

ДОДАТОК А

Графічний матеріал кваліфікаційної роботи

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Кафедра ЕОМ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: **Музичний стримінговий сервіс на основі
хмарних технологій**

виконав:
здобувач гр. КІУКІу-22-2
Какуня М.В.

керівник:
ас. кафедри ЕОМ
Сітніков В.І.

Харків 2025

Мета роботи

Метою роботи є розробка музичного стримінгового сервісу на основі хмарних технологій з акцентом на зниження вартості серверної інфраструктури, спрощення процесу залучення розробників до проекту та забезпечення масштабованості рішення відповідно до потреб бізнесу.



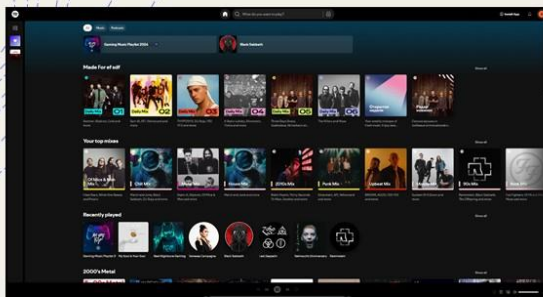
Цілі та задачі роботи

- проаналізувати ринок музичних стримінгових сервісів та обґрунтувати вибір хмарних технологій;
- реалізувати мікросервісний підхід для масштабованості та гнучкості;
- розробити серверну частину з використанням .NET, PostgreSQL, RabbitMQ та Docker;
- реалізувати клієнтську частину на React із сучасним масштабованим UI;
- побудувати хмарну інфраструктуру для зберігання, обробки та доставки медіаконтенту.

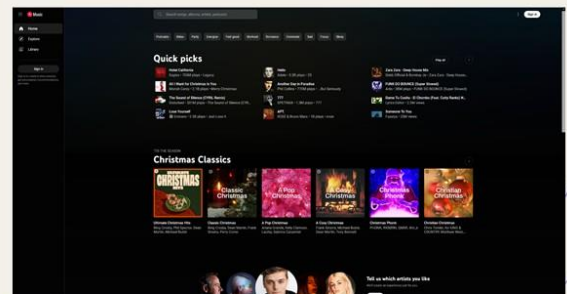


3

Існуючі рішення



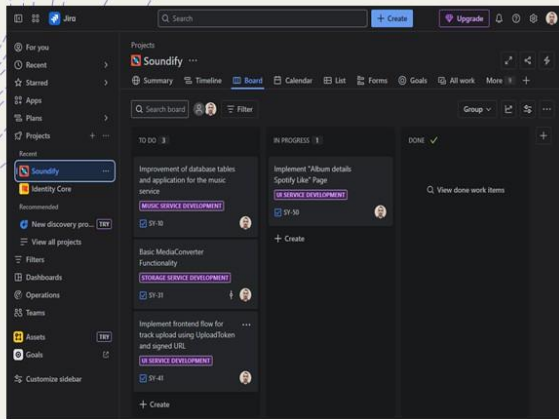
Spotify



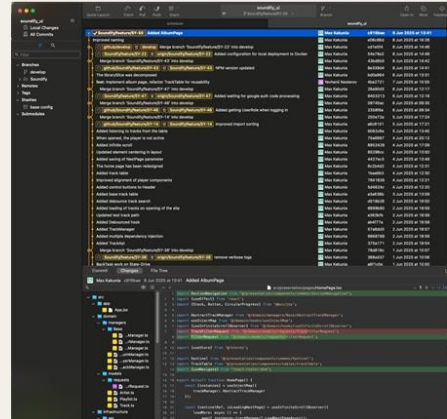
YouTube Music

4

Управління процесом розробки



Jira



Git

5

Стек технологій

Клієнтська частина



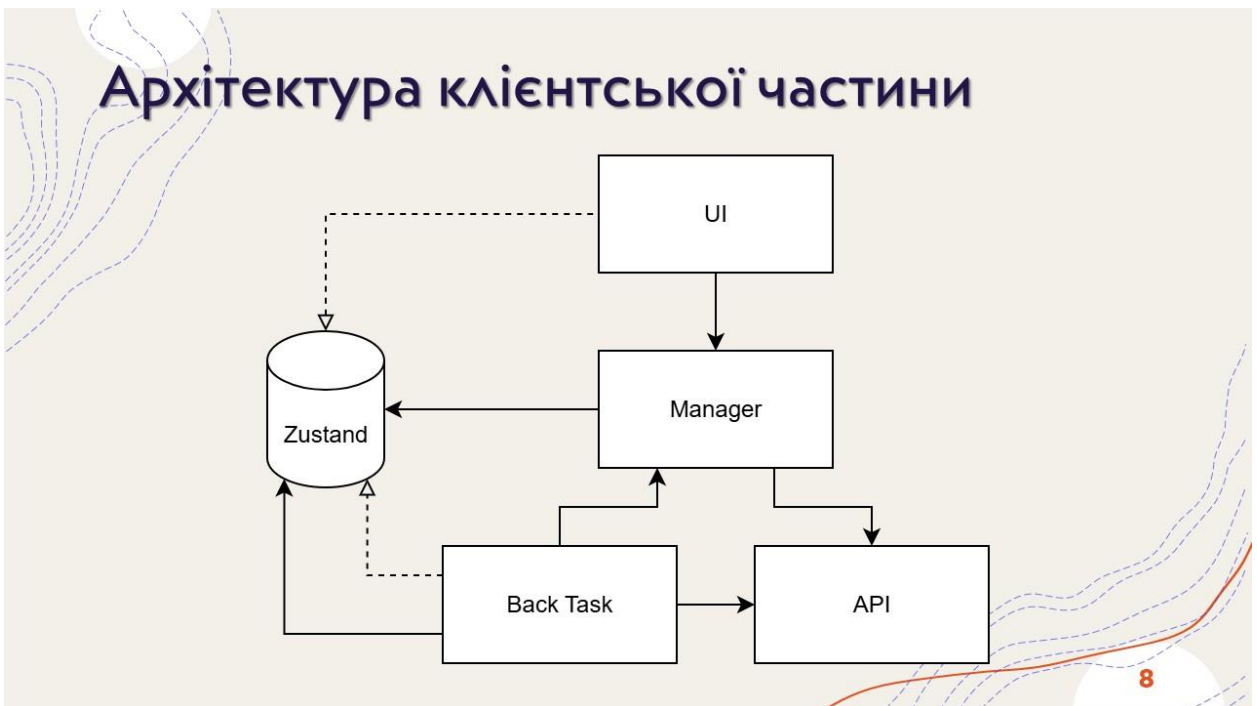
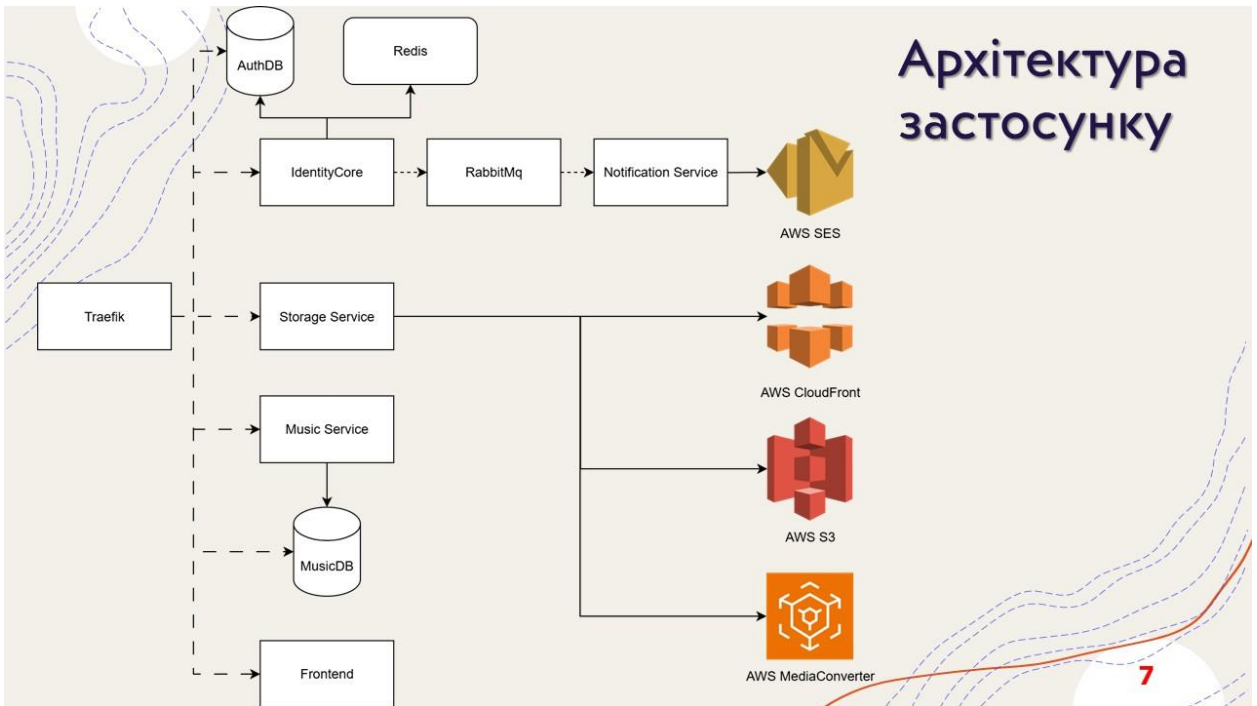
Серверна частина



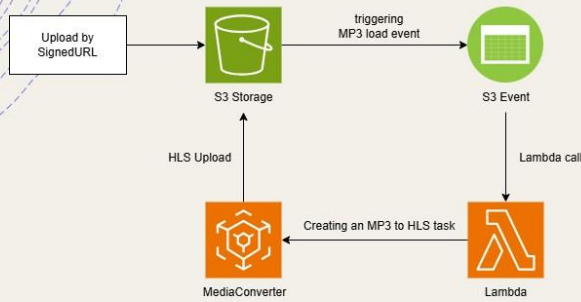
Хмарна частина



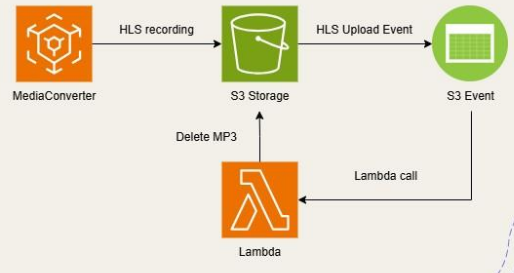
6



Отримання та конвертація MP3 у хмарі



Завантаження MP3 та конвертація в HLS

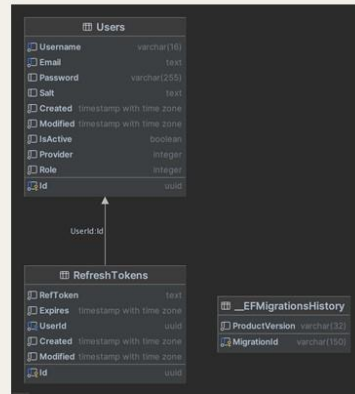


Видалення MP3 після конвертації в HLS

Структура баз даних

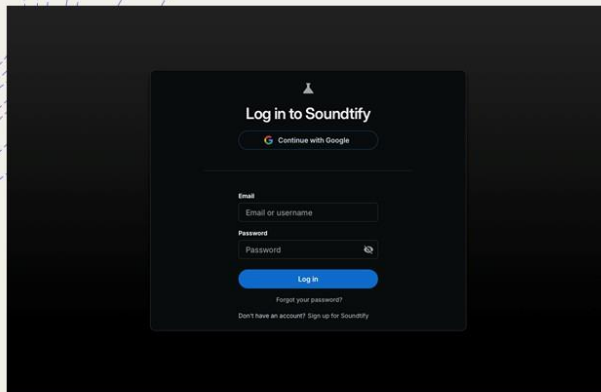


Структура MusicDB

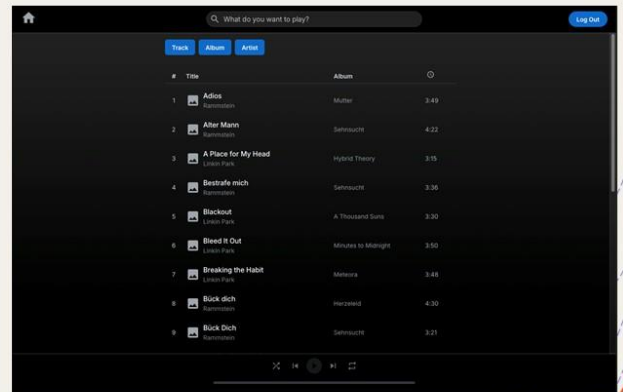


Структура AuthDB

Реалізація клієнтської частини



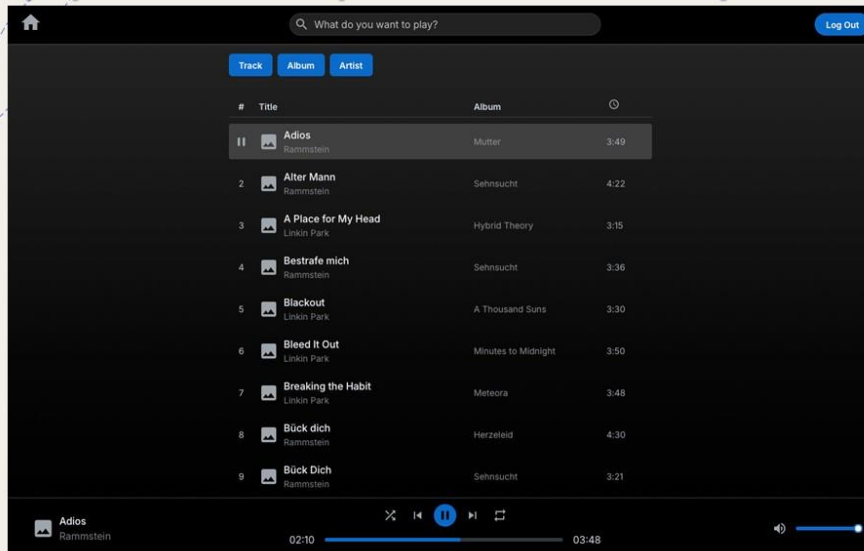
Форма авторизації у системі Soundify



Сторінка треків після входу в систему

11

Інтерфейс плеєра в активному стані



12

Висновки

В результаті виконання роботи було вирішено наступні задачі:

- проведено аналіз ринку стримінгових сервісів та визначено доцільність використання хмарних технологій;
- обґрунтовано та реалізовано мікросервісну архітектуру для досягнення масштабованості та спрощення розгортання;
- створено серверну частину з використанням сучасного стеку: .NET, Docker, PostgreSQL, RabbitMQ;
- реалізовано клієнтський застосунок на React із масштабованою архітектурою та сучасним UI;
- побудовано повноцінну хмарну інфраструктуру для зберігання, обробки та трансляції медіаконтенту.