

요구분석서

과제명: Mp3 Manager 요구 분석서

작성일: 2013/ 9/ 25

작성팀: BBC

팀원	기여도(%)	기여내용
한 길	40	다이어그램 작성 및 분석서 편집
조 우석	35	일정 및 조정안 계획
박 민우	25	분석서 편집

목 차

1. 제품 소개	3
2. 제품 기능 목록.....	3
2.1. 주요 기능	3
2.2. 부가 기능	3
3. 기본 일정 및 조정안	3
3.1. 기본 일정	3
3.2. 조정안1.....	4
3.3. 조정안2.....	4
3.4. 조정안3.....	4
4. Use Case Diagram	4
4.1. Use Case Diagram.....	4
4.2. Use Case 명세: DB구성.....	5
4.2.1. 개요	5
4.2.2. 주요 액터.....	5
4.2.3. 선행 조건.....	5
4.2.4. 후행 조건.....	5
4.2.5. 시나리오	6
4.3. Use Case 명세: 재생목록 생성	6
5. UI 프로토타입	8

1. 제품 소개

Mp3파일을 쉽게 관리하고, 사용자의 기호에 맞춘 재생목록 생성

2. 제품 기능 목록

2.1. 주요 기능

탐색한 파일을 앨범, 아티스트, 등으로 구분하여 DB를 구성한다.

보다 다양한 조건의 재생목록을 생성한다.

2.2. 부가 기능

구성된 DB를 다른 시스템에서도 불러오기가 가능하도록 한다.

3. 기본 일정 및 조정안

3.1. 기본 일정

9/23~9/25	분석서 작성
9/23~10/09	1차버전코딩
10/09~10/10	데모버전 발표자료 작성 및 준비
10/22~10/25	설계서작성 및 발표자료준비
10/11~11/19	2차버전코딩
11/20~11/21	주요메소드 테스트
11/21~11/28	최종코딩
11/29~12/4	발표자료 및 데모코드준비

3.2. 조정안1

1차ver: 데이터베이스연동을 미루고 자바프로그램만 구현

2차ver: 재생목록변경을 초점에 두고 우선구현

test: test버전구현이 늦어질 경우 주요기능을 바탕으로 테스트 후 추가기능구현

3.3. 조정안2

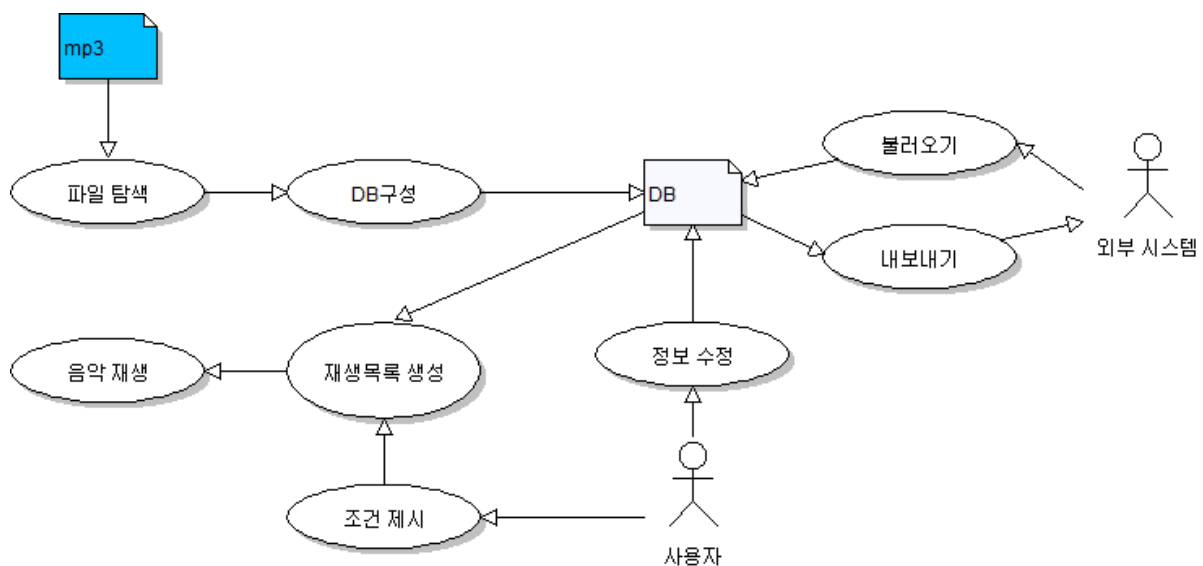
11/22일까지주요메소드미구현시테스트기간을줄이고주요메소드구현

3.4. 조정안3

11/29일까지최종코드미구현시추가기능은포기하고주요기능정상작동테스팅

4. Use Case Diagram

4.1. Use Case Diagram



4.2. Use Case 명세: DB구성



4.2.1. 개요

- 탐색된 파일의 속성을 애트리뷰트의 값으로 가지는 튜플을 삽입한다.
- 튜플이 삽입될 때 애트리뷰트의 디폴트 값은 파일의 속성과 같은 값이지만, 튜플의 이름과 재생시간을 제외한 애트리뷰트의 값은 실제 mp3파일의 속성과 다를 수 있다.
- 유저는 개인의 기호에 맞춰 특정 애트리뷰트 값들을 수정하여 데이터들을 정리할 수 있다.
- 애트리뷰트값들은 재생목록을 생성할 때 제시된 조건의 지침으로 사용된다.
- 곡명과 재생시간을 조합하여 키로 사용된다.

4.2.2. 주요 액터

유저

4.2.3. 선행 조건

- 목록으로 구성할 파일들은 모두 곡명 또는 재생시간이 각기 다르다.

4.2.4. 후행 조건

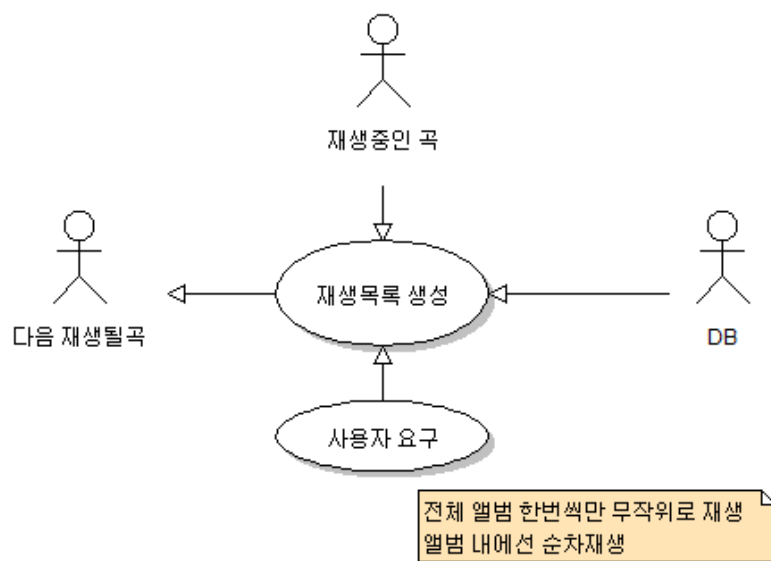
- 튜플로 저장된 후에 실제파일에선 각기 다른 그룹(앨범, 아티스트)으로 구분되어 있으나, 사용자 정의에 의해 DB에선 같은 그룹으로 묶일 수 있다.
- 반대로 실제파일에선 같은 그룹이었으나, DB에선 다른 그룹으로 구분될 수 있다.

4.2.5. 시나리오

기본 시나리오:

1. 싱글 앨범으로 나온 mp3파일을 가지고 있었는데 싱글로 나온 곡이 포함된 정규 앨범이 나와 기존에 가지고 있던 노래를 뺀 정규 앨범 곡들도 목록에 추가한다.
2. 기존에 가지고 있던 싱글발매곡과 새로 추가한 정규발매곡간의 속성이 달라서 관리가 불편하다.
3. 또는 다른 구매처에서 구한 음원파일의 속성이 달라서 관리하기가 불편하다.
4. 사용자는 DB 목록에 다른 속성값을 가진 음원파일들을 저장할 때 같은 그룹으로 묶고 싶을 경우 저장될 애트리뷰트값을 수정하여 연관이 있는 곡들인 것처럼 저장할 수 있다.

4.3. Use Case 명세: 재생목록 생성



4.3.1. 개요

- 사용자의 요구와 현재 재생 중인 곡을 대조하여 다음 재생될 곡의 조건을 충족하는 곡을 DB에서 검색한다
- 기존 재생 방법과 비교될 수 있는 점: 기존 앨범재생에선 단지 재생 범위를 재생중인 곡

과 같은 앨범인 곡들로 한정한다. 새로 고안할 재생방법은 앨범 단위로 재생하는 것이다.
기존의 앨범재생을 원한다면 현재 앨범만 재생 같은 요구를 주면 된다



- 앨범 뿐만 아니라 아티스트와 같은 그룹을 기준으로

4.3.2. 주요 액터

유저

4.3.3. 선행 조건

- 재생이 되었던 곡인지 구분이 가능해야 한다.

4.3.4. 후행 조건

- 재생이 끝나면 재생이 되었던 곡으로 표시를 해준다.

4.3.5. 시나리오

기본시나리오

1. 사용자가 아티스트 별 전체 아티스트, 무작위 순서로 한번씩 재생, 그룹 내에선 순차재생 요구 제시한다.

2. 현재 재생중인 곡과 같은 아티스트인 곡들을 검색하고, 그 중 재생되지 않았던 곡들을 내림차순으로 재생한다.
3. 해당 아티스트의 모든 곡들이 재생이 끝나면 나머지 아티스트 중 무작위 아티스트의 곡들을 내림차순으로 재생한다.
4. 모든 아티스트들의 곡을 한번씩 재생할 때까지 반복한다.

5. UI 프로토타입

