

쉐어드원 & 코카인의 기업 프레젠테이션

커피 원두 및 부자재 납품 ERP 시스템



2조 신지윤, 김수현, 홍나린, 강현구, 최해찬, 남상혁

CONTENTS

CHAPTER 1

프로젝트 개요

CHAPTER 2

프로젝트 팀 구성 및 역할

CHAPTER 3

프로젝트 팀 수행 절차 및 방법

CHAPTER 4

프로젝트 수행 경과

CHAPTER 5

트러블 슈팅

CHAPTER 6

자체 평가 의견

01 프로젝트 개요

프로젝트 주제 및 선정 배경, 기획의도

주제 선정 과정에서 다양한 산업군을 고려한 끝에, 급성장하는 커피 시장과 그에 따른 효율적 운영 시스템의 필요성에 주목하였습니다. 특히, 원두와 부자재 구매 및 재고관리, 부서별 인사 관리까지 통합적으로 지원하는 ERP 시스템이 적합하다고 결론지었습니다.

프로젝트 내용

프로젝트: 구매·재고·인사 관리를 통합한 ERP 시스템으로, 직관적 UI와 보안 강화를 구현
훈련 연관성: React와 Spring Boot로 UI/UX와 데이터 관리, MVC 아키텍처 적용

활용 장비 및 재료

서버 및 데이터베이스: PostgreSQL, Amazon, RDS, AWS
백엔드: Spring Boot & Spring Security, MyBatis, Java
프론트엔드: HTML5, CSS3, JavaScript, React.js, Thymeleaf, jQuery, AJAX
API 및 시각화: Chart.js, Daum API, FullCalendar
개발 도구 및 협업 툴: IntelliJ, Slack, GitHub, Figma, Gradle, VS Code
운영체제: Windows 10, macOS

01 프로젝트 개요

프로젝트 구조

백엔드는 Spring Boot로 Restful API를 제공하며,
Spring Security와 MyBatis를 사용

프론트엔드는 React.js와 Axios로 구현되었으며,
직관적인 UI를 제공

PostgreSQL을 데이터베이스로 사용하며, API는 CRUD 작업 처리
개발 및 배포는 GitHub, Gradle, AWS를 통해 관리

활용방안 및 기대효과

ERP 시스템은 인사관리, 재고관리, 영업관리로 구성되어
기업의 핵심 업무를 효율적으로 처리합니다.

인사관리: 사원 및 부서 정보 관리 및 조직 효율성 향상

재고관리: 재고 모니터링 및 재고 최적화

영업관리: 계약 및 판매 관리 및 매출 분석

02 프로젝트 팀 구성 및 역할



신지윤

백엔드 & 프론트엔드

상품 시스템 구현
상품 구매 시스템 구현
상품 가격 계약 시스템 구현
시나리오 작성



김수현

백엔드 & 프론트엔드

상품 가격 계약 시스템 구현
상품 판매 시스템 구현
프론트엔드 디자인 총괄
결과 PPT 제작



홍나린

백엔드 & 프론트엔드

공급업체 시스템 구현
상품 구매 시스템 구현
프론트엔드 디자인 총괄
결과 PPT 제작



강현구

백엔드 & 프론트엔드

시큐리티 권한 기능 구현
사원 시스템 구현
부서 시스템 구현
결과보고서 워드 작성



최해찬

백엔드 & 프론트엔드

메인페이지 구현
부서 시스템 구현
결과보고서 워드 작성



남상혁

백엔드 & 프론트엔드

계약 프로세스 구현
상품 판매 시스템 구현
시연 영상 제작

03 프로젝트 수행 절차 및 방법

개발 진행 기간 : 9/2 ~ 9/26

SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
	프로젝트 주제 선정 및 기획안 작성		개발 파트 분배, ERD 설계 및 화면 UI 피그마 구상 설계			html(css) 화면 설계
8	9	10	11	12	13	14
	개발 진행 및 개별 프로그램 1차 피드백 진행					
15	16	17	18	19	20	21
	1차 피드백 반영 및 2차 개발 진행					
22	23	24	25	26	27	28
		프로그램 2차 피드백	통합테스트 및 종료 산출물			발표
29	30					

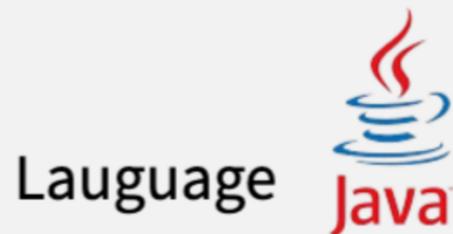
04 프로젝트 수행 결과 - 개발환경

Back-End

Database



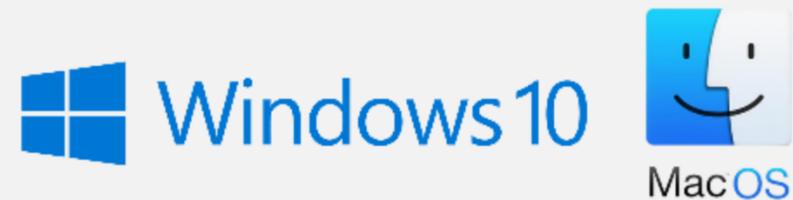
Framwork



Tools



OS

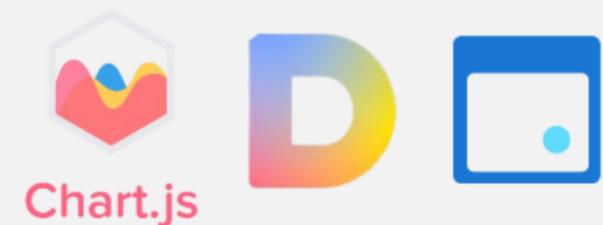


Front-End

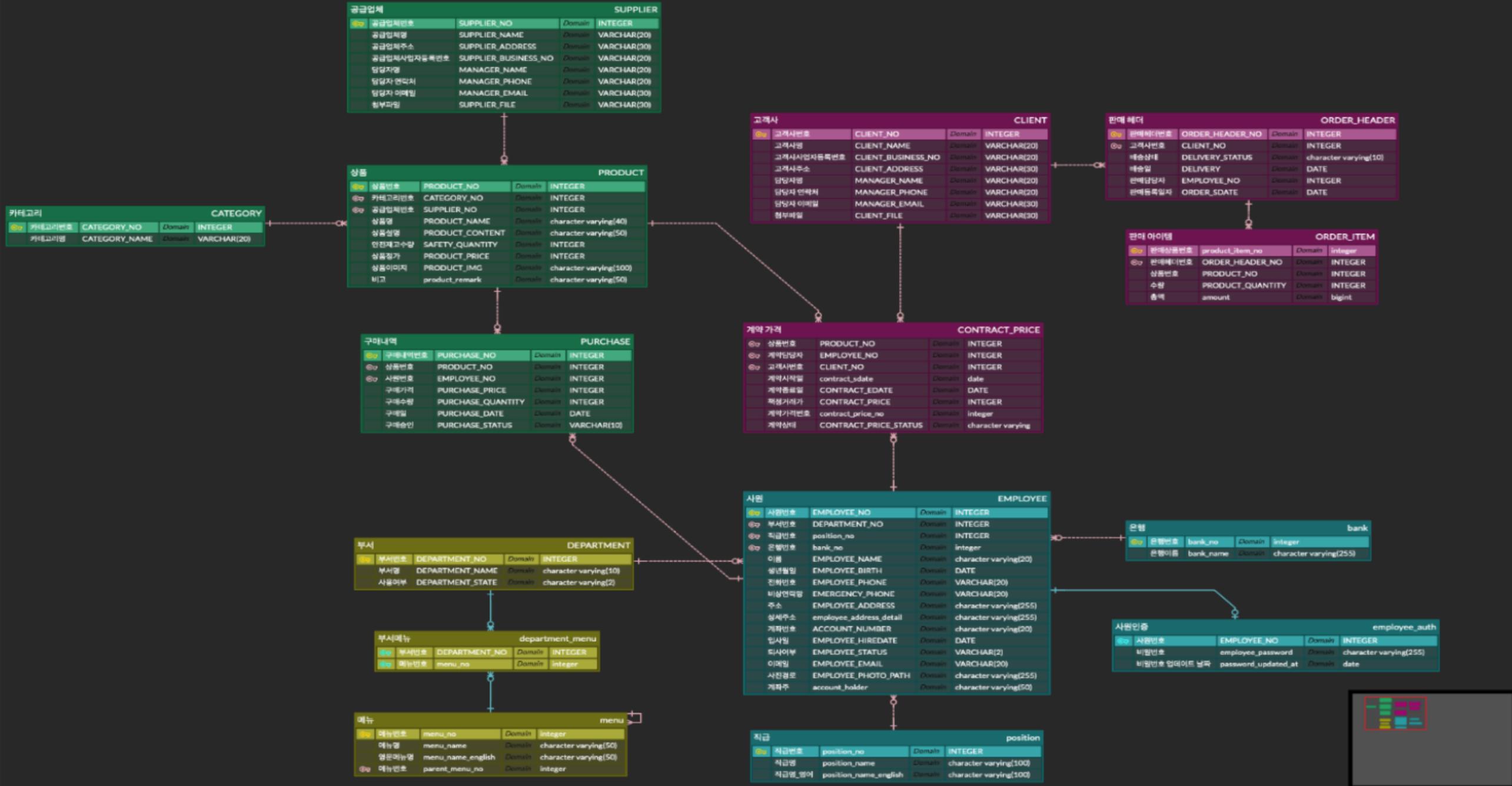
Web



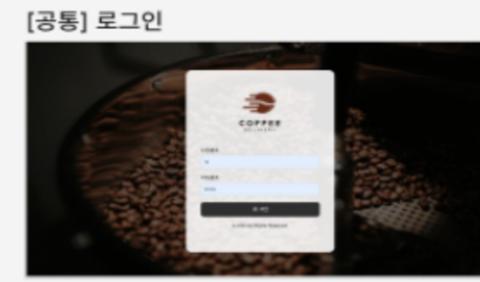
API



04 프로젝트 수행 결과 - ERD CLOUD



04 프로젝트 수행 결과 - Figma UI



04 프로젝트 수행 결과 - 주제 소개



< 공급업체 >

구매
→



< 본사 >

판매
→

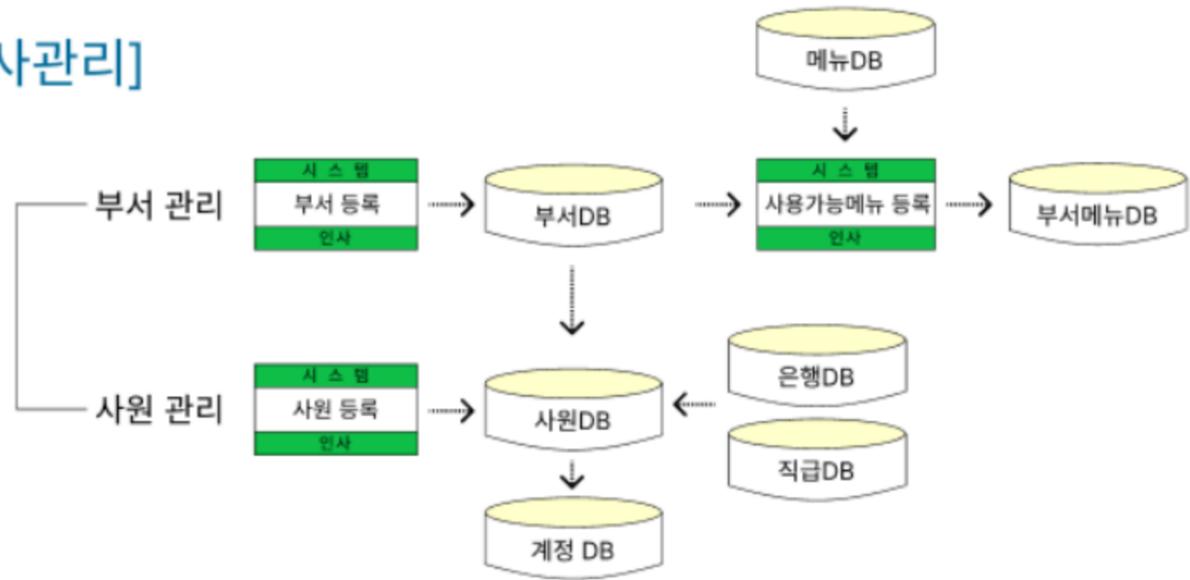


< 고객사 >

04 프로젝트 수행 결과 - 프로세스

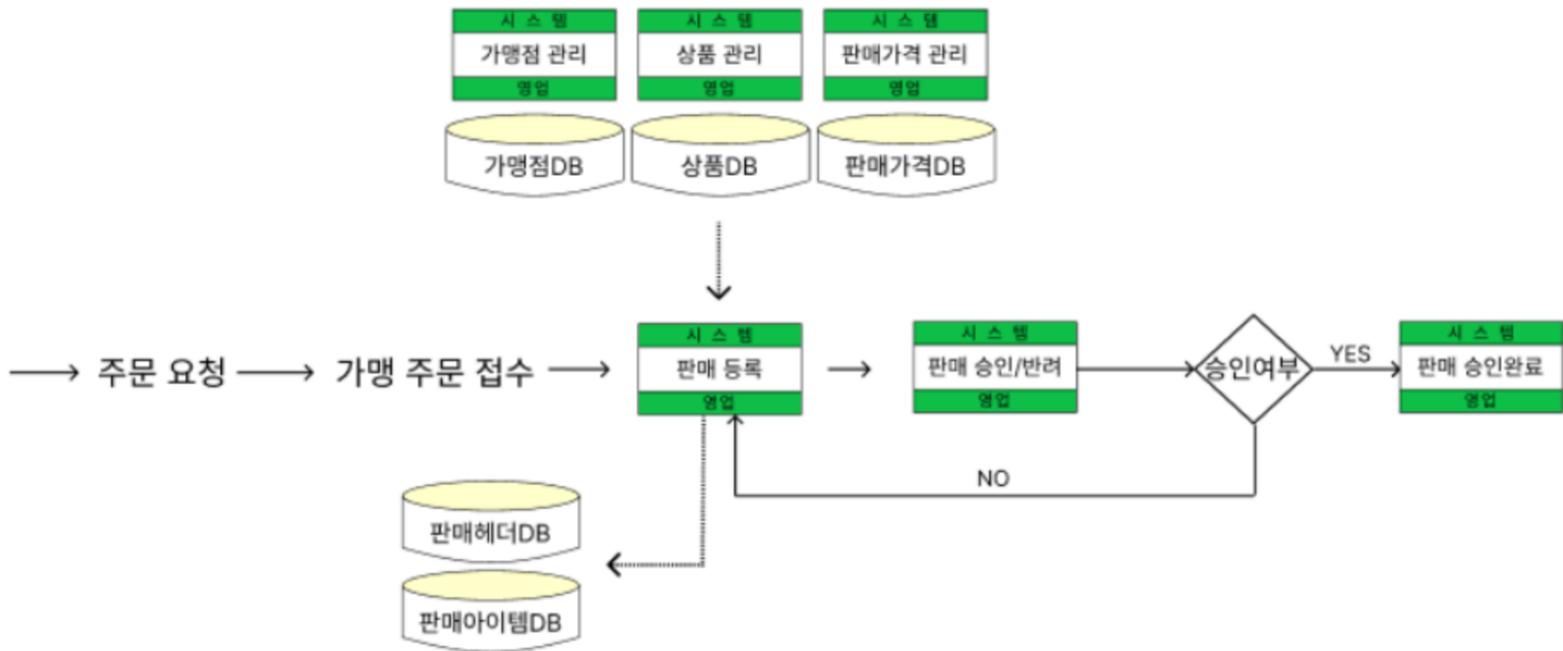
[인사관리]

본사



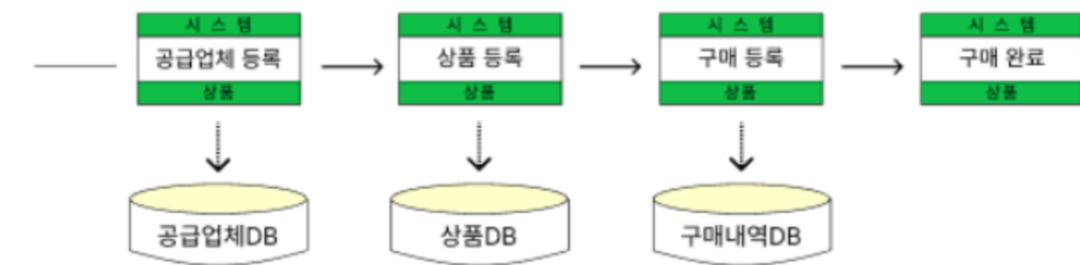
[영업관리(판매)]

가맹점



[재고관리(구매)]

본사



05 트러블 슈팅

문제식별

1. 계약 등록 시 계약 기간이 겹쳐서 동일한 기간에 계약이 중복 체결되거나 기존 계약과 충돌나는 문제점 발생
2. 판매 등록 시 하나의 등록건에 여러개의 상품을 저장하는 로직을 설계하고 같은 상품이 여러개 들어가는 문제점 발생
3. ERP 시스템에서 타임리프와 리액트 통합 중 서버 사이드 렌더링(SSR)과 클라이언트 사이드 렌더링(CSR) 충돌로 렌더링 오류 발생

문제 해결 접근 방법

1. 기간이 겹치는 케이스에 대해 계약일이 분할되어 처리되는 방법을 정확하게 정의하고 그에 따른 로직을 구체적으로 설계하였습니다.
2. foreach로 등록상품을 배열로 저장하였으며 같은 상품이 여러개 들어가는 부분에 대해서는 자바스크립트로 중복 검사 후 제어하였습니다.
3. 타임리프에서 reactInclude 조건을 사용해 리액트 전용 페이지에서 타임리프 인클루드 코드가 실행되지 않도록 설정하고, 각 프레임워크에 필요한 로직을 분리하였습니다.

해결 사례

1. 기존에 등록된 기간(9/1-9/30)에 새로운 계약(9/10-9/20)이 등록되면 계약이 분할되어 (9/1-9/9 | 9/10-9/20 | 9/21-9/30)으로 처리되도록 하였습니다.
2. 상품 등록 시 계약 체결된 상품은 모두 보여지지만 여러 상품 등록 시 이미 등록된 상품은 등록할 수 없게 보여지게 했습니다.
3. 다이어그램으로 문제 해결 전후의 흐름과 구조를 시각적으로 설명하고, 문제 해결 전후의 화면을 스크린샷으로 비교하였습니다.

결과 및 도훈

1. 계약 기간 중복 문제를 해결하면서 철저한 데이터 검증과 비즈니스 로직의 세밀한 설계의 중요성을 깨달았습니다.
2. 상품 등록 중복 문제를 해결하며 데이터 입력 시 중복 검증과 제어 로직의 중요성을 깨달았습니다.
3. 렌더링 충돌 해결과 로딩 속도 개선, 명확한 조건 분기 및 코드 구분의 중요성을 인식하였고 향후 유지보수와 확장성 고려가 필요함을 느꼈습니다.

06 자체평가 의견

완성도 평가

이번 프로젝트는 사전 기획 과정에서 다양한 아이디어를 수집하여 큰 그림을 그렸습니다. 사전 기획에서 수집한 다양한 아이디어 중 핵심 기능에 집중해 불필요한 부분을 배제함으로써 완성도 높은 결과물을 도출하였습니다. 그 결과, 저희는 프로젝트 결과물에 대해 10점 만점에 9점으로 평가할 수 있을 만큼 만족스러운 결과를 도출했다고 생각합니다.

잘한 부분과 아쉬운 점

저희 팀은 사용자 친화적인 UI 구현에 어려움이 있었으며, UX와 UI 디자인에 더 많은 집중이 필요하여 생각보다 많은 시간이 소요되었습니다. 그러나 서로의 강점을 잘 활용하며 협업을 이루었고, 각 구성원의 역량을 극대화하였으며, 이를 통해 프로젝트를 효율적으로 진행할 수 있었습니다.

추후 개선 및 보완할 점

이번 프로젝트를 진행하면서 가장 중요한 점은 이해도라는 것을 깨달았습니다. 프로젝트의 성공 여부는 각 팀원이 프로젝트의 목적과 요구 사항을 얼마나 깊이 이해하고 있는지에 달려 있었으며, 이 과정에서 다양한 경험이 실력 향상에 큰 도움을 주었다고 생각합니다.

느낀 점 또는 성과

이번 프로젝트를 진행하며 기업의 현직자들과 소통하고, 피드백을 받는 과정에서 실제 협업이 어떻게 이루어지는지 배울 수 있었습니다. 특히, 기업에서의 프로젝트 진행 방식과 개발 과정에 대한 구체적인 이해를 할 수 있었고, 이러한 경험은 간접적인 협업 경험으로 앞으로 취업과 근무 환경에서 매우 유용할 것이라 느꼈습니다.

Thank you

for watching