ДОДАТОК В

Специфікація програмного забезпечення

В.1 Вступ

В.1.1 Огляд продукту

Цей документ описує функціональні та нефункціональні вимоги до веб-додатку для організації відеозустрічей з використанням Stream SDK. Додаток забезпечує можливість створення, проведення та адміністрування онлайн-зустрічей із відео- та аудіозв’язком, чатами та базовим управлінням користувачами.

В.1.2 Мета

Метою є створення сучасного веб-застосунку, що забезпечує стабільну та зручну комунікацію між користувачами у форматі відеоконференцій без складної інфраструктури.

В.1.3 Межі

Система призначена для використання індивідуальними користувачами, малими командами та організаціями, які потребують швидкого та зручного засобу відеозв’язку без необхідності встановлення додаткового програмного забезпечення. Додаток не є повноцінною платформою для дистанційного навчання або корпоративного відеозв’язку, однак може використовуватись як допоміжний інструмент для проведення онлайн-зустрічей, консультацій, мітингів або інтерв’ю.

Програмна система включає такі функціональні можливості:

* створення та управління відеозустрічами;
* підключення учасників за унікальним посиланням;
* керування мікрофоном, камерою та виходом із зустрічі;
* інтеграція з сервісом Clerk для аутентифікації користувачів;
* збереження історії викликів;
* сучасний інтерфейс з темною темою;
* адаптація до десктопних браузерів.

Обмеження:

* Робота системи залежить від доступності сторонніх сервісів (Stream, Clerk);
* Вимагається стабільне інтернет-з’єднання для відеозв’язку у реальному часі.

В.1.4 Посилання

посилання на GitHub: <https://github.com/NureKushnirukDenys/2025_B_KKP_PZPI-22-8_Kushniruk_D_O>

В.1.5 Означення та абревіатури

**SRS** (Software Requirements Specification) – специфікація вимог до програмного забезпечення.

API (Application Programming Interface) – інтерфейс програмування додатків, набір визначень взаємодії між різними програмними компонентами.

SPA (Single Page Application) – односторінковий додаток, веб-додаток, який завантажується як єдина HTML-сторінка.

JWT (JSON Web Token) – стандарт для створення токенів доступу на основі JSON.

SDK (Software Development Kit) – набір інструментів для розробки програмного забезпечення.

В.2 Загальний опис

В.2.1 Перспективи продукту

Система є самостійним веб-застосунком і може бути розширена до SaaS-платформи з платною підпискою, розкладом зустрічей, аналітикою тощо.

В.2.2 Функції продукту

* Створення відеозустрічей;
* Приєднання за посиланням;
* Підтримка чату;
* Керування камерою/мікрофоном;
* Список активних зустрічей;
* Історія викликів.

В.2.3 Характеристики користувачів

Кінцеві користувачі — це звичайні інтернет-користувачі, яким потрібен зручний засіб відеозв’язку. Від них не вимагається спеціальна технічна підготовка.

В.2.4 Загальні обмеженя

Технічні межі:

* Система функціонує лише у сучасних браузерах з підтримкою WebRTC та сучасних API (Chrome, Firefox, Edge, Safari);
* Не підтримуються застарілі браузери (наприклад, Internet Explorer);
* Потребує стабільного широкосмугового інтернет-з’єднання для забезпечення належної якості відео;
* Усі відео та чат-функції реалізовані через зовнішні сервіси (Stream Video SDK); у разі недоступності стороннього API — функціональність частково обмежена;
* Аутентифікація реалізована через Clerk — у разі його недоступності система тимчасово не зможе здійснити вхід або реєстрацію користувачів.

Організаційні межі:

* Система не містить функцій моніторингу користувацьких дій чи аналітики, окрім базової інформації про активність дзвінків;
* Вся логіка зберігання історії, облікових записів та конфігурацій передана на сторонні сервіси (Clerk, Stream).

В.2.5 Припущення й залежності

* 1. Технологічні припущення:

1. Користувачі мають доступ до сучасних веб-браузерів, які підтримують WebRTC та сучасні вебтехнології;
2. Користувачі мають стабільне інтернет-з’єднання зі швидкістю, достатньою для відеозв’язку;
3. Користувачі мають мікрофони та вебкамери (вбудовані або зовнішні), що сумісні з браузерами.

б) Бізнес-припущення:

1. Існує попит на легкі, швидкі та приватні відеозустрічі без складної реєстрації;
2. Користувачі очікують мінімалістичний, зручний і безпечний інтерфейс без зайвого функціонального перевантаження;
3. Цільова аудиторія (звичайні інтернет-користувачі) готові використовувати сучасні вебрішення для онлайн-спілкування;
4. Потенційна комерціалізація сервісу передбачає інтеграцію платних функцій у майбутньому (аналітика, брендовані кімнати, тощо.).

в) Залежності від зовнішніх факторів:

1. Залежність від стабільної роботи Stream Video API — сервісу, що забезпечує відеозв’язок;
2. Залежність від Clerk як основного постачальника функцій автентифікації та управління користувачами.

В.3 Конкретні вимоги

В.3.1 Вимоги до зовнішніх інтерфейсів

В.3.1.1 Інтерфейс користувача

* Інтуїтивно зрозумілий інтерфейс з темною темою;
* Вікно відеодзвінка з відображенням учасників;
* Інтерфейс учасників відеозустрічи праворуч;
* Кнопки для керування мікрофоном, камерою, виходом;
* Додаткові кнопки управвління.

В.3.1.2 Адаптивний інтерфейс

Додаток має підтримувати адаптивний дизайн, що забезпечує зручне користування на різних пристроях, включаючи мобільні телефони, планшети та комп’ютери.

В.3.1.3 Програмний інтерфейс

Програмний інтерфейс включає в себе:

* Stream Video SDK;
* Stream Chat SDK;
* Clerk API;
* Next.js серверні функці.

В.3.1.4 Комунікаційний протокол

* HTTPS для безпечного обміну;
* WebRTC для відеозв’язку;
* REST API.

В.3.1.5 Обмеження пам’яті

* Вебклієнт має працювати у браузерах із оперативною пам’яттю від 2 ГБ, але оптимальна робота очікується при наявності 4 ГБ оперативної пам’яті;
* На стороні клієнта додаток завантажує залежності приблизно до 5–10 МБ (включаючи шрифти, стилі, бібліотеки Clerk, Stream Video SDK, інтерфейсні компоненти);
* Серверна частина не потребує значного дискового простору, оскільки дані зберігаються через зовнішні сервіси (Stream API, Clerk) або кешуються тимчасово.

В.3.1.6 Операції

Користувач відкриває сайт у браузері та проходить аутентифікацію через Clerk.

Після входу користувач може:

* Створити нову відеокімнату;
* Приєднатися до кімнати за посиланням;
* Керувати мікрофоном і камерою (вмикати/вимикати);
* Завершити дзвінок, вийти з кімнати.

Система може:

* Ініціювати створення відеосесії через Stream API;
* Контролювати доступ на основі аутентифікації;
* Відображати список учасників та їхні статуси в реальному часі.

В.3.1.7 Функції продукту

Основні функціональні можливості продукту:

* Аутентифікація користувачів: через Clerk;
* Створення відеозустрічей: генерація унікального ідентифікатора кімнати з можливістю підключення інших користувачів;
* Проведення відеозустрічей: передача аудіо та відео в реальному часі через WebRTC/Stream API;
* Контроль доступу: обмеження доступу до кімнат неавторизованим користувачам;
* Управління відеодзвінком: перемикання мікрофона та камери, завершення дзвінка;
* Адаптивний інтерфейс: коректна робота на десктопах, планшетах, смартфонах.

В.3.2 Властивості програмного продукту

* Надійність — забезпечується за рахунок стабільної роботи SDK;
* Доступність — 24/7 через веб-браузер;
* Безпека — автентифікація через Clerk, передача даних по HTTPS;
* Супроводжуваність — проект організовано відповідно до Next.js best practices;
* Продуктивність — оптимізовано під сучасні браузери.

В.3.3 Атрибути програмного продукту

В.3.3.1 Надійність

* Система забезпечує стабільну передачу відео та аудіо у реальному часі завдяки інтеграції з перевіреним API Stream;
* При втраті інтернет-з’єднання система обробляє відновлення сеансу без перезапуску всієї сесії;

В.3.3.2 Доступність

* Сервіс доступний 24/7 за умови стабільної роботи хостингу та зовнішніх API (Stream, Clerk);
* Користувач може отримати доступ з будь-якого пристрою, що підтримує сучасний браузер;
* Передбачено базову підтримку доступності інтерфейсу, адаптивний дизайн;
* У разі помилок авторизації чи підключення користувачу виводиться відповідне повідомлення з поясненням.

В.3.3.3 Безпека

* Аутентифікація та керування сесіями здійснюється через Clerk, який відповідає сучасним вимогам безпеки (JWT, OAuth, захист від CSRF/XSS);
* Відео та аудіопотоки передаються зашифрованими каналами через WebRTC із підтримкою TLS/DTLS;
* Доступ до кімнат може бути обмежений лише авторизованим користувачам;
* Дані користувачів не зберігаються на сервері проєкту — вони керуються безпечно через сторонні сервіси.

В.3.3.4 Супроводжуваність

* Кодова база структурована за принципами компонентного підходу (React + Next.js);
* Використання TypeScript спрощує виявлення помилок ще на етапі розробки;
* Уся логіка взаємодії з зовнішніми сервісами ізольована в окремих модулях, що спрощує підтримку та оновлення;
* Ведеться документація по ключових функціях, зокрема для API-інтеграцій.

В.3.3.5 Переносимість

* Клієнтська частина є повністю веборієнтованою та не потребує встановлення;
* Завдяки використанню Next.js можлива подальша трансформація в PWA (Progressive Web App) або експорт у мобільні додатки за допомогою Expo чи Capacitor;
* Платформа легко переноситься між хостингами, підтримуючи стандартний стек (Node.js + React);
* Залежність від зовнішніх API централізована, що полегшує перенесення на інші сервіси при потребі.

В.3.3.6 Продуктивність

* Швидке завантаження за рахунок оптимізації Next.js, використання шрифтів Google ;
* Передача відео адаптується до умов мережі — якість зменшується при низькій пропускній здатності (завдяки можливостям Stream SDK);
* Завдяки використанню WebRTC латентність відеозв’язку зводиться до мінімуму.

В.4 Додаткові матеріали

В.4.1 Глосарій

Табл. 1 - Глосарій

|  |  |
| --- | --- |
| Термін | Визначення |
| Clerk | Сервіс для аутентифікації та управління користувачами у веб-застосунках. |
| Stream API | Хмарне рішення для створення відео- та аудіозв’язку у реальному часі. |
| Next.js | Фреймворк на базі React для розробки веб-додатків із серверним рендерингом. |
| WebRTC | Технологія для забезпечення відео- та аудіозв’язку в реальному часі між браузерами. |
| PWA (Progressive Web App) | Веб-додаток, що може функціонувати як нативний мобільний застосунок. |
| JWT (JSON Web Token) | |  | | --- | |  |  |  | | --- | | Стандарт для безпечної передачі інформації між сторонами як JSON-об’єктів. | |
| SDK (Software Development Kit) | Набір інструментів для розробників для інтеграції сторонніх сервісів. |

В.4.2 Список використаних джерел

1. IEEE Standards Association. IEEE 29148-2011 – Systems and software engineering – Life cycle processes – Requirements engineering. – 2011.
2. Next.js Documentation. The React Framework for Production [Електронний ресурс]. – Режим доступу: <https://nextjs.org/docs>.
3. Clerk Docs. Authentication for modern web apps [Електронний ресурс]. – Режим доступу: <https://clerk.com/docs>.
4. Stream Documentation. Build real-time video apps [Електронний ресурс]. – Режим доступу: <https://getstream.io/video/>.
5. ShadCN UI. Modern UI components for React [Електронний ресурс]. – Режим доступу: <https://ui.shadcn.com/>.
6. Google Fonts. Typography for the web [Електронний ресурс]. – Режим доступу: <https://fonts.google.com/>.