

## <요구사항 분석 보고서>

2016024893 오성준

### 1. 요구사항

- Entity: 음원 관리자, 스트리밍 사용자, 음원, 플레이리스트, 앨범, 아티스트
- 서비스 요구사항
  - 음원 관리자는 음원을 관리하여 등록 및 삭제가 가능하다.
  - 음원 관리자는 사용자 신상을 관리할 수 있다.
  - 사용자는 여러 플레이리스트를 만들 수 있으며, 각 음원들을 본인만의 플레이리스트에 등록할 수 있다.
  - 서비스는 각 음원이 플레이 된 횟수를 파악하여 각 음원의 통계 등을 제공할 수 있다.

### 2. 요구사항 분석 (Entity & Attribute)

Entity	Attribute
음원 관리자	관리자 index, 관리자 권한
사용자	이름, 전화번호, 이메일, 주민번호, 나만의 플레이리스트(플레이리스트), 사용 서비스 index, 사용자 권한
음원	제목, 다운로드 횟수(날짜별로/달 별로), 스트리밍 횟수(날짜별로/ 달 별로), 수록된 앨범, 등록 날짜, 저작권자(아티스트 와 그 외 등등), 음원 파일, 음원 총 수입<달마다 갱신>
앨범	제목, 저작권자(아티스트), 수록 음원(음원 배열), 등록 날짜, 앨범 종류 index
아티스트	이름, 발매한 앨범(앨범 배열), 발매한 음원(음원 배열)
플레이리스트	이름, 플레이리스트 등록자(사용자), 수록 음원(음원 배열)

### 3. 세부 사항 설명

- 음원 관리자
  - 음원 관리자는 관리자의 권한을 가지고 있어서, 음원과 앨범 등의 삭제와 등록을 할 수 있습니다. 이 때 사용자는 사용자 권한만을 가져서 음원과 앨범의 삭제와 등록을 할 수 없습니다. 다만 사용자는 플레이리스트를 등록할 수는 있습니다.

- 사용자

- 사용자의 개인정보를 저장합니다. (이름, 전화번호, 이메일, 주민번호)
- 사용자는 관리자와 똑같이 음원 DB에 접근할 수 있지만, 사용자 권한이기 때문에 등록과 삭제를 하지 못합니다. 다만 그 음원들로 플레이리스트를 만들어서 본인만의 플레이리스트에 추가할 수 있으며, 선택에 따라서 전체 플레이리스트 DB에 들어가기도 합니다.
- 사용 서비스 index는 사용자가 사용하고 있는 서비스의 종류로 이 서비스 index에 따라서 지불하는 비용과 서비스의 종류가 달라집니다.
- 여기서 key역할은 사용자의 이름으로 해서 B+ tree를 구성합니다.

- 음원

- 음원의 기본적인 정보들을 저장합니다. (제목, 수록된 앨범, 등록 날짜, 저작권자, 음원파일)
- 스트리밍 횟수와 다운로드 횟수를 저장해서 이를 통해서 통계를 낼 수 있게 만듭니다. 기본적으로 날짜별로 저장하며, 이를 달마다 갱신합니다. 달마다 이들 통계를 통해서 음원 총수입을 계산해서 저작권자에게 정산합니다. 이 때, 갱신 시 갱신된 달의 총 수입은 따로 기록해서 향후 체크할 수 있도록 합니다. (가능하다면, 연도별로도 기록해서 체크할 수 있도록 하고 싶습니다.)
- 기본적으로 DB는 이름을 키로 선정해서 관리됩니다. 하지만, 통계를 내야하는 때에는 전체의 음원을 스트리밍이나 다운로드 횟수를 키로 선정해서 정렬시킵니다. (향후 수정 및 추가할 수 있음, 향후 장르 index를 추가해서 key를 만드는 것을 고려 중입니다.)

- 앨범

- 앨범의 기본적인 정보들을 저장합니다. (제목, 저작권자, 수록 음원)
- 수록 음원 같은 경우, 많아야 200곡을 넘기지 않으므로 단순히 배열에 저장할 예정입니다.
- 앨범 종류 index는 앨범의 종류를 나타내는 index입니다. 큰 의미는 없을 예정입니다.

- 아티스트

- 아티스트는 본인의 이름과 동시에 본인이 발매한 앨범과 음원들을 등록합니다.
- ⇒ 앨범과 아티스트라는 entity를 만든 것은 향후에 사용자가 원하는 음원 검색 방식에 따라

서, 이러한 entity가 필요할지도 모른다는 생각에서 만들었습니다. 만약에 어떤 노래를 만든 작곡가가 만든 다른 노래를 찾고 싶을 수도 있지 않겠습니까? 이렇듯 음원을 검색하는 편리한 interface를 제공하기 위해서 필요할지도 모른다고 생각이 들었습니다. 또한 통계를 낼 때에도, 이러한 entity를 이용한 통계가 필요할지도 모른다고 생각이 들었습니다. 아티스트별로 스트리밍 기록 등을 확인하고 싶을 수도 있으니까요. 향후 과제를 해가면서 현실적인 실행가능성을 보면서 기능을 추가, 수정, 삭제하겠습니다.

- 플레이리스트

- 플레이리스트는 그 이름을 가지고, 플레이리스트를 등록한 사용자의 정보를 가집니다. 또한 이 플레이리스트에 담긴 음원의 정보가 담기는데, 이 때 수록가능한 음원의 개수는 제한을 두어서 어느 정도 이상 음원을 담지 못하도록 만들 예정입니다. (그 한계선을 정하지는 않았습니다. 향후에 코드를 짜면서 정확한 수치를 정할 예정입니다. 최대한 많은 노래를 담을 수 있도록 만들 예정입니다.)