SCICOPUB

Chat Application

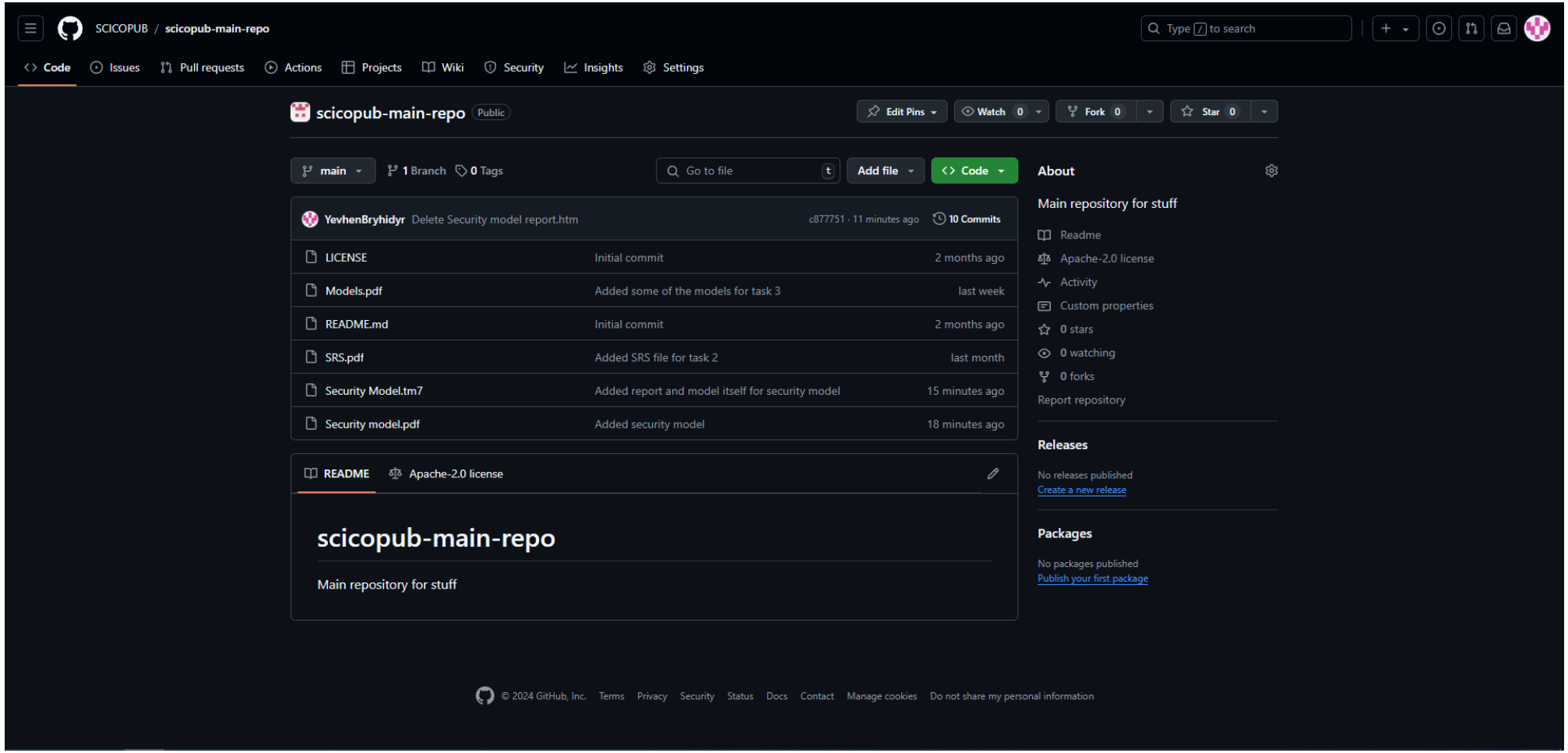
Команда

- Брошко Анастасія
- Бригідир Євген
- Гнатів Олексій
- Щетінін Роман

План

- Check 0
  - GitHub Repo
- Check 1
  - Опис продукту
  - Архітектура
  - Concurrency patterns usage
- Check 2
  - Data model
  - Security model
  - Analytics model
  - Monitoring&Alerting model
- Висновки
- Q&A

Check 0 - GithubRepo



https://github.com/SCICOPUB/scicopub-main-repo

Check 1 – Опис продукту

1.Вступ

1.1 Ціль

Метою цього документу є побудова десктопного месенджеру для безпечного та безкоштовного листування та обміну інформацією. Аплікація дозволить користувачеві надсилати та приймати повідомлення, обмінюватися даними та здійснювати аудіо та відеовиклики.

1.2 Цільова аудиторія

Аплікація буде корисною для людей всіх вікових та соціальних груп, оскільки пропонує швидкий, простий, безкоштовний та, найголовніше, захищений спосіб передачі інформації в мережі. Месенджер дозволить користувачам ділитися даними по зашифрованих каналах інформації, що унеможливить їх пошкодження або викрадення.

Опис продукту

2.1 Особливості продукту

Месенджер пропонує такі можливості:

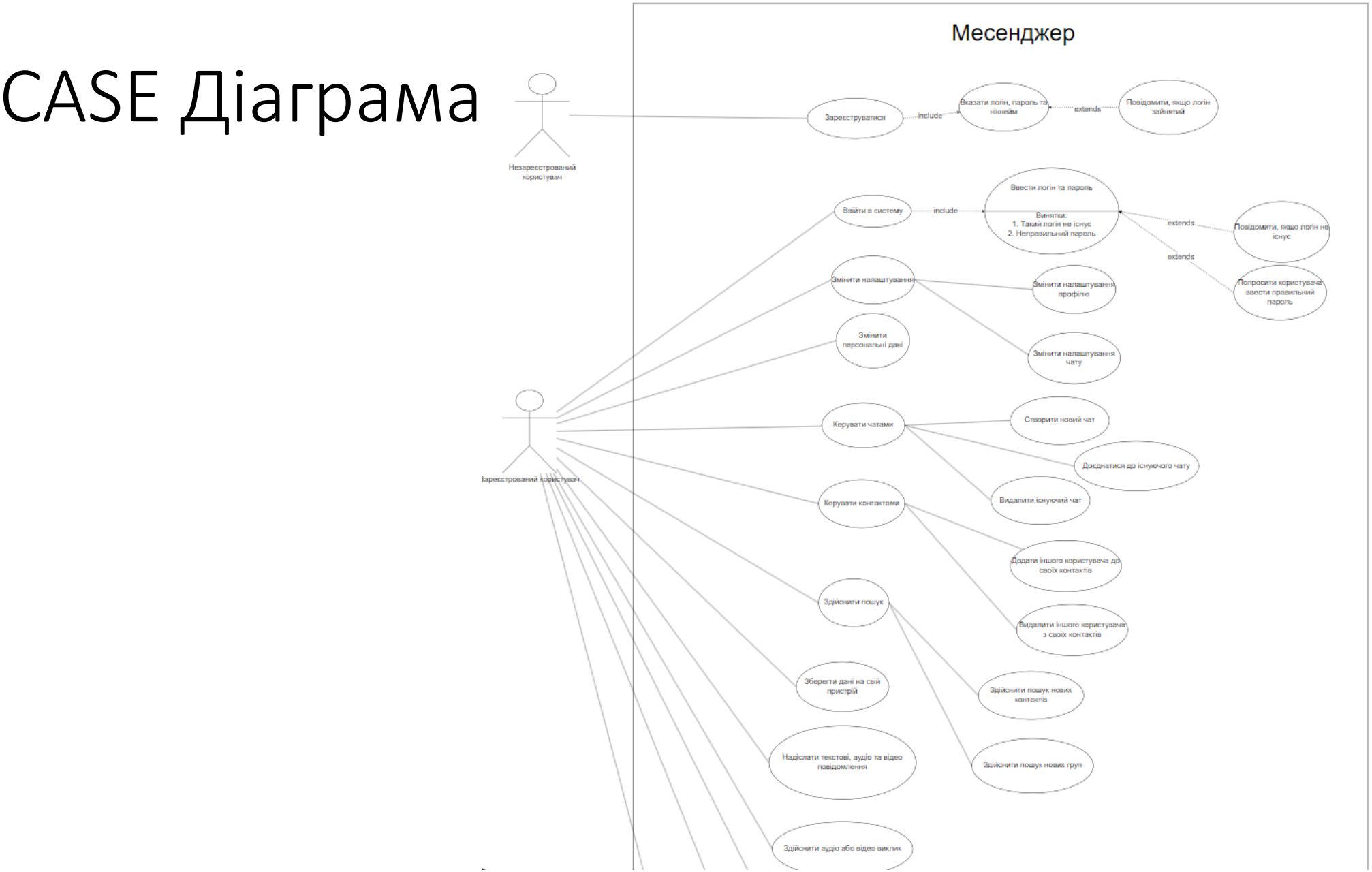
- **Зберігання даних в хмарі** – Ця особливість дозволяє зареєстрованим користувачам зберігати свої дані в хмарі та економити місце на своїх пристроях.
- **Функція автоматичного видалення повідомлень** – Зареєстрований користувач може ввімкнути автоматичне видалення повідомлень і вибрати час, через який всі надіслані повідомлення в конкретній розмові будуть видалені.
- **Інтеграція з OCR** – Користувач може конвертувати зображення з текстом у текстовий формат. З цим документом можна буде працювати як зі звичайним текстовим файлом, редагувати та змінювати його.

Функціональні вимоги

2.2 Функціональні вимоги

1. Незареєстрований користувач може зареєструватися в системі. Користувач повинен вказати логін (електронна пошта або номер телефону), пароль та нікнейм.
    a. Коли користувач вибере логін, який вже використовується системою, йому сповістять про це, та попросять вибрати інший логін.
    b. Користувач вказує незайнятий логін та пароль і успішно реєструється в системі.
2. Зареєстрований користувач може ввійти в систему, ввівши свій логін та пароль.
    a. Якщо користувач вводить неправильний логін або пароль, його сповістять про це та попросять ввести дані знову.
    b. Якщо користувач вводить правильний логін та пароль, він входить в систему.
3. Зареєстрований користувач може змінити налаштування профілю.
4. Зареєстрований користувач може змінити свої персональні дані.
5. Зареєстрований користувач може створити новий чат.
6. Зареєстрований користувач може видалити існуючий чат.
7. Зареєстрований користувач може додати іншого користувача до своїх контактів.
8. Зареєстрований користувач може здійснити пошук нових контактів.
9. Зареєстрований користувач може здійснити пошук нових груп.
10. Зареєстрований користувач може зберегти дані на свій пристрій.
11. Зареєстрований користувач може керувати налаштуваннями чату.
12. Зареєстрований користувач може видалити іншого користувача з своїх контактів.
13. Зареєстрований користувач може надіслати текстові, аудіо та відео повідомлення.
14. Зареєстрований користувач може здійснити аудіо або відео виклик.
15. Зареєстрований користувач може вийти з системи.
16. Зареєстрований користувач може переглянути оновлення застосунку.
17. Зареєстрований користувач може заблокувати іншого користувача.
18. Зареєстрований користувач може доєднатися до існуючого чату.

USE CASE Діаграма



USE CASE Таблиця

№	Ім'я	Функціональність	Опис
1	Зареєструватися	Створює новий акаунт	Користувач створює новий акаунт і заповнює його персональною інформацією
2	Ввійти в систему	Входить в існуючий акаунт	Користувач заходить в свій акаунт за допомогою логіну і паролю
3	Змінити налаштування	Користувач змінює налаштування	Користувач змінює налаштування профілю або чату
4	Змінити персональні дані	Користувач змінює свої персональні дані	Користувач може змінити свої дані або зовсім видалити їх
5	Керувати чатами	Користувач проводить дії над чатами	Користувач створює, видаляє або доєднується до вже існуючого чату
6	Керувати контактами	Користувач проводить дії над контактами	Користувач додає або видаляє іншого користувача зі своєї книжки контактів
7	Здійснити пошук	Користувач здійснює пошук	Користувач здійснює нових контактів або нових груп, в які згодом зможе доєднатися
8	Зберегти дані на свій пристрій	Користувач зберігає дані на пристрій	Користувач може зберегти фото, відео або аудіо з чатів та груп на свій пристрій
9	Надіслати текстові, аудіо та відео повідомлення	Користувач надсилає іншому користувачу повідомлення	Користувач може надіслати текст, аудіо, відео або інший файл іншому користувачу в чаті або групі
10	Здійснити аудіо або відео виклик	Користувач здійснює виклик іншому користувачу	Користувач може здійснити аудіо або відео виклик до будь-якого іншого користувача, незалежно чи користувач є в його контактній книжці, хіба що, інший користувач відключить цю можливість
11	Переглянути оновлення застосунку	Користувач переглядає останні оновлення аплікації	Користувач може подивитися всі зміни, які відбулися з випуском останнього оновлення
12	Заблокувати іншого користувача	Користувач блокує іншого користувача	Користувач може заблокувати іншого користувача, при цьому, користувач, якого заблокували не зможе надсилати повідомлення або здійснювати виклики до першого користувача
13	Вийти з системи	Користувач виходить з системи	Користувач може вийти з системи, при цьому при наступному вході в обліковий запис, система попросить його ввести логін та пароль

User Stories

15. Як користувач, я хочу мати можливість знищити акаунт, щоб позбутися даних на ньому
16. Як користувач, я хочу отримувати повідомлення щодо дії в моєму акаунті, щоб уникнути сторонньої активності
17. Як користувач, я хочу конвертувати голосові повідомлення в текст та навпаки, щоб пришвидшити процес спілкування

2.4 Операційне середовище

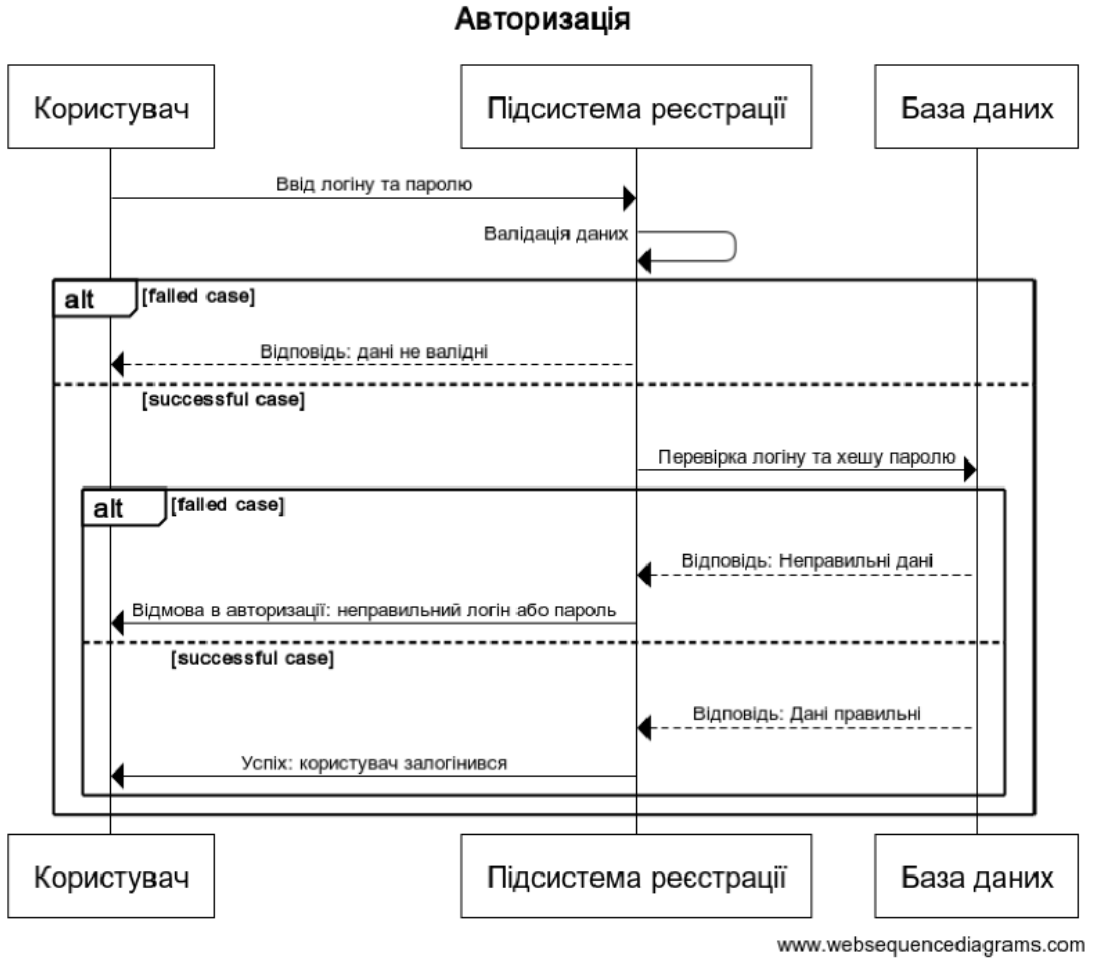
Операційне середовище для десктопної аплікації наведено нижче:

- База даних: MS SQL SERVER
- Клієнт/сервер: ASP.NET MVC / .NET Core

User Stories

1. Як користувач, я хочу реєструватися анонімно, щоб не хвилюватися про безпеку при використанні номера телефону
2. Як користувач, я хочу встановити фотографію профілю, щоб знайомі могли легко впізнати мене
3. Як користувач, я б хотів бачити список контактів та мати можливість пошуку, щоб швидко почати листування з потрібною людиною.
4. Як користувач, я хочу бачити функцію індикації присутності, щоб не турбувати людину в незручний час
5. Як користувач, я хочу мати можливість почати індивідуальний або груповий чат, щоб підтримувати зв'язок з друзями, родичами чи колегами
6. Як користувач, я хочу надсилати файли в чат для більшої насиченості спілкування
7. Як користувач, я хочу бачити місце розташування контактів за дозволеним запитом, щоб реагувати в надзвичайній ситуації
8. Як користувач, я хочу мати розподілений буфер для збереження повідомлень, посилань та корисних матеріалів, щоб не зберігати їх на пристрої
9. Як користувач, я хочу видаляти завантажені файли в хмару з пристрою, щоб економити місце на пристрої
10. Як користувач, я хочу обмежувати інформацію до свого телефонного номера, щоб його не можна було використовувати для сторонніх комунікацій
11. Як користувач, я хочу приховати інформацію про індикацію присутності, щоб не мати потреби відповідати користувачам негайно
12. Як користувач, я хочу дозволяти залишати свою індикацію присутності видимою, навіть якщо її вимкнуто для мене, щоб користувач міг бачити, коли я був онлайн
13. Як користувач, я хочу обмежувати пересилання моїх повідомлень, щоб забезпечити конфіденційність спілкування
14. Як користувач, я хочу ввімкнути тристапну верифікацію, щоб застерегтися від стороннього доступу до мого акаунту з мого пристрою

Concurrency patterns usage



Стимул/Послідовності відповідей

Користувач створює новий акаунт і заповнює його особистою інформацією.

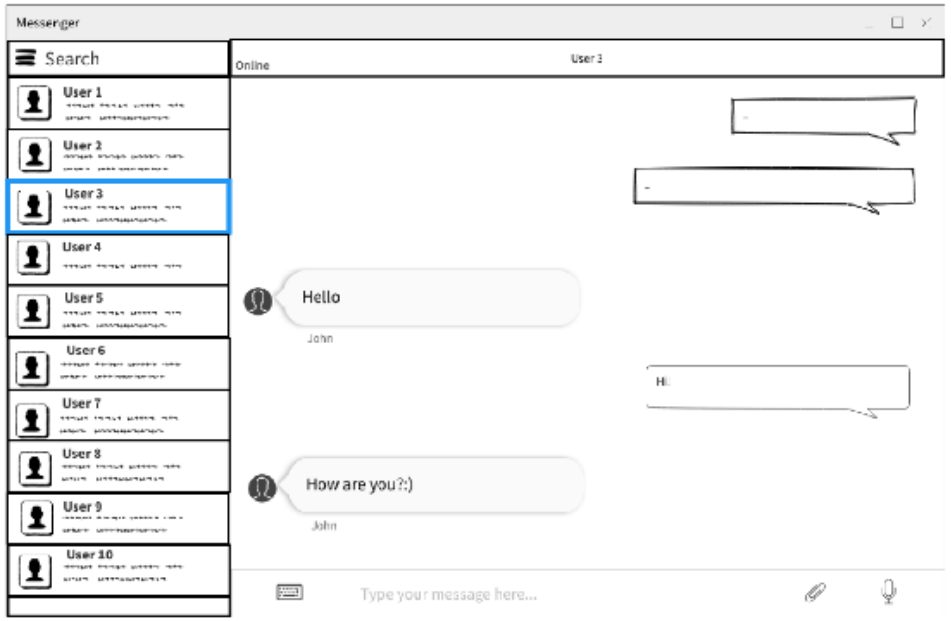
Користувач заходить в свій обліковий запис за допомогою логіну і паролю.

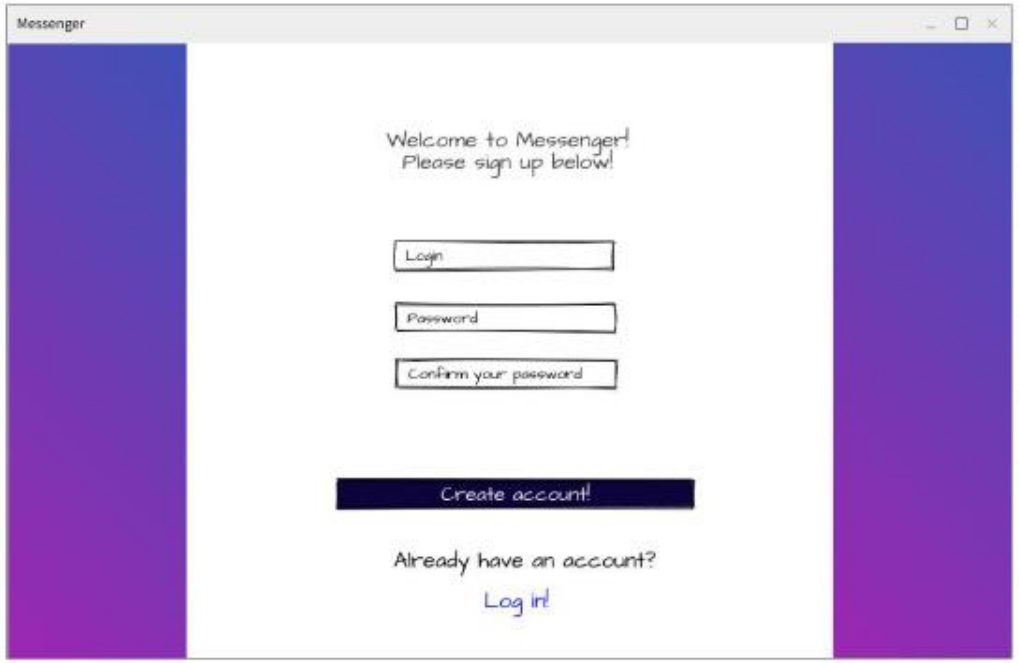
Користувач може почати чат або доєднатися до існуючого.

- Stimulus: Користувач створює акаунт з новим логіном та паролем
- Response: Акаунт створено
- Stimulus: Користувач створює акаунт з зайнятим логіном
- Response: Акаунт не створено, користувача повідомлено про це
- Stimulus: Вибраний чат
- Response: Вибраний чат підсвічується
- Stimulus: Користувач вийшов з системи
- Response: Сесія для поточного користувача закінчилася
- Stimulus: Відправлено повідомлення
- Response: Непрочитані повідомлення позначаються однією галочкою, прочитані двома

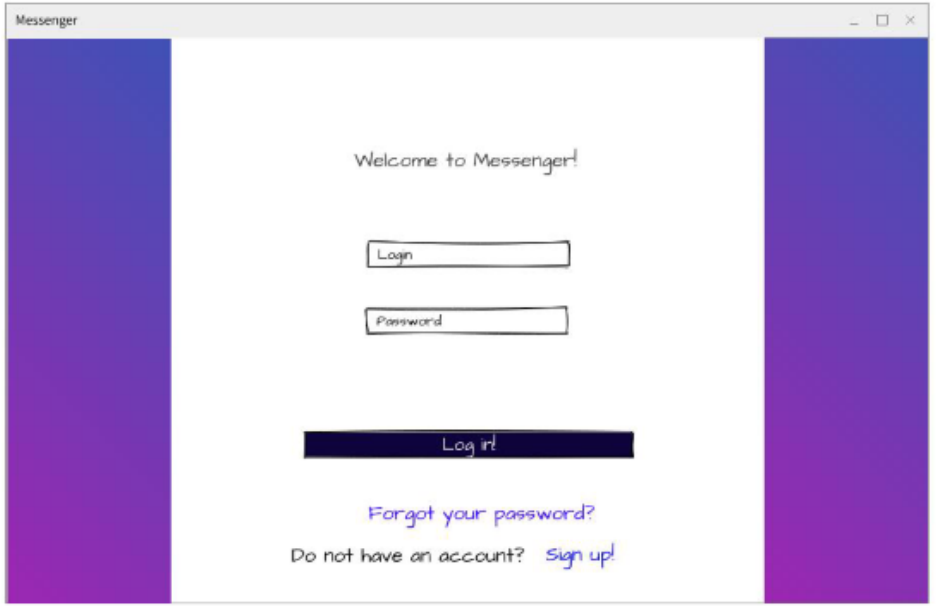
Зовнішній інтерфейс

Main page:





Log in page:



Нефункціональні вимоги

5.1 Вимоги до продуктивності

Аплікація розрахована на великий потік людей, а тому потрібно підтримувати швидкий та постійний зв'язок між користувачами та сервером. Невиконання цього пункту може призвести до негативного досвіду з боку користувача, а в подальшому і відмови від продукту.

5.2 Вимоги до безпеки використання

Так як месенджером можуть користуватися люди різного віку, потрібно приділити особливу увагу щодо захисту дітей від недоречного контенту. Перш за все це стосується адмінів груп та каналів. Потрібен контроль за наповненням груп, а також спеціальний режим для дітей, який буде приховувати недоречний контент.

5.3 Вимоги до сек'юріті

Месенджером можуть передаватися важливі дані, які не можуть бути розголошені. Паролі, інформація про кредитні картки тощо. В цьому випадку надійна передача даних грає ключову роль:

- Виключити можливість передачі інформації до третіх осіб.
- Виключити можливість доступу до бази даних з паролями та особистою інформацією користувачів.
- Виключити можливість витоку інформації про користувача при використанні анонімного режиму.
- Виключити можливість використання ненадійних каналів зв'язку.
- Виключити можливість невикористання шифруваннях даних.
- Виключити можливість несанкціонованого доступу до акаунтів користувачів.

Цього можна досягти при використанні новітніх засобів безпеки, криптографії, шифруванню даних і постійній підтримці та оновленням аплікації.

5.4 Атрибути якості програмного забезпечення

Безпечність: Месенджер повинен надавати захищений доступ до даних користувача, а також шифрувати повідомлення в каналах.

Багатоплатформність: Месенджер повинен працювати на всіх платформах незалежно від ОС або комплектації.

Постійна підтримка: Месенджер повинен отримувати регулярні оновлення та в разі виявлення багів або експлойтів, ці речі повинні бути негайно виправлені.

Доступність: Месенджер повинен бути безкоштовним та надавати повний функціонал у своїй стандартній версії. Платні версії не повинні кардинально змінювати користувацький досвід використання програми, а лише покращувати його різними доповненнями.

5.5 Масштабованість

Кількість активних чатів: Месенджер повинен підтримувати одночасне використання до 20 чатів без істотної деградації продуктивності.

Користувачі: Система повинна коректно працювати з великою базою одночасно активних користувачів.

Архітектура

Компоненти:

Chat user – користувач програми, взаємодіє з контролерами за допомогою представлення view.

Backend (ASP.NET MVC):

- Контролери приймають запити від користувача
- Контролери маніпулюють моделями задля отримання чи редагування інформації
- Контролери виконують бізнес логіку програми
- Контролери використовують логер для логування важливих процесів виконання програми
- Моделі доступаються до бази даних, виконують запити та отримують відповіді
- Моделі передають дані до views

Logging (log4net):

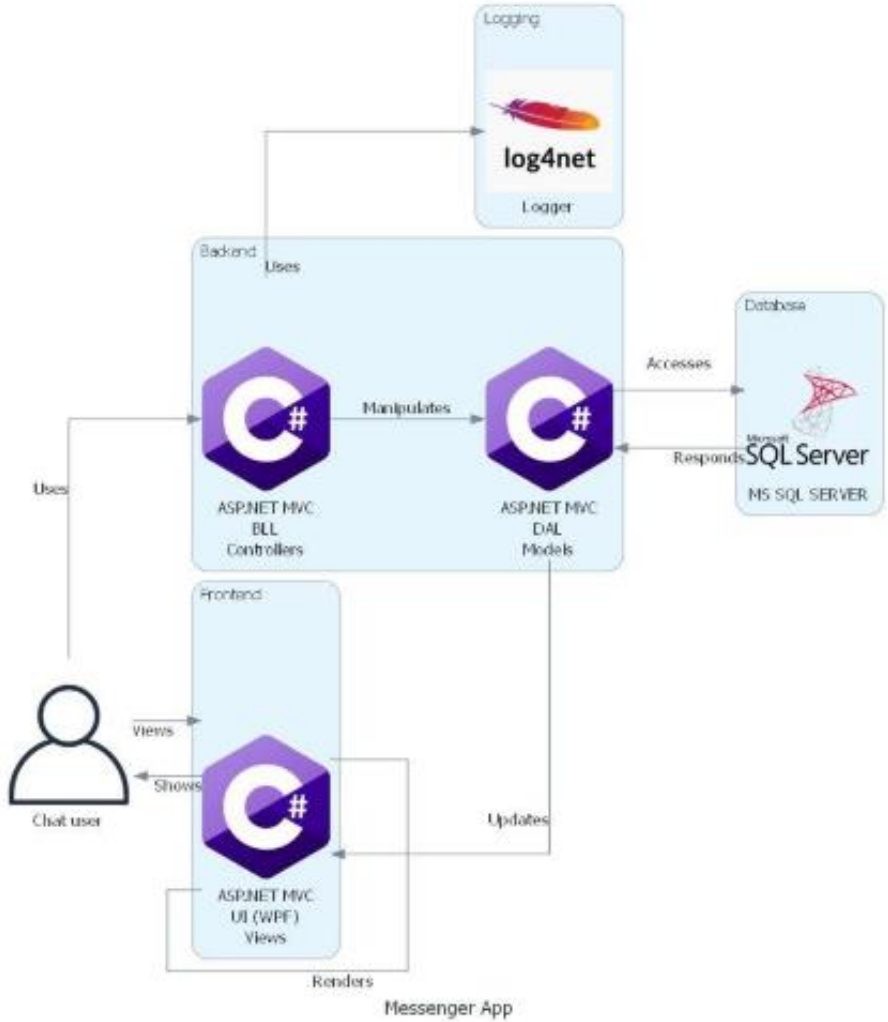
- Логер записує важливу інформацію у локальне сховище або у хмару
- Логер можна налаштувати під потреби розробника, як от логування входу користувачів в систему

Database (MSSQL Server):

- База даних зберігає всі дані про користувачів, чати, повідомлення тощо
- База даних напряму взаємодіє тільки з моделями

Frontend (ASP.NET MVC WPF):

- Views відображають дані на екрані користувача: чати, групи, друзі, повідомлення
- Views є основним способом взаємодії користувача і програми
- Views отримують дані з моделей та оновлюють свій стан



6. Архітектура високого рівня

Десктоп месенджер має трьохшарову архітектуру на основі патерну MVC. Presentation Layer відповідає за взаємодію користувача і додатку. Business Logic Layer обробляє запити і проводить обчислення. Data Access Layer доступається до бази даних, зберігає дані та передає потрібну інформацію в Business Logic Layer.

Check 2 – Data model

Опис сутностей:

PK-primary key, FK-foreign key

Users (Користувачі)

Id: Унікальний ідентифікатор користувача. PK

Name: Ім'я користувача.

Surname: Прізвище користувача.

BirthDate: Дата народження користувача.

Image: Шлях до місця зберігання фото профілю користувача.

Login: Логін користувача.

Password: Пароль користувача (захешований).

Email: Електронна пошта користувача.

Contacts (Контакти)

Id: Унікальний ідентифікатор запису контакту. PK

UserId: Ідентифікатор користувача, що має контакт. FK

ContactId: Ідентифікатор контакту користувача. FK

CreatedAt: Дата та час додавання контакту.

Status: Статус контакту (pending, accepted, blocked).

Класифікація даних за data retention policy:

Users (Користувачі)

Тип даних: Персональні дані

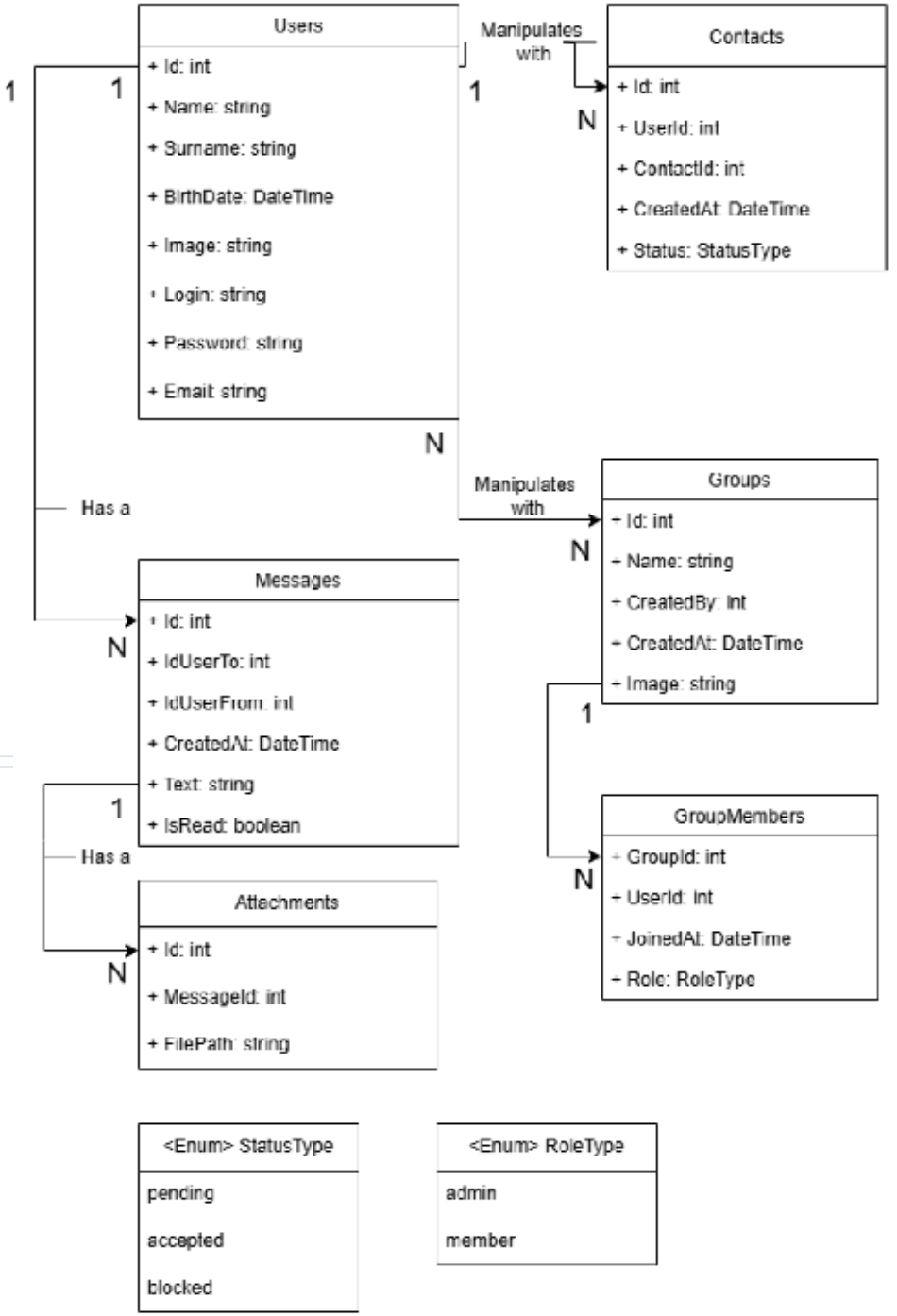
Ідентифікатор (Id): Унікальний числовий ідентифікатор користувача. **Конфіденційність**: Висока. **Зберігання**: Безстроково для ідентифікації користувача.

Ім'я (Name), Прізвище (Surname), Дата народження (BirthDate), Логін (Login), Email: Персональні дані, що використовуються для автентифікації та комунікації з користувачем. **Конфіденційність**: Висока. **Зберігання**: Зберігаються до деактивації облікового запису або на запит користувача.

Пароль (Password): Захищені дані, які зберігаються в зашифрованому вигляді. **Конфіденційність**: Висока. **Зберігання**: Безстроково, тільки в зашифрованому вигляді.

Зображення профілю (Image): Особисті дані, які використовуються для відображення в інтерфейсі. **Конфіденційність**: Висока. **Зберігання**: Зберігаються поки обліковий запис активний або до запиту на видалення.

Рекомендація: Персональні дані (окрім паролів) слід видаляти після 3 років неактивності або на запит користувача. Для даних користувачів, які деактивували обліковий запис, зберігання не повинно перевищувати 3 роки.



Monitoring&Alerting model

Список основних операційних метрик для моніторингу

Метрика	Виміри	Зв'язок з інфраструктурними ресурсами	Спосіб збору
Завантаження CPU	Відсоток використання процесора	Сервери обробки повідомлень	Моніторинг через системні агенти
Завантаження пам'яті	Відсоток, обсяг (GB)	Сервери для обробки запитів	Збір через системні агенти або Zabbix
Затримка обробки повідомлень	Мілісекунди, кількість запитів	Сервери передачі повідомлень	Моніторинг часу відповіді через API
Швидкість завантаження вікна додатку	Мілісекунди	Інтерфейс користувача (десктоп-додаток)	Вимірювання часу завантаження інтерфейсу
Час доставки повідомлень	Секунди	Сервери черг повідомлень	Моніторинг черг доставки через системи моніторингу
Використання дискового простору	GB, відсоток	Сервери зберігання повідомлень	Збір даних про сховища через системні агенти
Пропускна здатність мережі	Мбіт/с, кількість пакетів	Мережеві інтерфейси	Збір даних через мережеві монітори
Кількість активних підключень	Користувачі, сесії	Сервери обробки запитів	Збір через логи підключень
Кількість спроб входу	Кількість спроб, користувачі	Сервери авторизації	Логування через

			інструменти безпеки
Використання кешу	Відсоток, кеш-промахи	Сервери кешування (Redis, Memcached)	Моніторинг кешу через вбудовані інструменти
Середній час очікування відповіді	Мілісекунди	Сервери обробки запитів	Збір даних через аналіз логів
Кількість відправлених повідомлень	Повідомлення/хвилина	Сервери обробки повідомлень	Збір через системи аналітики
Температура серверів	Градуси Цельсія	Центри обробки даних	Збір через сенсори та системи охолодження
Кількість подій логування	Записи/хвилина, типи подій	Сервери, додатки	Збір логів через ELK-стек або Splunk
Час простою серверів	Години, хвилини	Сервери обробки повідомлень	Моніторинг через інструменти доступності

Monitoring&Alerting model

Alerting: Мін/Макс допустимі значення

Метрика	Мін/Макс значення	Тип	Критичність	План дій (Mitigation Plan)
Завантаження CPU	> 90%	Критичне	Висока	Оптимізація процесів, перерозподіл навантаження
Завантаження пам'яті	> 85%	Критичне	Висока	Збільшення пам'яті, очищення кешу
Затримка	> 200 мс	Критичне	Висока	Перевірка

Пояснення способів збору метрик

1. Системні агенти: Використовуються для збору метрик, пов'язаних з CPU, пам'яттю, диском тощо. Вони встановлюються на хостах і регулярно надсилають дані до центрального моніторингу.
2. APM-системи (Application Performance Monitoring): Застосовуються для моніторингу додатків і їх продуктивності, відстеження часу відгуку та запитів.
3. Мережеві монітори: Використовуються для збору даних про мережеву пропускну здатність і затримки.
4. Збір логів: Лог-файли аналізуються спеціальними сервісами для виявлення помилок, активності або подій у додатках.

Mitigation Plan для найкритичніших метрик

1. Завантаження CPU > 90%:
- Дії: Визначення процесів, що споживають найбільше ресурсів, оптимізація або перезапуск, масштабування інфраструктури.
- Резервний план: Розподіл навантаження між іншими серверами.

2. Завантаження пам'яті > 85%:
- Дії: Перевірка та очищення кешу, збільшення обсягу пам'яті, перевірка наявності витоків пам'яті.
- Резервний план: Зупинка та перезапуск нерелевантних служб для звільнення пам'яті.

3. Час відгуку додатку > 500 мс:
- Дії: Оптимізація запитів до бази даних, перевірка мережевих затримок, збільшення серверних ресурсів.
- Резервний план: Перехід на резервний сервер або масштабування за рахунок додаткових серверів.

4. Кількість помилок > 10 помилок/хв:
- Дії: Аналіз логів для виявлення причин, негайне виправлення коду або перезапуск сервісів.
- Резервний план: Переключення на резервну версію додатку або відкат оновлень.

5. Час простою > 5 хв:
- Дії: Перевірка та відновлення підключень, негайний перезапуск серверів.
- Резервний план: Перехід на резервний центр обробки даних або резервний сервер.

обробки повідомлень				черг, оптимізація серверів
Швидкість завантаження вікна додатку	> 3 сек	Критичне	Висока	Перевірка ресурсів фронтенду, оптимізація
Час доставки повідомлень	> 2 сек	Попередження	Середня	Оптимізація черг повідомлень
Використання дискового простору	< 15% або > 90%	Попередження	Середня	Очищення старих даних, збільшення обсягу диску
Кількість активних підключень	< 50 користувачів	Попередження	Середня	Перевірка підключень, моніторинг доступності
Кількість спроб входу	> 50 спроб/хвилина	Критичне	Висока	Блокування підозрілої активності, перевірка безпеки
Температура серверів	> 70°C	Критичне	Висока	Охолодження серверів, перевірка вентиляції
Час простою серверів	> 5 хвилин	Критичне	Висока	Перезапуск серверів, перевірка мережевих підключень

Analytics model

Аналітика: Метрики

Метрика	Виміри	Зв'язок з функціональністю чату
Середній час відкриття вікна додатку	Мілісекунди	Важливо для користувацького досвіду при запуску додатку
Час доставки повідомлень	Секунди	Показує, наскільки швидко повідомлення доходять до отримувача
Затримка обробки повідомлень	Мілісекунди, кількість запитів	Оцінює швидкість обробки запитів у системі
Кількість активних користувачів	Кількість, дні, години	Відображає загальну активність у системі, допомагає зрозуміти навантаження
Середня тривалість сесії	Хвилини, сесії	Важливо для розуміння залученості користувачів до чату
Кількість надісланих повідомлень	Повідомлення/день	Відображає активність користувачів у чаті
Частота помилок у відправленні повідомлень	Відсоток, кількість	Вказує на наявність технічних проблем із відправленням повідомлень
Відсоток успішних входів	Відсоток	Важливо для оцінки стабільності процесу авторизації
Кількість видалених повідомлень	Повідомлення/день	Аналізує, скільки повідомлень видаляється користувачами, що може бути показником фільтрації небажаного контенту

Воронки, які об'єднують вибрані метрики:

Воронка 1. Залучення та стабільність користувачів

Ця воронка зосереджена на оцінці того, як користувачі взаємодіють із додатком та наскільки стабільно працює система, починаючи від входу в додаток до активності в чаті.

Відсоток успішних входів
Оцінює, скільки користувачів можуть успішно увійти в додаток.

Час відкриття вікна додатку
Оцінює, наскільки швидко додаток завантажується після входу.

Кількість активних користувачів
Визначає кількість користувачів, які активно використовують додаток.

Середня тривалість сесії
Оцінює, скільки часу користувачі проводять у чаті за одну сесію.

Кількість надісланих повідомлень
Визначає загальну активність користувачів у чаті.

Воронка 2: Якість обміну повідомленнями

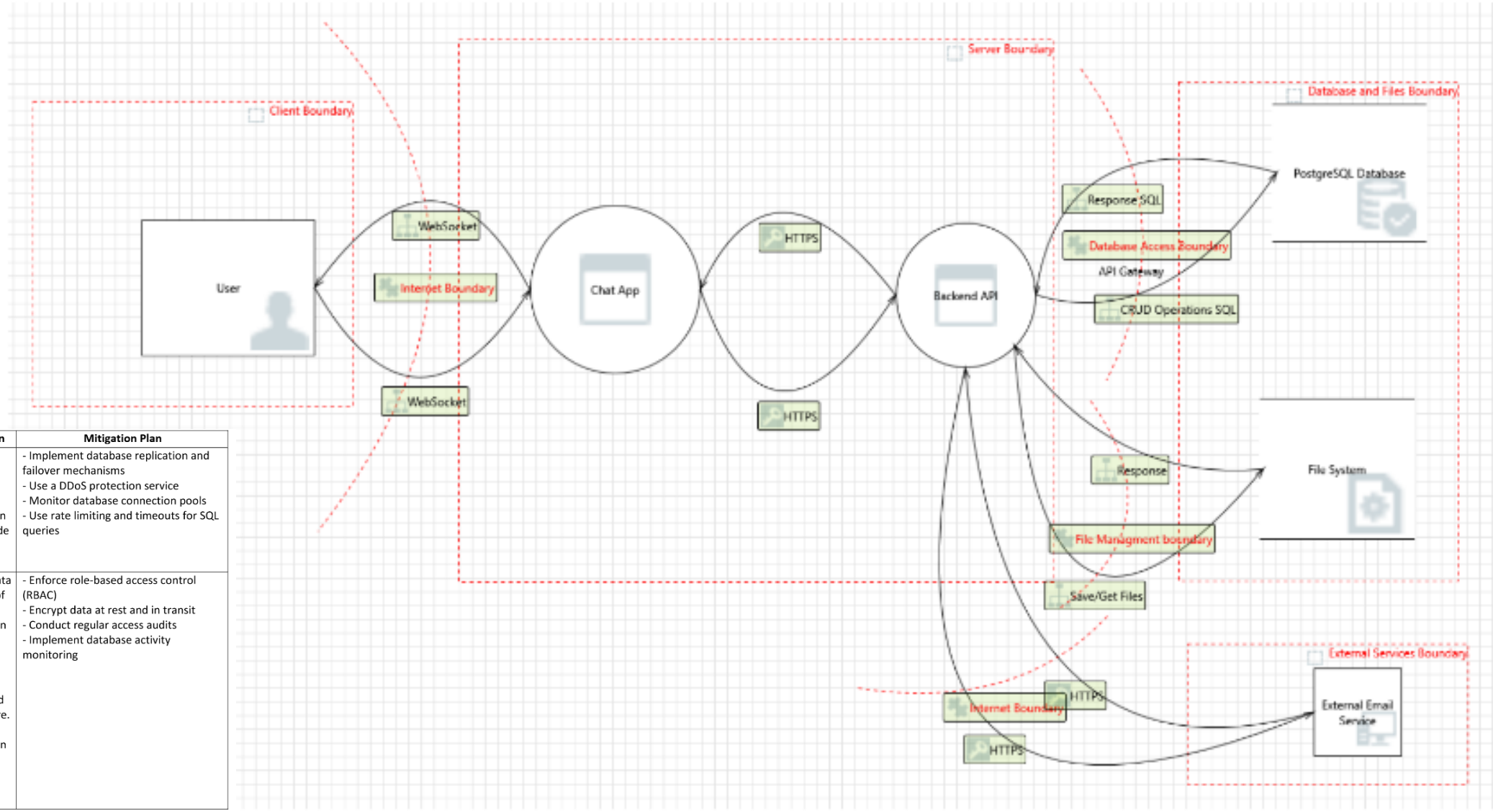
Ця воронка фокусується на якості передачі повідомлень та швидкості обробки, від моменту відправлення повідомлення до його доставки.

Час доставки повідомлень
Оцінює, скільки часу займає доставлення повідомлення до отримувача.

Затримка обробки повідомлень
Оцінює швидкість обробки запитів на сервері, що впливає на затримку повідомлень.

Частота помилок у відправленні повідомлень
Вказує, скільки повідомлень не можуть бути надіслані через технічні проблеми.

Security model



Title	Category	Interaction	Priority	Description	Mitigation Plan
Data Store Inaccessible	Denial Of Service	CRUD Operations SQL	High	An external agent prevents access to a data store on the other side of the trust boundary.	- Implement database replication and failover mechanisms - Use a DDoS protection service - Monitor database connection pools - Use rate limiting and timeouts for SQL queries
Weak Access Control for a Resource	Information Disclosure	Response SQL	High	Improper data protection of PostgreSQL Database can allow an attacker to read information not intended for disclosure. Review authorization settings.	- Enforce role-based access control (RBAC) - Encrypt data at rest and in transit - Conduct regular access audits - Implement database activity monitoring

Security model

Potential Excessive Resource Consumption	Denial Of Service	Save/Get Files	Medium	Does Backend API or File System take explicit steps to control resource consumption?	- Use rate limiting and quotas for file operations - Implement connection timeouts - Apply load balancing for file system operations - Monitor and alert on resource exhaustion
Potential Process Crash or Stop for Chat App	Denial Of Service	WebSocket	High	Chat App crashes, halts, stops, or runs slowly; in all cases violating an availability metric.	- Implement WebSocket connection health checks - Use automatic failover for Chat App - Deploy a circuit breaker pattern - Conduct load testing to simulate high traffic
Weak Access Control for a Resource	Information Disclosure	Response	High	Improper data protection of File System can allow an attacker to read information not intended for disclosure.	- Restrict access to file system operations - Encrypt sensitive files - Use intrusion detection for unauthorized file access - Regularly review and update file permissions
Data Store Inaccessible	Denial Of Service	Response	High	An external agent prevents access to a data store on the other side of the trust boundary.	- Enable database replication and backups - Use a Content Delivery Network (CDN) to cache data where possible - Apply rate limiting and IP blacklisting
Data Flow HTTPS Is Potentially Interrupted	Denial Of Service	HTTPS	High	An external agent interrupts data flowing across a trust boundary in either direction.	- Use TLS 1.3 for all HTTPS traffic - Implement mutual TLS authentication - Deploy a Web Application Firewall (WAF) - Use retry mechanisms and load balancers
Backend API Subject to Elevation of Privilege	Elevation Of Privilege	HTTPS	High	External Email Service may be able to remotely execute code for Backend API.	- Validate all incoming API requests - Apply input sanitization - Enforce strict authentication and authorization rules - Regularly patch and update the Backend API
PostgreSQL Database Data Store Could Be Corrupted	Tampering	CRUD Operations SQL	High	Data flowing across CRUD Operations SQL may be tampered with by an attacker, potentially corrupting the PostgreSQL Database.	- Encrypt data in transit using TLS - Validate SQL inputs to prevent injection - Implement database integrity checks - Log and audit all database write operations
Data Store Denies File System Potentially Writing Data	Repudiation	Save/Get Files	High	File System claims it did not write data received from an entity on the other side of the trust boundary.	- Implement secure and tamper-proof logging - Use digital signatures for file write operations - Enable audit trails for all file actions - Use timestamps to validate operation sequences

Висновки

Q&A