 Інтерфейс адміністративної панелі на Yii2	29
2.6 Інтерфейс телеграм бота	29
2.7 Гібридна архітектура програмного продукту	30
2.8 Захист даних, що передаються	30
3 Розробка програмного продукту	32
3.1 Вибрані технології розробки	32
3.2 Серверна частина програмного продукту	46
3.3 Адміністративна панель чат бота	53
Висновки	55
Перелік джерел посилання	56
Додаток А Перелік джерел посилання за науковими напрямками керівника та науковців кафедри програмної інженерії	58
Додаток Б Звіт результатів Перевірки на унікальність тексту в базі ХНУРЕ	59
Додаток В Презентаційні слайди для захисту кваліфікаційної роботи	60
Додаток Г Текст наукової публікації за темою кваліфікаційної роботи	72
Додаток Д Експертний висновок результатів перевірки кваліфікаційної роботи на відповідність оформлення Вимоги ДСТУ 3008:2015	75

## ВСТУП

Університет є одним з провідних центрів підготовки фахівців у сфері ІТ, бізнесу та телекомунікацій. Пропозиція напрямів підготовки та освітніх програм досить широка, що змушує абітурієнтів витратити години, дні і тижні на з'ясування деталей, які їх цікавлять, у кожного з основних представників у сфері освітніх послуг.

Стан комп'ютерного ринку диктує актуальність роботи. Щороку він наповнюється новим програмним забезпеченням, веб-сервісами та мобільними додатками, які можна використовувати не тільки для розвитку комунікації, але й для зміцнення бренду університету як навчального закладу.

Метою даної магістерської роботи є розробка програмного забезпечення "Абітурієнт-бот університету", яке автоматизує завдання, пов'язані з інформуванням та взаємодією з абітурієнтами у приймальній комісії.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

- визначити вимоги, які повинно мати програмне забезпечення;
- визначити вимоги, які повинен мати додаток для управління контентом цього програмного забезпечення;
- розробити архітектуру програмного забезпечення;
- розробити дружній користувацький інтерфейс для модуля управління ботами в Telegram;
- реалізувати розроблене програмне забезпечення.

Головним привілеєм будь-якого закладу у сфері вищої освіти є підготовка компетентних фахівців, які користуються попитом на ринку праці. В умовах сильної конкуренції серед закладів вищої освіти необхідно підтримувати комунікацію з кандидатами.

Таким чином, рішення про розробку чат-бота було мотивоване бажанням оптимізувати діяльність співробітників та студентів, які працюють стажерами у приймальній комісії університету та змушені витратити багато часу на ретрансляцію інформації з відкритих джерел.

Практична цінність розробленого чат-бота полягає в його прив'язці до конкретних користувачів - абітурієнтів ХНУРЕ.

Як джерела інформації ми використовували офіційну документацію з розробки РНР, бланки замовлень та інші документи, що містять інформацію про кількість бюджетних місць, форми навчання та напрями підготовки.

# 1 ТЕОРЕТИЧНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

Перш ніж приступити до проектування та розробки програмного продукту, необхідно вивчити існуючі аналоги та визначити їх переваги та недоліки. У даному розділі проведено огляд існуючих теоретичних та практичних напрацювань щодо автоматизації процесу вибору кар'єри абітурієнтом у конкретному університеті. Висвітлено актуальність вирішення проблеми, зроблено огляд використаної літератури, встановлено мету даної роботи та визначено основні завдання.

## 1.1 Аналіз літератури, яка використана у роботі

Літературу, присвячену основній частині роботи, можна розділити на дві групи. До першої групи належать [1-7]. Це джерела, з яких була почерпнута різноманітна інформація, використана для аналізу теми. Другу групу складають літературні джерела, які описують можливі рішення [8-13]. Ця група є найбільш важливою для проходження практики та написання магістерських робіт, оскільки на основі цих методів можна створити програмний продукт, який задовольняє всім вимогам розробки. Статті та публікації цієї групи в основному були взяті з відомих та популярних джерел, що займаються розробкою та супроводом програмного забезпечення. Аналіз показав, що обрані групи найкраще відображають обсяг і характер проблеми. Перша група літератури включає всі специфічні поняття, що характеризують обрані технології. Друга група описує загальні концепції та стандарти розробки програмного забезпечення для телеграм-ботів.

## 1.2 Аналіз методів взаємодії приймальної комісії університету з абітурієнтами

Взаємодія між приймальною комісією університету та абітурієнтами може відбуватися двома способами: офлайн або дистанційно.

Під офлайновими методами ми маємо на увазі:

- 1) консультації членів приймальної комісії університету;
- 2) участь абітурієнта у Дні відкритих дверей;
- 3) участь абітурієнта у наукових заходах, організованих університетом;
- 4) участь кандидата у творчих заходах, організованих університетом ("День відкритих дверей ХНУРЕ", конкурси тощо);
- 5) участь здобувача в екскурсіях, організованих ХНУРЕ.

Дистанційне навчання передбачає :

- 1) телефонне консультування абітурієнта членом приймальної комісії;
- 2) консультування абітурієнта поштою членом приймальної комісії;
- 3) консультування абітурієнта електронною поштою членом приймальної комісії;
- 4) консультація вступника членом приймальної комісії на сайті приймальної комісії (див. рисунок 1.1);
- 5) консультація вступника членом приймальної комісії у соціальній мережі Facebook (див. рисунок 1.2);
- 6) консультація члена приймальної комісії абітурієнту у месенджері Messenger.

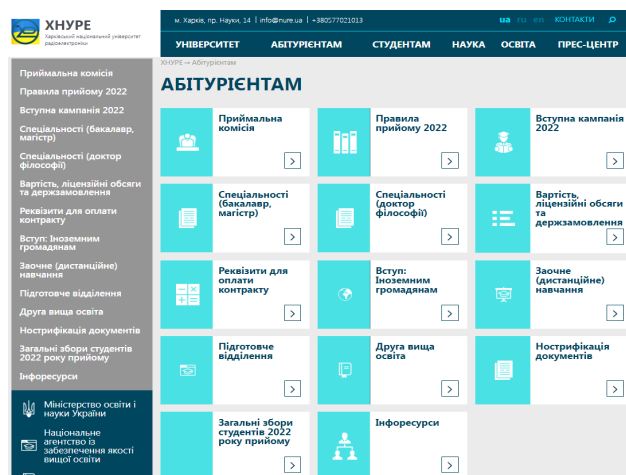


Рисунок 1.1 – Сторінка приймальної комісії на офіційному веб-сайті університету

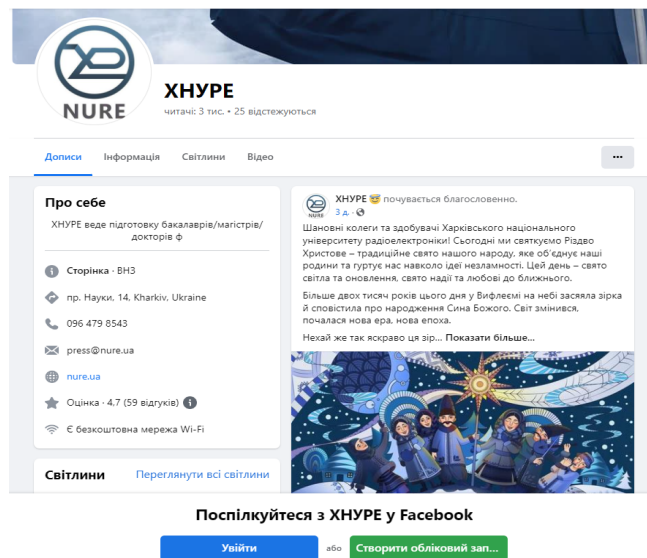


Рисунок 1.2 – Офіційна сторінка університету у соціальній мережі Facebook

Приймальна комісія зараз динамічно опановує методи дистанційної взаємодії з абітурієнтами. Процес збереження "старих" методів і засобів офлайнової взаємодії відбувається стрибкоподібно і протягом тривалого періоду часу, незважаючи на очевидні переваги дистанційних методів взаємодії, серед яких: загальнодоступність, низька вартість, практичність для всіх учасників взаємодії, наочність і можливість не втратити історію листування.

Месенджер - поки що вважається неперевершеною платформою, але ця остання кваліфікаційна робота має на меті виправити та компенсувати це упущення. Розглянемо докладніше основні недоліки особистої взаємодії:

1. Обмеження в часі. Реєстраційна служба має певні робочі години, які обмежені робочими днями кожного дня. Двері реєстраційної служби зачиняються о 16.00, працівники залишають свої робочі місця і повертаються лише по суботах на 4 години. Через обмежені години консультацій та роботи, університет часто втрачає записи.

2. Незадовільна "прозорість" комунікації. При спілкуванні з працівником приймальної комісії в інтернеті, наприклад, через соціальні мережі, електронну пошту або месенджери, зберігається історія листування. До неї можна отримати доступ у будь-який момент, якщо це буде потрібно для процедури апеляції.

3. Високі часові витрати. Організація спеціальних заходів для кандидатів, які можуть слугувати "комунікаційним майданчиком" для всіх зацікавлених сторін (кандидати, батьки, співробітники університету), вимагає багато часу, який рідко "амортизується" кількістю учасників. Коротше кажучи, настав час зосередитися на способах віддаленої взаємодії з кандидатами, особливо через Інтернет. Конкретно це означає консультації з абітурієнтами через:

- офіційний веб-сайт;
- сторінки та акаунти в соціальних мережах;
- месенджери.

Водночас месенджери розглядаються як невикористана платформа, яка може бути більш зручним способом зв'язку з абітурієнтами та приймальними комісіями.

### 1.3 Аналіз аудиторії користувачів месенджерів

Месенджер постійно стає найпоширенішим каналом комунікації і навіть завойовує публічний простір за допомогою чатів та каналів. Вже зараз можна сказати, що канали в месенджері стали зручною базою для використання контенту для багатьох користувачів медіа. Коли йдеться про споживання новинного контенту, користувачі Telegram обирають між новинними повідомленнями від агрегаторів та каналами, які публікують медіаконтент 1-2 рази на день - їхня аудиторія значно більша, ніж, наприклад, у ЗМІ, які бомбардують своїх підписників повідомленнями протягом усього дня.

Використання Messenger не лише для особистого спілкування, а й для спілкування в публічних чатах та читання публічних каналів - один із трендів 2019 року. Чати та канали перетворюють Messenger на соціальне середовище та залучають нових користувачів. Telegram, безсумнівно, є трендсеттером у цій сфері, що також може пояснити стрімке поширення Messenger. Коротше кажучи, можна лише сказати, що тенденція зростання інтересу до месенджерів є безперервною. В умовах, коли обсяг згадувань подвоївся, лідером зростання став Telegram, який збільшився в чотири рази.

Зростання інтересу до Telegram пов'язане з активним використанням публічних каналів: з року в рік кількість користувачів ефірних каналів збільшується з сотень до десятків тисяч. Найбільше зросли канали з нечастим авторським контентом, такі як розважальні, освітні та ті, що мають сильний політичний і новинний порядок денний [2].

Згідно з аналізом незалежного видання Telegramstore, за віком та соціальним статусом користувачі у секторі мовлення розподіляються наступним чином:

- вік 18-45 років;
- мають вищу освіту або вищу освіту;
- мають інтелектуальну професію (значна частка комп'ютерних спеціалістів).

Функціонал систем обміну повідомленнями, особливо Telegram, здебільшого орієнтований на мобільну аудиторію, яка швидко використовує телефони для роботи або дозвілля: користувачеві не потрібно заходити на сайт, йому достатньо надіслати повідомлення для роботи. Підсумовуючи цей розділ, можна зробити такі висновки:

- конкуренція між установами, що пропонують освітні послуги, сильна як ніколи;
- загальна кількість вищих навчальних закладів в Україні перевищує 600;
- класифіковано типи взаємодії між абітурієнтами та приймальними комісіями вищих навчальних закладів - виділено два основні типи взаємодії;
- встановлено, що найбільш ефективним методом взаємодії є дистанційний метод;
- виявлено різні види дистанційної взаємодії, зокрема використання месенджерів як платформи для взаємодії між абітурієнтами та приймальною комісією;
- проаналізувавши цільову аудиторію месенджерів, можна дійти невтішного висновку, що найцікавішою платформою для спілкування є Telegram Messenger.

## 1.4 Огляд існуючих рішень

Оскільки додаток розроблявся для харківського університету, варто розглянути харківських конкурентів. Наразі існує 2 більш-менш порівнянних рішення для університетів, які також є найближчими з точки зору необхідного функціоналу: "Абітурієнт ХНУРЕ" та "Хочу до ХНУРЕ".

"Абітурієнт ХНУРЕ" - це мобільний додаток для абітурієнтів, які вступають до ХНУРЕ [3]. Додаток має дуже приємний користувацький інтерфейс, з можливістю входу в особистий кабінет та редагування особистих даних, а також додаток пропонує можливість проходити вікторини і таким чином заробляти бали, з можливістю виграти один з головних призів.

"Хочу до ХНУРЕ" - мобільний додаток для абітурієнтів ХНУРЕ [4]. Додаток має багато функцій, але при натисканні на них половина з них перенаправляє на веб-сайт, що знижує автономність додатку. Крім того, реалізована функція "Калькулятор ЗНО", зміст якої полягає у виборі предметів єдиного національного іспиту, з яких складається ймовірність вступу абітурієнта на певні спеціальності.

Згідно з інформацією про реалізацію програм, усі без винятку концепції є повноцінними програмними продуктами, які в тій чи іншій мірі забезпечують основні функції, необхідні для комфортної роботи абітурієнтів. Для наочного представлення переваг та недоліків систем було складено таблицю 1.1.

З таблиці 1.1 видно, що еталонним продуктом для кандидатів у дослідженні є той, що містить найповніший функціонал програмного забезпечення. Це також свідчить про те, що ця програма є найбільш зручною у використанні.

Виходячи з цих факторів, було прийнято рішення розпочати розробку програмного забезпечення Telegram Bot University.

Перш ніж представити покроковий алгоритм розробки чат-бота, ми повинні повністю визначитися з його основним призначенням. По-перше, взаємодія у вигляді повідомлень створює високий рівень залученості. По-друге, за допомогою чат-бота можна визначити швидкість взаємодії. Електронна пошта та соціальні мережі не завжди є найкращим рішенням.

Таблиця 1.1 - Порівняння функціоналу систем

Параметри	Програми		
	«Телеграм бот університет»	«Абітурієнт університет»	«Бажаю до університету»
Окреме місце у телефоні для програми	+	+	+
Можливість перегляду інформації про напрямки ВНЗ	+	+	+
Можливість перегляду новин та заходів	+	+	+
Дизайн	+	-	-
Кросплатформеність	+	+	+
Автономність	+	-	-
Зручний та чуйний інтерфейс	+	-	-
Отримання інформації на пошту	+	-	-

Користувач не завжди знає, хто і коли відповість. Навіть найбільші групи в соціальних мережах не гарантують негайної відповіді, яка потрібна кандидату або користувачеві в цей момент. Користувачі публікують, видаляють і чекають (або не чекають) на відповідь, тому що сама модель комунікації не дозволяє миттєвої інтерактивності. З чат-ботом все відбувається інакше: користувач може отримати негайну відповідь на свої запитання. Близькість, швидкість та перелік функцій - це те, що пропонує обмін повідомленнями. Компанії просто зобов'язані бути там, де знаходяться їхні користувачі, щоб говорити з ними на рівних. І з цієї точки зору такий месенджер, як Telegram, є чудовою платформою для взаємодії з суспільством. Як ззовні, так і всередині університету [5].

Споживачі у віці 18-35 років є найбільш вимогливою та цікавою цільовою групою для рекламодавців. Традиційні соціальні мережі не здатні підтримувати їх інтерес протягом тривалого часу.

Підсумовуючи, перевагами використання телеграм-повідомлень прийнятною комісією університету є:

1) Залучення - чат-бот може бути доповнений цікавим переліком функцій, таких як ігри, опитування та інші форми взаємодії, які "втягують" користувача в діалог.

2) Мобільність і популярність за платформами - телеграм вважається крос-платформним додатком, доступним не тільки з мобільного телефону, але й зі стаціонарного комп'ютера.

3) Невразливість і безпека - розробники Telegram неодноразово заявляли, що додаток є безпечним завдяки використанню оригінального алгоритму шифрування MTProto.

4) Доступність і сфера застосування - немає жодних обмежень.

Аналіз, проведений з метою визначення відповідного програмного продукту для полегшення вибору абітурієнтами спеціальності, показав, що існуючі програмні продукти повністю вирішують поставлену задачу [6].

## 2 ПРОЕКТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ

Після того як проведено аналіз та обґрунтовано необхідність виконання даної роботи, необхідно провести проектування реалізованого програмного продукту.

### 2.1 Концептуальна модель програмного продукту

Для того, щоб описати роботу програмного забезпечення, необхідно створити концептуальну модель системи, що розробляється. Ця модель повинна бути адекватною предметній області, тобто містити знання про всіх акторів, що беруть участь у процесах системи.

Перш за все, необхідно визначити акторів, які будуть взаємодіяти з програмним продуктом:

- 1) Адміністратор програми – фахівець, який керує всім вмістом системи та має повний доступ до всього функціоналу;
- 2) Користувачі – переважно абітурієнти, які претендують на місце в університеті, але в подальшому додатком може користуватися і студент.

Також виділимо основні прецеденти:

- 1) Відображення повідомлень, контактів, інформації про вступ до ХНУРЕ та шкіл.
- 2) Виконавець – користувач. Реалізовано в додатку Telegram. Опис кроків:
  - запустити бота;
  - виконати опитування;
  - перейдіть до модуля "Повідомлення", "Контакти", "Як потрапити до ХНУРЕ" або "Школа" – за допомогою клавіш навігації;
  - оберіть пункт меню.
  - перейдіть до модуля "Меню" за допомогою навігаційних клавіш, щоб побачити навігаційне меню. Додати повідомлення, контакти, інформацію про вступ до ХНУРЕ та школи.

3) Виконавець – адміністратор. Виконується в адміністративній частині веб-додатку. Опис кроків:

- увійдіть в систему;
- перейдіть до модуля "Повідомлення", "Контакти", "Телезапис" або "Навчальні заклади" через навігаційне меню;
- додайте або змініть повідомлення, контакти або іншу інформацію.

Взаємодія акторів з програмним продуктом та опис прецедентів представлено графічно на рисунку 2.1 у вигляді діаграми прецедентів.

Діаграма прецедентів в UML – це діаграма, яка відображає взаємозв'язки між акторами та прецедентами і є компонентом прецедентної моделі, що дозволяє показати концепцію на концептуальному рівні [7].

Прецедент – це можливість модельованої концепції (частина її функціональності), на основі якої користувач може отримати конкретний, вимірюваний і бажаний результат. Факт відповідає окремому сервісу концепту, визначає одну з альтернатив його використання і представляє традиційний спосіб взаємодії користувача з концептом. Варіанти використання також використовуються для визначення зовнішніх умов концепції.

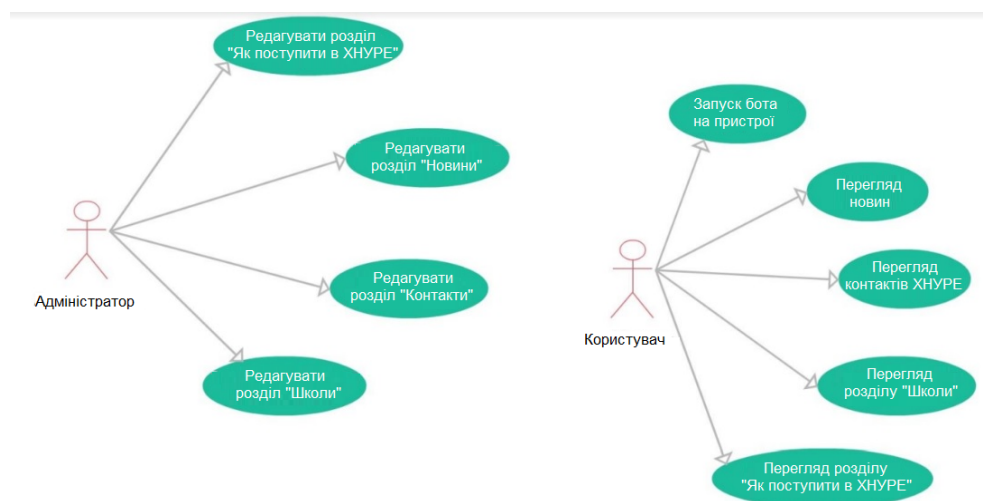


Рисунок 2.1 – Діаграма прецедентів програмного продукту

Процес відповіді користувача на запит може бути представлений діаграмою діяльності (див. рисунок 2.2).

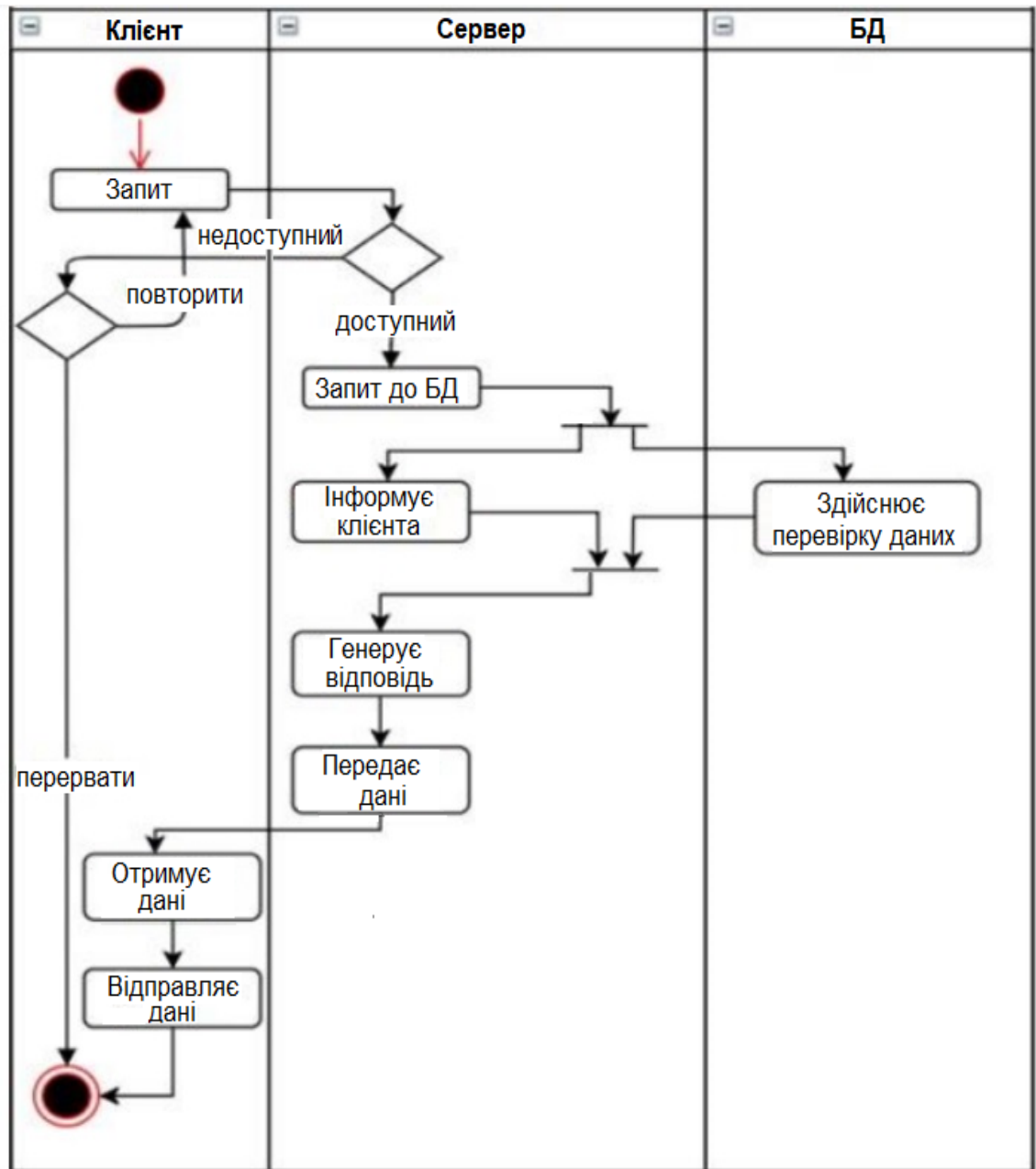


Рисунок 2.2 – Діаграма діяльності процесу «Пройдіть опитування»

Діаграма діяльності: діаграма UML, що представляє дії в станах, описаних на діаграмі станів. Під діяльністю розуміють класифікацію дії, що виконується послідовно і синхронно, на підпорядковані компоненти: вкладені дії та одиночні дії англійською мовою, дії, пов'язані потоками від виходів першого вузла до входів другого [7]. Діаграми діяльності використовуються для моделювання бізнес-процесів, науково-технічних процесів, паралельних і непаралельних концепцій. Діаграми діяльності складаються з обмеженої кількості фігур, з'єднаних стрілками.

Основні фігури :

- прямокутники з кривими - дії;
- ромб - рішення;
- великі смуги - початок (гілка) і кінець дії (гілка);
- темна область - початок процесу);
- темна область з лінією (кінець процесу).

Стрілки йдуть знизу до кінця процесу і вказують на потоки диспозицій або об'єктів.

## 2.2 Компонентна архітектура

Компонентна архітектура визначає підхід до проектування та розробки систем, який використовує методи розробки програмного забезпечення. Він наголошує на декомпозиції проекту на окремі функціональні або логічні компоненти, які забезпечують чітко визначені інтерфейси з методами, подіями та властивостями. Цей підхід пропонує вищий рівень абстракції, ніж об'єктно-орієнтована розробка, і не зосереджується на таких питаннях, як протоколи зв'язку або загальні стани [8].

Існує два підходи до реалізації програмного забезпечення для здобувачів вищої освіти:

- однокомпонентна архітектура;
- багатокомпонентна архітектура.

### 2.2.1. Однокомпонентна архітектура

В однокомпонентній архітектурі програма встановлюється з App Store разом із завантаженим контентом. Вся логіка знаходиться в одній програмі, індивідуально для кожної платформи.

До переваг такої архітектури відносяться:

- зменшення навантаження на розробників;
- незалежність від інтернет-з'єднання;
- безпека передачі даних.

До недоліків цієї архітектури слід віднести:

- довгий час для оновлення контенту;
- не дозволяє соціальну взаємодію між користувачами;
- труднощі при переході на нову платформу.

### 2.2.2. Багатокомпонентна архітектура

Багатокомпонентна або клієнт-серверна архітектура - це додаток з графічним інтерфейсом, який взаємодіє з сервером бази даних, що зберігає постійно оновлюваний контент і більшу частину бізнес-логіки. Загалом, цей стиль описує відносини між клієнтом і сервером (або серверами), де клієнт виконує ряд дій, чекає відповіді і обробляє відповідь після її отримання. Сервер, з іншого боку, дозволяє користувачеві виконати необхідну обробку для отримання результату і передає його.

Основними перевагами багатокомпонентної архітектури є:

- простота підтримки (шари не залежать один від одного, що дозволяє вносити оновлення або зміни, не впливаючи на програму в цілому);
- масштабованість (шари організовані відповідно до їх використання, що дозволяє легко масштабувати додаток);
- гнучкість (кожним шаром можна керувати та масштабувати незалежно, що підвищує гнучкість);
- доступність (додатки можуть використовувати модульну архітектуру, що дозволяє системі використовувати компоненти, які можна легко масштабувати, що підвищує доступність).

Порівнюючи переваги та недоліки розглянутих архітектур, було зроблено висновок, що багатокомпонентна архітектура є найбільш придатною для реалізації програмного продукту.

Використане архітектурне рішення є різновидом багатокomпонентної архітектури: трирівнева архітектура.

Ця архітектура дозволяє розумно розподілити модулі обробки даних, які передаються одним або декількома незалежними серверами. Очевидною перевагою такої архітектури є те, що сервери можуть взаємодіяти один з одним, що дозволяє розділити систему на більш детальні функціональні блоки з конкретними функціями. Графічно ця архітектура представлена на рисунку 2.3.

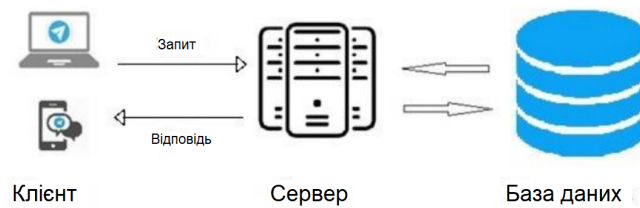


Рисунок 2.3 – Трирівнева компонентна архітектура

Трирівнева архітектура компонентів складається з трьох основних рівнів:

Клієнтський рівень.

Це графічний інтерфейс, який забезпечує функції введення та відображення даних. Цей рівень зберігає просту бізнес-логіку та містить стан додатку. У системі, що розробляється, цей рівень буде присутній:

- бізнес-логіка для відправки запиту на сервер, отримання та обробки отриманих даних;
- бізнес-логіка для підключення робочих компонентів через сервіси та провайдерів;
- взаємодія елементів графічного інтерфейсу користувача.

Рівень сервера додатків

Цей рівень є сполучною ланкою між клієнтським рівнем і рівнем бази даних, де зберігається більша частина бізнес-логіки. На цьому рівні система, що розробляється, буде містити наступні елементи

- бізнес-логіка отримання та обробки замовлень клієнтів;
- створення колекцій для бази даних;
- бізнес-логіка взаємодії з базою даних.

Рівень бази даних зберігає та обробляє дані і взаємодіє виключно з рівнем сервера додатків. Для представлення взаємозв'язку між логічними та фізичними модулями необхідно створити діаграму компонентів. Діаграма компонентів UML графічно зображена на рисунку 2.4.

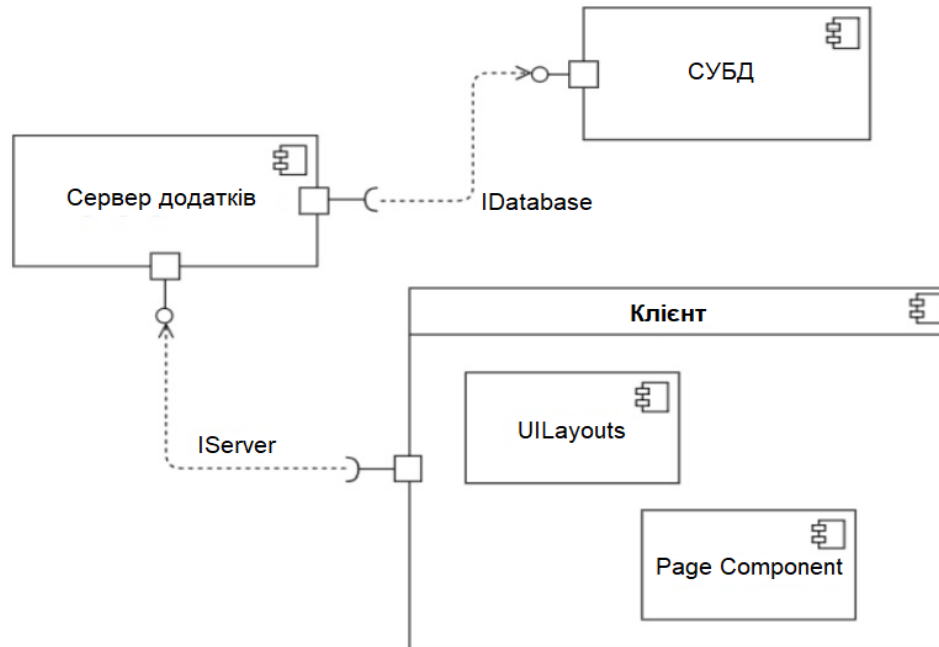


Рисунок 2.4 – UML діаграма компонентів

Компонент Сервер додатків реалізує бізнес-логіку отримання та обробки запитів клієнтів, надає інтерфейс `IServer` та використовує інтерфейс `IDatabase`.

Компонент СУБД реалізує доступ до даних та надає інтерфейс бази даних.

Клієнтський компонент - споживає інтерфейс `IServer`, складається з двох складових частин:

- `UILayouts` – частина компонента, що містить елементи інтерфейсу користувача;

- компонент сторінки - частина компонента, що слугує для відображення нової інформації на екрані та пунктів меню [9].

### 2.3 Визначення загальних вимог до програмного продукту

Цей розділ визначає вимоги до програмного продукту.

Вимоги до адміністратора програмного забезпечення :

- програмне забезпечення повинно дозволяти додавати нові розділи до навігаційного меню;
- програмне забезпечення повинно дозволяти працювати з різними даними;
- всі зміни повинні вноситися через інтерфейс адміністрування;
- додаток повинен зберігати всі дані в базі даних;
- програмне забезпечення має бути кросплатформним.

Визначте вимоги до користувача:

- всі зміни повинні бути автоматично доступні в програмному забезпеченні;
- програмне забезпечення повинно мати можливість запитувати список шкіл;
- програмне забезпечення повинно мати можливість переглядати повідомлення;
- програмне забезпечення повинно мати можливість переглядати контакти;
- програмне забезпечення повинно дозволяти доступ до розділу "Як зареєструватися в ХНУРЕ";
- програмне забезпечення повинно відповідати на питання щодо заповнення персональних даних.

Функціональні вимоги до програмного продукту

Виходячи з вищезазначених вимог, ми визначили перелік функціональних можливостей програмного продукту, які необхідно розробити:

- можливість створення повідомлень;
- можливість редагування шкіл;
- можливість редагування контактів;
- можливість редагування контактів;
- можливість редагування розділів у програмному забезпеченні;
- можливість редагування користувачів;
- можливість видалення користувачів;
- перевірка правильності введення інформації.

## 2.4 Архітектура програмного забезпечення

Обрана трирівнева архітектура розроблена в стилі сервіс-орієнтованої архітектури. Така архітектура дозволяє створювати розподілене програмне забезпечення, що складається з набору незалежних сервісів.

Обрана трирівнева архітектура розроблена в стилі сервіс-орієнтованої архітектури. Сервіс-орієнтована архітектура дозволяє створювати розподілене програмне забезпечення, що складається з набору незалежних сервісів [10].

Така архітектура має ряд переваг, однією з яких є її здатність швидко адаптуватися до мінливих потреб і завдань інформаційного світу.

Наразі найбільшого поширення набули веб-сервіси SOAP та REST. Основні відмінності між SOAP та REST підходами [10] :

Пакети запиту і відповіді в REST набагато менші, ніж відповідні пакети SOAP. SOAP розглядає рівень передачі даних HTTP як пасивний спостерігач, тоді як REST - це активна взаємодія з використанням існуючих методів HTTP, наприклад, GET, POST, PUT, DELETE.

Модель SOAP підтримує певний ступінь самоаналізу, що дозволяє розробникам сервісів описувати свій API у файлі WSDL. Модель REST, з іншого боку, уникає складності WSDL і використовує більш інтуїтивно зрозумілий інтерфейс, заснований на стандартних методах HTTP.

REST обмежується операціями CRUD з використанням відповідних методів HTTP. Порівнюючи переваги та недоліки двох підходів до реалізації веб-сервісів, веб-сервіси REST є найбільш зрозумілими та простими в реалізації, а також мають найменший розмір пакету.

RESTful API також можна кешувати. Це означає, що клієнт може кешувати всі відповіді. Це покращує продуктивність API.

RESTful API часто називають RESTful веб-сервісами, оскільки вони застосовують як принципи REST, так і протоколи HTTP.

В принципі, доступ до багатьох ресурсів здійснюється за допомогою методів HTTP і представлений у стандартному форматі, зазвичай XML. Поки гіпертекст

стандартизований, він працює для дійсних типів онлайн-медіа. Деякі приклади показані на рисунку 2.5.

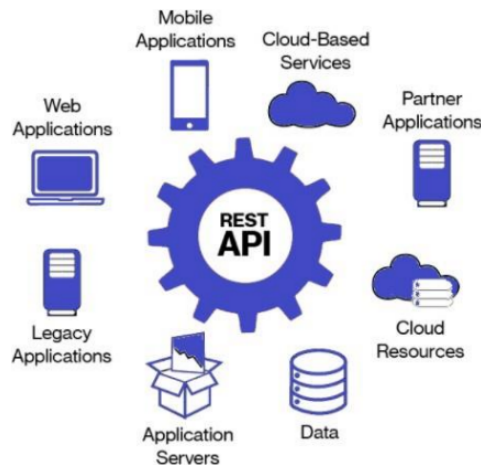


Рисунок 2.5 – Візуальне представлення REST

REST використовує 4 основні методи HTTP: GET, POST, PUT, DELETE. Здебільшого кожен з цих методів використовується для виконання певної дії з CRUD (Create, Read, Update, Delete).

POST - створення, GET - читання, PUT - оновлення, DELETE - видалення. Приклад реалізації REST-веб-сервісу графічно зображено на рисунку 2.6.

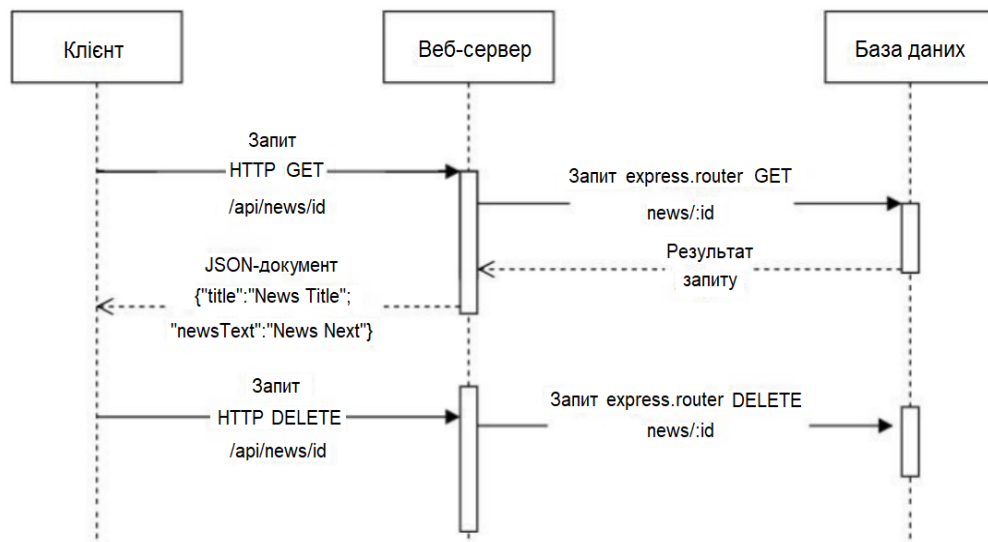


Рисунок 2.6 – REST веб-сервіс

На рисунку 2.6 показано початковий вигляд REST API, створений за допомогою Yii2 та PHP для забезпечення взаємодії з базою даних MySQL.

Цей програмний продукт складається з двох частин. Це орієнтований на користувача інтерфейс, який надає користувачам повідомлення, інформацію про школу тощо. Система управління контентом Telegram-бота - це також інтерфейс адміністрування, до якого адміністратори мають доступ для оновлення контенту та управління програмним забезпеченням.

Тут ми розглянемо архітектуру цих двох додатків окремо:

- інтерфейс, орієнтований на користувача;
- інтерактивний, недоступний інтерфейс адміністрування.

## 2.5 Інтерфейс адміністративної панелі на Yii2

Структура веб-сайту представлена на зображенні нижче (див. рисунок 2.7).

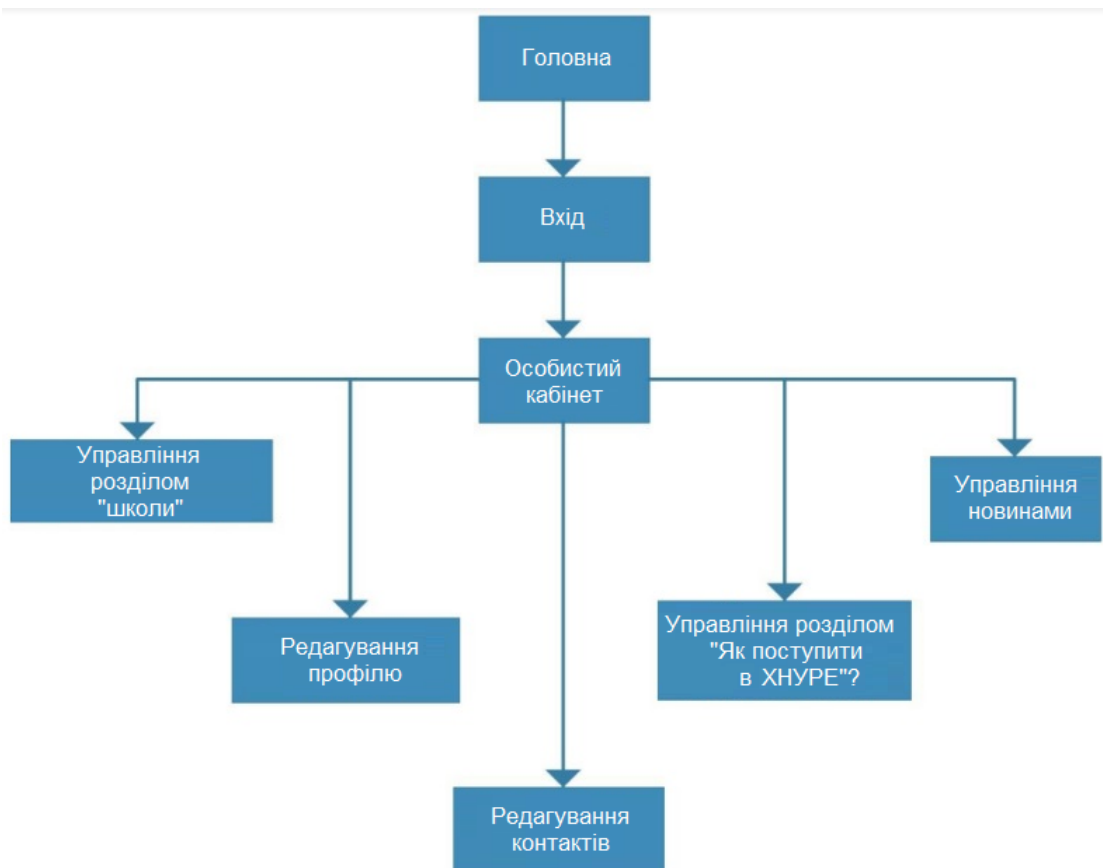


Рисунок 2.7 – Структура адміністративної панелі

Для керування контентом відображений користувачеві було розроблено веб-додатки на фреймворку Yii2 й основну структуру програми можна бачити на рисунку 2.7.

## 2.6 Інтерфейс телеграм бота

Основні елементи інтерфейсу чат-бота є універсальними і входять до складу всіх додатків для обміну повідомленнями. Як зазначено в документації Telegram API, бот може спілкуватися з телеграм-серверами двома способами:

1. `getUpdates` - pull: бот постійно звертається до телеграм-сервера для пошуку нових повідомлень;
2. `setWebhook` - push: при отриманні нових повідомлень телеграм-сервер надсилає їх боту.

Різниця між цими двома методами проілюстрована на рисунку 2.8.

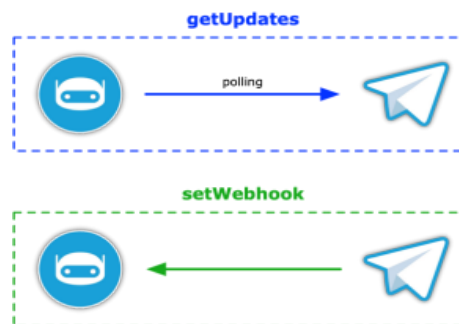


Рисунок 2.8 – Способи взаємодії з ботом

Очевидно, що другий спосіб (`setWebhook`) є більш раціональним для всіх учасників процесу. Однак він має певну складність: на стороні бота хтось має отримувати повідомлення з Telegram, а це означає, що потрібен відповідний веб-сервер або пристрій. Крім того, потрібен SSL-сертифікат, який гарантує, що комунікація відбувається через захищений HTTPS-канал. Це необхідна і фундаментальна частина взаємодії Webhook.

Програмне забезпечення включає в себе веб-сервер і телеграм-бот. Основні ідеї, реалізовані в додатку, представлені нижче:

- обробка повідомлень користувача в режимі реального часу;
- миттєве представлення можливих відповідей користувачеві;
- обробка рішення користувача в реальному часі та надсилання відповіді.

Веб-сервер складається з програмного забезпечення телеграм-сервера бота, тобто зв'язку з базою даних, команди якої запитує користувач.

## 2.7 Гібридна архітектура програмного продукту

Результатом є дві програми, що працюють окремо, використовуючи REST API. Цілісність архітектури з REST API, що взаємодіє з двома клієнтськими програмами, проілюстровано на рисунку 2.9.

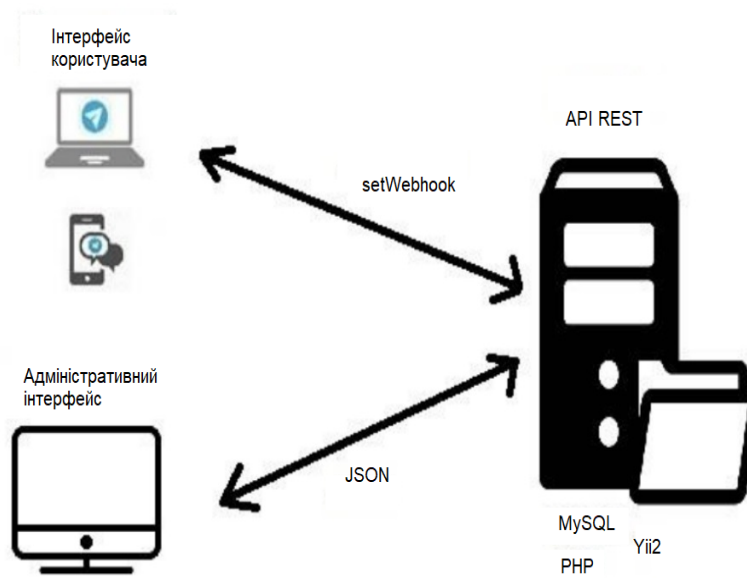


Рисунок 2.9 – Гібридна архітектура набору програм MEAN

Як показано на рисунку 2.9, один REST API надає дані двом додаткам на стороні користувача, що базуються на різних частинах технологічного стеку MEAN.

## 2.8 Захист даних, що передаються

Для того, щоб виключити можливість перехоплення даних сторонніми програмами при передачі даних від сервера до клієнта за протоколами HTTP, необхідно забезпечити захист даних, що передаються.

Для захисту даних, що передаються, на сервері організуються HTTPS-запити, а також ідентифікація та авторизація клієнта. Для цього під час виконання запиту передається токен. Веб-токен JSON складається з трьох частин, розділених крапками:

- HEADER
- PAYLOAD
- SIGNATURE

На рисунку 2.10 показано схему ідентифікації клієнта на сервері.

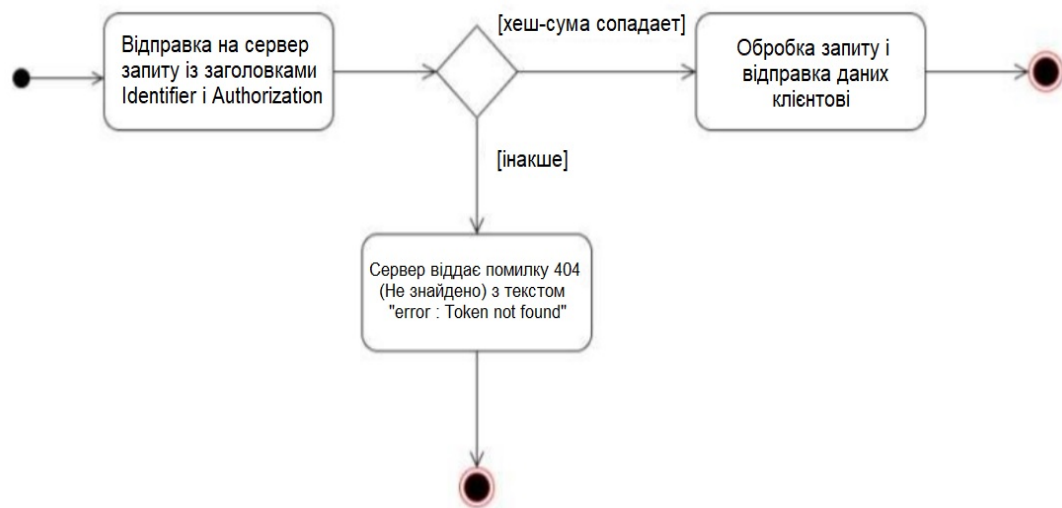


Рисунок 2.10 – UML діаграма ідентифікації клієнта на сервері

Вихідні дані складаються з трьох розділених крапками URL-рядків Base64, які можуть бути легко передані в середовищі HTML і HTTP і є більш компактними, ніж стандарти на основі XML, такі як SAML.

## 3 РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ

У цьому розділі буде розглянуто весь процес розробки програмного продукту і виявлено проблеми і методи їх вирішення.

### 3.1 Вибрані технології розробки

#### 3.1.1. Аналіз платформи для розробки

Існує багато мов програмування, які спеціалізуються на виконанні різних завдань. Кожна з них характеризується унікальним набором операторів і певним синтаксисом. Розглянемо 3 інші відомі мови, що використовуються у веб-розробці: PHP, Ruby та Python. Розглянувши їхні переваги та недоліки, ми проаналізуємо їхню популярність, чим вони відрізняються і наскільки затребувані певною аудиторією.

PHP - це серверна мова програмування, розроблена Расмусом Лерддорфом як інструментарій для створення динамічних та інтерактивних веб-сайтів.

Мова виявилася дуже гнучкою та потужною, саме тому вона користується великою популярністю і використовується в проектах різного масштабу: від простого блогу до найбільших веб-додатків в Інтернеті.

Ruby - це динамічна об'єктно-орієнтована мова програмування, розроблена Юкіхіро Мацумото. Ruby була натхненна подібними мовами, такими як Perl, Eiffel та Smalltalk. Ruby використовується у веб-розробці як частина веб-фреймворку з відкритим вихідним кодом Rails, широко відомого як Ruby on Rails (RoR). Python широко використовується як інтерпретована мова для написання програм у різних напрямках (хоча існують перекладачі Python).

Концепція мови датується кінцем 1980-х років і була розроблена Гвідо ван Россумом. Python є мультипарадигмальною мовою програмування: вона дозволяє поєднувати процедурні, об'єктно-орієнтовані, функціональні та процедурні стилі програмного коду.

Щоб отримати чітке уявлення про мови серверного програмування PHP, Python або Ruby, важливо розуміти їх переваги та недоліки. Кожна мова має свої переваги та недоліки. Все залежить від ваших потреб та очікувань від цих мов. Ви можете вибрати мову, яка найкраще відповідає потребам і вимогам роботи програміста. Нижче наведено переваги та недоліки кожної з трьох мов програмування (див. таблиці 3.1 та 3.2).

Таблиця 3.1 – PHP, Python та Ruby: переваги

PHP	Ruby	Python
Безкоштовне програмне забезпечення за ліцензією PHP	Кросплатформеність та відкритий вихідний код	Легкий та швидкий у вивченні
Легкий у освоєнні (висока швидкість навчання)	Може бути вбудований у мову розмітки гіпертексту	Підтримується безліччю платформ та операційних систем
Велика спільнота користувачів та розробників	Мова програмування надвисокого рівня (VHLL)	Читабельний та організований синтаксис
Розширена підтримка баз даних	Простий і зрозумілий синтаксис, що дозволяє розробнику-початківцю дуже швидко вивчити мов	Забезпечення швидкого прототипування та динамічних семантичних властивостей
Надає велику кількість доступних розширень та вихідних кодів	Просте підключення до баз даних DB2, MySQL, Oracle та Sybase	Величезна спільнота підтримки
Дозволяє виконання коду в обмежених середовищах виконання	Створені на Ruby великі масштабовані програми прості у супроводі	Проста побудова додатків шляхом тестування та імпорту необхідних функцій
Надається можливість управління нативними сесіями та розширення API	Наявність вбудованого відладчика та гнучкого синтаксису	Реюзабіліті (можливість повторного використання) за рахунок ретельної розробки пакетів та модулів
Хороша альтернатива конкурентам, таким як ASP (Active Server Pages) від Microsoft	Можливість написання зовнішніх бібліотек на Ruby або C	Об'єктно-орієнтований підхід до програмування
Працює практично на будь-якій операційній системі чи платформі	Можливість написання багато-поточкових програм з простим API	

Таблиця 3.2 – PHP, Python та Ruby: недоліки

PHP	Ruby	Python
Не підходить для розробки настільних програм	Можливі труднощі у вивченні	Недостатньо ефективна робота з багатоядерними та багатопроцесорними обчислювальними системами
Традиційно мізерний функціонал для обробки помилок	Нестача інформаційних ресурсів	Обмежений рівень доступу до баз даних
Глобальні параметри конфігурації можуть змінювати семантику мови, ускладнюючи процеси впровадження та сумісності	Великі витрати процесорного часу (CPU time) у порівнянні з іншими мовами	Відсутність комерційної підтримки навіть для Open Source проєктів (проте ця ситуація починає змінюватися)
Звернення до об'єктів за замовчуванням здійснюється методом «виклику по значенням (CallByValue), що суперечить аналогічним операціям для більшості мов і застає багатьох програмістів зненацька	Порівняно повільна розробка оновлень	Невелика кількість розробників Python у порівнянні з іншими мовами, наприклад Java
Загалом вважається менш захищеним у порівнянні з іншими мовами програмування		За відгуками повільніший у порівнянні з мовами типу Java

Проаналізувавши всі переваги та недоліки, було обрано мову програмування PHP.

- PHP - найкраща мова для створення динамічних веб-сайтів та веб-додатків, з великою кількістю фреймворків та бібліотек, які щороку оновлюються, а тому не є складними та дешевими для програміста;

- Python - універсальна мова програмування, на якій можна запускати все: від веб-сайтів і десктопних додатків до ботів і системних сервісів;

- Ruby - мова програмування високого рівня, яка дозволяє витратити менше часу на фронтенд-компоненти та зберігання даних і зосередитися на більш практичних завданнях.

### 3.1.2. Аналіз СУБД

Системи управління базами даних стали невід'ємною частиною розробки динамічних веб-продуктів. Вони допомагають організувати всі необхідні файли. Все це необхідно для швидкого доступу та оптимізації додатку або веб-сайту. Для аналізу ми обрали такі популярні СУБД як: MariaDB [13], MySQL та PostgreSQL.

СУБД - це таблиця, і ця система може мати різну структуру. Але дані в таблиці можуть бути різних типів. Деякі СКБД підтримують не так багато типів, деякі навіть вводять нові типи для обчислень. До них відносяться логічні дані, рядки, дані з плаваючою комою та багато інших. І всі ці дані пов'язані між собою відповідно до реляційної моделі.

Однак, щоб записати дані в рядок, потрібно вказати тип даних. Дуже схожу процедуру можна виконати з комірками в Excel та інших подібних програмах. На сайтах і в додатках комірки містять дані про користувачів ресурсу, сам контент, товари і так далі. Призначення бази даних у цьому випадку - надавати цю інформацію одразу на запит користувача: серверний скрипт в обмін на клієнтський скрипт.

### 3.1.3. Переваги та недоліки СУБД

Якщо порівняти файлову систему і сайт, створений на її основі, то можна помітити, що система управління базами даних працює більш плавно і ефективно. Серед недоліків сучасних систем управління базами даних, які часто обговорюються, можна виділити наступні:

- Їх нелегко вивчити. Щоб зрозуміти принципи роботи MySQL, потрібно розбиратися в базах даних.

- Вартість. Сама система управління базами даних може бути дуже дорогою, але так само як і обслуговування бази даних, стороннє програмне забезпечення та інші витрати. Навіть розміщення великої бази даних на хорошому хостингу коштує дорого.

- Вага. Просто файли в робочій програмі важать багато. Якщо перетворити їх у базу даних, обсяг різко зростає, адже файли тепер мають певні функції, логічно пов'язані один з одним і викликаються скриптами. Все це коштує не тільки грошей, але й пам'яті сервера. Особливо це актуально, коли сервером є домашній комп'ютер розробника.

- Централізований хостинг. Лише в останні роки програмісти почали використовувати розподілений реєстр для зберігання файлів. Якщо файли знаходяться в одній базі даних, вони є вразливими.

Хороша новина полягає в тому, що більшість цих недоліків маскуються перевагами СУБД. Це відкриває багато можливостей, яких немає у файлової системи. Перш за все, це економить місце на диску. Хоча сама СУБД займає певний обсяг пам'яті, вона дозволяє уникнути накопичення непотрібної інформації. Немає надмірності або дублікатів файлів. У той же час, вона зберігає ту інформацію, яку потрібно зберігати в надлишку.

Важливо, що СУБД уникає дублювання істин. Система самостійно відбирає елементи, які надають схожу інформацію, і стежить за тим, щоб вони не суперечили один одному. В результаті продукт отримує набагато більше інформації за той самий обсяг даних.

Системи управління базами даних відіграють важливу роль у спільній розробці. Отримати груповий доступ до дерева файлів досить складно, навіть якщо вжити всіх запобіжних заходів. Однак можна надати авторизований доступ до СУБД обмеженій кількості людей без шкоди для системи безпеки.

На сьогоднішній день не існує кращого рішення для зберігання даних, ніж сучасна СУБД. Використання бази даних дозволяє визначити необхідні заходи безпеки. Крім того, щодня з'являються нові засоби захисту баз даних. Зазвичай доступ відбувається шляхом якогось запилення, але при достатніх знаннях можна реалізувати все – від антропометрії до двофакторної автентифікації. Особливо це стосується СУБД з відкритим вихідним кодом, таких як MariaDB.

PostgreSQL – ще одна система управління базами даних, яка є не реляційною, а об'єктно-реляційною. Це означає, що користувач може створювати

об'єкти для операцій, які можуть містити різні дані. Вона повністю безкоштовна і найбільш гнучка. Деякі розробники вважають PostgreSQL найбільш професійним рішенням на ринку. Ця СУБД бере свій початок з некомерційного університетського проекту в Берклі під назвою Postgres. Ця система розроблялася протягом восьми років і підтримується досі.

Як це часто буває з подібними продуктами, вона була розроблена "програмістами для програмістів": неймовірно функціональна, але занадто складна, щоб її міг освоїти пересічний програміст. Спочатку СУБД навіть мала власну мову запитів, але згодом від цієї ідеї відмовилися, залишивши лише тривіальний "набір". Незважаючи на ентузіазм незалежних програмістів, PostgreSQL не настільки хороша, як про неї думають. У вихідному коді все ще є проблеми.

З точки зору масштабованості, PostgreSQL немає чого заздрити MySQL і MariaDB. Великі проекти з великими масивами даних будуються на цьому програмному забезпеченні, тому що розробники довіряють його стабільності. Кілька варіантів інтерфейсу дозволяють кастомізувати продукт.

Але PostgreSQL все ще далека від того, щоб стати масовим продуктом. Справа в тому, що ця система занадто складна для простого розробника. Навіть звернувшись до документації, не завжди можна отримати відповіді на найбільш логічні питання. Крім того, швидкість виконання запитів в PostgreSQL викликає подив, а ця СУБД чудово підходить для корпоративних рішень. Наприклад, база даних для IT-компанії, де кожен розробник може її підтримувати. Тим більше, що PostgreSQL повністю безкоштовна.

Проаналізувавши ці СУБД, ми дійшли висновку, що MySQL є найпоширенішою, а MariaDB та PostgreSQL - найбільш масштабованими. Після обговорення з контролюючим органом було вирішено використовувати MySQL.

#### 3.1.4. Фреймворки

Задіяні два фреймворки: Yii та Bootstrap. Yii - це продуктивний, компонентний PHP-фреймворк, що спеціалізується на розробці сучасних

веб-додатків. Завдяки своїй компонентній структурі та відмінній підтримці кешування, цей фреймворк особливо добре підходить для розробки серйозних проектів, таких як портали, форуми, CMS і комерційні платформи.

Для організації програмного коду Yii використовує архітектурну модель Model-View-Controller (MVC). Yii керується філософією легкого та елегантного програмного коду, не ускладнюючи дизайн дотриманням певних стандартів дизайну. Yii має високу розширюваність. Можна адаптувати або модифікувати майже всі частини рідного програмного коду [11].

Завдяки архітектурі розширення легко ділитися кодом або використовувати код спільноти. Він підтримується та розвивається висококваліфікованою командою та великою спільнотою програмістів. Розробники фреймворку слідкують за тенденціями у веб-розробці та появою інших проектів. Найкращі функції та найкращі практики постійно впроваджуються у фреймворк у вигляді регулярних та стильних інтерфейсів. Bootstrap - найпопулярніший фреймворк для розробки адаптивних і мобільних веб-проектів. Bootstrap використовує новітні технології CSS та HTML. Фреймворк дозволяє використовувати попередньо визначені класи з 25 стилями для створення привабливого веб-сайту для різних роздільних здатностей екрану.

### 3.1.5. Телеграм Bot API

Для вибору відповідної бібліотеки для розробки було проведено дослідження бібліотек. Після огляду бібліотек для Telegram було вирішено використовувати бібліотеку Telegram Bot API. Перевагою було названо більш проактивну підтримку з боку розробників та велику кількість користувачів, що дозволяє швидко знаходити рішення для проблем, які виникають. Бібліотека Telegram Bot API обробляє всі без винятку аспекти надсилання та отримання запитів, дозволяючи зосередитися безпосередньо на логіці. Інтегрувати бібліотеку дуже просто, достатньо виконати команду в консолі:

```
php composer.phar requires telegram-bot/api
```

Bot API - це HTTP-інтерфейс для роботи з ботами в Telegram. Кожен бот - це спеціальний акаунт, який створюється для автоматичної обробки та відправки повідомлень. Існує два логічно протилежних способи отримання оновлень від бота:

- Long Pull - додаток автоматично опитує сервер Telegram, щоб отримати оновлення для бота. За замовчуванням встановлено 100 мс;
- Webhook - сервери Telegram повідомляють додаток на сервері, як тільки з'являється оновлення [12].

Отримані оновлення зберігаються на сервері до моменту їх обробки, але не довше 24 годин. Незалежно від способу отримання оновлень, у відповідь надсилається послідовний об'єкт оновлення у форматі JSON.

Всі запити до API Telegram-бота повинні здійснюватися за протоколом HTTPS в наступному вигляді:

`https://api.telegram.org/bot<token>/METHOD_NAME`

Принцип взаємодії між чат-ботом і користувачем проілюстровано на рисунку 3.1.

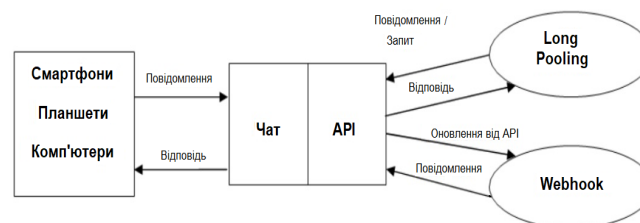


Рисунок 3.1 – Принцип роботи чат-бота на платформі Telegram

Щоб отримати токен, необхідно написати спеціальному боту @BotFather. Приклади методів, доступних в API, описані нижче:

- getUpdates – цей метод використовується для отримання оновлень за допомогою технології Long Polling;
- setWebhook – цей метод прив'язує URL домену, на якому знаходиться запусканий бот;
- sendMessage – метод надсилає текстове повідомлення в Telegram-клієнт;
- sendLocation – метод надсилає повідомлення з контактною інформацією до Telegram-клієнта;
- getFile – повертає завантажений файл на основі його назви тощо.

Допускаються POST і GET запити. Існує 4 можливості передачі параметрів до API бота:

- URL-запит;
- application/x-www-form-urlencoded;
- application/json (завантаження файлів заборонено);
- багатокomпонентні дані/формат (для завантаження файлів).

### 3.1.6. Вибір хостингу

Веб-сайти стали незамінними для брендингу та поширення інформації. Початківцю дуже важко створити повноцінний веб-сайт з обмеженим бюджетом. Тому тим, хто хоче створити сайт, потрібні дорогі послуги з веб-дизайну. Студенти, які навчаються або готуються до іспитів, не витратять стільки грошей. Тут на допомогу приходять найкращі провайдери безкоштовного хостингу. У цьому блозі ви знайдете всю основну інформацію про веб-хостинг та найкращі безкоштовні рішення для студентів. В кінці блогу ви отримаєте всю інформацію, необхідну для прийняття обґрунтованого рішення при купівлі послуги веб-хостингу. Цей блог призначений для студентів, які шукають безкоштовний хостинг для студентів. Читайте далі, щоб переконатися, що ви знайдете своє доменне ім'я та 1 рік безкоштовного хостингу.

Веб-хостинг – це домівка для вашого веб-сайту. Якщо у вас є веб-сайт, вам потрібно місце для його розміщення, і хостинг – це саме те місце. Хостинг-провайдер надає місце для вашого веб-сайту на веб-сервері. Веб-хостинг - це послуга, яка надає місце для вашого веб-сайту на сервері. Коли ви завантажуєте свій веб-сайт на цей сервер, він стає доступним для користувачів Інтернету. Користувачі вводять URL-адресу або доменне ім'я вашого веб-сайту у веб-браузер. Комп'ютер підключається до сервера і відображає всі веб-сторінки.

Сервери створюють серію файлів, що містять веб-сторінки та їхній код, зображення, вміст, графіку тощо. Кожна веб-сторінка, яку відвідує користувач,

розміщується на певному веб-сервері. Цей хостинг-сервер можна орендувати або придбати для розміщення веб-сторінки в Інтернеті.

Безкоштовний хостинг вважається найбезпечнішим способом розпочати веб-проект швидко і безкоштовно.

Багато невеликих і неприбуткових організацій використовують безкоштовний хостинг для створення простих веб-сайтів, щоб публікувати новини та анонси подій або спілкуватися з відвідувачами.

Безкоштовний хостинг дуже корисний для початківців, студентів та дослідників. Це чудовий спосіб творити, вчитися і розвиватися по максимуму. Тут ви можете створити свій власний веб-сайт.

Ви можете використовувати всі функції безкоштовно і робити все, що забажаєте. Безкоштовні веб-сайти корисні для студентів, тому що вони допомагають їм вивчити основи створення, побудови та підтримки веб-сайтів. Ці сайти дуже надійні (див. рисунки 3.2-3.7).

Хостинг	Рік	Сервери	Панель управління	Тест	Uptime	Оцінка	
<a href="#">Hostiq.ua</a>	2013	Україна		cPanel	30	99.99	4.8
<a href="#">Hostpro.ua</a>	2005	Україна		cPanel, ISPManager	5	99.99	4.7
<a href="#">S-host.com.ua</a>	2015	Україна		cPanel, ISPmanager, BrainyCP	7	99.99	4.6
<a href="#">Hostlife.net</a>	2005	Україна		cPanel, DirectAdmin	7	99.93	4.5
<a href="#">Ukraine.com.ua</a>	2006	Україна		Власна	5	99.98	4.5
<a href="#">Hyperhost.ua</a>	2009	Україна		ISPmanager, cPanel	7	99.97	4.4
<a href="#">Nebx.com.ua</a>	2018	Україна		ISPManager, cPanel, BrainyCP	7	99.93	4.4
<a href="#">Cityhost.ua</a>	2004	Україна		Власна, ISPmanager	10	99.99	4.4
<a href="#">Besthosting.ua</a>	2003	Україна		DirectAdmin, cPanel	7	99.98	4.3
<a href="#">Vps.ua</a>	2010	Україна		cPanel, DirectAdmin, ISPmanager	30		4.3
<a href="#">Wishhost.net</a>	2007	Україна		ISPmanager, VMmanager, cPanel и др.	30	99.99	4.3
<a href="#">Freehost.com.ua</a>	2003	Україна		Власна, ISPmanager, VestaCP	7		4.2
<a href="#">Svai.net</a>	2004	Україна		DirectAdmin			4.2
<a href="#">Ukrnames.com</a>	2011	Україна		cPanel, ISPmanager	5		4.2
<a href="#">Ohosting.me</a>	2013	Україна		DirectAdmin			4.2
<a href="#">Cloverhost.net</a>	2008	Україна		DirectAdmin		99.99	4.2
<a href="#">Prostohosting.com</a>	2004	Україна		DirectAdmin		100	4.2
<a href="#">Prohoster.info</a>	2012	Україна		ISPManager	14	99.99	4.2
<a href="#">Avahost.ua</a>	2008	Україна		cPanel			4.2
<a href="#">Link-host.net</a>	2009	Україна		ISPManager	30		4.2
<a href="#">Parkovka.ua</a>	2010	Україна		ISPmanager	7	99.94	4.2
<a href="#">Slimhost.com.ua</a>	2013	Україна		ISPManager	7		4.1

Рисунок 3.2 – Найкращі хостинги України (2022)

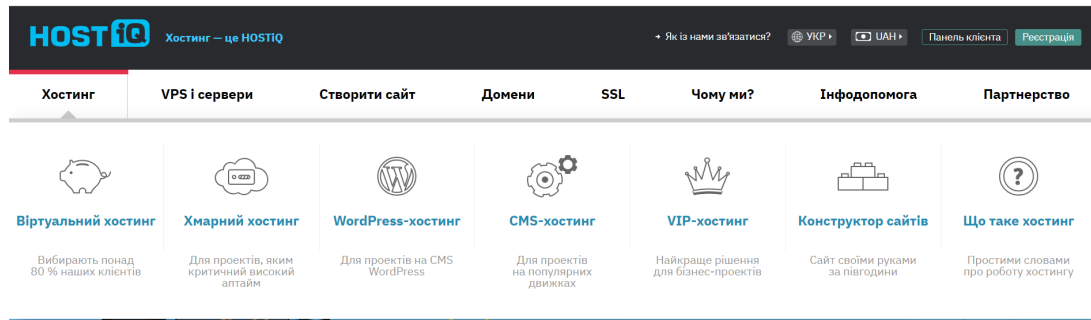


Рисунок 3.3 – Хостинг HOSTiQ

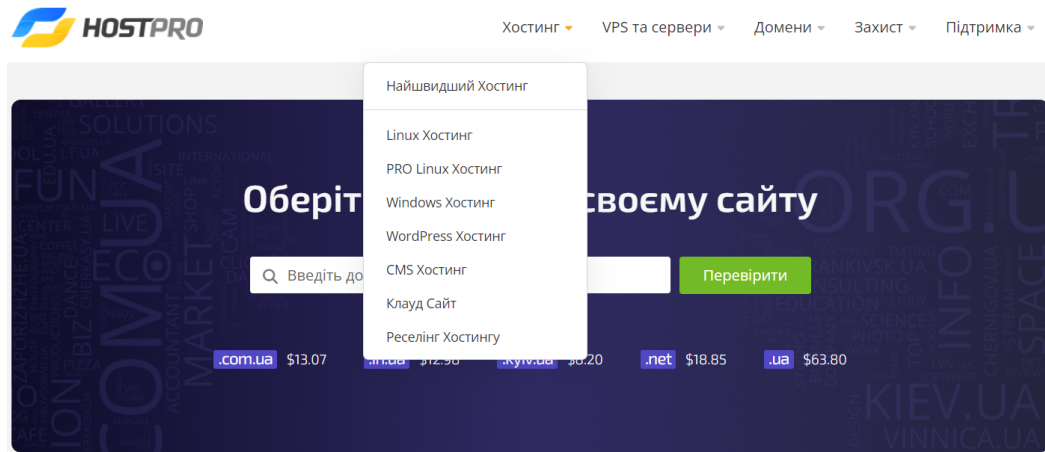


Рисунок 3.4 – Найкращий хостинг від Hostpro



Рисунок 3.5 – Хостинг Україна

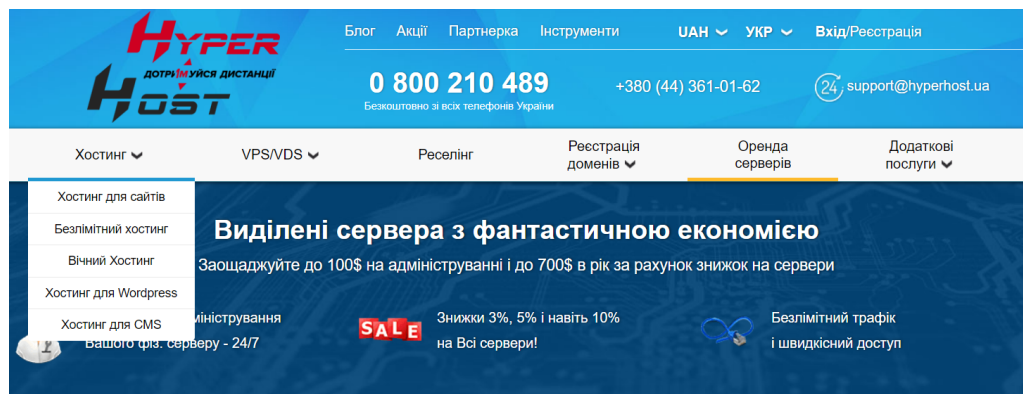


Рисунок 3.6 – Хостинг Hyper Host

**PRO-тарифи**

В нас також є [VPS](#) - [потужні VPS](#) із [виділеними накопичувачами](#) - [недорогі виділені сервери](#)

Назва тарифу	ПРО-СТАРТ	ПРО-МАКС	В ПОРІВНЯННІ ЗІ ЗВИЧАЙНИМ ХОСТИНГОМ
Ціна тарифу	від <b>417 €</b> на місяць у разі оплати за рік	від <b>1008 €</b> на місяць у разі оплати за рік	
МАКСИМАЛЬНА ЕКОНОМІЯ ПРИ ЗАМОВЛЕННІ ВІД 24 МІС			
Гарантовані ресурси ЦПУ	2 потоки процесора e5-2650v2 (або краще)	4 потоки процесора e5-2650v2 (або краще)	немає гарантій
Гарантовані ресурси оперативної пам'яті	10GB RAM	20GB RAM	немає гарантій

Рисунок 3.7 – Хостинг Netx

### 3.1.7. Середовище розробки

PhpStorm – це інтегроване середовище розробки PHP з просунутим редактором, який повністю розуміє ваш код, підтримує PHP 5.3-8.2.3 як для сучасних, так і для застарілих проектів, і пропонує найсучасніші функції завершення коду, рефакторингу, налагодження та змішування мов.

Ви можете перевірити свій код за допомогою сотень перевірок та аудитів під час розробки. PHPDoc, Code (Re)Arranger, Code Formatter з налаштованими стилями коду та інші функції допомагають розробникам написати простий і зручний для підтримки код.

PhpStorm підтримує сучасні технології веб-розробки, такі як HTML5, CSS, Sass, SCSS, Less, Stylus, Compass, CoffeeScript, TypeScript, ECMAScript Harmony, Jade, Zen Coding, Emmet Templates і, звичайно, JavaScript.

PhpStorm має всі можливості WebStorm (редактор HTML/CSS, редактор JavaScript) і пропонує повну підтримку PHP і баз даних/SQL.

### 3.1.8. Основні можливості

- Інтелектуальний редактор PHP коду з підсвічуванням синтаксису, компіляцією коду, розширеним форматуванням коду та налагодженням.
- Підтримує PHP 5.3-8.2.3, генератори, інструменти та всі версії розробки синтаксису
  - Рефакторинг PHP, управління кодом, виявлення дублікатів коду.
  - Docker, Composer, інтегрований REST-клієнт, інструменти командного рядка, підтримка SSH-консолі.
  - Підтримка фреймворків (MVC view для Symfony2, Yii) та кастомні розширення для основних PHP-фреймворків (Laravel, Symfony, Magento, Drupal, Yii, CakePHP, WordPress, Joomla! та багато інших).
  - Візуальна налагодження PHP, обробка помилок конфігурації, покриття коду за допомогою PHPUnit і Codeception (з підтримкою PHPUnit 6), інтеграція профілів.
  - Редактор для HTML, CSS та JavaScript. Налаштування та модульне тестування для JS. Підтримка HTML5, CSS, Sass, SCSS, Less, Stylus, Compass, CoffeeScript, TypeScript, ECMAScript Harmony, Emmet та інших сучасних технологій веб-розробки.
  - Повний набір інструментів для розробки інтерфейсів.
  - Підтримка стилів кодування з вбудованими стилями для PSR1/PSR2, Laravel, Symfony, Zend, Drupal та інших фреймворків.
  - Інтеграція з системами контролю версій через простий інтерфейс.
  - Віддалене розгортання додатків та автоматична синхронізація через FTP, SFTP, FTPS тощо.
  - Редагування в режимі реального часу: зміни коду відразу відображаються в браузері, без перезавантаження сторінки.
  - PHP UML
  - Інтеграція з інструментами повідомлення про помилки.
  - Інструменти для роботи з базами даних, редактор SQL.
  - Крос-платформна сумісність (Windows, Mac OS X, Linux).

## 3.2 Серверна частина програмного продукту

Для розробки серверної частини програмного продукту було використано мову PHP. Фреймворк Yii2 використовувався як платформа для визначення структури програмної системи. Для зберігання даних використовувалася об'єктно-орієнтована база даних MySQL.

### 3.2.1. Створення необхідних таблиць у MySQL

Для створення таблиць необхідно встановити MySQL. Вся інформація про встановлення MySQL в Windows взята з офіційної інсталяційної документації.

Після встановлення MySQL можна приступати до створення таблиць для додаткового зберігання даних (див. рисунок 3.8).

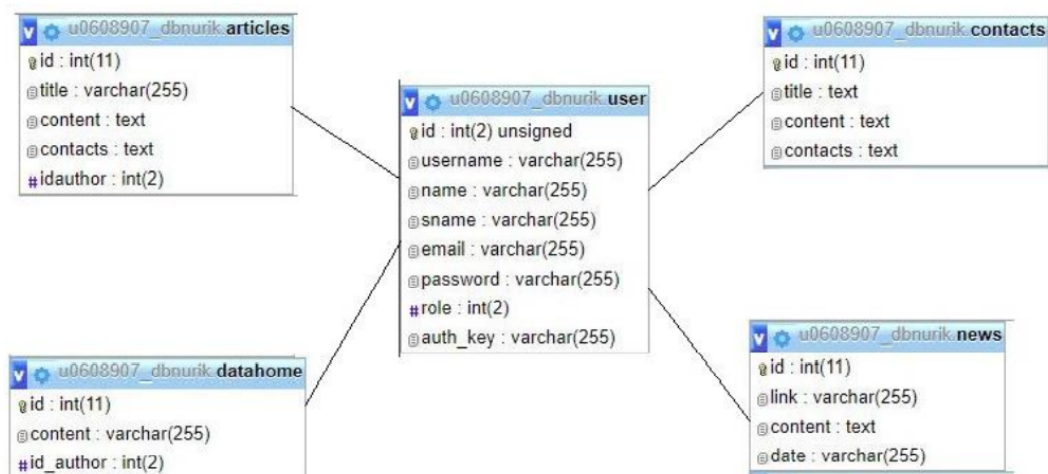


Рисунок 3.8 – Модель бази даних

### 3.2.2. Реєстрація чат-бота Телеграм

Першим кроком у розробці додатку є реєстрація в конкретному чат-боті BotFather. Реєстрація ініціюється командою "/newbot", після чого з'являється запит на введення імені чат-бота з умовою, яку необхідно виконати: Після імені

обов'язково має стояти "Bot" або "\_bot". Якщо всі умови дотримано, BotFather видає токен (спеціальний рядок для доступу до HTTP API Telegram Bot) та URL-адресу для доступу до чат-бота. Приклад реєстрації чат-бота показано на рисунку 3.9.

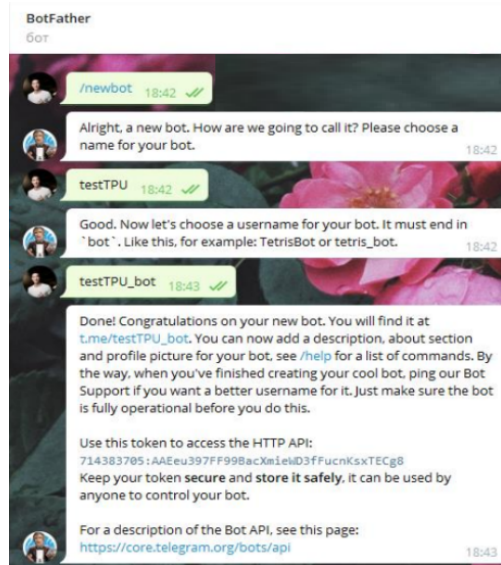


Рисунок 3.9 – Приклад реєстрації чат-бота

Наступні команди доступні для налаштування додаткових параметрів, таких як іконка чат-бота, привітальне повідомлення, опис чат-бота, а також для видалення існуючих чат-ботів (див. таблицю 3.3).

Таблиця 3.3 – Доступні команди для зміни чат-ботів

Команда	Опис
/setname	Змінює існуюче ім'я
/setdescription	Надає текст, який буде відображатися при першому відкритті чат-бота
/setabouttext	Надає текст у полі «Про чат-бот»
/setuserpic	Надає обрану картинку
/setcommands	Дозволяє створити список доступних команд
/deletebot	Видаляє обраний чат-бот

Окрім команд для зміни основних параметрів чат-бота, є кілька команд для відображення незмінених параметрів (токенів) і навіть для присвоєння значень, які у таблиці 3.4.

Таблиця 3.4 – Доступні команди для додаткового налаштування чат-бота

Команда	Опис
/token	Повертає отриманий раніше токену вибраного бота
/revoke	Анулює токен доступу до боту
/setinline	Включає або вимикає можливість викликати робот з інших чатів
/setinlinegeo	Включає або вимикає можливість передавати місцезнаходження бота з іншого чату
/setinlinefeedback	Дозволяє отримувати інформацію про кількість обраних користувачами команд
/setjoingroup	Визначає, чи може бути доданий в групові діалоги бот
/setprivacy	Вмикає режим конфіденційності. У цьому режимі бот отримує, обробляє та надсилає назад інформацію окремо для кожного користувача в чаті

Після того, як ви зробили налаштування на сайті Telegram та отримали токен, можна приступати до розробки програмної частини чат-бота.

Токен - це мітка, яка зберігає всі дані, необхідні для аутентифікації та авторизації, у зашифрованому вигляді. Після отримання токена необхідно перевірити його, перейшовши за наступним посиланням в адресному рядку браузера:

<https://api.telegram.org/714383705:AAEu397FF99VasXmieWD3fFucnKsxTECg8/getMe>

Якщо перевірка пройшла успішно, в браузері з'явиться наступний текст:

```
{"ok":true,"result":{"id":264543618,"first_name": "u0410\u0431\u0438\u0438\u0442u0443u0440\u0438\u0435\u043du0442\u041fu0413u0423u0422\u0418","username":"testTPU_bot"}}
```

Після цього можна переходити в область розробки і починати писати функції чат-бота. Але спочатку потрібно визначити зображення профілю чат-бота та опис, який буде показуватися користувачам (див. рисунки 3.10 та 3.11).

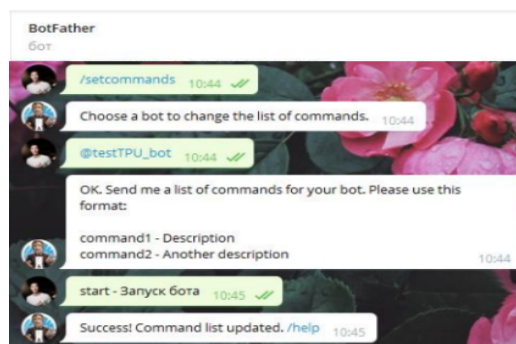


Рисунок 3.10 – Виконання команди «/setcommands»

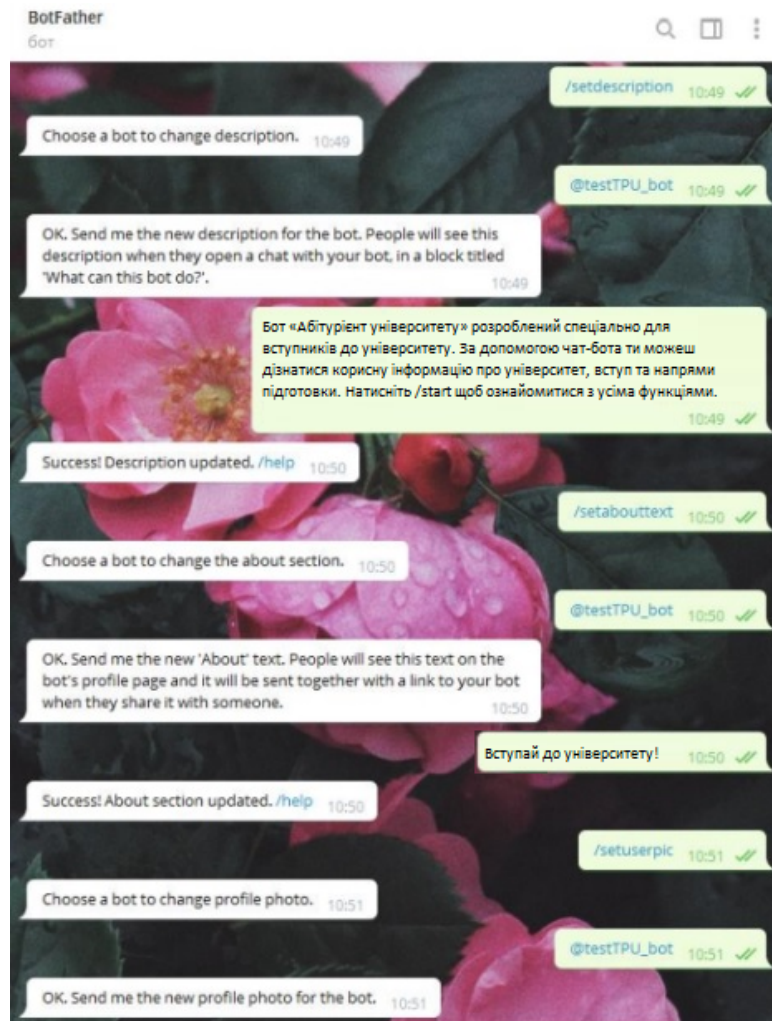


Рисунок 3.11 – Виконання команд «/setdescription, /setabouttext, /setuserpic»

### 3.2.3. Реалізація клієнтської частини чат-бота

Користувачі Telegram можуть взаємодіяти з чат-ботами двома способами: командами ("/start", "/help") з параметрами або вбудованою клавіатурою. Для зручності користувачів було вирішено створити інтерфейс з інтегрованою клавіатурою.

Для досягнення поставлених цілей потрібно п'ять різних меню:

- меню вибору школи;
- головне меню;
- меню для відображення повідомлень;
- меню для відображення розділу "Як отримати доступ до NURE";
- меню для відображення контактів NURE.

У месенджері Telegram кожна клавіатура реалізована як об'єкт, а її клавіші – як JSON-рядок. У документації Telegram Bot API зазначено, що кожна клавіатура повинна мати один обов'язковий параметр – назву кнопки (text) – і шість необов'язкових параметрів – посилання (url), дані зворотного дзвінка (callback\_data), можливість зробити вбудований запит (switch\_inline\_query), можливість відобразити клавіатуру (switch\_inline\_query\_current\_chat), зворотний дзвінок з описом поточної гри (callback\_game) і кнопку з можливістю оплатити.

На рисунку 3.12 показано приклад клавіатурної реалізації одного з меню чат-бота.

```

if (mb_stripos($mtext, "Меню") !== false) {
    $keyboard = new \TelegramBot\Api\Types\ReplyKeyboardMarkup([
        [{"text" => "Школи"}, {"text" => "Новини"}], [
            {"text" => "Як вступити до університету?"}, {"text" => "Контакти"}]], oneTimeKeyboard: false, resizeKeyboard: true);
    // $bot->sendMessage($message->getChat()->getId(), $result);
    $bot->sendMessage($message->getChat()->getId(), "Меню", false, null, null, $keyboard);
}

```

Рисунок 3.12 – Реалізована клавіатура головного меню

Під час першого запуску чат-бота запускається скрипт, який збирає персональні дані користувача, надані під час реєстрації в месенджері Telegram, і зберігає їх у базі даних. Після запуску скрипту на екрані користувача з'являється кнопка з проханням відповісти на кілька запитань, щоб заповнити свої дані, такі як ім'я та адреса електронної пошти (див. рисунок 3.13). Ці дані потрібні для того, щоб у майбутньому надсилати певну інформацію на особисту електронну адресу користувача.

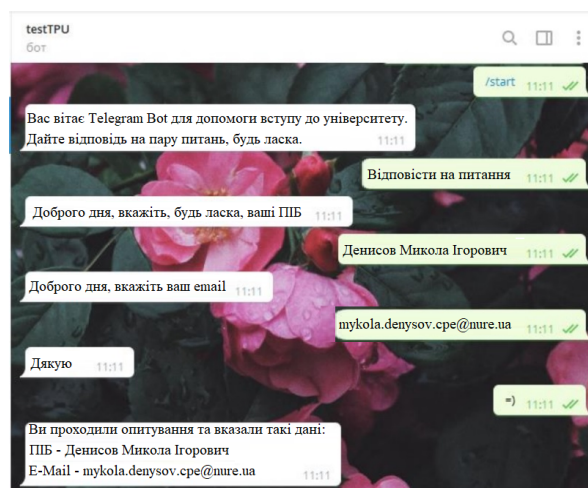


Рисунок 3.13 – Перший запуск бота та проходження опитування

Наступним завданням була реалізація головного меню. Це мало дозволити користувачеві обрати тип інформації, яку він хотів би отримати. Запропонована інформація включала відображення останніх новин, відображення контактів ХНУРЕ та можливість доступу до інших меню для отримання більш конкретної інформації (див. рисунок 3.14).



Рисунок 3.14 – Основне меню чат-бота

Далі було реалізовано одне з основних завдань, а саме відображення переліку предметів та навчальних програм (факультетів). Реалізоване меню показано на рисунку 3.15.

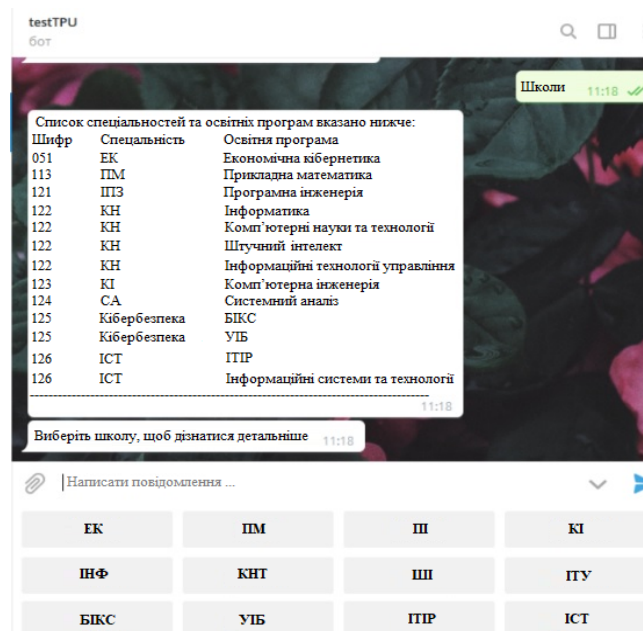


Рисунок 3.15 – Меню виведення списку спеціальностей та освітніх програм

Друга функція, яка буде реалізована, – це меню з низкою корисної інформації, яка допоможе при вступі до університету і, в залежності від вибору, така інформація, як:

- олімпіади – відображає інформацію про олімпіади, організовані для вступу на бюджетні місця;
- приймальна комісія – відображає інформацію про графік роботи приймальної комісії ЄКГ;
- документи для абітурієнтів – відображає інформацію про документи, які необхідно зібрати для вступу до університету.

Результат реалізації наведено на рисунку 3.16.

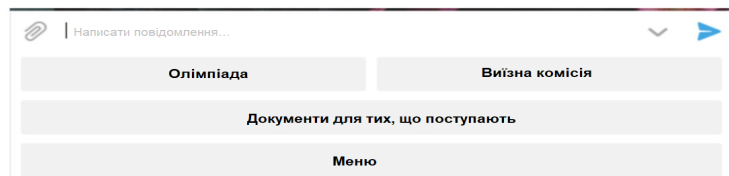


Рисунок 3.16 – Меню вибору розділу «Як вступити до ХНУРЕ?»

Третя функція полягає в ознайомленні з останніми новинами університету. Останнім кроком у реалізації клієнтського чат-бота було створення меню для відображення контактної інформації університету.

Таким чином, клієнтська частина чат-бота була реалізована для допомоги абітурієнтам при вступі до університету.

#### 3.2.4. Реалізація серверної частини чат-бота

Серверна частина чат-бота на платформі Telegram має вирішувати такі завдання:

- Отримувати та розпізнавати повідомлення, надіслані користувачем;
- Отримувати та порівнювати отриману інформацію для створення коректної відповіді;
- Витягувати інформацію з бази даних, виконуючи запити з отриманими параметрами;
- Складання та відправлення текстової відповіді на основі наданої користувачем інформації.

Щоб чат-бот міг розпізнавати повідомлення користувача, кожна кнопка на онлайн-клавіатурі має так звані дані для зворотного дзвінка. Коли користувач натискає кнопку, сервер порівнює відповідь зі значенням або набором значень.

В якості першого кроку ми реалізували функцію перегляду університетських адрес. Другою реалізованою функцією чат-бота є відображення інформації про обране користувачем меню. Як приклад для опису цієї функції візьмемо меню випускної комісії.

Для того, щоб отримати інформацію про випускну комісію, потрібно просто зайти в цей розділ, натиснувши на кнопку "Випускна комісія". У цьому розділі відображається інформація про те, де приймають документи в ХНУРЕ, а щоб отримати розклад, ви можете скористатися двома кнопками під цим повідомленням. Кнопка "Отримати розклад" повертає користувача на екран з PDF-документами, а кнопка "Надіслати поштою" надсилає цей документ користувачеві на електронну адресу, яку він ввів в опитуванні при першому запуску бота. Ця функція реалізована для того, щоб користувач не втратив цей документ у листуванні з ботом або міг надіслати його зі своєї пошти іншим людям, які цікавляться цим університетом.

Таким чином, у Telegram-платформі було реалізовано наступні функції чат-бота:

- навігація між адресами ХНУРЕ (факультетів);
- навігація в розділі "Як доїхати до ХНУРЕ";
- відображення останніх новин університету;
- пошук контактних даних університету.

### 3.3 Адміністративна панель чат бота

Модуль - це окрема частина програмного коду, фрагмент програми, який виконує свою задачу повністю самостійно.

Модуль розробляється та реалізується за моделлю MVC (Model-View-Controller). Концепція MVC (Model-View-Controller) останніми

роками багато обговорюється у світі веб-програмування.

MVC - це шаблон проектування, який визначає, як повинна бути побудована структура додатку, зони відповідальності та взаємодію різних елементів цієї структури.

Перше, що бачить адміністратор додатку при вході в адміністративну панель, – це сторінка авторизації.

Адміністратор програми має необмежені права на створення, читання, модифікацію та видалення даних.

### 3.3.1 Розробка компонентою структури веб-програми

Реалізація алгоритму агрегації повідомлень:

```
public function actionCreate()
{
    $model = new News();
    if ($model->load(Yii::$app->request->post()) && $model->save()) {
        return $this->redirect(['view', 'id' => $model->id]);
    }
    return $this->render('create', [
        'model' => $model,
    ]);
}
```

Реалізація алгоритму виведення повідомлень:

```
public function actionIndex()
{
    $searchModel = new NewsSearch();
    $dataProvider =
    $searchModel->search(Yii::$app->request->queryParams);
    return $this->render('index', [
        'searchModel' => $searchModel,
        'dataProvider' => $dataProvider,
    ]);
}
```

Результат виконання запиту на запис новини представлено на рисунку 3.17.

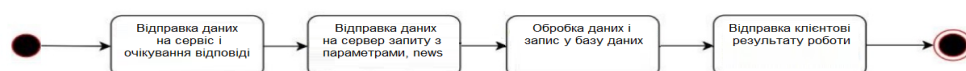


Рисунок 3.17 – Виконання запиту на запис новини

Наведений на рисунку 3.17 результат відповідає виконанню спрощеного запиту на запис новини.

## ВИСНОВКИ

Обмін повідомленнями стає дедалі популярнішим засобом комунікації. Компанії, сім'ї та друзі щодня користуються текстовими повідомленнями та мультимедійним контентом. Також варто згадати зростаючу популярність програмних продуктів на основі платформ обміну повідомленнями, таких як чат-боти. Цілодобове обслуговування клієнтів, конвертація документів і медіафайлів, замовлення таксі, пошук інформації та багато іншого тепер може здійснюватися за допомогою єдиного сервісу обміну повідомленнями. Користувачам більше не потрібно завантажувати кілька додатків для дуже специфічних завдань, оскільки їм потрібен лише один додаток для обміну повідомленнями та необхідні додатки для чату, які не займають місця на смартфоні.

У кваліфікаційній роботі магістра були виконані поставлені завдання. По-перше, було розглянуто системи обміну повідомленнями. Було порівняно та проаналізовано переваги та недоліки, а також обрано Telegram Messenger як найбільш зручний та доступний месенджер з урахуванням документації API Telegram Bot. По-друге, ми дослідили існуючі аналогові месенджери в місті Харкові та визначили їхні переваги та недоліки, а також цікаві рішення. На основі цього ми визначили вимоги до розробки кастомного чат-бота для допомоги абітурієнтам при подачі заявок на навчання в ХНУРЕ. В останньому завданні ми обрали технології та середовища для розробки чат-бота, серед яких PHP, MySQL, Yii2, Bootstrap та PhpStorm. Рішення розробити чат-бота було продиктоване бажанням оптимізувати роботу працівників приймальної комісії та абітурієнтів, які витрачають багато часу, покладаючись на інформацію з відкритих джерел.

Обмін повідомленнями досі залишався невикористаною платформою, але ця кваліфікаційна робота усунула та заповнила цю прогалину. Основними перевагами ботів над іншими формами взаємодії є зручність, надійність та доступність. На відміну від поглинаючих агентів, чат-бот не бере вихідних, не схильний до людської помилки і захищений спеціальним алгоритмом. Функціонал месенджерів, особливо Telegram, в першу чергу орієнтований на мобільну аудиторію, яка активно використовує смартфони для роботи та розваг.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Gartner Inc. // URL: <http://www.gartner.com/newsroom/id/3215217/> (дата звернення: 25.03.23).
2. Digital в Україні 2022: тренди українського інтернету та вплив війни // URL: <https://marketer.ua/ru/digital-in-ukraine-2022-ukrainian-internet-trends-and-the-impact-of-the-war/> (дата звернення: 25.03.23).
3. Офіційна сторінка Абітурієнт ХНУРЕ 2022 – Telegram // URL: [https://dec.nure.ua/wptelegram/widget/view/@nure\\_osvita/?q=%23%D0%A7%D0%BE%D0%BC%D1%83%D1%81%D0%B0%D0%BC%D0%B5](https://dec.nure.ua/wptelegram/widget/view/@nure_osvita/?q=%23%D0%A7%D0%BE%D0%BC%D1%83%D1%81%D0%B0%D0%BC%D0%B5) (дата звернення: 25.03.23).
4. Міносвіти запустило мобільний додаток для абітурієнтів // URL: [https://lb.ua/society/2017/07/23/372179\\_minobrazovaniya\\_zapustilo\\_mobilnoe.html](https://lb.ua/society/2017/07/23/372179_minobrazovaniya_zapustilo_mobilnoe.html) (дата звернення: 25.03.23).
5. Марголін О. UML для бізнес-моделювання: для чого потрібні діаграми процесів // URL: <https://evergreens.com.ua/ua/articles/uml-diagrams.html> (дата звернення: 25.03.23).
6. Використання компонентної архітектури у веб-додатках // URL: <https://fwdays.com/en/event/js-frameworks-day-2015/review/komponentnaia-arkhitektura-v-web-prilozheniiaqn> (дата звернення: 25.03.23).
7. Бондаренко С. Telegram API: наочний розбір прикладів // URL: <https://highload.today/telegram-api/> (дата звернення: 25.03.23).
8. Проектування інформаційних систем: Загальні питання теорії проектування ІС (конспект лекцій) : навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки» / КПІ ім. Ігоря Сікорського; уклад.: О. С. Коваленко, Л. М. Добровська. – Київ : КПІ ім. Сікорського, 2020. – 192 с. // URL: [https://ela.kpi.ua/bitstream/123456789/33651/1/PIS\\_KL.pdf](https://ela.kpi.ua/bitstream/123456789/33651/1/PIS_KL.pdf) (дата звернення: 25.03.23).
9. Шатаєва Т. Створення чат-бота для сайту: 7 корисних сервісів // URL: <https://helpcrunch.com/blog/uk/yak-stvoryty-chat-bota-dlia-saitu/> (дата звернення: 25.03.23).
10. Yii2 framework. Огляд екосистеми // URL: <https://habr.com/company/simpleweek/blog/254681/> (дата звернення: 25.03.23).

11. Офіційний сайт фреймворка Yii3 // URL: <https://www.yiiframework.com/> (дата звернення: 25.03.23).
12. Офіційний сайт Telegram API // URL: <https://core.telegram.org/api> (дата звернення: 25.03.23).
13. Офіційний сайт Maria DB // URL: <https://mariadb.org/> (дата звернення: 25.03.23).
14. Khovrat A., Kobziev V., Nazarov A., Yakovlev S. Parallelization of the VAR Algorithm Family to Increase the Efficiency of Forecasting Market Indicators During Social Disaster. CEUR Workshop Proceedings This, 2022, 3347, pp. 222–233.
15. Mahdi Q.A., Shyshatskyi A., Prokopenko Y., Ivakhnenko T., Kupriyenko D., Golian V., Lazuta R., Kravchenko S., Protas N., Momit A. Development of Estimation and Forecasting Method in Intelligent Decision Support Systems // Eastern-European Journal of Enterprise Technologies, 2021, 3, pp. 51–62. doi: 10.15587/1729-4061.2021.232718
16. Gordiievych, A. , Shubin, I. Forecasting of Airfare Prices Using Time Series // Information Technologies in Innovation Business Conference, ITIB 2015 - Proceedings, 2015, pp. 68-71, 7355055