소프트웨어공학 2팀 아키텍처 설계

****

**강의:**소프트웨어공학

**담당교수:** 허훈식

**팀명:** 2팀

**팀장:** 정한준

**팀원:**김광현, 민정준, 최이설

**제 출 일:** 2024.05.23

**목 차**

**I. 팀 소개**

1. 팀 소개

**II. 시스템 개요**

1. 시스템 소개 및 개요

**III. 시스템 요구사항**

1. 설계 목적

2. 전체 기능

3. 최우선 기능

4. 제약 사항

5. 품질 속성

6. 유스케이스 다이어그램

7. 유스케이스 서술

**IV. 아키텍처 설계 구조 생성 및 뷰**

1. 모듈 뷰

2. C&C 뷰

3. 할당 뷰

4. 뷰 매핑

**V. 근거(공통 설계 결정)**

1. 근거(공통 설계 결정)

**VI. 디렉토리(용어)**

1.디렉토리(용어)

**1. 팀 소개**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **팀 구성 보고서** | | | | | |
| **일자** | 2024. 4. 28. | **팀 리더** | | 정한준 | |
| **팀 구성**  **목적** | 소프트웨어 아키텍처 설계 팀 프로젝트 | | | | |
| **팀 운영**  **기간** | 2024. 04. 28. ~   2024. 05. 28. | | | | |
| **팀 이름** | 2팀 | | | | |
| **팀 구성원** | **학과** | | **학번** | | **이름** |
| 컴퓨터공학부 | | 2019150013 | | 정한준 |
| 컴퓨터공학부 | | 2020152004 | | 김광현 |
| 컴퓨터공학부 | | 2020156015 | | 민정준 |
| 컴퓨터공학부 | | 2021134043 | | 최이설 |

|  |  |
| --- | --- |
| **팀원 역할 분담** | |
| **팀명** | **2팀** |
| **이름** | **주요 역할** |
| **정한준**  **(팀장)** | - 시스템 개요 정리  - 유스케이스 다이어그램 작업  - 뷰 할당 작업, 디렉토리, 설계 결정 작성  - 보고서 작성 |
| 김광현 | - 모듈 뷰 작업  - 전체 기능, 최우선 기능 정리  - 보고서 작성  - 발표 |
| **최이설** | - 시스템 개요, 설계 목적 작업  - 컴포넌트 뷰 작업  - 보고서 작성  - PPT 작성 |
| **민정준** | - 할당 뷰 작업  - 제약사항, 품질속성 정리  - 보고서 작성  - 발표 |

**시스템명: 음악 스트리밍 서비스**

**2. 시스템 개요**

- 음악 스트리밍 서비스의 여러 기능 중 개인별 음악 추천 서비스에 초점을 두고 시스템을 설계하게 되었다.

- 이 시스템은 실시간 음악 스트리밍 서비스를 제공하며, 회원들에게 음악 추천, 음악 공유, 음악 검색 기능을 제공할 수 있도록 설계된다.

- 네트워크 연결이 가능한 환경이면 어떤 장소나 시간이어도 원하는 음악을 스트리밍 가능해야 한다.

- 웹, 안드로이드, ios 등 다양한 환경에서도 불편 없이 사용이 가능해야 한다.

**3. 시스템 요구사항**

1) **설계 목적**

개인 별 음악 추천 기능이 메인 서비스인 실시간 음악 스트리밍 시스템이다.

음악 검색, 공유, 회원관리 기능을 제공하며 사용자와 관리자 모두 편안하게 이용할 수 있는 시스템 설계가 목적이다.

2) 전체 기능

|  |  |  |
| --- | --- | --- |
| 기능 | | 설명 |
| UC-1 | 로그인 | 사용자를 회원, 비회원, 관리자로 구분하게 하는 기능이다. |
| UC-2 | 회원가입 | 회원이 될 수 있게 해주는 기능이다. |
| UC-3 | 음악 검색 | 사용자가 원하는 음악을 찾을 수 있게 해주는 기능이다. |
| UC-4 | 음악 재생 | 회원이 음악을 선택하여 재생할 수 있는 기능이다. |
| UC-5 | 음악 공유 | 회원이 음악 정보를 공유하는 기능이다. |
| UC-6 | 1분 미리 듣기 | 비회원인 사용자가 원하는 음악을 1분만 듣게 하는 기능이다. |
| UC-7 | 결제 | 회원이 개별 음악, 구독권을 결제하는 기능이다. |
| UC-8 | 서버 관리 | 관리자가 원활한 서버 작동을 보장하는 기능이다. |
| UC-9 | 사용자 관리 | 관리자가 사용자의 계정, 관련 기능을 관리하는 기능이다. |
| UC-10 | 콘텐츠 관리 | 관리자가 다양한 콘텐츠를 추가, 수정, 삭제하는 기능이다. |
| UC-11 | 재생 목록 수집 | 서버가 음악 추천을 위한 사용자가 재생했던 음악 목록을 수집하는 기능이다. |
| UC-12 | 음악 추천 | 서버가 재생 목록과 원하는 장르 별로 회원에게 분석하여 맞춤 음악을 추천하는 기능이다. |
| UC-13 | 앱 업데이트 | 서버가 새로운 기능, 개선 사항, 버그 수정들을 포함한 앱의 새로운 버전을 사용자에게 제공하는 기능이다. |
| UC-14 | 글쓰기 | 회원이 음악이나 앨범에 대해 글을 쓸 수 있는 기능이다. |
| UC-15 | 데이터 공유 | 저장소에 있는 데이터를 공유하는 기능이다. |
| UC-16 | 데이터 수정 | 저장소의 데이터를 수정하는 기능이다. |
| UC-17 | 데이터 삭제 | 저장소의 데이터를 삭제하는 기능이다. |
| UC-18 | 데이터 저장 | 저장소에 데이터를 저장하는 기능이다. |

**3) 최우선 기능**

|  |  |  |
| --- | --- | --- |
| **기능** | | **설명** |
| UC-7 | 결제 | 사용자가 유료 음악, 구독권을 결제하는 기능이다. |
| UC-4 | 음악 재생 | 사용자가 음악을 선택하여 재생할 수 있는 기능이다. |
| UC-3 | 음악 검색 | 사용자가 원하는 음악을 찾는 기능이다. |
| UC-12 | 음악 추천 | 재생 목록과 현재 날씨를 분석하여 사용자에게 맞춤 음악을 추천하는 기능이다. |
| UC-15 | 데이터 공유 | 저장소에 있는 데이터를 공유하는 기능이다. |

**4) 제약 사항**

1. 대규모의 음악 데이터를 처리하기 위해 부하 분산 시스템을 적용해야 한다.
2. 스트리밍을 위해 네트워크 안정성을 유지해야 한다.
3. 웹 혹은 안드로이드, IOS 등 모든 플랫폼에서 동작해야 한다.
4. 서버가 24시간 가용 되어야 하므로 다중화 서버가 필요하다.
5. 음악 저작권, 개인정보 보호 등 법적 요구사항 준수해야 한다.

**5) 품질 속성**

가용성.

1. 음악 재생, 검색, 플레이리스트 등 모든 기능은 항상 24시간 사용 가능해야 한다.
2. 음악 카탈로그의 업데이트나 음악 정보의 변경은 실시간으로 서버에 반영되어야 한다.

신뢰성.

1. 서버에서 제공되는 데이터는 항상 일관성이 유지되어야 한다.
2. 새로운 음악이나 가사, 앨범 아트 등의 업데이트는 즉시 사용자에게 반영되어야 한다.
3. 사용자가 음악 플레이리스트를 수정하거나 새로운 음악을 추가할 때, 해당 변경 사항은 즉시 반영되어야 한다.

성능.

1. 대규모 트래픽을 처리할 수 있는 성능 최적화가 필요하다.
2. 사용자의 요청에 빠르게 응답하여 음악 스트리밍 속도를 보장해야 한다.

보안성.

1. 사용자 개인정보와 결제 정보는 안전하게 암호화되어 저장되어야 한다.
2. 사용자의 개인정보는 해당 사용자만 접근할 수 있어야 하며, 외부에 노출되면 안 된다.

사용성.

1. PC, 모바일 기기 및 기타 플랫폼에서 모두 접근할 수 있는 사용자 친화적인 UI를 제공해야 한다.

각각의 플랫폼에 최적화된 사용자 경험을 제공해야 한다.

**6) 유스케이스 다이어그램**

그림, 도표, 스케치, 종이접기이(가) 표시된 사진

자동 생성된 설명

**7) 유스케이스 서술**

-18가지의 기능을 유스케이스로 표현하였다.

-액터는 사용자,관리자,서버로 구분했으며, 특히 유스케이스 중 로그인을 해야만 제공되는 기능들을 로그인 유스케이스에 포함 관계로 묶고, 음악 재생시 사용자가 재생한 목록을 수집하고 해당 목록을 기반으로 음악을 추천하여야 함으로 확장관계로 포함하였다.

음악 검색과 앱 업데이트는 고객과 서버 액터 모두가 관여하고, 데이터 공유, 데이터 수정, 데이터 삭제, 데이터 저장은 관리자와 서버 액터 모두가 관여한다.

**4. 아키텍처 설계 구조 생성 및 뷰**

**1.모듈 뷰**

1) 기본 프레젠테이션

스크린샷, 도표, 텍스트, 평면도이(가) 표시된 사진

자동 생성된 설명

2) 요소 카탈로그

(1) 요소와 속성

|  |  |  |
| --- | --- | --- |
| **티어** | **모듈** | **설명** |
| 클라이언트 | 클라이언트-서버 통신 | 음악 애플리케이션과 서버 간의 데이터 교환을 관리하는 역할을 한다. 클라이언트가 서버에 음악 데이터를 요청하거나 서버에 업로드하는 등의 작업을 수행한다. 클라이언트-서버 통신을 통해 음악 데이터의 신뢰성과 안정성을 보장하고, 사용자가 실시간으로 음악 데이터를 업데이트할 수 있도록 한다. |
| 음악 선택 및 재생 | 음악 목록을 표시하고 사용자가 음악을 선택할 수 있는 인터페이스를 제공한다. 선택한 음악을 재생하고, 재생 상태를 표시하며 재생 제어 기능을 통해 사용자가 음악을 찾고 재생한다. |
| 음악 검색 | 사용자가 원하는 음악을 키워드나 조건에 맞게 검색할 수 있는 기능을 제공한다. 음악의 제목, 아티스트, 앨범 등의 정보를 기반으로 검색 결과를 제공하고 사용자에게 편리한 검색 옵션을 통해 사용자가 다양한 음악을 쉽게 찾을 수 있도록 도와줍니다. |
| 결제 | 음악 애플리케이션 내에서 유료 음악을 구매하거나 구독 서비스에 대한 결제를 처리한다. 결제 수단 선택, 결제 정보의 입력, 결제 상태의 확인 등을 관리하여 사용자가 원하는 음악에 대한 액세스를 제공한다. |
| SNS 공유 | 사용자가 음악을 SNS 플랫폼에 공유한다. 음악 재생 중에 SNS에 공유하기 버튼을 누르면 해당 음악 정보와 함께 SNS에 게시할 수 있는 인터페이스를 제공한다. 사용자는 음악을 지인들과 공유하고 추천할 수 있다. |
| 재생 목록 관리 | 자신의 음악 재생 목록을 생성하고 관리할 수 있다. 사용자가 원하는 음악을 플레이리스트에 추가하거나 제거할 수 있으며, 재생 목록의 순서를 변경하거나 자동 재생 기능을 설정할 수 있다. |
| 음악 파일 관리 | 애플리케이션 내에서 음악 파일을 로드하고 관리한다. 음악 파일을 저장하거나 음악 데이터베이스에서 음악 파일을 검색할 수 있다. 또한, 음악 파일의 메타데이터를 추출하여 음악 정보를 표시하고 정리할 수 있다. |
| 앱 서버 | 정보 연동 | 클라이언트에서 취합한 데이터를 비즈니스 레이어의 결제 처리와 연동한다. 음악 차트 데이터, 아티스트 정보, 음반 정보 등을 외부 소스로부터 연동하여 데이터를 요청하고 응답을 처리하고 사용자에게 필요한 정보를 제공한다. |
| 사용자 재생 목록 분석 | 사용자의 음악 재생 기록을 분석하여 음악 추천을 하는 기능을 수행한다. 사용자의 음악 장르, 최근 재생한 음악 등을 분석하여 사용자에게 적합한 음악을 추천한다. |
| 결제 처리 | 제 요청을 처리하고 결제 서비스와의 연동을 담당한다. 음악을 구매하거나 구독 서비스에 가입할 때 필요한 결제 정보를 처리하고, 결제 상태를 확인하며, 결제 프로세스를 관리한다. |
| 저장소 액세스 | 취합한 정보를 저장소로 전달한다. 사용자 정보, 결제 정보 등의 데이터를 저장하고 음악 메타 데이터, 음악 파일 등을 공유한다. |
| 공통(인증, 보안, 로깅) | 로그인을 통해 관리자/사용자(회원, 비회원)를 구분한다. |
| 저장소 | 저장소 | 사용자의 접속 시간, 음악 메타데이터, 음악 파일, 3D 영상 파일, 결제 정보 등이 기록된다. |

2) 관계와 속성

|  |  |  |
| --- | --- | --- |
| **관계 속성** | | **설명** |
| 클라이언트 | 데이터, 작업 요청 | 클라이언트는 앱 서버에 데이터, 작업 요청을 한다. |
| 앱 서버 | 데이터, 작업 응답 | 클라이언트, 외부 시스템, 저장소에서 수신한 데이터를 통합적으로 분석, 처리하고 클라이언트로 응답한다. |
| 저장소 | 데이터 송/수신 & 저장 | 앱 서버에서 처리한 정보를 수신해서 저장한다. 앱 서버에 데이터를 송신한다.. |

3) 근거 (설계 결정)

|  |  |
| --- | --- |
| **티어** | **근거** |
| 클라이언트 | 클라이언트 티어는 사용자에게 직관적이고 편리한 인터페이스를 제공하여 음악 선택, 재생, 플레이리스트 관리 등을 수행할 수 있도록 한다. 사용자의 편의성과 사용성을 고려하여 애플리케이션을 직접 실행하며 앱 서버로 데이터, 작업을 요청한다. |
| 앱 서버 | 앱 서버 티어는 데이터 관리, 비즈니스 로직 처리, 보안과 인증, 외부 시스템과의 통합 등을 통해 클라이언트의 데이터, 작업 요청을 응답한다. |
| 저장소 | 저장소 티어는 앱 서버에서 수신한 데이터를 저장하고 관리한다. 또한 앱 서버 티어 데이터를 수신하는 역할을 한다. |

**2.컴포넌트 커넥터 뷰**

1)기본 프레젠테이션

도표, 평면도, 기술 도면, 라인이(가) 표시된 사진

자동 생성된 설명

1)요소 카탈로그

(1)요소와 속성

|  |  |
| --- | --- |
| **요소** | **설명** |
| Client | Client 컴포넌트는 음악 선택 및 재생, 결제, 음악 검색, SNS 공유, 재생 목록 관리, 음악 파일 관리 서비스, client-server 통신을 제공하는 역할을 한다. Music recommend algorithm 컴포넌트를 통해 음악 추천 작업을 하고, payment service 컴포넌트를 통해 결제 작업을 수행한다. |
| Music Recommend Algorithm | Music Recommend Algorithm 컴포넌트는 사용자가 즐겨 듣는 음악 정보를 분석하여 음악 추천 작업을 하고, 그 결과를 client 컴포넌트로 송신하는 역할을 한다. |
| Payment Service | Payment Service 컴포넌트는 유료 음원 구매, 정기 구독 등의 Client의 작업 요청을  처리하는 역할을 한다. 또한 작업 결과를 Repository Service 컴포넌트로 송신하는 역할을 한다. |
| Repository Service | Repository Service 컴포넌트는 사용자의 결제 정보를Payment Service 컴포넌트로부터 수신, VR 영상 파일을 VR Video Editing SW 컴포넌트로부터 수신하는 역할을 한다. 또한 음악 메타데이터, 음악 파일을 Music Recommend Algorithm 컴포넌트로 송신, VR 영상 파일을 VR Application 컴포넌트로 송신하는 역할을 한다. |

(1)관계와 속성

|  |  |
| --- | --- |
| **관계** | **설명** |
|  | Client 컴포넌트에서 작업을 Music Recommend Algorithm 컴포넌트와 Payment Service 컴포넌트로 작업을 요청한다.  Music Recommend Algorithm 컴포넌트는 Client가 요청한 작업(음악 추천)을 수행하고 응답한다.  Payment Service 컴포넌트는 Client가 요청한 작업(결제)을 수행하고 응답한다. |
|  | Music Recommend Algorithm 컴포넌트는 사용자의 음악 재생 목록 또는 날씨 정보를 분석하여 맞춤 음악에 대한 정보를 Repository Service 컴포넌트로부터 수신한다.  Payment Service 컴포넌트는 유료 음악, 정기권 등의 결제 정보를 처리하고, 결과를 Repository Service 컴포넌트로 수신한다.  Repository Service 컴포넌트는 수신한 데이터를 저장하고 요청 받은 데이터를 송신한다. |

**3.할당 뷰**

1) 기본 프리젠테이션

도표, 직사각형, 평면도, 스크린샷이(가) 표시된 사진

자동 생성된 설명

2) 요소와 관계

(1) 요소와 속성

|  |  |
| --- | --- |
| **요소** | **속성** |
| Client(PC) | Web Browser를 통해 UI를 제공하고 유저 생성 및 로그인, 음악을 검색, 재생, 결제, 공유 기능 이용 |
| Client(Smartphone) | Application를 통해 UI를 제공하고 유저 생성 및 로그인, 음악을 검색, 재생, 결제, 공유 기능 이용 |
| Server(EC2) | Amazon사 에서 제공하는 AWS EC2를 이용하여 Spring Boot 서버를 구동해 유저 관리, 결제 작업, 음악 데이터 관리 및 작업을 수행 |
| Payment | 신용카드 회사로부터 결제 시스템 제공 |
| Database(RDS) | Amazon사 에서 제공하는 AWS RDS를 이용하여 MySQL을 구동해 유저 정보, 결제 정보, 음악 메타 데이터를 저장 |
| Music  Recommendation  Algorithm | Amazon사 에서 제공하는 AWS Lambda를 사용하여 음악 추천 기능 제공 |

(2) 관계와 속성

|  |  |  |
| --- | --- | --- |
| **관계** |  | **속성** |
| PC | Windows10 | 개인용 컴퓨터에 windows10을 할당 |
| Smartphone | Android, ios | Samsung Smartphone에 Android 할당  iphone에 ios 할당 |
| EC2 | Ubuntu | EC2서버에 Ubuntu 22.04 LTS 할당 |
| RDS | MySQL | RDS서버에 MySQL 할당 |
| Lambda | Music Recommendation Algorithm | 음악 추천 알고리즘을 RDS에 할당 |
| Client | EC2 | Client와 Server 간 HTTPS 통신을 통해 데이터의 기밀성과 무결성 유지 |
| EC2 | RDS | EC2와 RDS 간 TCP/IP 통신을 통해 유저 정보와 결제 정보, 음악 재생 목록 등을 저장 |
| EC2 | Lambda | HTTPS(API Gateway) 통신을 통해 사용자에게 맞춤 음악 추천 |

(3) 근거

|  |  |
| --- | --- |
| **설계 결정** | **근거** |
| Windows10 | PC 운영체제 중 가장 보편적으로 사용하고 친숙하기 때문에 결정 |
| Android, ios | Smartphone 운영체제 중 가장 보편적으로 사용하기 때문에 결정 |
| EC2 | 클라우드 컴퓨팅 서버로 하드웨어 비용이 절감되고 빠르게 개발 및 배포가 가능하기 때문에 결정 |
| Ubuntu 22.04 LTS | Windows에 비해 간편하고 빠르며 보안성도 높기 때문에 결정 |
| RDS | 높은 확장성과 관리 용이성으로 결정 |
| MySQL | 데이터 관리의 효율성과 안정성 및 보안성을 제공하기 때문에 결정 |
| Lambda | 서버관리에 부담을 줄이면서 안정적으로 서비스를 제공하기 때문에 결정 |
| HTTPS | 데이터의 기밀성과 무결성을 보장하며 안전하게 데이터를 전송하기 때문에 결정 |
| TCP/IP | 유저 정보, 결제 정보, 음악 데이터 등을 저장 과정 중 손실되거나 훼손되면 안되기 때문에 결정 |

**4. 뷰 매핑**

**- 모듈 뷰/C&C 뷰 매핑**

**텍스트, 도표, 평면도, 라인이(가) 표시된 사진

자동 생성된 설명**

|  |  |
| --- | --- |
| **컴포넌트** | **모듈** |
| **Client** | 클라이언트 서버 통신, 음악 선택 및 재생, 음악 검색, 결제, SNS공유  재생 목록 관리, 음악 파일 관리 |
| **Music Recommend Algorithm** | 음악 선택 및 재생, 음악 검색 |
| **Payment service** | 정보 연동, 결제 처리, 저장소 액세스 |
| **Repository Service** | 저장소 |

**- C&C 뷰/할당 뷰 매핑**

**도표, 텍스트, 평면도, 원이(가) 표시된 사진

자동 생성된 설명**

|  |  |
| --- | --- |
| **할당** | **컴포넌트** |
| **PC,SmartPhone** | Client |
| **EC2** | Music Recommend Algorithm, Payment service |
| **RDS** | Repository Service |

**5. 근거(공통 설계 결정)**

**1. 근거(공통 설계 결정)**

|  |  |
| --- | --- |
| **설계 결정** | **근거** |
| **레이어(Layered)**  **스타일** | 다뤄야 할 모듈과 기능이 많고 지속적 업데이트를 필요로 하는  서비스이기 때문에 복잡성 관리와 변경용이성 측면에서 유리한  레이어 스타일로 모듈 뷰를 구축하였다. |
| **저장소**  **스타일** | 음악 스트리밍 서비스 특성 상 고객 디바이스, 메인 서버,  그리고 데이터 센터까지 수많은 지속성 데이터를 공유하며  수정해야 하므로 공유 데이터 관리에 유리한 저장소 스타일로  C&C 뷰를 구축하였다. |
| **배포(Deploy)**  **스타일** | 음악 스트리밍 서비스는 고객이 주로 사용하는 접속 환경의 정보와  시스템 구축 시 필요한 서버 장비의 비용 산정이 중요하다.  때문에 하드웨어 투자 비용과 고객의 예상 환경을 분석할 수 있고  마찬가지로 중요한 신뢰성과 보안까지 프로토콜로 점검할 수 있는  배포 스타일로 할당 뷰를 구축하였다. |

**6. 디렉토리(용어)**

**1. 디렉토리(용어)**

|  |  |
| --- | --- |
| **용어** | **설명** |
| **구독** | 매월 일정 요금을 지불하면 혜택을 제공받는 유료 서비스 |
| **HTTPS** | 인터넷에서 사용하는 웹 서버와 클라이언트의 인터넷 브라우저 사이에  문자를 전송하기 위한 통신 규약  보안 측면에서 장점 |
| **Lambda** | 서버를 프로비저닝하거나 관리하지 않고도 코드를 실행할 수 있게 해주는 컴퓨팅 서비스 |
| **MySQL 8** | Oracle 사의 오픈 소스 DBMS  상업적 이용 목적으로도 무료 이용 가능 |
| **TCP/IP** | 인터넷에서 컴퓨터들이 서로 정보를 주고받는데 쓰이는 통신규약  TCP와 IP를 기반으로 이루어져 있으며 송수신 과정에서  정보의 손실이 발생하지 않아 신뢰성에 이점 |
| **Ubuntu 22.04 LTS** | 데비안 기반의 OS  고유한 데스크톱 환경을 사용하는 리눅스 배포판  자유 소프트웨어를 기반 |