Вебсистема для керування колекціями музичних творів та їх прослуховування

Software Requirements Specification

1.0

21.03.2025

Проценко Артем Романович

Шпак Денис Олександрович

# **ІСТОРІЯ ЗМІН**

| **Дата** | **Опис** | **Автор** | **Коментарі** |
| --- | --- | --- | --- |
| 19.03.2025 | Створено пункти 1.1 - 1.3 | Шпак Денис Олександрович |  |
| 20.03.2025 | Створено пункти 1.4 - 1.5, 2.1 - 2.3 | Проценко Артем Романович |  |
| 21.03.2025 | Створено пункти 2.4, 2.5, 3.1 - 3.2 | Проценко Артем Романович |  |
| 22.03.2025 | Створено пункти 3.3 - 3.5 | Шпак Денис Олександрович |  |

# **ЗАТВЕРДЖЕННЯ ДОКУМЕНТУ**

Наступну специфікацію вимог до програмного забезпечення було прийнято та схвалено:

| **Підпис** | **Друковане ім’я** | **Назва** | **Дата** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Зміст**

[**ІСТОРІЯ ЗМІН 2**](#_1epsunl9gfa3)

[**ЗАТВЕРДЖЕННЯ ДОКУМЕНТУ 2**](#_322ip9wj56ks)

[**1. ВСТУП 4**](#_n54jb0vcocxb)

[1.1 Огляд продукту 4](#_tr6kukwhsxcq)

[1.2 Мета 4](#_a5c9dhplqn3)

[1.3 Межі 5](#_henqk1egpkmg)

[1.4 Посилання 6](#_aay1r3y86y5f)

[1.5 Означення та абревіатури 7](#_8z7hgv2xon8z)

[**2. ЗАГАЛЬНИЙ ОПИС 9**](#_5hweysddyceo)

[2.1 Перспективи продукту 9](#_x4ervtua0e75)

[2.2 Функції продукту 11](#_vqptrr8j2luo)

[2.3 Характеристики користувачів 12](#_bzyuona2tupd)

[2.4 Загальні обмеження 14](#_ce93sanht9)

[2.5 Припущення й залежності 15](#_5y49svcjn3z7)

[**3. КОНКРЕТНІ ВИМОГИ 18**](#_9uugjb4v0qku)

[3.1 Вимоги до зовнішніх інтерфейсів 18](#_19gre8kh044b)

[3.1.1 Інтерфейс користувача 18](#_2ed02cyawfqq)

[3.1.2 Апаратний інтерфейс 18](#_1kfsvhn2sg6j)

[3.1.3 Програмний інтерфейс 18](#_123lbedej2b6)

[3.1.4 Комунікаційний протокол 19](#_2j9p4l4mezms)

[3.1.5 Обмеження пам’яті 19](#_b5ihvbkn6m4l)

[3.1.6 Операції 19](#_iylqgw9kds7r)

[3.1.7 Функції продукту 20](#_zc4elvylz9of)

[3.1.8 Припущення й залежності 20](#_y67patxktifh)

[3.2 Властивості програмного продукту 20](#_wv7r339zlcu3)

[3.3 Атрибути програмного продукту 29](#_19pu3u541as6)

[3.3.1 Надійність 29](#_2z6zawdqdfjz)

[3.3.2 Доступність 29](#_khyack2igjm7)

[3.3.3 Безпека 29](#_g93cn6hv6cs9)

[3.3.4 Супроводжуваність 29](#_drkzn0tf1s8g)

[3.3.5 Переносимість 30](#_2bt3npnst8rf)

[3.3.6 Продуктивність 30](#_me739eflqvzx)

[3.4 Вимоги бази даних 30](#_bw9ec49hyjsg)

[3.5 Інші вимоги 32](#_k0dbsjskqu5h)

# 1. **ВСТУП**

## 1.1 **Огляд продукту**

Метою цього продукту є створення онлайн-платформи для завантаження, організації, оцінки та комунікації навколо музичних творів. Користувачі зможуть організовувати свої музичні колекції, оцінювати твори, коментувати їх і спілкуватися через приватні повідомлення. Функції пошуку, фільтрації та сортування забезпечують зручність навігації, а перегляд профілю користувача та його колекцій дозволяє інтерактивно взаємодіяти з іншими учасниками.

Цільова аудиторія охоплює меломанів, які шукають зручний інструмент для зберігання та організації музики, а також користувачів, які хочуть активно коментувати і обговорювати твори. Продукт буде цікавий тим, хто шукає приватне спілкування або соціалізацію з іншими користувачами, а також людям, які потребують швидкого доступу до музичних творів за конкретними критеріями. Платформа підходить для користувачів різних вікових груп і музичних уподобань.

## 

## **1.2 Мета**

Вебсистема керування музичними колекціями призначена для організації, збереження та прослуховування музичних творів, а також взаємодії між користувачами. Система надає можливість завантаження та управління аудіофайлами, створення персоналізованих колекцій, оцінювання треків, залишення коментарів і приватного листування. Реалізовано функціонал пошуку, фільтрації та сортування музичних композицій за різними критеріями, а також механізм персональних рекомендацій на основі активності користувача. Крім того, кожен користувач має власний профіль, де відображаються його створені колекції та взаємодії в системі.

Основна мета продукту — забезпечити зручну платформу для зберігання та організації особистих музичних колекцій із можливістю інтерактивної взаємодії. Вебдодаток не є комерційною платформою для розповсюдження або ліцензованого стримінгу музики, а передбачає лише використання аудіофайлів, завантажених користувачами відповідно до їхніх прав власності.

Ключові переваги системи включають централізацію музичної бібліотеки, можливість швидкого доступу та редагування плейлистів, інтерактивність через коментарі та приватні повідомлення, а також потокове відтворення музики без затримок. Завдяки веборієнтованому підходу система доступна з будь-якого пристрою з доступом до інтернету. Оптимізовані алгоритми пошуку та рекомендацій дозволяють користувачам швидко знаходити нові треки відповідно до їхніх уподобань.

Система розробляється з урахуванням високих стандартів безпеки, продуктивності та зручності використання, що робить її ефективним інструментом для управління музичними колекціями.

## 

## **1.3 Межі**

Програмний продукт включає серверну частину на Go з фреймворком Chi, базу даних PostgreSQL та фронтенд на React з TypeScript. Він забезпечує завантаження, організацію, оцінку та комунікацію навколо музичних творів. Користувачі можуть створювати колекції, коментувати твори, приватно спілкуватися, а також здійснювати пошук, фільтрацію та сортування контенту.

Система дозволяє ефективно взаємодіяти з музичним контентом, забезпечуючи швидку обробку запитів і стабільне зберігання даних. Використання Go гарантує продуктивність, PostgreSQL – надійність, а React із TypeScript – зручний та інтерактивний інтерфейс. Продукт відповідає сучасним вимогам масштабованості, безпеки та швидкодії, створюючи комфортне середовище для взаємодії користувачів з музичним контентом.

## **1.4 Посилання**

У цьому розділі наведено перелік документів та джерел, які використовуються в цій специфікації програмного забезпечення. Вони містять нормативні вимоги, технічні стандарти та інші документи, що впливають на реалізацію та функціональність вебсистеми керування музичними колекціями.

1. ISO/IEC 25010:2011 – Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models. Доступно на офіційному сайті International Organization for Standardization (ISO).
2. General Data Protection Regulation (GDPR) (EU) 2016/679 – Регламент Європейського Союзу щодо захисту персональних даних. Офіційний текст доступний на EUR-Lex (<https://eur-lex.europa.eu/>).
3. RFC 7519 – JSON Web Token (JWT) – Специфікація формату токенів для аутентифікації. Доступно на сайті Internet Engineering Task Force (IETF) (<https://datatracker.ietf.org/doc/html/rfc7519>).
4. RFC 6455 – The WebSocket Protocol – Специфікація протоколу WebSocket для двосторонньої комунікації. Доступно на сайті IETF (<https://datatracker.ietf.org/doc/html/rfc6455>).
5. Документація по React – Офіційна документація бібліотеки React для розробки фронтенду. Доступно на сайті Meta (<https://react.dev/>).
6. Документація по Go-chi – Офіційний гайд по фреймворку Go-chi для побудови RESTful API. Доступно на сайті GitHub (<https://github.com/go-chi/chi>).
7. Документація PostgreSQL – Інструкції з налаштування та використання бази даних PostgreSQL. Доступно на сайті PostgreSQL (<https://www.postgresql.org/docs/>).

Ці документи містять нормативні вимоги, технічні рекомендації та методичні вказівки, що використовуються при розробці вебсистеми. Доступ до офіційних джерел можливий через відповідні організації або їхні вебресурси.

## **1.5 Означення та абревіатури**

Означення:

* Музичний твір – окремий аудіофайл, завантажений у систему користувачем.
* Колекція – набір музичних творів, організований користувачем за певними критеріями.
* Рейтинг твору – система оцінювання музичних творів користувачами.
* Коментар – текстовий відгук, залишений під музичним твором користувачем.
* Приватне повідомлення – особисте повідомлення між користувачами системи.
* Профіль користувача – сторінка, що містить інформацію про користувача, його колекції та активність.
* Пошук та фільтрація – механізм знаходження музичних творів за різними критеріями (жанр, виконавець, дата тощо).

Абревіатури:

* SRS (Software Requirements Specification) – Специфікація вимог до програмного забезпечення.
* UI (User Interface) – Користувацький інтерфейс.
* API (Application Programming Interface) – Програмний інтерфейс взаємодії додатків.
* DB (Database) – База даних.
* JWT (JSON Web Token) – Формат токенів для аутентифікації.
* HTTPS (HyperText Transfer Protocol Secure) – Протокол безпечної передачі даних у мережі Інтернет.
* GDPR (General Data Protection Regulation) – Загальний регламент захисту персональних даних (ЄС).
* REST (Representational State Transfer) – Архітектурний стиль для побудови веб-сервісів.
* SQL (Structured Query Language) – Мова структурованих запитів для роботи з базами даних.
* OAuth 2.0 – Протокол авторизації для надання доступу до ресурсів без передачі пароля.

# **2. ЗАГАЛЬНИЙ ОПИС**

## **2.1 Перспективи продукту**

Платформа, яка передбачає функціонал завантаження музичних творів, їх організацію у колекції, оцінювання та комунікацію через коментарі, має низку важливих характеристик, які сприяють розвитку музичної спільноти та створенню середовища для глибокого обміну думками та ідеями між користувачами. Одним із ключових аспектів такої платформи є можливість завантажувати музичні твори, що дозволяє користувачам ділитися своєю творчістю або ж знаходити цікаві роботи інших. Це створює сприятливу атмосферу для розвитку музичних напрямів та стильових напрямків, даючи змогу кожному висловити себе через власне музичне мистецтво. Подібна можливість організовує простір для вивчення та оцінки музичних творів іншими користувачами, забезпечуючи інтерактивність і доступність до всіх учасників.

Особливу увагу на такій платформі приділяють організації музичних творів у колекції, що надає користувачам гнучкість у збереженні та впорядкуванні матеріалів відповідно до власних уподобань. Це створює умови для легкого доступу до певних груп творів, дозволяючи легко знаходити бажану музику за певними критеріями. Колекції можуть бути як приватними, так і публічними, що дозволяє гнучко управляти приватністю та доступом до створених матеріалів. Завдяки цьому механізму користувач має можливість більш ретельно контролювати, які твори будуть доступні для загального перегляду, а які залишатимуться виключно для особистого використання. Оцінювання творів є ще одним важливим елементом функціоналу, оскільки воно дає змогу учасникам платформи висловлювати свою думку щодо якості музичних композицій. Такий підхід не лише сприяє обміну думками, але й стимулює творчість, мотивуючи авторів до покращення своїх робіт та розвитку власного стилю.

Функціонал коментарів, що супроводжує кожен твір, також відіграє важливу роль у розвитку платформи. Завдяки коментарям користувачі можуть обговорювати деталі виконання, особливості аранжування, технічні аспекти твору або навіть ідеї, які стоять за його створенням. Це дозволяє не лише оцінити твір з різних точок зору, але й створює можливості для більш глибокого розуміння музичного процесу, а також для безпосередньої взаємодії між людьми з подібними інтересами. Важливим є також механізм приватного листування, який забезпечує конфіденційність у спілкуванні між користувачами. Це дає змогу обговорювати приватні питання чи особисті ідеї без сторонніх поглядів, що є важливим для розвитку більш інтимних і конструктивних дискусій.

Платформа також передбачає можливість пошуку, фільтрації та сортування музичних творів, що дає змогу швидко знаходити потрібні композиції за різними критеріями, такими як жанр, популярність, дата додавання тощо. Цей функціонал забезпечує зручність і ефективність пошуку, що дозволяє користувачам не витрачати час на надмірне перебування у великому обсязі контенту. Крім того, важливим є перегляд профілю користувача, який дає змогу ознайомитися з його музичними колекціями та творами. Це дозволяє не лише знайомитися з творчістю окремих осіб, але й взаємодіяти з ними на більш глибокому рівні, що сприяє розвитку спільноти та сприяє створенню нових колаборацій.

Таким чином, платформа, що поєднує всі ці функції, створює умови для активного та взаємодіючого музичного середовища. Вона дає можливість не лише для публікації та організації музичних творів, але й для глибоких комунікацій, які можуть включати обговорення, оцінювання та приватні дискусії між учасниками. Такий підхід сприяє розвитку творчості, забезпечує відкритість у спілкуванні, а також дає можливість користувачам не лише знаходити музику, але й взаємодіяти з іншими учасниками спільноти, що, у свою чергу, може призвести до нових творчих досягнень і колаборацій.

## **2.2 Функції продукту**

Розроблювана вебсистема має забезпечувати користувачам зручний та інтерактивний спосіб прослуховування музики, управління власними аудіоколекціями та соціальної взаємодії. Основні функції додатка включають потокове відтворення музичних треків у високій якості, можливість створення та редагування персональних плейлистів, додавання нових композицій у бібліотеку, а також налаштування рекомендацій на основі уподобань користувача.

Користувачі можуть завантажувати власні музичні твори, організовувати їх у колекції за власним бажанням, а також оцінювати та коментувати композиції, обговорюючи їх з іншими учасниками платформи. Додаток підтримує функцію приватного листування.

Передбачено також розширені можливості пошуку, фільтрації та сортування музичних творів за різними критеріями, що сприятиме швидкому доступу до бажаного контенту.

Також додаток має мати функції відображення персонального профілю користувача, у якому відображатимуться створені ним музичні колекції, завантажені треки та взаємодія з іншими користувачами.

Додаток має мати інтуїтивно зрозумілий інтерфейс з оптимізованою навігації та мати функції зручного пошуку за виконавцями, жанрами та альбомами.

Функціональність платформи орієнтована на створення комфортного та гнучкого середовища для музичних ентузіастів, яке поєднує можливості сучасних потокових сервісів із розширеними соціальними функціями, інструментами персоналізації контенту та зручним керуванням власними музичними колекціями

## **2.3 Характеристики користувачів**

Платформа, яка передбачає завантаження, організацію та оцінку музичних творів, має чітко визначені категорії користувачів, кожна з яких має свої функціональні можливості та доступ до різних елементів системи. Користувачі цієї платформи поділяються на чотири основні категорії: неавторизовані користувачі, авторизовані користувачі, модератори та адміністратори. Кожен тип користувача має свою специфічну роль та функціональні можливості, що сприяють ефективній взаємодії та управлінню контентом на платформі.

Неавторизовані користувачі є тими, хто не пройшов процес реєстрації або авторизації на платформі. Цей тип користувачів обмежений у своїх можливостях, але має доступ до базових функцій, які дозволяють ознайомитися з платформою. Неавторизовані користувачі можуть переглядати музичні твори, здійснювати пошук контенту, ознайомлюватися з найбільш популярними творами та користувачами, а також мати змогу реєструватися або входити в систему. Це дозволяє їм знайомитись із основними можливостями платформи, але без можливості брати участь у більш активних функціях, таких як коментування чи оцінка творів.

Авторизовані користувачі отримують розширений доступ до функціоналу платформи після реєстрації та входу в систему. Вони мають можливість створювати та публікувати власні музичні твори, коментарі та оцінки, що дозволяє їм активно взаємодіяти з іншими учасниками спільноти. Авторизовані користувачі можуть використовувати механізм оцінки творів і коментарів інших учасників, що сприяє формуванню активної та взаємодіючої спільноти. Окрім цього, авторизовані користувачі можуть подавати звіти про неналежний або неприпустимий контент, що дає їм можливість брати участь у контролі якості контенту на платформі, а також комунікувати з іншими учасниками через чат.

Модератори мають розширену роль, що включає всі можливості звичайних користувачів, але з додатковими функціями для управління контентом і поведінкою користувачів. Модератори мають право перевіряти та модерувати публікації та коментарі, що дозволяє їм видаляти або приховувати ті твори, які порушують правила платформи або містять неналежний контент. Вони також обробляють звіти, подані користувачами, стосовно порушень поведінки або контенту, що допомагає підтримувати порядок та забезпечувати безпеку користувачів. Модератори займаються активним контролем за якістю обговорень та публікацій, що сприяє створенню здорового середовища для взаємодії.

Адміністратори мають найвищі привілеї на платформі і володіють всіма можливостями модераторів, а також додатковими функціями для управління користувачами та їх правами. Вони можуть змінювати ролі інших користувачів, надаючи або позбавляючи їх прав модераторів чи адміністраторів. Адміністратори мають доступ до адміністративної панелі, яка дозволяє моніторити активність користувачів і контенту на платформі. Вони здійснюють загальне управління платформою, забезпечуючи її належне функціонування та відповідність встановленим стандартам.

Таким чином, чітке розмежування ролей та функціональних можливостей користувачів на платформі сприяє ефективному управлінню контентом, забезпечує високий рівень взаємодії між учасниками, а також дає можливість підтримувати безпечну та продуктивну атмосферу для творчості та комунікації. Кожна категорія користувачів виконує свою специфічну роль у забезпеченні функціональності платформи, що в кінцевому підсумку сприяє її успішному розвитку та підтримці активної спільноти.

## **2.4 Загальні обмеження**

Розробка музичного вебдодатка передбачає низку загальних обмежень, які впливають на дизайн та функціональність системи. Одним із ключових факторів є апаратні обмеження пристроїв користувачів, зокрема продуктивність процесора, обсяг оперативної пам’яті та швидкість інтернет-з’єднання. Ці фактори можуть впливати на швидкість завантаження сторінок, буферизацію потокового відтворення та плавність роботи інтерфейсу.

Інтеграція з іншими додатками та сервісами є важливим елементом роботи вебдодатка. Система передбачає взаємодію з базами даних для зберігання музичних треків, користувацьких колекцій та коментарів, а також із зовнішніми сервісами для отримання додаткових даних про композиції. Однак така інтеграція може мати технічні обмеження, зокрема затримки при отриманні інформації або залежність від стабільності роботи сторонніх API.

Паралельна робота та асинхронна обробка даних є важливими для забезпечення безперебійного функціонування додатка. Вебдодаток повинен підтримувати одночасне виконання кількох завдань, таких як відтворення музики, завантаження нових треків та перегляд профілів користувачів, що вимагає оптимізації роботи з потоками та ефективного управління ресурсами браузера.

Функції аудиту та контролю необхідні для збереження історії завантажень, оцінок та коментарів, що дозволить відслідковувати активність користувачів. Це створює додаткове навантаження на серверну частину та вимагає оптимізації зберігання даних.

Безпека та конфіденційність даних користувачів є пріоритетними у розробці додатка. Оскільки система підтримує приватне листування, коментарі та персоналізовані рекомендації, важливим є шифрування даних, надійна аутентифікація та контроль доступу. Ці вимоги можуть обмежувати вибір технологій або архітектурних рішень.

Критичність вебдодатка полягає у забезпеченні стабільної роботи та безперервного доступу до музичного контенту. Збої у відтворенні або втрата даних можуть негативно вплинути на користувацький досвід і призвести до зниження лояльності аудиторії.

## **2.5 Припущення й залежності**

Інтерфейс користувача та сумісність браузерів: Вебсистема має бути доступною на різних веб-браузерах та мобільних пристроях. Будь-які зміни в політиках сумісності браузерів або оновлення веб-стандартів можуть вимагати адаптації платформи, щоб забезпечити безперебійний доступ та функціональність на різних платформах. Оновлення браузерів або операційних систем може вимагати тестування нових версій інтерфейсу для забезпечення коректної роботи всіх функцій.

Інтеграція з музичними сервісами та форматами: Вебсистема передбачає завантаження та відтворення музичних творів у різних форматах. Якщо у майбутньому з’являться нові стандарти або зміни в популярних музичних форматах, система може потребувати оновлення для підтримки цих нововведень. Крім того, зміни в API або протоколах музичних платформ, що використовуються для інтеграції, можуть вимагати адаптації системи для коректної взаємодії з іншими сервісами.

Алгоритми пошуку та фільтрації контенту: Користувачі мають можливість здійснювати пошук і фільтрацію музичних творів за різними критеріями, такими як жанр, автор, популярність тощо. Якщо зміняться алгоритми пошуку або додадуться нові фільтри для сортування контенту, це може вплинути на ефективність пошукових запитів і на те, як користувачі взаємодіють з платформою. Оновлення алгоритмів може потребувати змін у базах даних та їх структурі для оптимізації процесу пошуку.

Механізм рекомендацій: Вебсистема може використовувати алгоритми для рекомендацій користувачам музичних творів на основі їхніх вподобань та оцінок. Оновлення в алгоритмах рекомендацій або введення нових технологій, таких як машинне навчання для покращення точності рекомендацій, може вимагати значних змін у backend системі для обробки та аналізу даних користувачів, а також для забезпечення коректного відображення рекомендацій на користувацьких сторінках.

Обсяг даних і навантаження на сервери: Вебсистема передбачає завантаження та зберігання великих обсягів медіа-контенту, що може значно вплинути на навантаження на сервери та бази даних. Якщо кількість користувачів або обсяг контенту збільшиться, необхідно буде здійснити масштабування серверної інфраструктури, щоб забезпечити безперебійне та швидке завантаження музичних творів, а також зберігання їх у високій якості. Це включає в себе оновлення серверів для зберігання медіа-файлів, оптимізацію баз даних та балансування навантаження.

Конфіденційність і захист даних: Вебсистема буде обробляти персональні дані користувачів, такі як їхні вподобання, музичні колекції, коментарі та приватні повідомлення. У разі змін у законах про захист персональних даних, такі як нові вимоги GDPR або зміни в міжнародних стандартах конфіденційності, система повинна адаптувати свої механізми зберігання та обробки персональних даних. Також можливі зміни в механізмах шифрування даних і конфіденційності повідомлень, які вимагатимуть змін в архітектурі системи.

Інтерактивні функції та комунікація між користувачами: Користувачі можуть коментувати твори та взаємодіяти через приватні повідомлення. Якщо в майбутньому з’являться нові формати комунікацій або зросте попит на більш інтерактивні функції (наприклад, відеочат чи інтеграція з соціальними мережами), це може вимагати значних змін в інтерфейсі користувача або в технології обробки даних. Такі нововведення можуть потребувати розширення функціональних можливостей сервера та змін в архітектурі баз даних.

Модерація контенту та управління користувачами: Вебсистема дозволяє користувачам взаємодіяти з контентом і один з одним через коментарі та відгуки. У разі розвитку нових механізмів для модерації контенту, таких як автоматичне виявлення неналежних коментарів або використання машинного навчання для фільтрації спаму, необхідно буде внести зміни в систему управління контентом. Зміни в політиці модерації або нові вимоги до контролю за взаємодією користувачів можуть також вимагати змін у внутрішній логіці платформи та розширення прав модераторів.

Залежність від сторонніх сервісів для хостингу контенту: Для зберігання великих медіа-файлів, таких як музичні твори, платформа може залежати від сторонніх хмарних хостинг-сервісів. Якщо ці сервіси змінять умови своїх послуг, це може вплинути на стабільність і доступність контенту на платформі. Можливі зміни в політиках хмарних провайдерів, таких як зміни тарифів або обмеження на обсяг зберігання, можуть вимагати адаптації інфраструктури платформи для забезпечення безперебійного функціонування.

Забезпечення масштабованості для великої аудиторії: У разі збільшення кількості користувачів або значного зростання кількості завантажених музичних творів платформа повинна бути здатна масштабувати свої ресурси. Це передбачає розширення серверної інфраструктури, оновлення алгоритмів обробки даних і використання більш потужних технологій для підтримки високої швидкості завантаження та відтворення контенту навіть за високого навантаження.

# **3. КОНКРЕТНІ ВИМОГИ**

## **3.1 Вимоги до зовнішніх інтерфейсів**

У цьому розділі визначаються, як система буде взаємодіяти з іншими системами, користувачами та пристроями. Деталізується, яким чином продукт обмінюватиметься інформацією з іншими системами чи пристроями.

### **3.1.1 Інтерфейс користувача**

Інтерфейс вебдодатка повинен бути інтуїтивно зрозумілим та адаптивним до різних розмірів екранів, включаючи мобільні пристрої, планшети та настільні комп'ютери. Він має підтримувати темний і світлий режими для зручності користувачів. Основні елементи взаємодії включають навігаційне меню, кнопки відтворення та керування треками, поля введення для пошуку, систему оцінювання та коментарів під треками. Користувачі повинні мати можливість створювати, редагувати та видаляти персоналізовані колекції, переглядати профілі інших користувачів з можливістю приватного листування.

### **3.1.2 Апаратний інтерфейс**

Оскільки додаток є веб орієнтованим, основними апаратними вимогами є пристрої з підтримкою сучасних браузерів (Google Chrome, Mozilla Firefox, Safari). Для забезпечення стабільної роботи рекомендується використовувати пристрої з не менш ніж 4 ГБ оперативної пам'яті, двоядерним процесором з частотою не менше 2.0 ГГц та стабільним підключенням до Інтернету зі швидкістю від 5 Мбіт/с.

### **3.1.3 Програмний інтерфейс**

Додаток використовує RESTful API для обміну даними між клієнтською та серверною частиною. Серверна частина побудована з використанням Go-chi, що забезпечує можливість виконання CRUD-операцій над треками, колекціями та профілями користувачів. Для аутентифікації та авторизації застосовується JWT та OAuth 2.0. Для збереження даних використовується реляційна бази даних PostgreSQL. Клієнтська частина побудована за допомогою бібліотеки React з використанням мови програмування TypeScript.

### **3.1.4 Комунікаційний протокол**

Для безпечної передачі даних використовується протокол HTTPS. Це забезпечує шифрування трафіку між клієнтом і сервером. Також, в додатку передбачається використання веб сокетів для реалізації функції реального часу, наприклад, для отримання повідомлень у приватному листуванні.

### **3.1.5 Обмеження пам’яті**

Вебдодаток повинен використовувати не більше 200 МБ оперативної пам'яті на стороні клієнта під час нормального навантаження. База даних на сервері повинна бути оптимізована для зберігання великих обсягів даних про треки, користувачів та їхню активність. Кешування використовується для прискорення доступу до популярних треків і зниження навантаження на сервер.

### **3.1.6 Операції**

Основні операції, доступні користувачам системи, включають керування музичними треками та колекціями, взаємодію між користувачами та пошук контенту. Користувачі можуть завантажувати, редагувати та видаляти треки, а також створювати, редагувати та видаляти колекції. Також передбачено можливість додавання треків у плейлисти та керування ними. Користувачі можуть переглядати та редагувати свій профіль, залишати оцінки треків, додавати коментарі, блокувати або додавати інших користувачів у друзі. Реалізовано систему приватного листування у режимі реального часу. Для полегшення навігації передбачено пошук треків за назвою, виконавцем, жанром, фільтрацію за рейтингом, популярністю та датою, а також сортування за алфавітом, популярністю.

### **3.1.7 Функції продукту**

Основні функціональні можливості вебдодатка включають реєстрацію нових користувачів та авторизацію через email/пароль або OAuth. Передбачено можливість відновлення пароля через email. Система містить вбудований музичний плеєр для потокового відтворення музики без необхідності завантаження файлів, що підтримує перемотку, регулювання гучності, перемикання треків та відображення обкладинок альбомів. Користувачі можуть створювати персоналізовані плейлисти, додавати та видаляти треки зі своїх колекцій. Взаємодія між користувачами включає можливість оцінювання треків, залишення коментарів та надсилання приватних повідомлень. Система рекомендацій генерує персоналізовані списки треків на основі активності користувачів, відображає популярні та трендові композиції, а також пропонує рекомендації щодо плейлистів на основі вподобань слухачів.

### **3.1.8 Припущення й залежності**

Припускається, що серверна частина додатка працює на операційній системі Linux з використанням PostgreSQL для зберігання даних. Для стабільної роботи додаток залежить від наявності стабільного інтернет-з'єднання у користувачів. Також передбачається, що браузери користувачів підтримують сучасні вебтехнології, такі як WebSockets та HTML5.

## 

## **3.2 Властивості програмного продукту**

**3.2.1 Завантаження музичних творів**

3.2.1.1 **Вступ**

Функціонал завантаження музичних творів передбачає можливість користувачам додавати аудіофайли до системи для подальшого використання. Це дозволяє користувачам створювати власні музичні бібліотеки, ділитися своїми творами з іншими учасниками системи, а також організовувати їх за допомогою різноманітних фільтрів і категорій.

3.2.1.2 **Вхідні дані**

До системи завантажуються аудіофайли у форматах, таких як MP3, WAV або FLAC. Вхідними даними є сам файл, його метадані (назва, виконавець, жанр, дата створення), а також додаткові дані, які можуть бути введені користувачем під час процесу завантаження, такі як опис твору, теги для пошуку, можливе додавання до певної колекції тощо.

3.2.1.3 **Обробка**

При завантаженні файлу система перевіряє його на відповідність допустимим форматам і розмірам. Якщо файл не відповідає вимогам, користувач отримує відповідне повідомлення про помилку. Після успішного завантаження файл зберігається на сервері, а метадані заносяться в базу даних. Всі дані, включаючи аудіофайл та метадані, будуть асоційовані з профілем користувача, який виконав завантаження, для подальшого доступу та обробки.

3.2.1.4 **Вихідні дані**

Після успішного завантаження музичного твору користувач отримує підтвердження з інформацією про успішне додавання. Твір з’являється в його персональному профілі, а також стає доступним для перегляду іншими користувачами, якщо це не налаштовується як приватний контент.

3.2.1.5 **Обробка помилок**

Якщо під час завантаження виникає помилка (наприклад, файл занадто великий або має невідомий формат), система виводить повідомлення про помилку із зазначенням причини. В разі некоректних метаданих (наприклад, порожні поля чи неправильно введені значення), користувачеві також буде надано зворотний зв'язок із вказівкою на необхідність виправлення введених даних.

**3.2.2 Організація музичних творів**

3.2.2.1 **Вступ**

Функціонал організації музичних творів у колекціях дозволяє користувачам групувати свої твори за категоріями, що можуть бути визначені самими користувачами. Колекції можуть мати різні теми або цілі, такі як жанри, настрій, персональні вподобання чи події.

3.2.2.2 **Вхідні дані**

До системи вводяться дані про колекцію: її назва, опис, категорії (якщо такі є), а також перелік творів, що додаються до цієї колекції. Користувач може створювати нові колекції або редагувати існуючі.

3.2.2.3 **Обробка**

Після введення назви і опису колекції, система асоціює створену колекцію з обраними музичними творами. Кожен твір може бути доданий до декількох колекцій одночасно. Користувач може змінювати склад колекції, додавати або видаляти твори. Всі зміни синхронізуються в реальному часі, що дозволяє користувачеві безперервно редагувати і оптимізувати свої колекції.

3.2.2.4 **Вихідні дані**

Після організації творів в колекціях користувач отримує змогу переглядати всі свої колекції, а також змінювати їх у будь-який час. Колекції стають доступними для інших користувачів, якщо це не налаштовується як приватний контент.

3.2.2.5 **Обробка помилок**

У разі невдалого створення чи редагування колекції система виводить повідомлення про помилку з описом проблеми, наприклад, якщо колекція не містить творів або її назва вже існує в базі даних. Користувач отримує інструкцію щодо виправлення помилки.

**3.2.3 Оцінювання музичних творів**

3.2.3.1 **Вступ**

Функціонал оцінювання музичних творів дозволяє користувачам ставити оцінки у вигляді вподобайок різним творам на основі їхніх власних вподобань. Це дозволяє створити певну рейтинг-систему, що відображає популярність та якість творів з точки зору користувачів.

3.2.3.2 **Вхідні дані**

Користувач натискає на кнопку у вигляді пальця направленого догори або донизу у залежності від думки щодо музичного твору. Оцінка фіксується в системі та асоціюється з користувачем та його профілем.

3.2.3.3 **Обробка**

Система зберігає кожну оцінку у базі даних та розраховує кількість вподобайок та протилежних оцінок для кожного твору. Даний показник може використовуватися як для загального відображення думки коритувачів так і для побудови рекомендацій або рейтингу на основі відношення кількості оцінок до часу за який їх виставлено або враховуючи те наскільки часто за останій час почали ставити ту ци іншу оцінку.

3.2.2.4 **Вихідні дані**

Користувачі бачать кількість вподобайок та негативних оцінок під піснею та мають розуміння думки суспільства про цю пісню.

3.2.3.5 **Обробка помилок**

У разі несправності мережі система виведе повідомлення про помилку.

**3.2.4 Комунікація через коментарі**

3.2.4.1 **Вступ**

Функціонал коментування дає можливість користувачам спілкуватися один з одним через коментарі, що залишаються під музичними творами. Це дозволяє не лише обговорювати сам твір, а й створювати простір для творчого обміну думками, порадами чи підтримкою між користувачами.

3.2.4.2 **Вхідні дані**

Вхідними даними є текстовий коментар, який користувач хоче залишити під певним музичним твором. Коментар може містити текстову інформацію, а також емодзі або інші символьні позначення для вираження емоцій.

3.2.4.3 **Обробка**

Після введення тексту коментаря система перевіряє його на наявність символів, аби уникати створення пустих коментарів. Якщо коментар відповідає стандартам, він додається до твору, де буде доступний для перегляду іншими користувачами.

3.2.4.4 **Вихідні дані**

Після успішного додавання коментаря, він стає доступним усім користувачам, що переглядають відповідний твір. Користувачі також можуть побачити інші коментарі, що залишені під твором.

3.2.4.5 **Обробка помилок**

Якщо коментар не відповідає правилам, система виводить повідомлення про помилку та пропонує відредагувати або видалити коментар.

**3.2.5 Приватне листування**

3.2.5.1 **Вступ**

Функціонал приватного листування дозволяє користувачам обмінюватися повідомленнями в особистому режимі, що не є доступним для інших учасників системи. Цей функціонал забезпечує конфіденційність і дає змогу налагоджувати персональні зв'язки.

3.2.5.2 **Вхідні дані**

Користувач вводить текст повідомлення, яке бажає надіслати іншому користувачу через інтерфейс системи. Повідомлення може включати текст або емодзі.

3.2.5.3 **Обробка**

Система зберігає повідомлення в базі даних, асоціюючи їх з профілями відповідних користувачів. Повідомлення надсилаються тільки тим користувачам, до яких вони адресовані, і є конфіденційними.

3.2.5.4 **Вихідні дані**

Користувач отримує підтвердження про відправлення повідомлення, а також може переглядати повідомлення, які отримав від інших користувачів.

3.2.5.5 **Обробка помилок**

Якщо повідомлення не було доставлено (через відсутність інтернет-з'єднання або іншій технічній проблемі), система виводить відповідне повідомлення про помилку і пропонує повторити спробу надсилання.

**3.2.6 Пошук, фільтрація та сортування творів**

3.2.6.1 **Вступ**

Функціонал пошуку, фільтрації та сортування дозволяє користувачам ефективно знаходити бажані музичні твори серед великої кількості контенту. Це дає можливість здійснювати пошук за різними критеріями, такими як назва, автор, жанр, дата створення тощо.

3.2.6.2 **Вхідні дані**

Користувач вводить запит у пошукову строку або налаштовує фільтри для конкретних параметрів. Які надсилатимуться параметрами до серверу.

3.2.6.3 **Обробка**

Система здійснює пошук серед усіх завантажених творів, застосовує фільтри і сортує результати відповідно до критеріїв запиту. Всі знайдені твори відображаються в результатах пошуку відповідно до встановлених фільтрів або порядку сортування.

3.2.6.4 **Вихідні дані**

Користувач отримує список музичних творів, що відповідають його запиту. Результати можуть бути відсортовані за різними параметрами, такими як рейтинг, дата додавання, тощо.

3.2.6.5 **Обробка помилок**

У разі відсутності результатів пошуку система виводить відповідне повідомлення про те, що жодні твори не відповідають заданим критеріям, і пропонує уточнити запит.

**3.2.7 Реєстрація та вхід**

3.2.7.1 **Вступ**

Функціонал реєстрації та входу забезпечує користувачам можливість створювати облікові записи на платформі та входити в систему для доступу до всіх функцій. Реєстрація може бути виконана як вручну (за допомогою електронної пошти та пароля), так і через сторонній сервіс, наприклад, Google. Цей функціонал є необхідним для персоналізації досвіду користувача та забезпечення безпеки особистих даних.

3.2.7.2 **Вхідні дані**

Для реєстрації вручну користувач надає такі дані: електронна пошта, пароль, а також можливе введення додаткової інформації, такої як ім’я, прізвище, фотографія профілю тощо. Якщо реєстрація здійснюється через Google, вхідними даними є акаунт користувача в Google, з якого буде отримано відповідний доступ до платформи. Для входу в систему користувач повинен ввести правильні облікові дані: електронну пошту та пароль у разі ручної реєстрації або авторизуватися через Google для використання відповідного акаунта.

3.2.7.3 **Обробка**

При реєстрації вручну система перевіряє коректність введених даних. Якщо зазначена електронна пошта вже зареєстрована в базі даних, система повідомляє користувача про необхідність використати іншу пошту або відновити доступ. У разі введення некоректних даних (наприклад, неправильно сформованого пароля або пошти) система видає відповідне повідомлення про помилку, вказуючи на причину відмови.

Якщо реєстрація відбувається через Google, система здійснює запит до сервісу Google для отримання доступу до базової інформації акаунта (ім’я, електронна пошта). Якщо авторизація через Google успішна, система автоматично створює обліковий запис користувача на основі отриманих даних або прив'язує його до вже існуючого акаунта в разі відповідності пошти.

Для входу система перевіряє надані дані з даними, що зберігаються в базі. У разі успішної автентифікації користувач отримує доступ до особистого профілю та можливості використовувати всі функції платформи. Для входу через Google система перевіряє автентичність запиту через API Google.

3.2.7.4 **Вихідні дані**

Після успішної реєстрації або входу користувач отримує підтвердження про успішний доступ до системи та переадресацію на головну сторінку або профіль. У разі невдалої реєстрації або входу система виводить повідомлення про помилку та можливі варіанти виправлення помилки (наприклад, неправильно введена пошта чи пароль).

3.2.7.5 **Обробка помилок**

У разі невірно введених даних при ручній реєстрації або вході система надає корисні підказки щодо причин помилки (наприклад, «пароль занадто слабкий», «пошта вже зареєстрована»). Якщо користувач забув пароль, система надає можливість скинути його за допомогою функції відновлення через електронну пошту. Якщо відбувся збій при вході через Google, система виводить повідомлення про помилку з рекомендаціями щодо перевірки підключення до Google акаунта або дозволів доступу.

## **3.3 Атрибути програмного продукту**

### **3.3.1 Надійність**

Система повинна забезпечувати стабільну роботу та мінімізувати ризики відмови. У разі збою додаток повинен автоматично відновлювати роботу або повідомляти адміністратора про критичні помилки. Очікуваний рівень безвідмовної роботи – не менше 99,5% часу на рік. База даних повинна підтримувати резервне копіювання для відновлення даних у разі збою.

### **3.3.2 Доступність**

Система повинна бути доступною для користувачів 99,9% часу на рік, за винятком запланованих технічних оновлень. Серверна інфраструктура повинна бути спроєктована таким чином, щоб забезпечити безперебійний доступ до сервісу, використовуючи балансування навантаження та механізми автоматичного масштабування у разі високого трафіку.

### **3.3.3 Безпека**

Усі дані, що передаються між клієнтом і сервером, повинні бути зашифровані за допомогою протоколу HTTPS та алгоритмів хешування SHA-256 для збереження інформації в базі даних. Аутентифікація користувачів реалізується через JWT, що забезпечує безпечний доступ до ресурсів. Реалізовано механізми захисту від атак типу SQL Injection, XSS, CSRF, а також обмеження спроб входу для запобігання атакам перебору паролів. Доступ до приватних повідомлень та інших персональних даних обмежений ролями та дозволами.

### **3.3.4 Супроводжуваність**

Архітектура системи повинна бути модульною, що спрощує підтримку та оновлення окремих компонентів. Кодова база повинна бути добре документованою, що дозволяє швидке впровадження змін. Система повинна підтримувати можливість безперервного розгортання (CI/CD).

### **3.3.5 Переносимість**

Система повинна працювати на різних браузерах (Google Chrome, Mozilla Firefox, Microsoft Edge, Safari) та підтримувати мобільні пристрої. Серверний додаток має бути сумісним з ОС Linux та мати можливість розгортання у контейнерах Docker.

### **3.3.6 Продуктивність**

Час відповіді на основні операції (запит інформації про трек, завантаження плейлиста, пошук) не повинен перевищувати 2 секунд при звичайному навантаженні. Вебсокети повинні забезпечувати доставку повідомлень у приватному листуванні не пізніше 1 секунди після їх надсилання. Оптимізація бази даних, використання індексів та кешування Redis.

## **3.4 Вимоги бази даних**

Вимоги до бази даних для системи, яка передбачає завантаження музичних творів, їх організацію у колекції, оцінювання, комунікацію через коментарі, приватне листування, пошук та фільтрацію творів, а також перегляд профілю користувача і створених ним музичних колекцій, є важливою складовою при проектуванні цієї системи. Основною метою проектування бази даних є забезпечення ефективного зберігання і обробки даних, що знижує час доступу до інформації та оптимізує роботу користувача з системою. Для досягнення цієї мети база даних повинна бути структурована таким чином, щоб забезпечити швидкий доступ до всіх типів даних, коректне виконання операцій, а також забезпечити надійність та безпеку інформації.

Насамперед, для зберігання музичних творів необхідно мати таблицю, яка б містила дані про кожен твір, зокрема його назву, виконавця, жанр, рік випуску, формат файлу та інші метадані, які можуть бути корисні для пошуку та фільтрації творів. Враховуючи, що користувач може завантажувати нові твори, система повинна забезпечити можливість додавання таких записів без порушення структури бази даних. Для підтримки функціоналу організації музичних творів у колекції необхідно забезпечити зв'язок між творами та колекціями, що створюються користувачами.

Оскільки система передбачає комунікацію через коментарі, потрібно створити таблицю для зберігання коментарів під кожним твором. Важливо, щоб кожен коментар містив ідентифікатор користувача, текст коментаря, а також дату та час його публікації, що дозволить організувати правильну хронологію коментарів. Крім того, для забезпечення приватного листування користувачів потрібно розробити відповідну таблицю, яка б містила дані про кожне повідомлення: відправника, отримувача, текст повідомлення та дату його відправлення.

Враховуючи, що система передбачає пошук, фільтрацію та сортування творів, база даних повинна підтримувати індексацію за ключовими полями, такими як назва твору, виконавець, жанр та рік випуску. Це дозволить значно пришвидшити операції пошуку та фільтрації, навіть при великому обсязі даних. Для забезпечення швидкого доступу до профілю користувача та перегляду створених ним музичних колекцій необхідно розробити структуру, яка б містила дані про кожного користувача, зокрема його ім’я, контактну інформацію, а також зв’язок з його колекціями та творами.

Необхідно забезпечити регулярне резервне копіювання бази даних, щоб у разі збою або втрати даних можна було відновити інформацію.

Оскільки система повинна бути масштабованою і забезпечувати підтримку великої кількості користувачів і музичних творів, важливо врахувати, що база даних має бути спроектована таким чином, щоб її можна було легко розширювати. Це означає, що структура бази повинна бути гнучкою, з можливістю додавання нових таблиць або змінення існуючих без порушення роботи системи.

## **3.5 Інші вимоги**

**3.5.1 Вимоги до захисту персональних даних**

Система повинна відповідати вимогам GDPR. щодо обробки та зберігання персональних даних користувачів. Усі персональні дані повинні зберігатися у зашифрованому вигляді, а користувачі повинні мати можливість отримати копію своїх даних або видалити обліковий запис за запитом. Реалізація політики конфіденційності повинна передбачати чітке інформування користувачів про обробку їхніх даних та отримання згоди перед збором персональної інформації.

**3.5.2 Вимоги до прав інтелектуальної власності**

Система повинна забезпечувати дотримання авторських прав на музичні твори. Користувачі не можуть завантажувати контент, який порушує авторські права третіх осіб. Передбачено механізм модерації контенту та можливість подання скарг на незаконне розповсюдження треків.

**3.5.3 Вимоги до екологічної ефективності**

Розгортання серверної інфраструктури має бути оптимізованим для мінімального використання ресурсів. Система повинна використовувати ефективні алгоритми кешування та оптимізацію запитів для зниження навантаження на сервери, що сприяє зменшенню споживання електроенергії.