

# 영화정보 검색 시스템

## (Movie Information Search System)

본 보고서를 2022년도 Project **MISS**의  
최종보고서로 제출합니다.

학부(전공)/학과: 응용소프트웨어공학과

담당(지도)교수: 김남규

팀명: 데이터구조

제출일: 2022년 12월 13일

팀 장 : 이 승 훈 (20193134)

팀원1 : 이 주 환 (20193125)

팀원2 : 이 성 민 (20193127)

팀원3 : 백 승 준 (20193129)

동의대학교 응용SW학과 교수 김남규 귀하

## 〈 요약 문 〉

<p><b>프로젝트 목표</b> (300자내외)</p>	<p>일상 생활에서 사용자가 필요한 조건에 부합하는 영화관에서 영화를 예매하기 위해서는 사용자가 원하는 영화 시간대와 영화관 위치가 필요로 한다. 하지만 전국에 다수의 멀티플렉스가 존재하고 있기 때문에 각 영화관의 위치와 영화사에서 제공하는 영화의 시간대를 알아보기 위해서는 분산되어있는 3사(CGV, 롯데시네마, 메가박스)의 위치와 상영정보를 각각의 홈페이지에 접속하여 확인해야 하는 번거로움이 있었다. 이 문제점을 해결하기 위해 멀티플렉스 3사의 상영 정보를 통합하여 제공하는 서비스를 프로젝트의 목표로 선정하였다</p>		
<p><b>내용</b> (500자내외)</p>	<p>[3사 통합 상영정보 제공] 3사별로 상영정보를 제공하는 방식이 다른데, 이 중 3사가 모두 공통으로 가지고 있는 상영정보(상영시간, 지점, 영화 타입(2D, 3D...), 상영관, 잔여 좌석)를 일정 시간마다 크롤링하여 5일 치를 데이터베이스에 저장해둔 뒤, 사용자가 상세 영화정보를 조회할 때 데이터베이스에서 정보를 가져와 표 형식으로 표시하였다.</p> <p>[영화정보 제공] 3사 통합 상영정보를 크롤링할 때 인식된 영화들의 정보를 영화정보제공사이트 'TMDB'의 API를 사용해 정보를 가져와서 데이터베이스에 저장하였다. 가져온 포스터 정보를 이용하여 메인화면과 영화목록에 표시하고, 상영정보와 같이 영화의 간단한 정보를 표시함으로써 사용자에게 추가적인 정보를 제공한다.</p> <p>[메인화면 정보노출] 영화진흥위원회의 오픈API를 사용하여 주간 박스오피스정보를 가져온 뒤, 이를 바탕으로 최상위권 영화 4개를 사용자가 처음 마주하는 화면인 메인화면에 포스터형식으로 표시한다.</p>		
<p><b>기대효과</b> (200자내외)</p>	<p>기존 멀티플렉스 3사에 퍼져있던 정보를 한곳에 모아 통합해 보여줌으로써, 자신의 조건에 맞는 영화 상영 정보를 찾는데 걸리는 시간을 절약하여 일상생활에서의 편의를 제공한다. 또한 영화에 관련된 내용을 올리는 커뮤니티를 제공하여 시청자들의 실제 후기로 자신이 영화를 볼 것인지 결정하는 지표로 삼을 수 있는 효과와 예매를 유도하는 효과를 가져온다.</p>		
<p><b>Keywords</b></p>	<p>멀티플렉스</p>	<p>영화</p>	<p>상영정보</p>
	<p>통합</p>	<p>한줄평</p>	<p>편의제공</p>

# 목 차

<b>I. 서론 .....</b>	<b>1</b>
1. 프로젝트의 배경 또는 필요성 .....	1
2. 프로젝트의 목표 .....	1
<b>II. 프로젝트 수행 내용 .....</b>	<b>2</b>
1. 프로젝트의 추진전략 및 방법(체계) .....	2
2. 프로젝트 수행 과정 .....	6
3. 프로젝트 수행 결과 .....	18
<b>III. 결론 .....</b>	<b>29</b>
1. 프로젝트 달성도 .....	29
2. 프로젝트 결과 논의 .....	29
3. 프로젝트 결과의 활용방안 .....	30
4. 프로젝트 결과의 기대효과 .....	30
<b>IV. 참고문헌 .....</b>	<b>30</b>

## I. 서론

### 1. 프로젝트의 배경 또는 필요성

우리가 영화를 볼 때, 영화관 상관없이 가장 이른 시간에 보고 싶어 하는 사람도 있을 것이고, 또는 특정 영화관에 혜택(ex영화표, 기프트콘)이 있어 이를 누리하고자 특정 영화관을 선택하는 사람이 있을것 이다. 사용자가 필요한 조건에 부합하는 영화관에서 영화를 예매하기 위해서는 사용자가 원하는 영화 시간대와 영화관 위치가 필요로 한다. 하지만 전국에 다수의 멀티플렉스가 존재하고 있고 어느 영화사를 선택할지 선택권이 넓다는 장점이 있지만, 각 영화관의 위치와 영화사에서 제공하는 영화의 시간대를 알아보기 위해서는 분산되어있는 3사(CGV, 롯데시네마, 메가박스)의 위치와 상영정보를 각각의 홈페이지에 접속하여 확인해야 하는 번거로움이 있었다. 이 문제점을 해결하기 위해 멀티플렉스 3사의 상영 정보를 통합하여 제공하는 사이트가 있다면 유용할 것이라 생각하여 이를 프로젝트의 주제로 선정하였다.

### 2. 프로젝트의 목표

#### 가. 기능 : 회원가입

제공하는 서비스의 목표 중 하나인 한줄평 커뮤니티를 사용하기 위하여 사용자의 닉네임이 필요로 하였기 때문에 개개인의 닉네임을 설정하도록 회원가입 기능 추가를 첫 목표로 하였다.

#### 나. 기능 : 공지사항

사용자가 영화정보를 조회할 때의 데이터는 각각의 영화사의 홈페이지에서 제공하는 정보를 기반으로 제공하는데 예를 들어 영화사의 문제(ex 서버문제)가 있을 경우 필자가 제공하는 서비스 또한 문제가 야기될 것으로 예상이 된다. 하지만 사용자는 이러한 정보를 얻기 어렵기 때문에 영화사의 문제가 발생한 내용을 공지사항을 통해 사용자들에게 알려주는 기능 추가를 두 번째 목표로 하였다.

#### 다. 기능 : 영화정보 통합 제공

일정 시간마다 부산지역의 멀티플렉스 3사(CGV, 롯데시네마, 메가박스)의 영화관에 상영 중인 영화의 정보들 8가지 (1. 영화제목 2. 지점 3. 상영날짜 4. 상영시간 5. 조조 여부 6. 영화 타입 (ex 2D, 3D 등) 7. 상영관 8. 전제/잔여 좌석 수)를 조회일로부터 +4일간의 영화정보를 영화진흥위원회 오픈 API를 이용하여 크롤링한 뒤 크롤링한 데이터를 데이터베이스에 저장하여 이를 가공한 뒤 가독성 좋은 UI를 통해 영화 데이터를 제공하는 기능 추가를 세 번째 목표로 하였다.

#### 라. 기능 : 한줄평 커뮤니티

우리의 서비스를 사용하여 영화를 시청한 사용자들이 영화에 관한 이야기를 할 수 있는 커뮤니티 기능 추가를 네 번째 목표로 하였다.

### 마. 최종목표

위 기능을 토대로 하여 각 영화관의 위치와 영화사에서 제공하는 영화의 시간대를 포함한 구체적인 상영정보를 제공하였고 기존의 상영정보를 알아보기 위해서는 부산지역 3사에 분산되어있는 위치와 상영정보를 각각의 홈페이지에 접속하여 확인해야 하는 번거로움을 해소할 수 있는 사이트를 구축하는 것이 이번 프로젝트의 최종 목표이다.

## II. 프로젝트 수행내용

### 1. 프로젝트 추진전략 및 방법

#### 가. 사전조사

프로젝트 진행을 위해 2주 동안 사전 조사를 하기로 하였다. 프로젝트의 핵심 목표 중 하나인 영화정보 통합 후 제공을 위해 필요한 데이터를 어떻게 수집하고 가공할 것인가를 정했다. 영화 상영 정보수집 후 CGV, 롯데시네마, 메가박스의 홈페이지를 분석하여 페이지의 구조를 파악한 뒤 3사 영화사의 웹 크롤링을 어떻게 할지 구상했고, 이에 사용되는 플랫폼이나 프로그래밍 언어를 어떤 것을 사용할지 정했다.

#### 나. UI디자인

3사 영화사의 메인 홈페이지, 상영시간표 등 UI를 각자 확인한 후 팀 회의를 통하여 메가박스의 디자인을 참고하기로 하였다. 또한 프론트엔드 구현을 손쉽게 하기위해 프론트엔드 프레임워크로 부트스트랩을 사용하기로 하였다. 홈페이지의 아이콘과, 각 기능을 구성하는 네비게이션 바, 컨테이너, 체크박스, 버튼 등 여러 컴포넌트는 부트스트랩에서 제공하는 아이콘과 컴포넌트를 사용하기로 하였다.

#### 다. 프로젝트 환경구축

24시간 구동과 프로젝트 진행을 위한 접근성을 위해 실제 운영되고 있는 팀장의 개인 서버를 사용하여 필요한 서비스들을 구축하였다.

웹서버 가상머신 (nginx - 가상호스트(misc.~))	데이터베이스 가상머신 (Docker - MariaDB)	Node.js 가상머신 (Express Module, API)
ESXi 하이퍼바이저		

또한 외부에서 접속하기 위해 개인 도메인을 연결하고 포트포워딩을 하였다.

#### 라. 프론트엔드, 백엔드 작업

##### 1) 영화정보, 상영일정 수집

먼저 크롤러가 영화진흥위원회에서 제공하는 영화 오픈API를 사용하여 하루 전날 박스오피스 1~10위의 영화정보를 가져온 후, TMDb API를 사용하여 전 단계에서 조회한 영화의 정보를 보충하고 데이터베이스에 삽입한다. 이러한 과정으로 영화정보를 저장한 후 팀원들이 만든 영화사별 크롤러를 통해 3사의 상영일정을 수집하고, 앞서 저장한 영화정보에 포함되어있는 영화의 상영 일정만 데이터베이스에 삽입한다.



2) 프론트엔드, 백엔드 연결



백엔드 데이터를 기반으로 프론트엔드 조작을 쉽게하기위해 자바스크립트(jQuery)를 적극적으로 사용한다. 데이터베이스 접근은 자바스크립트에서 PHP로 ajax 요청을 보내고, PHP단에서 유효성검증(사용자 권한확인, 쿠키 변조여부)을 한 후 쿼리를 실행하여 요청지로 응답을 보낸다. 자바스크립트는 받은데이터를 통해 웹페이지 요소를 동적으로 생성하거나 조작하여 사용자에게 정보를 적절하게 표시한다.

[표 1] 프로젝트 추진 일정

No	프로젝트 활동 내용	추진 일정											기간 (주)	추진 방법
		1	2	3	4	5	6	7	8	9	10	11		
1	프로젝트 사전조사												2	CGV, 롯데시네마, 메가박스 홈페이지 구조 파악
2	프로젝트 UI 디자인												1	팀 회의
3	프로젝트 환경구축												1	팀장 개인서버 이용 MariaDB 이용
4	FE 작업 (웹)												5	BE와 상호작업
	BE 작업 (API, PHP)												5	FE와 상호작업
5	프로젝트 마무리												2	사이트 작동확인, 세부사항 마감 수정

[표 2] 프로젝트 추진 체계

성명 / 학번		수행내용	역할
	이승훈	<ul style="list-style-type: none"> <li>• 팀원 이승민이 고안한 사이트 초기 디자인을 바탕으로 전반적인 디자인을 구상</li> <li>• 팀원들의 멘티가 되어 팀원들이 프로젝트를 진행하면서 막히는 부분이나 어려워 하는 부분을 해결해주고 조언해 줌</li> <li>• 전체적인 자바스크립트, PHP 백엔드 로직 구현, 다수의 PHP파일을 통합하여 간소화함</li> <li>• 팀원들의 아이디어를 바탕으로 사용자의 입력을 받아 로그인하는것과, 회원가입, 영화 내용 디스플레이, 한줄평 예외처리 등 홈페이지 대부분의 자바스크립트 로직을 구현</li> <li>• 팀원들의 일정을 반영하여 회의계획을 수립하고 주도함</li> <li>• 메가박스 크롤러 구현을 위해 팀원 이주환과 함께 메가박스 홈페이지를 분석하였으며, 이를 바탕으로 크롤러의 기본을 구현함. 후에 이것을 바탕으로 팀원들이 구현한 크롤러를 통합함.</li> <li>• 팀원들이 받아온 피드백을 수용하여 가능한부분은 즉시수정하여 사용자들의 만족을 이끌어냄.</li> <li>• 팀원 백승준이 정리한 보고서를 받아 서식을 정리한 후 최종본을 작성하였다.</li> </ul>	팀장
	20193134		
	이주환	<ul style="list-style-type: none"> <li>• 각 영화사 홈페이지 분석 후 메인 배너 UI부분은 메가박스 UI를 참고하면 좋을 것 같다고 의견을 냈다.</li> <li>• 팀장 이승훈이 FE 개발 중 화면 비율, 텍스트 위치 조정, 디자인 및 기능 추가사항 등 세세한 부분들을 보조했다.</li> <li>• 팀원 백승준의 보고서 작성 요령을 받고 그 지식을 토대로 하여 기능 명세서 부분을 작성했다.</li> <li>• 자바스크립트 기본적인 문법, 구조 공부 후 node.js 플랫폼, 자바스크립트 언어를 사용하여 팀장 이승훈의 메가박스 상영시간표 크롤링을 보조했다.</li> <li>• 3사 영화사 홈페이지의 최하단 내용을 확인하고 최하단 배너에 로고, 팀 구성원 정보, 저작권 표시 디자인을 구상하고 의견을 냈다.</li> </ul>	팀원
	20193125		

		<ul style="list-style-type: none"> <li>• 팀장 이승훈이 BE 구현(크롤링) 중 멘탈이 나갔을 때 격려의 말로 멘탈케어했고, 팀원 이성민이 CGV 상영정보를 크롤링 하는데 어려움을 느껴서 위로의 말과 지식공유로 멘탈케어했다.</li> </ul>	
	이성민	<ul style="list-style-type: none"> <li>• 프론트엔드 초기 디자인 및 구축 HTML, CSS, 자바스크립트를 기반으로 처음 사이트 접속 시 로딩화면, 사이트의 메인화면, 로그인화면, 회원가입화면 등을 제작하였으며, 이렇게 초기버전의 개발을 진행하였다.</li> <li>• 초기 디자인을 바탕으로 개선된 사이트 개발을 진행할 때 사이트의 컨셉을 제시하는 등 개선된 사이트의 개발에 대한 의견을 제시하였다.</li> <li>• 크롤러 개발을 위한 CGV 홈페이지 분석을 진행하였다.</li> <li>• 크롤러 개발을 위한 학습을 진행하였다. 크롤러에 사용되는 node.js, npm, request, cheerio에 대해 공부하였다.</li> </ul>	팀원
	20193127		
	백승준	<ul style="list-style-type: none"> <li>• 본 프로젝트에서 팀원들이 각자 맡은 역할에서의 수행 내용들을 취합하여 보고서를 작성하였다.</li> <li>• npm(자바스크립트 패키지 관리자)과, node.js 플랫폼, 자바스크립트 언어를 사용하여 롯데시네마 부산 전 지점 크롤링하였다.</li> <li>• 팀원 이성민의 역할인 CGV 크롤링 과정 중 해당 팀원이 완벽히 수행하지 못한 부분을 추가로 작업하여 CGV 부산 전 지점 크롤링을 마무리하였다.</li> <li>• 홈페이지의 완성도를 높이기 위해 몇몇의 지인들에게 피드백을 받아 팀장 이승훈에게 전달하였고 위 내용들을 보고서에 작성하였다.</li> <li>• 팀원 이주환이 보고서 작성 중 어려움을 느껴 최소한의 보고서 작성법에 대한 지식을 공유하여 기능명세 작성 부분에 도움을 주었다.</li> <li>• 구축한 웹에서 팀장 이승훈이 미처 처리하지 못한 예외를 찾아 피드백하는 작업을 수행하였다.</li> </ul>	팀원
	20193129		



## 2. 프로젝트 수행과정

### 가. 프로젝트 주제선정

제일 처음 수행한 과제인 프로젝트의 주제 브레인스토밍 과정에서 제안한 주제는

1. 사용자 위치 기준 주변 버스 정보
2. 사이버 헬스트레이너
3. 가장 빠른 배달 음식점 추천 서비스
4. 영화 정보 통합 검색 서비스
5. 전적검색으로 위 다섯 가지가 있었다.

브레인스토밍 후 우리 팀의 프로젝트 주제 선정 기준은 아래와 같은 두 가지로 나누었다.

1. 우리가 일상생활을 지낼 때 IT에 관련한 불편함을 경험한 곳에서 '이러한 서비스가 있으면 좋을 것 같다' 라고 생각되는 것.
2. 존재하지 않는 서비스이거나, 이미 기존에 존재하는 서비스이지만 추가 또는 개선되면 좋을 것 같은 서비스. 위 두 조건을 기준으로 주제로 선정하기로 하였다.

#### 1) 제안 1 : 버스 도착 정보 제공 서비스

버스 도착 정보 제공 서비스는 위치를 기준으로 주변 버스 정류장의 도착 버스 정보와 도착 시간을 제공하는 서비스이다. 위 안건은 팀원 다수가 일상생활에 불편함을 경험하여 주제 선정 기준에 부합하였지만 사용자의 위치를 기반으로 하는 서비스다 보니 우리 팀의 현재 지식으로는 구현할 수 없다고 판단하여 제외하였다.

#### 2) 제안 2 : 사이버 헬스 트레이너

사이버 헬스트레이너는 사용자의 신체 정보를 입력하면 그에 맞는 식단과 운동 계획을 짜 주는 서비스이다. 위 안건은 이를 제안한 팀원이 구체적인 구현 내용과 기술 스택을 충분히 설명하지 못하였고, 다른 팀원들이 위와 같은 불편한 경험을 공감하지 못하여 제외하였다.

#### 3) 제안 3 : 배달앱

배달 앱은 위치 기준 가장 빠른 배달 음식점 추천과 서비스에 가입한 유저들끼리 파티를 꾸려 사람들과 같이 식사할 수 있게 하는 서비스이다. 위 안건은 제안한 내용이 다른 팀원들은 불편함을 공감하지 못하였고, 매력적인 서비스가 아니라고 판단하여 제외하였다.

#### 4) 제안 5 : 전적검색 서비스

다섯 번째 제안인 게임 전적검색은 말 그대로 롤, 발로란트, 오버워치 등 게임 전적 검색 서비스이다.

위 안건은 이미 많이 상용화된 서비스이기도 하며 숙련된 개발자들이 개발한 서비스기 때문에 우리가 생각하는 유용한 기능들이 충분히 구현되어 있었기에 개선할 점을 찾지 못하여 제외하였다.

## 5) 채택된 제안 – 영화 상영정보 통합 제공 서비스

앞의 네 가지 안건들은 두 가지 주제 선정 기준 조건에 부합하지 못하였기 때문에 제외되었다. 남은 네 번째 제안인 영화 정보 통합 검색 서비스는 전국의 멀티플렉스 3사(CGV, 롯데시네마, 메가박스)에서 상영중인 영화의 정보를 한곳에 모아주는 서비스이다. 위 안건은 팀원들이 한 번씩 영화를 예매할 때 각각의 홈페이지를 들어가 비교해야 하는 불편한 경험을 한 적이 있었고, 찾아본 바로는 모든 영화의 정보를 제공하는 서비스는 존재하였으나 전국 멀티플렉스 3사의 정보를 통합하여 보여주는 서비스는 존재하지 않았기 때문에 두 가지 주제 선정 기준에 모두 부합하여 채택하였다

## 나. 기술 스택 조사 및 구현 방안 구체화

### 1) 구현 방안 구체화

프로젝트 주제 선정 후 위 프로젝트에서 사용되는 기술 스택의 조사와 구현 방안의 구체화, 역할 분담을 실시하였다. 우리가 선정한 주제는 영화 정보 한눈에 보기 서비스인데 위 서비스에 필요한 데이터를 어떻게 수집할 것인지 3사(CGV, MEGABOX, 롯데시네마) 각각의 메인 홈페이지 구조를 파악하였다. 처음 메인 홈페이지를 접했을 때는 해당 페이지에 나오는 영화들로 데이터를 수집하려고 하였다. 하지만 CGV와 롯데시네마는 메인 화면에 영화가 1위부터 20위까지 나열되어 있었으나, 어떠한 기준으로 순위를 정해 보여주고 있는지 알 수 없었고, 심지어 메가박스는 박스오피스 1위부터 4위까지의 영화만 보여주고 있었다. 각 영화사의 메인 홈페이지에서 보여주는 영화들이 모두 달라 메인 홈페이지에서는 필요로 하는 데이터를 수집할 수 없다고 판단하였다. 또한 메인 홈페이지의 데이터는 영화의 상세정보나 예매율, 별점 등은 있었으나 우리가 필요로 하는 상영 시간대나 세부 정보가 없었다. 이에 팀장 이승훈은 '우리는 부산의 모든 지점의 영화관 데이터가 필요하기 때문에 부산 내 모든 지점 영화관에서 상영중인 영화 정보를 전부 가져오자' 라는 의견을 제시했다. 이러한 방식으로 데이터를 수집하면 부산의 각 지점의 영화관에서 선택한 날짜에 상영하는 영화와 상영 시간, 상영관, 좌석 정보 등의 데이터를 추가로 수집할 수 있었다. 또한 모든 영화관 홈페이지에는 해당 영화사의 부산 모든 지점 영화관 상영 정보를 볼 수 있도록 하는 페이지가 구현되어 있었다. 따라서 우리는 이 의견을 토대로 각 지점에서 상영하는 모든 영화의 데이터를 조회일 +4일치 분량으로 수집하기로 하였다. 이와 같은 데이터 수집으로 기존의 메인 홈페이지에서 데이터를 가져오기로 했던 의견보다 더욱 많은 데이터와 조조여부, 상영관, 영화 타입 등 더 세세한 정보 또한 가져올 수 있게 되었다.

### 2) 기술스택 조사

#### 가) 프론트엔드

프론트엔드 분야 현업에서 자주 사용하는 React.js 또는 Vue.js를 사용하여 프로젝트를 진행하려 했으나, 단기간에 해당 기술 스택을 배우고 프로젝트에 적용하기엔 무리라고 판단하여 HTML, CSS, 바닐라 자바스크립트(순수 자바스크립트, 일부 라이브러리)를 사용하기로 하였다. 자바스크립트 측에서 AJAX를 편하게 사용하기 위해 jQuery 라이브러리를 추가로 사용하기로 하였다.

#### 나) 백엔드

백엔드 측에선 크게 두 가지 작업을 한다. 먼저 데이터 크롤링이다. 웹 크롤링은 크게 파이썬, node.js를 사용하여 할 수 있는데, 개발 속도를 높이기 위해 익숙한 자바스크립트 기반인 node.js를 사용하기로 하였다. 크롤링한 데이터를 저장할 데이터베이스로는 지금까지 배워왔고 무료인 MariaDB를 사용하기로 하였다. 또한 백엔드와 프론트엔드를 연결하기 위해 PHP를 사용하기로 하였다.

### 3) UI 디자인

서비스를 제공하는 홈페이지의 UI 디자인은 팀원의 과반수 동의로 가장 가독성이 좋다고 생각하는 메가박스 홈페이지를 참고로 하여 만들었다.

#### 가) 메인화면 배너

MISS 홈페이지는 메가박스의 메인 홈페이지와 비슷하게 만들기로 하였다. 메인 화면 배너의 왼쪽 부분에는 CGV, 롯데시네마, 메가박스의 로고 이미지를 삽입하여 이미지를 클릭하면 각 영화사의 홈페이지로 이동하게 하며, 중간 부분에는 MISS로고를, 오른쪽 부분에는 공지사항, 전체 영화, 로그인 부분을 구현하였다.

#### 나) 공지사항

사용자들에게 주요 정보를 제공하는 공지사항 커뮤니티는 동의대학교 공지사항 UI를 참고하여 만들기로 하였다. 공지사항에 업로드되는 게시글은 ID, 제목, 게시 날짜, 분류(공지, 이벤트 등), 조회수 형식으로 게시되며 위 게시글을 클릭하면 게시판이 팝업 형식으로 출력된다. 팝업된 창에는 공지 제목, 작성일, 조회수, 공지사항 내용 등이 출력되고 우측 하단의 닫기 버튼이나 우측 상단의 X 버튼을 클릭 시 창을 닫는다.

#### 다) 현재 상영 영화, 박스오피스에 있는 인기 영화 나열

배너의 밑에는 현재 상영 영화 중 영화진흥위원회에서 제공하는 어제의 박스오피스 API를 활용하여 1위부터 4위에 해당하는 영화의 포스터를 띄운다. 현재 상영 영화 텍스트의 바로 우측에는 전체 영화를 볼 수 있는 바로가기 링크를 배치하여 1위부터 4위까지의 영화뿐만 아니라 10위까지의 영화를 볼 수 있고 1위부터 4위까지의 포스터를 클릭하면 영화 상세정보 페이지로 이동하게 된다. 위 페이지로 이동하면 영화 포스터가 좌측에 작게 배치되어 있으며 위에서부터 영화 제목, 개봉 날짜, 누적 관객과 총 매출액, 줄거리 설명 등 영화 상세정보를 볼 수 있도록 구현하였다.

#### 라) 상영 일정

박스오피스에 나열된 10가지의 영화 중 사용자가 보고 싶은 영화를 선택하면 사용자가 원하는 조건에 따라 영화의 정보를 출력해야 하는데, 이를 위해 필터 박스를 추가하였다. 위 필터를 통해 사용자가 원하는 영화사와 지점, 날짜를 선택하면 정렬기준에 따라 영화의 상영 정보가 출력된다.

#### 마) 한줄평

한줄평 또한 메가박스의 UI를 참고하여 만들기로 하였다. 한줄평 기능은 작성자의 닉네임을 필요로 하기 때문에 로그인을 하지 않으면 작성 할 수 없도록 하였다. 또한 통합된 한줄평 커뮤니티가 아닌 사용자가 선택한 영화에 작성된 한줄평만 모아놓은 영화 각각의 한줄평 커뮤니티를 개설하기로 하였다.

#### 바) 전체영화

메가박스의 메인 화면에서 전체 영화를 확인해보면 포스터 좌측 상단에 순위를 나타내고 순위별 오름차순 정렬로 나열했습니다. 메가박스는 전체 영화들을 다 나열했기 때문에 4개로 나눠서 보여주고 있지만 저희 MISS 홈페이지는 주간별 박스오피스 10개 영화의 데이터만 가져오기 때문에 2개만 크게 띄워놓고 나머지 영화들은 4개씩 나눠서 순위별 내림차순 정렬로 나열합니다.

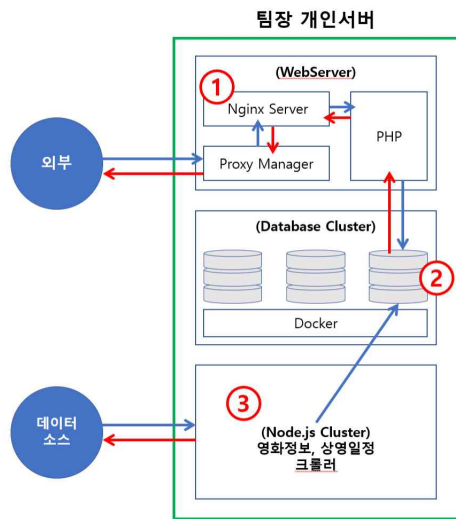
#### 사) 로그인, 회원가입

로그인 및 회원가입 UI는 부트스트랩을 사용하여 만들기로 하였다. 한줄평 또는 공지 사항을 작성하기 위해서 회원가입을 해야 하고, 회원가입 구성요소는 아이디, 중복확인 버튼, 비밀번호, 비밀번호 확인, 닉네임, 중복확인 버튼으로 구성하기로 하였다. 위와 같은 형식에 맞게 입력한 후 회원가입이 성공하면 회원가입 성공 알림이 뜬 후 메인화면 홈페이지로 이동한다. 회원가입 시 입력한 정보들을 로그인할 때 입력한다.

### 4) 환경 구축

프로젝트 초기에 다른 팀들은 어떻게 환경을 구축했는지 조사해보았는데, 대부분 학교 실습컴퓨터에 XAMPP를 설치하여 웹 서버와 데이터베이스 서버를 구현하였다. 이렇게 했을 때 단점으로는 학교 에너지 정책과 같은 문제로 컴퓨터가 항상 켜져있다는 보장이 없기 때문에 24시간 구동이 불가하고, 외부에서 접근할 때 학교 NAC(Network Access Control)에 차단되거나 포트개방 불가능 등 여러 문제가 있어 외부에서 작업하기 어렵다는 점이 있다. 실제로 이러한 문제에 막혀 진행을 못하는 팀도 보았다. 따라서 필자의 팀은 팀장의 개인서버를 사용하여 프로젝트를 진행하기로 하였다. 개인서버의 운영체제로는 가상머신 여러개를 띄울 수 있는 베어메탈 하이퍼바이저인 VMWare사의 ESXi가 설치되어있다. 현재 구동중인 가상머신은 다음과 같다.

가상머신 이름	설명
Node.js Cluster	Discord 봇, API 서버 구동 중
Service Cluster	Docker가 설치되어있고, JupyterLab, 다수의 code-server가 컨테이너로 구동 중
WebServer	Nginx Proxy Manager, Nginx 서버 구동 중
Database Cluster	Docker가 설치되어있고, 다수의 Redis, MariaDB서버가 컨테이너로 구동 중



WebServer에 설치되어있는 Nginx 웹 서버에 가상호스트로 프로젝트를 위한 공간을 마련한다

(①, [miss.dev-friox.com](http://miss.dev-friox.com)).

Database Cluster 가상머신에 영화정보, 게시판 데이터 등 서비스에 관련된 데이터를 저장하는 MariaDB 서버를 Docker 컨테이너로 구동하도록 하고(②), Node.js Cluster 가상머신에 영화정보, 멀티플렉스 3사의 상영정보를 수집하여 데이터베이스에 삽입하는 크롤러를 구동하도록 한다(③).

또한 외부에서 쉽게 접속하기 위해 서버 메인도메인인 dev-friox.com의 서브도메인으로 miss를 사용하여 가상호스트를 연결하였다. ([miss.dev-friox.com](http://miss.dev-friox.com))

모든 연결은 SSL이 적용된 Proxy Manager(Nginx Proxy Manager)를 통해 수립되므로, POST요청의 데이터를 포함한 연결정보는 암호화되어 전송된다. 그리고 가정용 UPS가 연결되어있기 때문에 정전과 같은 재해로부터 데이터는 안전하다.

## 5) 프론트엔드, 백엔드 작업

### 가) 백엔드

백엔드 작업으로는 백엔드를 담당하기로 한 팀원이 각각 멀티플렉스 3사의 홈페이지를 맡아 크롤러 구현을 진행하였으며, 팀장을 제외한 나머지 팀원은 크롤링에 대한 기초적인 지식조차 없었기 때문에 각자 자신에게 필요한 기술 지식을 습득 후 크롤러 구현을 시작하였다. CGV와 롯데시네마는 팀원 백승준, CGV 보조로 팀원 이성민, 메가박스는 팀장 이승훈, 메가박스 보조로 팀원 이주환이 각각 맡았다. 팀원들이 개인적으로 공부한 내용으로 크롤러 구현을 진행하였기 때문에, 동일한 규칙으로 코드를 작성하지 않아 일관성이 없는 등 여러문제가 있었다. 이러한 문제를 해결하기 위해 팀장 이승훈이 최종적으로 코드를 검수하여 수정하고, 3사 크롤러 코드를 통합하였다.

#### (1) 롯데시네마, CGV 크롤링

CGV와 롯데시네마 홈페이지에 대한 크롤러 구현은 팀원 백승준이 수행하였다. 해당 팀원 또한 기본적인 지식이 없었기에 팀장 이승훈은 크롤러 구현을 수월하게 진행할 수 있도록 npm(자바스크립트 패키지(모듈) 관리자)과 node.js, 자바스크립트 언어에 대한 기초적인 지식을 공유하고 가르쳐주었다. npm을 사용하기 위해 node.js를 설치하였고, npm init 명령어를 통해 새로운 프로젝트를 만들었다. 해당 명령어를 실행하면 프로젝트 이름부터 버전, 작성자 등 프로젝트의 정보를 바탕으로 패키지의 정보를 생성해주는 명령어이다. 명령어를 실행한 후 프로젝트 진입점인 index.js 파일을 만들어 코드를 작성한 뒤 실행하면 크롤러가 작동하는 과정을 볼 수 있다. 롯데시네마 홈페이지를 크롤링하기 위하여 Cheerio와 Puppeteer 모듈을 설치해 사용하였는

데, Cheerio는 홈페이지 소스를 받아 필요한 부분의 정보를 손쉽게 파싱할 수 있게 해주는 모듈이다. Cheerio의 장점은 QuerySelector를 그대로 사용할 수 있다는 점인데, 해당 기술을 사용하면 CSS 선택자처럼 필요한 부분을 쉽게 선택할 수 있다. 또한 1학기 웹프로그래밍 I 수업에서 습득한 마크업 언어, 선택자 지식을 적절히 사용할 수 있다. Puppeteer는 DevTools 프로토콜을 통해 크로미움엔진을 제어하기 위한 API를 제공하는 Node.js 패키지이다. Puppeteer를 사용하여 크롤링을 진행한 이유는 위 라이브러리의 특징인 동적 웹 페이지의 크롤링이 가능하다는 점이다. 본 서비스는 조회일로부터 +4일간, 즉 5일치의 상영일정을 제공해야하는데, 멀티플렉스 3사 홈페이지 모두 한 페이지에 여러 날짜의 상영일정이 표시되는 것이 아니고 하루를 조회한 후 다음날짜가 적힌 버튼을 클릭해야 다른날짜의 상영일정이 표시된다. 이렇게 웹페이지에 요청하자마자 돌아오는 응답을 사용하는 것이 아닌 우리가 평소에 웹페이지 요소를 클릭하고 입력하는 것처럼 실제로 사용하는 브라우저와 비슷한 과정을 거쳐야한다. 이와같이 웹페이지 로딩 후 추가적으로 일련의 과정을 거친뒤 다시 정보를 렌더링하는 페이지를 동적 웹페이지라고 하는데, 이것을 크롤링하기 위해 puppeteer에 내장된 click() 메서드를 사용해 이를 구현할 수 있다. 홈페이지를 가져오기 위해서는 해당 홈페이지의 URL이 필요한데 기존에 시도한 방식은 첫 페이지를 goto메서드와 URL을 사용하여 접속하고 이후에는 다른지점으로 이동하는 버튼을 하나씩 누르게 하여 페이지를 이동하도록 했지만, 팀장 이승훈은 버튼을 눌러서 넘어가는것보다 모든 페이지를 전부 goto메서드와 URL을 사용하여 이동하게 하는 방법이 더욱 효율적이라고 조언하였다. 마침 각 지점별 URL의 내용은 모두 동일했고, URL의 마지막에 지점별 고유 코드만 달랐다. 자바스크립트 문법인 백틱과 \${}를 사용하면 문자열에 변수를 적절히 사용할 수 있다는 이승훈 팀장의 말을 듣고, 배열에 각 지점의 코드를 저장하여 반복문과 백틱을 사용하여 페이지를 넘어갈 수 있게 구현하였다.

#### (가) 롯데시네마 크롤러 시행착오

항상 홈페이지의 구조가 같을것이라고 생각하여 모든 데이터를 처음 로딩된 페이지의 DOM 구조를 바탕으로 가져오게 하였다. 하지만 다음날의 상영일정 데이터를 가져오기 위하여 다음 날에 해당하는 날짜 버튼을 누르니 페이지의 DOM 구조또한 바뀌어 코드가 정상적으로 작동하지 않게 되었다. 이 문제를 해결하기 위해 홈페이지의 DOM 구조가 바뀌지 않게 하면서 데이터를 가져올 수 있게 하는 방법을 찾아보았다.

#### [해결방안]

홈페이지가 로드 된 후 다른날의 상영일정을 볼 수 있는 날짜 버튼을 클릭하면 새로고침을 하지 않는 이상 다른 날의 데이터를 불러와도 구조가 바뀌지 않는 것을 알게 되었다. 따라서 홈페이지가 로드되면 조회일에 해당하는 날의 버튼을 한번 클릭하여 DOM구조를 한번 바꾼 후 그 구조에 맞게 데이터를 가져오도록 코드를 수정하였다.

(나) CGV 크롤러 시행착오 1

CGV 상영일정 홈페이지 구조는 iframe 이라는 인라인 프레임 요소가 들어가있는데, 이는 현재 페이지안에 다른 HTML 페이지를 삽입하는 것이다. 우리가 필요로 하는 데이터는 iframe안에 존재하였고, 롯데시네마와 똑같은 방법으로 크롤링을 시도하였으나 데이터를 가져오지 못했다. 로드한 페이지에 해당하는 DOM구조와 iframe 내부에 존재하는 HTML의 DOM구조가 달라 Cheerio에서 읽어내지 못하여 데이터를 가져오지 못하였다. 즉 Cheerio에 넘긴 페이지는 부모 페이지이지만 전달한 인자들은 iframe 내부 페이지의 선택자이므로 부모페이지에 존재하지 않는 선택자들이라 데이터를 가져올 수 없었다. 위 문제를 해결하기 위하여 팀원 백승준은 iframe에 접근할 수 있는 방법을, 팀원 이성민은 모바일 웹 CGV 홈페이지에 들어가 크롤링을 하는 방법을 모색하였다. iframe에 접근하여 데이터를 가져올 수 있는 방법을 알아내기 전, 팀원 이성민은 모바일 웹 CGV 홈페이지에서 데이터를 추출할 수 있었다. 하지만 추출해낸 데이터는 팀장 이승훈조차 가공하기 힘들만큼 알아보기 힘들었고, 팀원 백승준 또한 모바일 웹을 크롤링한 결과 모바일 웹 홈페이지의 DOM구조가 팀장 이승훈이 지시한 데이터 가공 형식에 맞게 데이터를 추출, 가공해 올 수 없을 정도로 요연하지 못하였다.

[illegible]

▲ 모바일 웹에서 크롤링한 데이터. 알아볼 수 없을 정도로 난잡하다.

[해결방안]

위와같은 문제점으로 CGV는 iframe 내부에 접근하여 크롤링하는 법 말고는 데이터를 추출해 낼 수 없다고 판단하였다. 이에 팀원 백승준과 이성민은 해결방안을 모색하였고, puppeteer 모듈에서는 iframe의 프레임 id만 안다면 접근할 수 있다는 것을 알게되었다. 다행히 iframe의 ID는 쉽게 알아낼 수 있었고, 기존의 Cheerio에서 페이지를 로드하는것과 같은 방식으로 iframe을 로드하니 내부 DOM 구조에 접근할 수 있었다.

(다) CGV 크롤러 시행착오 2

iframe 내부에 접근하여 추출한 데이터를 콘솔에 출력하였더니 추출한 데이터의 앞, 뒤로 수많은 공백과 줄 바꿈 문자가 존재하여 데이터가 정돈되지 못한 상태로 들어왔다는 것을 관찰할 수 있었다.

[해결방안]

위 문제를 해결하기 위해 공백을 없애는 trim()함수, 특정문자를 기준으로 앞, 뒤를 나누어 주는 split() 함수에 대한 자바스크립트 언어 공부를 진행하여 데이터를 가공하는 것으로 문제를 해결하였다. 또한 영화 상영시간의 시작시간은 있었으나 종료시간은 존재하지 않았다. 이를 해결하기 위해 영화 상영 정보에 존재하는 영화의 러닝 타임을 가져와 시작시간에 더해줌으로써 종료 시간을 구현해낼 수 있었다.

(2) 메가박스 크롤러

메가박스 홈페이지 크롤링은 팀장 이승훈이 담당하였고, 팀원 이주환이 보조하였다. 먼저 지점별 상영일정페이지의 DOM구조를 분석했다. 분석 후 지점별 구조는 모두 동일한 것을 알아내고, 대표로 메가박스 덕천지점을 기준으로 하여 크롤러 구현을 진행하였다. 메가박스 홈페이지 구조는 CGV, 롯데시네마 홈페이지에 비해 단순한 구조라 크롤러 구현을 쉽게 할 수 있었다. 크게 날짜를 선택하는 부분, 그 밑에 선택된 날짜에 해당하는 상영일정이 나오는 부분이 있다. 먼저 조회가능한 날짜를 수집하였다. 날짜가 나오는 부분의 첫 번째는 오늘이고, 비활성화된 부분의 HTML요소에는 disabled 클래스가 존재했다. 본 서비스는 휴일상관없이 조회일로부터 +4일간의 데이터를 보여주기 때문에 날짜를 하나씩 클릭하는데, disabled가 존재하는 버튼은 클릭하지 않고 넘어가도록 하였다. 클릭 후 아래에 상영일정이 나열되는데, 영화별로 컨테이너가 있고, 그 안에 상영관, 또 그안에 영화타입(2D, 4DX), 시간이 나열되어있다. 구조에 맞게 요소를 뽑아내고 가공하는데 성공하여 이어서 영화관 지점을 이동하여 크롤링하도록 했다. 하지만 이 과정에서 다른날짜를 클릭해도 이전에 크롤링한 정보를 가져오는 문제가 있었다.

[해결방안]

분명 다시봐도 과정에 문제가없어 애를먹던중, 문득 캐시가 생각났다. 해당 크롤러는 브라우저를 화면에 띄우지않고 내부적으로 돌아가는 크롤러(헤드리스모드)라 한번들어간 페이지는 캐시를 통해 빠르게 불러온다. 특정홈페이지는 들어갈때마다 캐시를 초기화해주기도하는데, 메가박스는 그렇지 않았다. puppeteer 캐시를 끄는 옵션을 주어 해결하였다.



### (3) 일관되지 않은 3사 크롤러

크롤러는 한 곳(특정 기기)에서 한번에 3사에 대한 크롤러가 실행되어야 한다. 따라서 팀원 각자가 만든 크롤러들을 하나로 모을 필요가 있었다. 하지만 필자의 팀은 따로 코드 표준을 정하지 않은 상태에서 팀원 각자가 자유롭게 크롤러를 제작했기 때문에 그대로 통합할 경우 중복되는 코드가 발생하고, 실행되는 과정에서 충돌이 있을 수 있었다. 또한 데이터베이스로 전송하는 코드도 제각각이라 누군가가 통합을 해야만 하는 상황이었다.

#### [해결방안]

팀장 이승훈이 팀원 각자가 자유롭게 만든 크롤러를 모아 리팩터링과 통합을 진행했다. 세 크롤러 모두 cheerio모듈과 puppeteer모듈을 사용했기 때문에 중복된 코드는 하나로 통합하고 올바르게 못한 셀렉터 사용을 수정하였다. 그리고 수집한 데이터의 규격을 맞춰 3사 모두 하나의 배열에 결과를 넣도록하고 그 배열을 데이터베이스로 전송하는 식으로 수정하였다.

#### [한계점]

롯데시네마와 메가박스 크롤러는 헤드리스 모드(브라우저를 띄우지 않고 내부적으로 작동하는 모드)에서 제대로 작동하기 때문에 서버에서 사용하는데 문제가 없었지만 CGV 크롤러는 iframe에 대한 처리를 하는 탓에 헤드리스 모드에서 사용이 불가능하다. 따라서 롯데시네마와 메가박스 크롤러만 서버에서 구동하고, CGV 크롤러는 수동으로 서버가 아닌 GUI모드가 있는 컴퓨터(윈도우 노트북 OR 맥북)에서 구동해야하는 한계점이 존재한다.

### (4) 영화진흥위원회 데이터와 크롤링 데이터 비교

통합한 크롤러가 수집한 데이터에는 영화진흥위원회에서 제공한 박스오피스 1위~10위영화 말고도 다양한 영화가 존재한다. 본 서비스는 박스오피스 1위부터 10위까지의 데이터만 이용하여 상영일정을 제공하는데, 이를 위해 크롤러가 수집한 데이터에서 영화진흥위원회에서 제공한 데이터와 동일한 영화만 필터링을 해야했다. 따라서 제목을 기준으로 필터링을 하도록 했는데, 영화진흥위원회에서 제공하는 영화 데이터(제목)와 우리가 크롤링한 데이터들의 제목이 조금이라도 다르면 필터에 걸러지는 문제가 발생했다. 극단적으로 띄어쓰기 하나, 쉼표하나라도 틀리면 다른영화라 인식하여 걸러지게 된다.

#### [해결방안]

롯데시네마와 메가박스의 영화제목은 영화진흥위원회에서 제공하는 영화정보의 제목과 동일한 제목을 사용하여 문제가 없었지만, CGV는 그렇지않았다. 대표적으로 ':' (콜론)문자를 '-'(대시)문자로 바꿔표시하는 경우가 있었다. 예를들어, 진흥위원회에서 제공하는 데이터는 '블랙팬서: 와칸다 포에버'이지만, CGV에서는 '블랙팬서-와칸다 포에버'로 표기한다. 다른 영화도 똑같은 패턴인것으로 보아 CGV는 ':' (콜론, 띄어쓰기)문자를 '-'(대시)문자로 바꿔 표기한다는것을 알아내고 자바스크립트의 replace함수를 이용하여 해결하였다.

[한계점]

콜론 이외의 문자말고도 자체적으로 변형하여 표기하는 경우도 있었는데, 이는 불가피하게 직접 수정하는수밖에 없었다. 뭔가 띄어쓰기를 없애고, 전처리를 거치면 할 수 있을 것 같기도 하지만 지금 상황으로서는 할 수 없다는 것이 조금 아쉬운 부분이다.

(5) TMDB API를 사용하여 데이터 보충

TMDB는 사용자들이 자유롭게 영화정보를 생성하고 수정하는 서비스이다. 영화진흥위원회 오픈API를 통해 가져온정보는 그렇게 풍부하지 않기때문에 본 서비스는 TMDB를 이용하여 포스터이미지, 한줄요약 등 정보를 보충하기로 하였는데, 이 또한 앞서 서술한 문제처럼 제목이 달라 조회가 불가능한 경우가 생겼다. 아무래도 공식적으로 영화정보를 제공하는 사이트가아니고 오픈위키처럼 사용자가 자유롭게 수정할수있다보니 일관되지 않는 문제가 있다.

[해결방안]

거대한 TMDB 사이트의 데이터를 이 서비스를 위해 마음대로 변경할 수는 없다고 생각한다. 따라서 이것은 크롤러단에서 특정 영화제목이 검색되면 TMDB에 검색할때만 replace메서드를 사용하여 영화제목을 TMDB에 조회되는 제목으로 변경한 후 검색하도록 했다. 반 수동인 셈이다.

(6) 로그인 후 쿠키 관리

로그인 과정은 다음과 같다.

사용자가 아이디, 비밀번호를 입력하여 로그인을 요청하면 자바스크립트가 그 정보를 이용하여 사용자 정보를 핸들링하는 user\_handler.php로 요청을 보낸다. php측에선 받은 데이터 중 아이디를 이용하여 데이터베이스에 검색 쿼리를 보내 수행하고, 결과가 있으면 php의 password\_verify 함수를 이용하여 받은 데이터 중 비밀번호와 암호화되어있는 데이터베이스의 내용을 비교하여 비밀번호가 유효한지 검증한다. 만약 유효하다면 JWT를 이용하여 사용자 uid, 닉네임, 역할을 특정키로 암호화하여 해시를 생성한다. JWT는 데이터와 전자서명을 가지고있는데, 전자서명은 서버측에서 개인키로 서명한것으로 클라이언트가 서버에 정보를 요청할때 이 JWT를 함께보내면 서버에서 JWT의 전자서명이 유효한지 검증한다. 서버의 개인키가 유출되지않는한 사용자의 유효성은 안전하다.하지만 JWT는 쿠키에 존재하고, 만약 이 쿠키가 유출되면 사용자의 세션이 안전하지 못하다.

[해결방안]

쿠키는 10분뒤에 자동으로 삭제되도록 설정하였고, 페이지를 이동할때마다 쿠키를 갱신하도록 하였다. 그리고 만약 누군가의 토큰이 유출됐다면 서버의 개인키를 변경하는것으로 부적절한 사용을 막았다.

(7) 데이터베이스 구조변경

처음 계획으로는 3개의 테이블을 구축하여 운영하기로 했었다.

theater_info	<b>code</b>	<b>name</b>	<b>location</b>		
	식별코드	지점이름	위치		
user_info	<b>uid</b>	<b>id</b>	<b>pw</b>	<b>nickname</b>	
	고유번호	아이디	비밀번호	닉네임	
post_data	<b>postID</b>	<b>title</b>	<b>date</b>	<b>user</b>	<b>contents</b>
	게시물 아이디	제목	작성날짜		

기존에는 영화정보와 상영일정을 따로 저장해두지않고 사용자가 요청하는 즉시 크롤링하여 제공하기로했었는데, 생각보다 과정이 오래걸려 두 데이터를 일정시간마다 크롤링하여 저장하기로 하였다.

따라서 아래 두개의 테이블을 추가하여 해결하였다.

movie_data	<b>id</b>	<b>name</b>	<b>tagline</b>	<b>overview</b>	<b>release_date</b>
	TMDB 영화아이디	영화이름	한줄요약	줄거리	개봉일
	<b>audiAcc</b>	<b>salesAcc</b>	<b>poster_path</b>	<b>update</b>	
	총 관람객 수	총 매출	포스터 이미지경로	업데이트 날짜	
schedule_date	<b>movie_ name</b>	<b>cinema_ name</b>	<b>cinema_ location</b>	<b>date</b>	<b>time</b>
	영화이름	영화관 이름	영화관지점	상영날짜	상영시간
	<b>morning</b>	<b>movie_ type</b>	<b>theater_ name</b>	<b>remain_ chair</b>	<b>all_chair</b>
	조조영화 여부	영화타입	상영관이름	잔여좌석	전체좌석

그리고 게시판정보라고 모호하게 정해서 공지사항과 한줄평 데이터를 분리할 필요가 있었다. 따라서 아래 테이블을 추가했다.

notice_data	<b>noticeID</b>	<b>title</b>	<b>content</b>
	공지아이디	제목	내용
	<b>date</b>	<b>category</b>	<b>view</b>
	작성일	분류	조회수

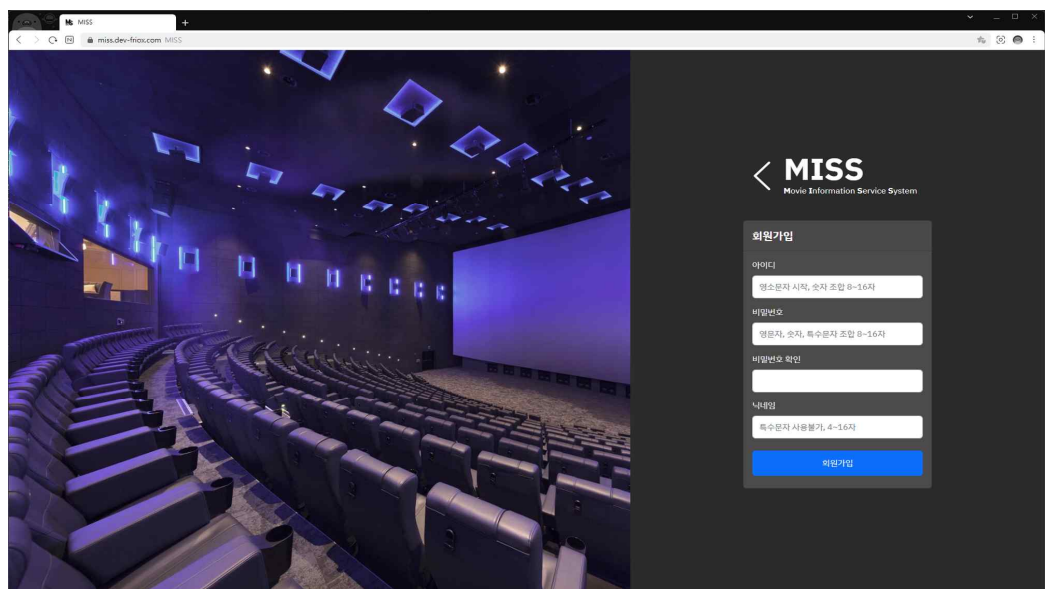
## 나) 프론트엔드

### (1) 회원가입 갈아엎은 이유

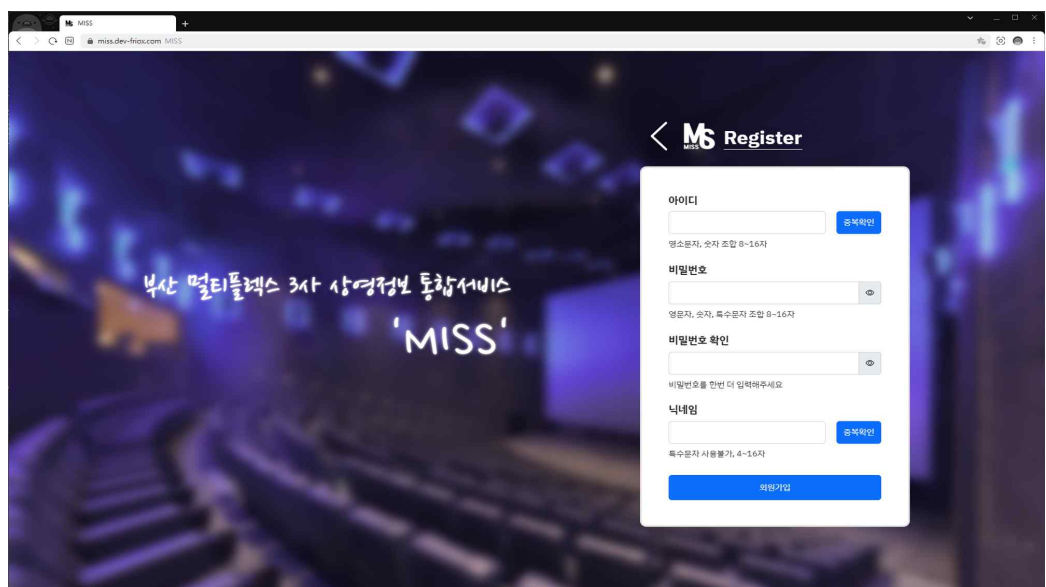
기존 회원가입 페이지는 좌측에 영화관 이미지를 삽입하고, 우측에 어두운 회색 공간을 배치하여 그 안에 회원가입 폼을 배치했었다. 회원가입 페이지는 홈페이지 메인 다음으로 사용자에게 보여지는 페이지인만큼, 보여지는 첫 인상이 매우 중요하다. 투명하지않은 배경때문에 답답한 느낌이 들고, 아이디, 닉네임 중복확인 버튼도 없다. 그리고 무엇보다 이 서비스가 무엇을 하는 서비스인지 설명조차 없어 사용자에게 좋지못한 사용자 경험을 줄 것이다.

### [해결방안]

#### 변경 전



#### [변경 후]



답답해보이는 뒷배경을 지우고 전체적으로 블러처리를 하여 개방감을 주었다. 좌측 빈공간에 서비스를 설명할 수 있는 한줄문구와 서비스 이름을 넣었고, 회원가입 폼에 아이디 중복확인, 닉네임 중복확인을 넣었다. 또한 입력한 비밀번호를 볼 수 있게하는 토글버튼을 배치하였고, 폼을 입력하고 포커스가 이동하면 즉시 유효한 아이디, 비밀번호, 닉네임인지 표시되게 하였다.

## (2) 메인화면

사이트에 처음 들어오면 현재 상영 영화 중인 박스오피스 1위부터 10위까지 영화들을 가져온다. 영화진흥위원회 주간별 박스오피스 순위가 변경되면 그에 맞춰서 메인화면의 영화와 포스터가 바뀐다. 상영 영화 포스터 밑에 더 보기 화살표를 디자인 함으로써 밑에 내용이 더 있음을 알려준다. 포스터 하단 부분에는 공지 사항, 한줄평을 보여준다. 메인 배너에 공지 사항 항목이 따로 있지만 메인화면 하단 부분에도 간단하게 볼 수 있도록 구현하였고. 또한 하단 부분에 공지 사항 우측에 한줄평을 볼 수 있도록 디자인을 구상했다. 현재 상영 중인 영화를 들어가서 영화의 한줄평을 볼 수 있지만 메인화면에서 최근에 올라온 한줄평을 바로 볼 수 있도록 구현하였다.

## (3) 회원 정보관리

회원가입 후 로그인 상태에서 닉네임 옆에 있는 아이콘을 클릭하는 경우 회원 정보 관리 창이 띄워진다. 상단에서부터 회원 정보 관리, 기존 비밀번호, 새로운 비밀번호, 새로운 비밀번호 확인 순으로 나열하기로 하였고 회원 정보 관리 창에 내용을 전부 입력한 후 기존 비밀번호를 변경할 수 있도록 비밀번호 변경 버튼을 구현하였다. 또한 비밀번호 변경 버튼 하단에 회원을 탈퇴할 수 있는 회원 탈퇴 버튼도 구현하였다.

## (4) 로그인

3사의 영화사 로그인 페이지를 확인 후 메가박스의 로그인 창을 참고하기로 하였다. 로그인 텍스트 및 아이콘을 누르면 현재 페이지에 로그인 창이 띄워지게 만들기로 하였고 창이 나오면 상단에서부터 로그인, ID, PW 순으로 나열했다. 로그인 창에서는 회원가입 페이지로 갈 수 있도록 회원가입 버튼을 구현하였고, ID, PW를 입력하고 로그인하는 로그인 버튼도 구현하였다.

# 3. 프로젝트 수행결과

## 가. 수행결과

### 1) 프로젝트 사전조사

#### 가) 고려해야할 내용

영화의 정보를 어떻게 가져올까 고민하던 중 크롤링 기법을 사용하여 데이터를 가져오기로 결정하였고, 필요로 하는 데이터를 가져오기 위해서는 각 영화사 홈페이지의 내부 DOM 구조를 파악하여야 했다. 또한 DOM 구조 파악을 위한 방법과 크롤링을 위한 적절한 플랫폼이나 프로그래밍 언어를 고려해보아야 했다

#### 나) 수행결과

DOM 구조 파악은 크롬에서 지원하는 개발자 도구를 사용하기로 했다. 개발자 도구의 대표적 기능으로는 웹페이지 소스보기, 엘리먼트별 CSS보기, 콘솔 기능 등이 있는데, 이 중 웹 페이지 소스 보기, 엘리먼트 선택 기능을 적극 활용하기로 했다. 위 기능들로 우리가 필요로 하는 데이터를 담고 있는 DOM 정보의 선택자를 가져올 수 있었기 때문에 수월한 크롤링이 가능하였다. 크롤링을 수행하기 위한 플랫폼과 프로그래밍 언어는 팀장 이승훈의 기술 스택인 Node.js와 자바스크립트 언어를 채택하여 사용했다

### 2) 프로젝트 UI 디자인

#### 가) 고려해야할 내용

MISS 홈페이지에 접속했을 때 전반적인 UI 디자인에 대한 문제, 어떻게 하면 가공한 데이터를 표시할 때 사용자들에게 가독성이 좋다고 느껴질지 고려하였다.

#### 나) 수행결과

필자의 팀이 제공하는 서비스 또한 영화와 관련된 서비스이기 때문에 기존에 존재하는 영화 사이트를 참고하여 팀원의 과반수 동의로 가독성이 좋다고 생각한 홈페이지 UI를 참고하기로 하였다.

### 3) 프로젝트 환경구축

#### 가) 고려해야할 내용

프로젝트 환경을 구축할 때 초기에는 학교 실습컴퓨터에 서버를 구동시켜 웹과 데이터베이스를 구현하였다. 이러한 방식의 문제점은 학교의 서버이기 때문에 24시간 구동이 불가능하고, 외부에서 접근할 때 학교의 NAC에 차단되거나 포트 개방 불가능 등의 문제로 외부에서 작업하기 힘들어 이를 해결하기 위한 방안을 고려했다.

#### 나) 수행결과

이러한 문제를 해결하기 위해 필자의 팀은 팀장 이승훈의 개인 서버를 사용하여 프로젝트를 진행하기로 하였다. 또한 외부에서 쉽게 접속하기 위해 서버 도메인에 가상호스트를 연결하여 miss.dev-friox.com으로 바로 접속 할 수 있도록 하였다.

### 4) 프로젝트 프론트엔드, 백엔드 작업

#### 가) 고려해야할 내용

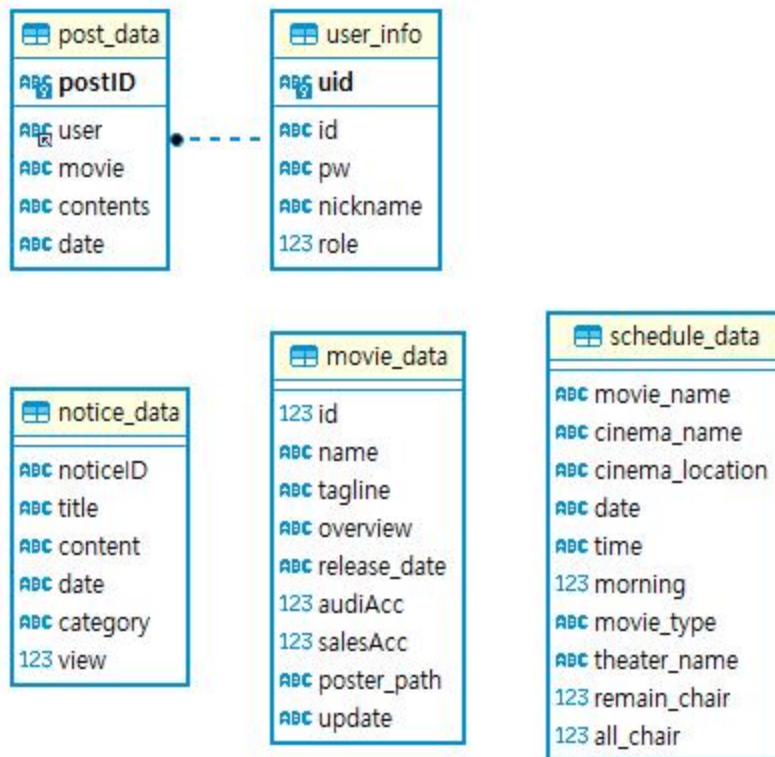
프론트엔드를 구현할 때 CSS와 HTML만으로는 어떠한 분기, 예를 들어 로그인에 실패하였을 경우 '로그인에 실패했습니다' 라는 경우를 구현 할 수 없는 문제와 가공하여 가져온 데이터들을 표시할 수 없는 문제가 있다. 이러한 문제를 해결하기 프론트엔드를 구축할 때 어떠한 프레임워크와 언어를 추가로 사용할지 고려하였다. 또한 백엔드 측면으로는 node.js에서 작동하는 자바스크립트 기반 크롤러가 수집한 데이터를 어떻게 데이터베이스에 삽입할지 생각해보아야 한다.

## 나) 수행결과

프론트엔드측에서 받아온 데이터를 처리하여 분기하기위해 자바스크립트를 적극적으로 활용하기로 하였다. 또한 알림창을 보기좋게 표시하기위해 SweetAlert2 라이브러리를 사용하기로 했다. 크롤러가 수집한 데이터를 데이터베이스로 보내는 방법은 크게 두가지가 있었는데, 하나는 node.js에서 수집한 데이터를 axios를 통해 데이터베이스에 데이터를 삽입하는 php로 요청을 보내는 방법과 자바스크립트상에서 바로 데이터베이스에 접근하여 데이터를 삽입하는 방법이 있었다. 전자의 방법은 데이터가 과도하게 많을경우 데이터가 손실되어 전달된다는 문제가있어 두번째 방법으로 결정하였다.

## 나. 프로젝트 성과물

### 1) 데이터베이스 ERD







## 2) 기능명세

페이지	주요 기능	업무 대분류	업무 소분류	세부 항목	코멘트
전체영화	박스오피스 영화 제공	X	X	전체영화	영화진흥위원회에서 제공하는 박스오피스 순위 1위부터 10위까지의 정보입니다. 문구 표시  1, 2위의 포스터는 크게 보여주고 나열 나머지 3위~10위 영화들은 작게 나열
영화 상세정보 페이지	영화 상세정보 제공 & 상영 일정 & 한줄평	영화 상세정보	선택한 영화의 상세정보	영화 제목	영화 제목 텍스트 표시
				개봉일	영화 개봉일 텍스트 표시
				누적 관람객	아이콘 우측에 누적 관람객 몇 명인지 표시
				총 매출액	아이콘 우측에 총매출액 원으로 표시
				핵심 문구	영화의 핵심 문구 텍스트 표시
				줄거리	영화의 줄거리 텍스트 표시
		상영 일정	필터	영화사	- 각 영화사를 선택하기 전은 체크박스 비활성화 - 사용자가 선택한 영화의 영화사를 선택
				지점(CGV)	선택한 영화사가 CGV면 부산지역에 있는 CGV 지점들 활성화
				지점(롯데시네마)	선택한 영화사가 롯데시네마면 부산지역에 있는 롯데시네마 지점들 활성화
				지점(메가박스)	선택한 영화사가 메가박스면 부산지역에 있는 메가박스 지점들 활성화
				정렬 기준	상영시간, 잔여 좌석, 상영관 기준으로 선택 가능
				정렬 방법	오름차순, 내림차순 정렬 선택 가능
				재설정 & 적용	- 필터에 있는 내용을 선택한 후 적용 버튼을 누르면 필터 기준에 따라 상영시간, 조조 여부, 영화사, 지점, 영화 종류, 상영관, 잔여 좌석 정보, 바로가기 예매 버튼 제공  - 선택한 영화의 요일 선택 후 아무것도 없는 경우 '검색된 항목이 없습니다' 텍스트 표시 - 필터에 있는 체크박스를 아무것도 선택 안 한 경우 '영화관을 하나 이상 선택해주세요' 알림창 - 필터에 있는 체크박스를 전부 선택한 후 적용 버튼 클릭 시 '잠시만 기다려주세요' 알림창 - 필터에 있는 내용을 선택한 후 재설정 버튼을 누르면 선택한 내용 전부 초기화 - 바로가기 예매 버튼 클릭 시 각 영화사 예매 사이트로 방문
		한줄평	한줄평	한줄평 쓰기	- 한줄평에 아무 내용도 없는 경우 '내용을 확인해주세요' 알림창  - 한줄평에 내용을 쓴 경우 '이대로 등록할까요?' 알림창 - '정상적으로 등록되었습니다' 알림창 - 한줄평 테러를 방지하기 위해 글자 수, 줄바꿈 수를 제한 후 텍스트 표시
				한줄평 수정	-게시한 한줄평의 내용을 수정 - 변경사항이 없는 경우 '변경사항이 없어요' 알림창 - 변경사항이 있는 경우 '이렇게 수정할까요?' 알림창 - '정상적으로 수정되었습니다' 알림창
				한줄평 삭제	'삭제하면 되돌릴 수 없어요' 알림창 '정상적으로 삭제되었습니다' 알림창

페이지 & 팝업창	주요 기능	업무 중분류	업무 소분류	세부 항목	코멘트
회원가입 & 로그인	회원 관리	회원가입	회원가입	아이디	아이디 형식에 맞는 경우 '형식에 맞는 아이디입니다.' 텍스트 표시 아이디 형식에 맞지 않는 경우 '아이디가 형식에 맞지 않습니다' 텍스트 표시 아이디 중복확인 버튼 클릭 시 '사용가능한 아이디입니다' 알림창
				비밀번호	비밀번호 형식에 맞는 경우 '형식에 맞는 비밀번호입니다.' 텍스트 표시 비밀번호 형식에 맞지 않는 경우 '비밀번호가 형식에 맞지 않습니다' 텍스트 표시
				비밀번호 확인	입력한 비밀번호가 일치하는 경우 '비밀번호가 일치합니다' 텍스트 표시 입력한 비밀번호가 일치하지 않는 경우 '비밀번호가 일치하지 않습니다' 텍스트 표시
				닉네임	- 닉네임이 형식에 맞는 경우 '사용가능한 닉네임입니다.' 텍스트 표시 - 닉네임 중복확인 버튼 클릭 시 '사용가능한 닉네임입니다' 알림창 - 회원가입 세부 항목이 전부 형식에 맞는 경우 '회원가입 성공! 잠시 후 메인으로 이동합니다' 알림창  - 닉네임이 형식에 맞지 않는 경우 '닉네임이 형식에 맞지 않습니다.' 텍스트 표시 - 닉네임 중복확인 버튼 클릭 후 '닉네임을 확인해주세요' 알림창
		로그인	로그인 화면	ID	'가입되어 있지 않거나, 정보가 일치하지 않습니다.' 알림창
				PW	'가입되어 있지 않거나, 정보가 일치하지 않습니다.' 알림창
				로그인	로그인 클릭 시 ID, PW 팝업창 띄우기 로그인 버튼 클릭시 로그인 중 알림창 로그인 후 메인 배너 오른쪽에 닉네임, 역할 제공
				로그아웃	로그아웃 아이콘 클릭 시 로그아웃 하시겠습니까? 알림창
		로그인	비밀번호 변경	비밀번호 변경	- 아무것도 입력하지 않는 경우 '빈칸이 있어요' 알림창 - 회원 정보에 없는 ID, PW를 입력한 경우 '가입되어 있지 않거나 정보가 일치하지 않습니다' 알림창  - 기존 비밀번호가 형식과 맞는 경우 '형식에 맞는 비밀번호입니다.' 텍스트 표시 - 기존 비밀번호가 형식과 다른 경우 '비밀번호가 형식에 맞지 않습니다.' 텍스트 표시  - 새로운 비밀번호가 형식과 맞는 경우 '사용가능한 비밀번호입니다.' , 라는 텍스트 표시 - 새로운 비밀번호가 형식과 다른 경우 '비밀번호가 형식에 맞지 않습니다.' 텍스트 표시 - 새로운 비밀번호 확인이 맞는 경우 '비밀번호가 일치합니다.' , 라는 텍스트 표시 - 새로운 비밀번호가 확인이 다른 경우 '비밀번호가 일치하지 않습니다' 알림창
					- 회원 탈퇴 버튼 클릭 시 '정말 탈퇴하실건가요?... 사용자 정보와 작성하신 모든 한줄평이 사라집니다' 알림창 - 회원 탈퇴 시 '회원 탈퇴 완료' 알림창

페이지	주요 기능	업무 대분류	업무 중분류	업무 소분류	세부항목	코멘트
메인 화면	현재 상영 영화 포스터 제공 & 다른 페이지 이동	메인 화면	현재 상영 영화	현재 상영 영화 & 전체영화	박스오피스 영화 포스터 선택	'영화정보 조회 중 잠시만 기다려주세요' 알림창
					전체영화 보기	
			메인 배너	메인 배너	CGV 로고	CGV 사이트 방문
					롯데시네마 로고	롯데시네마 사이트 방문
					메가박스 로고	메가박스 사이트 방문
					MISS 로고	MISS 메인 홈페이지 방문
					공지사항	각 영화사의 서비스 문제 정보 제공 '조회 중 잠시만 기다려주세요!' 알림창
					전체영화	박스오피스 10개의 영화 데이터 나열 '영화 정보 조회 중 잠시만 기다려주세요!' 알림창
					로그인	로그인 클릭 시 로그인 팝업창 띄우기
공지사항	각 영화사 서비스 문제 알림	공지사항	글쓰기	분류	공지	글 분류가 공지로 표시
					이벤트	글 분류가 이벤트로 표시
					직접입력	글 분류가 직접 입력한 글로 표시
				제목 & 내용	취소	취소 버튼을 누르면 글쓰기 팝업창 닫기
					초기화	초기화 버튼을 누르면 제목 & 내용 초기화
					글쓰기	<ul style="list-style-type: none"> <li>- 글쓰기 버튼을 클릭할 시 공지사항 작성 작성자, 제목, 내용 팝업창 띄우기</li> <li>- 제목이 비었다면 '제목이 비었어요.' 알림창</li> <li>- 내용이 비었다면 '내용이 비었어요.' 알림창</li> <li>- 제목, 내용 입력 후 글쓰기 버튼 클릭 시 '작성한 내용을 다시 한번 확인해주세요' 알림창</li> <li>- 글쓰기가 완료되었다면 '정상적으로 작성했어요' 알림창</li> </ul>
				기타	공지사항 글 확인	<ul style="list-style-type: none"> <li>- 잠시만 기다려주세요! 알림창</li> <li>- 삭제 버튼 클릭 시 '삭제 후 복구할 수 없어요' 알림창</li> <li>- 삭제 완료되었다면 '정상적으로 삭제되었어요' 알림창</li> <li>- 수정 버튼 클릭 시 내용 입력 후 '작성한 내용을 다시 한번 확인해주세요' 알림창</li> <li>- 내용이 수정되었다면 '정상적으로 수정되었어요' 알림창</li> <li>- 닫기 버튼 클릭 시 팝업창 닫기</li> </ul>

### 3) 사용자 피드백

페이지	피드백 내용	피드백 반영 여부
메인 홈페이지	회원정보 관리 아이콘이 눌러보기 전까지는 어떤 기능인지 알 수 없다. 한 눈에 알아볼 수 있는 아이콘으로 바뀌었으면 좋겠다	반영 가능
반영 결과		
<div style="display: flex; align-items: center;">  <div style="margin-left: 20px;"> <p>◀ 피드백 전 회원 정보 관리 아이콘</p> </div> </div> <div style="display: flex; align-items: center; margin-top: 10px;">  <div style="margin-left: 20px;"> <p>◀ 피드백 후 회원 정보 관리 아이콘</p> </div> </div>		
사용자들의 평가	<p>타 사이트를 둘러본 경험으로는 사람형 아이콘이 개인정보 또는 회원 정보의 의미를 내포한다고 느꼈기 때문에 아이콘을 사람형으로 바꿈으로써 위 아이콘이 회원정보를 나타낸다는 것을 알 수 있었다</p>	

페이지	피드백 내용	피드백 반영 여부
전체 영화	전체 영화가 1위에서 10위 까지만 있는게 아쉽다. 더 많은 종류의 영화가 있었으면 좋겠다	반영 불가능
반영 결과		
<p>필자의 팀이 구축한 서비스는 영화진흥위원회에서 제공하는 박스오피스 데이터를 받아와 해당 영화에 대한 상영 일정을 제공하는 것인데, 위 기관에서 제공하는 데이터가 1위부터 10까지 밖에 없기 때문에 많은 종류의 영화 데이터를 가져올 수 없다.</p>		
사용자들의 평가	—	

페이지	피드백 내용	피드백 반영 여부																											
공지사항	본인은 글을 읽을 때 분류를 먼저 확인하고 글을 읽는데 공지사항의 분류가 오른쪽에 있으니 눈 동선이 어지럽다. 분류가 글 제목 앞에 있었으면 좋겠다	반영 가능																											
반영 결과																													
<div>공지사항</div> <table><tr><th>ID</th><th>제목</th><th>날짜</th><th>분류</th><th>조회수</th></tr><tr><td>2</td><td>이벤트 게시물 작성 테스트</td><td>2022.12.13</td><td>이벤트</td><td>6</td></tr><tr><td>1</td><td>공지 게시물 작성 테스트</td><td>2022.12.13</td><td>공지</td><td>7</td></tr></table> <div>▲피드백 전 공지사항 게시물.</div> <div>공지사항</div> <table><tr><th>ID</th><th>제목</th><th>날짜</th><th>조회수</th></tr><tr><td>2</td><td>[이벤트] 이벤트 게시물 작성 테스트</td><td>2022.12.13</td><td>7</td></tr><tr><td>1</td><td>[공지] 공지 게시물 작성 테스트</td><td>2022.12.13</td><td>8</td></tr></table> <div>▲피드백 후 공지사항 게시물</div>			ID	제목	날짜	분류	조회수	2	이벤트 게시물 작성 테스트	2022.12.13	이벤트	6	1	공지 게시물 작성 테스트	2022.12.13	공지	7	ID	제목	날짜	조회수	2	[이벤트] 이벤트 게시물 작성 테스트	2022.12.13	7	1	[공지] 공지 게시물 작성 테스트	2022.12.13	8
ID	제목	날짜	분류	조회수																									
2	이벤트 게시물 작성 테스트	2022.12.13	이벤트	6																									
1	공지 게시물 작성 테스트	2022.12.13	공지	7																									
ID	제목	날짜	조회수																										
2	[이벤트] 이벤트 게시물 작성 테스트	2022.12.13	7																										
1	[공지] 공지 게시물 작성 테스트	2022.12.13	8																										
사용자들의 평가	공지사항의 분류와 제목이 한 눈에 들어와 사용자가 필요로 하는 정보인지 알아보기 수월해졌다.																												

페이지	피드백 내용	피드백 반영 여부
로그인	로그인 정보를 전부 입력한 후 엔터로 로그인이 되었으면 좋겠다. 마우스로 클릭하는게 번거롭다	반영 가능
반영 결과		
 <pre>mainPWInputEl.onkeydown = (e) =&gt; {   if (e.code == 'Enter') {     e.preventDefault();     loginAction()   } }</pre> <p>▲ 위 코드를 추가함으로써 엔터키를 입력했을 경우 로그인이 가능하게 수정하였다</p>		
사용자들의 평가	마우스 클릭이 번거로웠는데 개선되어 상당히 편안해졌다.	

페이지	피드백 내용	피드백 반영 여부
로그인	로그인 페이지에서 아이디, 비밀번호 찾기 또는 비밀번호 변경 기능이 있으면 좋을 것 같다,	<b>반영 불가능</b>
반영 결과		
<p>회원가입을 할 때 사용자를 식별할 수 있는 개인정보, 예를 들어 전화번호 또는 이메일 등의 내용을 기재하지 않았기 때문에 아이디와 비밀번호를 찾는 것은 불가능하다.</p>		
사용자들의 평가	—	

페이지	피드백 내용	피드백 반영 여부
회원가입	회원가입 할 때 제약이 너무 많다. 완화 해 줬으면 좋겠다.	<b>반영 가능</b>
반영 결과		
<div> <div> <b>닉네임</b>  <input type="text"/>  <small>특수문자 사용불가, 4~16자</small> </div> <div> <b>중복확인</b> </div> </div> <p>◀ 피드백 전 닉네임 설정 제약</p> <div> <div> <b>닉네임</b>  <input type="text"/>  <small>특수문자 사용불가, 1~16자</small> </div> <div> <b>중복확인</b> </div> </div> <p>◀ 피드백 후 변경된 닉네임 설정 제약</p>		
사용자들의 평가	<p>회원가입 할 때 닉네임 제약을 딱히 생각하지 않아도 된다는 점이 개선되었다고 느껴진다</p>	

페이지	피드백 내용	피드백 반영 여부
상영 정보	조건을 다 선택해두고 예매를 클릭하여 영화사 홈페이지로 이동하면 클릭 해 놓았던 조건이 다 초기화 된다.	<b>반영 불가능</b>
반영 결과		
<p>만약 영화사의 예매 홈페이지가 URL에서 정보를 가져오는 형식이고, URL이 지정된 형식에 따라 바뀌어 사용자가 선택한 조건을 읽을 수 있다면 정보를 그대로 예매 사이트에 반영할 수 있지만, 영화사 홈페이지 구조상 URL에서 정보를 가져오는 시스템이 아니기 때문에 필자의 팀이 제공하는 사이트에서 선택한 정보를 해당 홈페이지에 전달한다 하여도 예매 사이트에 반영이 불가능하다.</p> <p>따라서 사용자가 MISS 사이트에서 선택한 조건을 예매 사이트에 표시 할 수 없다.</p>		
사용자들의 평가	—	

페이지	피드백 내용	피드백 반영 여부
상영 정보	상영 정보를 선택 할 때 날짜 선택 창 잘 보이지 않는다.	<b>반영 가능</b>
반영 결과		
 <p>◀ 피드백 전 날짜 선택 창. 확실히 창이 눈에 띄지 않는다</p>  <p>◀ 피드백 후 날짜 선택 창. 창이 있다는 것이 눈에 들어온다</p>		
사용자들의 평가	테두리와 아이콘을 추가함으로써 날짜 선택 창의 시인성이 좋아졌다	

### III. 결론

#### 1. 프로젝트 달성도

[표 3] 설계 목표의 중요도 및 달성도

목표	중요도(%)	달성도(%)	수행내용
회원가입	20%	100%	1. 아이디, 비밀번호, 닉네임 정규식이용 형식검사 (O) 2. 아이디, 닉네임 중복검사 (O) 3. 입력한 비밀번호 확인 (O)
공지사항	10%	80%	1. 데이터베이스에 저장된 공지사항 수집 후 디스플레이 (O) 2. 공지사항 수정 및 삭제 (O) 3. 공지사항 팝업으로 구현 (O) 4. 공지사항 작성할 때 글씨크기, 색상 등 스타일 적용 (X)
영화정보 통합제공	60%	60%	1. 영화정보 크롤링(영화진흥위원회 오픈API) (O) 2. 박스오피스 1~10위 데이터를 가져오기 (O) 3. 멀티플렉스 3사에 대한 헤드리스 모드 크롤러 제작 (X) 4. 영화정보 필터링 (X) 5. 상영일정 필터링 (O)
한줄평 커뮤니티	20%	83%	1. 한줄평 작성 (O) 2. 한줄평 수정 및 삭제 (O) 3. 자신이 쓴 한줄평 구분 (O) 4. 한줄평 페이지 세분화 (X) 5. 관리자가 모든 한줄평을 삭제할 수 있게 (O) 6. 글자수, 줄바꿈 제한 (O)
합계	100%	80.75%	

#### 2. 프로젝트 결과 논의

##### 가. 장점

전체 영화를 순위권으로 보아 관람 영화 선택 용이하고 3사의 영화 정보 통합과 필터 기능 추가로 가독성이 보다 향상되었다. 한줄평 커뮤니티 활성화로 영화의 정보나 관람평 데이터를 수집 가능하고, 공지사항 게시판으로 사용자에게 주요 정보 전달이 가능하다.

##### 나. 단점

박스오피스 1~10위권 영화만 존재하여 사용자가 원하는 영화가 없을 가능성 있다. 또한 박스오피스 1~10위권 영화중 특정 영화를 3사에서 상영하지 않으면 정보가 나오지 않는다. 검색 기능을 추가하지 않아 사용자가 원하는 영화를 찾는데 불편을 겪을 수 있다.



### 3. 프로젝트 결과의 활용방안

추후 3사(CGV, 롯데시네마, 메가박스)를 제외한 전국 단위 멀티플렉스 영화사가 입점할 경우 다른 영화 정보 검색 서비스의 활성화에 필자의 팀이 미리 구축해놓은 틀인 MISS 서비스가 활용되어 새로운 상영 정보 통합 제공 서비스가 개발 될 수 있다.

### 4. 프로젝트 결과의 기대성과

기술적 측면	본 프로젝트에서 필요로 하는 영화 데이터 수집을 위해 각 3사의 데이터를 가져 올 수 있도록 하는 크롤러를 만들었다.
경제적 측면	3사의 데이터를 통합하여 제공하는 서비스는 지금까지 존재하지 않았기 때문에 본 프로젝트에서 구현한 서비스를 사용함으로써 사람들이 자신이 필요한 조건에 맞는 영화 정보를 찾는 시간 절약할 수 있다.
사회적 측면	본 프로젝트에서 구현한 서비스를 기반으로 한 다양한 정보 통합 서비스의 활성화를 기대할 수 있다.

## IV. 참고문헌

[1] 영화관입장권통합전산망(KOBIS) 일별 박스오피스, KOBIS 영화관 입장권 통합 전산망, 게시일자 없음, 11월 20일 접속

<https://www.kobis.or.kr/kobis/business/stat/boxs/findDailyBoxOfficeList.do>

[2] TMDb(The Movie Database), TMDb, 게시일자 없음, 11월 20일 접속

<https://www.themoviedb.org/?language=ko>

[3] 부트스트랩(프론트엔드 프레임워크), Bootstrap · 세계에서 가장 인기있는 HTML, CSS, JS 라이브러리, 게시일자 없음, 2022년 11월 20일 접속

<https://getbootstrap.kr/>

[4] Puppeteer 공식 홈페이지, Puppeteer, 게시일자 없음, 접속일 2022년 11월 28일

<https://pptr.dev/>

[5] Cheerio 공식 홈페이지, Cheerio, 2022년 11월 28일

<https://cheerio.js.org/>

[6] "Puppeteer로 크롤러 만들기 - 준비", 깃허브 (yangeok), 2019년 9월 9일 수정, 2020년 11월 28일 접속

<https://yangeok.github.io/node.js/2019/09/09/puppeteer-crawler-pre.html>

[7] Node.js|초간단 API 서버 만들기 - 1/2, 티스토리 by 아기상어 뚜루루뚜루, 2021년 7월 6일 수정, 2022년 11월 28일 접속

<https://kitty-geno.tistory.com/62?category=960541>

<https://kitty-geno.tistory.com/63?category=960541>

[8] [Node.js 강좌] 13. 웹 스크래핑 - Cheerio 모듈, 티스토리 by 양다, 2016년 2월 2일 수정, 2022년 11월 28일 접속

<https://dalkomit.tistory.com/104>

[9] Puppeteer를 이용한 웹 크롤링, 10분코딩, 2018년 12월 15일 수정, 2022년 11월 28일 접속(유튜브 강의)

[https://youtu.be/Jt\\_Ryy83hp4](https://youtu.be/Jt_Ryy83hp4)

[10] 문법과 자료형 - JavaScript - MDN Web Docs, 모질라 공식 웹 사이트, 2022년 11월, 27일 수정, 2022년 11월 28일 접속

[https://developer.mozilla.org/ko/docs/Web/JavaScript/Guide/Grammar\\_and\\_types](https://developer.mozilla.org/ko/docs/Web/JavaScript/Guide/Grammar_and_types)

[11] Puppeteer를 이용한 웹 크롤링 해보기 (예제 1) - velog, jinuku, 2020년 6월 8일 수정, 2022년 11월 29일 접속

<https://velog.io/@jeffyoung/Puppeteer%EB%A1%9C-%ED%95%99%EA%B5%90-%EA%B3%B5%EC%A7%80-%EA%B8%80-%ED%81%AC%EB%A1%A4%EB%A7%81-%ED%95%98%EA%B8%B0>

[12] JWT(JSON Web Token)의 개념부터 구현까지 알아보기, 개인 블로그 by 정상우, 2021년 7월 29일 수정, 2022년 12월 6일 접속

<https://pronist.dev/143>

[13] PHP와 JWT (JSON Web Tokens)를 이용한 예제, 티스토리 by 개발왕자, 2021년 5월 17일 수정, 2022년 12월 7일 접속

<https://sysdocu.tistory.com/1651>

[14] nodejs express mysql 콜백지옥 탈출하기2(mysql2 async await), 티스토리 by 안양사람, 2020년 5월 27일 수정, 2022년 12월 3일 접속

<https://ms3864.tistory.com/27>