


캡스톤 디자인 I 종합설계 프로젝트

프로젝트 명	멜벅지
팀 명	윤하 - 26 많관부
문서 제목	결과보고서

Version	1.5
Date	2022-MAY-24

팀원	이인호 (조장)
	박태범
	서범석
	서필립

 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24


CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 및 소프트웨어학부 개설 교과목 다학제간캡스톤디자인I 수강 학생 중 프로젝트 "멜벅지"를 수행하는 팀 "윤하-26 많관부"의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 "윤하-26 많관부"의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역


Filename	수행결과보고서-멜벅지.doc
원안작성자	이인호
수정작업자	서필립, 박태범, 서범석

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2022-05-21	이인호	1.0	최초 작성	
2022-05-21	서필립	1.1	내용 수정	개요,목표 수정
2022-05-22	박태범	1.2	내용 수정	개요, 목표, 연구/개발내용, 시스템 기능 요구사항, 활용/개발기술, 결과물 목록 수정
2022-05-23	이인호	1.3	내용 수정	구조 및 설계도 수정
2022-05-23	박태범	1.4	내용 수정	제한요소, 참고문헌, 테스트케이스 수정
2022-05-24	서범석	1.5	내용 추가	화면구성, 개발기술 추가

 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

목 차

1	개요	4
1.1	프로젝트 개요	4
1.2	추진 배경 및 필요성	4
1.2.1	음원 사재기	4
1.2.2	순위 집계 방식	5
2	개발 내용 및 결과물	6
2.1	목표	6
2.2	연구/개발 내용 및 결과물	6
2.2.1	연구/개발 내용	6
2.2.2	시스템 기능 요구사항	9
2.2.3	시스템 비기능(품질) 요구사항	11
2.2.4	시스템 구조 및 설계도	11
2.2.5	활용/개발된 기술	12
2.2.6	현실적 제한 요소 및 그 해결 방안	13
2.2.7	결과물 목록	14
2.3	기대효과 및 활용방안	14
3	자기평가	15
4	참고 문헌	15
5	부록	15
5.1	사용자 매뉴얼	15
5.2	운영자 매뉴얼	16
5.3	배포 가이드	16
5.4	기술문서	16
5.5	테스트 케이스	17

 <div> 국민대학교 소프트웨어학부 다학제간캡스톤디자인I </div>	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

1 개요

1.1 프로젝트 개요

최근 음원 사이트들이 다양하게 늘어나고 있는 상황에서 사이트마다 각기 다른 음원 순위를 가지고 있기 때문에 음악 트렌드를 한 눈에 파악하기가 어렵다.

본 프로젝트는 멜론, 벅스, 지니 3사의 통합차트 제공을 목표로 한다. 임의의 가중치를 설정한 타 유사 서비스들과는 다르게 '사이트 별 좋아요의 합'을 가중치로 두어 고정된 값이 아닌 유동적으로 변화하는 값을 통해 통합차트를 생성하여 기존 서비스들보다 객관적인 정보를 사용자에게 제공할 수 있다.

1.2 추진 배경 및 필요성

1.2.1 음원 사재기



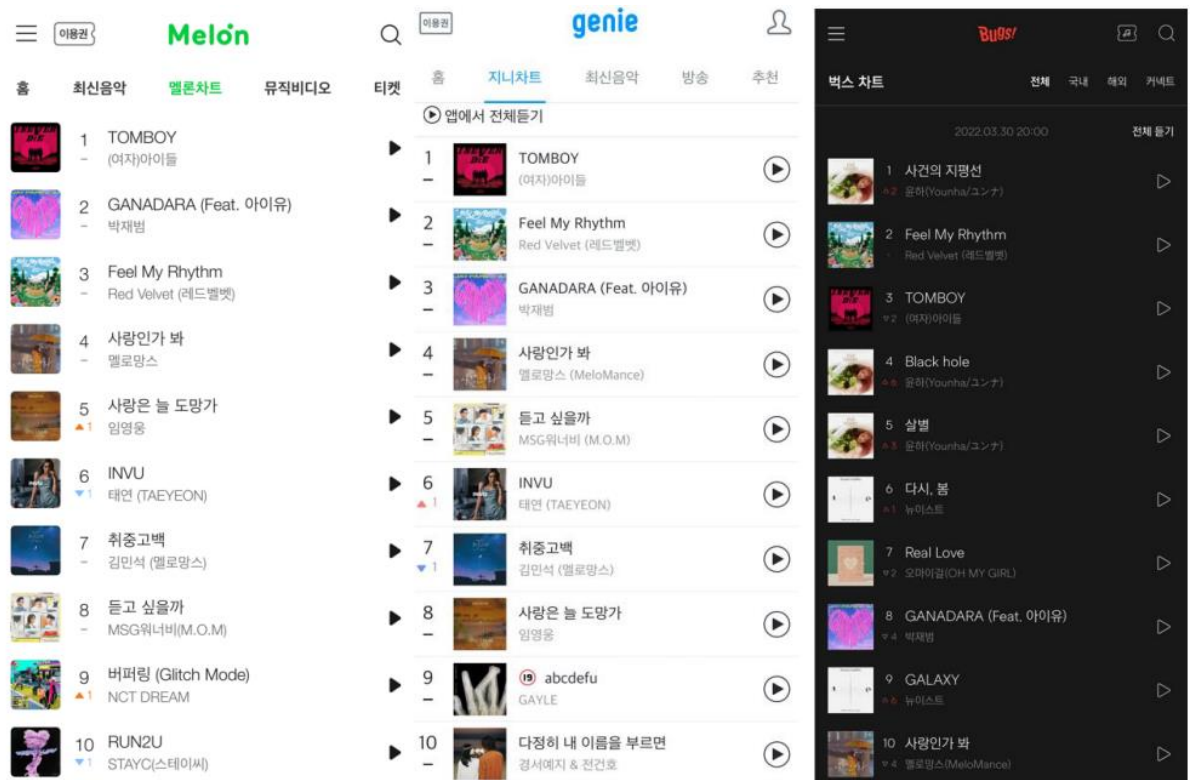
[그림 1] 사재기 관련 인터넷 기사

음원 사재기는 2010년대부터 꾸준히 논란이 되어 오다 2018~2019년에 이르러 음악계에 서 많은 증언이 나오며 대중들에게 널리 알려지게 되었다. 이렇듯 음원 사재기가 공공연 하게 이루어지고 있는 상황에서 하나의 음원 사이트 순위보다 여러 사이트의 데이터를 통 합 한 순위가 실제 사용자들의 순위와 유사할 것이다.

 <div> 국민대학교 소프트웨어학부 다학제간캡스톤디자인I </div>	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24


1.2.2 순위 집계 방식

아티스트가 음원을 발매한 지 얼마 지나지 않았을 때 특정 음원사이트의 차트에서 강세를 보이는 경향이 있다. 이는 음원사이트의 순위 집계 방식에서의 차이라고 추정되는데 평균 사용자 수가 가장 적은 벅스에서 이러한 현상이 두드러지게 나타난다. 다음은 가수 '윤하'의 앨범 발매일 당시의 실시간 차트이다.



[그림 2] 2022년 3월 30일 20시 멜론, 지니, 벅스 실시간 차트

위와 같이 멜론, 지니의 차트에서 윤하의 곡을 찾아볼 수 없는 것에 비해 벅스 차트에서는 상위권을 차지하고 있는 것을 볼 수 있다. 멜론, 지니를 이용하던 사용자는 벅스 차트를 보고 혼란이 생길 것이고, 반대로 벅스를 이용하던 사용자는 멜론, 지니를 보고 혼란이 생길 것이다. 이러한 상황에서 멜론, 지니, 벅스를 통합한 차트가 있다면 사용자의 혼란을 줄여줄 수 있을 것이다.

 국민대학교 소프트웨어학부 다학제간캡스톤디자인	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

2 개발 내용 및 결과물

2.1 목표

본 프로젝트 '멜벅지'는 국내 유명 음원사이트인 멜론, 벅스, 지니의 데이터를 분석하여 편향성을 줄인 독자적인 통합차트 제공을 목표로 한다. 이로 인해 사용자는 전체적인 음원 트렌드를 확인할 수 있을 것이다. 또한, 멜론, 벅스, 지니 3사의 데이터를 한 눈에 비교할 수 있게 제공함으로써 사용자의 객관적인 판단에 도움을 줄 것이다.

2.2 연구/개발 내용 및 결과물

2.2.1 연구/개발 내용

2.2.1.1 크롤링

- 언어 : Python
- 기술 : AWS lambda, BeautifulSoup
- 1시간마다 데이터를 크롤링하기 위해 AWS lambda를 사용하였다.
- BeautifulSoup를 사용해 멜론, 지니, 벅스의 실시간 차트를 크롤링한다.
- 크롤링 된 데이터의 좋아요 숫자를 토대로 한 가중치를 생성한다.
- 크롤링 된 데이터를 통합하고 생성된 가중치를 기반으로 한 상위 100개의 곡으로 통합 순위를 생성한다.

2.2.1.2 Database 구축

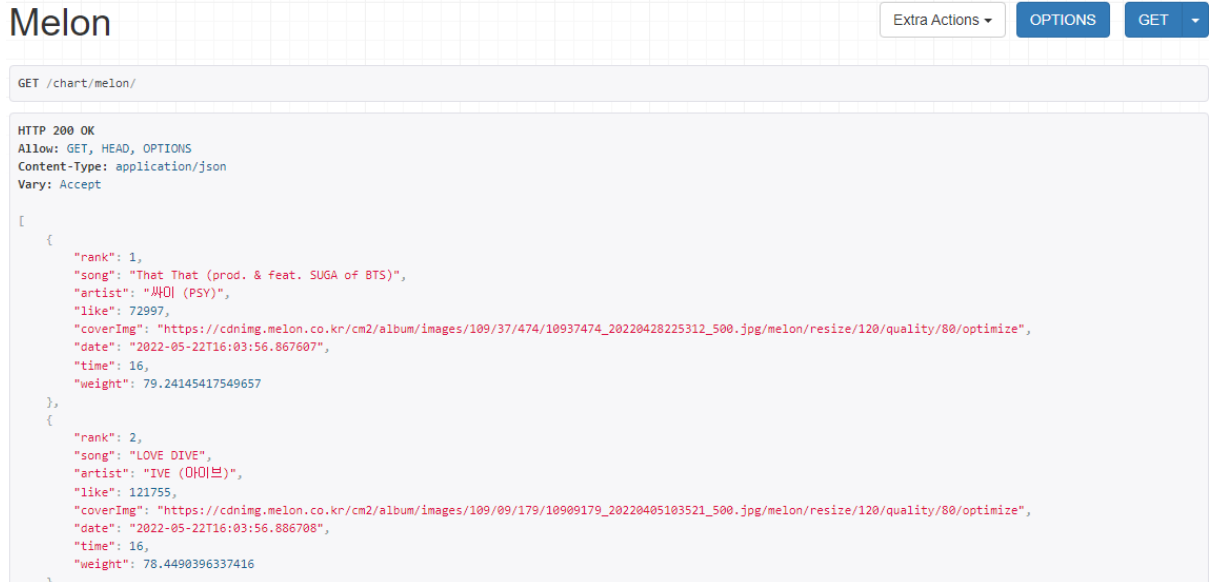
- 기술 : MySQL, AWS RDS
- 데이터 관리 용이성을 위해 AWS RDS를 사용하였다.
- 크롤링 한 데이터를 RDS 데이터베이스에 저장한다.
- 원하는 데이터 조회를 위해 현재 시간을 datetime형태로 같이 저장한다.
- 음원의 전체적인 순위 동향을 파악하기 위해 이전 3일치의 데이터를 저장한다.

2.2.1.3 API 서버 구축

- 프레임워크 : Django REST framework
- RESTFUL API를 개발하고 API를 호출하면 AWS RDS DB에서 조건에 맞는 데이터를 전달

 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

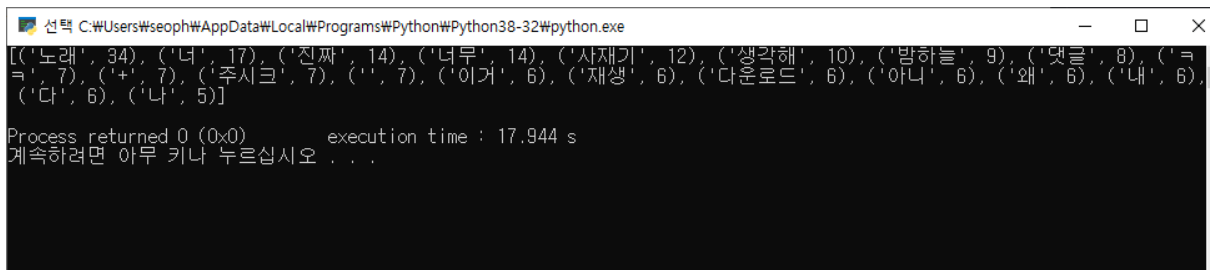
한다.



[그림 3] API 호출 예제 (GET /chart/melon)

2.2.1.4 사재기 댓글 패널티

- 언어 : Python
- 기술 : Selenium
- 사재기로 의심되는 곡들의 댓글을 보면 '사재기', '조작' 등의 부정적인 단어가 대부분이다. 따라서 각 음원의 댓글을 같이 크롤링 한 뒤 부정적인 단어의 빈도 수를 구하고 댓글의 대부분이 이러한 내용일 경우 순위를 보여줄 때 사재기 의심 경고메세지를 같이 보여준다.
- 자바스크립트로 동적으로 생성되는 댓글의 경우 BeautifulSoup을 사용해 크롤링이 불가능하여 Selenium을 사용한다.




[그림 4] 사재기 의심 곡 댓글 크롤링 예제

 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

2.2.1.5 화면 구성

- 사용 프레임워크 및 라이브러리 : React.js, Chart.js
- Home, Chart 등의 페이지를 좌측 사이드바를 통해 접근
- React.js를 이용한 SPA(Single Page Application) 방식으로 구현
- Axios로 Django에서 데이터를 받아오는 코드를 분리해 작업하여 재사용성을 높임
/frontend/src/apis


 국민대학교 소프트웨어학부 다학제간캡스톤디자인	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

2.2.2 시스템 기능 요구사항

기능적 요구사항	내용	진행사항(완료/변경/미완료)
음원사이트 별 랭킹	음원사이트 별 데이터 크롤링	완료
	크롤링 한 데이터 DB저장	완료
	조건에 맞는 데이터 추출	완료
통합 랭킹	크롤링 한 데이터에서 가중치를 기준으로 통합 랭킹 제작	완료
	조건에 맞는 데이터 추출	완료
	댓글을 추출하여 '사재기' 등과 같은 부정적인 단어의 빈도 수에 따른 패널티 부여	미완료
Dashboard	실시간 순위 동향을 보기 쉽게 Dashboard 제작	완료
뮤직비디오 재생	차트 아래 재생 가능한 뮤직비디오버튼 제공	완료
음원 상세페이지	해당 음원의 자세한 정보와 사이트 별 순위 동향 표시	미완료

UC_01	크롤러
설명	멜론, 벅스, 지니의 Top 100 차트 데이터를 크롤링한다.
행위자	없음
선행조건	없음
후행조건	DB에 새롭게 크롤링 된 데이터가 저장된다.
기본 동작	1. 각 사이트의 Top 100 차트를 크롤링한다. 2. 각 곡별 가중치를 분석한다. 3. 크롤링 된 데이터를 DB에 저장한다.
추가 동작	없음

UC_02	통합 차트
설명	가중치를 부여해 나온 통합 차트를 열람할 수 있다.
행위자	사용자

 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

선행조건	DB에 크롤링된 데이터가 있어야한다.
후행조건	통합 차트를 보여준다.
기본 동작	1. DB에서 통합 차트 데이터를 가져온다. 2. Top 100 통합 차트를 보여준다. 3. 각 곡별 유튜브 뮤직비디오를 보여준다.
추가 동작	없음

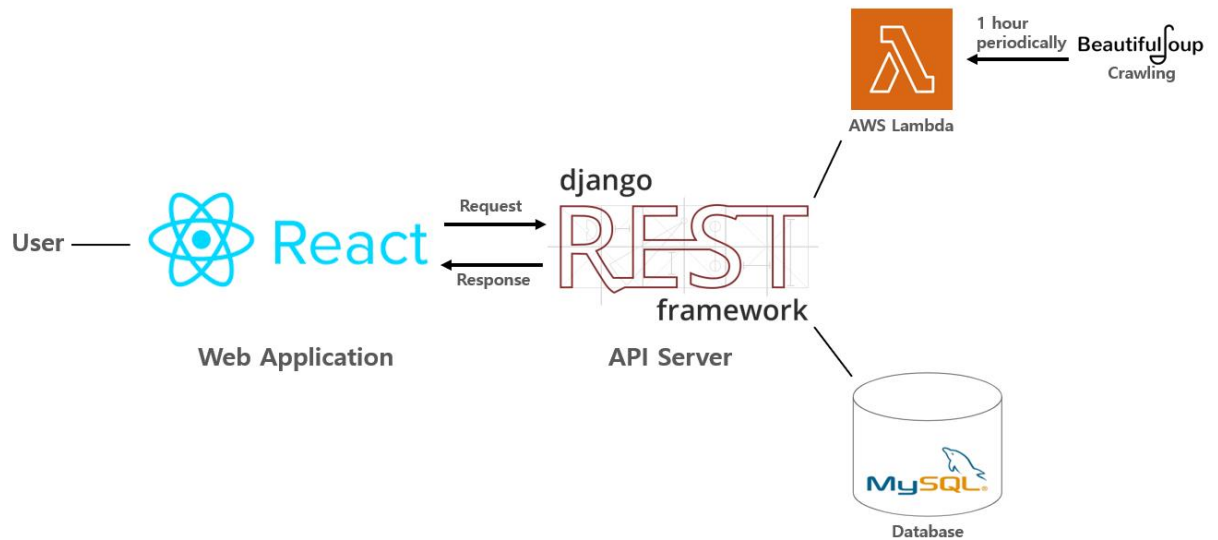
UC_03	대시보드
설명	통합 차트 상위 10곡의 시간별 가중치 변화량 대시보드를 열람할 수 있다.
행위자	사용자
선행조건	DB에 3일간의 데이터가 있어야한다.
후행조건	시간마다 변하는 가중치를 보여준다.
기본 동작	1. 상위 10곡의 그저께~현재까지의 가중치 값을 DB에서 가져온다. 2. 대시보드를 그려 보여준다.
추가 동작	없음

UC_04	멜론, 벅스, 지니 차트
설명	통합 차트와의 비교를 위해 각 사이트의 Top 100 차트를 보여준다.
행위자	사용자
선행조건	DB에 크롤링된 데이터가 있어야한다.
후행조건	각 사이트의 차트를 보여준다.
기본 동작	1. DB에서 선택된 사이트의 데이터를 받아온다. 2. 해당 사이트의 Top 100 차트를 보여준다.
추가 동작	없음

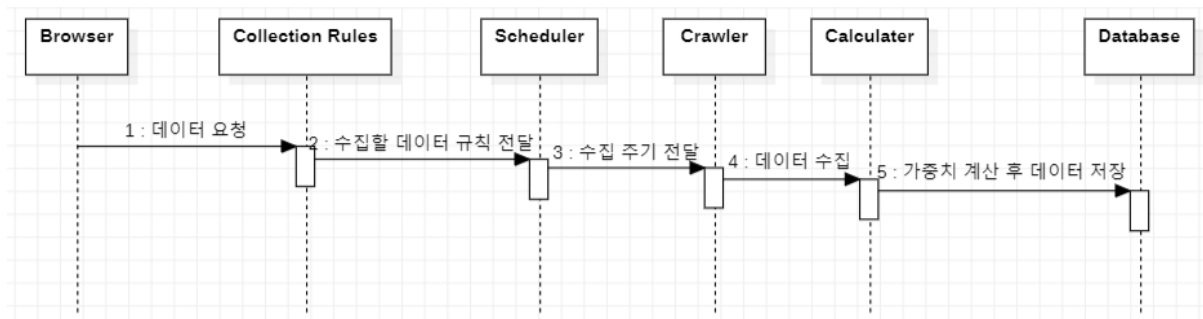
2.2.3 시스템 비기능(품질) 요구사항

비기능적 요구사항	품질 속성	내용	결과
NFR_01	Performance	DB에서 3일 이후 데이터는 삭제되도록 한다. 3개 사이트 크롤링이 1분 이내에 되도록 한다.	달성
NFR_02	Usability	차트를 한눈에 알아볼 수 있는 UI를 만든다.	달성
NFR_03	Interoperability	Client Server간 통신, Server Database간 통신에 제약이 없도록 한다.	달성


2.2.4 시스템 구조 및 설계도

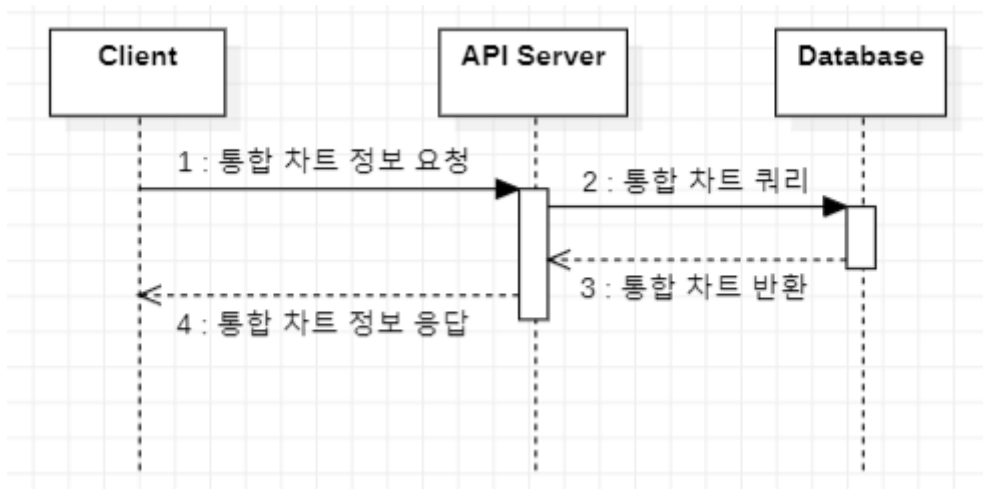


[그림 5] 시스템 구조

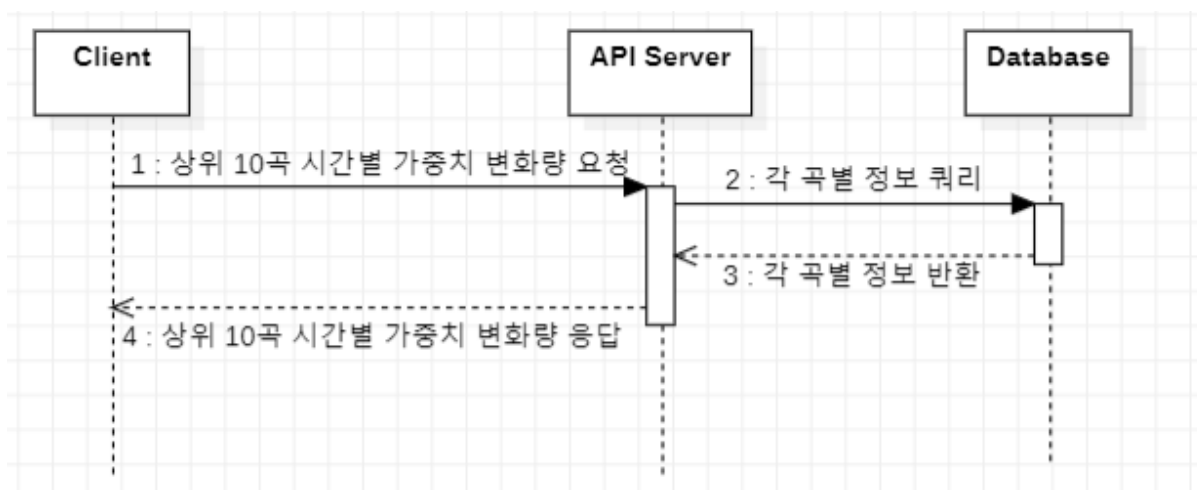


[그림 6] 크롤러 Sequence Diagram

 국민대학교 소프트웨어학부 다학제간캡스톤디자인	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24



[그림 7] 통합 차트 열람 Sequence Diagram



[그림 8] 시간별 가중치 변화량 대쉬보드 Sequence Diagram

2.2.5 활용/개발된 기술

1. BeautifulSoup: HTML, XML, JSON 등 파일의 구문을 분석하는 모듈로 웹 페이지를 표현하는 HTML을 분석하여 크롤링을 하기 위해 사용한다.
2. Selenium: 자바스크립트로 동적으로 생성되는 댓글 내용을 크롤링하기 위해 사용한다.
3. Django REST Framework: Django 사용하여 컴포넌트를 만들어 프론트엔드의 여러 컨테이너에서 재사용이 가능하도록 효율적으로 구현하였다.
4. AWS Lambda: 크롤러가 1시간 주기로 돌아가도록 CloudWatch 트리거를 사용한다.
5. AWS RDS: 크롤링한 데이터를 데이터베이스에 저장하여 조건에 맞게 데이터를 조회할 수 있도록 구현하였다. 데이터베이스는 MySQL을 사용한다.


 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

6. AWS EC2: AWS 클라우드의 가상 서버로 이를 사용해 REST API 를 배포한다.
7. React.js: 화면 구성 담당 라이브러리. Django 에서 가져온 데이터를 화면에 보여준다.
8. Chart.js: 전체적인 순위변화를 Dashboard 에 표현하기 위해 Chart.js 를 사용한다.

2.2.6 현실적 제한 요소 및 그 해결 방안

- 크롤링에 소요되는 시간

시간마다 변경되는 차트를 반영하기 위해 정각에 크롤링을 진행하게 된다. 하지만 멜론, 벅스, 지니 3개 사이트 크롤링에 소요되는 시간이 있어 실시간 반영이 지연된다. 크롤링이 완료되기 전에 접속한 사용자를 위해 정각부터 새로운 크롤링 완료 전 까지 차트 상단에 랭킹 업데이트중입니다 라는 문구를 표시해주었다.

 국민대학교 소프트웨어학부 다학제간캡스톤디자인	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

2.2.7 결과물 목록

대분류	소분류	기능	형식	기술문서
알고리즘	Crawler	각 음원사이트의 Top 100 차트를 크롤링하여 가져온다.	모듈	X
알고리즘	DB 저장	크롤링 된 데이터를 DB 에 저장한다.	모듈	X
알고리즘	통합차트	‘좋아요’ 의 수를 기준으로 가중치를 만들어 통합차트를 생성한다.	모듈	X
웹페이지	차트조회	사이트 별 순위, 통합차트 순위를 조회한다.	컴포넌트	X
웹페이지	대시보드	특정 음원의 순위 동향을 대시보드에 차트로 표현한다.	컴포넌트	X

2.3 기대효과 및 활용방안

- 사용자가 특정 음원사이트 순위에 국한되지 않고 편향성을 줄인 통합 차트를 확인할 수 있다.
- 여러 음원사이트의 순위를 한 눈에 확인할 수 있다.

 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

3 자기평가

최종결과물	통합 차트 서비스 개발
주요 평가 기준	1. 통합 결과를 신뢰할 수 있는가?
자기 평가	종합된 지표를 활용해 편향성을 제거하였고 그 결과 사재기 의혹 및 특정 시간대의 비정상적인 순위 변동을 보여주는 곡들에 대해서 순위 조정을 할 수 있었다.

최종결과물	통합 차트 서비스 개발
주요 평가 기준	2. 사용자가 원하는 서비스인가?
자기 평가	빠르게 변하는 트렌드 속에서 여러 플랫폼을 한눈에 확인할 수 있으며 점수로 음원 순위 별 격차도 정확히 알 수 있도록 구현하였다.

4 참고 문헌

번호	종류	제목	출처	발행년도	저자	기타
1	웹 페이지	Django REST Framework Document	https://www.django-rest-framework.org/			
2	웹 페이지	React Document	https://ko.reactjs.org/			
3	웹 페이지	Chart.js Document	https://www.chartjs.org/docs/			
4	웹 페이지	Bootstrap Document	https://getbootstrap.com/docs/			
5	웹 페이지	PyMySQL Document	https://pymysql.readthedocs.io/			

5 부록

5.1 사용자 매뉴얼

설치 가이드, 따라하기 등 포함

 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

5.2 운영자 매뉴얼

Start API Server

Install virtualenv: \$ pip install virtualenv

Create virtualenv: \$ virtualenv venv

Activate the virtualenv for your project: \$ venv/Scripts/activate

Install project dependencies: \$ pip install -r requirements/local.txt

Then simply apply the migrations: \$ python manage.py migrate

You can now run the development server: \$ python manage.py runserver

5.3 배포 가이드

API Server 배포 가이드

Public DNS: ec2-13-125-170-254.ap-northeast-2.compute.amazonaws.com

1. AWS EC2 서버 생성
2. AWS 서버에서 Github 프로젝트 clone
3. Uwsgi 와 Nginx 연결

5.4 기술문서

API 문서

<https://docs.google.com/spreadsheets/d/1w16TIG3n6Vkruhch60sfRc44Cv9pf6k/edit?usp=sharing&ouid=106055170704077638123&rtpof=true&sd=true>

 국민대학교 소프트웨어학부 다학제간캡스톤디자인I	결과보고서		
	프로젝트 명	멜벅지	
	팀 명	윤하 - 26 많관부	
	Confidential Restricted	Version 1.5	2022-MAY-24

5.5 테스트 케이스

대분류	소분류	기능	테스트 방법	기대 결과	테스트 결과
알고리즘	Crawler	각 음원사이트의 Top 100 차트를 크롤링하여 가져온다.	AWS lambda 를 통해 일정 시간마다 크롤링이 실행되도록 한다.	설정된 주기에 맞춰서 크롤링이 실행된다.	성공
알고리즘	DB 저장	크롤링 된 데이터를 DB 에 저장한다.	크롤링 한 데이터를 pyMySQL 을 통해 AWS RDS 에 저장한다.	AWS RDS 에 해당 내용의 데이터가 저장된다.	성공
알고리즘	통합차트	‘좋아요’의 수를 기준으로 가중치를 만들어 통합차트를 생성한다.	크롤링 한 데이터를 통합한 뒤 가중치와 순위를 비교한다.	가중치의 최대 값이 100 인 통합차트가 생성된다.	성공
웹페이지	차트조회	사이트 별 순위, 통합차트 순위를 조회한다.	좌측 메뉴바에서 Chart, Melon, Bug 메뉴를 선택해 차트를 확인한다.	DB 로부터 현재 시간의 데이터를 가져와서 차트로 보여준다.	성공
웹페이지	대시보드	특정 음원의 순위 동향을 대시보드에 차트로 표현한다.	좌측 메뉴바에서 Chart 를 선택해 대시보드를 확인한다.	DB 로부터 현재 시간의 데이터를 가져와서 그래프로 보여준다.	성공