

РОЗРОБКА JAVA-ВЕБЗАСТОСУНКУ ДЛЯ
ПЕРЕНЕСЕННЯ ПЛЕЙЛИСТІВ
КОРИСТУВАЧІВ МІЖ ПЛАТФОРМАМИ
SPOTIFY І YOUTUBE ТА СТВОРЕННЯ
КРОСПЛАТФОРМНИХ РЕКОМЕНДАЦІЙ

Виконав:

Кузнець Ілля, БП ІПЗ-4

Науковий керівник:

Борозенний Сергій Олександрович

Мета розробки та актуальність теми

Мета розробки: розробити Java-вебзастосунок, який вирішуватиме проблеми перенесення плейлистів користувачів між платформами Spotify і YouTube та створення кросплатформних рекомендацій між ними.

Актуальність теми:

- Якщо користувач хоче перейти з однієї музичної платформи на іншу, як перенести дані?
- Якщо користувач не хоче переходити на іншу музичну платформу, але хоче використовувати рекомендаційну систему тієї платформи, як це зробити?

Відомі розв'язки задачі

Перенесення плейлистів:

- Soundiiz
- Tune My Music
- Free Your Music

Кросплатформні рекомендації:

- Не було знайдено



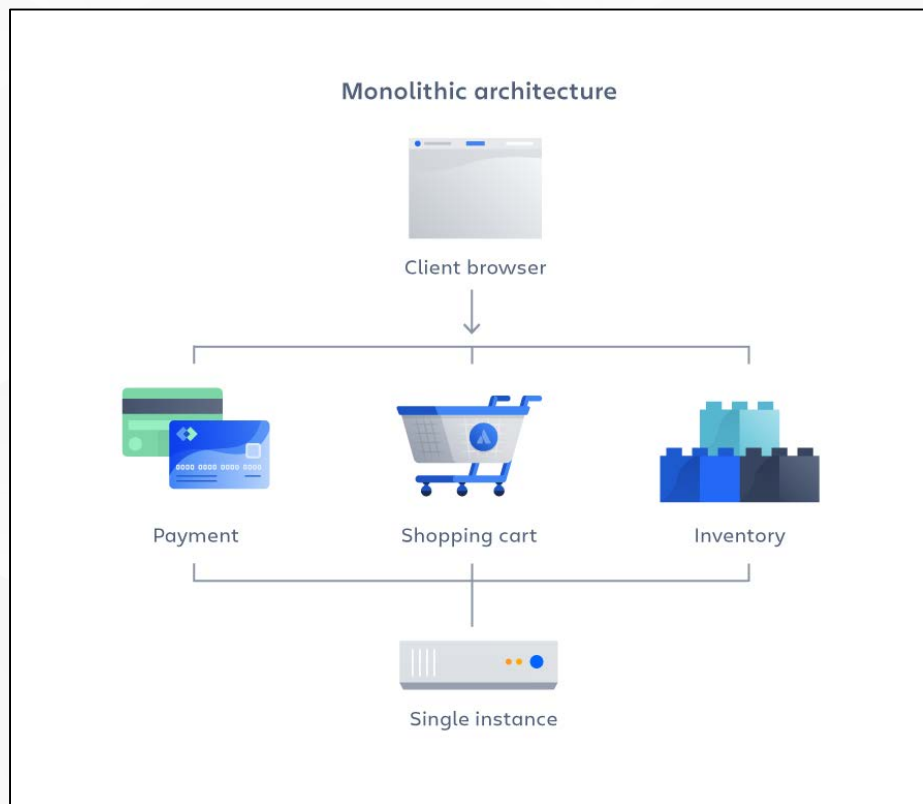


Використані технології та засоби

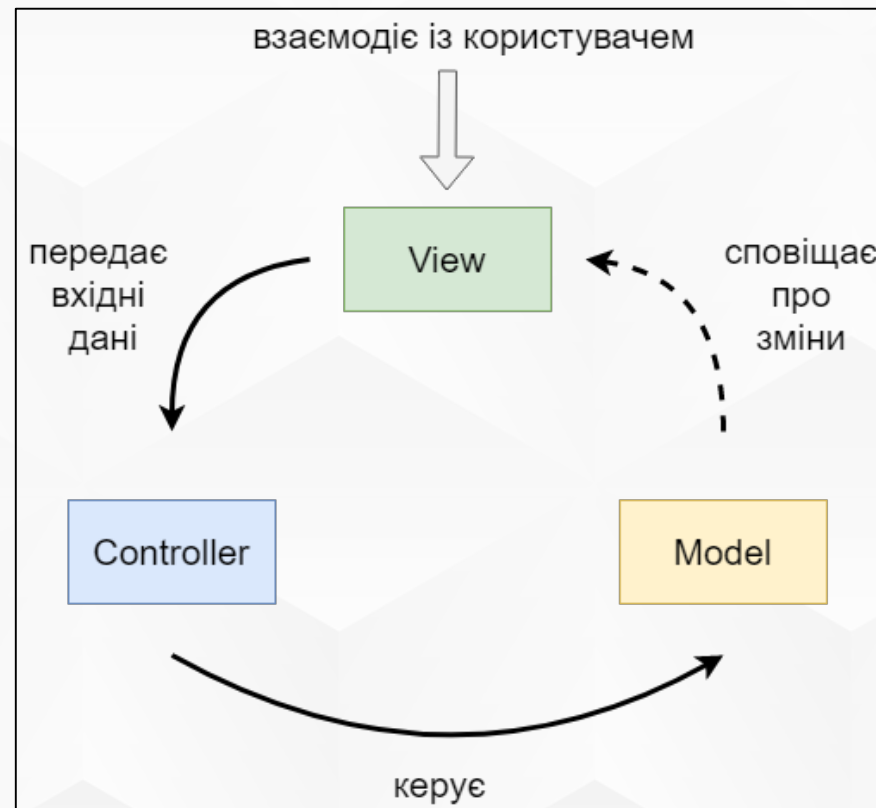
- Java
- Gradle
- Spring Boot (Spring Security, Web, Data JPA)
- Thymeleaf
- MySQL
- Spring Cloud OpenFeign
- OAuth2
- YouTube Data API, Spotify Web API



Архітектура застосунку



Монолітна архітектура



Архітектурний підхід MVC
(Model-View-Controller)

Реалізація застосунку : Автентифікація, Авторизація

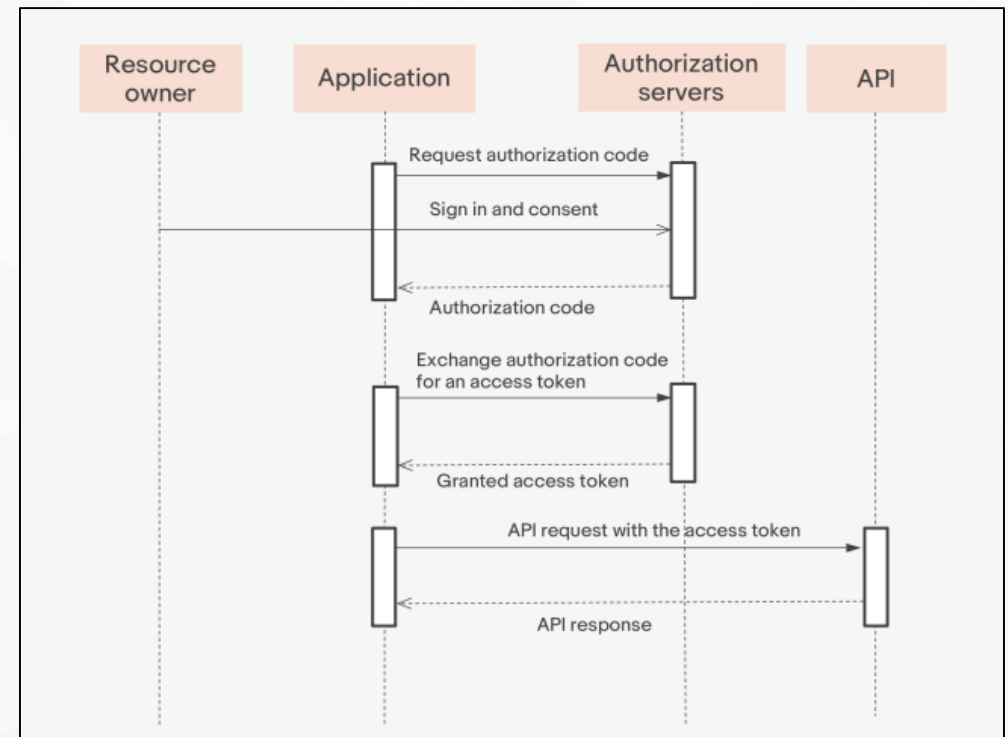
- OAuth 2.0 для Spring Boot (`spring-boot-starter-oauth2-client`)

```
spring.security.oauth2.client.registration.spotify.client-id=${spotify.client-id}
spring.security.oauth2.client.registration.spotify.client-secret=${spotify.client-secret}
spring.security.oauth2.client.registration.spotify.scope[0]=user-library-read
spring.security.oauth2.client.registration.spotify.scope[1]=playlist-read-private
spring.security.oauth2.client.registration.spotify.scope[2]=user-read-private
spring.security.oauth2.client.registration.spotify.scope[3]=user-read-email
spring.security.oauth2.client.registration.spotify.scope[4]=playlist-modify-private
spring.security.oauth2.client.registration.spotify.scope[5]=playlist-modify-public
spring.security.oauth2.client.registration.spotify.client-name=Spotify
spring.security.oauth2.client.registration.spotify.authorization-grant-type=authorization_code
spring.security.oauth2.client.registration.spotify.redirect-uri={baseUrl}/login/oauth2/code/{registrationId}
spring.security.oauth2.client.registration.spotify.provider=spotify

spring.security.oauth2.client.provider.spotify.authorization-uri=https://accounts.spotify.com/authorize
spring.security.oauth2.client.provider.spotify.token-uri=https://accounts.spotify.com/api/token
spring.security.oauth2.client.provider.spotify.user-info-uri=https://api.spotify.com/v1/me
spring.security.oauth2.client.provider.spotify.user-name-attribute=display_name

spring.security.oauth2.client.registration.google.client-id=${google.client-id}
spring.security.oauth2.client.registration.google.client-secret=${google.client-secret}
spring.security.oauth2.client.registration.google.redirect-uri={baseUrl}/login/oauth2/code/{registrationId}
spring.security.oauth2.client.registration.google.scope=profile,email,https://www.googleapis.com/auth/youtube
```

Налаштування OAuth2 для Spring Boot



Authorization code flow

Реалізація застосунку : Сутності

Сутності у базі даних:

- **User** – сутність користувача
- **UserAuthentication** – сутність, що містить дані користувача із провайдера даних (музичного сервісу)
- **UserToken** – сутність, що містить токени доступу до провайдера
- **Playlist** – сутність плейлиста

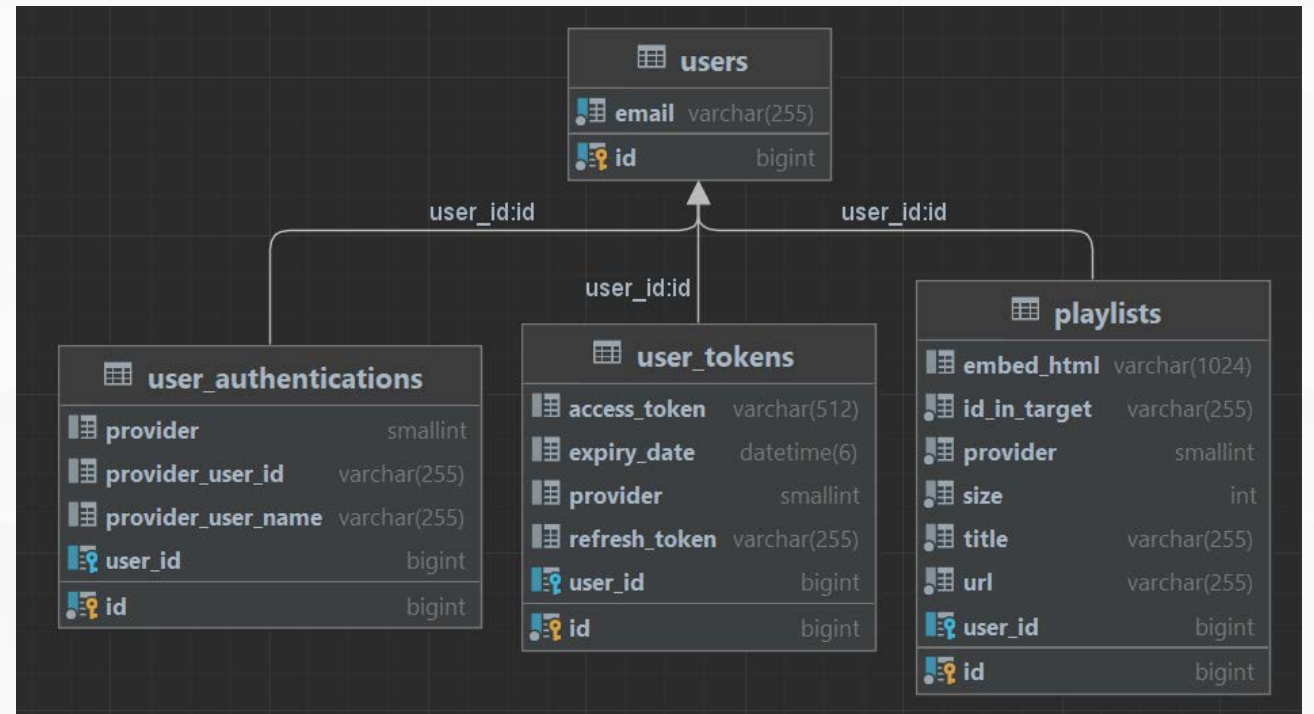
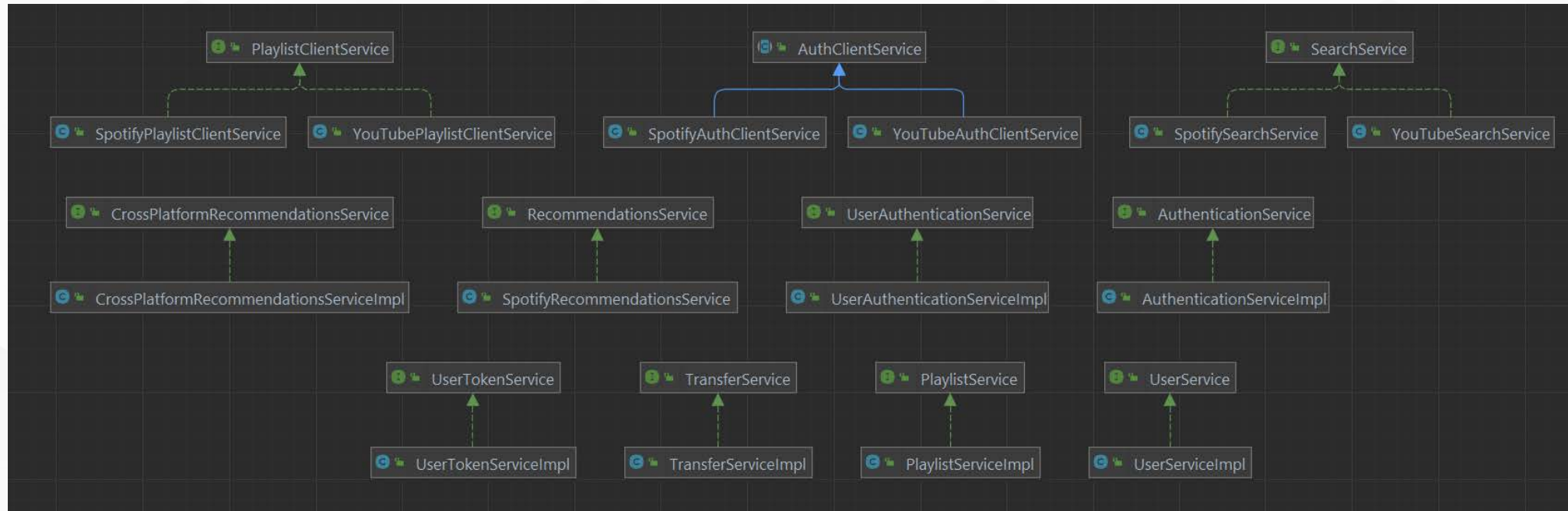


Схема реляцій

Реалізація застосунку : Сервіси

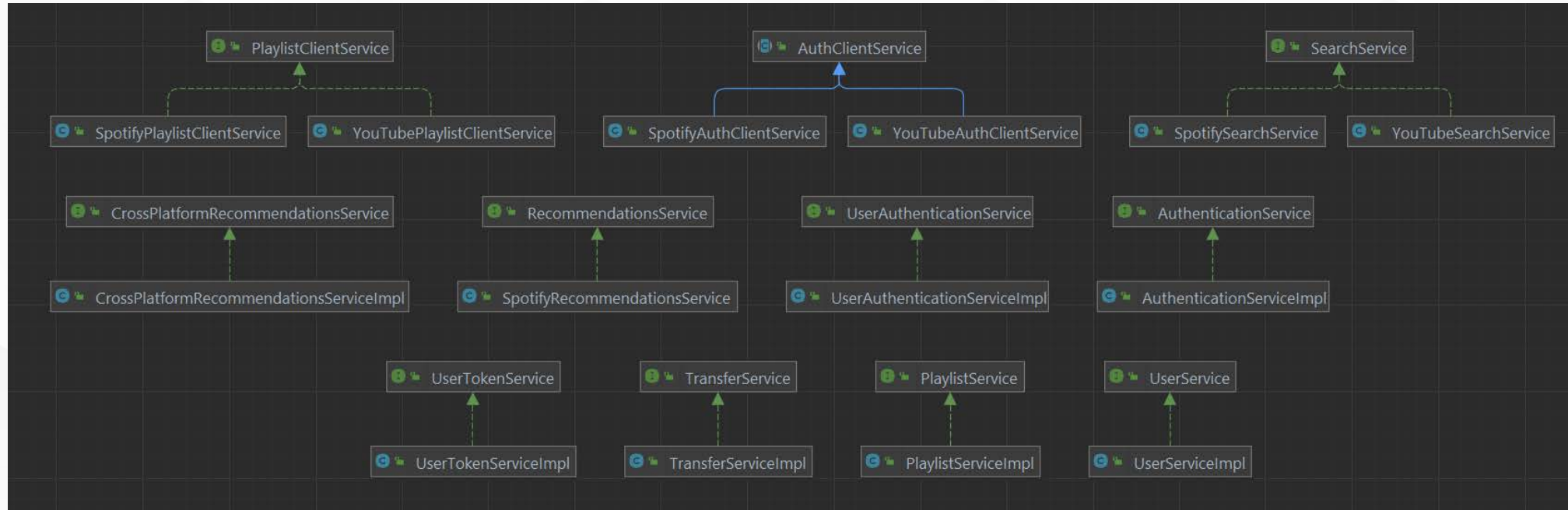


Діаграма інтерфейсів та класів сервісів

Загальні сервіси:

- **AuthenticationService** – сервіс для знаходження нинішнього автентифікованого користувача
- **TransferService** – сервіс для перенесення плейлистів з одного музичного сервісу на інший
- **CrossPlatformRecommendationsService** – сервіс для створення схожих плейлистів із використанням кросплатформних рекомендацій

Реалізація застосунку : Сервіси



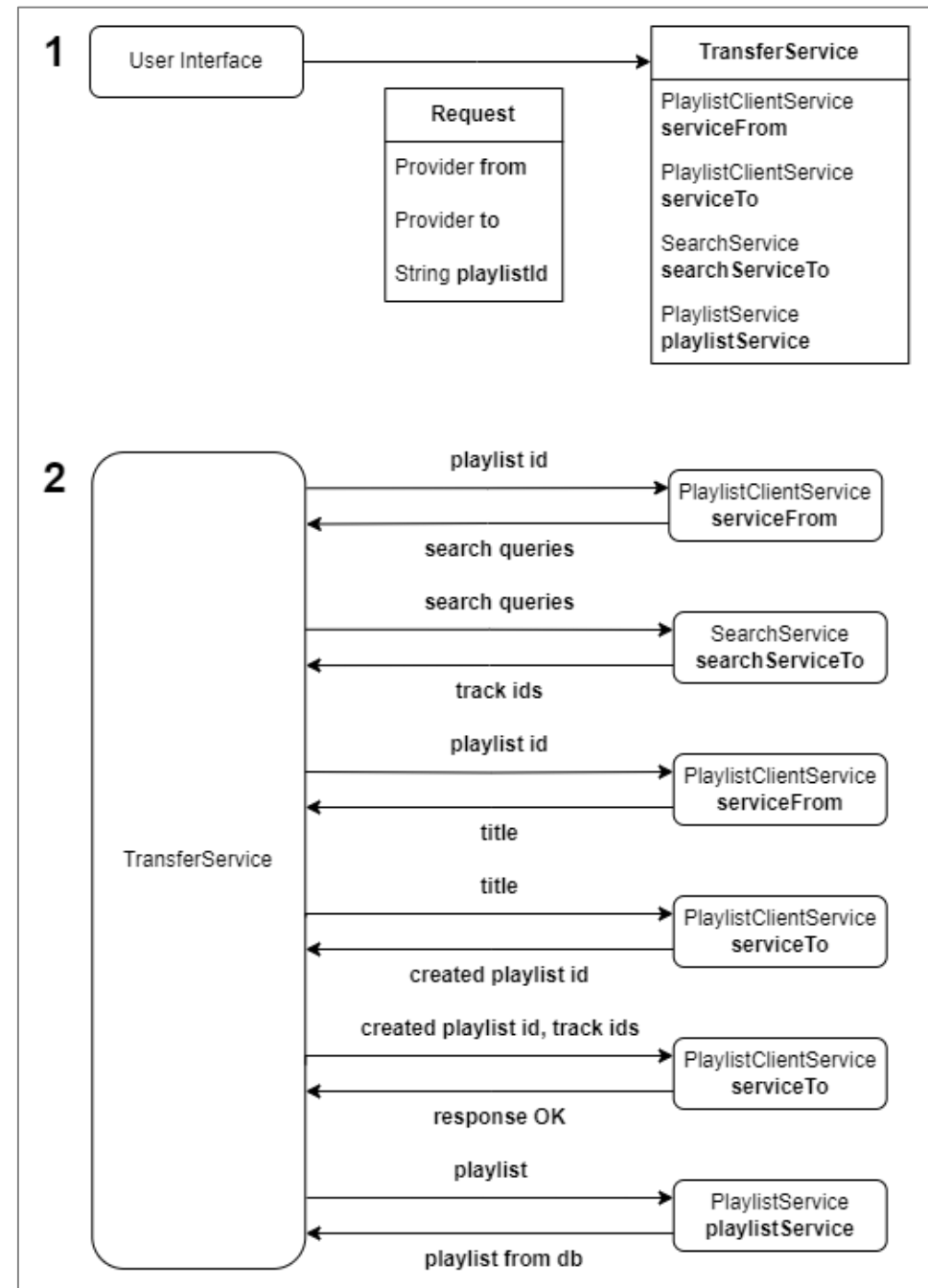
Діаграма інтерфейсів та класів сервісів

Окремі сервіси (окремі для кожного провайдера):

- [PlaylistClientService](#) – сервіс, що використовує HTTP-клієнт для роботи із плейлистами у музичних сервісах
- [RecommendationService](#) – сервіс, що використовує HTTP-клієнт для знаходження рекомендацій у музичних сервісах
- [SearchService](#) – сервіс, що використовує HTTP-клієнт для здійснення пошуку у музичних сервісах

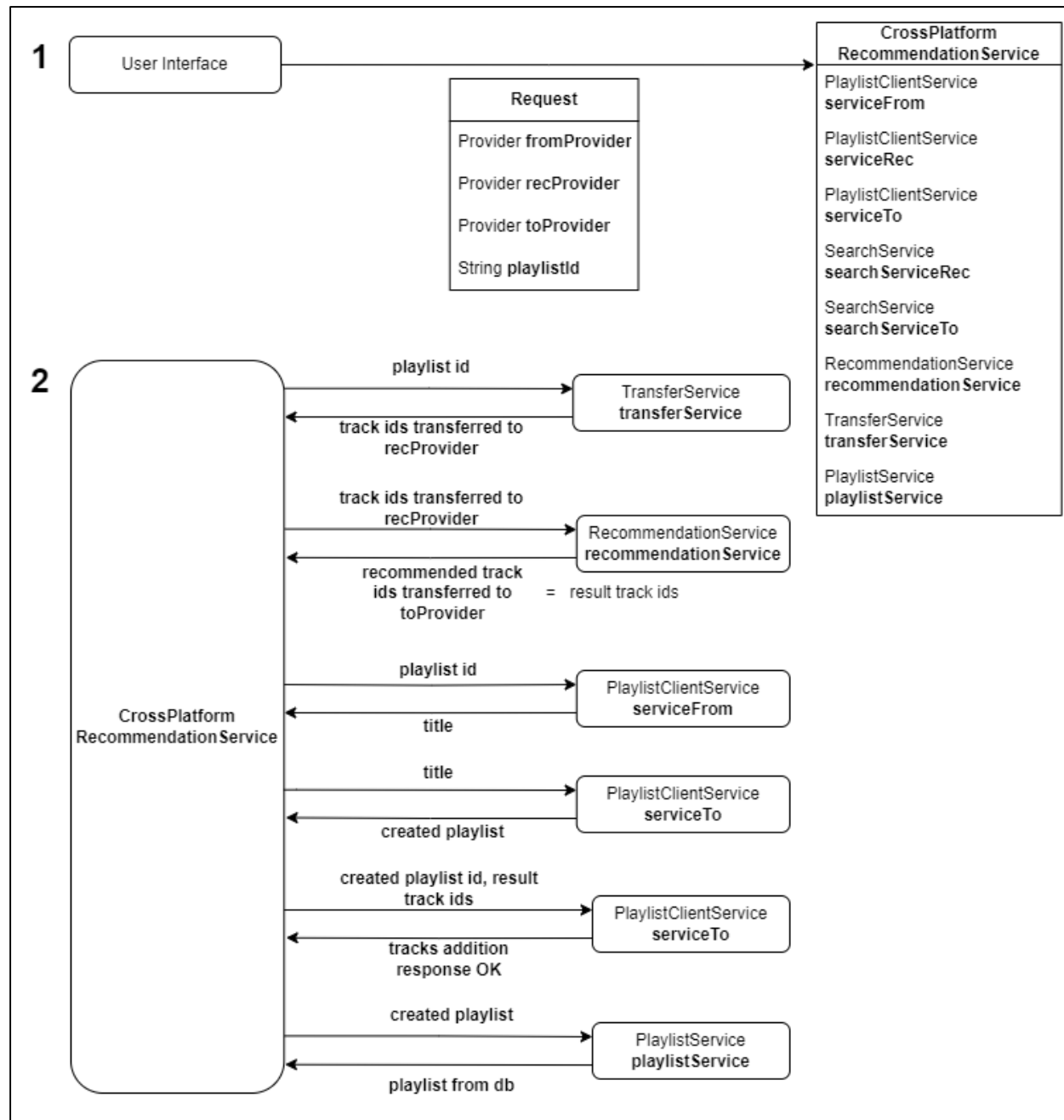
Реалізація
застосунку :

Алгоритм
перенесення
плейлистів



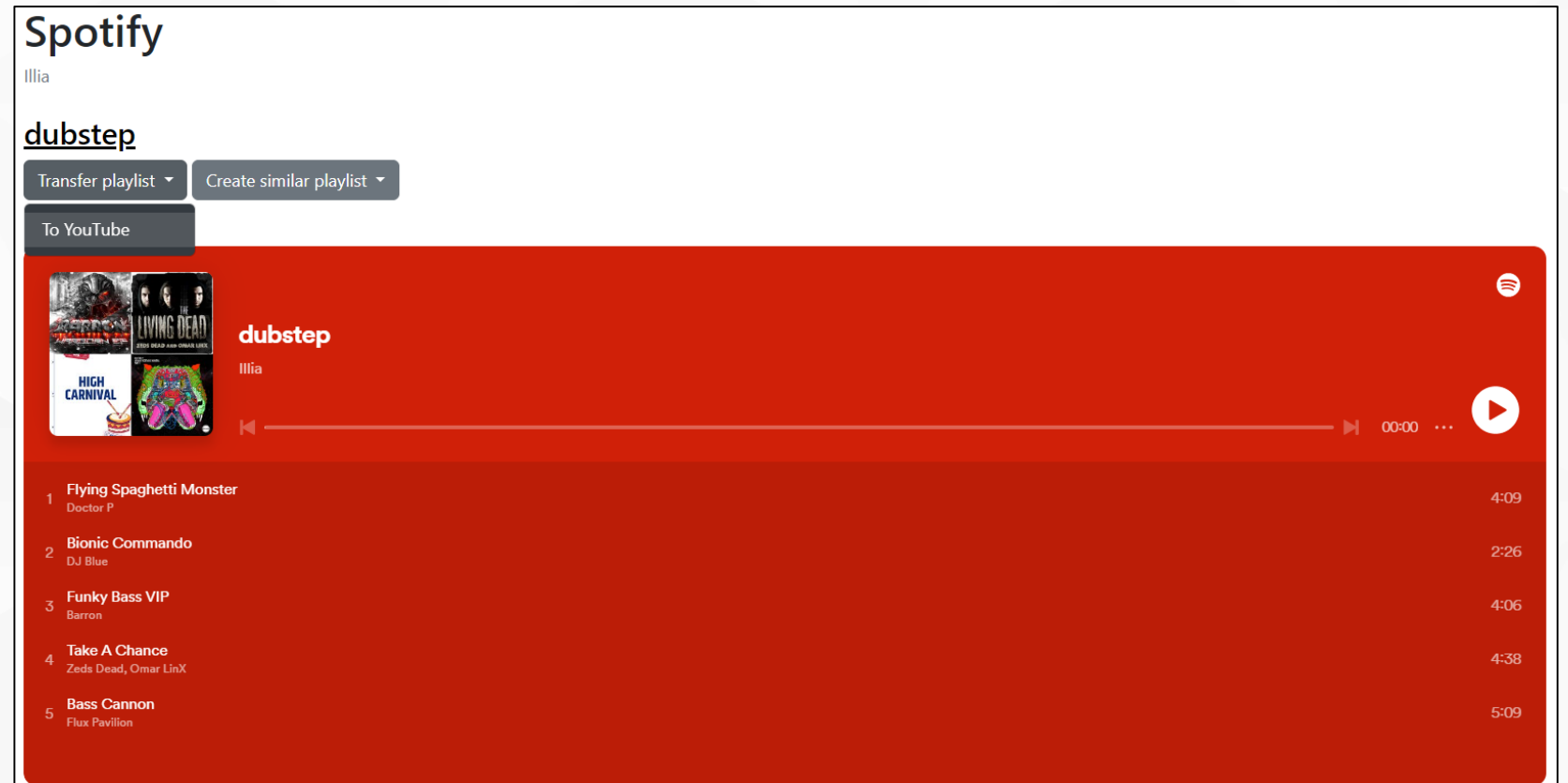
Реалізація
застосунку :

Алгоритм
створення схожих
плейлистів



Приклад застосування методу : перенесення плейлистів

Користувач має плейлист **dubstep** із п'ятьма треками на Spotify і хоче перенести його на YouTube.



Spotify

Illia

dubstep

Transfer playlist ▾ Create similar playlist ▾

To YouTube

dubstep

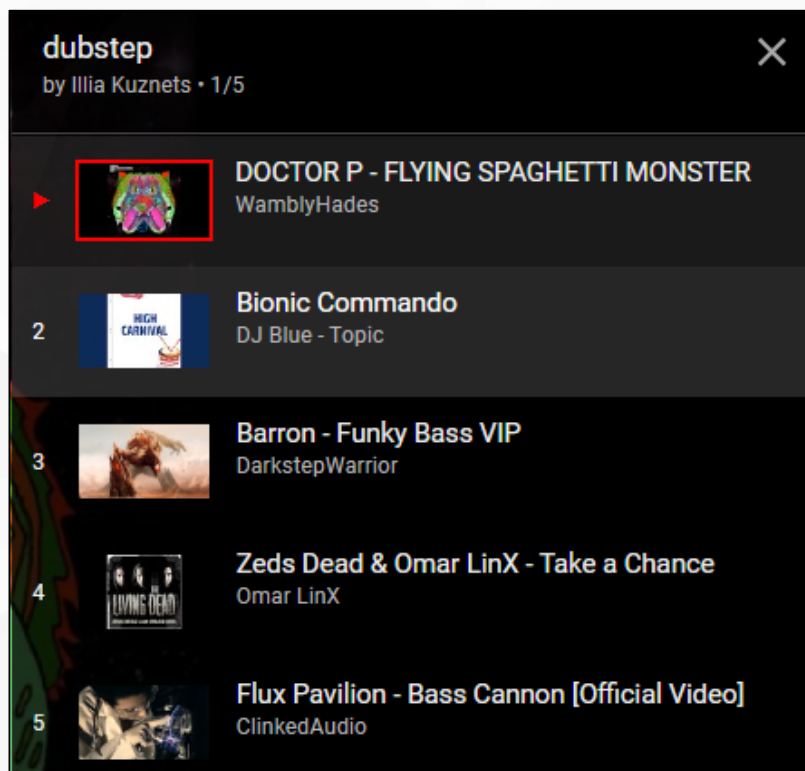
Illia

00:00 ...

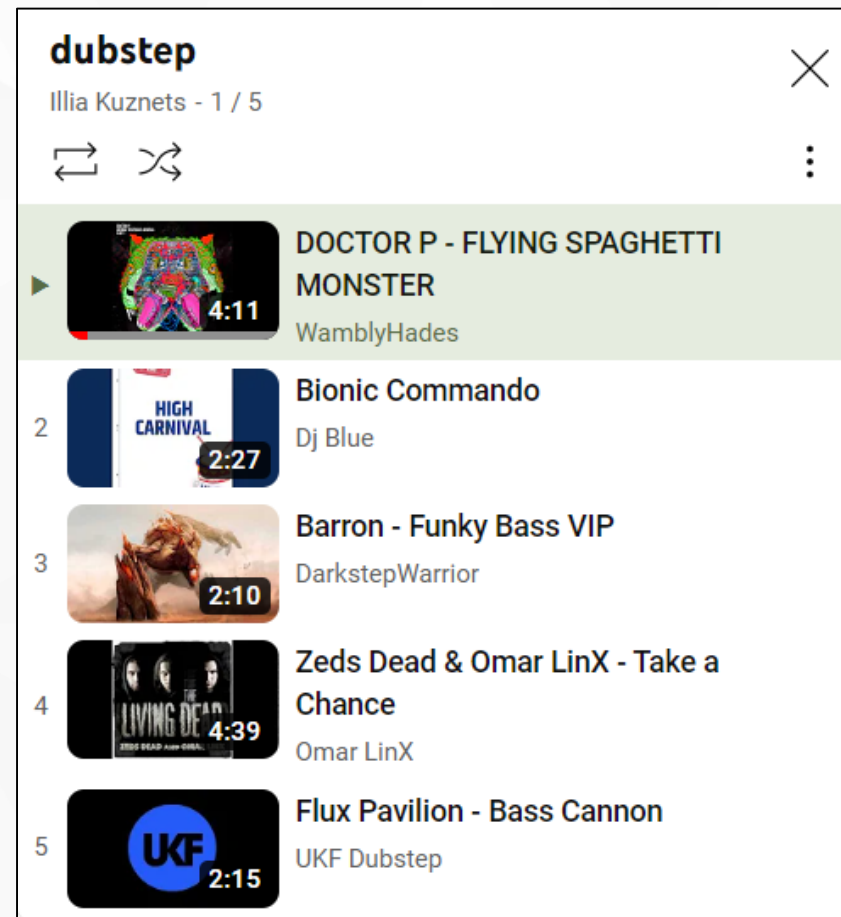
- 1 Flying Spaghetti Monster
Doctor P 4:09
- 2 Bionic Commando
DJ Blue 2:26
- 3 Funky Bass VIP
Barron 4:06
- 4 Take A Chance
Zeds Dead, Omar LinX 4:38
- 5 Bass Cannon
Flux Pavilion 5:09

Порівняння результату: перенесення плейлистів

- Порівнюючи із рішенням, що уже існує, можемо зробити висновок, що результат успішний.
- Єдина відмінність – останнє відео. Розроблений застосунок RhythmBridge повернув офіційне відео до пісні, а Soundiiz – відео, що містить статичне зображення і потрібний аудіотрек



RhythmBridge



Soundiiz

Приклад застосування методу : кросплатформні рекомендації

Користувач має плейлист *summer* із п'ятьма треками на YouTube і хоче створити схожий на нього плейлист на YouTube, проте використовуючи рекомендації Spotify.

YouTube

Illia Kuznets

summer

Transfer playlist ▾ Create similar playlist ▾

On YouTube

On Spotify

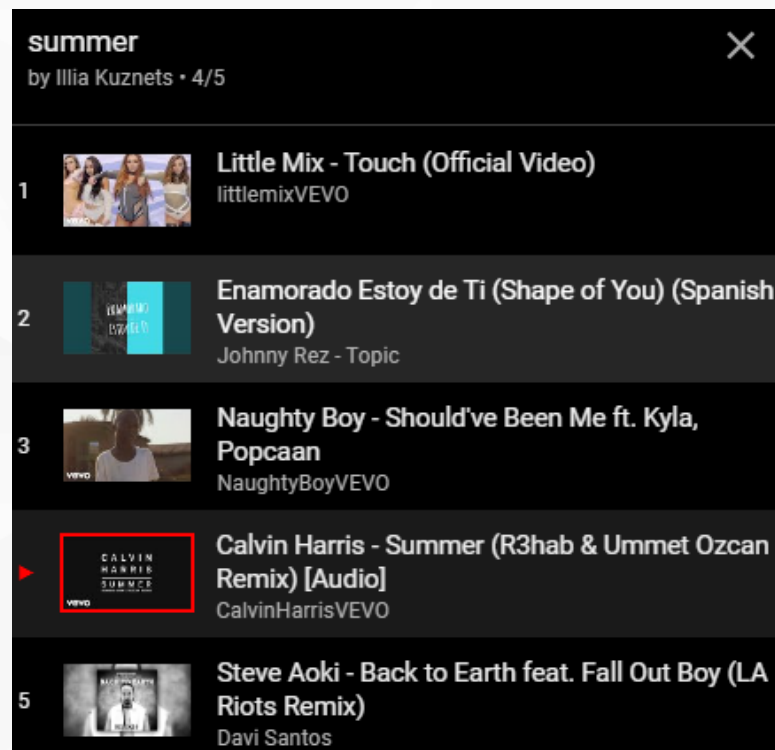
Watch on YouTube

summer
by Illia Kuznets • 1/5

- Luis Fonsi - Despacito ft. Daddy Yankee
LuisFonsiVEVO
- Ed Sheeran - Shape of You (Official Music Video)
Ed Sheeran
- Calvin Harris - My Way (Official Video)
CalvinHarrisVEVO
- Shakira - Waka Waka (This Time For Africa) (Official HD Video) ft. Freshlyground
shakiraVEVO
- Steve Aoki, NERVO & Tony Junior - Lightning Strikes (Official Video)
Ultra Records

Порівняння результату: кросплатформні рекомендації

- Оглянуті найбільш відомі сервіси для перенесення музики між платформами не дозволяють використовувати міжплатформні рекомендації (створення схожих плейлистів)



RhythmBridge



Soundiiz,
Tune My Music,
Free Your Music

Результат роботи

У результаті виконання роботи мовою Java був розроблений вебзастосунок RhythmBridge, який використовує API Spotify та YouTube для того, щоб дозволяти користувачам переглядати свої плейлисти, переносити їх між зазначеними платформами та створювати схожі плейлисти, використовуючи кросплатформні рекомендації.

Практична значимість розробки

Розроблений застосунок буде дуже корисним для користувачів музичних стрімінгових платформ, оскільки він вирішує такі проблеми:

- Проблема перенесення плейлистів для користувачів, що бажають змінити платформу
- Проблема створення кросплатформних рекомендацій для користувачів, що не бажають змінювати платформу, але хочуть отримувати кращі рекомендації



Рекомендації для подальшої розробки

- Покращення точності пошуку треків-відповідників
- Покращення користувацького інтерфейсу
- Збільшення функціоналу (наприклад постійна синхронізація плейлистів між сервісами)
- Додавання підтримки більшої кількості музичних сервісів
- При значному збільшенні застосунку – міграція на іншу архітектуру
- Вирішення проблем із квотами для запитів на сторонні API



Дякую за увагу!