Bokado

Software Requirements Specification

1.0

08.06.2025

Anastasiia Bystrytska

Lead Software Engineer

**ІСТОРІЯ ЗМІН** [**II**](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.37m2jsg)

**ЗАТВЕРДЖЕННЯ ДОКУМЕНТУ** [**II**](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1mrcu09)

**1. ВСТУП** [**1**](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.46r0co2)

1.1 Мета [1](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2lwamvv)

1.2 Обсяг [1](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.111kx3o)

1.3 Визначення, акроніми, та абревіатури [1](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.3l18frh)

1.4 Посилання [1](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.206ipza)

1.5 Огляд [1](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.4k668n3)

**2. ЗАГАЛЬНИЙ ОПИС** [**2**](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2zbgiuw)

2.1 Перспектива продукту [2](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1egqt2p)

2.2 Функції продукту [2](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.3ygebqi)

2.3 Характеристики користувачів [2](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2dlolyb)

2.4 Загальні обмеження [2](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.sqyw64)

2.5 Припущення та залежності [2](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.3cqmetx)

**3. СПЕЦИФІЧНІ ВИМОГИ** [**2**](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1rvwp1q)

3.1 Вимоги до зовнішніх інтерфейсів [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.4bvk7pj)

3.1.1 Інтерфейси користувача [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2r0uhxc)

3.1.2 Апаратні інтерфейси [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1664s55)

3.1.3 Програмні інтерфейси [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.3q5sasy)

3.1.4 Інтерфейси зв’язку [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.25b2l0r)

3.2 Функціональні вимоги [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.kgcv8k)

3.2.1 Реєстрація користувачів [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.34g0dwd)

3.2.2 Редагування профілю [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.34g0dwd)

3.2.3 Пошук користувачів [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.34g0dwd)

3.2.4 Обмін повідомленнями [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.34g0dwd)

3.3 Use Cases [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.43ky6rz)

3.3.1 UC-01: Реєстрація користувача [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2iq8gzs)

3.3.2 UC-02: Надсилання повідомлення [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2iq8gzs)

3.3.3 UC-03: Додавання друга [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2iq8gzs)

3.4 Класи / Об’єкти [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.3hv69ve)

3.4.1 Клас User [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1x0gk37)

3.4.2 Клас Message [3](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.4h042r0)

3.5 Нефункціональні вимоги [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2w5ecyt)

3.5.1 Продуктивність [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1baon6m)

3.5.2 Надійність [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.3vac5uf)

3.5.3 Доступність [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2afmg28)

3.5.4 Безпека [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.pkwqa1)

3.5.5 Підтримуваність [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.39kk8xu)

3.5.6 Переносимість [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1opuj5n)

3.6 Зворотні вимоги [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.48pi1tg)

3.7 Обмеження проектування [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2nusc19)

3.8 Вимоги до логічної бази даних [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1302m92)

3.9 Інші вимоги [4](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.3mzq4wv)

**4. АНАЛІТИЧНІ МОДЕЛІ** [**4**](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.2250f4o)

4.1 Діаграма послідовність [5](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.haapch)

4.3 Діаграма потоків даних [5](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.319y80a)

4.2 Діаграма станів даних і переходів [5](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.1gf8i83)

**5. ПРОЦЕСС КЕРУВАННЯ ЗМІНАМИ** [**5**](https://docs.google.com/document/d/19G-DauUv-cdR2ZjC7yqEe3fXlWD6y_vR-x4-aKN8c_c/edit#heading=h.40ew0vw)

**1. Вступ**

**1.1 Мета**

Цей документ визначає вимоги до веб-додатку Bokado, який функціонує як соціальна платформа для пошуку друзів за спільними інтересами. Документ призначено для розробників, тестувальників, менеджерів проєкту, а також інших зацікавлених сторін, які братимуть участь у створенні, тестуванні та впровадженні продукту.

**1.2 Обсяг**

Продуктом є веб-додаток **Bokado**, який дозволяє користувачам реєструватися, створювати профілі, шукати друзів за інтересами, спілкуватися через повідомлення, а також взаємодіяти через групові чати. Продукт не підтримує відео- або голосові дзвінки, не має мобільного застосунку на першому етапі розробки. Основні цілі: забезпечення релевантного пошуку друзів, зручний інтерфейс та безпечне середовище спілкування.

**1.3 Визначення, акроніми, та скорочення**

* SRS – Software Requirements Specification
* JWT – JSON Web Token
* DB – Database (база даних)
* UI – User Interface
* GDPR – General Data Protection Regulation
* XSS, CSRF – Види веб-атак

**1.4 Посилання**

1. IEEE Std 830-1998 – IEEE Recommended Practice for Software Requirements Specifications.
2. OWASP Foundation:<https://owasp.org>
3. PostgreSQL Documentation:<https://www.postgresql.org/docs/>
4. React Documentation:

**1.5 Огляд**

Цей документ структуровано таким чином:

* Розділ 1 описує мету, область застосування та терміни
* Розділ 2 – загальний опис продукту
* Розділ 3 – детальні вимоги до інтерфейсів, функціональності, об’єктів
* Розділ 4 – аналітичні моделі (UML)

**2. Загальний опис**

**2.1 Перспектива продукту**

Продукт є самостійною веб-платформою з клієнтською частиною, сервером та БД. Він не залежить від сторонніх систем.

**2.2 Функції продукту**

* Реєстрація та автентифікація
* Пошук користувачів за інтересами
* Обмін повідомленнями (текст, зображення, голосові)  
  Управління профілем користувача
* Модерація контенту (через адмін-панель)
* Головна сторінка та навігація
* Свайп-система для друзів  
  Управління друзями
* Дошка подій та взаємодія з організаторами
* Історія взаємодій
* Налаштування користувача
* Елементи гейміфікації
* Преміум-функції

**2.3 Характеристики користувачів**

Цільова аудиторія – користувачі віком 14–30 років. Більшість має базові навички використання веб-додатків. Додаток повинен бути інтуїтивно зрозумілим, адаптивним та простим у навігації.

**2.4 Загальні обмеження**

* Frontend: браузери Chrome, Firefox, Safari, Edge
* Backend: .NET 8
* База даних: PostgreSQL
* Інтерфейс українською мовою

**2.5 Припущення та залежності**

* Очікується доступність HTTPS-сертифікату для захищеного з'єднання
* Продукт розгортається на хмарному середовищі (наприклад, Azure)

**3. Специфічні вимоги**

**3.1 Вимоги до зовнішніх інтерфейсів**

**3.1.1 Інтерфейс користувача**

Інтерфейс користувача повинен бути інтуїтивно зрозумілим, адаптивним до різних розмірів екранів, з підтримкою мобільних пристроїв. Він має передбачати перемикання між темною та світлою темами, а також сповіщення в реальному часі про події, такі як нові повідомлення чи запити в друзі. Важливо забезпечити послідовність стилів та кольорів у всьому додатку, використовуючи сучасні UI-фреймворки.

**3.1.2 Апаратні інтерфейси**

Продукт не вимагає спеціалізованого обладнання та працює у звичайному браузері. Для користування додатком необхідно мати пристрій із доступом до Інтернету (ПК, ноутбук, смартфон чи планшет). Мінімальні технічні вимоги включають наявність сучасного браузера з підтримкою JavaScript і WebSocket.

**3.1.3 Програмні інтерфейси**

Програмне забезпечення взаємодіє з зовнішніми сервісами через API. Зокрема, аутентифікація реалізується за допомогою Bearer Token. Дані зберігаються у PostgreSQL, а доступ до них здійснюється через ORM-бібліотеку Entity Framework Core.

**3.1.4 Інтерфейси зв’язку**

Передача даних між клієнтом і сервером здійснюється через HTTPS-протокол, що забезпечує шифрування та захист від атак типу man-in-the-middle Передбачено також REST API для базових запитів та обробки подій.

**3.2 Функціональні вимоги**

#### **3.2.1 Реєстрація та авторизація користувачів**

Фронтенд-частина повинна реалізувати систему реєстрації та авторизації користувачів, зокрема через email, із підтримкою відновлення паролю, що включає можливість скидання паролю через email.

#### **3.2.2 Головна сторінка**

Головна сторінка має містити динамічний лічильник активних користувачів і забезпечувати зручну навігацію по основних розділах, таких як профіль, пошук друзів, чати, преміум можливості, події та челенджі.

#### **3.2.3 Профіль користувача**

Профіль користувача повинен надавати можливість редагувати аватар, біографію, інтереси, пароль, дату народження і статус.

#### **3.2.4 Відображення користувачів та пошук**

Інтерфейс має відображати картки інших користувачів та забезпечувати систему пошуку за фільтрами, зокрема за містом і інтересами, а також можливість перегляду повного списку користувачів.

#### **3.2.5 Свайп-система**

Для зручності прийняття або відхилення потенційних друзів має бути інтегрована свайп-система.

#### **3.2.6 Управління друзями**

Користувачі повинні мати можливість додавати інших користувачів до списку улюблених друзів, а також переглядати тих, кого вони лайкнули, і тих, хто їх лайкнув.

#### **3.2.7 Особисті чати**

Особисті чати повинні підтримувати обмін текстовими і голосовими повідомленнями, зображеннями, GIF та смайликами, відображати онлайн-статус співрозмовників, а також надавати можливість відключення повідомлень і видалення їх.

#### **3.2.8 Дошка подій**

Дошка подій повинна містити фільтрацію за локацією, а також дозволяти створення, приєднання та покидання подій.

#### **3.2.9 Взаємодія користувачів та організаторів подій**

Після приєднання до події має бути реалізована взаємодія між користувачем і організатором події.

#### **3.2.10 Налаштування користувача**

Розділ налаштувань повинен дозволяти перемикання між темами, зміну особистих даних і налаштувань приватності.

#### **3.2.11 Елементи гейміфікації**

У систему має бути впроваджено елементи гейміфікації, такі як система рівнів, балів, бейджів і щотижневих челенджів з можливістю участі в них.

#### **3.2.12 Преміум-функції**

Преміум-функції мають включати додаткові фільтри, кольорові ніки, пріоритет у пошуку та візуальне виділення профілю.

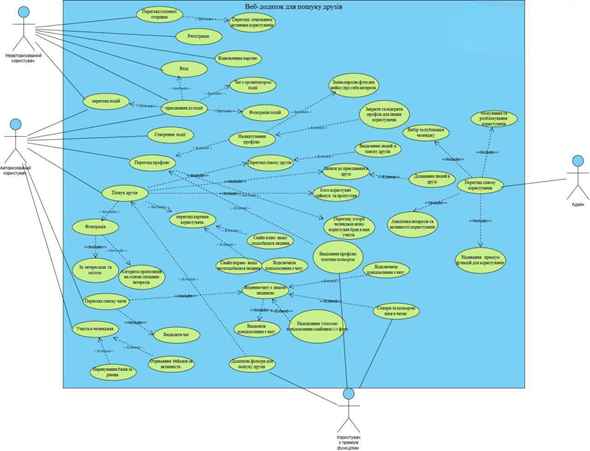
#### **3.2.13 Адаптивність та сумісність**

Інтерфейс повинен бути повністю адаптивним для мобільних і десктопних пристроїв, враховуючи вимоги доступності та кросбраузерної сумісності.

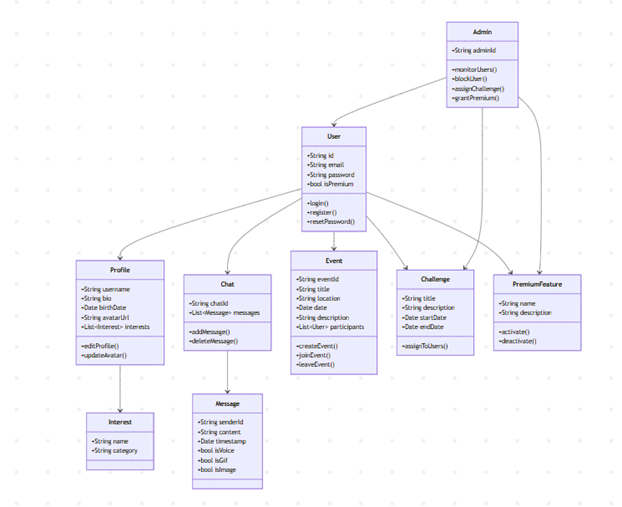
#### **3.2.14 Адміністративна панель**

Адміністративна панель має надавати можливість моніторингу зареєстрованих користувачів і їхньої активності в реальному часі, керування преміум-функціями, блокування та розблокування облікових записів у разі порушень, оновлення активних челенджів, а також перегляд аналітики щодо зростання користувачів, активних сесій, популярних інтересів і рівня взаємодії з функціоналом.

**3.3 Use Cases**

****

**3.4 Класи / Об’єкти**

****

**3.5 Нефункціональні вимоги**

Non-functional requirements may exist for the following attributes. Often these requirements must be achieved at a system-wide level rather than at a unit level. State the requirements in the following sections in measurable terms (e.g., 95% of transaction shall be processed in less than a second, system downtime may not exceed 1 minute per day, > 30 day MTBF value, etc).

**3.5.1 Продуктивність**

Система повинна відповідати на більшість запитів за час не більше 1.5 секунд. Вона має масштабуватись для підтримки до 5000 одночасних сесій, особливо в пікові години. Важливо забезпечити ефективну роботу індексів у базі даних, асинхронну обробку довготривалих операцій, а також кешування часто використовуваних даних, щоб мінімізувати навантаження на сервер. Крім того, передбачено застосування механізмів балансування навантаження для рівномірного розподілу трафіку між серверами. Продуктивність повинна регулярно тестуватись із використанням інструментів типу JMeter або k6, щоб виявити вузькі місця ще до потрапляння продукту у продуктивне середовище.

**3.5.2 Надійність**

Система має бути стійкою до відмов і забезпечувати стабільну роботу навіть у разі часткових збоїв обладнання або мережі. Для цього необхідно впровадити механізми автоматичного резервного копіювання бази даних, що здійснюється щонайменше раз на добу, з можливістю швидкого відновлення даних. Логи системи повинні вестися постійно, щоб можна було провести аналіз причин відмов. Також доцільно реалізувати автоматичне сповіщення адміністратора у разі виникнення критичних помилок. Надійність повинна перевірятись під час регресивного тестування та навантажувального тестування після кожного великого оновлення.

**3.5.3 Доступність**

Доступність сервісу має становити не менше 99.9% часу протягом місяця, тобто дозволено не більше ніж 1 година простою на місяць. У випадку технічних робіт або оновлень, система повинна попереджати користувачів заздалегідь через банер або повідомлення. Якщо сервіс тимчасово недоступний, користувач повинен побачити зрозуміле повідомлення з поясненням ситуації та приблизним часом відновлення. Крім того, рекомендується реалізувати функцію HealthCheck для перевірки доступності API та інших критичних компонентів у реальному часі. Також має бути передбачено аварійне переключення на резервний сервер у разі критичної відмови основного.

**3.5.4 Безпека**

Безпека даних користувачів має найвищий пріоритет. Всі паролі повинні зберігатися у хешованому вигляді за допомогою bcrypt з додаванням salt. Для автентифікації використовується JWT з обмеженим часом дії токена, після чого потрібна повторна авторизація. Система повинна бути захищена від типових веб-загроз: XSS, CSRF, SQL Injection. Для цього застосовуються валідація вхідних даних, обмеження прав доступу та регулярні перевірки коду на вразливості. Усі з'єднання з сервером здійснюються виключно через HTTPS. Додатково рекомендовано впровадити багатофакторну автентифікацію для адміністративних облікових записів.

**3.5.5 Підтримуваність**

Підтримуваність системи має бути високою завдяки добре організованій, модульній архітектурі. Весь код повинен супроводжуватись коментарями та технічною документацією, яка зберігається у репозиторії. Перед кожним злиттям змін у основну гілку мають виконуватись автотести, які покривають критичну функціональність. Логи мають бути достатньо інформативними для виявлення проблем без необхідності ручного дебагу. Крім того, структура проєкту повинна дозволяти легку інтеграцію нових функцій без необхідності змін у вже стабільних модулях. Регулярно необхідно проводити аудит залежностей для уникнення застарілих чи вразливих бібліотек.

**3.5.6 Переносимість**

Система повинна бути легко портованою на різні середовища розгортання. Продукт має підтримувати запуск у контейнерах Docker, що дозволяє швидке розгортання у хмарних інфраструктурах, таких як Azure або AWS. Крім того, кодова база повинна бути незалежною від конкретного хостинг-провайдера, що забезпечить її універсальність. Необхідно передбачити можливість зберігання конфігурацій у змінних середовища для полегшення CI/CD-процесів. Документація повинна включати інструкції з розгортання, конфігурації і перевірки працездатності в різних середовищах. Також слід впровадити стандартизовані API та зберігати сумісність із основними браузерами.

**3.6 Зворотні вимоги**

Система не повинна зберігати паролі у відкритому вигляді або у застарілих форматах хешування, таких як MD5 або SHA1. Заборонено передача облікових даних через незахищені канали (HTTP). Також веб-додаток не має містити трекерів, які збирають персональні дані без відома користувача. Не допускається використання сторонніх бібліотек із відомими вразливостями без оновлення. Сервіс не повинен мати прихованих функціональностей, які дозволяють доступ до даних без авторизації.

**3.7 Обмеження проектування**

Проєкт має відповідати вимогам GDPR щодо обробки персональних даних. Крім того, необхідно дотримуватись стандартів безпеки OWASP для вебзастосунків. Умови розгортання на хмарному сервері (наприклад, обмеження за обсягом пам’яті або CPU) також накладають обмеження на складність окремих сервісів. Архітектура має бути сумісна з .NET 8 та PostgreSQL. Інтерфейс користувача повинен коректно відображатись на всіх основних браузерах і підтримувати масштабування.

**3.8 Вимоги до логічної бази даних**

У додатку використовуватиметься реляційна база даних PostgreSQL. Формати зберігання даних мають відповідати нормалізованій схемі з мінімізацією дублювання. Всі поля, що містять персональні дані, повинні бути захищені належним чином – через шифрування або обмеження доступу. Передбачено щоденне резервне копіювання та журналювання критичних змін. Також слід передбачити перевірку цілісності даних та транзакційність при критичних операціях, зокрема при реєстрації та зміні профілю.

**3.9 Інші вимоги**

Усі сторонні бібліотеки, які використовуються в проєкті, повинні бути з відкритим кодом або мати відповідну ліцензію. Необхідно забезпечити механізм логування помилок та дій користувачів, що дозволить здійснювати аудит безпеки та діагностику. Веб-додаток повинен бути адаптованим для людей з обмеженими можливостями, зокрема передбачати сумісність із скрінрідерами. Крім того, слід забезпечити локалізацію інтерфейсу, починаючи з української мови з подальшим додаванням англійської. Також рекомендується наявність інструментів збору статистики для аналізу взаємодії користувача із системою.

**4. Моделі аналізу**

**4.1 Діаграма послідовності**

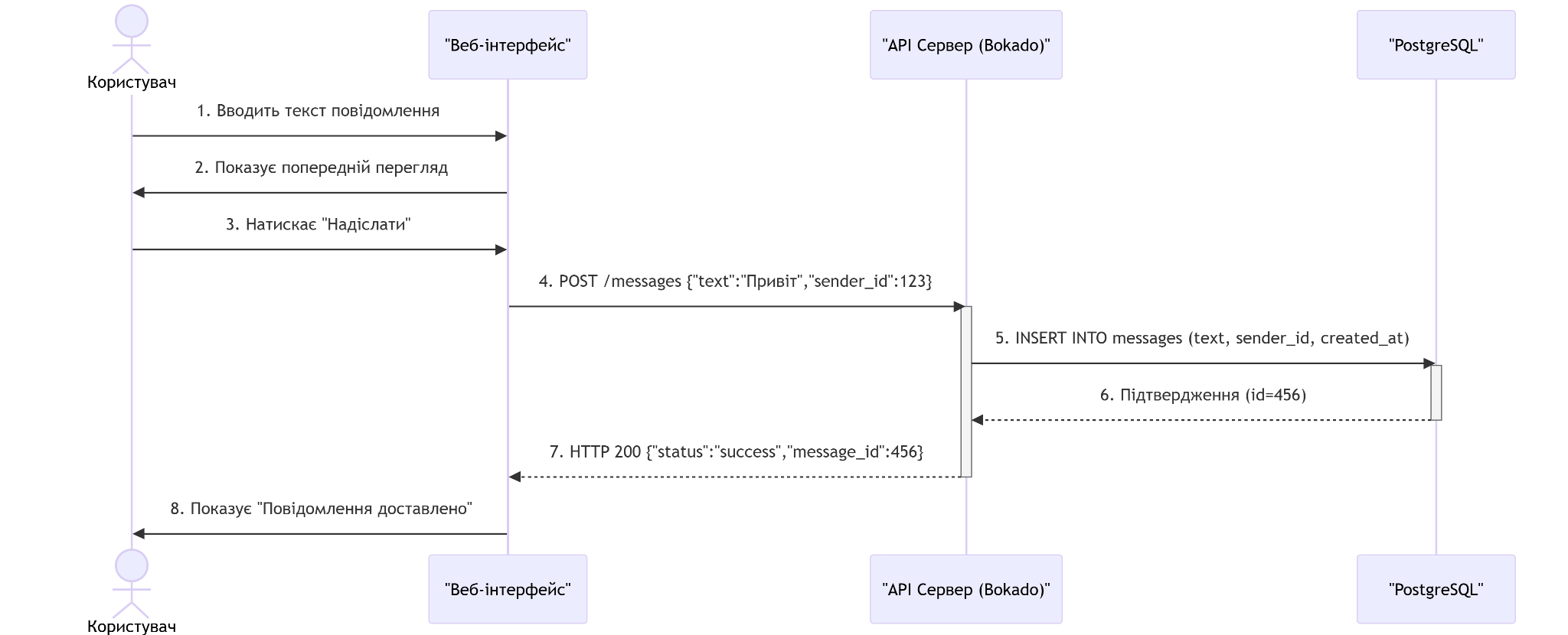


Рисунок 4.1 - Діаграма послідовності для сценарію надсилання повідомлення

**4.2 Діаграма станів-переходів (STD)**

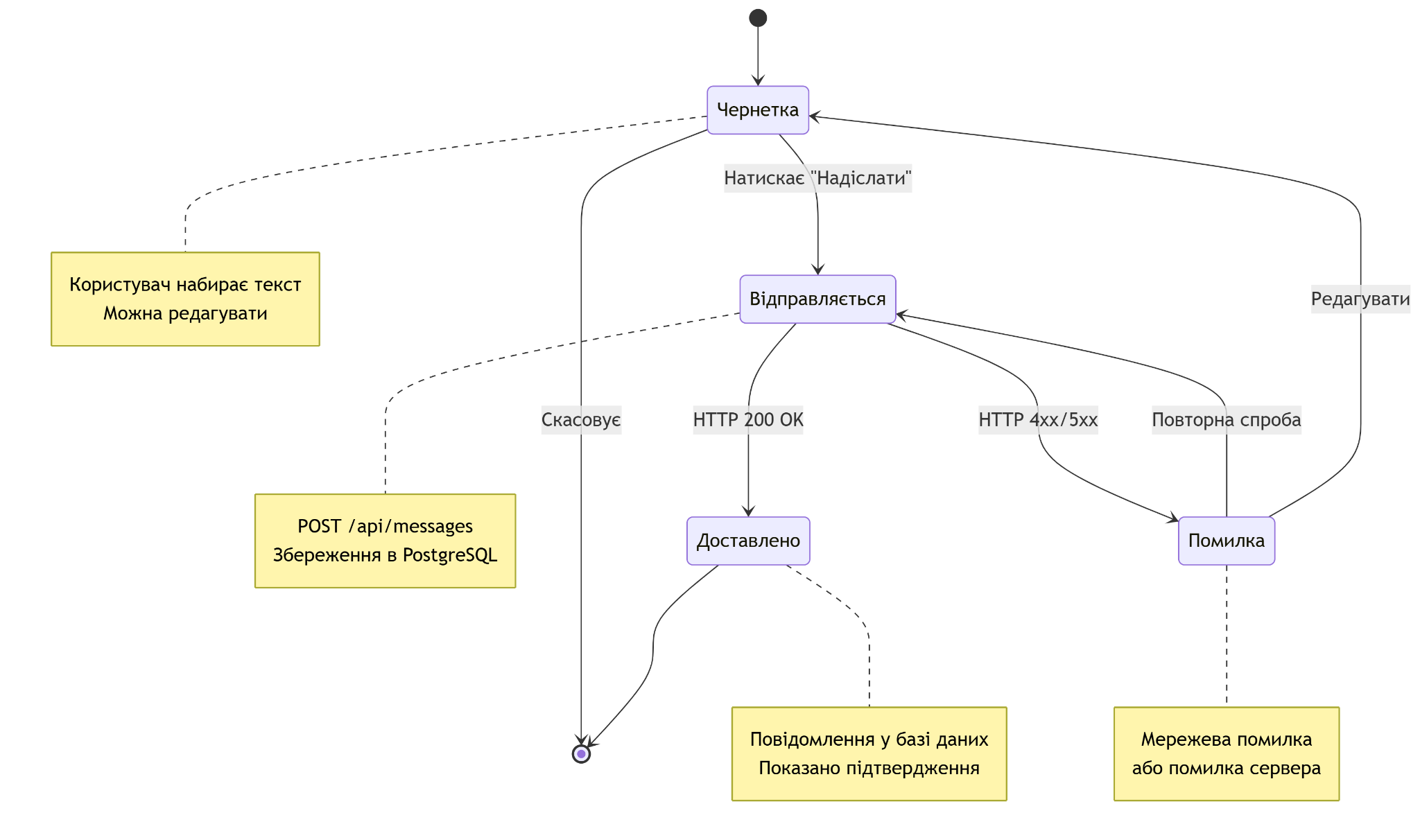
****

Рисунок 4.2 - Діаграма станів-переходів

**4.3 Діаграма потоків даних (DFD)**

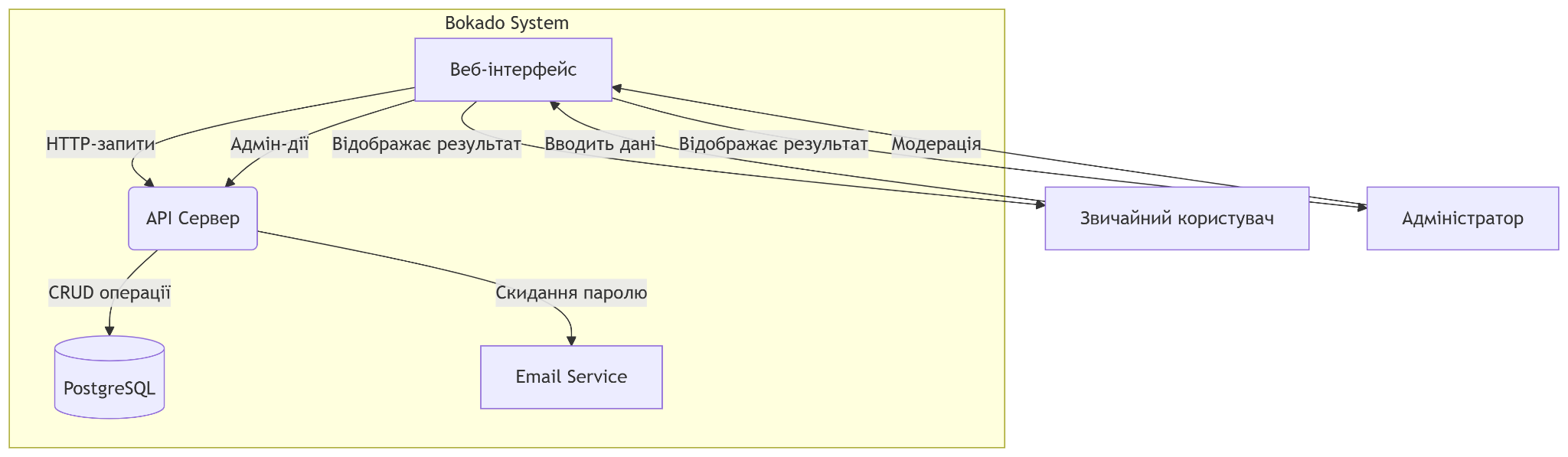
****

Рисунок 4.3 - Діаграма потоків даних

**5. Процес керування змінами**

Будь-які зміни до цього документу повинні ініціюватися через створення запиту змін (Change Request), що подається в систему керування проєктом (наприклад, GitHub Issues або Jira). Запити можуть створювати учасники команди розробки, тестувальники або менеджер проєкту. Після подачі запиту він розглядається технічним лідом і затверджується або відхиляється. У разі схвалення відповідальний розробник вносить зміни до SRS з відповідною документацією в системі контролю версій. Усі зміни мають проходити рев’ю перед злиттям у основну гілку документації.