

소규모 사업장 관리 플랫폼

포트폴리오 목차

포트폴리오 목차

1. 프로젝트 개요

2. 프로젝트 설명

1) 사용 기술

3. 담당기능

1) 본인이 구현한 주요 기능

2) 프로젝트를 통해 발휘한 역량 및 성장한 점

1. 프로젝트 개요

항목	내용
프로젝트 소개	자영업 사장과 직원간의 스케줄 및 급여를 관리할 수 있는 웹어플리케이션
개발기간	2023.11.09~2023.12.09 (총 30일)
담당 역할	API개발(달력,근태관리,인증인가), 데이터베이스 테이블 설계
개발 인원	프론트엔드 2명, 백엔드 3명 (총 5명)
성과 및 결과	기한 내 중요 기능 구현 완료, 서버 배포 완료, 포스코x코딩온 부트캠프 프로젝트 발표 대상

프로젝트 Github 링크

AWS로 배포한 웹 어플리케이션 링크 <http://albaon.store/>

Swagger 링크 <http://ec2-3-39-203-178.ap-northeast-2.compute.amazonaws.com/swagger-ui/index.html#/Attendance>

2. 프로젝트 설명

1) 사용 기술

구분	기술
Frontend	TypeScript React Axios Zustand NPM
Backend	Spring SpringBoot SpringSecurity SpringDataJPA JWT
Database	MySQL AWS S3
배포	AWS EC2
API	Swagger

3. 담당기능

1) 본인이 구현한 주요 기능

1.캘린더 일정 조회 기능

Aa url	≡ 메서드	≡ body	≡ response	≡ error	≡ 설명
<u>/admin/schedule/{memberid}/{storeid}/{month}</u>	GET		resultcode, message, List<Attendance>	1. List<Attendance> 가 null일때 2. member가 null일 때	캘린더에서 모든 근로자의 일정을 조회하는 기능
<u>/user/schedule/{memberid}/{storeid}/{month}</u>	GET		resultcode, message, List<Attendance>	1. List<Attendance> 가 null일때 2. member가 null일 때	근로자 본인의 일정만을 조회하는 기능
<u>/admin/attendance/create</u>	POST	AttendanceDto	resultcode, message	1. 권한이 admin이 아닐때 2. member가 null일 때	관리자 권한이 근로자

Aa url	≡ 메서드	≡ body	≡ response	≡ error	≡ 설명
					의 출 퇴 근 시 간, 시 급 을 정 할 수 있 도 록 하 는 기 능

2. 근태관리 기능

Aa url	≡ 메서드	≡ body	≡ response	≡ error	≡ 설명
/admin/attendance/update	PATCH	AttendanceUpdatedDto	resultcode, message	1. 권한이 admin이 아닐 때 2. member가 null일 때	관리자권한이 근로자의 출 퇴 근, 시 급 을 수 정 할 수 있 도 록 하 는 기 능
/admin/attendance/delete	DELETE	AttendanceUpdatedDto	resultcode, message	1. 권한이 admin이 아닐 때 2. member가 null일 때	관리자권한이 근로자의 일 정을 삭 제 할 수 있 는 기 능
/user/attendance/leavework	PATCH	AttendanceUpdatedDto	resultcode, message	1. 권한이 user가 아닐 때 2. member가 null일 때 3. 출근기록이 null일 때	근로자가 퇴근 후 시간을 등 록 하 는 기 능
/admin/attendance/confirm	PATCH	AttendanceUpdatedDto	resultcode, message	1. 권한이 admin이 아닐 때 2. member가 null일 때 3. 퇴근기록이 null일 때	관리자가 근로자의 하루 일 당을 승 인 하 는 기 능. 승인이 되면 (일한 시간*시 급)만큼 근로자에게 급여가 기록된다.
제목 없음					

3. 토큰 인증인가 기능

Aa url	메서드	body	response	error	설명
<u>/tokenValidation</u>	POST	TokenValidationDto	valid , resultcode, message	1. Jwt토큰이 존재하지 않는 경우	Jwt토큰의 유효성 검사
<u>/refreshToken</u>	POST	TokenDto	LoginResponseDto , resultcode, message	1. member가 null일때 2. 기존 Jwt 토큰과 데이터베이스에 저장된 Jwt 토큰이 다른 경우	Jwt토큰을 재발급하는 기능 보안을 위해 로그인 상태의 Jwt토큰과 데이터베이스에 저장된 Jwt토큰의 유효성 검사를 한다.

4. 이메일 비밀번호찾기 기능

Aa url	메서드	body	response	error	설명
<u>/findPassword</u>	POST	FindPasswordDto	resultcode, message	1. member가 null일 때	임시비밀번호를 발급하여 이메일로 알려주는 기능

2) 프로젝트를 통해 발휘한 역량 및 성장한 점

1. 프로젝트의 높은 기여도

- 백엔드 API의 약 40%를 담당했다.
- trello를 통해 팀원들의 전체적인 계획을 정리했고 정해진 기간내에 핵심기능을 완성도있게 마쳤다.
- 프론트엔드와 지속적인 회의를 주도했고 필요한 업무를 분배했다.

2. 컨트롤러와 프론트엔드 사이 전달 방법

전반적으로 Entity에서 setter사용을 지양하고 대신 DTO를 사용했다.

그래서 패스워드와 같이 보내지않아도 될 데이터를 클라이언트에 노출이 되지 않도록 했다.

3. 쿼리 성능 개선

문제의 원인

기능 위주 구현을 마치고 쿼리를 점검했을때 하나의 조회를 실행했을 뿐인데 많은 양의 쿼리가 발생하는 것을 알게되었다.

이는 기존의 모든 회원을 찾고 반복문을 통해 한 회원당 근태기록을 조회하면서 N+1 현상이 일어난것이 원인이었고 더 미데이터값으로 회원수 10명과 근태기록 10개를 넣었을때 아래와 같이

회원정보와 join된 사업장정보 쿼리 ⇒ 1개

회원정보 조회 쿼리 ⇒ 1개

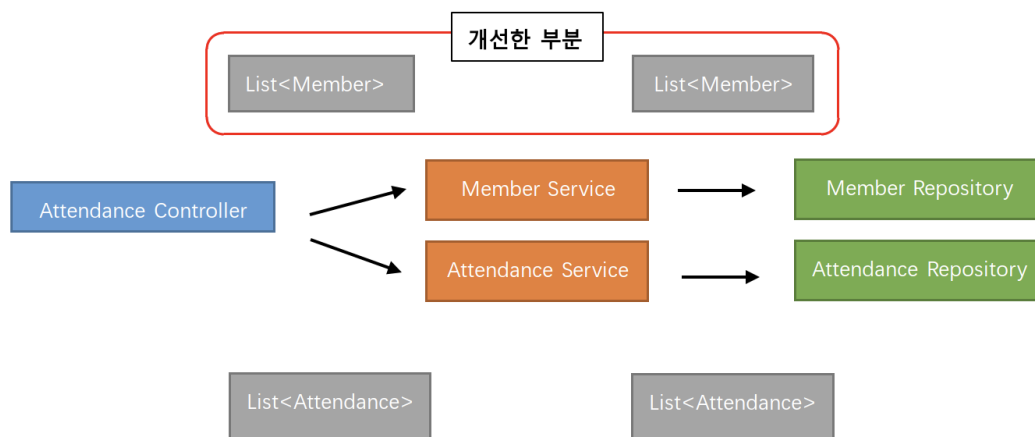
근태기록 모두 조회⇒ 쿼리 x 20

총 22개의 쿼리가 발생하게 되었다.

해결방법

성능개선

	기존	개선 후	개선 효과
발생한 쿼리(N=20)	22	3	43%
평균(ms)	104.6	84.8	19%
호출 1번	173	158	8.7%
호출 2번	191	157	17.9%
호출 3번	58	48	17.3%
호출 4번	47	28	40.5%
호출 5번	54	33	38.9%



4. 부족한점과 앞으로 개선할 점

[데이터베이스 테이블 설계 신중히 하기]

처음에 데이터베이스 테이블을 작성할때 확실하지 않아도 넘어가거나 나중에 추가하면된다는 단순한 생각을 가졌습니다.

그러나 기능이 점점 많아질수록 N+1문제 등 쿼리발생이 생각보다 많았고 중복되는 로직을 정리하거나 지연로딩을 쓰는 등의 노력으로 개선했으나 근본적인 해결책이 되는건 아니라는것을 느꼈습니다.

그래서 다음부터 개발을 진행한다면 관계설정부분부터 신중하게 작성해야겠다고 생각했습니다.

[추가적으로 채팅 기능 만들기]

처음 프로젝트 아이디어를 회의할때 웹소켓을 이용해 채팅기능도 구현하기로 했었습니다.

그러나 한정된 시간과 역량으로 인해 계획을 조정했고 대신 예외처리와 오류를 점검하며 기존의 기능들에 대한 최적화를 하면서 남은 시간을 적절하게 분배했습니다.

이번 프로젝트를 통해 정해진 기한동안 내가 어느정도까지 구현할 수 있는지 아는 것 또한 중요하다고 느꼈습니다.

감사합니다.