

Business presentation

Java 기반 영화 예 매 사이트 구현 프 로젝트

CGV 벤치마킹을 통한 실제 웹 서비스 개발 경험



목차

Java와 JSP/Servlet 기반 영화 예매 사이트 구현 프로젝트 소개

프로젝트 개요

- CGV 벤치마킹
- 영화 목록 조회
- 장르/제목 검색
- 상영 시간표 확인

개발 배경 및 목표

- Java와 MVC 패턴 적용
- 사용자 친화적 플랫폼
- 개발 기간 및 배포 정보

기술 스택

- Java, JSP, Servlet
- HTML, CSS
- Oracle DB
- Tomcat, AWS EC2

주요 기능

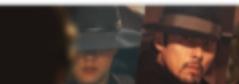
- 영화 목록 조회
- 영화 검색 기능
- 상영 시간표 확인

ERD 및 데이터베이스 구조

- 핵심 테이블 구조
- 테이블 간 관계
- 주요 필드 설명

프로젝트 성과 및 향후 계획

- 개발 과정 및 역할
- 프로젝트 성과
- 향후 개선 방향



로그인

회원가입

공지사항

영화

상영시간표

영화예약

영화 검색



무파사:라이온 킹

'라이온 킹' 탄생 30주년 기념작

의로운 고아에서 전설적인 왕으로 거듭난 '무파사'의 숨겨진 이야기가 베일을 벗는다!

상세보기



무비차트



상영예정작

공지사항		고객센터	
제목	등록일	고객센터	
HGV서비스 약관 개정	2025-01-02	1544-1122	고객센터 운영시간 (평일 09:00~18:00)
4DX 레드 카드 서비스 종료 안내 (~24/1--)	2025-01-02		업무시간 외 자동응답 안내 가능합니다.
개인(위치)정보처리방침 개정 공지 (24.09.--)	2025-01-02		

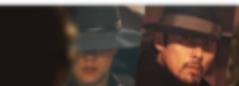
프로젝트 개요

Java와 JSP/Servlet 기반 영화 예매 사이트

- 영화 목록 조회 기능 구현
- 장르/제목 검색 기능 개발
- 상영 시간표 확인 시스템 구축

CGV 웹사이트 벤치마킹

- 사용자 중심 UI/UX 설계
- 영화 정보 노출 방식 참고
- 실제 예매 흐름 반영
- 필터 UI 구현
- 상영 시간 정보 구조화
- 사용자 편의성 극대화



로그인

회원가입

공지사항

영화

상영시간표

영화예약

영화 검색



무파사:라이온 킹

'라이온 킹' 탄생 30주년 기념작
의로운 고아에서 전설적인 왕으로 거듭난 '무파사'의 숨겨진 이야기가 베일을 벗는다!

상세보기



무비차트



상영예정작

공지사항	
제목	등록일
HGV서비스 약관 개정	2025-01-02
4DX 레드 카드 서비스 종료 안내 (~24/1--)	2025-01-02
개인(위치)정보처리방침 개정 공지 (24.09.--)	2025-01-02

고객센터

1544-1122

고객센터 운영시간 (평일 09:00~18:00)
업무시간 외 자동응답 인내 가능합니다.

개발 배경 및 구현 목표

개발 목표

Java 언어를 활용하여 MVC 패턴을 적용시킨 영화 예매 사이트 구현

개발 기간

2024.12.05 ~ 2025.01.10

서버 주소

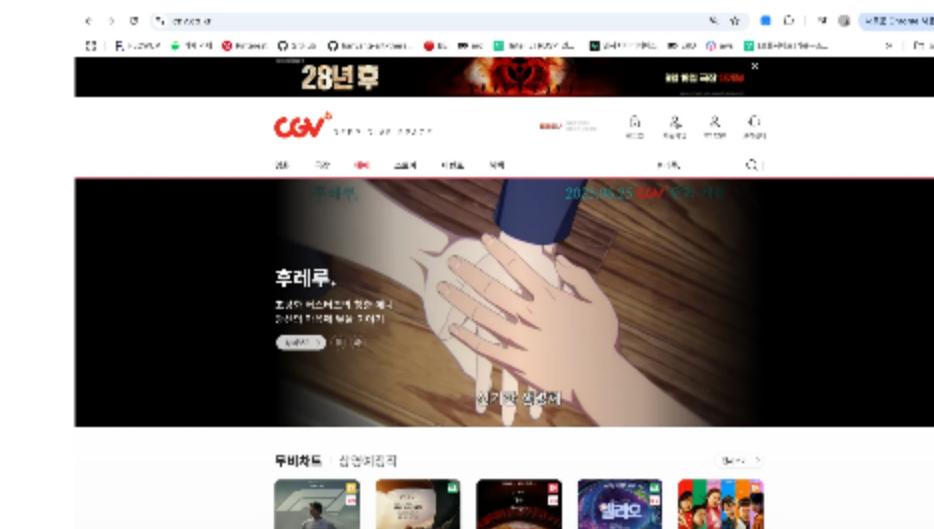
<http://3.34.183.11:9090/MovieProject/>

주제

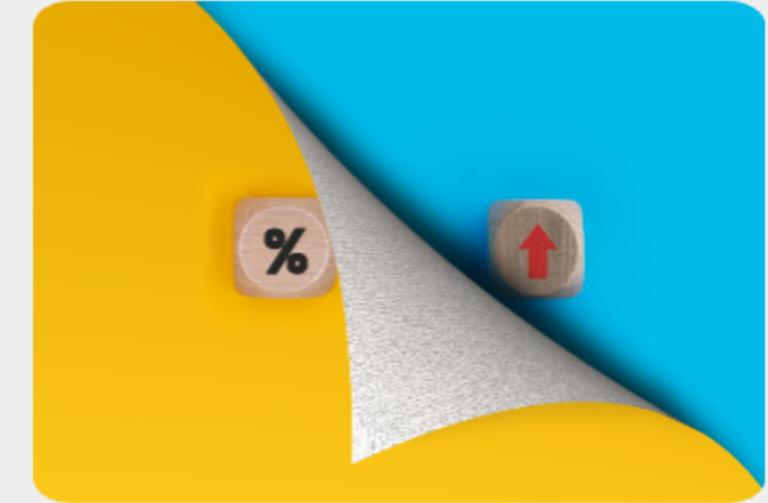
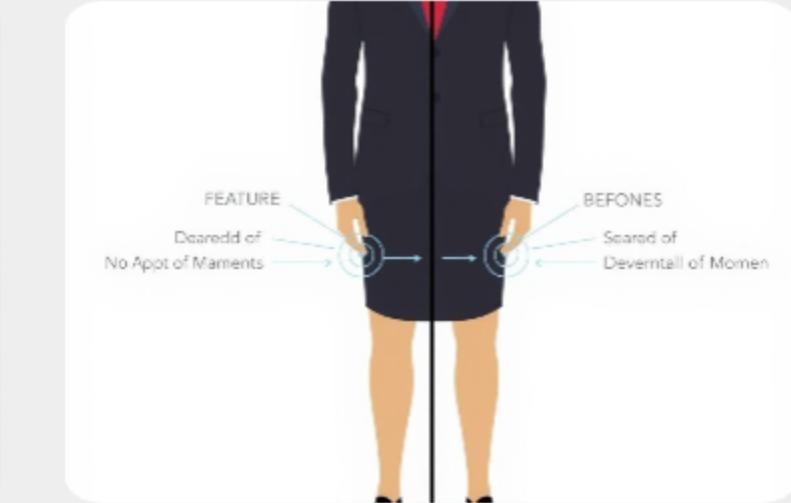
영화 예매 플랫폼 구축

벤치마킹사이트

CGV 공식 사이트



개발 배경 및 목표



Java & MVC 적용

- 실제 웹 프로젝트
- Java 언어 활용
- MVC 패턴 적용
- 실무 역량 강화



사용자 친화적 구축

- 쉬운 영화 정보 탐색
- 간편한 상영시간 확인
- 직관적인 UI/UX
- 사용자 경험 개선



벤치마킹

- CGV 공식 웹사이트
- 영화 정보 노출 방식
 - 필터 UI 참고
- 실제 예매 흐름 반영



개발 정보

- 기간: 2024.12.05~2025.01.10
- 배포: <http://3.34.183.11:9090/MovieProject/>



기술 스택

01 백엔드 기술

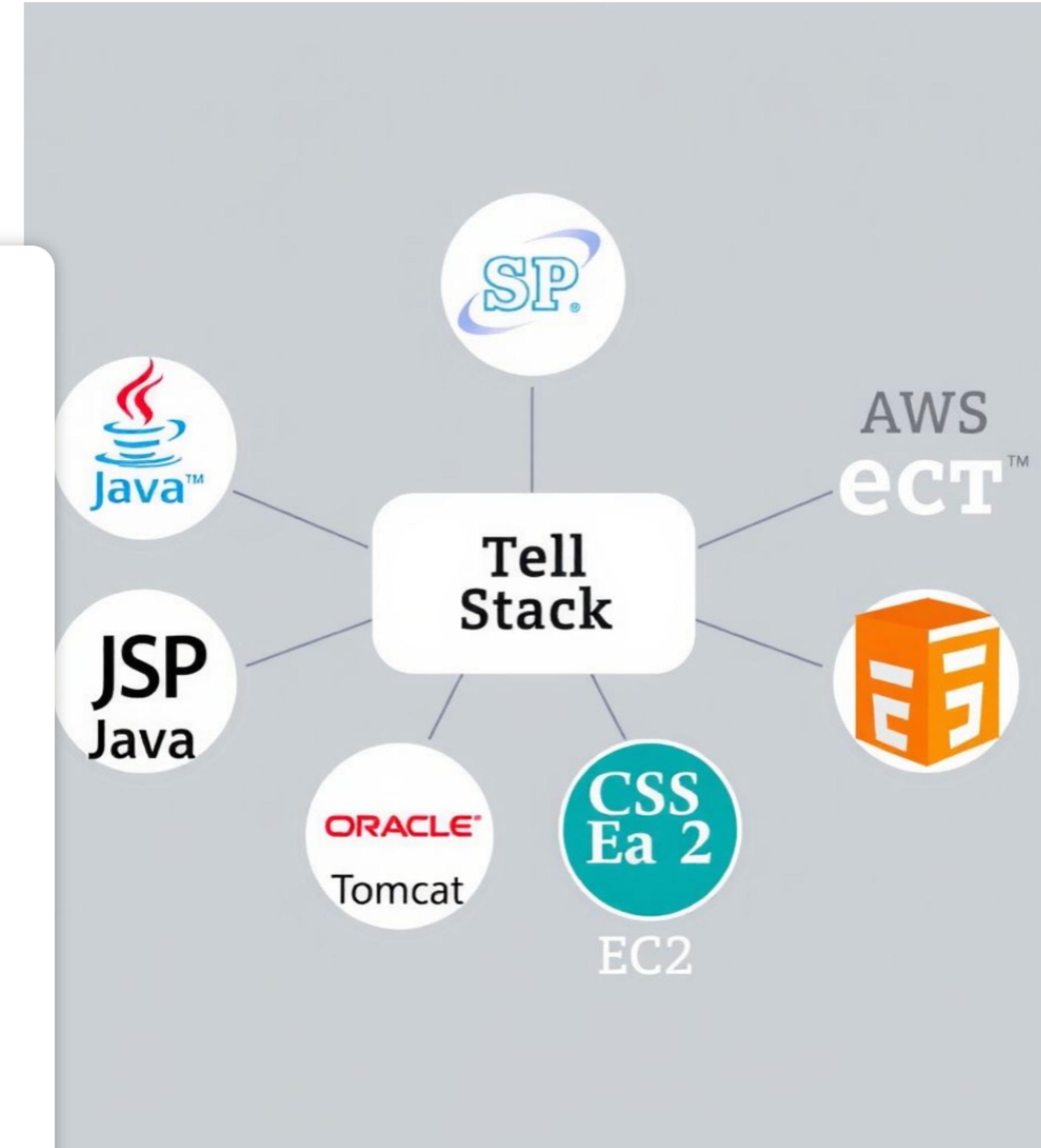
- Java: 핵심 비즈니스 로직 구현
- JSP: 동적 웹 페이지 생성
- Servlet: HTTP 요청 처리
- Oracle DB: 데이터 저장 및 관리

02 프론트엔드 기술

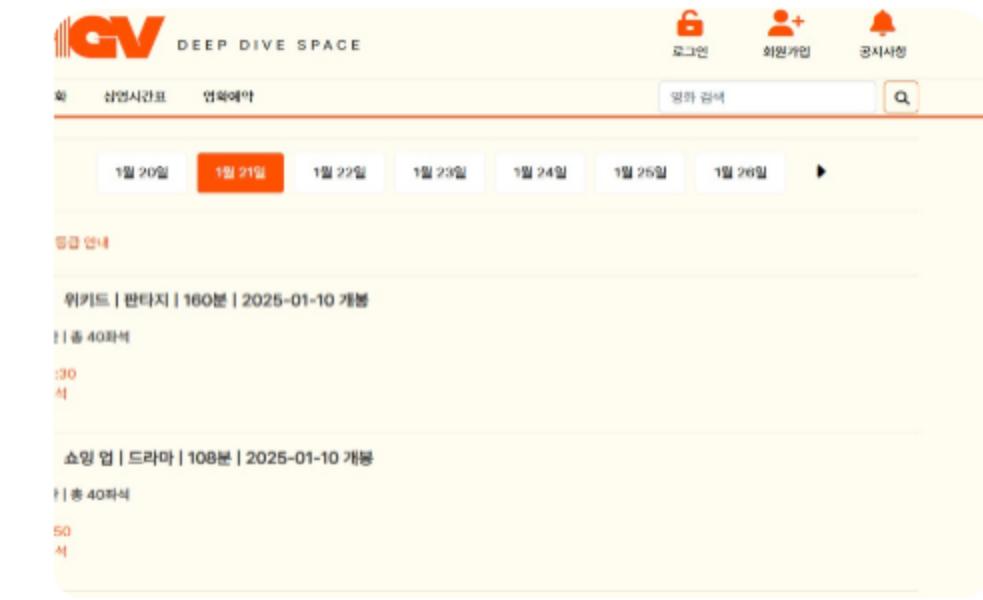
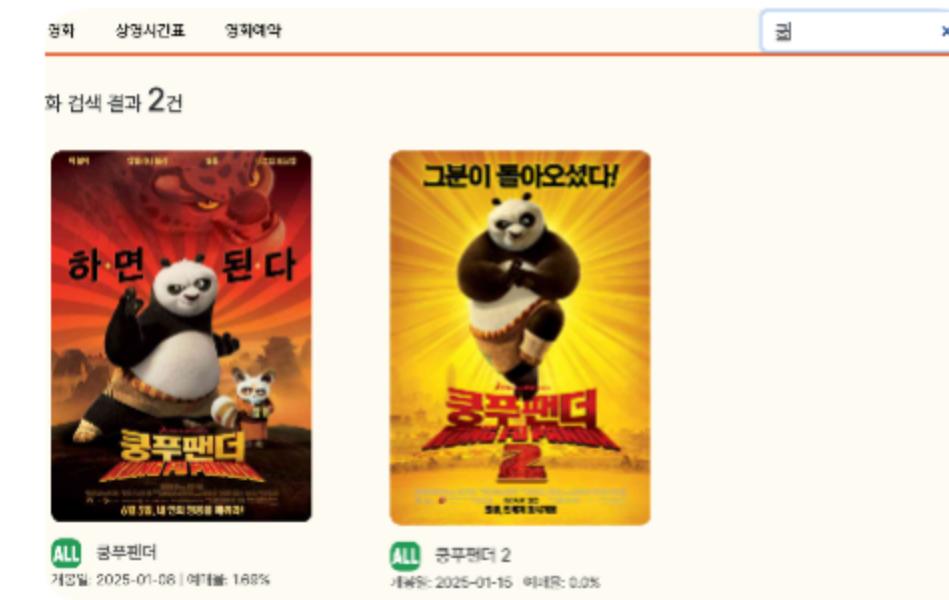
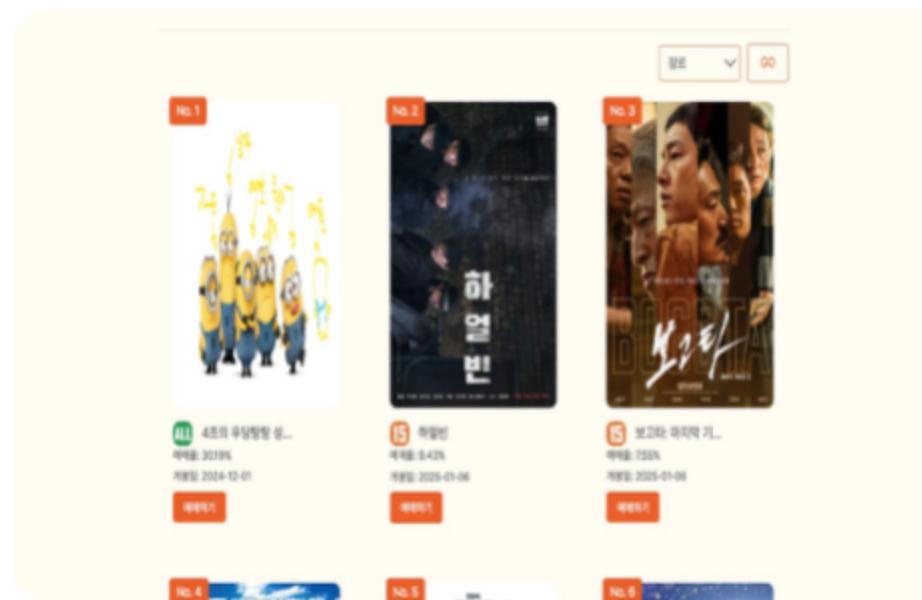
- HTML: 웹 페이지 구조 설계
- CSS: 스타일링 및 레이아웃 구성
- JSTL: JSP 페이지 내 로직 처리
- EL: 데이터 출력 간소화

03 인프라 기술

- Tomcat: 웹 애플리케이션 서버
- AWS EC2: 클라우드 서버 호스팅
- Git: 버전 관리 시스템
- Maven: 프로젝트 빌드 관리



주요 기능 (1)



영화 목록 조회

- DB에서 전체 영화 데이터 조회
- JSP로 목록 형태 출력
- JSTL 활용 동적 구성
- 페이지 처리 구현

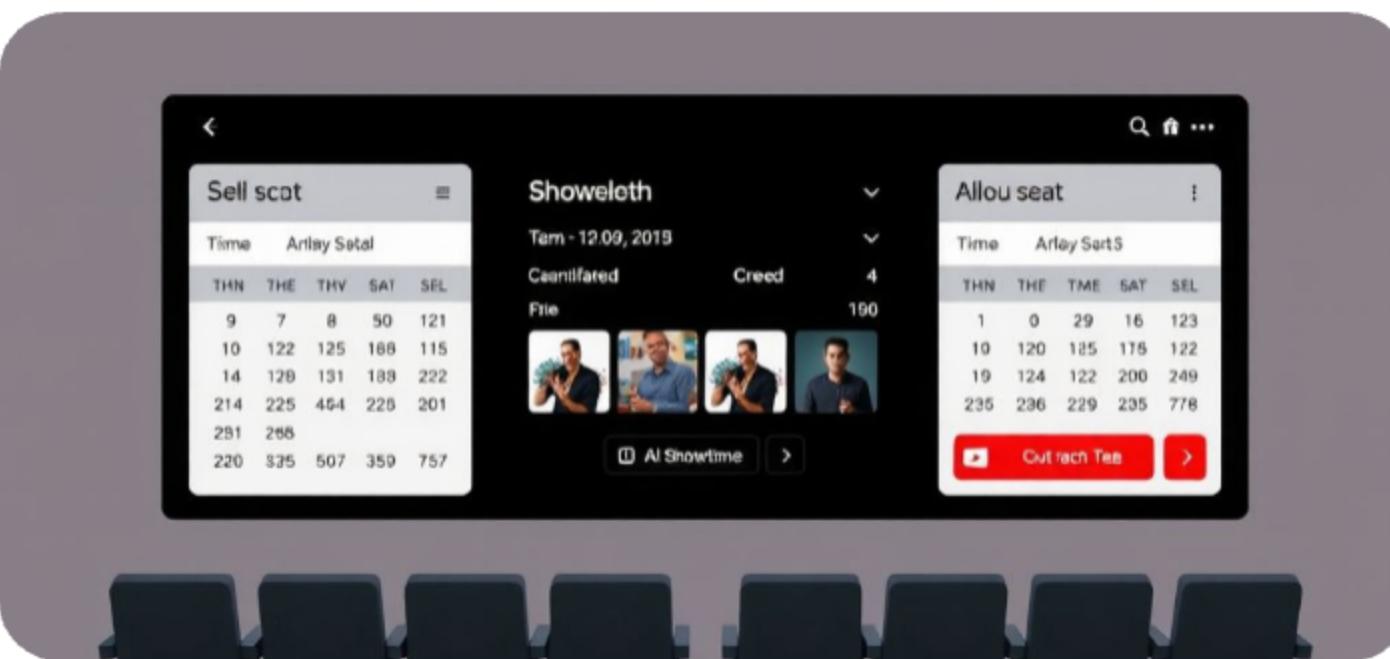
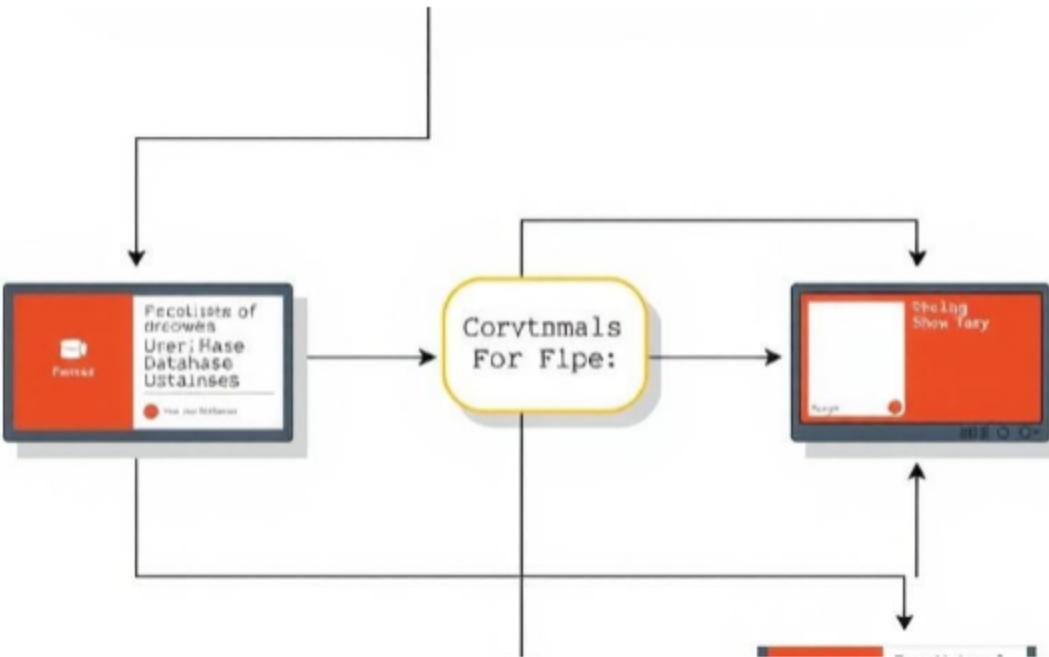
영화 검색 기능

- 장르 선택 옵션 제공
- 제목 키워드 입력 기능
- Controller에서 파라미터 처리
- DAO 조건 쿼리 수행

상영 시간표 확인

- 영화별 상영 일정 표시
- 시간대별 정렬 기능
- 날짜 선택 캘린더 구현
- 실시간 좌석 현황 연동

주요 기능 (2)



상영 시간표 확인

- 영화 선택 시 해당 영화의 상영 일정 및 시간 표시
- 테이블 형태로 정보 출력
- 사용자 편의성 고려한 UI 설계
- 실시간 데이터 반영으로 정확성 확보

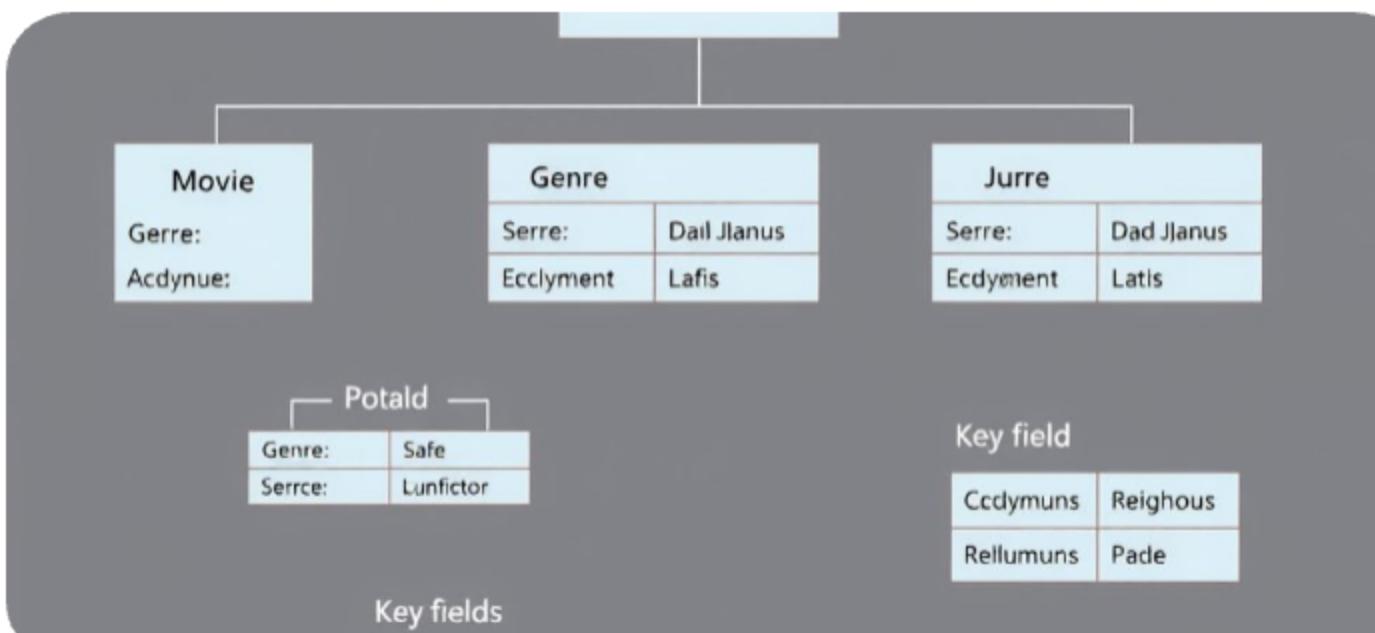


데이터 처리 및 출력

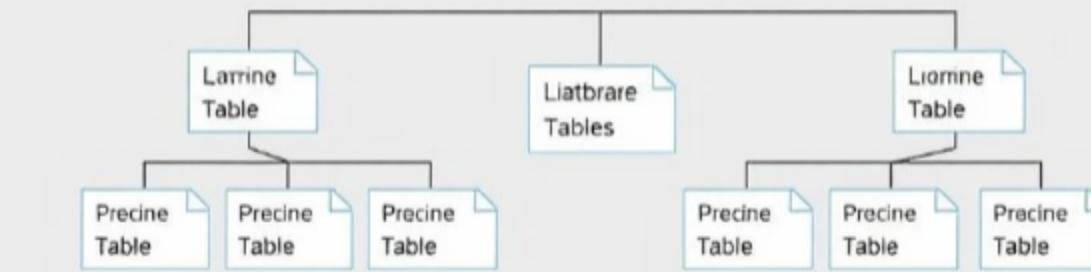
- Controller에서 영화 ID 기반 시간표 데이터 조회
- DAO를 통한 데이터베이스 접근 및 데이터 추출
- JSP에서 JSTL 활용하여 동적 테이블 생성
- 날짜별, 시간대별 정렬 기능 구현

ERD 및 데이터베이스 구조

핵심 테이블 구조



테이블 간 관계



- movie: movie_id, title, genre_id
- genre: genre_id, genre_name
- schedule: schedule_id, movie_id, date, time
 - 각 테이블은 Primary Key 설정
 - Foreign Key로 테이블 간 관계 정의

- movie → genre: N:1 관계
(genre_id로 연결)
- movie → schedule: 1:N 관계
(movie_id로 연결)
- 정규화를 통한 데이터 중복 최소화
- 인덱싱으로 검색 성능 최적화

벤치마킹 포인트

1

CGV 사이트 UI/UX 참고

- 영화 검색 필터 UI 벤치마킹
- 정보 배치 방식 분석 및 적용

2

사용자 중심 설계

- 메인 페이지 인기 영화 노출 기능
- 직관적인 필터링 기능 구현

3

실제 예매 흐름 반영

- 상영 시간 데이터 구조화
- 실제 예매 과정과 유사한 UX 설계

개발 과정 및 역할



본 프로젝트에서는 Java 기반의 웹 애플리케이션 개발에 주도적으로 참여했습니다.

영화 검색 및 시간표 확인 기능 구현, JSP 페이지 구성, DB 테이블 설계 및 연동, 쿼리 성능 개선 등 전반적인 백엔드 개발을 담당했으며,

MVC 패턴 구조 설계와 JSP, Servlet을 활용한 동적 페이지 구현을 통해 실시간 예매 정보 제공 기능을 완성했습니다.

또한 JDBC 기반으로 Oracle DB와 연동하고, PreparedStatement 및 인덱스 설계를 통해 쿼리 효율성을 높였습니다.



MVC 구조 설계 및 구현

- Controller, DAO, JSP 구조 설계
- Servlet을 이용한 요청 처리
- 모델 객체를 통한 데이터 전달
- View와 비즈니스 로직 분리



JSP 및 JSTL 활용

- JSTL, EL 문법 활용
- 동적 데이터 출력 구현
- 조건문, 반복문 처리
- 페이지 모듈화 및 재사용성 향상



데이터베이스 연동

- JDBC를 이용한 Oracle DB 연동
- PreparedStatement 사용
- 트랜잭션 처리 구현
- 연결 풀을 통한 리소스 관리



성능 최적화

- 인덱스 설계 및 적용
- 쿼리 실행 계획 분석
- 조인 최적화
- 페이지 처리 구현

프로젝트 성과 및 느낀 점

실제 구현 경험

- 영화 예매 사이트 구조 분석
- 백엔드와 프론트엔드 통합 개발
- 실제 서비스와 유사한 기능 구현

MVC 패턴 이해

- Controller와 DAO 분리 설계
- View와 Model 간 상호작용 이해
- 유지보수 가능한 구조의 중요성 체감

웹 개발 실무 역량 향상

- JSP 동적 출력 방식 숙달
- JSTL, EL 문법 활용 능력 향상
- 웹 서버 구축 및 운영 경험

프로젝트 라이프사이클 경험

- AWS EC2 활용한 실제 배포 경험
- 개발부터 배포까지 전 과정 참여
- 문제 해결 능력 향상

팀워크 및 의사소통 능력

- 기능별 역할 분담 및 협업
- 코드 리뷰를 통한 품질 향상
- 프로젝트 일정 관리 능력 습득

향후 개선 방향

