

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ
ЗАСОБІВ АДМІНІСТРУВАННЯ ГРУПОВИХ ЧАТІВ
RESEARCH AND SOFTWARE IMPLEMENTATION OF
GROUP CHAT ADMINISTRATION TOOLS

Виконав(ла): студент(ка) 2 курсу, групи КНТ-123М

Спеціальності 121 Інженерія програмного

(код і найменування спеціальності)

забезпечення

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

ПАХОМОВ Р.С .

(ПРИЗВИЩЕ та ініціали)

Керівник ІЛЬЯШЕНКО М.Б.

(ПРИЗВИЩЕ та ініціали)

Рецензент КОЦУР М.І.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет КНТ
Кафедра програмних засобів
Ступінь вищої освіти магістр
Спеціальність 121 Інженерія програмного забезпечення
(код і найменування)
Освітня програма (спеціалізація) Інженерія програмного забезпечення
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН
“ ” 2024 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ПАХОМОВА Романа Сергійовича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна реалізація засобів адміністрування групових чатів. Research and Software Implementation of Group Chat Administration Tools

керівник проєкту (роботи) к.т.н., доцент, ІЛЬЯШЕНКО Матвій Борисович,
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “ 07 ” жовтня 2024 року № 398

2. Строк подання студентом проєкту (роботи) 02 грудня 2024 року

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

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Матеріали і методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

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

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	ІЛЪЯШЕНКО М.Б., доцент		
Нормоконтроль	КАМІНСЬКА Ж.К., асистент		

7. Дата видачі завдання “ 9 ” вересня 2024 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки.	2 тиждень	Розділ 2
4	Розробка структури програми.	3 тиждень	Розділ 3
5	Розробка програми.	4-7 тижні	Розділи 3,4
6	Тестування та експериментальне дослідження програмного забезпечення.	8 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	9-10 тижні	Додатки
8	Нормоконтроль та рецензування.	11 тиждень	
9	Захист роботи.	12 тиждень	

Студент(ка)

(підпис) Роман ПАХОМОВ
(Імя ПРИЗВИЩЕ)

Керівник проєкту (роботи)

(підпис) Матвій ІЛЪЯШЕНКО
(Імя ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
88 с., 3 табл., 36 рис., 3 дод., 16 джерел.

C#, MICROSOFT VISUAL STUDIO, АДМІНІСТРУВАННЯ, МОВА ПРОГРАМУВАННЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ЧАТ.

Об'єкт дослідження – процес розробки програмного забезпечення для адміністрування чатів.

Предмет дослідження – програмні засоби для підтримки процесу адміністрування групових чатів.

Мета роботи – дослідження та розробка програмного забезпечення для адміністрування групових чатів.

Матеріали, методи та технічні засоби: мова програмування C#, середовище розробки Microsoft Visual Studio.

Результати. Розроблено програмне забезпечення для адміністрування групових чатів за допомогою мови програмування C# та середовища розробки Microsoft Visual Studio.

Практична цінність роботи полягає у розробці програмного забезпечення для адміністрування групових чатів.

Висновки. Виконано проєктування програмного забезпечення для адміністрування групових чатів. Розроблено програмне забезпечення для адміністрування групових чатів. Здійснено тестування розробленого програмного забезпечення для адміністрування групових чатів.

Галузь використання – програмні засоби для спілкування користувачів.

ABSTRACT

Explanatory note to the diploma qualifying work of the master: 88 pages, 3 tables, 36 figures, 3 appendixes, 16 sources.

C#, MICROSOFT VISUAL STUDIO, ADMINISTRATION, PROGRAMMING LANGUAGE, SOFTWARE, CHAT.

The object of research is the process of developing software for the administration of chats.

The subject of the research is software for supporting the process of group chat administration.

The purpose of this work is to develop the software for the administration of group chats.

Materials, methods and technical tools: C# programming language, Microsoft Visual Studio development environment.

Results. The software tools for group chat administration using the C# programming language and the Microsoft Visual Studio development environment have been developed.

The practical value of the work lies in the development of software for the administration of group chats.

Conclusions. The software for the administration of group chats has been designed. The software for the administration of group chats has been developed. The developed software for the administration of group chats has been tested.

Scope of use – software tools for user communication.

ЗМІСТ

	С.
Перелік скорочень та умовних позначок	8
Вступ	9
1 Аналіз предметної області	11
1.1 Програмне забезпечення для групових чатів	11
1.2 Аналіз програмного забезпечення для групових чатів	21
1.3 Висновки за розділом 1	31
2 Матеріали і методи	33
2.1 Вибір мови програмування.....	33
2.2 Вибір середовища розробки для створення програмного забезпечення для адміністрування групових чатів	35
2.3 Розробка моделі взаємодії користувача з програмним забезпеченням для адміністрування групових чатів	37
2.4 Висновки за розділом 2	39
3 Опис програми	40
3.1 Структура програмного забезпечення для адміністрування групових чатів	40
3.2 Функціонування програмного забезпечення для адміністрування групових чатів	41
3.3 Розробка бази даних програмного забезпечення для адміністрування групових чатів	43
3.4 Проєктування інтерфейсу програмного забезпечення для адміністрування групових чатів	44
3.5 Висновки за розділом 3	48
4 Експлуатація, тестування та експериментальне дослідження програми	49
4.1 Призначення й умови застосування програми	49
4.2 Характеристики програми для адміністрування групових чатів	50
4.3 Інструкція по експлуатації програми	50
4.3.1 Звернення до програми.....	50

4.3.2 Вхідні й вихідні дані.....	50
4.3.3 Повідомлення.....	51
4.4 Виконання програмного забезпечення для адміністрування групових чатів	51
4.5 Тестування програмного забезпечення для адміністрування групових чатів	54
4.6 Висновки за розділом 4	55
Висновки.....	56
Перелік джерел посилання	59
Додаток А Технічне завдання.....	61
Додаток Б Фрагмент тексту програми	65
Додаток В Слайди презентації	80

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

MVC – Model-View-Controller;

БД – база даних;

ПЗ – програмне забезпечення;

ТЗ – технічне завдання;

ШІ – штучний інтелект.

ВСТУП

З переходом багатьох компаній на віддалену або гібридну модель роботи, групові чати стали основним інструментом для забезпечення безперервної комунікації та співпраці між співробітниками, незалежно від їхнього місцезнаходження. Програмне забезпечення для адміністрування чатів дозволяє командам ефективно співпрацювати, обмінюватися файлами, ідеями та інформацією в режимі реального часу [1]-[7].

Програмне забезпечення для групових чатів дозволяє створювати окремі канали або групи для різних тем, проєктів або відділів, що допомагає структурувати комунікацію та забезпечує швидкий доступ до потрібної інформації. Збереження історії повідомлень та можливість пошуку дозволяють легко знаходити попередні обговорення та документи, що є критично важливим для довготривалих проєктів. Групові чати забезпечують миттєву передачу повідомлень, що сприяє швидкому прийняттю рішень та вирішенню проблем [1]-[7].

Програмні інструменти для моніторингу та адміністрування групових чатів дозволяють забезпечити відповідність поведінки співробітників політикам компанії та запобігати некоректним діям [1]-[7].

Проте деякі програми для підтримки групових чатів не забезпечують достатній рівень шифрування, що може стати проблемою для компаній, які обробляють чутливі дані. Інше програмне забезпечення для підтримки адміністрування групових чатів може мати проблеми з інтеграцією і бути несумісним з програмами, що вже використовуються в організації. Деякі програми для адміністрування чатів можуть не забезпечувати достатній рівень функціональності.

Тому актуальною є розробка програмного забезпечення для адміністрування групових чатів. У дипломній кваліфікаційній роботі магістра розв'язується актуальне завдання дослідження та розробки програмного забезпечення для адміністрування групових чатів.

Для досягнення поставленої мети у кваліфікаційній роботі магістра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для підтримки групових чатів;
- розробити модель взаємодії користувача з програмним забезпеченням адміністрування групових чатів;
- здійснити проєктування програмного забезпечення для адміністрування групових чатів;
- створити програмне забезпечення для адміністрування групових чатів;
- виконати тестування розробленого програмного забезпечення для адміністрування групових чатів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Програмне забезпечення для групових чатів

Чати являють собою програмні інструменти, що дозволяють людям спілкуватися в режимі реального часу. Вони сприяють взаємодії між людьми, які можуть бути географічно віддаленими один від одного. Програмні засоби для спілкування за допомогою чатів значно змінилися з роками, оновивши способи взаємодії між користувачами [1]-[6].

Чати є невід'ємною частиною повсякденного життя та формують спосіб взаємодії з іншими членами суспільства [1]-[6].

Програмні засоби, що являють собою чат-платформи, складаються з віртуальних кімнат, де користувачі можуть надсилати текстові, голосові або відео повідомлення. Вони дозволяють миттєво спілкуватися, забезпечуючи безперервний обмін інформацією навіть на великих відстанях. Незалежно від того, чи це швидке текстове повідомлення, голосовий або відео чат, такі платформи пропонують різноманітні формати для задоволення різних потреб користувачів у спілкуванні [1]-[6].

Ідея чату виникла на ранніх етапах розвитку Інтернету. Спочатку текстові кімнати використовувалися для розмов у перших програмах обміну миттєвими повідомленнями. З розвитком інформаційних технологій та програмних засобів з'явилася можливість голосового та відео зв'язку, що суттєво змінило способи взаємодії користувачів [1]-[6].

Проаналізуємо класифікацію чатів. Чати можна розділити на такі основні категорії: текстовий, голосовий та відео чат [1]-[6].

Текстові чати забезпечують спілкування за допомогою текстових повідомлень. Сучасні текстові програми мають багато форм, включаючи додатки для миттєвого обміну повідомленнями та електронну пошту. Програми для обміну миттєвими повідомленнями широко використовуються для особистого та професійного спілкування. Вони дозволяють користувачам обмінюватися повідомленнями, ділитися мультимедійним контентом та

створювати групові чати [1]-[6]. Онлайн застосунки дозволяють спілкуватися безпосередньо через вебсайти. Ці функції забезпечують можливість здійснення дзвінків у реальному часі без необхідності встановлення додаткового програмного забезпечення [1]-[6].

Голосовий чат дозволяє користувачам спілкуватися за допомогою голосових команд. Технології, що використовуються для передачі голосових команд, дозволяють здійснювати голосові дзвінки через Інтернет. Програмні платформи використовують такі інформаційні технології для полегшення голосового спілкування між користувачами [1]-[6]. Програмне забезпечення для голосового зв'язку є популярним серед гравців та їх спільнот для спілкування під час багатокористувацьких ігор та соціальних зустрічей. Ці програми забезпечують голосовий зв'язок у реальному часі, сприяючи взаємодії та співпраці [1]-[6].

Програмні системи відео чату дозволяють користувачам спілкуватися віч-на-віч за допомогою камер. Існує багато видів відео чатів. Програмне забезпечення для відео дзвінків дозволяє користувачам здійснювати відео дзвінки як з окремими особами, так і з групами. Ці програмні засоби забезпечують візуальні елементи спілкування, створюючи можливість особистої взаємодії [1]-[6]. Програмні інструменти для відео конференцій призначені для проведення конференцій і нарад. Вони пропонують функції спільного використання екрана, параметри презентації та можливість запису, що полегшує співпрацю між учасниками [1]-[6].

Для кращого розуміння платформ спілкування шляхом використання програмних засобів соціальних медіа, розглянемо їх інфраструктуру, взаємодію з користувачем і процес обміну повідомленнями. Інфраструктура програмної платформи для спілкування користувачів використовує сторонні програмні сервіси для забезпечення безпечного та надійного спілкування. Основні компоненти цієї інфраструктури включають сервери, бази даних, протоколи, стандарти та засоби шифрування [1]-[6].

Сервери та центри обробки даних мають ключове значення для функціонування чатів, оскільки вони приймають і доставляють повідомлення одержувачам. Постачальники послуг керують потоком повідомлень, забезпечуючи стабільну роботу системи [1]-[6].

Протоколи та стандарти визначають правила обміну повідомленнями на платформах чату. Для забезпечення конфіденційності та безпеки спілкування за допомогою чатів використовуються шифрування та інші заходи безпеки. Багато чат-платформ застосовують наскрізне шифрування, яке гарантує, що лише відправник і одержувач можуть прочитати зміст повідомлень [1]-[6].

Зазвичай програмне забезпечення для підтримки чатів мають зручний інтерфейс, який полегшує спілкування між людьми. Більшість програмних засобів у вигляді чат-платформ вимагають створення облікового запису для доступу до їхніх послуг. Це включає вибір імені користувача, введення адреси електронної пошти або номера телефону та встановлення безпечного пароля [1]-[6].

Після реєстрації у програмному забезпеченні для підтримки чатів користувачі можуть додавати контакти або приєднуватися до групових чатів за інтересами. Це дозволяє їм спілкуватися з друзями, родиною, колегами або учасниками різних спільнот [1]-[6].

Основною функцією програмного забезпечення для підтримки чатів є спрощення обміну повідомленнями між користувачами. Розглянемо різні типи повідомлень, які можна надсилати та отримувати за допомогою програмного забезпечення для підтримки чатів [1]-[6].

Обмін текстовими повідомленнями є основою програмного забезпечення для підтримки чатів, що дозволяє користувачам спілкуватися за допомогою тексту. Програмне забезпечення для чатів пропонує такі функції, як відправка повідомлень, відображення вхідних повідомлень і можливість ставити питання для полегшення спілкування [1]-[6].

Різні програмні засоби для підтримки чатів також підтримують голосові та відео повідомлення. Користувачі можуть записувати та надсилати голосові нагадування або короткі відео кліпи для покращення своїх текстових повідомлень [1]-[6].

Проаналізуємо особливості та функції програмних платформ для чату [1]-[6].

Програмні платформи чату пропонують багато можливостей і функцій, які роблять їх популярними та універсальними. Обмін повідомленнями в реальному часі дозволяє користувачам спілкуватися миттєво, коли повідомлення одразу з'являються на екрані одержувача. Це сприяє швидкому та ефективному спілкуванню, усуваючи затримки, притаманні традиційним соціальним медіа [1]-[6].

Програмні засоби для чату також часто дозволяють користувачам легко ділитися мультимедійним контентом. Це можуть бути зображення, відео, документи чи аудіо файли, і такі платформи забезпечують зручний спосіб обміну різними типами медіа [1]-[6].

Програмні засоби для групових чатів та голосових чатів дозволяють спілкуватися з кількома людьми одночасно. Ця функція є надзвичайно корисною для групової співпраці, колективних обговорень або просто спілкування з друзями та родиною [1]-[6].

Технології для голосових та відео дзвінків додають миттєвості та персоналізації до чатів. Користувачі такого програмного забезпечення можуть здійснювати голосові або відео дзвінки, що забезпечує більш змістовну взаємодію, навіть якщо вони знаходяться на відстані [1]-[6].

Багато програмних засобів у вигляді чат-платформ пропонують інтеграцію з іншими інструментами та сервісами для підвищення їх функціональності. Вони можуть взаємодіяти з робочими інструментами, що дозволяє безперебійно співпрацювати в чатах [1]-[6].

Існує багато переваг використання чатів для окремих осіб та організацій. Чат-платформи забезпечують миттєвий обмін інформацією в

реальному часі, що дозволяє користувачам швидко обмінюватися даними. Незалежно від того, чи потрібно керувати проектами, організовувати заходи або просто спілкуватися з друзями, чати дозволяють вести змістовні розмови у будь-який момент [1]-[6].

Чат-платформи пропонують рівень зручності та доступності, що перевершує традиційні методи спілкування. Користувачі можуть надсилати повідомлення, здійснювати дзвінки та брати участь у відео конференціях з будь-якого місця за наявності доступу до Інтернету [1]-[6].

Багато чат-платформ пропонують інтеграцію з іншими інструментами та сервісами для підвищення їх функціональності.

Для забезпечення безпечного середовища в чатах необхідно, щоб працівники були обізнані про можливі загрози. Користувачам варто бути уважними під час надання особистої інформації, уникати підозрілих посилань і користуватися надійними джерелами [1]-[6].

Індустрія чатів продовжує розвиватися, відкриваючи нові можливості. Штучний інтелект (ШІ) стає важливим чинником у розвитку комунікаційних платформ. Чат-боти та віртуальні помічники, засновані на ШІ, стають дедалі досконалішими, щоб покращити обслуговування клієнтів, автоматизувати процеси та прискорити взаємодію [1]-[6].

Чат-боти та віртуальні помічники активно допомагають користувачам чат-платформ, надаючи інформацію. Ці інтелектуальні програми пропонують автоматичні відповіді, персоналізовані рекомендації та швидкий доступ до послуг [1]-[6].

Чат-кімнати можуть бути інтегровані з іншими новітніми технологіями, такими як доповнена та віртуальна реальність. Це може створити більш захоплюючий та інтерактивний досвід спілкування, розширюючи межі між цифровим та реальним світом. Незалежно від того, чи це текстові, голосові чи відео чати, чат-кімнати пропонують широкий спектр функцій і послуг, які полегшують спілкування та співпрацю. З розвитком технологій майбутнє чатів

відкриває захоплюючі можливості для більш глибокого та складнішого спілкування [1]-[6].

Програмне забезпечення для чату являє собою Інтернет платформу для обміну повідомленнями, яка дозволяє взаємодіяти між двома або більше людьми. Цей термін також стосується інструментів або сервісів, які сприяють такій комунікації. Існує декілька видів чатів. Чати для текстових повідомлень дозволяють спілкування через текстові повідомлення. Користувачі такого програмного забезпечення можуть також ділитися зображеннями, відео, посиланнями та документами [1]-[6].

Голосові та відео чати є поєднанням голосу, звуку і відео в одному місці, що дозволяє спілкуватися в режимі реального часу. Відео дзвінки полегшують взаємодію між людьми, які знаходяться в різних містах. Вони також дозволяють обмінюватися документами та іншими матеріалами під час спілкування [1]-[6].

Більшість програм обміну повідомленнями підтримують надсилання голосових повідомлень. Це особливо зручно для групових розмов [1]-[6].

Спілкування у чаті відбувається в режимі реального часу. Наприклад, коли ви спілкуєтеся з кимось, весь відформатований текст одразу стає доступним для інших учасників. На відміну від цього, інші форми текстового спілкування, такі як електронна пошта, є асинхронними і не відбуваються в реальному часі [1]-[6].

Чат може складатися з кількох людей, які спілкуються одночасно, і все сказане в чаті бачать всі учасники. Користувачі програмного забезпечення для чату також можуть надіслати приватні повідомлення або прямі повідомлення, щоб спілкуватися лише з однією особою в розмові [1]-[6].

Чат у реальному часі є програмною платформою обміну повідомленнями, що дозволяє поточним і потенційним клієнтам взаємодіяти з вашою компанією через вебсайт або додаток. Поєднуючи цей вид спілкування з іншими функціями, такими як боти та автоматизація, можна миттєво відповідати на запитання клієнтів, отримувати більше корисної інформації,

покращувати користувацький досвід і надавати більш складні повідомлення для цільових груп. Зрештою, живий чат допомагає компаніям будувати більш особисті та прямі стосунки з клієнтами [1]-[6].

Традиційно більшість підприємств розглядали обслуговування клієнтів як центральний елемент у ланцюжку створення вартості, ніж це є насправді. Тому вони намагалися зменшити ці витрати, покладаючись на програмні системи підтримки клієнтів, які більше орієнтовані на обслуговування, а не на досвід клієнтів [1]-[6].

Живий чат – є ефективним засобом обслуговування клієнтів, що відповідає потребам сучасного ринку. З усіма цими можливостями стає зрозуміло, що програмні засоби підтримки живого чату можуть мати значні переваги як для компаній, так і для клієнтів. Проаналізуємо важливі переваги, які можна отримати, використовуючи програмні засоби підтримки живого чату [1]-[6].

Програмні засоби підтримки живого чату дозволяють забезпечити надання швидкої допомоги клієнтам та є важливими для забезпечення оперативної допомоги. Використовуючи поєднання людських та автоматизованих відповідей за допомогою відповідного програмного забезпечення, можна швидко надавати підтримку в будь-який час і будь-де. Оскільки найкращі чат-інструменти використовують цінні дані, програмні агенти та боти мають доступ до важливої цифрової інформації [1]-[6].

Програмні засоби підтримки живого чату дозволяють компаніям реагувати швидше та надавати більш персоналізовану допомогу, використовуючи дані про історію поведінки клієнтів для надання кращих і більш точних відповідей [1]-[6].

На відміну від інших засобів підтримки клієнтів, програмні засоби живого чату працюють за сценарієм, що дозволяє зберігати всі розмови в реальному часі в пошуковій базі даних інформації про клієнтів. Використовуючи ці дані та спеціалізоване програмне забезпечення можна створити детальні профілі клієнтів, які дозволять надавати більш

персоналізовану підтримку. Чат також дозволяє легко визначати тенденції в розмовах клієнтів, що допомагає виявляти важливі для них проблеми та швидко їх вирішувати [1]-[6].

Програмні засоби підтримки живого чату можна використовувати на всіх етапах взаємодії з клієнтом, що робить його ідеальним інструментом для створення єдиного досвіду у відділах продажів, маркетингу та підтримки. Застосування програмного забезпечення для підтримки живого чату на всіх етапах взаємодії з клієнтом дозволяє створити цілісний і послідовний клієнтський досвід у відділах продажів, маркетингу та підтримки. Розуміючи, хто клієнт і на якому етапі знаходиться взаємодія з ним, можна ефективно використовувати цільові соціальні мережі для взаємодії в потрібний час, уникаючи надсилання непотрібних повідомлень [1]-[6].

Ефективне програмне забезпечення для підтримки чату поєднує людську та автоматизовану підтримку, що може принести значні операційні переваги для бізнесу. За допомогою автоматизації та чат-ботів можна швидко вирішувати прості питання. Програмне забезпечення підтримки чатів та чат-боти дозволяють персоналу служби підтримки зосередитися на складніших завданнях, які потребують людського втручання, таких як технічні проблеми чи складні запити щодо оплати. Оптимізуючи процеси підтримки за допомогою відповідного програмного забезпечення, можна економити час, гроші та зусилля, одночасно забезпечуючи високий рівень обслуговування клієнтів [1]-[6].

Ефективна чат-підтримка використовує поєднання людських ресурсів та програмних систем, таких як чат-боти. Це означає, що іноді користувач спілкується з ботом, а іноді з реальною людиною. Чат-боти можуть швидко й ефективно відповідати на прості запитання, надаючи персоналізовану допомогу в будь-який час доби. Водночас, для вирішення складніших або делікатних питань працівники служби підтримки з використанням спеціалізованого програмного забезпечення можуть запропонувати більш детальні та ефективні рішення [1]-[6].

Програмне забезпечення для підтримки чату дозволяє автоматизувати спрямування запитів. Щоб забезпечити надходження правильних запитів до відповідних модулів програмного забезпечення, можна використовувати ботів для автоматичного спрямування розмов до певних груп або осіб на основі наявної інформації. Наприклад, запити від поточних клієнтів можна направляти до служби підтримки, а від потенційних клієнтів до відділу продажів. Це економить час, оскільки клієнти не повинні повторно пояснювати свої потреби, а спеціалісти можуть швидше розпочати роботу над вирішенням питань [1]-[6].

Для підприємств, які щорічно залучають тисячі або мільйони відвідувачів на свої вебсайти, особисте спілкування з кожним клієнтом і потенційним клієнтом може бути надзвичайно важким і дорогим. Натомість програмні засоби підтримки живого чату допомагають команді максимально ефективно використовувати наявні ресурси [1]-[6].

Використовуючи підтримку за допомогою програмних засобів чат-платформи, є можливість покращити користувацький досвід, допомагаючи клієнтам із взаємодією на вебсайті компанії, продукті чи додатку, вирішуючи їхні проблеми. Поєднуючи миттєвий зворотний зв'язок, який може надати програмне забезпечення підтримки живого чату, з підтримкою клієнтів, можна швидше розв'язувати проблеми, що призведе до більш продуктивної та ефективної взаємодії з клієнтами [1]-[6].

За допомогою чат-ботів і програмних засобів обслуговування клієнтів більшість інструментів живого чату дозволяють проводити динамічні розмови, навіть коли на іншому кінці немає агента. Наприклад, можна налаштувати автоматизовані сценарії, які запитують у відвідувачів їхні електронні адреси, щоб зв'язатися з ними пізніше [1]-[6]. Також можна підняти обслуговування на новий рівень за допомогою інтелектуального чат-бота, який автоматично й негайно вирішує проблеми клієнтів. Розробку чат-ботів треба організовувати таким чином, щоб клієнти могли легко знаходити потрібну інформацію, надаючи відповіді, що відповідають їхнім запитам [1]-[6].

Використовуючи ефективні програмні засоби живого чату, користувачі можуть ефективно обробляти скарги, зберігаючи персоналізований, людський підхід до обслуговування. Програмні засоби живого чату варто використовувати їх як допомогу для команди підтримки, продажів і маркетингу, щоб передати клієнтам розмовне повідомлення, яке підкреслить особливості відповідного продукту [1]-[6].

Програмні засоби живого чату можна використовувати для автоматичного вітання цільових відвідувачів, які вже демонструють інтерес до бренду компанії. Це дозволяє підвищити взаємодію, запропонувавши персоналізоване вітання через чат, яке інтегрує візуальну комунікацію з елементами штучного інтелекту [1]-[6].

Живий чат на основі відповідних програмних засобів може забезпечити спілкування з постійними відвідувачами, які часто повертаються на сайт і активно переглядають певні сторінки. Це може свідчити про їхній інтерес до покупки продукту або до особистого консультування. Живий чат дозволяє вчасно відкликатися на запитання клієнтів і надавати необхідну інформацію про товари або послуги, які вони розглядають [1]-[6].

Варто ввикористовувати живий чат, щоб стимулювати відвідувачів до здійснення покупок. За допомогою програмних засобів підтримки чату можна стимулювати клієнтів до оформлення замовлення, суттєво збільшуючи таким чином продаж [1]-[6].

Програмні засоби живого чату дозволяють використовувати внутрішній зв'язок між відділами продажу, підтримки і маркетингу для забезпечення узгодженості життєвого циклу клієнта і підвищення ефективності організації [1]-[6]. Живий чат може використовуватися як частина багатоканальної стратегії організації [1]-[6]. При цьому важливо переконатися, що відповідні програмні засоби для живого чату інтегрується з іншими інструментами та каналами, які використовуються, щоб надавати підтримку клієнтам там, де вони знаходяться, і забезпечити їм кращий досвід. Живий чат може застосовуватися для створення персоналізованих розмов, які

підкреслюють їхню цінність для клієнтів, зберігаючи при цьому легкий і невимушений тон, клієнти при цьому можуть бачити імена та фотографії профілів співробітників, з якими вони спілкуються [1]-[6].

Програмні засоби живого чату можуть використовуватися для отримання додаткового контексту запитів клієнтів шляхом інтеграції з іншими соціальними медіа та активами, що дозволяє отримати більше інформації та надавати більш точні відповіді клієнтам. Завчасно допомагаючи клієнтам, використовуючи відповідні програмні засоби підтримки живих чатів, навіть до того, як вони самі усвідомлюють свої потреби, можна зменшити кількість скарг до служби підтримки та суттєво збільшити рівень задоволеності клієнтів компанії [1]-[6].

Визначено, що чати являють собою програмні інструменти, що дозволяють людям спілкуватися в режимі реального часу, вони сприяють взаємодії між людьми, які можуть бути географічно віддаленими один від одного. Відзначено, що програмні засоби, які являють собою чат-платформи, складаються з віртуальних кімнат, де користувачі можуть надсилати текстові, голосові або відео повідомлення, такі програми дозволяють миттєво спілкуватися, забезпечуючи безперервний обмін інформацією навіть на великих відстанях. Проаналізовано класифікацію чатів. Програмні засоби для підтримки функціонування чатів, як правило, дозволяють надсилати текстові, голосові або відео повідомлення, а також пропонують різноманітні формати для задоволення різних потреб користувачів у спілкуванні. Проаналізовано особливості та функції програмних платформ для чату.

1.2 Аналіз програмного забезпечення для групових чатів

Проаналізуємо програмне забезпечення для групових чатів [7]-[12].

Програмне забезпечення Acquire (рис. 1.1) пропонує багатофункціональну платформу для обміну повідомленнями, включаючи інструменти для співпраці. Завдяки автоматизованій допомозі новим клієнтам

і швидким відповідям на запити, живий чат забезпечує швидку, персоналізовану підтримку, що сприяє збільшенню кількості взаємодій з конкретними клієнтами [7]-[10].

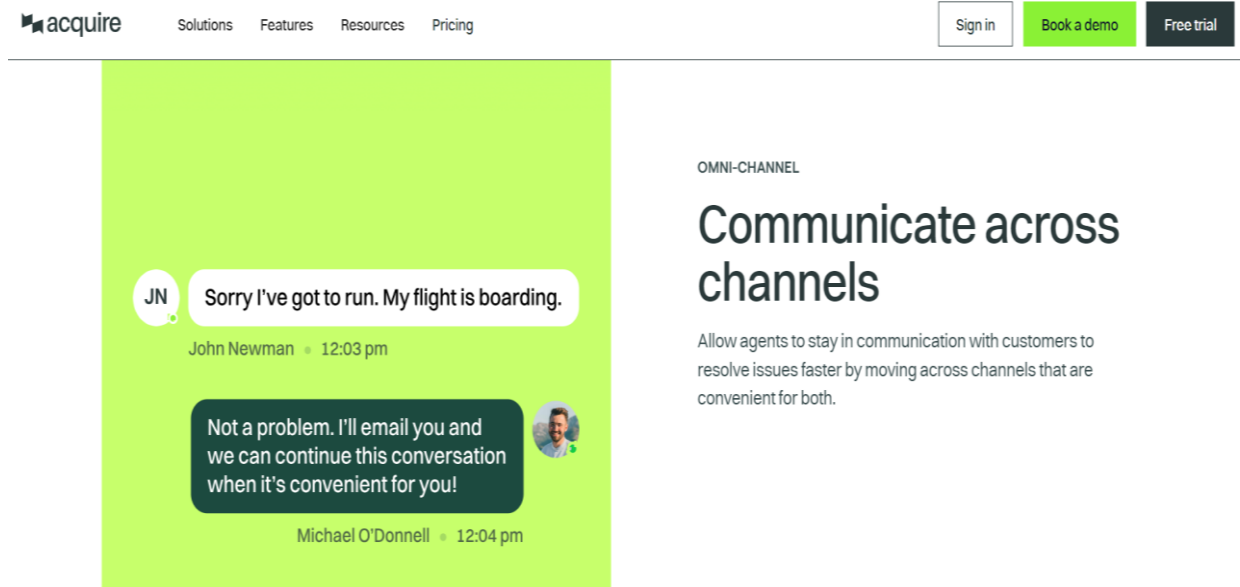


Рисунок 1.1 – Програмне забезпечення Acquire [10]

Програмне забезпечення Acquire виділяється серед інших подібних рішень завдяки своїм можливостям спільного використання комп'ютерів і мобільних пристроїв. Завдяки функції підключення команди продажів можуть упорядковувати розмови та зменшувати тертя в процесі продажу [7]-[10].

Програмне забезпечення Acquire надає можливості використання автоматизованих або стандартних повідомлень для відповіді на часті запити. Завдяки інтелектуальному зворотному зв'язку, штучний інтелект розпізнає певні слова та автоматично відповідає на них. Таким чином, він надає відповіді на поширені запитання клієнтів без необхідності залучення вашої команди [7]-[10].

Іншими унікальними функціями Acquire є засоби інтеграції та багатомовне спілкування. Журнали чату дозволяють команді ділитися конфіденційною інформацією про розмови з іншими членами команди для

швидшого вирішення питань. Функція розпізнавання та перекладу мови клієнта дозволяє спілкуватися з клієнтами їхньою рідною мовою [7]-[10].

Програмне забезпечення Asquire підтримує процеси для живого чату на вебсайті компанії. Програма Asquire для чату підтримує автоматичний переклад на велику кількість мов, що дозволяє агентам підтримки відповідати на запити клієнтів без знання іноземних мов. Можливості обміну мультимедійними файлами є ще однією причиною, чому цей живий чат для вебсайтів виділяється. Програмне забезпечення Asquire дозволяє ділитися файлами, спільно переглядати документи та спілкуватися з клієнтами за допомогою аудіо. Програмне забезпечення Asquire також підтримує відео чат, що робить Asquire відповідним вибором для компаній, які шукають програмне забезпечення для відео чату [7]-[10].

Програмне забезпечення Asquire надає потужну платформу для підтримки клієнтів, яка є інтегрованим рішенням для взаємодії через живий чат, відео, голос та повідомлення інших видів. Програмне забезпечення Asquire дозволяє компаніям налаштовувати та автоматизувати робочі процеси, керувати пріоритетами та чергами, а також безперешкодно перемикатися між різними каналами зв'язку. Унікальна функція бізнес-правил платформи дозволяє створювати складні робочі процеси підтримки, які можуть направляти взаємодію з клієнтами до найбільш відповідних агентів, відділів або каналів [7]-[10].

Програмне забезпечення Asquire підходить для компаній, які шукають комплексне багатоканальне рішення для підтримки клієнтів. Ця платформа особливо корисна для компаній із глобальною клієнтурою, завдяки своїй багатомовній підтримці. Вона також підходить для підприємств, яким потрібно автоматизувати та налаштовувати робочі процеси підтримки клієнтів, щоб оптимізувати розподіл ресурсів та підвищити ефективність обслуговування [7]-[10].

Основними характеристиками програмного забезпечення Asquire є такі [7]-[10]:

- багатоканальна підтримка: програмне забезпечення Asquire надає функції, що підтримують живий чат, повідомлення у вигляді відео та голосові повідомлення [7]-[10];

- автоматизація та налаштування: програмне забезпечення Asquire підтримує робочі процеси, що налаштовуються, а також автоматизовану маршрутизацію запитів за допомогою бізнес-правил [7]-[10];

- засоби інтеграції: програмне забезпечення Asquire забезпечує повну інтеграцію з існуючими робочими процесами та програмними платформами користувачів [7]-[10];

- можливості мультимедійної взаємодії: програмне забезпечення Asquire підтримує обмін файлами, спільний перегляд, відео чат [7]-[10];

- багатомовна підтримка: програмне забезпечення Asquire підтримує переклад повідомлень на більш велику кількість мов [7]-[10];

- аналітика та звітність: програмне забезпечення Asquire надає засоби аналітики, а також сповіщень про виконання завдань [7]-[10];

- можливість вбудовування у вебсайти: програмне забезпечення Asquire надає засоби для вбудовування у вебсайти [7]-[10];

- інтелектуальні відповіді: програмне забезпечення Asquire підтримує використання штучного інтелекту для автоматичної відповіді на часті запити, інтелектуальний зворотний зв'язок [7]-[10];

Серед недоліків програмного забезпечення Asquire можна відзначити відсутність мобільного застосунку для роботи на ходу, що може бути незручно для агентів, які потребують доступу до системи поза офісом [7]-[10]. Іншим недоліком є обмежена функціональність для складних робочих процесів: система не завжди підтримує складні робочі процеси або не відповідає специфічним вимогам великих підприємств [7]-[10].

Програмне забезпечення Pure Chat (рис. 1.2) виділяється серед програмного забезпечення для підтримки живого чату завдяки інтеграції аналітики відвідувачів, історії контактів і функції чату на основі тригерів. Його акцент на взаємодії з клієнтами на основі даних дозволяє компаніям

надавати персоналізований досвід протягом усього життєвого циклу клієнта. Відсутність плати за кожного користувача або контакт робить програмне забезпечення Pure Chat привабливим вибором для компаній, які шукають економічно вигідне рішення [7]-[9], [11].

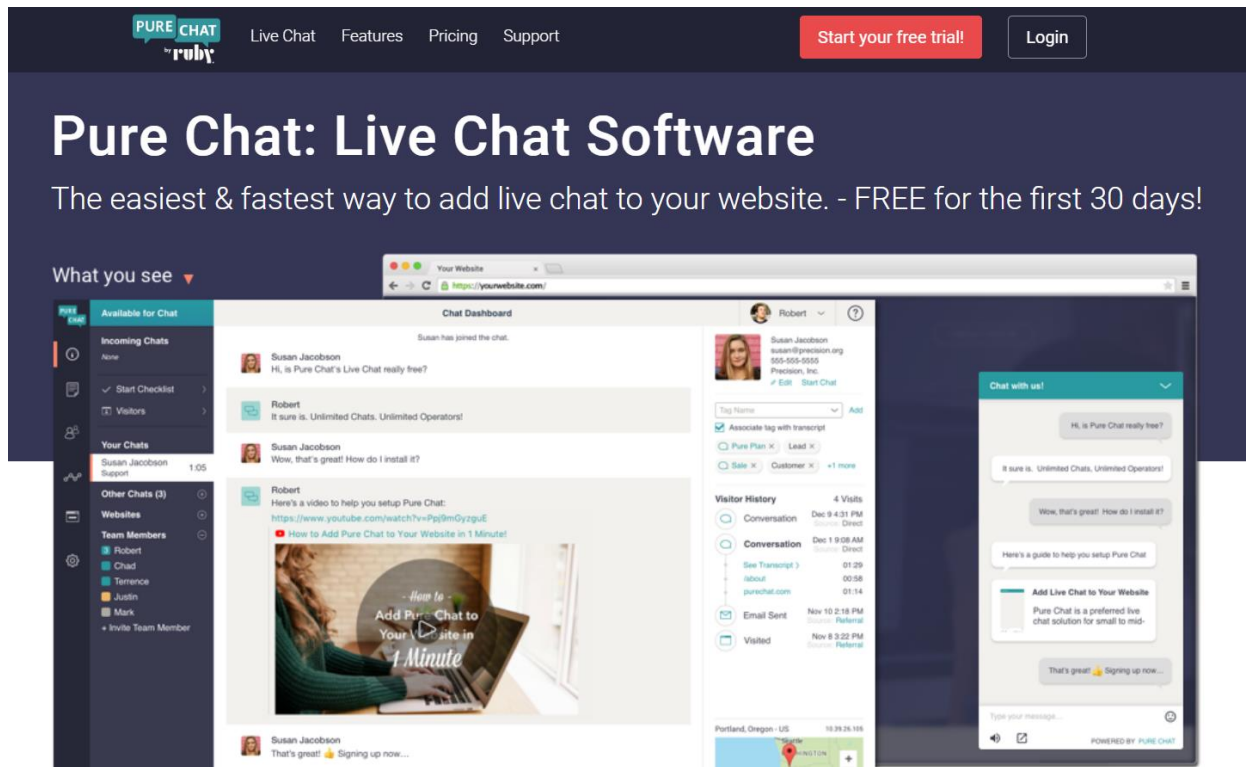


Рисунок 1.2 – Програмне забезпечення Pure Chat [11]

Зручність у використанні, налаштування чату та сумісність із популярними програмами сприяють його привабливості. Крім того, мобільні додатки для iOS і Android дозволяють користувачам керувати взаємодією в чаті в будь-який час. Нарешті, 30-денна гарантія повернення грошей від Pure Chat дає користувачам можливість випробувати програмне забезпечення без фінансових ризиків [7]-[9], [11].

Програмне забезпечення Pure Chat є одним із найкращих рішень для Інтернет чату для компаній середнього розміру завдяки аналітиці відвідувачів у реальному часі. Команди продажів можуть використовувати такі дані відстеження відвідувачів, як історія розмов, відвідані сторінки та контактну

інформацію, щоб легше знаходити потенційних клієнтів і підвищувати ефективність своїх маркетингових зусиль [7]-[9], [11].

Програмне забезпечення Pure Chat можна інтегрувати з такими інструментами, як Google Analytics, Salesforce, MailChimp і HubSpot. Це дозволяє компанії підтримувати співробітників протягом усього шляху клієнта [7]-[9], [11].

Програмне забезпечення Pure Chat є відмінним вибором, якщо користувач готовий перейти на платний план, оскільки безкоштовна пробна версія включає всі функції і дозволяє випробувати їх. Програмне забезпечення Pure Chat пропонує необмежену кількість чатів і стенограм, розширені можливості налаштування, інтеграцію з системою управління підприємством, сповіщення в чаті, готові відповіді, інформацію про відвідувачів і повідомлення в чаті [7]-[9], [11]. Pure Chat найкраще підходить для досвідчених команд з продажу та маркетингу, які шукають комплексне рішення для живого чату, що пропонує збалансоване поєднання статистичних даних, доступності та можливостей налаштування [7]-[9], [11].

Програмне забезпечення Pure Chat підходить для компаній, яким потрібна інтеграція з популярними маркетинговими та аналітичними інструментами, а також з багатьма іншими програмами. Хоча це може не бути найкращим варіантом для невеликих компаній або тих, хто має прості потреби, Pure Chat є чудовим вибором для організацій, які прагнуть оптимізувати кожну взаємодію з відвідувачами свого вебсайту за допомогою даних та аналітики [7]-[9], [11].

Програмне забезпечення Pure Chat надає повнофункціональний продукт для живого чату з аналітикою відвідувачів та історією контактів, інтегрується з різними програмами [7]-[9], [11].

Програмне забезпечення Pure Chat має такі особливості [7]-[9], [11]:

- аналітика відвідувачів у реальному часі: програмне забезпечення Pure Chat надає засоби збору та аналізу даних про відвідувачів сайту, засоби

відстеження поведінки відвідувачів, включаючи відвідані сторінки та історію розмов [7]-[9], [11];

- історія контактів: програмне забезпечення Pure Chat підтримує зберігання всієї історії чатів для кожного відвідувача, а також надає можливість перегляду попередніх розмов для кращого обслуговування клієнтів [7]-[9], [11];

- чат на основі тригерів: автоматичний запуск чатів на основі певних дій відвідувачів (наприклад, час перебування на сторінці, перегляд певної кількості сторінок тощо) [7]-[9], [11];

- інтеграція з популярними інструментами: програмне забезпечення Pure Chat підтримує інтеграцію з Google Analytics, Salesforce, MailChimp, HubSpot та іншими програмами [7]-[9], [11];

- підтримка налаштувань кольорів відповідно до стилю компанії та програмного забезпечення [7]-[9], [11];

- мобільні застосунки для iOS та Android: можливість керування чатами з мобільних пристроїв, сповіщення про нові чати та повідомлення на мобільних пристроях [7]-[9], [11];

- сповіщення про нові повідомлення в чаті, можливість використання готових відповідей для швидкого реагування на запити клієнтів [7]-[9], [11];

- розширені налаштування та інтеграція з системою управління компанією: інтеграція з системою управління компанією для кращого управління клієнтськими даними, розширені можливості налаштування для задоволення специфічних потреб бізнесу [7]-[9], [11];

- повна статистика та звіти про використання чату та ефективність взаємодії, засоби аналітики для покращення маркетингових та продажних стратегій [7]-[9], [11].

Серед недоліків програмного забезпечення Pure Chat можна відзначити обмежені функції у безкоштовній версії: безкоштовна версія Pure Chat має обмежені можливості порівняно з платними планами, що може бути недостатнім для певних бізнес-потреб [7]-[9], [11]. Іншим недоліком є те, що

Pure Chat найбільше підходить для середніх та великих компаній з високими потребами щодо аналітики та інтеграції з іншими системами. Для невеликих підприємств або тих, хто має простіші потреби, програмне забезпечення Pure Chat може бути занадто потужним та складним у використанні [7]-[9], [11].

Програмне забезпечення Tidio (рис. 1.3) є універсальним програмним забезпеченням для живого чату, створеним спеціально для компаній електронної комерції, які бажають взаємодіяти з відвідувачами своїх сайтів, залучати потенційних клієнтів і підвищувати продажі [7]-[9], [12].

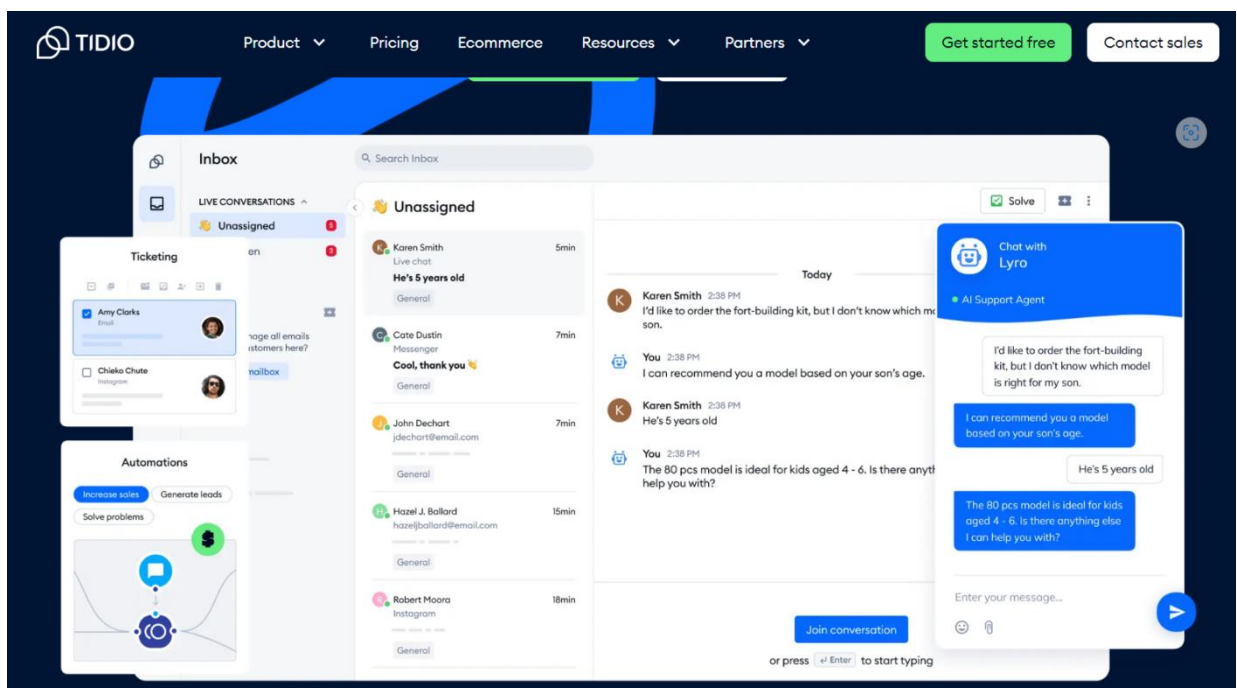


Рисунок 1.3 – Програмне забезпечення Tidio [12]

Одна з ключових функцій Tidio – це чат-боти, які допомагають компаніям виявляти потреби та бажання клієнтів і надавати їм рішення в режимі реального часу. Всі відповіді клієнтів збираються в єдиній організованій папці, що дозволяє агентам служби підтримки ефективніше вирішувати проблеми клієнтів разом [7]-[9], [12].

Програмне забезпечення Tidio також пропонує різні можливості автоматизації. Наприклад, можна налаштувати автоматичні відповіді, які активуються певними умовами або подіями, як-от повторне відвідування

клієнтом сторінки сайту. Це персоналізує взаємодію з клієнтами і допомагає усунути бар'єри на їхньому шляху [7]-[9], [12].

Крім того, програмне забезпечення Tidio можна інтегрувати з багатьма програмними інструментами та платформами, зокрема, його можна використовувати разом з такими сервісами, як Shopify і WordPress, що робить роботу команди зручнішою й ефективнішою [7]-[9], [12].

Програмне забезпечення Tidio має такі особливості [7]-[9], [12]:

- живий чат у реальному часі: програмне забезпечення Tidio надає можливість спілкування з відвідувачами сайту в реальному часі [7]-[9], [12];
- зручний інтерфейс для швидкої та ефективної взаємодії [7]-[9], [12];
- чат-боти: автоматизовані чат-боти для допомоги клієнтам і виявлення їхніх потреб, програмне забезпечення Tidio надає можливість налаштування чат-ботів для виконання різних завдань, таких як відповідь на запити, збирання даних і надання інформації [7]-[9], [12];
- єдина організована папка, в якій зберігаються всі повідомлення клієнтів, що спрощує їх обробку, а також надає можливість спільної роботи агентів служби підтримки над вирішенням проблем клієнтів [7]-[9], [12];
- функції автоматизації: налаштування автоматичних відповідей, які активуються певними умовами або подіями, персоналізація взаємодії з клієнтами шляхом автоматизації процесів [7]-[9], [12];
- інтеграція з іншими інструментами та платформами: підтримка інтеграції з популярними платформами, такими як Shopify, WordPress, та іншими, можливість використання Tidio в різних екосистемах для зручності роботи [7]-[9], [12];
- мобільні застосунки: доступні мобільні застосунки для iOS і Android, можливість керування чатами з мобільних пристроїв [7]-[9], [12];
- персоналізація та налаштування: можливість змінювати зовнішній вигляд і функціональність чату [7]-[9], [12];

– аналітика та звіти: вбудовані інструменти аналітики для відстеження ефективності взаємодії з клієнтами; звіти про діяльність чату, що допомагають покращити стратегії обслуговування клієнтів [7]-[9], [12].

Серед недоліків програмного забезпечення Tidio можна відзначити обмежені функції у безкоштовній версії: безкоштовна версія Tidio має обмежені можливості порівняно з платними планами, що може бути недостатнім для певних бізнес-потреб [7]-[9], [12]. Іншим недоліком є те, що Tidio найбільше підходить для середніх та великих компаній з високими потребами щодо аналітики та інтеграції з іншими системами. Для невеликих підприємств або тих, хто має простіші потреби, програмне забезпечення Tidio може бути занадто потужним та складним у використанні [7]-[9], [12].

Порівняльну характеристику програмного забезпечення для групових чатів наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння програмного забезпечення для групових чатів

Критерій порівняння	Acquire [10]	Pure Chat [11]	Tidio [12]
Можливості мультимедійної взаємодії	+	+—	+—
Багатомовна підтримка	+	—	—
Спільний перегляд	+	—	—
Доцільність використання організаціями різних розмірів	—	+—	+—
Засоби інтеграції	+	+	+—
Мобільний застосунок	—	+	+
Автоматизація та налаштування	+	+—	+

За результатами проведеного аналізу можна зробити висновок, що у наш час існує досить багато програмних засобів для підтримки групових чатів.

Проте деякі програми для підтримки групових чатів не забезпечують достатній рівень шифрування, що може стати проблемою для компаній, які обробляють чутливі дані. Інше програмне забезпечення для підтримки адміністрування групових чатів може мати проблеми з інтеграцією і бути несумісним з програмами, що вже використовуються в організації. Деякі програми для адміністрування чатів можуть не забезпечують достатній рівень функціональності. Тому актуальною є розробка програмного забезпечення для адміністрування групових чатів.

При розробці програмного забезпечення для адміністрування групових чатів необхідно забезпечити такі функціональні вимоги:

- можливість надання інформації щодо чату;
- можливість отримання інформації про користувачів;
- підтримка можливості додавання та видалення користувачів чату;
- підтримка можливості блокування деяких користувачів;
- підтримка можливості керування каналами чатів;
- підтримка можливості зміни налаштувань чату.

1.3 Висновки за розділом 1

Визначено, що чати являють собою програмні інструменти, що дозволяють людям спілкуватися в режимі реального часу, вони сприяють взаємодії між людьми, які можуть бути географічно віддаленими один від одного. Відзначено, що програмні засоби, які являють собою чат-платформи, складаються з віртуальних кімнат, де користувачі можуть надсилати текстові, голосові або відео повідомлення, такі програми дозволяють миттєво спілкуватися, забезпечуючи безперервний обмін інформацією навіть на великих відстанях. Проаналізовано класифікацію чатів. Програмні засоби для підтримки функціонування чатів, як правило, дозволяють надсилати текстові, голосові або відео повідомлення, а також пропонують різноманітні формати

для задоволення різних потреб користувачів у спілкуванні. Проаналізовано особливості та функції програмних платформ для чату.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для підтримки групових чатів. Проте деякі програми для підтримки групових чатів не забезпечують достатній рівень шифрування, що може стати проблемою для компаній, які обробляють чутливі дані. Інше програмне забезпечення для підтримки адміністрування групових чатів може мати проблеми з інтеграцією і бути несумісним з програмами, що вже використовуються в організації. Деякі програми для адміністрування чатів можуть не забезпечують достатній рівень функціональності. Тому актуальною є розробка програмного забезпечення для адміністрування групових чатів. Тому актуальною є розробка програмного забезпечення для адміністрування групових чатів.

Сформульовано функціональні вимоги до програмного забезпечення для адміністрування групових чатів.

2 МАТЕРІАЛИ І МЕТОДИ

2.1 Вибір мови програмування

При розробці програмного забезпечення для адміністрування групових чатів було використано мову програмування C# [13], [14].

Мова програмування C# є однією з найпотужніших, швидкозростаючих і найпопулярніших мов у сфері розробки інформаційних технологій. Вона використовується для створення різноманітних додатків, від невеликих настільних програм до великих вебпорталів і вебсервісів, які обслуговують мільйони користувачів щодня [13], [14].

Мова програмування C# використовується на багатьох платформах. Мова пройшла значний шлях з моменту випуску її першої версії разом з Microsoft Visual Studio.Net. Мова програмування C# подібна до C і в цьому аспекті ближча до C++ і Java [13], [14].

Завдяки своїм можливостям C# стала основною мовою для движка Unity, який є провідною платформою для створення ігор для Windows. Підтримка машинного імпорту та висока ефективність роблять C# ще більш популярною [13], [14].

Останні програмні продукти були створені з використанням різних інструментів та рідко запускалися виключно на чистому C#. Якщо порівнювати C# з іншими мовами, ця мова є відносно новою, хоча й пройшла значний розвиток. Повна версія мови була випущена разом з Microsoft Visual Studio. C# є структурованою мовою, що запозичує багато концепцій з Java та C++. Наприклад, C# підтримує поліморфізм, успадкування, перевантаження та статичну типізацію. Об'єктно-орієнтований підхід дозволяє вирішувати складні задачі створення масштабованих додатків. C# продовжує розвиватися, додаючи нові корисні функції, такі як лямбда-вирази, динамічні посилання, асинхронні методи тощо [13], [14].

Оскільки C# є продуктом Microsoft, він має чудову підтримку та багату бібліотеку. У світі мови програмування C# доступні різноманітні бібліотеки,

включаючи ті, що працюють з пам'яттю та машинним навчанням, наприклад, ML.NET [13], [14].

Мова програмування C# дозволяє використовувати всі можливості сучасних підходів до розробки ПЗ, об'єднуючи їх в єдину мову програмування. Як реляційна мова, C# може застосовуватися для машинного навчання в будь-якому застосуванні, від мобільних до веброзробок [13], [14].

Обґрунтування вибору мови програмування для розробки програмного забезпечення для адміністрування групових чатів наведено у таблиці 2.1.

Таблиця 2.1 – Обґрунтування вибору мови програмування для розробки програмного забезпечення для адміністрування групових чатів

Критерій порівняння мов програмування	Мова програмування		
	C#	PHP	JavaScript
Механізми безпеки даних	+	—	—
Зручність синтаксису	+	+—	+
Підтримка асинхронності	+	+—	—
Зручність для створення ПЗ для адміністрування групових чатів	+	—	—
Об'єктно-орієнтованість мови	+	+—	—

Отже, для реалізації програмного забезпечення для адміністрування групових чатів обрано мову програмування C#, яка має вбудовані механізми безпеки, зокрема, засоби управління автентифікацією та авторизацією користувачів, засоби захисту даних та інші функції, що важливо для захисту конфіденційної інформації чату та збереження даних користувачів. Крім того, C# є об'єктно-орієнтованою мовою програмування, що дозволяє організувати код у вигляді класів і об'єктів, полегшуючи його підтримку, розширюваність та повторне використання.

2.2 Вибір середовища розробки для створення програмного забезпечення для адміністрування групових чатів

В якості середовища розробки для створення програмного забезпечення для адміністрування групових чатів було обрано середовище Microsoft Visual Studio [15], [16].

Середовище розробки Microsoft Visual Studio творена для розробки програмних застосунків різного призначення, зокрема, для вебслужб і вебзастосунків, які дозволяють отримувати доступ до контенту будь-де, будь-коли та на будь-якому пристрої [15], [16].

Середовище Microsoft Visual Studio забезпечує підтримку створення XML та вебзастосунків. Середовище Microsoft Visual Studio забезпечує підвищення продуктивності розробників, розширення використання мов програмування. Крім того, середовище Microsoft Visual Studio забезпечує підтримку всіх аспектів процесу створення ПЗ [15], [16].

Середовище Microsoft Visual Studio забезпечує можливість завантаження додаткових засобів розробки зі сторонніх сайтів. Є вкладки спільноти для доступу до джерел даних, а також вкладки для додаткових сайтів і прикладів, які можна завантажити [15], [16].

В інтегрованому середовищі розробки Microsoft Visual Studio доступними є інструменти для планування, аналізу, проєктування, конфігурації та налаштування корпоративних програм. Нова архітектура включає інструменти для проєктування застосунків, використовуючи мови моделювання даних [15], [16].

Різні компанії використовують середовище розробки Microsoft Visual Studio для швидкого та надійного створення ПЗ, зокрема, великих та складних проєктів програмного забезпечення [15], [16].

Середовище Microsoft Visual Studio надає панель керування, яка робить процес адміністрування простішим і швидшим. Середовище Microsoft Visual Studio підтримує сторонні бібліотеки, що дає розробникам можливість

розширювати функціонал за необхідності. Середовище Microsoft Visual Studio пропонує безкоштовні рішення для роботи з фреймворком .NET [15], [16]. Середовище Microsoft Visual Studio надає потужні інструменти для створення вебзастосунків [15], [16].

Обґрунтування вибору середовища розробки для створення програмного забезпечення для адміністрування групових чатів наведено у таблиці 2.2.

Таблиця 2.2 – Обґрунтування вибору середовища розробки для створення програмного забезпечення для адміністрування групових чатів

Критерій порівняння середовищ розробки	Середовища розробки		
	Microsoft Visual Studio	IntelliJ IDEA	JetBrains Rider
Розширений редактор коду	+	—	—
Зручна адміністративна панель	+	+—	+
Засоби інтеграції	+	+—	+—
Засоби авто доповнення	+	+—	+—
Зручність редактору коду	+	+	+—

Таким чином, для створення програмного забезпечення для адміністрування групових чатів обрано середовище розробки Microsoft Visual Studio, яке пропонує розширений редактор коду, засоби авто доповнення, рефакторингу та інші інструменти, які значно полегшують написання, читання та підтримку коду. Крім того, середовище розробки Microsoft Visual Studio забезпечує інтеграцію з .NET Framework, що дозволяє розробникам ефективно використовувати можливості цих платформ та розробляти програмні продукти мовою C#.

2.3 Розробка моделі взаємодії користувача з програмним забезпеченням для адміністрування групових чатів

З метою можливості розробки програмного забезпечення для адміністрування групових чатів створимо відповідну модель у вигляді діаграми прецедентів.

При розробленні моделі взаємодії користувача з програмним забезпеченням для адміністрування групових чатів враховуємо те, що користувачі програми мають можливості отримання інформації щодо чату, отримання інформації про користувачів (членів чату), додавання та видалення користувачів чату (встановлення прав доступу), блокування деяких користувачів (редагування прав членів чату), керування каналами чатів, зміни налаштувань чату та ін.

Основний успішний сценарій користувача при роботі з програмним забезпеченням для адміністрування групових чатів можна описати таким чином:

- користувач програмного забезпечення для адміністрування групових чатів авторизується;
- після авторизації користувача сервер програмного забезпечення для адміністрування групових чатів у відповідь повертає основну сторінку програми;
- користувач бачить відповідну сторінку програмного забезпечення для адміністрування групових чатів та обирає потрібну дію, наприклад, додавання користувача чату;
- користувач програми (адміністратор чату) додає користувача як нового члена чату;
- користувач натискає відповідну кнопку для додавання нового члену чату;
- користувач завершує роботу з програмним забезпеченням для адміністрування групових чатів.

Враховуючи сценарії роботи користувачів при роботі з програмою, модель взаємодії користувача з програмним забезпеченням для адміністрування групових чатів подамо у вигляді діаграми прецедентів, зображеної на рис. 2.1.

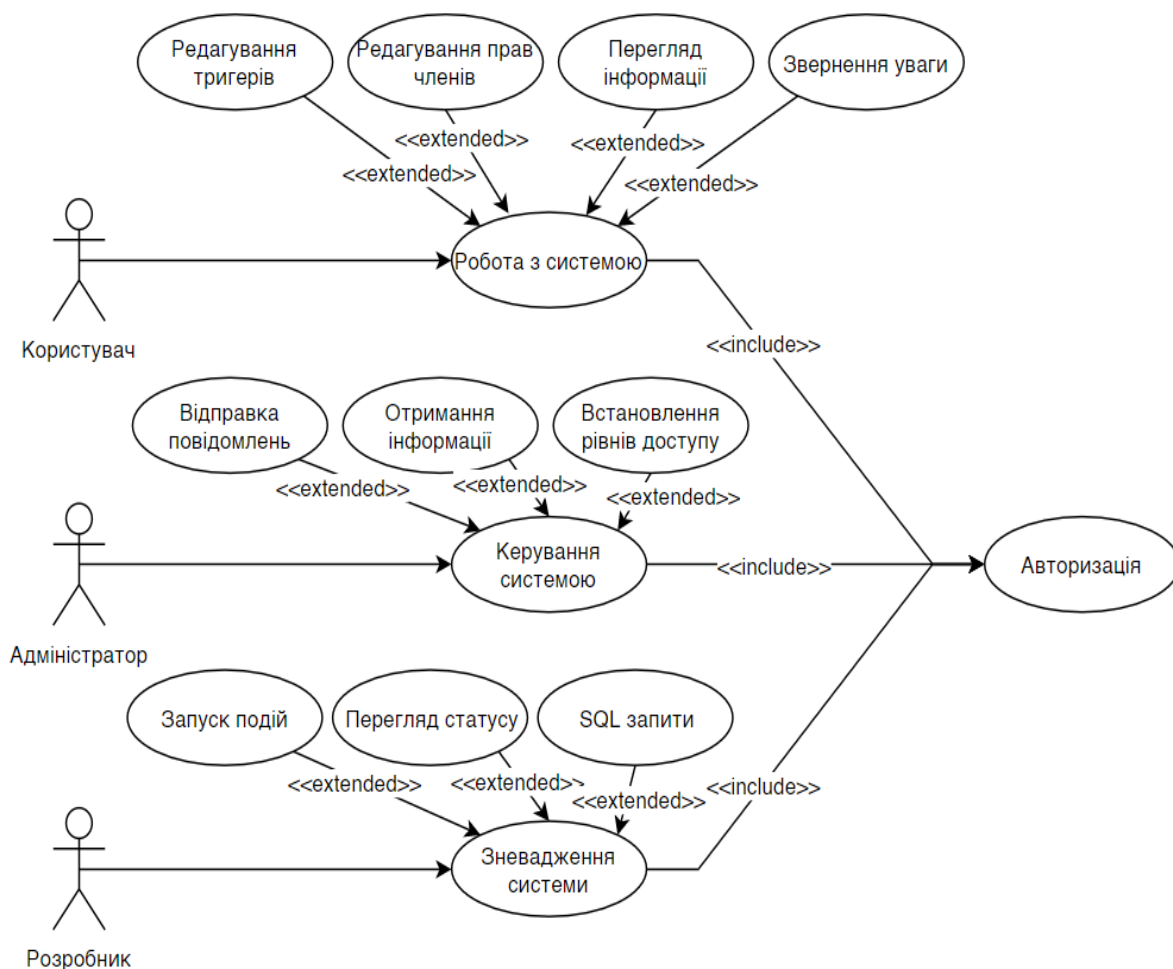


Рисунок 2.1 – Модель взаємодії користувача з програмним забезпеченням для адміністрування групових чатів

Таким чином, створено модель взаємодії користувача з програмним забезпеченням для адміністрування групових чатів, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення для адміністрування групових чатів.

2.4 Висновки за розділом 2

Для реалізації програмного забезпечення для адміністрування групових чатів обрано мову програмування C#, яка має вбудовані механізми безпеки, зокрема, засоби управління автентифікацією та авторизацією користувачів, засоби захисту даних та інші функції, що важливо для захисту конфіденційної інформації чату та збереження даних користувачів. Крім того, C# є об'єктно-орієнтованою мовою програмування, що дозволяє організувати код у вигляді класів і об'єктів, полегшуючи його підтримку, розширюваність та повторне використання.

Для створення програмного забезпечення для адміністрування групових чатів обрано середовище розробки Microsoft Visual Studio, яке пропонує розширений редактор коду, засоби авто доповнення, рефакторингу та інші інструменти, які значно полегшують написання, читання та підтримку коду. Крім того, середовище розробки Microsoft Visual Studio забезпечує інтеграцію з .NET Framework, що дозволяє розробникам ефективно використовувати можливості цих платформ та розробляти програмні продукти мовою C#.

Створено модель взаємодії користувача з програмним забезпеченням для адміністрування групових чатів, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення для адміністрування групових чатів.

3 ОПИС ПРОГРАМИ

3.1 Структура програмного забезпечення для адміністрування групових чатів

Структуру програмного забезпечення для адміністрування групових чатів наведено у вигляді діаграми класів на рис. 3.1.

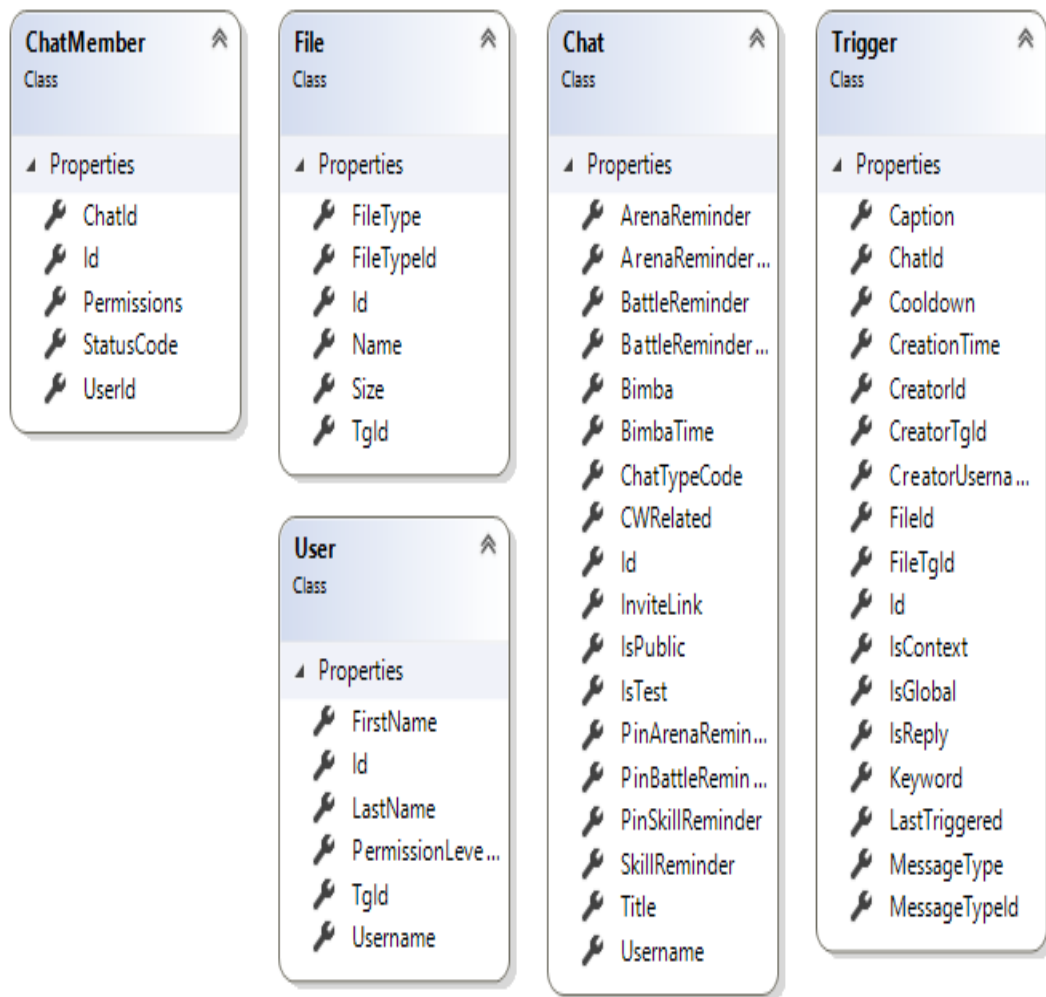


Рисунок 3.1 – Структура програмного забезпечення для адміністрування групових чатів у вигляді діаграми класів

Як видно з рис. 3.1, основними класами є чат, учасник чату, користувач та інші.

3.2 Функціонування програмного забезпечення для адміністрування групових чатів

Функціонування програмного забезпечення для адміністрування групових чатів подамо за допомогою схеми у вигляді діаграми кооперацій, зображеної на рис. 3.2.

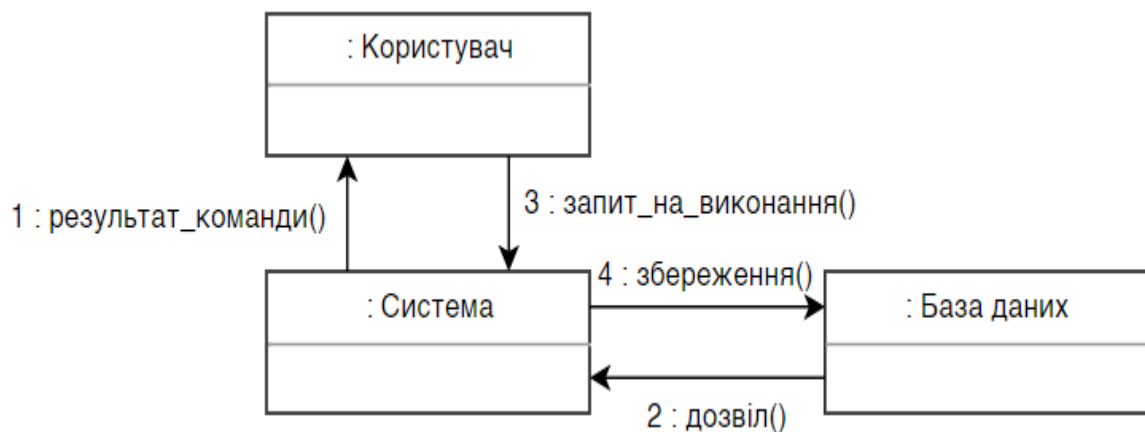


Рисунок 3.2 – Схема функціонування у вигляді діаграми кооперацій програмного забезпечення для адміністрування групових чатів

Користувач взаємодіє з інтерфейсом програмного забезпечення для адміністрування групових чатів, натискаючи на кнопки керування. У випадку відправки даних, на стороні клієнта програмного забезпечення для адміністрування групових чатів проводиться їх попередня перевірка на коректність перед відправкою на сервер. Сервер програмного забезпечення для адміністрування групових чатів приймає ці дані, обробляє їх і, за потреби, звертається до бази даних програмного забезпечення для адміністрування групових чатів. Після цього сервер програмного забезпечення для адміністрування групових чатів відправляє оброблені дані до користувача.

Діаграму діяльності програмного забезпечення для адміністрування групових чатів наведено на рис. 3.3.

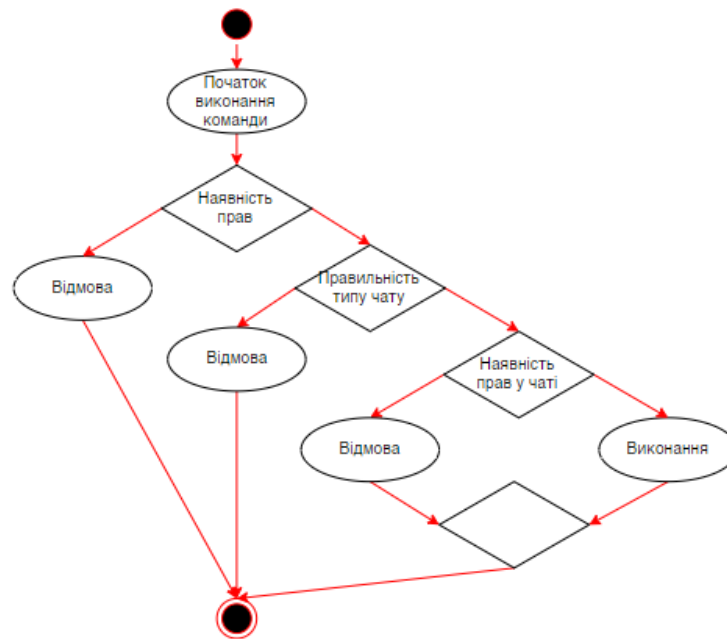


Рисунок 3.3 – Діаграма діяльності програмного забезпечення для адміністрування групових чатів

Діаграму послідовностей програмного забезпечення для адміністрування групових чатів наведено на рис. 3.4.



Рисунок 3.4 – Діаграма послідовностей програмного забезпечення для адміністрування групових чатів

3.3 Розробка бази даних програмного забезпечення для адміністрування групових чатів

База даних програмного забезпечення для адміністрування групових чатів призначена для зберігання даних про чати, користувачів, передану інформацію та ін.

Схему бази даних програмного забезпечення для адміністрування групових чатів наведено на рис. 3.5.

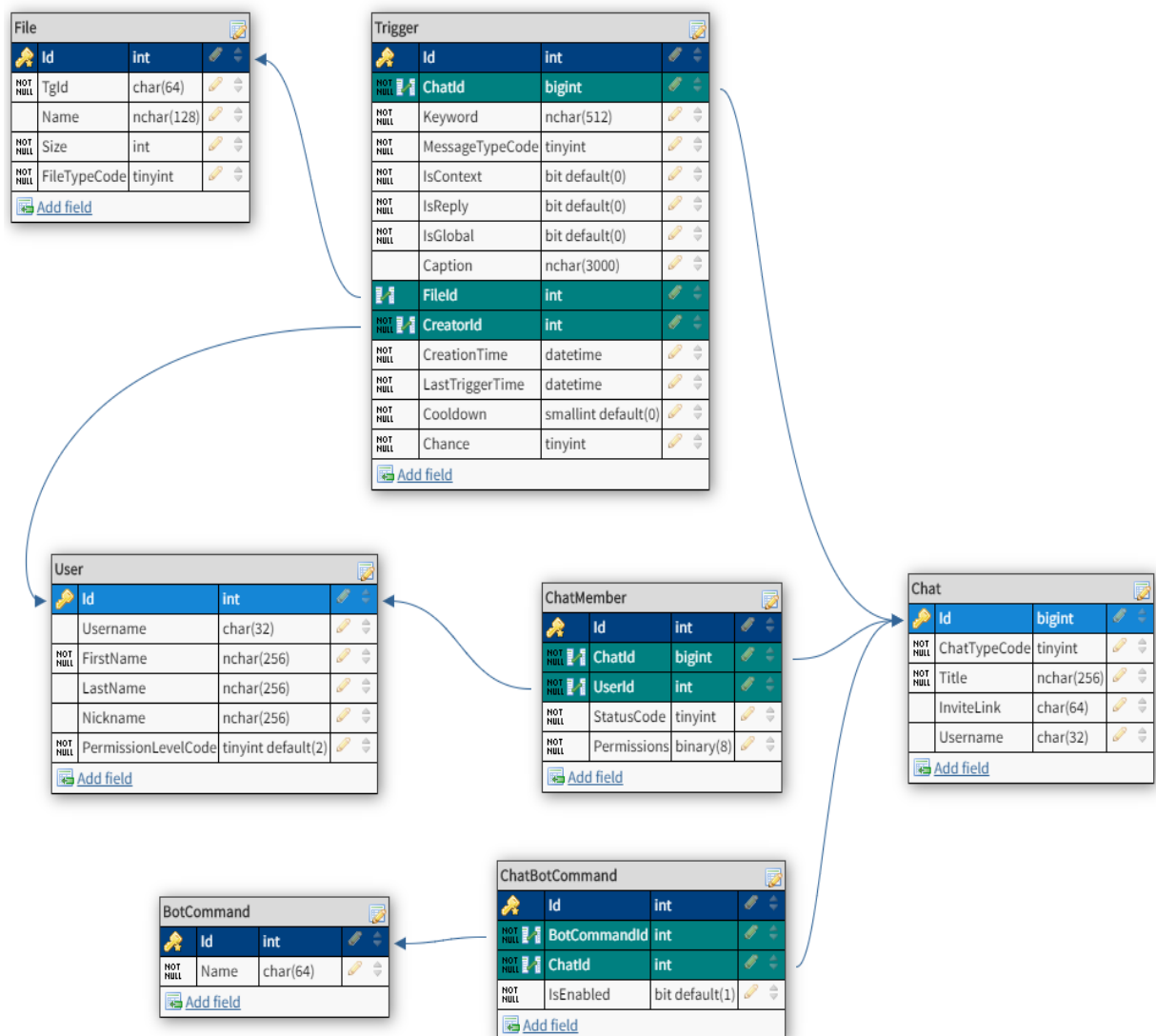


Рисунок 3.5 – Схема бази даних програмного забезпечення для адміністрування групових чатів

Схема бази даних програмного забезпечення для адміністрування групових чатів відображає концептуальні зв'язки між обраними елементами у відповідній предметній області. Основними сутностями бази даних обрано такі: чат (Chat), користувач (User), тригер чату (Trigger) та інші. Тригером чату є автоматизована подія або умова, яка запускає певну дію в чаті. Вони використовуються для створення динамічних чатів, які можуть автоматично реагувати на певні умови або дії користувачів.

3.4 Проєктування інтерфейсу програмного забезпечення для адміністрування групових чатів

Проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для адміністрування групових чатів виконано з урахуванням вимог технічного завдання. При розробленні програмного забезпечення для адміністрування групових чатів враховані функціональні вимоги до програми:

- можливість надання інформації щодо чату;
- можливість отримання інформації про користувачів;
- підтримка можливості додавання та видалення користувачів чату;
- підтримка можливості блокування деяких користувачів;
- підтримка можливості керування каналами чатів;
- підтримка можливості зміни налаштувань чату.

Інтерфейс головної форми програмного забезпечення для адміністрування групових чатів наведено на рис. 3.6.

Форму додавання користувача наведено на рис. 3.7.

Панель адміністратора групових чатів

Користувачі	Канали	Налаштування
Додати користувача	Додати канал	Загальні
Видалити користувача	Видалити канал	Безпека
Змінити права	Налаштування каналу	Сповіщення

Інформація про груповий чат

Назва чату: Проєкт з розробки бази даних.

Кількість учасників: 23.

Дата створення: 05.02.2024.

Загальна кількість повідомлень: 218.

Рисунок 3.6 – Інтерфейс головної форми програмного забезпечення для адміністрування групових чатів

Додати користувача

Ім'я користувача:

Електронна пошта:

Роль у чаті:

Додати

Скасувати

Рисунок 3.7 – Форма додавання користувача

Форма зміни прав користувача наведено на рис. 3.8.

Зміна прав користувача

Ім'я користувача:	Іван Іванов
Електронна пошта:	nov@xam.com
Поточна роль:	Член
Нова роль:	Адміністратор
Зберегти Скасувати	

Рисунок 3.8 – Форма зміни прав користувача

Форму додавання каналу чату наведено на рис. 3.9.

Додавання каналу чату

Назва каналу:	Новий канал
Опис:	Це канал для обговорення нових проектів
Тип каналу:	Приватний
Додати Скасувати	

Рисунок 3.9 – Форма додавання каналу чату

Форму загальних налаштувань чату наведено на рис. 3.10.

Зміна загальних налаштувань чату

Назва чату:	Командний чат	
Опис:	Чат для обговорення проектів команди	
Максимальна кількість учасників:	150	
Тема чату за замовчуванням:	Проекти	
Відкритий чат:		Так
Сповіщення:		Так
	Зберегти Скасувати	

Рисунок 3.10 – Форма загальних налаштувань чату

Форму налаштувань сповіщень чату наведено на рис. 3.11.

Налаштування сповіщень чату	
Сповіщення про нові повідомлення	Так
Сповіщення про згадки	Так
Звукові сповіщення	Ні
Сповіщення на електронну пошту	Так
Щоденний дайджест	Так
Зберегти Скасувати	

Рисунок 3.11 – Форма налаштувань сповіщень чату

3.5 Висновки за розділом 3

Розроблено програмне забезпечення для адміністрування групових чатів.

Запропоновано структуру програмного забезпечення для адміністрування групових чатів. Визначено, що основними класами є чат, учасник чату, користувач та інші.

Розроблено базу даних програмного забезпечення для адміністрування групових чатів, яка відображає концептуальні зв'язки між обраними елементами у відповідній предметній області. Основними сутностями бази даних обрано такі: чат (Chat), користувач (User), тригер чату (Trigger) та інші.

Описано функціонування програмного забезпечення для адміністрування групових чатів.

Описано особливості реалізації програмного забезпечення для адміністрування групових чатів. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для адміністрування групових чатів.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Призначення й умови застосування програми

Програма призначена для адміністрування групових чатів. Програма написана на мові C#, яка має вбудовані механізми безпеки, зокрема, засоби управління автентифікацією та авторизацією користувачів, засоби захист даних та інші функції. В якості середовища розробки для створення програмного забезпечення для адміністрування групових чатів обрано Microsoft Visual Studio, яке пропонує розширений редактор коду, засоби авто доповнення, рефакторингу та інші інструменти, які значно полегшують написання, читання та підтримку коду.

Програмне забезпечення для адміністрування групових чатів забезпечує виконання таких функцій:

- можливість надання інформації щодо чату;
- можливість отримання інформації про користувачів;
- підтримка можливості додавання та видалення користувачів чату;
- підтримка можливості блокування деяких користувачів;
- підтримка можливості керування каналами чатів;
- підтримка можливості зміни налаштувань чату.

Для експлуатації програмного забезпечення для адміністрування групових чатів необхідно таке апаратне та програмне забезпечення: два гігабайти вільного диску на вінчестері комп'ютера, оперативна пам'ять об'ємом не менше чотирьох гігабайтів, процесор із тактовою частотою не менше 2.2 ГГц, а також встановлена операційна система та веббраузер.

Функціональні характеристики розробленого програмного забезпечення для адміністрування групових чатів наведено у технічному завданні (додаток А). Фрагмент тексту програмного забезпечення для адміністрування групових чатів наведено у додатку Б.

4.2 Характеристики програми для адміністрування групових чатів

Програмне забезпечення для адміністрування групових чатів засновано на архітектурі Модель-Вид-Контролер. Всі файли програмного забезпечення для адміністрування групових чатів у директорії «Controllers» відповідають за обробку користувацьких запитів. Файли програмного забезпечення для адміністрування групових чатів у директорії «Models» описують моделі, забезпечують виконання запитів до бази даних та реалізують логіку функціонування програми. Файли та паки в директорії «Views» програмного забезпечення для адміністрування групових чатів відповідають назві відповідного контролера, а файли всередині них відповідають методам. Також використовуються файли з розширенням .cshtml, які служать для генерації розмітки сторінок.

4.3 Інструкція по експлуатації програми

4.3.1 Звернення до програми

Для запуску програмного забезпечення для адміністрування групових чатів на серверній частини необхідно запустити відповідний файл.

4.3.2 Вхідні й вихідні дані

Вхідними даними до програмного забезпечення для адміністрування групових чатів є інформація про користувачів чату, повідомлення, необхідні налаштування та ін.

Вихідними даними програмного забезпечення для адміністрування групових чатів є створені у базі даних записи про налаштування чату, користувачів та ін.

4.3.3 Повідомлення

Користувач програмного забезпечення для адміністрування групових чатів може отримати такі повідомлення: повідомлення про успішне додавання нового члена чату, повідомлення про зміну налаштувань чату та ін.

4.4 Виконання програмного забезпечення для адміністрування групових чатів

Після запуску програмного забезпечення для адміністрування групових чатів користувачу відображається головна сторінка (рис. 4.1), що являє собою панель адміністратора групових чатів. На головній формі розташовано блоки для роботи з користувачами (додавання, видалення, зміна прав користувачів), каналами чатів та налаштуваннями чатів.

Панель адміністратора групових чатів

Користувачі	Канали	Налаштування
Додати користувача	Додати канал	Загальні
Видалити користувача	Видалити канал	Безпека
Змінити права	Налаштування каналу	Сповіднення

Інформація про груповий чат

Назва чату: Проект з розробки бази даних.
Кількість учасників: 23.
Дата створення: 05.02.2024.
Загальна кількість повідомлень: 218.

Рисунок 4.1 – Інтерфейс головної форми програмного забезпечення для адміністрування групових чатів

Для додавання користувача необхідно натиснути на кнопку Додати користувача на головній формі та ввести його дані (рис. 4.2).

Додати користувача

Ім'я користувача:

Електронна пошта:

Роль у чаті:

Рисунок 4.2 – Форма додавання користувача

Для зміни прав користувача у чаті необхідно натиснути відповідну кнопку та у формі, що з'явилася (рис. 4.3), обрати відповідну роль.

Зміна прав користувача

Ім'я користувача:

Електронна пошта:

Поточна роль:

Нова роль:

Рисунок 4.3 – Форма зміни прав користувача

Для створення нового каналу чату необхідно в області Канали панелі адміністратора натиснути кнопку Додати канал, потім ввести інформацію про канал та натиснути кнопку Додати (рис. 4.4).

Додавання каналу чату

Назва каналу:	Новий канал
Опис:	Це канал для обговорення нових проектів
Тип каналу:	Приватний
Додати Скасувати	

Рисунок 4.4 – Форма додавання каналу чату

Форма загальних налаштувань чату дозволяє обмежити максимально допустиму кількість учасників чату, змінити опис каналу та внести іншу інформацію про чат (рис. 4.5).

Зміна загальних налаштувань чату

Назва чату:	Командний чат	
Опис:	Чат для обговорення проектів команди	
Максимальна кількість учасників:	150	
Тема чату за замовчуванням:	Проекти	
Відкритий чат:		Так
Сповіщення:		Так
Зберегти Скасувати		

Рисунок 4.5 – Форма загальних налаштувань чату

Форму налаштувань сповіщень чату наведено на рис. 4.6.

Налаштування сповіщень чату

Сповіщення про нові повідомлення	Так
Сповіщення про згадки	Так
Звукові сповіщення	Ні
Сповіщення на електронну пошту	Так
Щоденний дайджест	Так
Зберегти Скасувати	

Рисунок 4.6 – Форма налаштувань сповіщень чату

4.5 Тестування програмного забезпечення для адміністрування групових чатів

Виконано тестування програмного забезпечення для адміністрування групових чатів.

Виявлено, що програма для підтримки процесу адміністрування групових чатів функціонує правильно та злагоджено. Вона реалізує всі функціональні вимоги та успішно виконує свою основну задачу підтримки процесу адміністрування групових чатів.

Розроблене програмне забезпечення дозволяє забезпечувати автоматизацію процесів, пов'язаних з підтримкою процесу адміністрування групових чатів.

4.6 Висновки за розділом 4

Описано програмне забезпечення для адміністрування групових чатів. Виконано тестування розробленого програмного забезпечення для адміністрування групових чатів. Результати тестування програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з підтримкою процесу адміністрування групових чатів.

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи магістра було проаналізовано та досліджено процес розробки програмного забезпечення для адміністрування чатів.

Визначено, що чати являють собою програмні інструменти, що дозволяють людям спілкуватися в режимі реального часу, вони сприяють взаємодії між людьми, які можуть бути географічно віддаленими один від одного. Відзначено, що програмні засоби, які являють собою чат-платформи, складаються з віртуальних кімнат, де користувачі можуть надсилати текстові, голосові або відео повідомлення, такі програми дозволяють миттєво спілкуватися, забезпечуючи безперервний обмін інформацією навіть на великих відстанях. Проаналізовано класифікацію чатів. Програмні засоби для підтримки функціонування чатів, як правило, дозволяють надсилати текстові, голосові або відео повідомлення, а також пропонують різноманітні формати для задоволення різних потреб користувачів у спілкуванні. Проаналізовано особливості та функції програмних платформ для чату.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для підтримки групових чатів. Проте деякі програми для підтримки групових чатів не забезпечують достатній рівень шифрування, що може стати проблемою для компаній, які обробляють чутливі дані. Інше програмне забезпечення для підтримки адміністрування групових чатів може мати проблеми з інтеграцією і бути несумісним з програмами, що вже використовуються в організації. Деякі програми для адміністрування чатів можуть не забезпечувати достатній рівень функціональності. Тому актуальною є розробка програмного забезпечення для адміністрування групових чатів. Тому актуальною є розробка програмного забезпечення для адміністрування групових чатів.

Сформульовано функціональні вимоги до програмного забезпечення для адміністрування групових чатів.

Для реалізації програмного забезпечення для адміністрування групових чатів обрано мову програмування C#, яка має вбудовані механізми безпеки, зокрема, засоби управління автентифікацією та авторизацією користувачів, засоби захист даних та інші функції, що важливо для захисту конфіденційної інформації чату та збереження даних користувачів. Крім того, C# є об'єктно-орієнтованою мовою програмування, що дозволяє організувати код у вигляді класів і об'єктів, полегшуючи його підтримку, розширюваність та повторне використання.

Для створення програмного забезпечення для адміністрування групових чатів обрано середовище розробки Microsoft Visual Studio, яке пропонує розширений редактор коду, засоби авто доповнення, рефакторингу та інші інструменти, які значно полегшують написання, читання та підтримку коду. Крім того, середовище розробки Microsoft Visual Studio забезпечує інтеграцію з .NET Framework, що дозволяє розробникам ефективно використовувати можливості цих платформ та розробляти програмні продукти мовою C#.

Новизна роботи полягає у тому, що створено модель взаємодії користувача з програмним забезпеченням для адміністрування групових чатів, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення для адміністрування групових чатів.

Практична цінність результатів роботи полягає у тому, що розроблено програмне забезпечення для адміністрування групових чатів.

Запропоновано структуру програмного забезпечення для адміністрування групових чатів. Визначено, що основними класами є чат, учасник чату, користувач та інші.

Розроблено базу даних програмного забезпечення для адміністрування групових чатів, яка відображає концептуальні зв'язки між обраними елементами у відповідній предметній області. Основними сутностями бази даних обрано такі: чат (Chat), користувач (User), тригер чату (Trigger) та інші.

Описано функціонування програмного забезпечення для адміністрування групових чатів.

Описано особливості реалізації програмного забезпечення для адміністрування групових чатів. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для адміністрування групових чатів.

Виконано тестування розробленого програмного забезпечення для адміністрування групових чатів. Результати тестування програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з підтримкою процесу адміністрування групових чатів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. The Ultimate Guide – What is Chat and How Does it Work [Electronic resource]. – Access mode: <https://chatableapps.com/technology/the-ultimate-guide-what-is-chat-and-how-does-it-work/>.
2. What is Chat [Electronic resource]. – Access mode: <https://www.arimetrics.com/en/digital-glossary/chat>.
3. Chat [Electronic resource]. – Access mode: <https://www.computerhope.com/jargon/c/chat.htm>.
4. The ultimate guide to live chat support [Electronic resource]. – Access mode: <https://www.intercom.com/blog/live-chat-guide/>.
5. Online chat [Electronic resource]. – Access mode: https://en.wikipedia.org/wiki/Online_chat.
6. What is ChatGPT and why does it matter [Electronic resource]. – Access mode: <https://www.zdnet.com/article/what-is-chatgpt-and-why-does-it-matter-heres-everything-you-need-to-know/>.
7. The 29 Best Live Chat Software & Services of the Year [Electronic resource]. – Access mode: <https://blog.hubspot.com/sales/best-live-chat-software>.
8. Best Live Chat Software for Websites [Electronic resource]. – Access mode: <https://www.tidio.com/blog/best-live-chat-software/>.
9. Best Live Chat Software [Electronic resource]. – Access mode: <https://www.forbes.com/advisor/business/software/best-live-chat-software/>.
10. Acquire [Electronic resource]. – Access mode: <https://www.acquire.io/>.
11. Pure Chat [Electronic resource]. – Access mode: <https://www.purechat.com/>.
12. Tidio [Electronic resource]. – Access mode: <https://www.tidio.com/>.
13. Características de C# [Electronic resource]. – Access mode: <https://desarrolloweb.com/home/c>.
14. Dynamic C# 7 Fundamentals [Electronic resource]. – Access mode: <https://www.pluralsight.com/courses/c-sharp-dynamic-fundamentals>.

15. User Interface [Electronic resource]. – Access mode:
<https://code.visualstudio.com/docs/getstarted/userinterface>.

16. Copilot intégré à Visual Studio [Electronic resource]. – Access mode:
<https://visualstudio.microsoft.com/fr/>.

ДОДАТОК А
Технічне завдання

Вступ

Програмне забезпечення може використовуватися для підтримки процесу адміністрування групових чатів.

A.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Дослідження та програмна реалізація засобів адміністрування групових чатів», затверджене наказом Національного університету «Запорізька політехніка» № 398 від 07 жовтня 2024 р.

A.2 Призначення розробки

Програмний продукт призначений для адміністрування групових чатів.

A.3 Основні вимоги до програми, що розробляється

A.3.1 Вимоги до функціональних характеристик

Програмне забезпечення для підтримки процесу адміністрування групових чатів забезпечує виконання таких функцій:

- можливість надання інформації щодо чату;
- можливість отримання інформації про користувачів;
- підтримка можливості додавання та видалення користувачів чату;
- підтримка можливості блокування деяких користувачів;
- підтримка можливості керування каналами чатів;
- підтримка можливості зміни налаштувань чату.

А.3.2 Вимоги до інтерфейсу програми

Інтерфейс програмного забезпечення для адміністрування групових чатів повинен бути зручним для користувачів. Повинна бути забезпечена можливість роботи в візуальному режимі.

А.3.3 Вимоги до надійності

Програмне забезпечення для адміністрування групових чатів повинно забезпечити надійне функціонування.

А.3.4 Умови експлуатації

Для експлуатації програмного забезпечення для підтримки процесу адміністрування групових чатів необхідна наявність персонального комп'ютера.

А.3.5 Вимоги до складу та параметрів технічний засобів

Для експлуатації програмного забезпечення для адміністрування групових чатів необхідно таке апаратне та програмне забезпечення: два гігабайти вільного диску на вінчестері комп'ютера, оперативна пам'ять об'ємом не менше чотирьох гігабайтів, процесор із тактовою частотою не менше 2.2 ГГц, а також встановлена операційна система та веббраузер.

A.3.6 Вимоги до маркування і пакування

Програма для підтримки процесу адміністрування групових чатів може бути записана на будь-якому носії інформації.

На пакуванні повинна бути назва програми – «Програмне забезпечення для підтримки процесу адміністрування групових чатів».

A.4 Вхідні дані до роботи

Вхідними даними до програмного забезпечення для адміністрування групових чатів є інформація про користувачів чату, повідомлення, необхідні налаштування та ін.

Вихідними даними програмного забезпечення для адміністрування групових чатів є створені у базі даних записи про налаштування чату, користувачів та ін.

ДОДАТОК Б
Фрагмент тексту програми

```

public class Db
{
    public static string KnktnStr =
Environment.GetEnvironmentVariable("SDDFrameBot_db_knktn");

    public bool VldSql(string sql)
    {
        using (var knktn = new SqlKnktn(KnktnStr))
        using (var kmnd = new SqlKmnd("SET NOEXEC ON;" + sql, knktn))
        {
            knktn.Open();
            try
            {
                kmnd.ExecuteNonQuery();
            }
            catch
            {
                return false;
            }
        }

        return true;
    }

    public lst<string> ExctSql(string sql)
    {
        var zapss = new lst<string>();

        string frstKwrD = sql.Substring(0, sql.IndexOf(' ')).ToUpper();

        switch (frstKwrD)
        {
            case "SELECT":
                using (var knktn = new SqlKnktn(KnktnStr))
                using (var kmnd = new SqlKmnd(sql, knktn))
                {
                    knktn.Open();
                    using (var rdr = kmnd.ExecuteRdr())
                    {
                        if (rdr.HasRows)
                        {
                            while (rdr.Read())

```

```

        {
            string zaps = "";
            for (int i = 0; i < rdr.FieldCount; i++)
            {
                string lkk =
rdr.GetValue(i).ToString().TrimEnd() + " | ";
                zaps += lkk;
            }
            zapss.Add(zaps);
        }
    }
}
break;
case "ALTER":
    using (var knktn = new SqlKnktn(KnktnStr))
    using (var kmnd = new SqlKmnd(sql, knktn))
    {
        knktn.Open();

        var affctd = kmnd.ExecuteNonQuery();
        zapss.Add("Rows affctd: " + affctd.ToString());
    }
    break;
default:
    zapss.Add($"{firstKwrd} error /sql");
    break;
}

return zapss;
}

public void UpdateKrtg(Krtg krtg)
{
    string sql = @"
UPDATE [dbo].[krtg] SET krtgimrt = @krtgimrt, frst_imrt = @frst_imrt, lst_imrt
= @lst_imrt
WHERE tgd = @tgd
IF @@RWKNT=0
INSERT INTO [dbo].[krtg] (tgd, krtgimrt, frst_imrt, lst_imrt)
VLS (@tgd, @krtgimrt, @frst_imrt, @lst_imrt);";

    var tgdPrm = new SqlPrmeter("@tgd", krtg.Id);

```

```

        var krtgimrtPrm = new SqlPrmeter("@krtgimrt",
!string.IsNullOrEmpty(krtg.Krtgimrt) ? krtg.Krtgimrt : (object)DBNull.Value);
        var frstImrtPrm = new SqlPrmeter()
        {
            PrmeterImrt = "@frst_imrt",
            SqlDTp = SqlDTp.NChar,
            Znch = krtg.FrstImrt
        };
        var lstImrtPrm = new SqlPrmeter()
        {
            PrmeterImrt = "@lst_imrt",
            SqlDTp = SqlDTp.NChar,
            Znch = !string.IsNullOrEmpty(krtg.LstImrt) ? krtg.LstImrt :
(object)DBNull.Value
        };

        using (var knktn = new SqlKnktn(KnktnStr))
        using (var kmnd = new SqlKmnd(sql, knktn))
        {
            knktn.Open();

            kmnd.Prmeters.Add(tgdPrm);
            kmnd.Prmeters.Add(krtgimrtPrm);
            kmnd.Prmeters.Add(frstImrtPrm);
            kmnd.Prmeters.Add(lstImrtPrm);

            kmnd.ExecuteNonQuery();
        }
    }

    public lst<Types.Krtg> GetAllKrtgs()
    {
        string sql = $"SELECT id, tgd, krtgimrt, frst_imrt, lst_imrt,
Perms_level_code FROM [dbo].[krtg]";

        var krtgs = new lst<Types.Krtg>();

        using (var knktn = new SqlKnktn(KnktnStr))
        using (var kmnd = new SqlKmnd(sql, knktn))
        {
            knktn.Open();
            using (var rdr = kmnd.ExecuteRdr())
            {

```

```

        if (rdr.HasRows)
        {
            while (rdr.Read())
            {
                var krtg = new Types.Krtg()
                {
                    Id = rdr.GetInt32(0),
                    Tgd = rdr.GetInt32(1),
                    FrstImrt = rdr.GetString(3).TrimEnd(),
                    PermLvlId = rdr.GetInt32(5)
                };

                if (!rdr.IsDBNull(2))
                    krtg.Krtgimrt = rdr.GetString(2).TrimEnd();

                if (!rdr.IsDBNull(4))
                    krtg.LstImrt = rdr.GetString(4).TrimEnd();

                krtgs.Add(krtg);
            }
        }
    }

    return krtgs;
}

public Types.Krtg GetKrtgById(int id)
{
    string sql = $"SELECT id, tgd, krtgimrt, frst_imrt, lst_imrt,
    Perms_level_code FROM [dbo].[krtg] WHERE id = {id};";

    Types.Krtg krtg = null;

    using (var knktn = new SqlKnktn(KnktnStr))
    using (var kmnd = new SqlKmnd(sql, knktn))
    {
        knktn.Open();
        using (var rdr = kmnd.ExecuteRdr())
        {
            if (rdr.HasRows)
            {
                rdr.Read();
            }
        }
    }
}

```

```

        krtg = new Types.Krtg()
        {
            Id = rdr.GetInt32(0),
            Tgd = rdr.GetInt32(1),
            FrstImrt = rdr.GetString(3).TrimEnd(),
            PermLvlId = rdr.GetInt32(5)
        };

        if (!rdr.IsDBNull(2))
            krtg.Krtgimrt = rdr.GetString(2).TrimEnd();

        if (!rdr.IsDBNull(4))
            krtg.LstImrt = rdr.GetString(4).TrimEnd();
    }
}

return krtg;
}

public Types.Krtg GetKrtgByTgd(int tgd)
{
    string sql = $"SELECT id, tgd, krtgimrt, frst_imrt, lst_imrt,
    Perms_level_code FROM [dbo].[krtg] WHERE tgd = {tgd}";

    Types.Krtg krtg = null;

    using (var knktn = new SqlKnktn(KnktnStr))
    using (var kmnd = new SqlKmnd(sql, knktn))
    {
        knktn.Open();
        using (var rdr = kmnd.ExecuteRdr())
        {
            if (rdr.HasRows)
            {
                rdr.Read();

                krtg = new Types.Krtg()
                {
                    Id = rdr.GetInt32(0),
                    Tgd = rdr.GetInt32(1),
                    FrstImrt = rdr.GetString(3).TrimEnd(),

```

```

        PermLvlId = rdr.GetInt32(5)
    };

    if (!rdr.IsDBNull(2))
        krtg.Krtgimrt = rdr.GetString(2).TrimEnd();

    if (!rdr.IsDBNull(4))
        krtg.LstImrt = rdr.GetString(4).TrimEnd();
    }
}

return krtg;
}

public Types.Krtg GetKrtgByGuildAndHero(string heroImrt, string
guildTag = null)
{
    return new Types.Krtg();
}

public void UpdateCkat(Ckat ckat)
{
    string sql = @"
UPDATE [dbo].[ckat] SET ckat_type_code = @ckat_type_code, ttg = @ttg, krtgimrt
= @krtgimrt, ispblc = @ispblc
WHERE id = @id
IF @@RWKNT=0
INSERT INTO [dbo].[ckat] (id, ckat_type_code, ttg, krtgimrt, ispblc)
VLS (@id, @ckat_type_code, @ttg, @krtgimrt, @ispblc);";

    var idPrm = new SqlParameter("@id", ckat.Id);
    var ckatTpPrm = new SqlParameter("@ckat_type_code", (int)ckat.Type);
    var ttgPrm = new SqlParameter()
    {
        PrmeterImrt = "@ttg",
        SqlDTP = SqlDTP.NChar,
        Znch = ckat.Ttg
    };

    var krtgimrtPrm = new SqlParameter("@krtgimrt",
!string.IsNullOrEmpty(ckat.Krtgimrt) ? ckat.Krtgimrt : (object)DBNull.Value);
    var ispblcPrm = new SqlParameter("@ispblc",
!string.IsNullOrEmpty(ckat.Krtgimrt) ? true : false);

```

```

using (var knktn = new SqlKnktn(KnktnStr))
using (var kmnd = new SqlKmnd(sql, knktn))
{
    knktn.Open();

    kmnd.Prmeters.Add(idPrm);
    kmnd.Prmeters.Add(ckatTpPrm);
    kmnd.Prmeters.Add(ttgPrm);
    kmnd.Prmeters.Add(krtgimrtPrm);
    kmnd.Prmeters.Add(isPblcPrm);

    kmnd.ExecuteNonQuery();
}

}

public lst<Ckat> GetAllCkats()
{
    string sql = "SELECT id, ttg FROM [dbo].[ckat]";

    var ckats = new lst<Ckat>();

    using (var knktn = new SqlKnktn(KnktnStr))
    using (var kmnd = new SqlKmnd(sql, knktn))
    {
        knktn.Open();

        using (var rdr = kmnd.ExecuteRdr())
        {
            while (rdr.Read())
            {
                var ckat = new Ckat
                {
                    Id = rdr.GetInt64(0),
                    Ttg = rdr.GetString(1).TrimEnd()
                };

                ckats.Add(ckat);
            }
        }
    }

    return ckats;
}

```



```

        Ttg = rdr.GetString(1).TrimEnd(),
        InvtLnk          =          !rdr.IsDBNull(2)          ?
rdr.GetString(2).TrimEnd() : null,
        Krtgimrt          =          !rdr.IsDBNull(3)          ?
rdr.GetString(3).TrimEnd() : null,
        IsPblc = rdr.GetBoolean(4),
        Bmbd = rdr.GetBoolean(5),
        BmbdTime = !rdr.IsDBNull(6) ? rdr.GetDateTime(7) :
DateTime.MinValue,
        CWRltd = rdr.GetBoolean(7),
        ArnRemnder = rdr.GetBoolean(8),
        PinArnRemnder = rdr.GetBoolean(9),
        ArnRemnderText =          !rdr.IsDBNull(10)          ?
rdr.GetString(11).TrimEnd() : null,
        BtlRemndr = rdr.GetBoolean(11),
        PinBtlRemndr = rdr.GetBoolean(12),
        BtlRemndrText =          !rdr.IsDBNull(13)          ?
rdr.GetString(14).TrimEnd() : null,
        SklRmndr = rdr.GetBoolean(14),
        PinSklRmndr = rdr.GetBoolean(15),
        IsTst = rdr.GetBoolean(16)
    };
}
}
}

return ckat;
}

public lst<Types.Ckat> GetTestCkats()
{
    string sql = $"{@"
SELECT
    ckat_type_code,
    ttg,
    invtlnk,
    krtgimrt,
    ispblc,
    bmbd,
    bmbd_time,
    cwrl,
    arnrminder,
    pin_arnrminder,

```

```

    arnrmindertxt,
    bttlerminder,
    pin_bttlerminder,
    bttlermindertxt,
    skllrminder,
    pin_skllrminder,
    id
FROM [dbo].[ckat]
WHERE is_test = {1};"

```

```

    var lst = new lst<Types.Ckat>();

    using (var knktn = new SqlKnktn(KnktnStr))
    using (var kmnd = new SqlKmnd(sql, knktn))
    {
        knktn.Open();

        using (var rdr = kmnd.ExecuteRdr())
        {
            while (rdr.Read())
            {
                lst.Add(new Types.Ckat()
                {
                    CkatTpCode = rdr.GetInt32(0),
                    Ttg = rdr.GetString(1).TrimEnd(),
                    InvtnLnk = !rdr.IsDBNull(2) ?
rdr.GetString(2).TrimEnd() : null,
                    Krtgimrt = !rdr.IsDBNull(3) ?
rdr.GetString(3).TrimEnd() : null,
                    IsPblc = rdr.GetBoolean(4),
                    Bmbd = rdr.GetBoolean(5),
                    BmbdTime = !rdr.IsDBNull(6) ? rdr.GetDateTime(7) :
DateTime.MinValue,
                    CWRltd = rdr.GetBoolean(7),
                    ArnRemnder = rdr.GetBoolean(8),
                    PinArnRemnder = rdr.GetBoolean(9),
                    ArnRemnderText = !rdr.IsDBNull(10) ?
rdr.GetString(11).TrimEnd() : null,
                    BtlRemndr = rdr.GetBoolean(11),
                    PinBtlRemndr = rdr.GetBoolean(12),
                    BtlRemndrText = !rdr.IsDBNull(13) ?
rdr.GetString(14).TrimEnd() : null,
                    SklRmndr = rdr.GetBoolean(14),

```

```

        PinSk1Rmndr = rdr.GetBoolean(15),
        Id = rdr.GetInt64(16),
        IsTst = true
    });
    }
}

return lst;
}

public lst<Types.Ckat> GetArnRemnderCkats()
{
    string sql = @"
SELECT id, pin_arnrminder, arnrmindertxt
FROM [dbo].[ckat]
WHERE arnrminder = {1};"

    var lst = new lst<Types.Ckat>();

    using (var knktn = new SqlKnktn(KnktnStr))
    using (var kmnd = new SqlKmnd(sql, knktn))
    {
        knktn.Open();

        using (var rdr = kmnd.ExecuteRdr())
        {
            while (rdr.Read())
            {
                lst.Add(new Types.Ckat()
                {
                    Id = rdr.GetInt64(0),
                    ArnRemnder = true,
                    PinArnRemnder = rdr.GetBoolean(1),
                    ArnRemnderText = !rdr.IsDBNull(2)
                        ? rdr.GetString(2).TrimEnd() : null
                });
            }
        }
    }

    return lst;
}

```

```

        public lst<Types.Ckat> GetBtlRemndrCkats()
        {
            string sql = @"
SELECT id, pin_bttlerminder, bttlermindertxt
FROM [dbo].[ckat]
WHERE bttlerminder = {1}";

            var lst = new lst<Types.Ckat>();

            using (var knktn = new SqlKnktn(KnktnStr))
            using (var kmnd = new SqlKmnd(sql, knktn))
            {
                knktn.Open();

                using (var rdr = kmnd.ExecuteRdr())
                {
                    while (rdr.Read())
                    {
                        lst.Add(new Types.Ckat()
                        {
                            Id = rdr.GetInt64(0),
                            BtlRemndr = true,
                            PinBtlRemndr = rdr.GetBoolean(1),
                            BtlRemndrText = !rdr.IsDBNull(2) ?
rdr.GetString(2).TrimEnd() : null
                        });
                    }
                }

                return lst;
            }

            public lst<Types.Ckat> GetSkllRmndrCkats()
            {
                string sql = @"
SELECT id, pin_skllrminder
FROM [dbo].[ckat]
WHERE skllrminder = {1}";

                var lst = new lst<Types.Ckat>();

```

```

using (var knktn = new SqlKnktn(KnktnStr))
using (var kmnd = new SqlKmnd(sql, knktn))
{
    knktn.Open();

    using (var rdr = kmnd.ExecuteRdr())
    {
        while (rdr.Read())
        {
            lst.Add(new Types.Ckat()
            {
                Id = rdr.GetInt64(0),
                SklRmndr = true,
                PinSklRmndr = rdr.GetBoolean(1)
            });
        }
    }

    return lst;
}

public void SetCkatRmnds(long ckatId, lst<string> Rmnds, bool state)
{
    string vls = "";
    for (int i = 0; i < Rmnds.Count; i++)
    {
        switch (Rmnds[i])
        {
            case "arena":
                vls += $"arnrminder = {Convert.ToInt32(state)}";
                break;
            case "battle":
                vls += $"bttlerminder = {Convert.ToInt32(state)}";
                break;
            case "skill":
                vls += $"skllrminder = {Convert.ToInt32(state)}";
                break;
        }

        if (i != Rmnds.Count - 1)
            vls += ',';
    }
}

```

```
        string sql = $"{@"
UPDATE [dbo].[ckat] SET {vls}
WHERE id = {ckatId}";

        using (var knktn = new SqlKntn(KnktnStr))
        using (var kmnd = new SqlKmnd(sql, knktn))
        {
            knktn.Open();
            kmnd.ExecuteNonQuery();
        }
    }
```

ДОДАТОК В
Слайди презентації

Міністерство освіти і науки України
Національний університет «Запорізька політехніка»
Кафедра програмних засобів

Дослідження та програмна реалізація засобів адміністрування групових чатів

Виконав

ст. гр. КНТ-123м

Роман ПАХОМОВ

Керівник:

к.т.н., доцент

Матвій ІЛ'ЯШЕНКО

Рисунок В.1 – Слайд 1

Об'єкт, предмет та мета роботи

Об'єкт дослідження – процес розробки програмного забезпечення для адміністрування чатів.

Предмет дослідження – програмні засоби для підтримки процесу адміністрування групових чатів.

Мета роботи – дослідження та розробка програмного забезпечення для адміністрування групових чатів.

Рисунок В.2 – Слайд 2

Завдання роботи

Для досягнення поставленої мети необхідно було розв'язати такі завдання:

- виконати аналіз предметної області та програмних засобів для підтримки діяльності онлайн каси продажу квитків;
- розробити модель взаємодії користувача з програмним забезпеченням онлайн каси продажу квитків;
- здійснити проєктування програмного забезпечення онлайн каси продажу квитків;
- створити програмне забезпечення онлайн каси продажу квитків;
- виконати тестування розробленого програмного забезпечення онлайн каси продажу квитків.

3

Рисунок В.3 – Слайд 3

Аналіз програмного забезпечення для групових чатів

Критерій порівняння	Acquire	Pure Chat	Tidio
Можливості мультимедійної взаємодії	+	+-	+-
Багатомовна підтримка	+	-	-
Спільний перегляд	+	-	-
Доцільність використання організаціями різних розмірів	-	+-	+-
Засоби інтеграції	+	+	+-
Мобільний застосунок	-	+	+
Автоматизація та налаштування	+	+-	+

4

Рисунок В.4 – Слайд 4

Вибір мови програмування

Критерій порівняння мов програмування	Мова програмування		
	C#	PHP	JavaScript
Механізми безпеки даних	+	–	–
Зручність синтаксису	+	+–	+
Підтримка асинхронності	+	+–	–
Зручність для створення ПЗ для адміністрування групових чатів	+	–	–
Об'єктно-орієнтованість мови	+	+–	–

5

Рисунок В.5 – Слайд 5

Вибір середовища розробки

Критерій порівняння середовищ розробки	Середовища розробки		
	Microsoft Visual Studio	IntelliJ IDEA	JetBrains Rider
Розширений редактор коду	+	–	–
Зручна адміністративна панель	+	+–	+
Засоби інтеграції	+	+–	+–
Засоби авто доповнення	+	+–	+–
Зручність редактору коду	+	+	+–

6

Рисунок В.6 – Слайд 6

Модель взаємодії користувача з ПЗ для адміністрування групових чатів

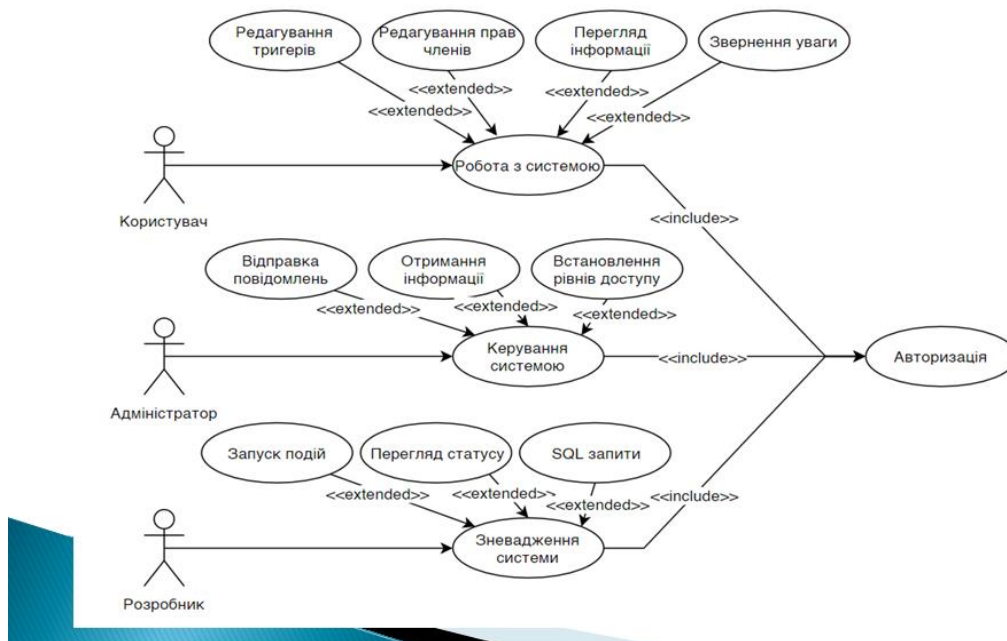


Рисунок В.7 – Слайд 7

Діаграма класів ПЗ для адміністрування групових чатів

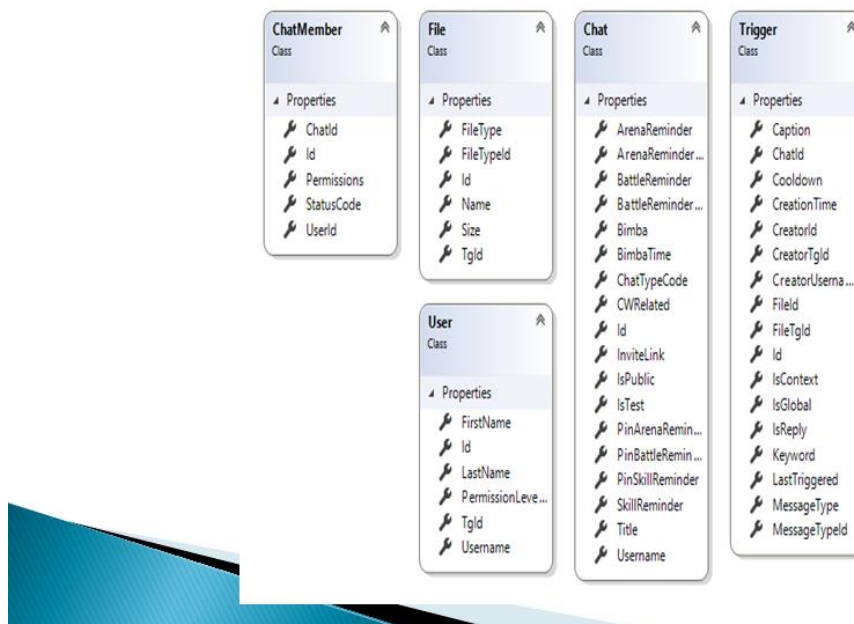
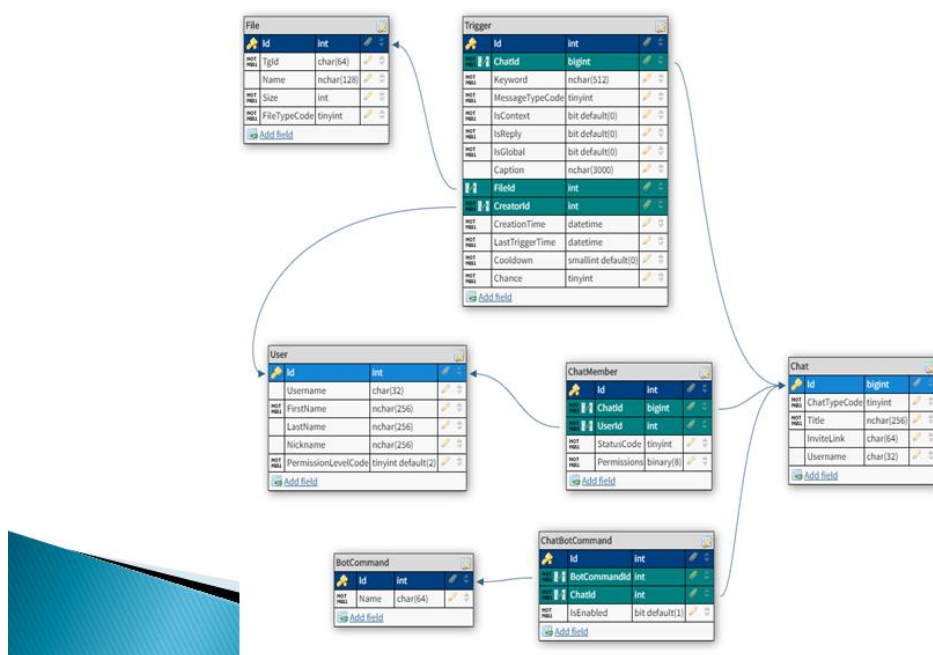


Рисунок В.8 – Слайд 8

Схема бази даних ПЗ для адміністрування групових чатів



11

Рисунок В.11 – Слайд 11

Робота з програмою. Голова форма

Панель адміністратора групових чатів

Користувачі	Канали	Налаштування
Додати користувача	Додати канал	Загальні
Видалити користувача	Видалити канал	Безпека
Змінити права	Налаштування каналу	Сповідження

Інформація про груповий чат

Назва чату: Проект з розробки бази даних.

Кількість учасників: 23.

Дата створення: 05.02.2024.

Загальна кількість повідомлень: 218.

12

Рисунок В.12 – Слайд 12

Робота з програмою. Додавання та зміна прав користувача

Додати користувача	Зміна прав користувача
Ім'я користувача:	Ім'я користувача: Іван Іванов
Електронна пошта:	Електронна пошта: nov@xam.com
Роль у чаті:	Поточна роль: Член
	Нова роль: Адміністратор
Додати Скасувати	Зберегти Скасувати



13

Рисунок В.13 – Слайд 13

Робота з програмою. Додавання каналу та зміна налаштувань чату

Додавання каналу чату	Зміна загальних налаштувань чату
Назва каналу: Новий канал	Назва чату: Командний чат
Опис: Це канал для обговорення нових проектів	Опис: Чат для обговорення проектів команди
Тип каналу: Приватний	Максимальна кількість учасників: 150
	Тема чату за замовчуванням: Проекти
Додати Скасувати	Відкритий чат: Так
	Словіщення: Так
	Зберегти Скасувати



14

Рисунок В.14 – Слайд 14

ВИСНОВКИ

У ході виконання дипломної кваліфікаційної роботи магістра було проаналізовано та досліджено процес розробки програмного забезпечення для адміністрування чатів.

Розв'язано всі поставлені завдання.

Новизна роботи полягає в тому, що створено модель взаємодії користувача з програмним забезпеченням для адміністрування групових чатів, яка подана у вигляді діаграми прецедентів, та дозволяє розроблювати програмне забезпечення для адміністрування групових чатів.

Практичне значення роботи полягає у тому, що розроблено програмне забезпечення для адміністрування групових чатів.

Результати тестування підтвердили, що програма може використовуватися за призначенням.

15

Рисунок В.15 – Слайд 15