

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)

бакалавр

(ступінь вищої освіти)

на тему ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ
ДЛЯ ШВИДКОГО СПІЛКУВАННЯ
SOFTWARE FOR QUICK COMMUNICATION

Виконав(ла): студент(ка) 4 курсу, групи КНТ-211

Спеціальності 122 Комп'ютерні науки

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Комп'ютерні науки

КАРМАЛИГА А.О.

(ПРИЗВИЩЕ та ініціали)

Керівник СКРУПСЬКИЙ С.Ю.

(ПРИЗВИЩЕ та ініціали)

Рецензент ІЛ'ЯШЕНКО М.Б.

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет КНТ
Кафедра програмних засобів
Ступінь вищої освіти бакалавр
Спеціальність 122 Комп'ютерні науки
(код і найменування)

Освітня програма (спеціалізація) Комп'ютерні науки
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН
“ ” 2025 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

КАРМАЛИГИ Артема Олександровича

(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Програмне забезпечення для швидкого спілкування.
Software for Quick Communication

керівник проєкту (роботи) к.т.н., доцент, СКРУПСЬКИЙ Степан Юрійович,
(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “ 28 ” квітня 2025 року № 209

2. Строк подання студентом проєкту (роботи) 02 червня 2025 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	СКРУПСЬКИЙ С.Ю., доцент		
Нормоконтроль	КАЛІНІНА М.В., асистент		

7. Дата видачі завдання “ 14 ” квітня 2025 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.	2 тиждень	Розділ 3
5	Розробка програми.	3-4 тижні	Розділи 3, 4
6	Тестування та експериментальне дослідження програмного забезпечення.	5 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	6 тиждень	Додатки
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

_____ Артем КАРМАЛИГА
(підпис) (Імя ПРИЗВИЩЕ)

Керівник проєкту (роботи)

_____ Степан СКРУПСЬКИЙ
(підпис) (Імя ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра:
86 с., 3 табл., 33 рис., 3 дод., 18 джерел.

JAVASCRIPT, REACTJS, ЗАСТОСУНОК, ІНФОРМАЦІЙНА СИСТЕМА, МОВА ПРОГРАМУВАННЯ, ОБМІН ПОВІДОМЛЕННЯМИ, СПІЛКУВАННЯ КОРИСТУВАЧІВ.

Об'єкт дослідження – процес розробки інформаційної системи для спілкування.

Предмет дослідження – інформаційні системи для швидкого спілкування.

Мета роботи – розробка програмного забезпечення інформаційної системи для швидкого спілкування.

Матеріали, методи та технічні засоби: мова програмування JavaScript, бібліотека для побудови інтерфейсів ReactJS.

Результати. Розроблено проєктні рішення для створення програмного забезпечення інформаційної системи для швидкого спілкування. Створено програмне забезпечення інформаційної системи для швидкого спілкування. Проведено тестування розробленого програмного забезпечення інформаційної системи для швидкого спілкування.

Висновки. Мету роботи досягнуто. Розроблено програмне забезпечення інформаційної системи для швидкого спілкування за допомогою мови програмування JavaScript та бібліотеки для побудови інтерфейсів React.

Галузь використання – інформаційні системи для спілкування користувачів.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 86 pages, 3 tables, 33 figures, 3 appendixes, 18 sources.

JAVASCRIPT, REACTJS, APPLICATION, INFORMATION SYSTEM, PROGRAMMING LANGUAGE, MESSAGE EXCHANGE, USER COMMUNICATION.

The object of research is the process of developing information system for communication.

The subject of the research is information systems for rapid communication.

The purpose of this work is to develop the software of the information system for rapid communication.

Materials, methods and technical tools: JavaScript programming language, React library for building interfaces.

Results. Project solutions for the creation of software for the information system for rapid communication has been designed. The software of the information system for rapid communication has been developed. The developed software of the information system for rapid communication has been tested.

Conclusions. The goal of the work has been achieved. The software tools of information system for rapid communication using the JavaScript programming language and the React library for building interfaces have been developed.

Scope of use – information systems for user communication.

ЗМІСТ

	С.
Перелік скорочень та умовних позначок	8
Вступ	9
1 Аналіз предметної області	11
1.1 Інформаційні системи для швидкого спілкування	11
1.2 Аналіз інформаційних систем для швидкого спілкування	22
1.3 Висновки за розділом 1	32
2 Матеріали і методи	33
2.1 Вибір мови програмування	33
2.2 Вибір бібліотеки для побудови інтерфейсів	36
2.3 Висновки за розділом 2	38
3 Опис програми	39
3.1 Структура програмного забезпечення інформаційної системи для швидкого спілкування	39
3.2 Функціонування програмного забезпечення інформаційної системи для швидкого спілкування	42
3.3 Розробка бази даних програмного забезпечення інформаційної системи для швидкого спілкування	45
3.4 Проєктування інтерфейсу програмного забезпечення інформаційної системи для швидкого спілкування	47
3.5 Висновки за розділом 3	50
4 Експлуатація, тестування та експериментальне дослідження програми	52
4.1 Призначення й умови застосування програми	52
4.2 Характеристики програмного забезпечення інформаційної системи для швидкого спілкування	53
4.3 Інструкція по експлуатації програми	54
4.3.1 Звернення до програми	54
4.3.2 Вхідні й вихідні дані	55
4.3.3 Повідомлення	55

4.4 Виконання програмного забезпечення інформаційної системи для швидкого спілкування	55
4.5 Тестування програмного забезпечення інформаційної системи для швидкого спілкування	59
4.6 Висновки за розділом 4	59
Висновки	60
Перелік джерел посилання	63
Додаток А Технічне завдання	65
Додаток Б Фрагмент тексту програми	69
Додаток В Слайди презентації	79

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

БД – база даних;

ІС – інформаційна система;

ПЗ – програмне забезпечення;

ШС – швидке спілкування.

ВСТУП

Програмні засоби інформаційних систем для швидкого спілкування відіграють ключову роль у сучасному суспільстві, оскільки забезпечують миттєвий обмін інформацією незалежно від географічного розташування користувачів. Актуальність такого програмного забезпечення особливо помітна в бізнесі, освіті, медицині та екстрених службах, де оперативність комунікації має вирішальне значення [1].

Віддалена робота та онлайн навчання стали невід'ємною частиною сучасного життя, і саме інформаційні системи для спілкування, такі як месенджери та системи відео зв'язку, дозволяють ефективно організовувати робочий процес. Швидкий обмін повідомленнями, файлами та організація відео зв'язку значно підвищують продуктивність роботи [2], [3].

Автоматизація комунікацій також стає все більш популярною. Чат-боти та штучний інтелект допомагають компаніям обробляти запити клієнтів без залучення операторів, що значно економить ресурси. Інтеграція інформаційних систем швидкого спілкування з іншими програмними платформами сприяє покращенню обслуговування клієнтів і дозволяє бізнесу працювати більш ефективно [4], [5].

Завдяки інформаційним системам для швидкого спілкування люди можуть підтримувати зв'язок незалежно від відстані, що сприяє глобальній взаємодії. Крім того, впровадження технологій для людей з обмеженими можливостями, таких як голосові помічники та адаптивні інтерфейси, сприяє підвищенню рівня доступності [6], [7].

Проте деякі програмні засоби інформаційних систем для швидкого спілкування є недостатньо надійними через проблеми з безпекою та конфіденційністю даних. Проблеми сумісності між різними платформами можуть ускладнювати інтеграцію таких систем у робочі процеси. Деякі месенджери працюють лише у певних екосистемах, що змушує користувачів адаптуватися до конкретного програмного забезпечення або використовувати

кілька інструментів одночасно. Це може викликати незручності та ускладнювати обмін файлами, контактами та іншими даними між різними сервісами. Також варто враховувати технічні обмеження та вимоги до ресурсів пристрою. Деякі сучасні програми для швидкого спілкування споживають значну кількість оперативної пам'яті та процесорних ресурсів, що може сповільнювати роботу пристроїв, особливо старіших моделей. Окрім цього, монетизаційні моделі деяких програмних засобів можуть створювати дискомфорт для користувачів. Безкоштовні месенджери часто містять рекламу, а платні версії вимагають додаткових витрат. Деякі сервіси обмежують функціонал у безкоштовних версіях, що змушує користувачів купувати підписку або переходити на інші платформи [1]-[7].

Тому актуальною є розробка інформаційної системи для швидкого спілкування. У дипломній кваліфікаційній роботі бакалавра розв'язується актуальне завдання розробки програмного забезпечення інформаційної системи для швидкого спілкування.

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

- виконати аналіз предметної області та інформаційних систем для швидкого спілкування;
- здійснити проектування програмного забезпечення інформаційної системи для швидкого спілкування;
- створити програмне забезпечення інформаційної системи для швидкого спілкування;
- виконати тестування розробленого програмного забезпечення інформаційної системи для швидкого спілкування.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Інформаційні системи для швидкого спілкування

Інформаційні системи для швидкого спілкування забезпечують оперативний обмін повідомленнями між користувачами в режимі реального часу. Вони використовують мережеві технології для передачі текстової, голосової та відеоінформації, що дозволяє ефективно підтримувати комунікацію незалежно від відстані [1].

У таких системах реалізуються механізми миттєвої синхронізації даних, що дозволяє отримувати повідомлення без затримок. Вбудовані алгоритми шифрування забезпечують захист інформації під час її передачі, а розподілена архітектура сприяє стабільності роботи та швидкому масштабуванню відповідно до зростаючих потреб користувачів [1].

Функціональність таких платформ передбачає можливість обміну мультимедійними файлами, створення групових чатів, здійснення аудіо- та відеодзвінків. Використання хмарних технологій дозволяє зберігати повідомлення та файли, забезпечуючи доступ до них з різних пристроїв [1].

Адаптивний інтерфейс і підтримка мобільних пристроїв роблять ці системи зручними для використання у будь-якому середовищі. Додаткові інтеграції з іншими сервісами дають змогу розширювати можливості комунікації, підвищуючи ефективність обміну інформацією [1].

Миттєвий обмін повідомленнями передбачає можливість спілкування в реальному часі або надсилання асинхронних повідомлень за допомогою спеціалізованих програм. Для здійснення такої комунікації використовуються платформи, що дозволяють передавати текстову, голосову та мультимедійну інформацію між користувачами [1].

Застосування таких програм дає змогу легко підтримувати зв'язок із потрібними контактами, що є особливо важливим у робочому середовищі. Взаємодія відбувається через модель клієнт-сервер, де повідомлення спочатку надходить на сервер, а потім передається отримувачу. Використання списків

контактів забезпечує швидкий доступ до необхідних співрозмовників, а вбудовані сповіщення дозволяють відстежувати зміну їхнього статусу в мережі [1].

Швидкий обмін повідомленнями має низку переваг. Такий спосіб комунікації є зручним, оперативним і значно ефективнішим за традиційні методи обміну інформацією. Окрім текстових повідомлень, доступна передача зображень, відео, аудіофайлів і документів у різних форматах. Функція надсилання геолокації полегшує орієнтацію та пошук необхідного місця. Обмежень у часі користування такими сервісами немає, тому комунікація може відбуватися у будь-який зручний момент [1].

Такі системи підтримують групові чати, у яких одночасно можуть брати участь кілька користувачів. Простий та інтуїтивний інтерфейс забезпечує легкість у використанні, а швидке отримання повідомлень робить їх більш ефективним засобом комунікації, ніж електронна пошта [1].

Попри численні переваги, такі системи мають і певні недоліки. Деякі з них можуть бути недостатньо захищеними, що створює ризики для конфіденційності даних. У деяких випадках історія повідомлень не зберігається між сесіями, що може ускладнювати роботу. Надмірна кількість сповіщень або складність використання певних функцій також можуть впливати на зручність взаємодії з такими програмами [1].

Сучасні технології значно змінили спосіб взаємодії між людьми, надаючи можливість обмінюватися інформацією миттєво, незалежно від відстані. Цей спосіб комунікації набув широкого поширення та вплинув на різні аспекти суспільного життя, формуючи нові підходи до спілкування та взаємодії [2].

Швидка передача повідомлень, відома також як онлайн чат, дозволяє обмінюватися текстовими повідомленнями в реальному часі через спеціалізовані цифрові платформи. Завдяки цьому з'явилася можливість оперативного зв'язку, що сприяє зручності та ефективності взаємодії.

Усвідомлення значення цього явища допомагає краще зрозуміти його розвиток і вплив на щоденне життя [2].

Основна перевага такої комунікації полягає в її швидкості та простоті використання. Відстань перестала бути перешкодою для підтримки зв'язку, а традиційні засоби спілкування, що вимагали значного часу або фінансових витрат, поступово втратили актуальність. Обмін інформацією тепер відбувається без затримок, що значно покращує якість взаємодії [2].

Окрім текстових повідомлень, можливість надсилати зображення, відео чи документи робить спілкування більш ефективним і зручним. Така функціональність особливо корисна у професійному середовищі, де важливо швидко обмінюватися файлами та координувати спільну роботу [2].

Можливість підтримувати зв'язок із потрібними людьми є однією з важливих особливостей миттєвого обміну повідомленнями. Завдяки цьому можна легко знаходити потрібні контакти, налагоджувати взаємодію та формувати соціальні або ділові зв'язки. Такий формат комунікації сприяє підтримці стосунків як у особистій сфері, так і в професійному середовищі [2].

Сучасні платформи для спілкування надають широкий спектр інструментів, що сприяють зручності та ефективності обміну інформацією. Окрім текстових повідомлень, доступні мультимедійні матеріали, можливість створювати списки контактів, встановлювати статуси, а також здійснювати голосові та відео дзвінки. Функціональність цих сервісів постійно розширюється, адаптуючись до потреб користувачів [2].

Передача текстових повідомлень у режимі реального часу забезпечує миттєву взаємодію без затримок, що робить спілкування динамічним і ефективним. Очікування відповіді зведено до мінімуму, що значно підвищує швидкість обміну інформацією [2].

Функція обміну мультимедіа дозволяє передавати зображення, відео та різноманітні файли, що значно збагачує комунікацію та робить її більш виразною. Візуальний контент відіграє важливу роль у передачі емоцій та інформації, що особливо цінується сучасними користувачами [2].

Організація списків контактів полегшує доступ до важливих співрозмовників, спрощує пошук необхідних людей та підтримку зв'язку. Це дає змогу легко орієнтуватися в мережі знайомств і швидко виходити на зв'язок у разі потреби [2].

Можливість оновлення статусу дозволяє ділитися інформацією про поточні події, настрій або зайнятість. Це допомагає оточенню розуміти, чи є можливість для спілкування, що сприяє більш комфортній комунікації [2].

Додаткові можливості, такі як відео та голосові дзвінки, роблять взаємодію більш персоналізованою, дозволяючи спілкуватися в різних форматах. Групові чати надають змогу вести розмови одразу з кількома людьми, що є особливо корисним для координації спільних дій, обговорення проєктів та підтримки командної роботи [2].

Передача файлів стала однією з ключових можливостей сучасних інформаційних систем швидкого спілкування. Завдяки цьому з'явилася зручність у відправленні та отриманні документів, мультимедійних матеріалів і різноманітних даних безпосередньо на пристрої. Особливо значущою ця можливість є для професійної діяльності, де швидкий і зручний обмін інформацією суттєво впливає на продуктивність [2].

Швидкість і зручність комунікації, яку забезпечує миттєвий обмін повідомленнями, зробили його незамінним елементом повсякденного спілкування. Відстань і часові рамки перестали бути перешкодами, а різноманітні функції дозволили повністю змінити підхід до взаємодії між людьми. Ця технологія стала важливою частиною цифрового способу життя, вплинувши на всі сфери комунікації [2].

Ідея миттєвого обміну повідомленнями бере початок із перших спроб онлайн спілкування. В епоху становлення Інтернету з'явилися перші текстові чати, що поступово розширювали можливості цифрової взаємодії. Поступове вдосконалення технологій дозволило створити ті засоби зв'язку, які використовуються сьогодні [2].

З поширенням Інтернету з'явилася можливість миттєвого дистанційного спілкування. Основним способом взаємодії залишалася електронна пошта, проте вона не завжди відповідала потребам оперативного обміну інформацією. Саме тому почали з'являтися технології, що дозволяли вести діалог у режимі реального часу. Зростаюча популярність Інтернету призвела до підвищення попиту на швидке та зручне спілкування. Хоча електронна пошта вже надавала можливість обміну повідомленнями, вона не могла забезпечити миттєвий зворотний зв'язок, що було особливо важливо для багатьох користувачів. Виникла необхідність у технологіях, що поєднували б оперативність особистих розмов із цифровими можливостями, що й стало основою для розвитку нових комунікаційних платформ. Ідея швидкого текстового спілкування почала набувати популярності, оскільки розробники прагнули створити технологію, яка б заповнила прогалину між традиційною електронною поштою та безпосередніми діалогами [2].

Поява першої системи миттєвого обміну повідомленнями започаткувала новий етап в онлайн спілкуванні. Завдяки цій платформі стало можливим надсилати короткі текстові повідомлення та перевіряти статус доступності контактів у режимі реального часу. Це значно спростило взаємодію між користувачами, дозволяючи підтримувати зв'язок із близькими людьми незалежно від відстані [2].

Важливою інновацією, яку принесли системи швидкого спілкування, стало використання спеціальних символів. Завдяки цим графічним елементам з'явився новий спосіб вираження емоцій у текстових розмовах. Візуальне підсилення повідомлень зробило комунікацію активнішою, допомагаючи передавати настрій і додавати розмовам особистісний характер [2].

Успіх перших платформ миттєвого обміну повідомленнями заклав основу для подальшого розвитку цифрових засобів комунікації. Ця технологія поступово вдосконалювалася, полегшуючи взаємодію між людьми та прокладаючи шлях для сучасних інформаційних систем швидкого спілкування, які активно використовуються сьогодні [2].

Розвиток Інтернету сприяв еволюції технологій обміну миттєвими повідомленнями, що зробило їх доступнішими та зручнішими у використанні. Збільшення кількості користувачів і вдосконалення функціоналу спричинили стрімке зростання та розширення можливостей таких сервісів [2].

Інформаційні платформи для спілкування, серед яких Yahoo Messenger, Google Talk і Skype залучили величезну аудиторію по всьому світу та сприяли подальшому розвитку цифрових комунікацій. Водночас поширення соціальних мереж, зокрема Facebook і WhatsApp, докорінно змінило способи взаємодії між людьми, інтегрувавши можливості миттєвих повідомлень у повсякденне спілкування [2].

Прогрес у сфері технологій значною мірою вплинув на обмін повідомленнями. Впровадження мобільних додатків зробило цей процес ще зручнішим, а поява нових функцій, таких як сповіщення про прочитання, голосові повідомлення та наскрізне шифрування, підвищила рівень безпеки та конфіденційності цифрової комунікації [2].

Зміни в цій сфері вплинули не лише на особисте спілкування, а й на ділові комунікації. Миттєві повідомлення стали важливим інструментом для взаємодії між людьми, допомагаючи підтримувати зв'язок незалежно від відстані. Обмін текстами, зображеннями, відео чи навіть участь у голосових і відео дзвінках перетворилися на звичну частину повсякденного життя [2].

Компанії також адаптували ці технології, використовуючи чати для швидкої координації та ефективної співпраці. Робочі процеси стали більш автоматизованими, а команди отримали можливість оперативно обговорювати ідеї, приймати рішення та оптимізувати взаємодію між співробітниками [2].

Оскільки цифрові комунікації продовжують розвиватися, технологічні інновації, такі як штучний інтелект і чат-боти, поступово займають все більш значну роль. Автоматизовані системи, що здатні надавати персоналізовані відповіді та рекомендації, значно покращують взаємодію користувачів із сервісами, сприяючи створенню ще більш ефективних інструментів для спілкування [2].

Сучасні технології обміну миттєвими повідомленнями відкривають нові перспективи для спілкування. Використання доповненої та віртуальної реальності дає змогу створювати більш інтерактивний досвід обміну інформацією, розширюючи межі традиційного текстового формату. Крім того, інтеграція з пристроями Інтернету речей сприяє підвищенню ефективності комунікаційних систем, забезпечуючи зручність і швидкість взаємодії [2].

Перетворення у сфері миттєвого обміну повідомленнями докорінно змінили способи взаємодії, сприяючи подоланню бар'єрів і зміцненню зв'язків незалежно від відстані. Від перших розробок до сучасних технологічних рішень ця галузь постійно розвивається, формуючи цифровий простір і адаптуючись до нових викликів. З кожним новим етапом відкриваються безмежні можливості, що продовжують впливати на щоденне життя та майбутнє глобальної комунікації [2].

Миттєвий обмін повідомленнями дозволяє спілкуватися в реальному часі через Інтернет або локальні мережі. Останнім часом популярність таких сервісів зростає завдяки їхній простоті використання та широкому спектру можливостей [3].

Сучасні програми для швидкого спілкування дають змогу оперативно передавати інформацію та підтримувати зв'язок з друзями, родиною чи колегами. Такі платформи, як WhatsApp, Facebook Messenger і Telegram, забезпечують приватний обмін повідомленнями, що контролюється лише користувачами. Для професійного використання рекомендується вибирати месенджери, які легко інтегруються в корпоративні комунікаційні системи та гарантують безпеку даних. Варто також уважно ставитися до змісту повідомлень, оскільки на публічних платформах інформація може бути сприйнята неоднозначно. Незважаючи на ефективність текстового спілкування, важливо зважати на його доцільність порівняно з іншими засобами комунікації, такими як електронна пошта або телефонні дзвінки [3].

Розвиток цифрових технологій значно вплинув на способи взаємодії між людьми. Завдяки мобільним пристроям і швидкісному доступу до

Інтернету месенджери стали невід'ємною частиною повсякденного життя, замінивши традиційні текстові повідомлення та телефонні дзвінки. Окрім звичайного листування, багато додатків підтримують групові чати, відео дзвінки, голосові повідомлення та обмін файлами, що значно спрощує спілкування [3].

Ринок програм для швидкого спілкування пропонує широкий вибір додатків, кожен з яких має власний набір функцій та особливостей. Серед найбільш популярних сервісів можна виділити Signal, Telegram, WhatsApp, WeChat, QQ, Viber, Line і Snapchat. Вибір конкретної програми залежить від індивідуальних потреб користувача: деякі додатки орієнтовані на підвищений рівень безпеки, інші пропонують розширені мультимедійні можливості або інтеграцію з соціальними мережами. З огляду на це важливо обирати додаток, який найкраще відповідає особистим або професійним вимогам [3].

Миттєве повідомлення — це технологія, що забезпечує обмін текстовими повідомленнями, голосовими та відео дзвінками в режимі реального часу через Інтернет або інші цифрові мережі. Такі сервіси можуть працювати як окремі програми або інтегруватися з іншими платформами та пристроями, забезпечуючи зручний і швидкий спосіб комунікації [4].

Функціональність сучасних програм для швидкого спілкування виходить далеко за межі звичайного текстового спілкування. Завдяки таким інструментам стає можливим передавання мультимедійного контенту, включно з фотографіями, відео, посиланнями та різними файлами. Взаємодія між користувачами відбувається миттєво, що робить ці сервіси ефективним засобом як для особистого, так і для робочого спілкування [4].

Популярність програм для швидкого спілкування постійно зростає, а разом із нею з'являються нові можливості та розширюється спектр доступних функцій. Вони розробляються з урахуванням різних потреб — від базового обміну повідомленнями до комплексних рішень для бізнесу. Крім традиційного текстового формату, такі сервіси можуть підтримувати

інтеграцію з іншими платформами або застосовуватися у вигляді самостійних програм [4].

Системи обміну повідомленнями в соціальних мережах надають можливість вести комунікацію як у приватному, так і в публічному режимі. Ці платформи активно використовуються не лише для особистого листування, а й для бізнес-комунікацій. Завдяки таким сервісам компанії можуть підтримувати зв'язок із клієнтами, надавати оперативну підтримку та залучати нову аудиторію. Розширені можливості, як-от відео дзвінки, голосові повідомлення та інтеграція чат-ботів, значно підвищують ефективність взаємодії. Використання таких інструментів допомагає підприємствам зміцнювати відносини з клієнтами та покращувати рівень обслуговування [5].

Використання електронної пошти в поєднанні з миттєвими повідомленнями створює зручний гібридний формат комунікації, що поєднує переваги обох технологій. Такий підхід особливо актуальний у професійному середовищі, де важливо не лише швидко обмінюватися інформацією, а й зберігати історію розмов [5].

Системи електронного листування із вбудованими чатами часто підтримують обмін файлами значного розміру, інтеграцію з адресними книгами та можливість створювати тематичні обговорення. Це сприяє ефективнішій взаємодії між співробітниками та підвищенню продуктивності командної роботи [5].

Обмін відео повідомленнями дозволяє здійснювати відео дзвінки в асинхронному режимі, коли записані відео фрагменти передаються одержувачу, який може переглянути їх у зручний час і відповісти аналогічним способом. Такі інструменти підтримують не лише відео та голосовий зв'язок, а й передавання зображень та аудіо файлів. Вони сумісні з різними пристроями, що забезпечує гнучкість у використанні [5].

Миттєві повідомлення стали незамінними у внутрішньому спілкуванні компаній та у сфері клієнтської підтримки. Окрім стандартного текстового формату, такі платформи можуть включати інтеграцію чат-ботів, визначення

місцезнаходження користувача та навіть проведення фінансових операцій, що робить їх надзвичайно функціональними [5].

Однією з ключових особливостей є можливість миттєвого обміну інформацією, що забезпечує оперативність взаємодії. Це дозволяє без затримок передавати текстові повідомлення, мультимедійний контент і голосові записи [5].

Групові чати також є важливою частиною таких систем, оскільки вони надають можливість обговорювати робочі питання, організовувати спільні проєкти та швидко обмінюватися ідеями в режимі реального часу. Такі інструменти сприяють не лише ефективному вирішенню завдань, а й підвищенню рівня взаємодії між учасниками комунікації [5].

Окрім можливості обмінюватися текстовими повідомленнями, чат сприяє створенню більш глибокої та особистої взаємодії, ніж відео чи голосові дзвінки. Завдяки інформаційним технологіям комунікаційні платформи забезпечують миттєвий обмін інформацією, що є важливим як у повсякденному житті, так і в професійній діяльності [6].

Такий формат дозволяє легко підтримувати зв'язок із колегами незалежно від їхнього місцезнаходження, проводити онлайн зустрічі, координувати спільну роботу та сприяти зміцненню робочих відносин між учасниками команди [6].

Передача файлів у режимі реального часу дає змогу надсилати різноманітні матеріали, включаючи документи, посилання, зображення, відео та анімації, індивідуальним користувачам або групам. Використання наскрізного шифрування забезпечує максимальний рівень безпеки під час обміну будь-якими файлами [6].

Можливість ділитися екраном значно полегшує вирішення складних питань, зокрема у сфері клієнтської підтримки, де часто виникають труднощі через недостатнє розуміння ситуації. Така функція дає змогу демонструвати екран у процесі відео чи аудіо зв'язку, що сприяє оперативному усуненню

проблем, підвищує продуктивність робочих процесів та покращує зворотний зв'язок між членами команди [6].

Функція відображення введеного тексту під час набору доступна на більшості чат-платформ і додає динамічності спілкуванню. Вона допомагає зрозуміти, що відповідь уже формується, що робить взаємодію більш живою та природною [6].

Функціональність програм для швидкого спілкування дозволяє зберігати історію спілкування для подальшого використання. Завдяки запису дзвінків і повідомлень можна зберігати важливі дані, контролювати ефективність роботи команди та мінімізувати труднощі, що виникають у процесі комунікації. Серед переваг миттєвого обміну повідомленнями можна виокремити можливість спілкування без затримок, моніторинг робочих процесів, підтримку особистих розмов, а також безпечний формат передачі інформації [6].

Система обміну повідомленнями створена для оптимізації взаємодії як у діловому, так і в особистому середовищі. Вона дозволяє вести розмови між окремими користувачами або всередині групового спілкування. Для надсилання та отримання повідомлень обидві сторони мають бути підключені до мережі Інтернет через будь-який із доступних пристроїв – мобільних телефонів, планшетів, ноутбуків або комп'ютерів [6].

Передача інформації в системах швидкого спілкування здійснюється за допомогою спеціалізованих протоколів. Для обміну миттєвими повідомленнями використовуються технології, які забезпечують структурування даних, а безпеку взаємодії гарантують протоколи шифрування, що забезпечує конфіденційність і захист переданих даних [7].

Процес обміну повідомленнями в системі швидкого спілкування включає кілька основних етапів. Спочатку необхідно встановити відповідну програму на пристрій, після чого вона підключається до сервера через спеціальний протокол. Під час запуску система запитує облікові дані, такі як ім'я користувача та пароль, які використовуються для авторизації. Після

успішного входу здійснюється збір та передача важливих даних, серед яких контактна інформація, IP-адреса та номер порту, що надсилаються на сервер [7].

Сервер обробляє отримані дані, створює тимчасові файли для збереження списку контактів та надсилає запити на реєстрацію в системі. Щоб розпочати розмову, достатньо вибрати контакт зі списку, після чого відкривається вікно для введення тексту. Повідомлення, що вводиться, проходить процес шифрування та надсилається на сервер, де відбувається його обробка. Якщо одержувач перебуває в мережі, сервер перенаправляє повідомлення безпосередньо на його пристрій, розшифровує та відображає у відповідній програмі [7].

При отриманні повідомлення можна відповісти аналогічним чином. Вікно системи швидкого спілкування дозволяє переглядати історію розмови та відповідати на нові повідомлення в режимі реального часу. Після завершення діалогу доступна можливість закрити вікно або вийти із системи [7].

Обмін миттєвими повідомленнями та написання листів використовуються для комунікації, але відрізняються за структурою, змістом та технічними аспектами. Завдяки економії часу та підвищенню рівня задоволеності користувачів все більше компаній інтегрують чат-боти у свої комунікаційні процеси [7].

Відзначено, що інформаційні системи для швидкого спілкування забезпечують оперативний обмін повідомленнями між користувачами в режимі реального часу. Вони використовують мережеві технології для передачі текстової, голосової та відеоінформації, що дозволяє ефективно підтримувати комунікацію незалежно від відстані.

1.2 Аналіз інформаційних систем для швидкого спілкування

Проаналізуємо інформаційні системи та програмні засоби для швидкого спілкування [8]-[14].

Програмне забезпечення інформаційної системи Twist (рис. 1.1) для швидкого спілкування забезпечує миттєвий обмін повідомленнями та надає всі необхідні інструменти для ефективної взаємодії. Завдяки багатофункціональному підходу ця система для швидкого спілкування дозволяє організувати комунікацію без затримок, що особливо важливо для командної роботи. Twist створений для того, щоб забезпечити миттєвий зв'язок між користувачами, пропонуючи зручний доступ до групових чатів, електронної пошти та функцій обміну файлами [8], [9], [12].

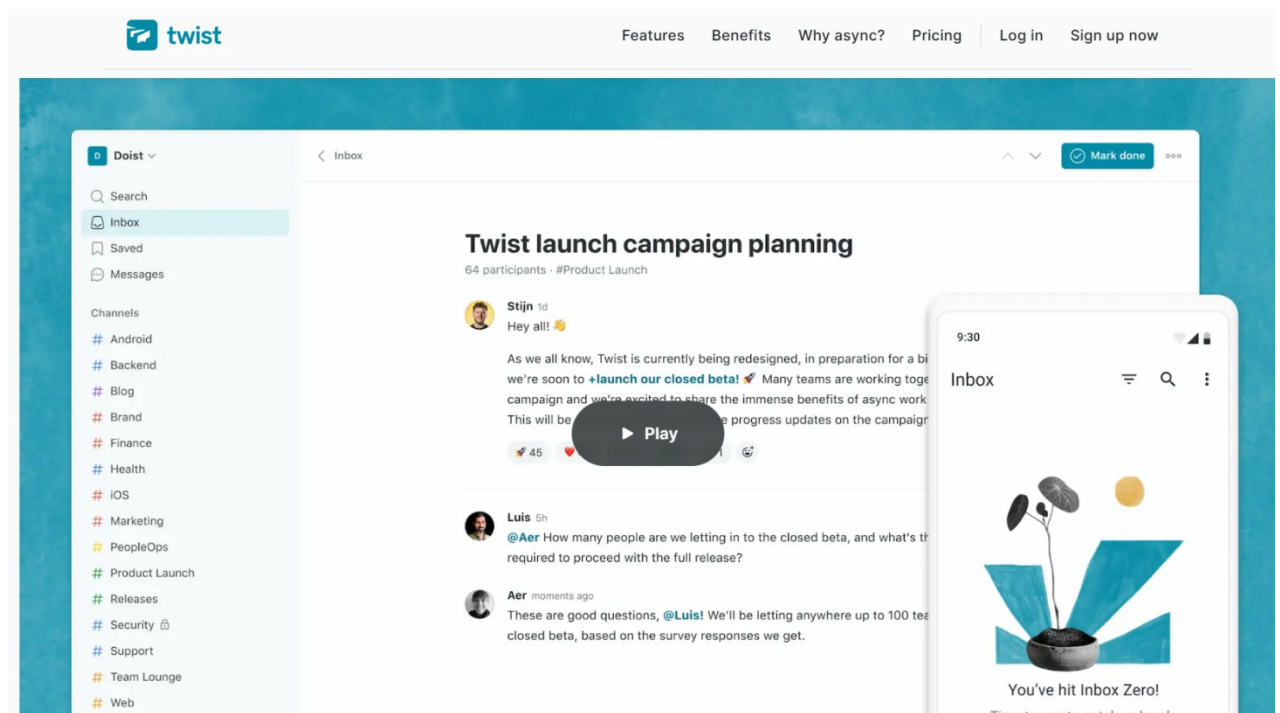


Рисунок 1.1 – Програмне забезпечення інформаційної системи Twist [12]

Програмне забезпечення інформаційної системи Twist легко інтегрується з популярними програмами, такими як Microsoft Outlook, а також іншими сервісами, що використовуються для ведення ділової переписки. Twist спрощує спілкування, дозволяючи швидко взаємодіяти без необхідності перемикатися між різними платформами [8], [9], [12].

Ця інформаційна система орієнтована на організований та асинхронний обмін інформацією, що сприяє зменшенню навантаження від великої кількості повідомлень. Використання тематичних обговорень та інтеграція з

інструментами підвищення продуктивності робить спілкування не тільки швидким, але й впорядкованим. Програмне забезпечення Twist розроблений для тих, хто прагне підтримувати ефективний та миттєвий зв'язок без втрати важливої інформації. [8], [9], [12].

Основними особливостями та характеристиками програмного забезпечення інформаційної системи Twist є такі [8], [9], [12]:

- швидке спілкування: програмне забезпечення інформаційної системи Twist підтримує миттєвий обмін повідомленнями та швидку взаємодію між користувачами [8], [9], [12];

- організоване спілкування: програмне забезпечення інформаційної системи Twist підтримує структурування розмов за темами, що допомагає уникнути хаосу в комунікації.

- асинхронний режим: програмне забезпечення інформаційної системи Twist дозволяє користувачам відповідати в зручний час, зменшуючи перевантаження повідомленнями [8], [9], [12];

- інтеграція з іншими сервісами: програмне забезпечення інформаційної системи Twist сумісне з Microsoft Outlook та іншими популярними інструментами для продуктивності [8], [9], [12];

- підтримка групової взаємодії: програмне забезпечення інформаційної системи Twist надає можливість спілкуватися в командних чатах та обговорювати робочі питання [8], [9], [12];

- обмін файлами: програмне забезпечення інформаційної системи Twist дозволяє швидко передавати документи, зображення та інші файли безпосередньо в чаті [8], [9], [12];

- захищеність даних: програмне забезпечення інформаційної системи Twist використовує сучасні методи шифрування для безпечної передачі повідомлень [8], [9], [12];

- багатоплатформність: програмне забезпечення інформаційної системи Twist доступне на різних пристроях, включаючи комп'ютери, мобільні телефони та планшети [8], [9], [12];

– гнучке налаштування сповіщень: програмне забезпечення інформаційної системи Twist дозволяє керувати отриманням повідомлень, щоб уникнути зайвих відволікань [8], [9], [12];

– зручний інтерфейс: інформаційна система Twist має простий та інтуїтивно зрозумілий дизайн, що спрощує використання [8], [9], [12].

Серед недоліків програмного забезпечення Twist можна відзначити обмежені можливості для голосового та відео зв'язку, що може стати перешкодою для команд, які потребують частих відео конференцій. Також можливі труднощі з інтеграцією у вже існуючі корпоративні екосистеми, особливо якщо компанія використовує нестандартні або власноруч розроблені рішення. Інтерфейс, хоча й зручний, може вимагати часу на адаптацію для користувачів, які звикли до інших платформ для комунікації. Деякі функції можуть бути доступні лише в платних версіях, що може стати фінансовою перешкодою для невеликих команд. Крім того, для ефективного використання необхідно мати стабільне Інтернет з'єднання, а можливість роботи в автономному режимі обмежена [8], [9], [12].

Програмне забезпечення DocsBot AI (рис. 1.2) для швидкого спілкування створене саме для того, щоб забезпечити максимально оперативну взаємодію між користувачами [8], [13].

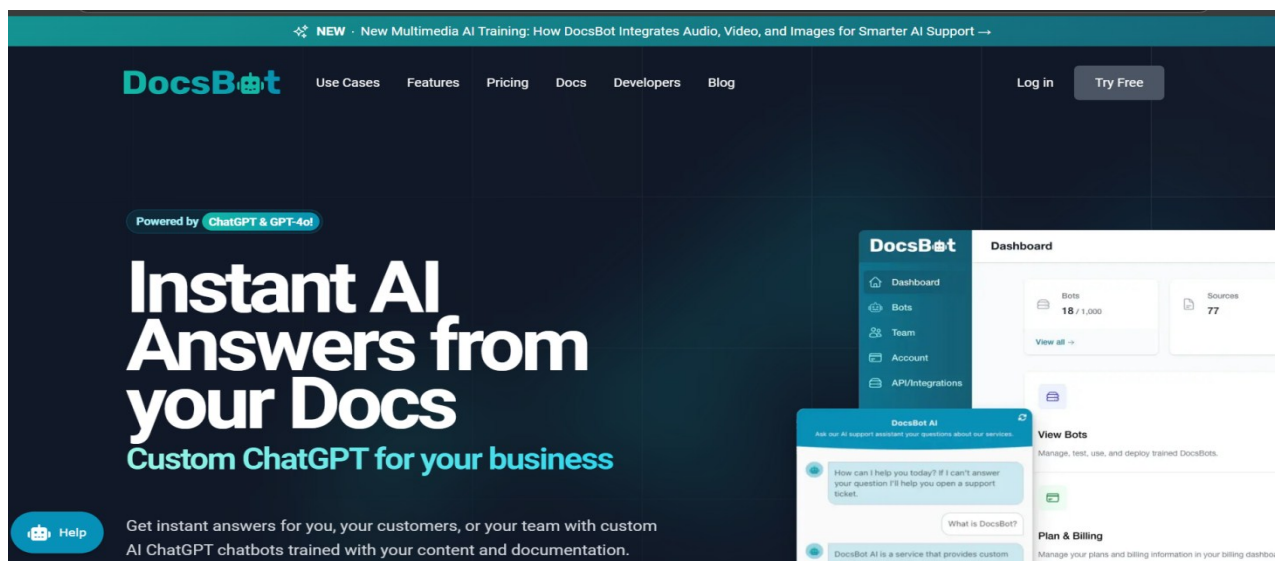


Рисунок 1.2 – Програмне забезпечення інформаційної системи DocsBot AI [13]

AI Завдяки функціям обміну повідомленнями в режимі реального часу комунікація між членами команди або з клієнтами відбувається миттєво, що дозволяє значно прискорити прийняття рішень [8], [13].

Програмне забезпечення DocsBot AI ефективно взаємодіє з іншими сервісами, об'єднуючи управління електронною поштою, обмін файлами та інші канали зв'язку в єдиному просторі. Це дає змогу швидко отримувати потрібну інформацію та зменшувати затримки у комунікаційному процесі. Функція пошуку дозволяє миттєво знаходити необхідні повідомлення з минулих розмов, що робить спілкування ще ефективнішим [8], [13].

Вбудовані параметри безпеки гарантують надійність переданих даних. Додатково передбачена можливість автоматизації відповідей та організації групових чатів, що сприяє покращенню координації. Освоєння розширених функцій може зайняти певний час, але після адаптації їх використання значно спрощує комунікаційний процес [8], [13].

Програмне забезпечення DocsBot AI має такі особливості [8], [13]:

- швидке спілкування: програмне забезпечення DocsBot AI інформаційної системи підтримує миттєвий обмін повідомленнями, що дозволяє оперативно реагувати на запити та приймати рішення [8], [13];

- інтеграція з іншими сервісами: програмне забезпечення DocsBot AI інформаційної системи підтримує взаємодію з електронною поштою, сховищами файлів та іншими бізнес-інструментами, об'єднуючи все в єдину систему [8], [13];

- режим реального часу: програмне забезпечення DocsBot AI інформаційної системи дозволяє надсилати та отримувати повідомлення без затримок, що покращує ефективність командної роботи [8], [13];

- розширені функції пошуку: програмне забезпечення DocsBot AI інформаційної системи підтримує швидкий доступ до історії повідомлень та файлів для зручного управління комунікацією [8], [13];

- безпека даних: програмне забезпечення DocsBot AI інформаційної системи підтримує використовує сучасні механізми шифрування для захисту конфіденційної інформації під час передачі [8], [13];

- автоматизація відповідей: програмне забезпечення DocsBot AI інформаційної системи надає можливість налаштовувати боти для автоматичної обробки повідомлень, що зменшує навантаження на користувачів [8], [13];

- групові чати: програмне забезпечення DocsBot AI інформаційної системи дозволяє створювати окремі канали для обговорення проєктів та швидкої координації команди [8], [13];

- зручний інтерфейс: інтуїтивно зрозумілий дизайн спрощує освоєння програмного забезпечення [8], [13];

- підтримка голосових повідомлень: програмне забезпечення DocsBot AI інформаційної системи підтримує можливість надсилати аудіоповідомлення для ще швидшого обміну інформацією [8], [13];

- гнучкі налаштування: програмне забезпечення DocsBot AI інформаційної системи дозволяє змінювати параметри доступу, сповіщень та персоналізації відповідно до потреб користувачів [8], [13].

Серед недоліків програмного забезпечення DocsBot AI можна відзначити складність освоєння розширених функцій, що може вимагати додаткового часу для адаптації. Інтерфейс, хоча й інтуїтивно зрозумілий, може здаватися перевантаженим через велику кількість налаштувань та інтеграцій. У разі високого навантаження система може демонструвати затримки в обробці запитів або надсиланні повідомлень. Крім того, хоча рівень безпеки досить високий, для деяких користувачів може бути недостатньо гнучких параметрів конфіденційності. Інтеграція з деякими менш популярними інструментами може вимагати додаткових налаштувань або не підтримуватися повністю [8], [13].

Програмне забезпечення інформаційної системи SleekFlow (рис. 1.3) дозволяє ефективно налагоджувати комунікацію, використовуючи

персоналізовані та автоматизовані відповіді, що значно прискорює обмін інформацією. Особливо ефективним є використання інтеграції з популярними платформами, такими як WhatsApp, що дає змогу швидко та зручно підтримувати зв'язок із клієнтами [9], [14].

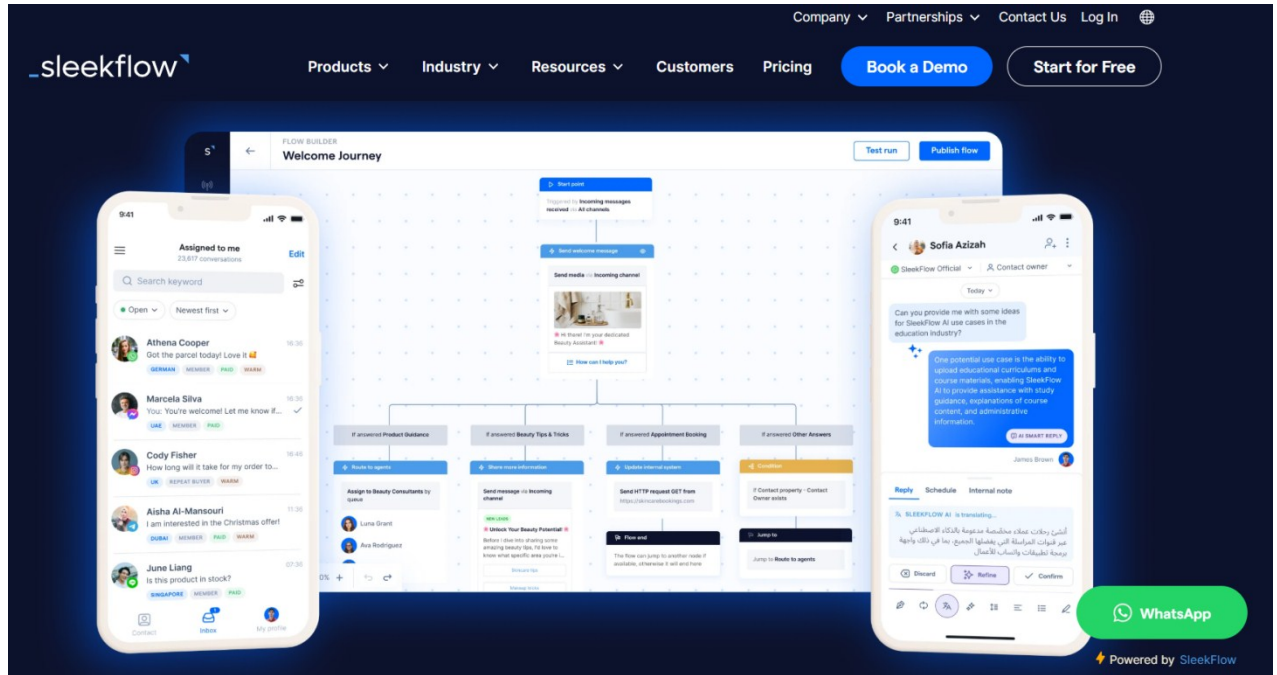


Рисунок 1.3 – Програмне забезпечення інформаційної системи SlekFlow [14]

Програмне забезпечення інформаційної системи SlekFlow дозволяє без затримок перемикатися між різними платформами, забезпечуючи миттєвий обмін повідомленнями та підвищуючи швидкість взаємодії. Це дає змогу не лише швидко відповідати на запити, а й значно покращувати рівень обслуговування користувачів. Додатково система пропонує аналітичні інструменти, що допомагають оцінювати ефективність комунікацій, сприяючи швидкому прийняттю рішень [9], [14].

Важливим елементом швидкого спілкування є можливість здійснювати платежі безпосередньо в чаті. Це спрощує процес транзакцій, усуваючи зайві кроки та прискорюючи взаємодію з клієнтами. Завдяки такому підходу комунікація стає не лише швидкою, а й зручною, що особливо важливо в сучасному цифровому середовищі [9], [14].

Програмне забезпечення інформаційної системи SleekFlow має такі особливості [9], [14]:

- швидке спілкування: програмне забезпечення інформаційної системи SleekFlow підтримує миттєву взаємодію з клієнтами через різні месенджери, включаючи WhatsApp, Facebook Messenger та інші платформи [9], [14];

- багатоканальна комунікація: програмне забезпечення інформаційної системи SleekFlow підтримує інтеграцію з різними каналами, що дозволяє швидко перемикатися між ними без втрати контексту розмови [9], [14];

- автоматизація повідомлень: програмне забезпечення інформаційної системи SleekFlow дає змогу налаштовувати боти та автоматичні відповіді, що значно пришвидшує процес комунікації [9], [14];

- персоналізовані взаємодії: програмне забезпечення інформаційної системи SleekFlow використовує дані про клієнтів для надання індивідуальних відповідей та пропозицій у режимі реального часу [9], [14];

- аналітика та звітність: програмне забезпечення інформаційної системи SleekFlow надає статистичні дані про комунікації, дозволяючи аналізувати ефективність повідомлень користувачів та покращувати взаємодію між ними [9], [14];

- платежі в чаті: програмне забезпечення інформаційної системи SleekFlow дозволяє здійснювати оплату безпосередньо в розмові, спрощуючи процес транзакцій для користувачів [9], [14];

- інтеграція з іншими системами: програмне забезпечення інформаційної системи SleekFlow легко поєднується з різними програмними системами, що дозволяє зберігати історію комунікацій і покращувати обслуговування клієнтів [9], [14];

- безпека даних: програмне забезпечення інформаційної системи SleekFlow використовує захищені канали зв'язку для забезпечення конфіденційності інформації [9], [14].

Серед недоліків програмного забезпечення SleekFlow можна відзначити певну складність у налаштуванні автоматизації для нових

користувачів, що може вимагати додаткового часу на освоєння. Інтеграція з деякими програмними системами потребує налаштувань вручну, що може створювати труднощі для компаній без технічної підтримки. Вартість тарифних планів може бути високою для малого бізнесу, особливо якщо необхідний доступ до розширених функцій. Деякі користувачі також відзначають, що швидкість обробки великих обсягів повідомлень може знижуватися під час пікових навантажень. Крім того, хоча платформа підтримує багато месенджерів, не всі з них мають однаковий рівень функціональності, що може обмежувати можливості взаємодії на певних каналах [9], [14].

Порівняльну характеристику інформаційних систем для швидкого спілкування наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння інформаційних систем для швидкого спілкування

Критерій порівняння	Twist [12]	DocsBot AI [13]	SleekFlow [14]
Зручність інтерфейсу	+	+—	+
Інтеграція з іншими програмними сервісами	+—	+	+
Можливість командної роботи	+	+—	+
Гнучкість налаштувань	+—	+	+—
Безпека даних	+	+	+—
Доступність безкоштовної версії	+	+—	—

За результатами проведеного аналізу можна зробити висновок, що у наш час існує досить багато інформаційних систем для швидкого спілкування. Проте деякі інформаційні системи для швидкого спілкування є недостатньо надійними через проблеми з безпекою та конфіденційністю даних. Проблеми

сумісності між різними платформами можуть ускладнювати інтеграцію таких систем у робочі процеси. Деякі месенджери працюють лише у певних екосистемах, що змушує користувачів адаптуватися до конкретного програмного забезпечення або використовувати кілька інструментів одночасно. Це може викликати незручності та ускладнювати обмін файлами, контактами та іншими даними між різними сервісами. Також варто враховувати технічні обмеження та вимоги до ресурсів пристрою. Деякі сучасні програми для швидкого спілкування споживають значну кількість оперативної пам'яті та процесорних ресурсів, що може сповільнювати роботу пристроїв, особливо старіших моделей. Окрім цього, монетизаційні моделі деяких програмних засобів можуть створювати дискомфорт для користувачів. Безкоштовні месенджери часто містять рекламу, а платні версії вимагають додаткових витрат. Деякі сервіси обмежують функціонал у безкоштовних версіях, що змушує користувачів купувати підписку або переходити на інші платформи. Тому актуальною є розробка програмного забезпечення інформаційної системи для швидкого спілкування.

При розробці програмного забезпечення інформаційної системи для швидкого спілкування необхідно забезпечити такі функціональні вимоги:

- підтримка можливості швидкого спілкування між користувачами за допомогою текстових повідомлень;
- підтримка можливості пошуку контактів користувачів, зареєстрованих у системі;
- можливість реєстрації та авторизації користувачів для подальшого спілкування;
- можливість редагування особистих даних користувача, зокрема, номеру телефону та імені;
- підтримка можливості додавання контактів для спілкування;
- підтримка можливості отримання повідомлень від інших користувачів, зареєстрованих у системі.

1.3 Висновки за розділом 1

Відзначено, що інформаційні системи для швидкого спілкування забезпечують оперативний обмін повідомленнями між користувачами в режимі реального часу. Вони використовують мережеві технології для передачі текстової, голосової та відеоінформації, що дозволяє ефективно підтримувати комунікацію незалежно від відстані.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато інформаційних систем для швидкого спілкування. Проте деякі інформаційні системи для швидкого спілкування є недостатньо надійними через проблеми з безпекою та конфіденційністю даних. Проблеми сумісності між різними платформами можуть ускладнювати інтеграцію таких систем у робочі процеси. Деякі месенджери працюють лише у певних екосистемах, що змушує користувачів адаптуватися до конкретного програмного забезпечення або використовувати кілька інструментів одночасно. Це може викликати незручності та ускладнювати обмін файлами, контактами та іншими даними між різними сервісами. Також варто враховувати технічні обмеження та вимоги до ресурсів пристрою. Деякі сучасні програми для швидкого спілкування споживають значну кількість оперативної пам'яті та процесорних ресурсів, що може сповільнювати роботу пристроїв, особливо старіших моделей. Окрім цього, монетизаційні моделі деяких програмних засобів можуть створювати дискомфорт для користувачів. Безкоштовні месенджери часто містять рекламу, а платні версії вимагають додаткових витрат. Деякі сервіси обмежують функціонал у безкоштовних версіях, що змушує користувачів купувати підписку або переходити на інші платформи. Тому актуальною є розробка програмного забезпечення інформаційної системи для швидкого спілкування.

Сформульовано функціональні вимоги до програмного забезпечення інформаційної системи для швидкого спілкування.

2 МАТЕРІАЛИ І МЕТОДИ

2.1 Вибір мови програмування

При розробці програмного забезпечення інформаційної системи для швидкого спілкування було використано мову програмування JavaScript [15], [16].

JavaScript є мовою програмування, що базується на специфікації ECMAScript і має відкритий вихідний код. Вона широко застосовується для створення веб-додатків, дозволяючи користувачам переглядати та взаємодіяти з інтерфейсами на різних мовах за допомогою браузера [15], [16].

Архітектура JavaScript включає в себе динамічні об'єкти, автоматизоване управління пам'яттю, а також механізм створення прототипів, що надає їй унікальні особливості. Завдяки схожості з Java ця мова користується популярністю серед розробників, які мають досвід роботи з різними технологіями. Важливо зазначити, що назва JavaScript є зареєстрованим товарним знаком корпорації Oracle [15], [16].

Ця мова програмування реалізує концепцію об'єктно-орієнтованого підходу, але відрізняється від класичних мов завдяки використанню прототипного наслідування. Також підтримуються різні типи, що дозволяє застосовувати функціональні підходи [15], [16].

У своїй структурі JavaScript об'єднує три основні компоненти: базову частину (ECMAScript), механізми для роботи з браузером та моделі документа. У безбраузерних середовищах підтримуються лише певні частини цієї функціональності [15], [16].

Для серверного виконання програмного коду можна використовувати як JavaScript, так і Java. Саме JavaScript є одним з основних інструментів для розробки сучасних веб-додатків. Його архітектура побудована на основі функцій та об'єктів, що дозволяє створювати складні системи з модульною структурою [15], [16].

Середовище JavaScript насичене великою кількістю фреймворків, що підтримують розробку кросплатформних рішень. Завдяки відкритій ліцензії ця мова отримала широке поширення в програмній індустрії. Окрім того, існують спеціалізовані розширення, що дозволяють застосовувати JavaScript у різних сферах. Організація коду базується на функціональній та модульній архітектурі, яка підтримує компіляцію в окремі пакети [15], [16].

Різноманітні реалізації JavaScript, що поширюються з відкритим кодом, надають розробникам можливість використовувати мову у гнучкий спосіб. Інструменти для керування залежностями дозволяють оптимізувати розробку, мінімізувати труднощі з багатопотоковими обчисленнями та вводити нові функціональні можливості [15], [16].

JavaScript підтримує асинхронне програмування за допомогою Promises, async/await та WebSockets, що дозволяє обробляти повідомлення, оновлювати список контактів та статуси користувачів без перезавантаження сторінки. Завдяки цьому інформаційна система для швидкого спілкування працює миттєво та ефективно [15], [16].

Спочатку JavaScript застосовувався для створення інтерфейсів у веб-браузерах, але з часом він перетворився на універсальний інструмент, що використовується як на клієнтській, так і на серверній стороні. Він надає можливість створювати інтерактивні веб-додатки, змінювати вміст сторінок у режимі реального часу та реагувати на дії користувача [15], [16].

Популярність цієї мови пояснюється її гнучкістю та широким спектром можливостей. JavaScript активно використовується не лише для побудови інтерфейсів, а й для серверної розробки. Він є однією з найзатребуваніших мов у сфері веб-технологій, що сприяє розвитку проектів з відкритим кодом та постійним вдосконаленням екосистеми [15], [16].

За допомогою JavaScript можна легко реалізувати реальну комунікацію через WebSockets або Socket.io, що є ключовим для систем швидкого обміну повідомленнями. Це дозволяє миттєво передавати дані між користувачами без затримок [15], [16].

Ця мова програмування включає в себе бібліотеки, які розширюють її функціональні можливості, а також підтримує різні підходи до програмування, серед яких структурне, об'єктно-орієнтоване та функціональне. До основних особливостей відносяться автоматизоване керування пам'яттю, ефективні алгоритми кешування, методи обробки даних, багатопотокові можливості та висока продуктивність при роботі з ресурсомісткими задачами [15], [16].

Обґрунтування вибору мови програмування для розробки програмного забезпечення інформаційної системи для швидкого спілкування наведено у таблиці 2.1.

Таблиця 2.1 – Обґрунтування вибору мови програмування для розробки програмного забезпечення інформаційної системи для швидкого спілкування

Критерій порівняння мов програмування	Мова програмування		
	JavaScript	C#	PHP
Продуктивність у реальному часі	+	+-	-
Гнучкість	+	-	-
Швидкість розробки	+	+-	+
Зручність для створення ПЗ інформаційної системи для швидкого спілкування	+	-	-
Кросплатформеність	+	+-	+

Отже, для реалізації програмного забезпечення інформаційної системи для швидкого спілкування обрано мову програмування JavaScript, оскільки ця мова забезпечує високу продуктивність, гнучкість та зручність у розробці вебзастосунків у реальному часі. Крім того, JavaScript є універсальною мовою, яка працює як у браузерях, так і на сервері, що дозволяє створити єдиний стек технологій для клієнтської та серверної частин програми, що спрощує розробку та підтримку інформаційної системи. Іншою перевагою є те, що

JavaScript має широкий набір інструментів для швидкої розробки чатів та інформаційних систем.

2.2 Вибір бібліотеки для побудови інтерфейсів

В якості бібліотеки для побудови інтерфейсів у процесі створення програмного забезпечення інформаційної системи для швидкого спілкування було обрано бібліотеку ReactJS [17], [18].

Бібліотека ReactJS, створена компанією Facebook, забезпечує ефективний механізм розробки інтерфейсів користувача. Завдяки її можливостям значно спрощується створення динамічних веб-додатків із високою продуктивністю [17], [18].

Одна з головних переваг React полягає в активній підтримці та постійному вдосконаленні. Великий обсяг навчальних матеріалів та документації дозволяє швидше освоювати технологію, а можливість використання JSX спрощує розробку контенту, оптимізованого для пошукових систем. Завдяки React розробка мобільних додатків стає більш доступною. Усе це робить React однією з найбільш затребуваних бібліотек JavaScript для створення швидких вебзастосунків [17], [18].

Ця технологія забезпечує лише візуальне відображення додатків і застосовується для створення компонентів інтерфейсу. Наприклад, у розробці користувацького інтерфейсу Instagram використовували саме React, що дозволило ефективно розподілити функціональність між різними частинами додатку [17], [18].

React базується на використанні об'єктної моделі, що є основою будь-якої вебсторінки. Браузер інтерпретує HTML-документ і перетворює його у структуру, з якою можна взаємодіяти за допомогою JavaScript. Використання JSX у React дозволяє писати код у стилі HTML, що прискорює процес розробки та забезпечує ефективну маніпуляцію DOM-елементами [17], [18].

Щоб підвищити продуктивність, у бібліотеці React застосовується концепція віртуальної об'єктної моделі. Ця технологія дозволяє оновлювати лише змінені частини сторінки, а не весь документ цілком, що значно знижує навантаження на систему і прискорює відтворення контенту. Завдяки цьому час завантаження веб-сторінок скорочується, що покращує загальний користувацький досвід. [17], [18].

Обґрунтування вибору бібліотеки для побудови інтерфейсів у процесі створення програмного забезпечення інформаційної системи для швидкого спілкування наведено у таблиці 2.2.

Таблиця 2.2 – Обґрунтування вибору бібліотеки для побудови інтерфейсів

Критерій порівняння бібліотек	Бібліотеки		
	ReactJS	SolidJS	Vue.js
Продуктивність	+	+	+—
Легкість у вивченні	+	+—	+
Зрілість екосистеми	+	—	+—
Сумісність з іншим ПЗ	+	+—	+—
Підтримка компаніями	+	—	+—

Таким чином, для створення програмного забезпечення інформаційної системи для швидкого спілкування обрано бібліотеку ReactJS, що забезпечує зручну модульну структуру розроблюваної програми через компонентний підхід, що дає змогу легко розширювати систему, додаючи нові функціональні можливості без значного переписування коду. Крім того, ReactJS має велику спільноту розробників, а також підтримується провідними компаніями, що

гарантує регулярні оновлення, довгострокове супроводження та інтеграцію з сучасними вебтехнологіями.

2.3 Висновки за розділом 2

Для реалізації програмного забезпечення інформаційної системи для швидкого спілкування обрано мову програмування JavaScript, оскільки ця мова забезпечує високу продуктивність, гнучкість та зручність у розробці вебзастосунків у реальному часі. Крім того, JavaScript є універсальною мовою, яка працює як у браузерях, так і на сервері, що дозволяє створити єдиний стек технологій для клієнтської та серверної частин програми, що спрощує розробку та підтримку інформаційної системи. Іншою перевагою є те, що JavaScript має широкий набір інструментів для швидкої розробки чатів та інформаційних систем.

Для створення програмного забезпечення інформаційної системи для швидкого спілкування обрано бібліотеку ReactJS, що забезпечує зручну модульну структуру розроблюваної програми через компонентний підхід, що дає змогу легко розширювати систему, додаючи нові функціональні можливості без значного переписування коду. Крім того, ReactJS має велику спільноту розробників, а також підтримується провідними компаніями, що гарантує регулярні оновлення, довгострокове супроводження та інтеграцію з сучасними вебтехнологіями.

3 ОПИС ПРОГРАМИ

3.1 Структура програмного забезпечення інформаційної системи для швидкого спілкування

Структуру програмного забезпечення інформаційної системи для швидкого спілкування наведено на рис. 3.1.

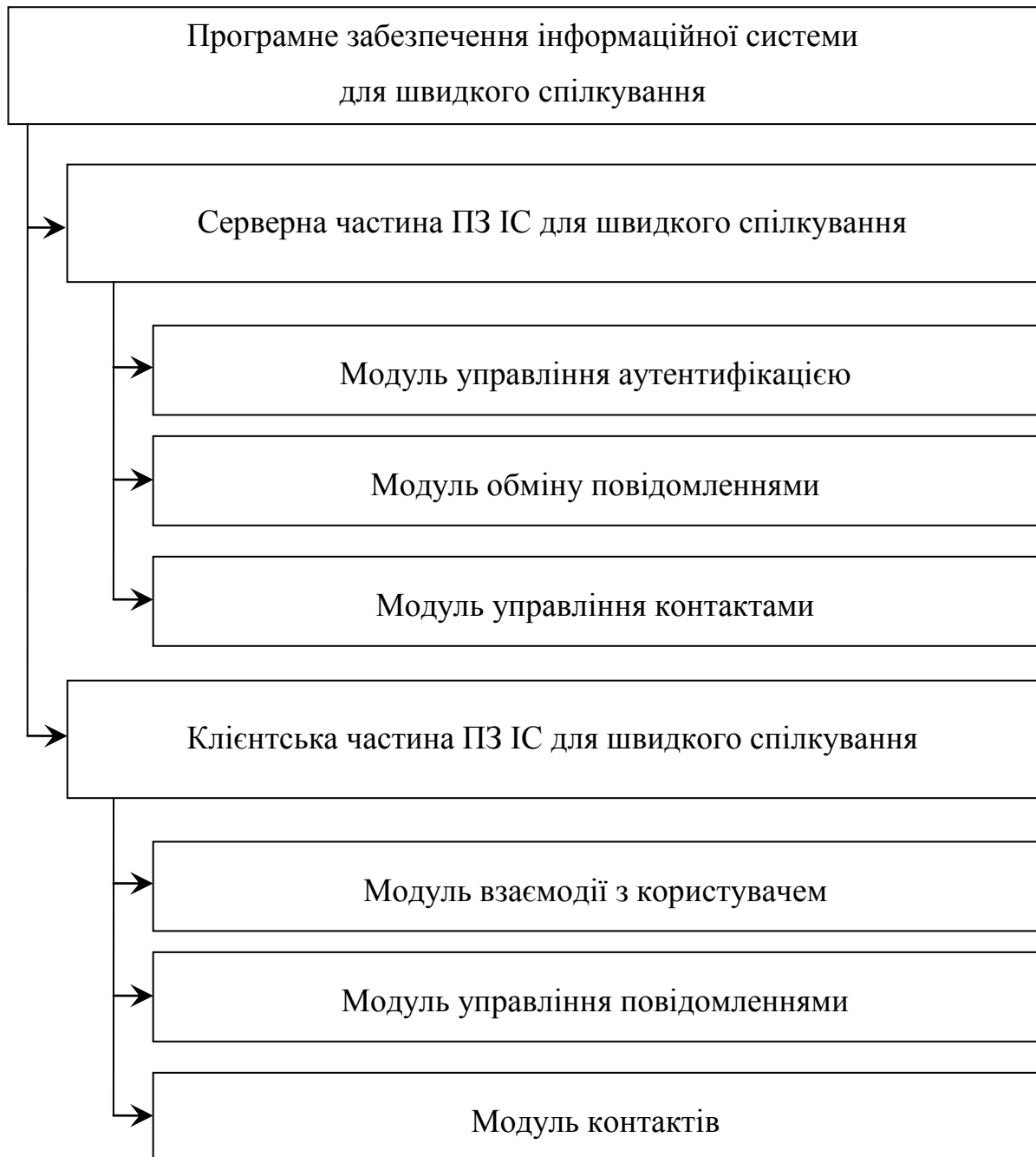


Рисунок 3.1 – Структура програмного забезпечення інформаційної системи для швидкого спілкування

Структура програмного забезпечення інформаційної системи для швидкого спілкування включає клієнтську та серверну частину.

Серверна частина програмного забезпечення інформаційної системи для швидкого спілкування включає три основні модулі, кожен з яких виконує свою функцію.

Модуль управління аутентифікацією відповідає за реєстрацію, авторизацію та перевірку користувачів. Він використовує механізми хешування паролів та токенів безпеки для захисту облікових записів.

Модуль обміну повідомленнями забезпечує роботу з реальними чатами, отримує та зберігає повідомлення, підтримує функціонал онлайн статусу та використання засобів для миттєвої доставки даних.

Модуль управління контактами відповідає за додавання, видалення та пошук користувачів, а також інтегрується з механізмами рекомендацій для швидкого знаходження знайомих або колег.

Клієнтська частина складається з трьох основних модулів, що забезпечують зручний інтерфейс користувача.

Модуль взаємодії з користувачем містить екранні компоненти для реєстрації, входу та налаштувань профілю, що дозволяють зручно керувати своїм акаунтом.

Модуль управління повідомленнями відповідає за відображення та відправлення текстових даних, оновлює список чатів у режимі реального часу, а також містить інтерфейс для пошуку та фільтрації розмов.

Модуль контактів надає зручний список друзів та можливість пошуку нових знайомих, синхронізується з бекендом для отримання актуальних даних і дозволяє додавати або блокувати контакти.

Файлова структура проєкту (рис. 3.2) організована так, щоб забезпечити зручність у розробці та підтримці. Всі зовнішні бібліотеки та залежності зберігаються в каталозі, що відповідає за пакети для Node.js, а файли, необхідні для запуску вебзастосунку, розміщені в директорії, яка містить початкові сторінки.

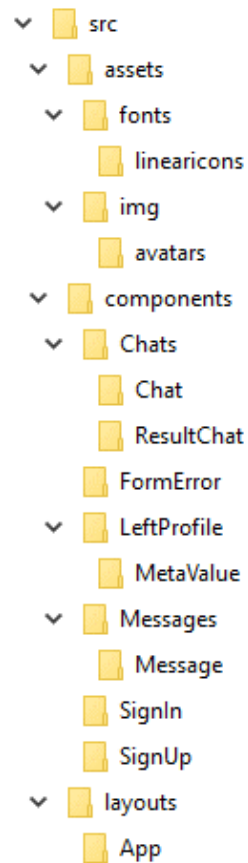


Рисунок 3.2 – Файлова структура програмного забезпечення інформаційної системи для швидкого спілкування

Серверна частина включає набір файлів (рис. 3.3), що містять налаштування, маршрутизацію, логіку роботи із сокетом для реального часу та модулі, що забезпечують коректне функціонування сервера. Всі необхідні елементи для роботи серверної частини зосереджені в спеціальному каталозі.

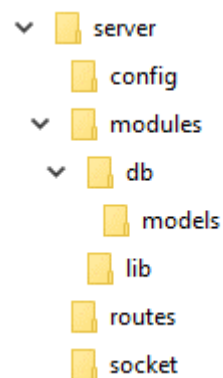


Рисунок 3.3 – Файлова структура серверної частини програмного забезпечення інформаційної системи для швидкого спілкування

Клієнтська частина містить набір ресурсів та компонентів, які забезпечують взаємодію користувачів із системою. Відповідні файли згруповані за функціональністю: окремо розташовані графічні елементи, стилі, загальні компоненти та ключові файли, які відповідають за рендеринг інтерфейсу.

Таким чином, програмне забезпечення структуроване так, щоб розділяти логіку між клієнтською та серверною частинами, що спрощує масштабування, підтримку та розширення системи.

3.2 Функціонування програмного забезпечення інформаційної системи для швидкого спілкування

Функціонування програмного забезпечення інформаційної системи для швидкого спілкування подамо за допомогою діаграм прецедентів.

На рис. 3.4 зображено діаграму прецеденту входу користувача до системи.

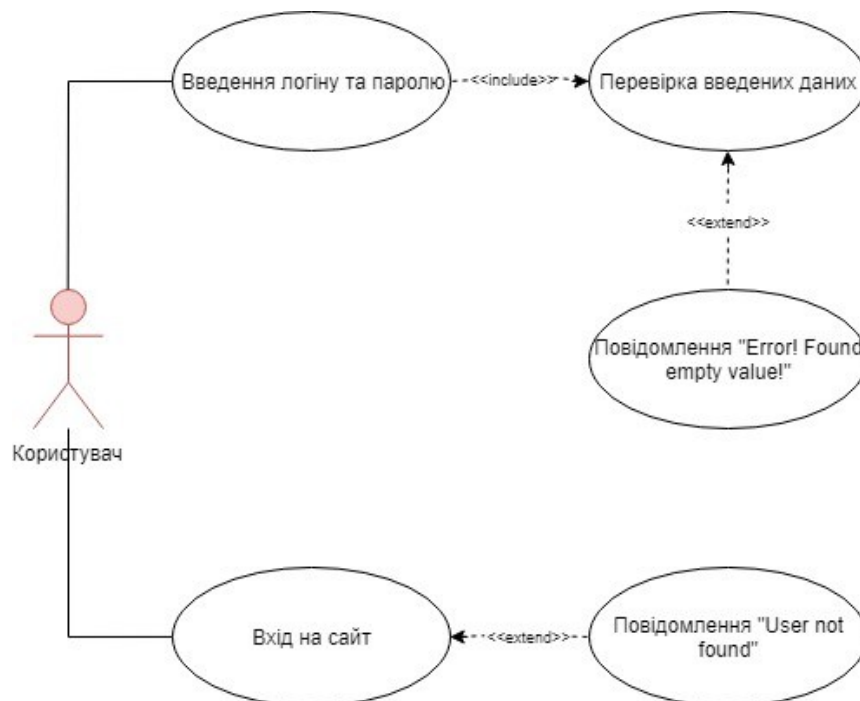


Рисунок 3.4 – Діаграма прецеденту входу користувача до програмного забезпечення інформаційної системи для швидкого спілкування

Процес входу користувача до інформаційної системи для швидкого спілкування починається з введення облікових даних. Після заповнення полів для логіна та пароля користувач ініціює перевірку, натиснувши кнопку авторизації. Потім програмне забезпечення інформаційної системи для швидкого спілкування аналізує введену інформацію, і якщо деякі обов'язкові поля залишилися порожніми, на екрані з'являється повідомлення про помилку, яке сповіщає про необхідність заповнення всіх даних перед повторним введенням.

Якщо всі дані введено коректно, програмне забезпечення інформаційної системи для швидкого спілкування порівнює їх із збереженими у своїй базі. У разі відсутності збігу користувач отримує відповідне повідомлення про помилку, після чого йому необхідно повторити спробу входу, перевіряючи правильність введених даних. Якщо ж введені облікові дані відповідають тим, що збережені у базі даних програмного забезпечення інформаційної системи для швидкого спілкування, відбувається успішна авторизація, і користувач отримує доступ до можливостей платформи.

На рис. 3.5 зображено діаграму прецеденту редагування особистих даних користувача.

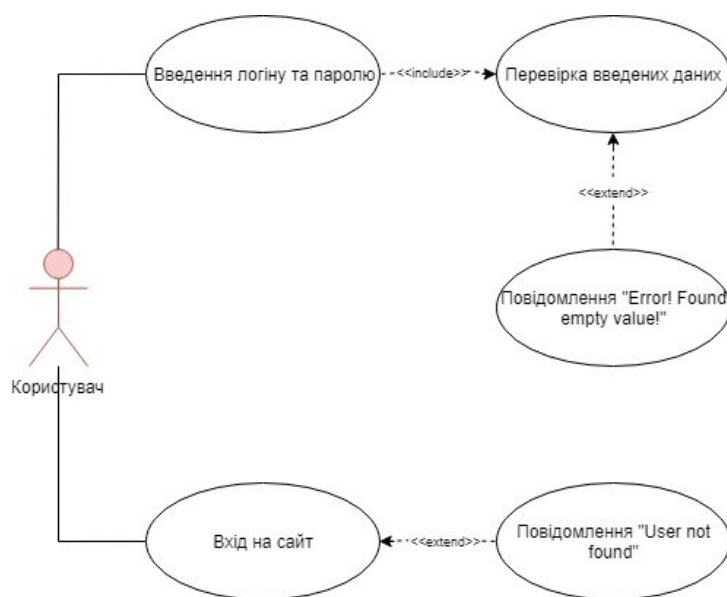


Рисунок 3.5 – Діаграма прецеденту редагування особистих даних користувача програмного забезпечення інформаційної системи для швидкого спілкування

Процес оновлення особистих даних користувача в базі даних програмного забезпечення інформаційної системи для швидкого спілкування для швидкого спілкування починається з переходу до відповідного розділу в меню налаштувань профілю. Після цього користувач програмного забезпечення інформаційної системи для швидкого спілкування активує функцію редагування, вводячи нову інформацію в запропоновані поля. Програмне забезпечення інформаційної системи для швидкого спілкування аналізує отримані дані на предмет коректності та перевіряє можливість їх зміни. Якщо виявлено некоректні значення або певні дані не можуть бути оновлені, користувач отримує повідомлення про помилку та повертається до початкового етапу редагування.

У разі успішної перевірки програмне забезпечення інформаційної системи для швидкого спілкування ініціює процес оновлення інформації, зберігаючи зміни в базі даних та відображаючи оновлені особисті дані в профілі користувача.

Після успішної авторизації та, за потреби, оновлення особистих даних, користувач переходить до основного процесу взаємодії в інформаційній системі для швидкого спілкування. На головному екрані програмного забезпечення інформаційної системи для швидкого спілкування відображається список доступних контактів. Користувач може обрати вже існуючий чат або розпочати новий діалог, знайшовши потрібного співрозмовника через пошук.

При надсиланні повідомлення введений текст передається на сервер програмного забезпечення інформаційної системи для швидкого спілкування. Після успішної перевірки повідомлення записується в базу даних та надсилається отримувачу. Якщо адресат онлайн, повідомлення одразу відображається у його вікні чату, а також програмне забезпечення інформаційної системи для швидкого спілкування може надіслати звукове або візуальне сповіщення. Якщо ж користувач не в мережі, повідомлення зберігається на сервері та буде доставлене при наступному підключенні.

У разі отримання нового повідомлення програмне забезпечення інформаційної системи для швидкого спілкування негайно оновлює інтерфейс, додаючи його до потоку чату. Реакції користувачів, такі як смайлики, відповіді чи пересилання, також обробляються сервером та синхронізуються між усіма пристроями.

У будь-який момент користувач програмного забезпечення інформаційної системи для швидкого спілкування може вийти з чату.

3.3 Розробка бази даних програмного забезпечення інформаційної системи для швидкого спілкування

База даних програмного забезпечення інформаційної системи для швидкого спілкування складається з кількох основних таблиць, що забезпечують збереження та швидке отримання даних для обробки повідомлень, контактів і профілів користувачів. База даних програмного забезпечення інформаційної системи для швидкого спілкування містить інформацію про користувачів, повідомлення, контакти, групові чати, а також додаткові параметри, які забезпечують зручність використання та безпеку взаємодії. Вона організована таким чином, щоб забезпечити ефективне зберігання інформації, її швидкий пошук та взаємодію між користувачами.

Таблиця користувачів містить унікальні ідентифікатори, логіни, паролі (у зашифрованому вигляді), імена, електронні адреси, аватари та статуси онлайн присутності. Вона також містить додаткові налаштування, такі як конфіденційність або налаштування сповіщень.

Таблиця чату зберігає інформацію про створені діалоги. Кожен чат має унікальний ідентифікатор, тип (приватний або груповий), список учасників та часову мітку останньої активності.

Таблиця повідомлень пов'язана з чатами та зберігає текстові повідомлення, їхніх авторів, часові позначки, статус доставки (надіслано, отримано, прочитано) та можливі вкладення, наприклад, файли чи

зображення. У разі видалення повідомлення може бути передбачене поле, що вказує на його статус.

Окремо ведеться таблиця контактів, у якій зберігаються зв'язки між користувачами, відображаючи список друзів або заблокованих осіб. Для групових чатів використовується таблиця, що містить дані про створені чати, їх учасників, адміністраторів, а також історію змін у складі групи.

Крім основних даних, база містить таблиці зі списками налаштувань користувачів, такими як параметри сповіщень, вибір теми оформлення та персональні налаштування конфіденційності. З метою підвищення продуктивності часто використовувані дані можуть кешуватися, а всі операції з критичними даними ведуться з обліком транзакцій для запобігання втратам або некоректному оновленню записів.

Зв'язки між таблицями бази даних інформаційної системи для швидкого спілкування організовані відповідно до її логічної структури.

Таблиця користувачів має первинний ключ `user_id`, який використовується як зовнішній ключ у кількох інших таблицях. Зв'язок між користувачами та їхніми контактами реалізується через таблицю "контакти", де кожен запис містить `user_id` та `contact_id`, створюючи відношення "багато до багатьох" між користувачами.

Таблиця повідомлень зберігає `message_id` як первинний ключ і має зовнішні ключі `sender_id` та `receiver_id`, які посилаються на `user_id` з таблиці користувачів. Це створює зв'язок "один до багатьох", оскільки один користувач може надсилати багато повідомлень, але кожне повідомлення належить лише одному відправнику та отримувачу.

Таблиця групових чатів містить `chat_id` як первинний ключ, а зв'язок між чатами та їхніми учасниками реалізується через проміжну таблицю "учасники_чату", що містить `chat_id` та `user_id`, забезпечуючи відношення "багато до багатьох" (M:N) між користувачами та групами.

Налаштування користувачів знаходяться у таблиці "налаштування", що містить `settings_id` як первинний ключ і `user_id` як зовнішній, встановлюючи

зв'язок "один до одного" між користувачем і його персональними параметрами.

Таким чином, структура зв'язків бази даних забезпечує ефективне управління даними, підтримку багатокористувацької взаємодії та швидку обробку запитів.

3.4 Проєктування інтерфейсу програмного забезпечення інформаційної системи для швидкого спілкування

Проєктування інтерфейсу взаємодії користувача з програмним забезпеченням інформаційної системи для швидкого спілкування виконано з урахуванням вимог технічного завдання. При розробленні програмного забезпечення інформаційної системи для швидкого спілкування враховані функціональні вимоги до програми:

- підтримка можливості швидкого спілкування між користувачами за допомогою текстових повідомлень;
- підтримка можливості пошуку контактів користувачів, зареєстрованих у системі;
- можливість реєстрації та авторизації користувачів для подальшого спілкування;
- можливість редагування особистих даних користувача, зокрема, номеру телефону та імені;
- підтримка можливості додавання контактів для спілкування;
- підтримка можливості отримання повідомлень від інших користувачів, зареєстрованих у системі.

Інтерфейс головної форми програмного забезпечення інформаційної системи для швидкого спілкування наведено на рис. 3.6.

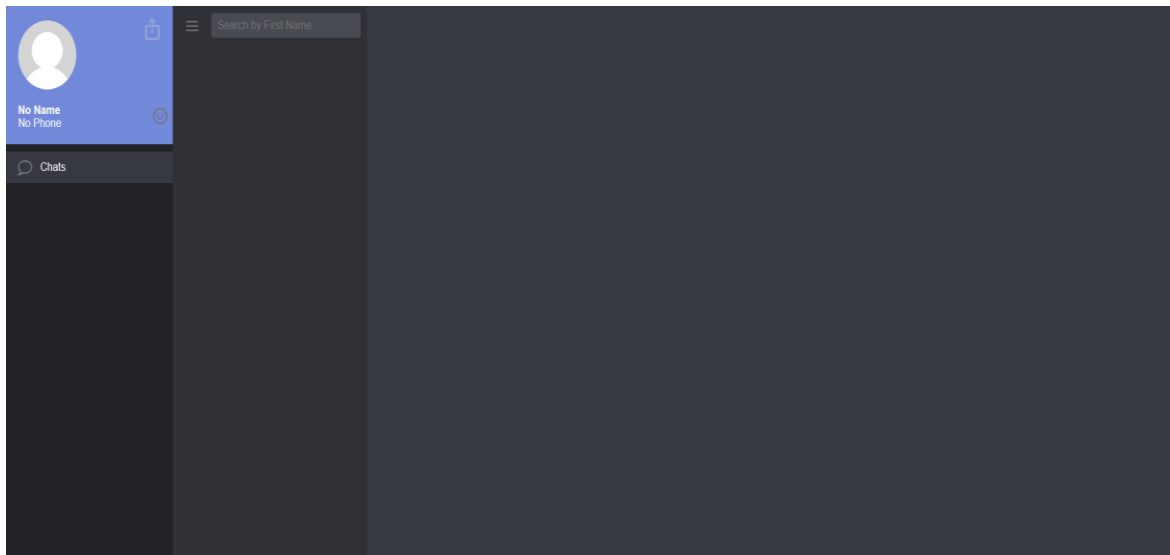


Рисунок 3.6 – Інтерфейс головної сторінки програмного забезпечення інформаційної системи для швидкого спілкування

Інтерфейс сторінки авторизації програмного забезпечення інформаційної системи для швидкого спілкування наведено на рис. 3.7.

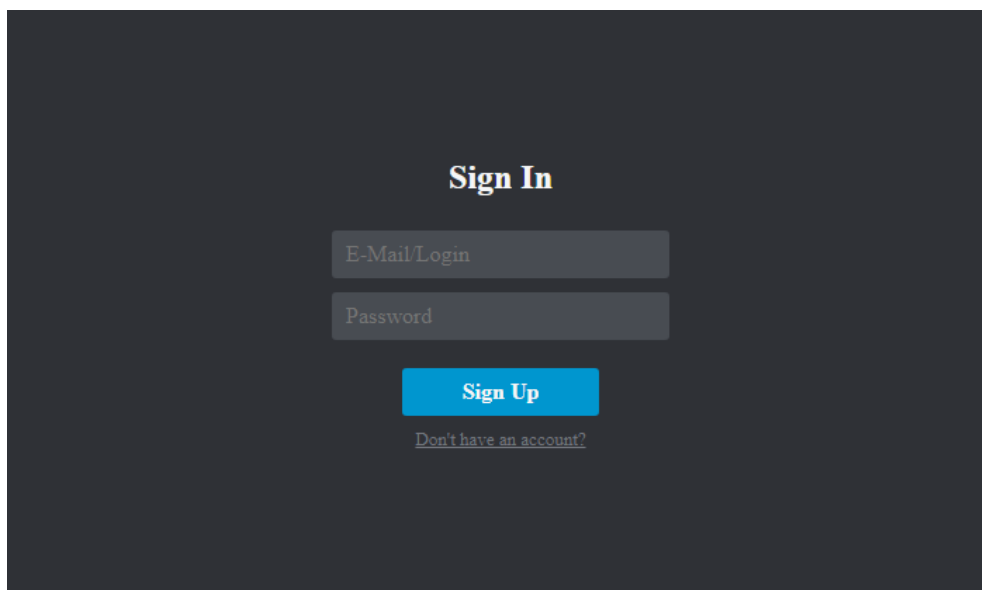
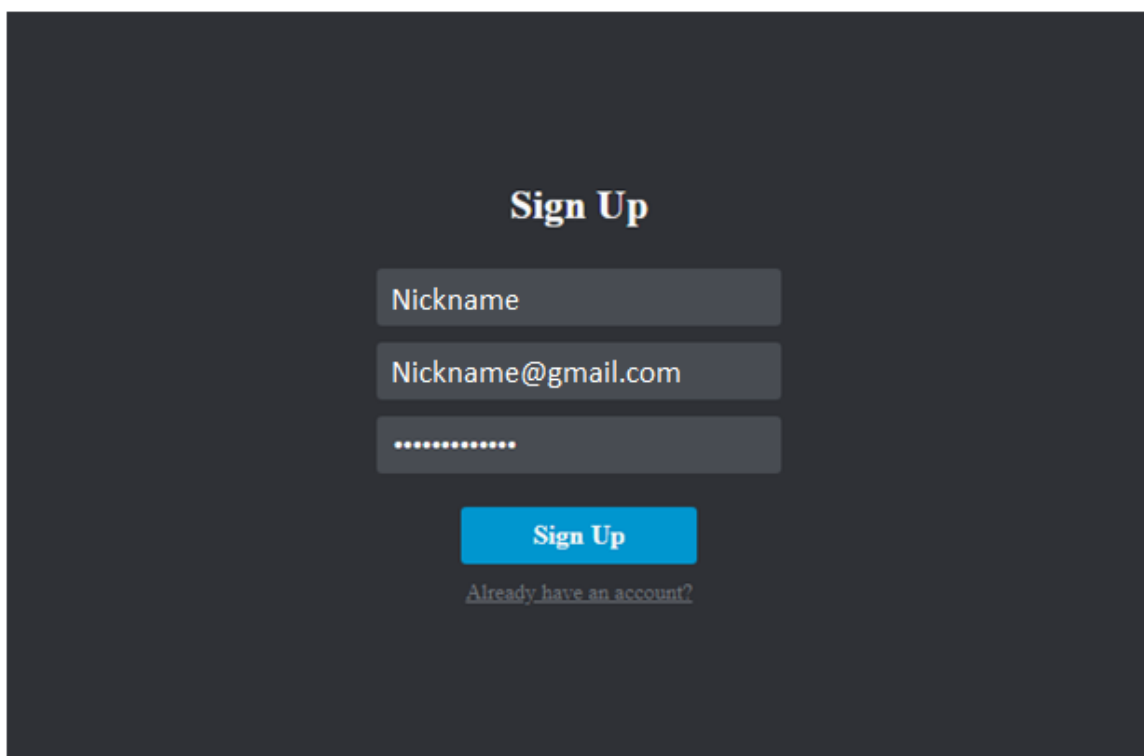


Рисунок 3.7 – Інтерфейс сторінки авторизації програмного забезпечення інформаційної системи для швидкого спілкування

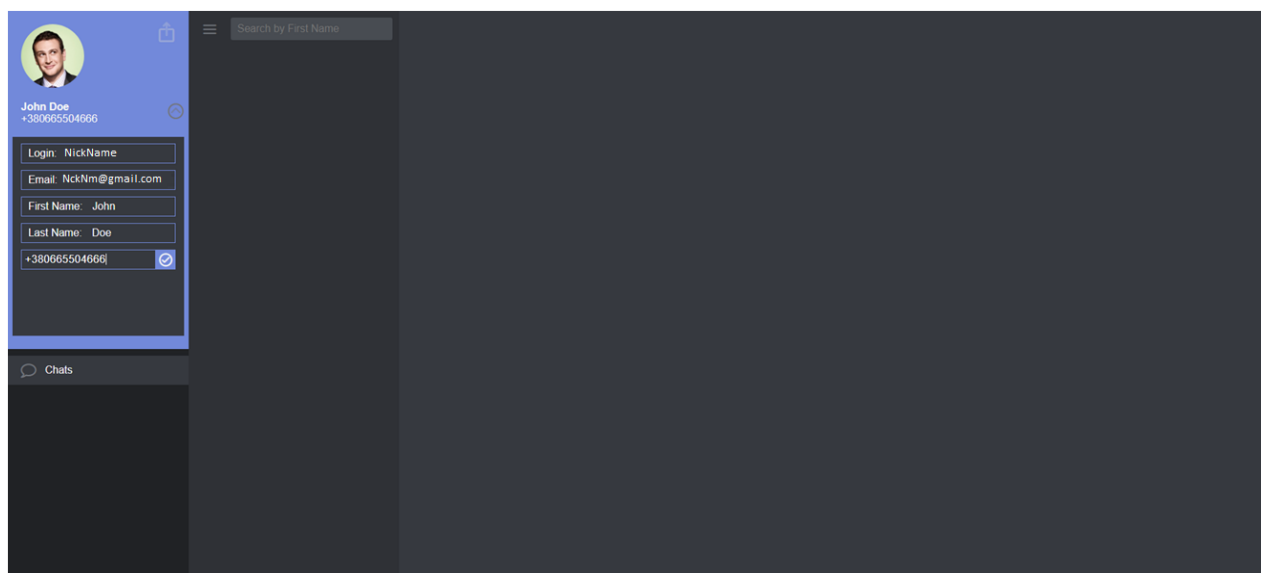
Інтерфейс сторінки реєстрації у програмному забезпечення інформаційної системи для швидкого спілкування наведено на рис. 3.8.



The image shows a 'Sign Up' form on a dark background. The form consists of three input fields: 'Nickname', 'Email' (containing 'Nickname@gmail.com'), and 'Password' (represented by dots). Below the fields is a blue 'Sign Up' button and a link that says 'Already have an account?'.

Рисунок 3.8 – Інтерфейс сторінки реєстрації програмного забезпечення інформаційної системи для швидкого спілкування

Область зміни персональних даних наведено на рис. 3.9.



The image shows a user profile interface. On the left, there is a profile card for 'John Doe' with a phone number '+380665504666'. Below the profile card is a 'Chats' section. The main area of the interface is a dark grey panel with a search bar at the top labeled 'Search by First Name'. Below the search bar, there is a list of input fields for updating personal data: 'Login: NickName', 'Email: NckNm@gmail.com', 'First Name: John', 'Last Name: Doe', and a phone number field '+380665504666' with a checkmark icon.

Рисунок 3.9 – Область зміни персональних даних

Сторінку з результатами спілкування наведено на рис. 3.10.

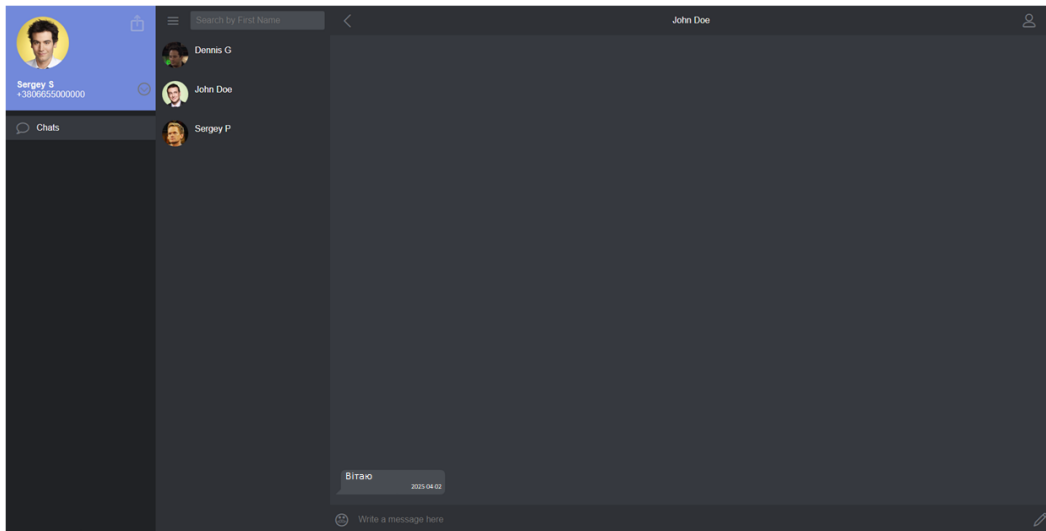


Рисунок 3.10 – Сторінка з результатами спілкування

3.5 Висновки за розділом 3

Розроблено програмне забезпечення інформаційної системи для швидкого спілкування.

Запропоновано структуру програмного забезпечення інформаційної системи для швидкого спілкування. Визначено, що структура програмного забезпечення інформаційної системи для швидкого спілкування включає клієнтську та серверну частину. Серверна частина програмного забезпечення інформаційної системи для швидкого спілкування включає модулі управління аутентифікацією, обміну повідомленнями та управління контактами. Клієнтська частина складається з трьох основних модулів (модуль взаємодії з користувачем, модуль управління повідомленнями, модуль контактів), що забезпечують зручний інтерфейс користувача. Розподіл логіки роботи програмного забезпечення інформаційної системи для швидкого спілкування між клієнтською та серверною частинами спрощує масштабування, підтримку та розширення системи.

Розроблено базу даних програмного забезпечення інформаційної системи для швидкого спілкування, яка складається з кількох основних таблиць, що забезпечують збереження та швидке отримання даних для

обробки повідомлень, контактів і профілів користувачів. База даних програмного забезпечення інформаційної системи для швидкого спілкування містить інформацію про користувачів, повідомлення, контакти, групові чати, а також додаткові параметри, які забезпечують зручність використання та безпеку взаємодії. Вона організована таким чином, щоб забезпечити ефективне зберігання інформації, її швидкий пошук та взаємодію між користувачами.

Описано функціонування програмного забезпечення інформаційної системи для швидкого спілкування.

Описано особливості реалізації програмного забезпечення інформаційної системи для швидкого спілкування. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням інформаційної системи для швидкого спілкування.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Призначення й умови застосування програми

Програма призначена для підтримки процесу функціонування для швидкого спілкування. Програма написана на мові JavaScript, оскільки ця мова забезпечує високу продуктивність, гнучкість та зручність у розробці вебзастосунків у реальному часі. Для створення програмного забезпечення інформаційної системи для швидкого спілкування обрано бібліотеку ReactJS, що забезпечує зручну модульну структуру розроблюваної програми через компонентний підхід, що дає змогу легко розширювати систему, додаючи нові функціональні можливості без значного переписування коду.

Програмне забезпечення інформаційної системи для швидкого спілкування забезпечує виконання таких функцій:

- підтримка можливості швидкого спілкування між користувачами за допомогою текстових повідомлень;
- підтримка можливості пошуку контактів користувачів, зареєстрованих у системі;
- можливість реєстрації та авторизації користувачів для подальшого спілкування;
- можливість редагування особистих даних користувача, зокрема, номеру телефону та імені;
- підтримка можливості додавання контактів для спілкування;
- підтримка можливості отримання повідомлень від інших користувачів, зареєстрованих у системі.

Для роботи програмного забезпечення інформаційної системи для швидкого спілкування потрібні такі технічні та програмні ресурси, як сервер для обробки запитів користувачів у реальному часі, високошвидкісне підключення до мережі для забезпечення мінімальної затримки під час передавання повідомлень, а також системи балансування навантаження для

рівномірного розподілу трафіку між серверами. З програмного боку необхідні серверні технології для асинхронної обробки подій, система управління базами даних (наприклад, PostgreSQL або MongoDB) для збереження історії повідомлень і даних користувачів, а також вебфреймворк ReactJS для підтримки роботи та модифікації (за необхідності) інтерактивного клієнтського інтерфейсу.

Функціональні характеристики розробленого програмного забезпечення інформаційної системи для швидкого спілкування наведено у технічному завданні (додаток А). Фрагмент тексту програмного забезпечення інформаційної системи для швидкого спілкування наведено у додатку Б.

4.2 Характеристики програмного забезпечення інформаційної системи для швидкого спілкування

Програмне забезпечення інформаційної системи для швидкого спілкування складається з клієнтської та серверної частин. Серверна частина програмного забезпечення інформаційної системи для швидкого спілкування включає модулі управління аутентифікацією, обміну повідомленнями та управління контактами. Клієнтська частина складається з трьох основних модулів (модуль взаємодії з користувачем, модуль управління повідомленнями, модуль контактів), що забезпечують зручний інтерфейс користувача. Розподіл логіки роботи програмного забезпечення інформаційної системи для швидкого спілкування між клієнтською та серверною частинами спрощує масштабування, підтримку та розширення системи.

Базу даних програмного забезпечення інформаційної системи для швидкого спілкування складається з кількох основних таблиць, що забезпечують збереження та швидке отримання даних для обробки повідомлень, контактів і профілів користувачів. База даних програмного забезпечення інформаційної системи для швидкого спілкування містить інформацію про користувачів, повідомлення, контакти, групові чати, а також

додаткові параметри, які забезпечують зручність використання та безпеку взаємодії. Вона організована таким чином, щоб забезпечити ефективне зберігання інформації, її швидкий пошук та взаємодію між користувачами.

Програмне забезпечення інформаційної системи для швидкого спілкування має ряд важливих характеристик, що визначають його ефективність та зручність у використанні. Висока продуктивність забезпечує швидку обробку повідомлень та мінімальні затримки у передачі даних між користувачами. Надійність і стійкість до збоїв гарантують безперебійну роботу системи навіть при високому навантаженні або короткочасних збоях у мережі. Масштабованість дозволяє системі адаптуватися до збільшення кількості користувачів і підвищення навантаження без втрати продуктивності.

Гнучкість архітектури забезпечує можливість розширення функціоналу, інтеграції з іншими сервісами та адаптації до змін у вимогах користувачів. Зручність інтерфейсу спрощує взаємодію користувачів із системою, роблячи процес комунікації швидким і зрозумілим.

Підтримка роботи в реальному часі дозволяє користувачам отримувати миттєві сповіщення про нові повідомлення та зміни в чатах. Кросплатформеність забезпечує коректну роботу системи на різних пристроях та операційних системах, що дозволяє користувачам спілкуватися як із мобільних пристроїв, так і з персональних комп'ютерів.

4.3 Інструкція по експлуатації програми

4.3.1 Звернення до програми

Для запуску програмного забезпечення інформаційної системи для швидкого спілкування на клієнтській стороні необхідно у рядку браузера ввести відповідну адресу. При цьому серверна частина програмного забезпечення інформаційної системи для швидкого спілкування повинна бути запущена на серверному обладнанні.

4.3.2 Вхідні й вихідні дані

Вхідними даними до програмного забезпечення інформаційної системи для швидкого спілкування є реєстраційні дані користувача, інформація для пошуку чатів та інших користувачів, повідомлення іншим користувачам.

Вихідними даними програмного забезпечення інформаційної системи для швидкого спілкування є повідомлення від інших користувачів.

4.3.3 Повідомлення

Користувач програмного забезпечення інформаційної системи для швидкого спілкування може отримати такі повідомлення: повідомлення про успішну реєстрацію, повідомлення про некоректні дані.

4.4 Виконання програмного забезпечення інформаційної системи для швидкого спілкування

Після запуску програмного забезпечення інформаційної системи для швидкого спілкування користувачу відображається сторінка авторизації (рис. 4.1), де необхідно ввести логін та пароль.

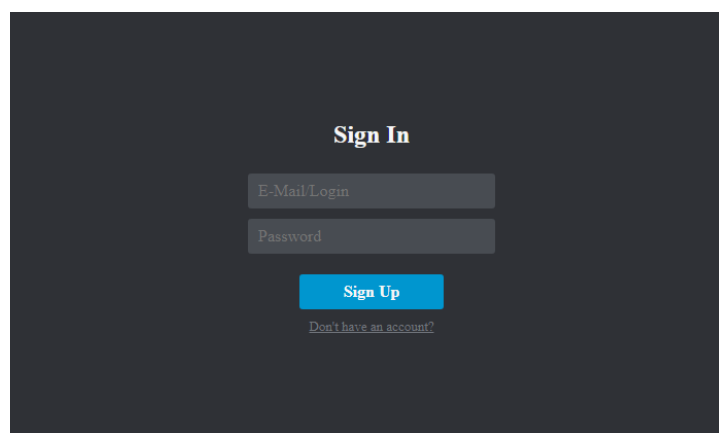


Рисунок 4.1 – Інтерфейс сторінки авторизації програмного забезпечення інформаційної системи для швидкого спілкування

Якщо користувач не проходив процедуру реєстрації, її необхідно пройти, ввівши потрібні дані на відповідній сторінці (рис. 4.2).

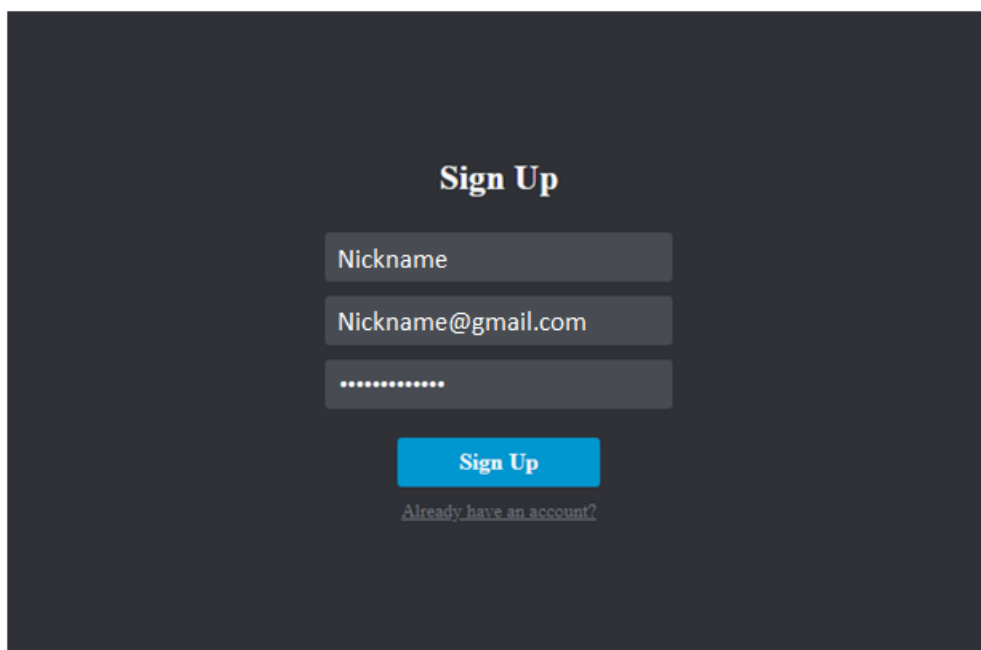
The image shows a 'Sign Up' form on a dark background. At the top, the text 'Sign Up' is displayed in a light blue font. Below it are three input fields: the first is labeled 'Nickname', the second contains the text 'Nickname@gmail.com', and the third is a password field with masked characters (dots). A blue button with the text 'Sign Up' is positioned below the input fields. At the bottom, there is a link that reads 'Already have an account?' in a small, light blue font.

Рисунок 4.2 – Інтерфейс сторінки реєстрації програмного забезпечення інформаційної системи для швидкого спілкування

Після успішної авторизації користувачу відображається головна сторінка (рис. 4.3), де зліва знаходиться список чатів користувача, його профіль, а також поле для пошуку контактів для спілкування.

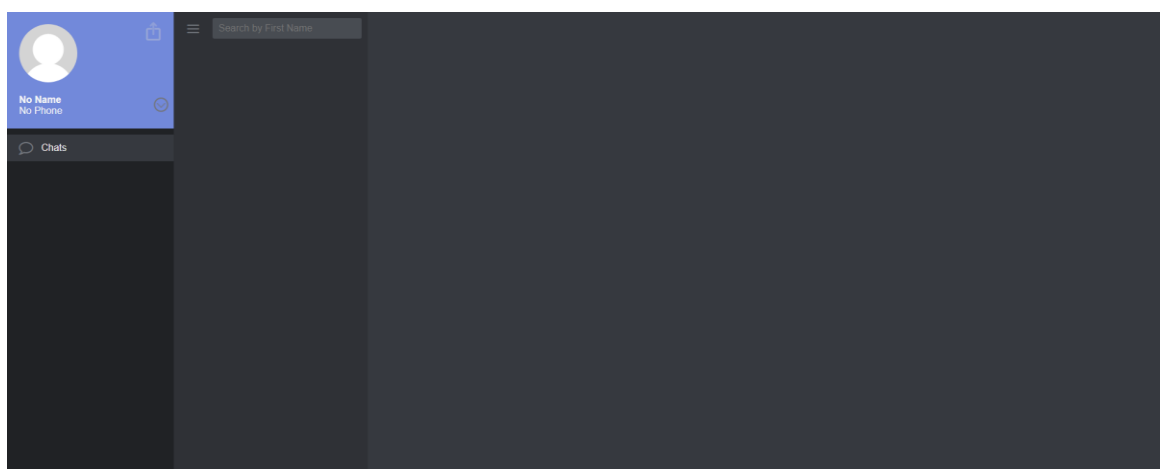


Рисунок 4.3 – Головна сторінка програмного забезпечення інформаційної системи для швидкого спілкування

Користувач має можливість змінити свої персональні дані (рис. 4.4) у відповідній області, ввівши ім'я, номер телефону та, за необхідності, деяку іншу інформацію.

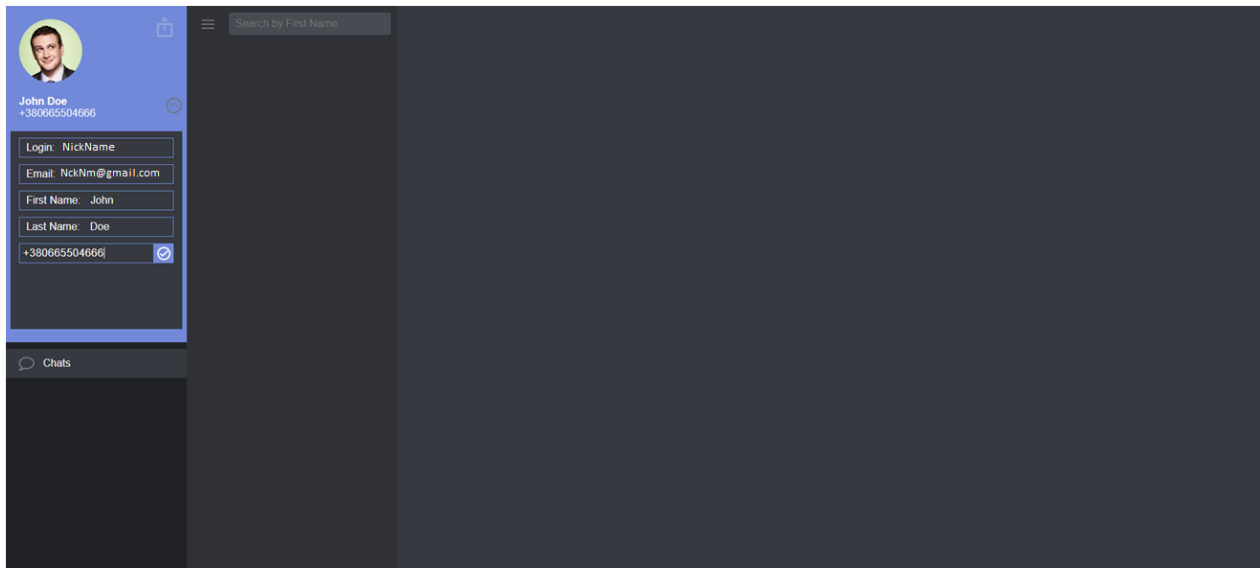


Рисунок 4.4 – Зміна персональних даних користувача

Користуючись полем для пошуку (рис. 4.5) існує можливість пошуку та додавання контактів для спілкування.

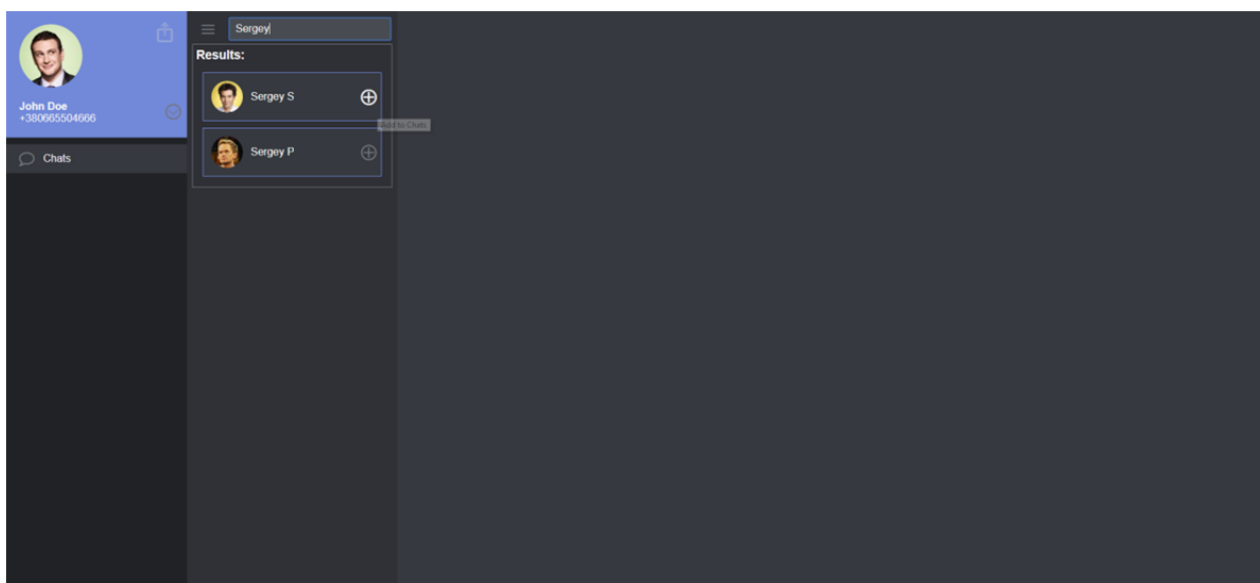


Рисунок 4.5 – Пошук та додавання контактів для спілкування

Для спілкування необхідно вибрати користувача та ввести текст повідомлення. Після вводу повідомлення шляхом натискання кнопки Вводу на клавіатурі або візуальної кнопки з зображенням олівця, повідомлення буде надіслано іншому користувачу (рис. 4.6).

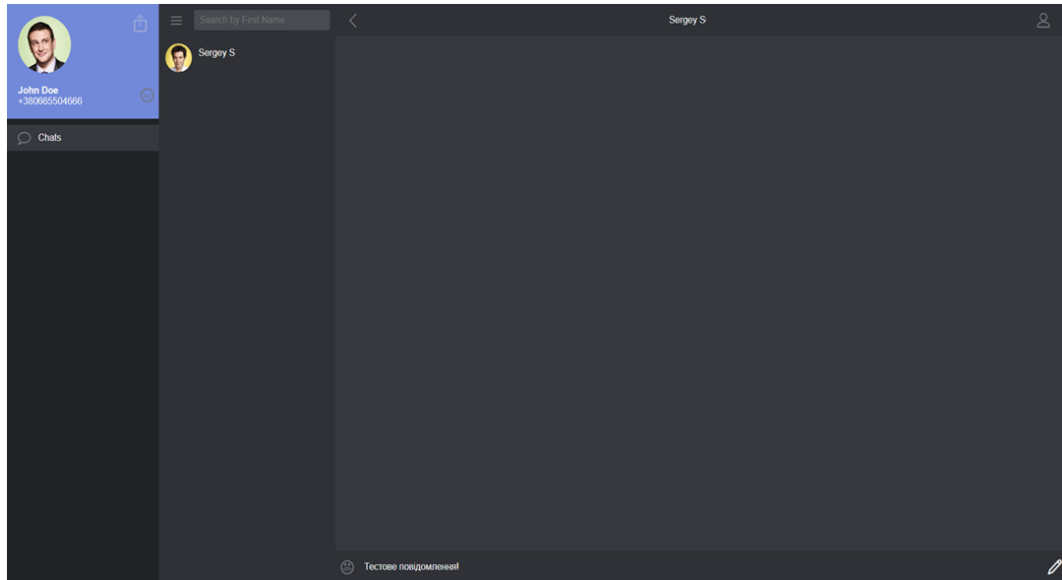


Рисунок 4.6 – Сторінка з областю для відправлення повідомлень іншим користувачам

Інший користувач отримає повідомлення на своєму пристрої (рис. 4.7).

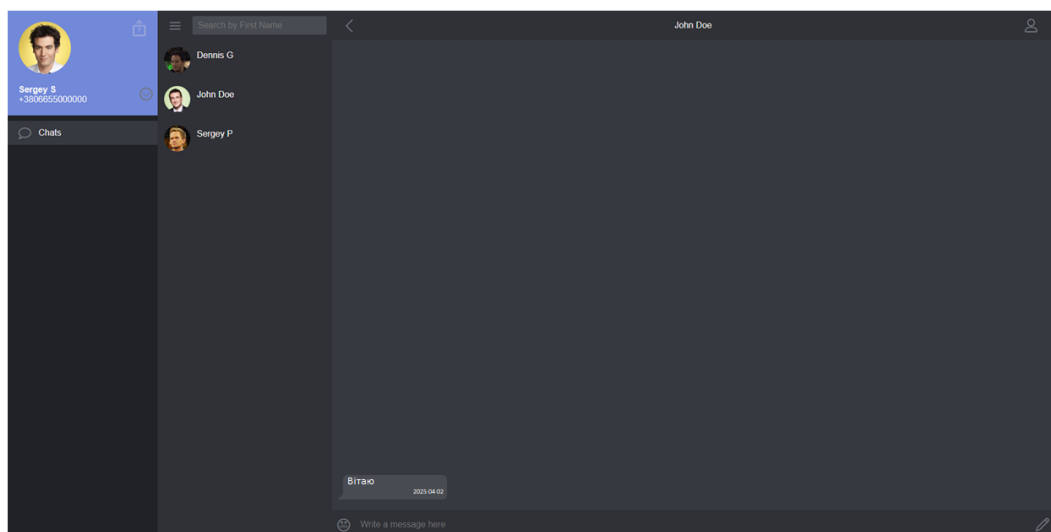


Рисунок 4.7 – Отримання повідомлення іншим користувачем

4.5 Тестування програмного забезпечення інформаційної системи для швидкого спілкування

Проведено тестування програмного забезпечення інформаційної системи для швидкого спілкування.

Підтверджено, що програмне забезпечення інформаційної системи для швидкого спілкування працює коректно і стабільно, виконуючи всі необхідні функціональні вимоги та успішно справляючись із основною задачею підтримки процесу комунікації між користувачами.

Розроблене програмне забезпечення забезпечує автоматизацію операцій інформаційної системи, пов'язаних із підтримкою процесів комунікації між користувачами.

4.6 Висновки за розділом 4

Описано програмне забезпечення інформаційної системи для швидкого спілкування.

Проведено тестування створеного програмного забезпечення інформаційної системи для швидкого спілкування. Результати тестування показали, що програмне забезпечення інформаційної системи для швидкого спілкування забезпечує автоматизацію процесів, пов'язаних із підтримкою процесів комунікації між користувачами.

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес розробки інформаційної системи для спілкування.

Відзначено, що інформаційні системи для швидкого спілкування забезпечують оперативний обмін повідомленнями між користувачами в режимі реального часу. Вони використовують мережеві технології для передачі текстової, голосової та відеоінформації, що дозволяє ефективно підтримувати комунікацію незалежно від відстані.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато інформаційних систем для швидкого спілкування. Проте деякі інформаційні системи для швидкого спілкування є недостатньо надійними через проблеми з безпекою та конфіденційністю даних. Проблеми сумісності між різними платформами можуть ускладнювати інтеграцію таких систем у робочі процеси. Деякі месенджери працюють лише у певних екосистемах, що змушує користувачів адаптуватися до конкретного програмного забезпечення або використовувати кілька інструментів одночасно. Це може викликати незручності та ускладнювати обмін файлами, контактами та іншими даними між різними сервісами. Також варто враховувати технічні обмеження та вимоги до ресурсів пристрою. Деякі сучасні програми для швидкого спілкування споживають значну кількість оперативної пам'яті та процесорних ресурсів, що може сповільнювати роботу пристроїв, особливо старіших моделей. Окрім цього, монетизаційні моделі деяких програмних засобів можуть створювати дискомфорт для користувачів. Безкоштовні месенджери часто містять рекламу, а платні версії вимагають додаткових витрат. Деякі сервіси обмежують функціонал у безкоштовних версіях, що змушує користувачів купувати підписку або переходити на інші платформи. Тому актуальною є розробка програмного забезпечення інформаційної системи для швидкого спілкування.

Сформульовано функціональні вимоги до програмного забезпечення інформаційної системи для швидкого спілкування.

Для реалізації програмного забезпечення інформаційної системи для швидкого спілкування обрано мову програмування JavaScript, оскільки ця мова забезпечує високу продуктивність, гнучкість та зручність у розробці вебзастосунків у реальному часі. Крім того, JavaScript є універсальною мовою, яка працює як у браузерях, так і на сервері, що дозволяє створити єдиний стек технологій для клієнтської та серверної частин програми, що спрощує розробку та підтримку інформаційної системи. Іншою перевагою є те, що JavaScript має широкий набір інструментів для швидкої розробки чатів та інформаційних систем.

Для створення програмного забезпечення інформаційної системи для швидкого спілкування обрано бібліотеку ReactJS, що забезпечує зручну модульну структуру розроблюваної програми через компонентний підхід, що дає змогу легко розширювати систему, додаючи нові функціональні можливості без значного переписування коду. Крім того, ReactJS має велику спільноту розробників, а також підтримується провідними компаніями, що гарантує регулярні оновлення, довгострокове супроводження та інтеграцію з сучасними вебтехнологіями.

Розроблено програмне забезпечення інформаційної системи для швидкого спілкування.

Запропоновано структуру програмного забезпечення інформаційної системи для швидкого спілкування. Визначено, що структура програмного забезпечення інформаційної системи для швидкого спілкування включає клієнтську та серверну частину. Серверна частина програмного забезпечення інформаційної системи для швидкого спілкування включає модулі управління аутентифікацією, обміну повідомленнями та управління контактами. Клієнтська частина складається з трьох основних модулів (модуль взаємодії з користувачем, модуль управління повідомленнями, модуль контактів), що забезпечують зручний інтерфейс користувача. Розподіл логіки роботи

програмного забезпечення інформаційної системи для швидкого спілкування між клієнтською та серверною частинами спрощує масштабування, підтримку та розширення системи.

Розроблено базу даних програмного забезпечення інформаційної системи для швидкого спілкування, яка складається з кількох основних таблиць, що забезпечують збереження та швидке отримання даних для обробки повідомлень, контактів і профілів користувачів. База даних програмного забезпечення інформаційної системи для швидкого спілкування містить інформацію про користувачів, повідомлення, контакти, групові чати, а також додаткові параметри, які забезпечують зручність використання та безпеку взаємодії. Вона організована таким чином, щоб забезпечити ефективне зберігання інформації, її швидкий пошук та взаємодію між користувачами.

Описано функціонування програмного забезпечення інформаційної системи для швидкого спілкування.

Описано особливості реалізації програмного забезпечення інформаційної системи для швидкого спілкування. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням інформаційної системи для швидкого спілкування.

Проведено тестування створеного програмного забезпечення інформаційної системи для швидкого спілкування. Результати тестування показали, що програмне забезпечення інформаційної системи для швидкого спілкування забезпечує автоматизацію процесів, пов'язаних із підтримкою процесів комунікації між користувачами.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. What is Instant Messaging [Electronic resource]. – Access mode: <https://www.geeksforgeeks.org/what-is-instant-messaging/>.
2. The Evolution of Instant Messaging: A Comprehensive Guide [Electronic resource]. – Access mode: <https://nativemsg.com/resources/text-marketing/the-evolution-of-instant-messaging-a-comprehensive-guide/>.
3. What is Instant Messaging? Top Examples & Benefits [Electronic resource]. – Access mode: <https://cyberinsight.co/what-is-instant-messaging-examples/>.
4. What is Instant Messaging & How Does IM Work with Examples [Electronic resource]. – Access mode: <https://www.revechat.com/blog/instant-messaging/>.
5. Instant Communication [Electronic resource]. – Access mode: <https://www.sciencedirect.com/topics/computer-science/instant-communication>.
6. How Instant Messaging Works [Electronic resource]. – Access mode: <https://computer.howstuffworks.com/e-mail-messaging/instant-messaging.htm>.
7. Ultimate Guide to What Is Instant Messaging [Electronic resource]. – Access mode: <https://trtc.io/blog/details/what-is-instant-messaging>.
8. Best Instant Messaging Software [Electronic resource]. – Access mode: <https://fixthephoto.com/best-instant-messaging-software.html>.
9. Maximize Efficiency: Best 15 Instant Messaging Platforms For Business Communication [Electronic resource]. – Access mode: <https://blog.rannkly.com/instant-messaging-platforms-for-business>.
10. Best Instant Messaging Software [Electronic resource]. – Access mode: <https://financesonline.com/c/instant-messaging-software/>.
11. Best Business Instant Messaging Software [Electronic resource]. – Access mode: <https://www.g2.com/categories/business-instant-messaging>.
12. Twist [Electronic resource]. – Access mode: <https://twist.com/>.
13. DocsBot AI [Electronic resource]. – Access mode: <https://docsbot.ai/>.

14. SleekFlow [Electronic resource]. – Access mode: <https://sleekflow.io/>.
15. JavaScript Tutorial [Electronic resource]. – Access mode: <https://www.javascripttutorial.net/>.
16. JavaScript [Electronic resource]. – Access mode: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>.
17. React Community [Electronic resource]. – Access mode: <https://react.dev/community>.
18. React Tutorial [Electronic resource]. – Access mode: <https://www.w3schools.com/react/>.

ДОДАТОК А
Технічне завдання

Вступ

Програмне забезпечення може використовуватися для підтримки процесу спілкування між користувачами.

A.1 Підстава для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Програмне забезпечення для швидкого спілкування», затверджене наказом Національного університету «Запорізька політехніка» № 209 від 28 квітня 2025 р.

A.2 Призначення розробки

Програмний продукт призначений для підтримки процесу швидкого спілкування.

A.3 Основні вимоги до програми, що розробляється

A.3.1 Вимоги до функціональних характеристик

Програмне забезпечення інформаційної системи для швидкого спілкування забезпечує виконання таких функцій:

- підтримка можливості швидкого спілкування між користувачами за допомогою текстових повідомлень;
- підтримка можливості пошуку контактів користувачів, зареєстрованих у системі;
- можливість реєстрації та авторизації користувачів для подальшого спілкування;
- можливість редагування особистих даних користувача, зокрема, номеру телефону та імені;

- підтримка можливості додавання контактів для спілкування;
- підтримка можливості отримання повідомлень від інших користувачів, зареєстрованих у системі.

А.3.2 Вимоги до інтерфейсу програми

Інтерфейс програмного забезпечення інформаційної системи для швидкого спілкування повинен бути зручним для користувачів. Повинна бути забезпечена можливість роботи в візуальному режимі.

А.3.3 Вимоги до надійності

Програмне забезпечення інформаційної системи для швидкого спілкування повинно забезпечити надійне функціонування.

А.3.4 Умови експлуатації

Для експлуатації програмного забезпечення інформаційної системи для швидкого спілкування необхідна наявність персонального комп'ютера.

А.3.5 Вимоги до складу та параметрів технічний засобів

Для роботи програмного забезпечення інформаційної системи для швидкого спілкування потрібні такі технічні та програмні ресурси, як сервер для обробки запитів користувачів у реальному часі, високошвидкісне підключення до мережі для забезпечення мінімальної затримки під час передавання повідомлень, а також системи балансування навантаження для рівномірного розподілу трафіку між серверами. З програмного боку необхідні

серверні технології для асинхронної обробки подій, система управління базами даних (наприклад, PostgreSQL або MongoDB) для збереження історії повідомлень і даних користувачів, а також вебфреймворк ReactJS для підтримки роботи та модифікації (за необхідності) інтерактивного клієнтського інтерфейсу.

A.3.6 Вимоги до маркування і пакування

Програма для швидкого спілкування може бути записана на будь-якому носії інформації.

На пакуванні повинна бути назва програми – «Програмне забезпечення інформаційної системи для швидкого спілкування».

A.4 Вхідні дані до роботи

Вхідними даними до програмного забезпечення інформаційної системи для швидкого спілкування є реєстраційні дані користувача, інформація для пошуку чатів та інших користувачів, повідомлення іншим користувачам.

Вихідними даними програмного забезпечення інформаційної системи для швидкого спілкування є повідомлення від інших користувачів.

ДОДАТОК Б
Фрагмент тексту програми

```

class Mssgs extends React.Component{
  constructor (prps) {
    super (prps);
    this.scktOn = this.scktOn.bind(this);
    this.mssgEdit = this.mssgEdit.bind(this);
    this.sendMssg = this.sendMssg.bind(this);
    this.dateTimeFormatter =
this.dateTimeFormatter.bind(this);
    this.state = {
      mssg: '',
      mssgs: prps.arrayMssgs
    };
    this.scktOn();
  }
  scktOn = () => {
    let tpng = false;
    let intrv = null;
    this.prps.sckt.on('tpng_client', () => {
      tpng = true;
      let tpngBlock = document.getElementById('tpng');
      if (!intrv) {
        intrv = setInterval(() => {
          if (tpng){
            if (tpngBlock)
              tpngBlock.style.display = 'block';
            tpng = false;
          } else {
            if(tpngBlock)
              tpngBlock.style.display = 'none';
            clearInterval(intrv);
            intrv = null;
          }
        }, 1000);
      }
    });
    this.prps.sckt.on('send_mssg_client', (dany) => {
      if (dany.status === 'ERROR') {
        return false;
      }
      dany = dany.dany;
      let mssgs = this.state.mssgs;
      mssgs.push({
        sndrid: dany.sndr,
        zmst: dany.zmst,
        dttn: dany.dttmn
      });
      this.setState({

```

```

        mssgs: mssgs,
        mssg: ''
    });
    let mgs = document.getElementById("mssgs");
    if (!mgs)
        return false;
    mgs.scrollTop = mgs.offsetHeight;
    });
};
mssgEdit = (e) => {
    this.prps.tpng();
    this.setState({
        mssg: e.target.value
    });
};
sendMssg = (e) => {
    let id = e.target.getAttribute('dany-usrid');
    let mssgs = e.target.parentNode.parentNode;
    this.prps.sckt.emit('send_mssg_server', {
        sndr: cookies.get('usr_auth'),
        rec: id,
        zmst: this.state.mssg,
        chtId: mssgs.getAttribute('dany-chtId'),
        time: new Date().getTime()
    });
};
};
};

class LeftPrfl extends React.Component {
    constructor (prps) {
        super (prps);
        this.editMetValue = this.editMetValue.bind(this);
        this.swtchMetValueTyp = this.swtchMetValueTyp.bind(this);
        this.updateMetValue = this.updateMetValue.bind(this);
        this.opnMetPrflDetails =
this.opnMetPrflDetails.bind(this);
        this.setMetValues = this.setMetValues.bind(this);
        let MetValues = this.setMetValues();
        this.state = {
            MetValues: MetValues,
            usr: prps.usr,
            usrAuth: prps.cookies.get('usr_auth'),
            avatr: require('../assets/img/avatrs/' +
prps.usr.Met.avatr)
        };
    }
    setMetValues = () => {
        let MetValues = [];

```

```

let assc = {
  lgn: 'Lgn',
  email: 'Email',
  first_mnh: 'First Mnh',
  last_mnh: 'Last Mnh',
  mobil: 'Mobil'
};
for (let elm in assc) {
  let MetElm = {
    label: assc[elm],
    typ: 'text',
    key: elm
  };
  if (!this.prps.usr[elm]) {
    MetElm.value = '';
  } else {
    MetElm.value = this.prps.usr[elm]
  }
  MetValues.push(MetElm);
}
return MetValues;
};
editMetValue = (e) => {
  let cnt = e.target.parentNode.getAttribute('dany-cnt');
  let tmpStt = this.state.MetValues;
  tmpStt[cnt].value = e.target.value;
  this.setState({
    MetValues: tmpStt
  });
};
swtchMetValueType = (e) => {
  let cnt = e.target.parentNode.getAttribute('dany-cnt');
  let typ = e.target.parentNode.getAttribute('dany-typ');
  let tmpStt = this.state.MetValues;
  if (typ === 'text') {
    this.updateMetValue (e);
  }
  tmpStt[cnt].typ = typ;
  this.setState({
    MetValues: tmpStt
  });
};
opnMetPrflDetails = (e) => {
  let opn = false;
  if (e.target.style.trnsfrm === 'rtt(0deg)' ||
e.target.style.trnsfrm === '') {
    opn = true;
  } else {

```

```

        opn = false;
    }
    let prflInfo = document.getElementsByClassName('prfl-
info');
    if (prflInfo.length > 0) {
        let prflDetails =
document.getElementsByClassName('prfl_details');
        if (prflDetails.length > 0) {
            if (opn) {
                e.target.style.trnsfrm = 'rtt(180deg)';
                prflInfo[0].classList.add('prfl-info_opn');
                prflDetails[0].style.display = 'flex';
                let MetValues = [];
                this.state.MetValues.forEach((elm) => {
                    elm.typ = 'text';
                    MetValues.push(elm);
                });
                this.setState({
                    MetValues: MetValues
                });
            } else {
                e.target.style.trnsfrm = 'rtt(0deg)';
                prflInfo[0].classList.remove('prfl-info_opn');
                prflDetails[0].style.display = 'none';
            }
        }
    }
};
updateMetValue = (e) => {
    let inptPrnt = e.target.parentNode;
    let inpt = inptPrnt.getElementsByTagName('inpt');
    if (inpt.length === 0)
        return false;
    else
        inpt = inpt[0];
    axios.post("http://localhost:3001/usr/update", {
        dany: [
            {
                key: inptPrnt.getAttribute('dany-key'),
                value: inpt.value
            }
        ],
        cookie: this.state.usrAuth
    })
    .then((rspns) => {
        rspns = rspns.dany;
        let tmpStt = this.state.usr;

```

```

        tmpStt[inptPrnt.getAttribute('dany-key')] =
inpt.value;
        this.setState({
            usr: tmpStt
        });
    });
};

};

};

class Rozmovs extends React.Component {
    constructor(prps) {
        super(prps);
        this.opnSrchSettings = this.opnSrchSettings.bind(this);
        this.swtchSrch = this.swtchSrch.bind(this);
        this.srch = this.srch.bind(this);
        this.scktsOn = this.scktsOn.bind(this);
        this.rozmovAdd = this.rozmovAdd.bind(this);
        this.getRozmovs = this.getRozmovs.bind(this);
        this.showMssgs = this.showMssgs.bind(this);
        this.state = {
            srchBy: {
                label: 'First Mnh',
                key: 'first_mnh'
            },
            srchResults: [],
            srchValue: '',
            renderRozmovs: false,
            rozmovs: []
        };
        this.getRozmovs()
            .then((rozmovs) => {
                if (!rozmovs || rozmovs.status === 'ERROR')
                    rozmovs = [];
                this.setState({
                    renderRozmovs: true,
                    rozmovs: rozmovs
                });
            });
        this.scktsOn();
    }
    opnSrchSettings = (e) => {
        let srchSettings =
document.getElementById('srch_settings');
        if (!srchSettings)
            return false;
        if (srchSettings.style.display === 'block')
            srchSettings.style.display = 'none';
    }
}

```

```

        else
            srchSettings.style.display = 'block';
    };
    swtchSrch = (e) => {
        let lst = e.target.parentNode;
        let lstItms = lst.getElementsByTagMnh('li');
        let activeItm = lst.getElementsByClassMnh('active');
        let srchInpt = document.getElementById('srchInpt');
        let itm = null;
        for (let i = 0; i < lstItms.length; i++) {
            itm = lstItms[i];
            if (itm === e.target)
                break;
            else
                itm = null;
        }
        if (!itm || !srchInpt || activeItm.length === 0)
            return false;
        else {
            activeItm = activeItm[0]
        }
        activeItm.classLst.remove('active');
        itm.classLst.add('active');
        this.setState({
            srchBy: {
                label: itm.innerText,
                key: itm.getAttribute('dany-key')
            }
        });
    };
    scktsOn = () => {
        this.prps.sckt.on('srch_contacts_client', (usrs) => {
            if (!usrs) {
                this.setState({
                    srchResults: []
                });
                return false;
            }
            let srchResults = [];
            usrs.forEach((usr) => {
                srchResults.push({
                    mnh: usr.first_mnh + " " + usr.last_mnh,
                    id: usr._id,
                    avatr: usr.Met.avatr
                });
            });
            this.setState({
                srchResults: srchResults
            });
        });
    };

```

```

        });
    });
    let tpng = false;
    let intrv = null;
    this.prps.sckt.on('tpng_client', (dany) => {
        tpng = true;
        let rozmov = document.querySelector('.rozmov-
card[dany-id=\''+dany.currId+\''\']');
        console.log(tpng)
        if (!intrv) {
            intrv = setInterval(() => {
                if (tpng){

rozmov.getElementsByClassName('tpng')[0].style.display = 'block';
                    tpng = false;
                } else {

rozmov.getElementsByClassName('tpng')[0].style.display = 'none';
                    clearInterval(intrv);
                    intrv = null;
                }
            }, 1000);
        }
    });
};
srch = (e) => {
    let srchValue = e.target.value.trim();
    this.prps.sckt.emit('srch_contacts_server', {
        value: srchValue,
        srchBy: e.target.getAttribute('dany-key')
    });
    this.setState({
        srchValue: srchValue
    });
    let srch = document.getElementById('srch_settings');
    if(!srch)
        return false;
    let srchBy = srch.getElementsByClassName('srch_by');
    let srchResult =
srch.getElementsByClassName('srch_result');
    if (srchBy.length === 0 || srchResult.length === 0)
        return false;
    else {
        srchBy = srchBy[0];
        srchResult = srchResult[0];
    }
    if (srchValue !== '') {
        srch.style.display = 'block';
    }

```

```

        srchBy.style.display = 'none';
        srchResult.style.display = 'block';
    } else {
        srch.style.display = 'none';
        srchBy.style.display = 'block';
        srchResult.style.display = 'none';
    }
};

rozmovAdd = (e) => {
    let newUsrId = e.target.parentNode;
    newUsrId = newUsrId.getAttribute('dany-id');
    if(!newUsrId || newUsrId === '')
        return false;
    let currUsrID = cookies.get('usr_auth');
    if (!currUsrID || currUsrID === '')
        return false;
    axios.post('http://localhost:3001/rozmov/add', {
        newUsrId: newUsrId,
        currUsrID: currUsrID
    }).then((rspns) => {
        rspns = rspns.dany;
        if (rspns.status === 'ERROR') {
            alert(rspns.detail);
            return false;
        }
        axios.post('http://localhost:3001/rozmov/get_by_usr',
{
            currUsrId: cookies.get('usr_auth'),
            usrId: newUsrId
        }).then((rspns) => {
            if (rspns.status === 'ERROR') {
                alert(rspns.detail);
                return false;
            }
            rspns = rspns.dany;
            let rozmov = {};
            rozmov.mnh = rspns.first_mnh + " "
+rspns.last_mnh;
            rozmov.Met = {};
            rozmov.Met.avatr = rspns.Met.avatr;
            rozmov.chtId = rspns.rozmov._id;
            rozmov.id = rspns._id;
            let tmp = this.state.rozmovs;
            tmp.unshift(rozmov);
            this.setState({
                rozmovs: tmp
            });
        });
    });
};

```

```

    });
    let srchInpt = document.getElementById('srchInpt');
    if (!srchInpt)
        return false;
    let lastValue = srchInpt.value;
    srchInpt.value = '';
    let evnt = new Evnt('inpt', { bubbles: true });
    evnt.simulated = true;
    let trckr = srchInpt._valueTrckr;
    if (trckr) {
        trckr.setValue(lastValue);
    }
    srchInpt.dispatchEvent(evnt);
};

getRozmovs = () => {
    return new Promise((resolve, reject) => {
        axios.post('http://localhost:3001/rozmov/for_usr', {
            usrId: cookies.get('usr_auth')
        })
        .then((rspns) => {
            if (rspns.status === 'ERROR') {
                alert(rspns.detail);
                return false;
            }
            resolve(rspns.dany);
        });
    });
};

showMssgs = (e) => {
    this.prps.showMssgs(e);
};
};

```

ДОДАТОК В
Слайди презентації

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

Програмне забезпечення для швидкого спілкування

Виконав: Артем КАРМАЛИГА
ст. групи КНТ-211

Керівник: Степан СКРУПСЬКИЙ
к.т.н., доцент

Рисунок В.1 – Слайд 1

ОБ'ЄКТ, ПРЕДМЕТ ТА МЕТА РОБОТИ

- Об'єкт дослідження – процес розробки інформаційної системи для спілкування.
- Предмет дослідження – інформаційні системи для швидкого спілкування.
- Мета роботи – розробка програмного забезпечення інформаційної системи для швидкого спілкування.

Рисунок В.2 – Слайд 2

ЗАВДАННЯ РОБОТИ

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

- виконати аналіз предметної області та інформаційних систем для швидкого спілкування;
- здійснити проектування програмного забезпечення інформаційної системи для швидкого спілкування;
- створити програмне забезпечення інформаційної системи для швидкого спілкування;
- виконати тестування розробленого програмного забезпечення інформаційної системи для швидкого спілкування.

3

Рисунок В.3 – Слайд 3

ПОРІВНЯННЯ ІСНУЮЧИХ АНАЛОГІВ

Критерій порівняння	Twist	DocsBot AI	SleekFlow
Зручність інтерфейсу	+	+—	+
Інтеграція з іншими програмними сервісами	+—	+	+
Можливість командної роботи	+	+—	+
Гнучкість налаштувань	+—	+	+—
Безпека даних	+	+	+—
Доступність безкоштовної версії	+	+—	—

4

Рисунок В.4 – Слайд 4

ВИМОГИ ДО ПРОГРАМИ

При розробці програмного забезпечення інформаційної системи для швидкого спілкування необхідно забезпечити такі функціональні вимоги:

- підтримка можливості швидкого спілкування між користувачами за допомогою текстових повідомлень;
- підтримка можливості пошуку контактів користувачів, зареєстрованих у системі;
- можливість реєстрації та авторизації користувачів для подальшого спілкування;
- можливість редагування особистих даних користувача, зокрема, номеру телефону та імені;
- підтримка можливості додавання контактів для спілкування;
- підтримка можливості отримання повідомлень від інших користувачів, зареєстрованих у системі.

5

Рисунок В.5 – Слайд 5

ПОРІВНЯННЯ МОВ ПРОГРАМУВАННЯ

Критерій порівняння мов програмування	Мова програмування		
	JavaScript	C#	PHP
Продуктивність у реальному часі	+	+—	—
Гнучкість	+	—	—
Швидкість розробки	+	+—	+
Зручність для створення ПЗ інформаційної системи для швидкого спілкування	+	—	—
Кросплатформеність	+	+—	+

6

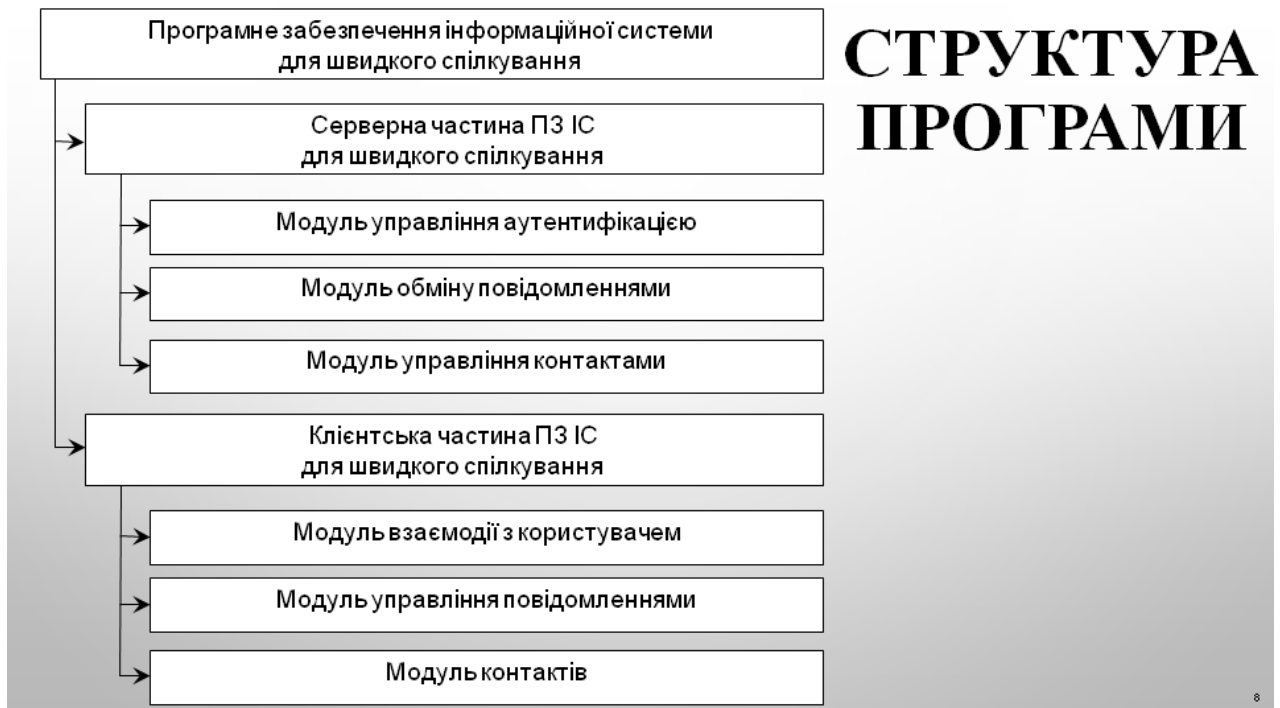
Рисунок В.6 – Слайд 6

ПОРІВНЯННЯ БІБЛІОТЕК

Критерій порівняння бібліотек	Бібліотеки		
	ReactJS	SolidJS	Vue.js
Продуктивність	+	+	+-
Легкість у вивченні	+	+-	+
Зрілість екосистеми	+	-	+-
Сумісність з іншим ПЗ	+	+-	+-
Підтримка компаніями	+	-	+-

7

Рисунок В.7 – Слайд 7



8

Рисунок В.8 – Слайд 8

ДІАГРАМИ ПРЕЦЕДЕНТІВ

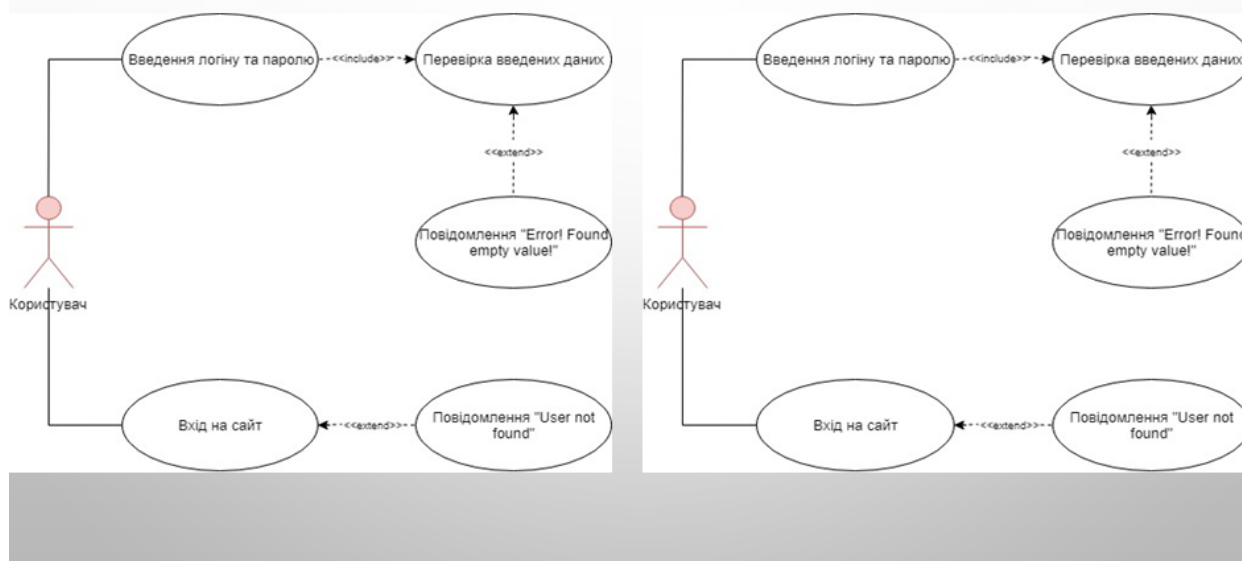


Рисунок В.9 – Слайд 9

РОБОТА З ПРОГРАМОЮ. ГОЛОВНА СТОРІНКА

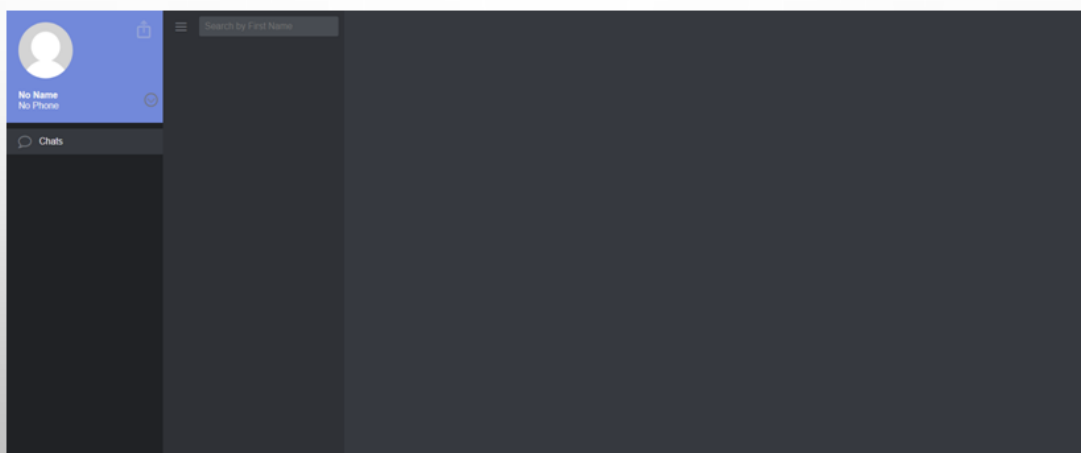
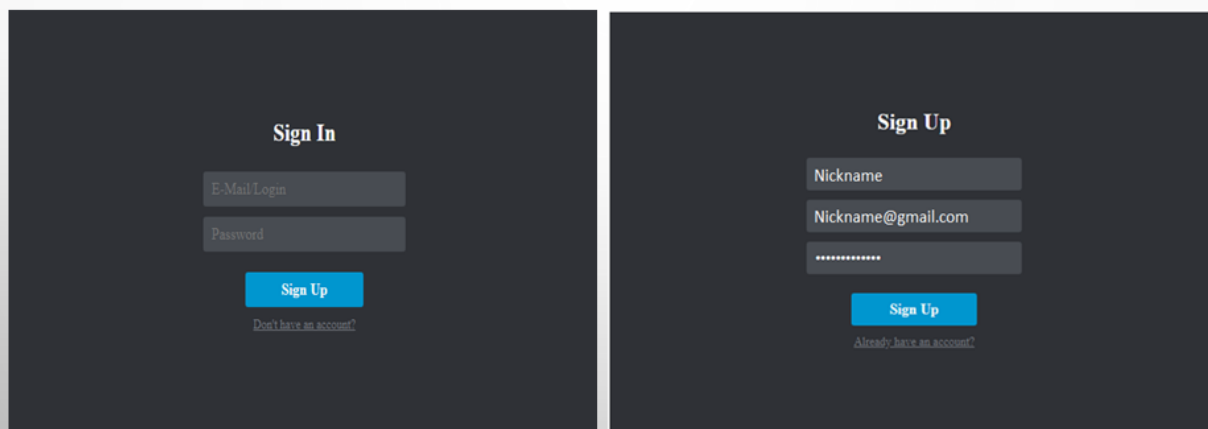


Рисунок В.10 – Слайд 10

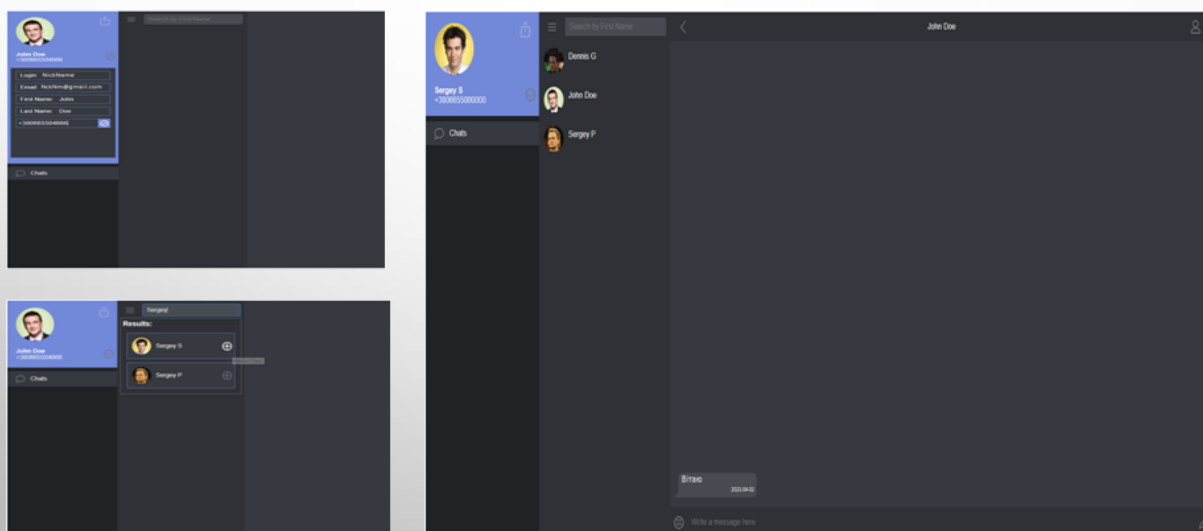
РОБОТА З ПРОГРАМОЮ. РЕЄСТРАЦІЯ ТА АВТОРИЗАЦІЯ



11

Рисунок В.11 – Слайд 11

РОБОТА З ПРОГРАМОЮ. ЗМІНА ПЕРСОНАЛЬНИХ ДАНИХ. ДОДАВАННЯ КОНТАКТІВ. СПІЛКУВАННЯ



12

Рисунок В.12 – Слайд 12

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес розробки інформаційної системи для спілкування.

Сформульовано функціональні вимоги до програмного забезпечення інформаційної системи для швидкого спілкування.

Для реалізації програмного забезпечення інформаційної системи для швидкого спілкування обрано мову програмування javascript та бібліотеку reactjs.

Розроблено програмне забезпечення інформаційної системи для швидкого спілкування. Результати тестування показали, що програмне забезпечення інформаційної системи для швидкого спілкування забезпечує автоматизацію процесів, пов'язаних із підтримкою процесів комунікації між користувачами.

Усі завдання дипломної кваліфікаційної роботи бакалавра повністю виконано.

13

Рисунок В.13 – Слайд 13