

객체지향 설계기법

팀 프로젝트 결과보고서

7조 조재영 정유라 최인호



목 차

1. 발표 자료 정리
2. 실행 캡처 화면
3. 발표 때 Q & A
4. 개인적 소감
5. 실행 환경

1. 발표자료 정리

“

객체지향 설계기법

”

— 상품관리 서비스

— 7조
조재영 14011038
최인호 16011039
정유라 15012987

목차



Contents

- 01 . 팀원 소개
- 02 . 프로젝트 개요
- 03 . 핵심 기술
- 04 . 시연
- 05 . Q & A

팀원 소개



조재영

- Server Controller
- Client Controller



정유라

- Client UserView
- Client AdminView
- Client Controller



최인호

- Server DataBase
- 프로젝트 설계

MVC 패턴을 적용하여 각 모듈별로 파트를 나눴습니다.
분리와 의존성 분리를 통해 단위테스트가 가능한 형태

프로젝트 개요



프로젝트 개요






아주머니가 직접 품질처리
=> 제품의 재료 재고 관리 기능을 추가




자동화!

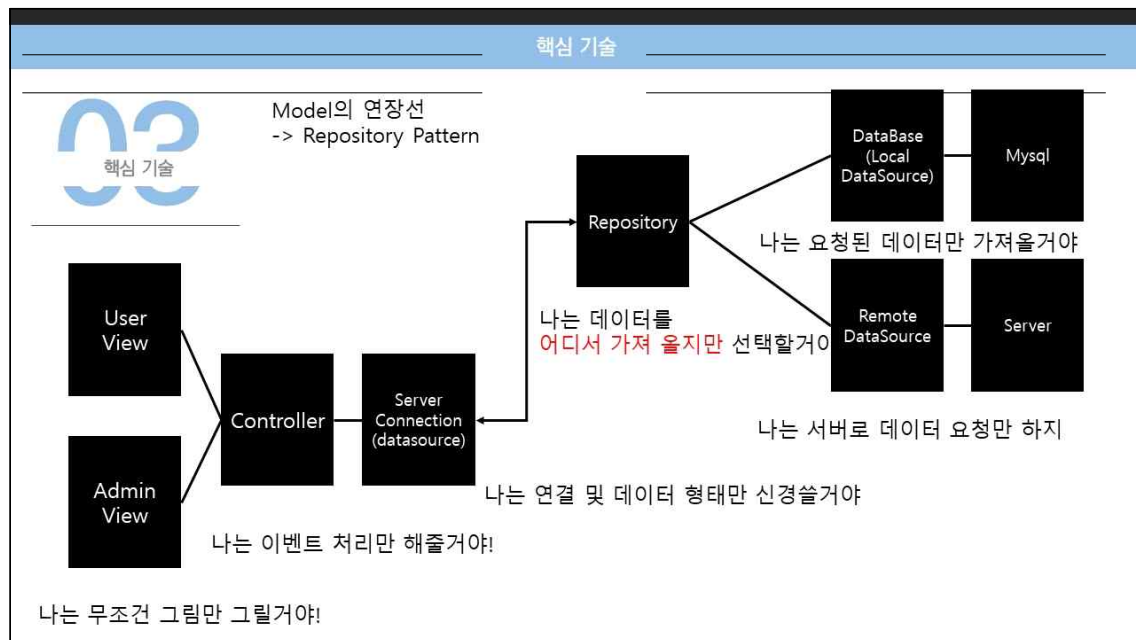
프로젝트 개요는 키오스크에서 직접 관리하는 것이 아닌 자동화를 통해 품질관리를 해 보자는 취지로 시작하게 되었습니다.

핵심 기술				
				
PrCode	PrName	PrPrice	PrNumber	PrIngredient
1001	돈가스	100	0	101-200/104-10/108-50
1002	치킨가스	200	2	102-500/104-10/108-50
1003	김치우동	300	3	105-100/106-300/109-30
IgCode	IgName	IgPrice	IgNumber	IgProduct
101	돼지고기	1300	9750	1001-200/1004-100/1007-30/1008-250
102	닭고기	2500	7500	1002-500
103	소고기	3000	9900	1009-100/1010-100

DataBase 간의 제약조건을 이용하지 않고, 직접 내부알고리즘을 통해 PrIngredient, IgProductattribute를 통해 데이터 관리를 하였습니다.

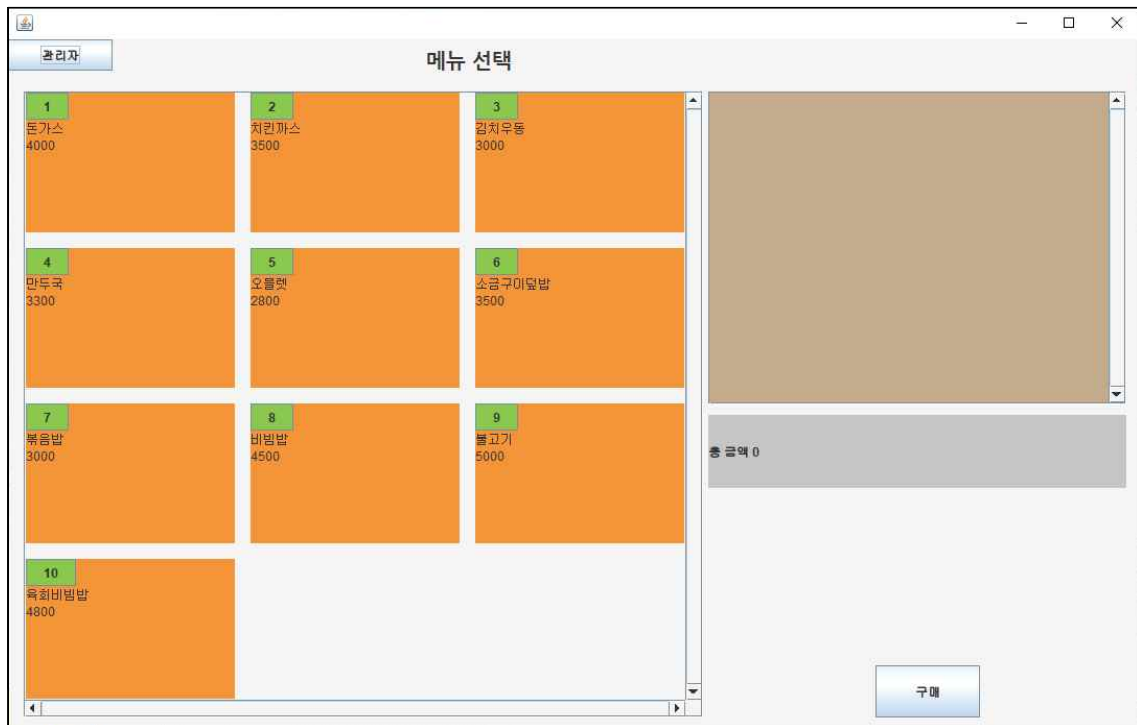
핵심 기술				
				
Current Items				
No	재고	재고수량		
101	돼지고기	30000		
102	닭고기	30000		
103	소고기	30000		
104	계란	30000		
105	김치	29900		
106	면	29700		
107	쌀	29400		
108	알가류	30000		
109	알파	29870		
110	파	29900		
IgCode	IgName	IgPrice	IgNumber	IgProduct
1	101 돼지고기	1300	30000	1001-200/1004-100/1007-30/1008-250
2	102 닭고기	2500	30000	1002-500
3	103 소고기	3000	30000	1009-100/1010-100
4	104 계란	4000	30000	1001-10/1002-10/1005-100/1007-30/1009-30
5	105 김치	5000	29900	1003-100
6	106 면	3000	29700	1003-300
7	107 쌀	10000	29400	1005-300/1006-300/1007-300/1008-250/1009-250/1010-300
8	108 알가류	2000	30000	1001-50/1002-50/1004-80
9	109 알파	3000	29870	1003-30/1008-200
10	110 파	2500	29900	1004-80/1005-30/1006-50/1010-40

실제로 데이터가 실시간 품절되는 핵심 기술을 볼 수 있으며, 서버 DB에도 반영되는 것을 확인할 수 있습니다.

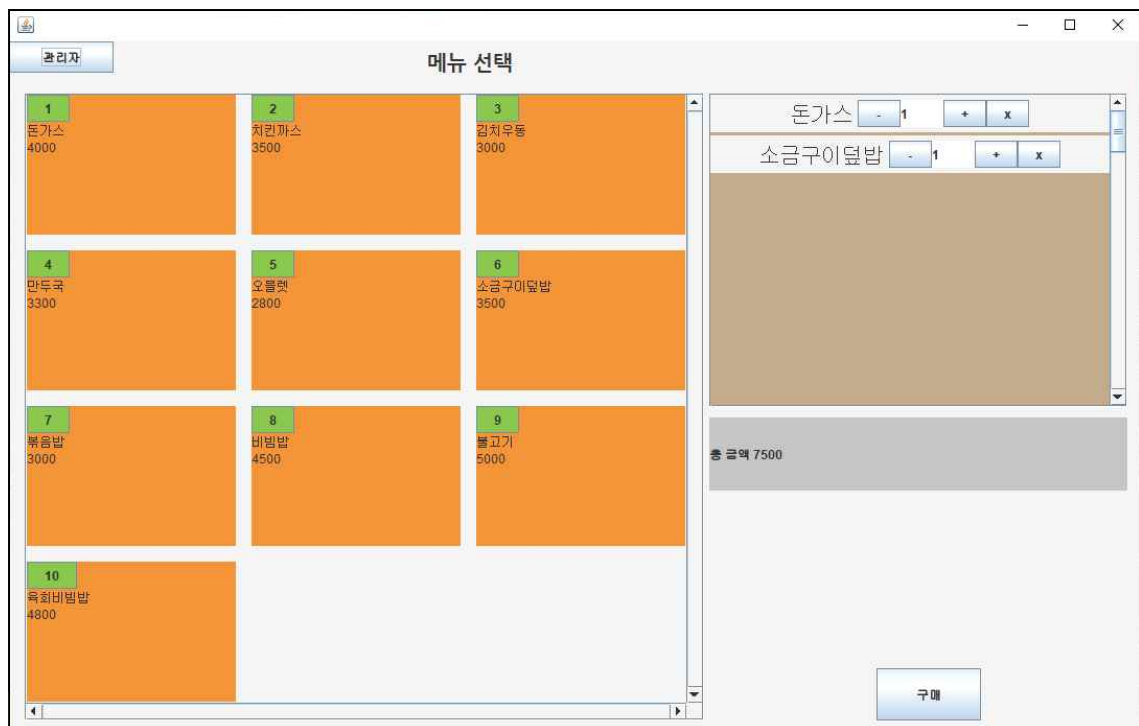


- Model 파트를 Repository Pattern을 적용하여 데이터 요청을 분기, 선택하는 로직을 가진다.
- Passive한 View형태를 만들어 넘어온 데이터를 보여주지만 하는 View
- 사용자 입력에 따라 DataSource를 통해 서버로 데이터 요청만 하는 Controller
- Client의 입력에 따라 DB에서 무슨 데이터가 올지 선택하는 Repository
- Client와 연결에만 신경쓰고, 받은 데이터를 가공하여 필요한 데이터 형태로 넘겨 주기만 하는 RemoteDataSource
- 요청받은 데이터를 가공하는 LocalDataSource
- 요청받은 데이터를 가져오기만 하는 MySQLDB

2. 실행 화면 [Client]

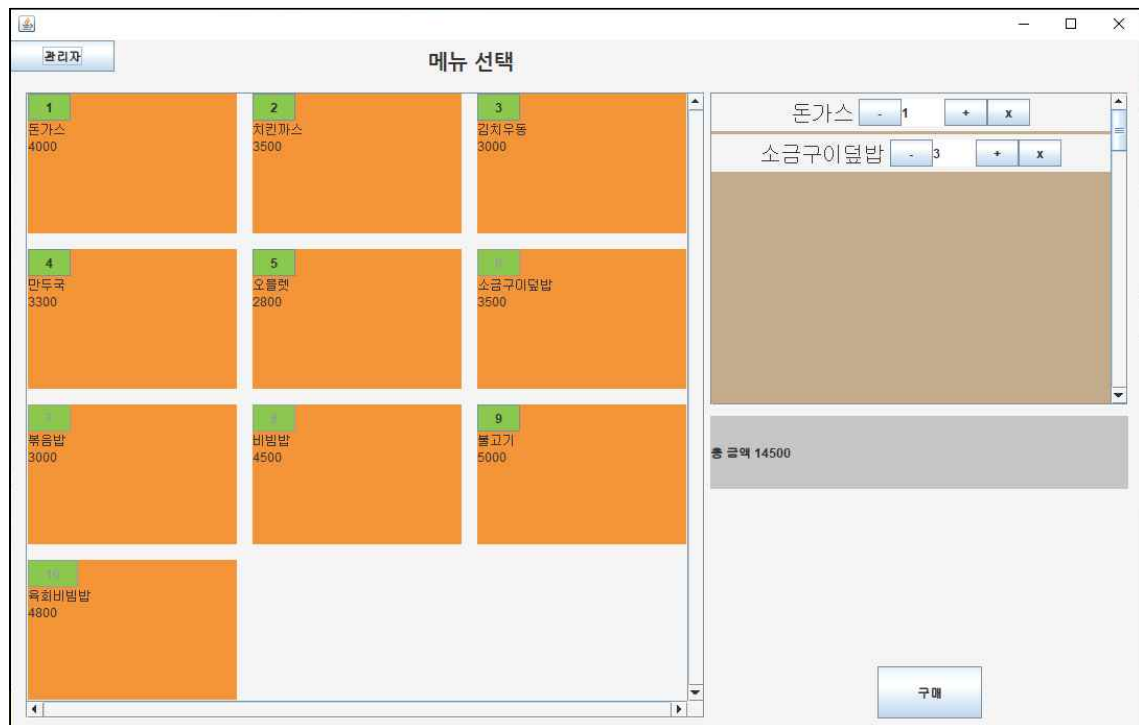


< 최초 실행 >

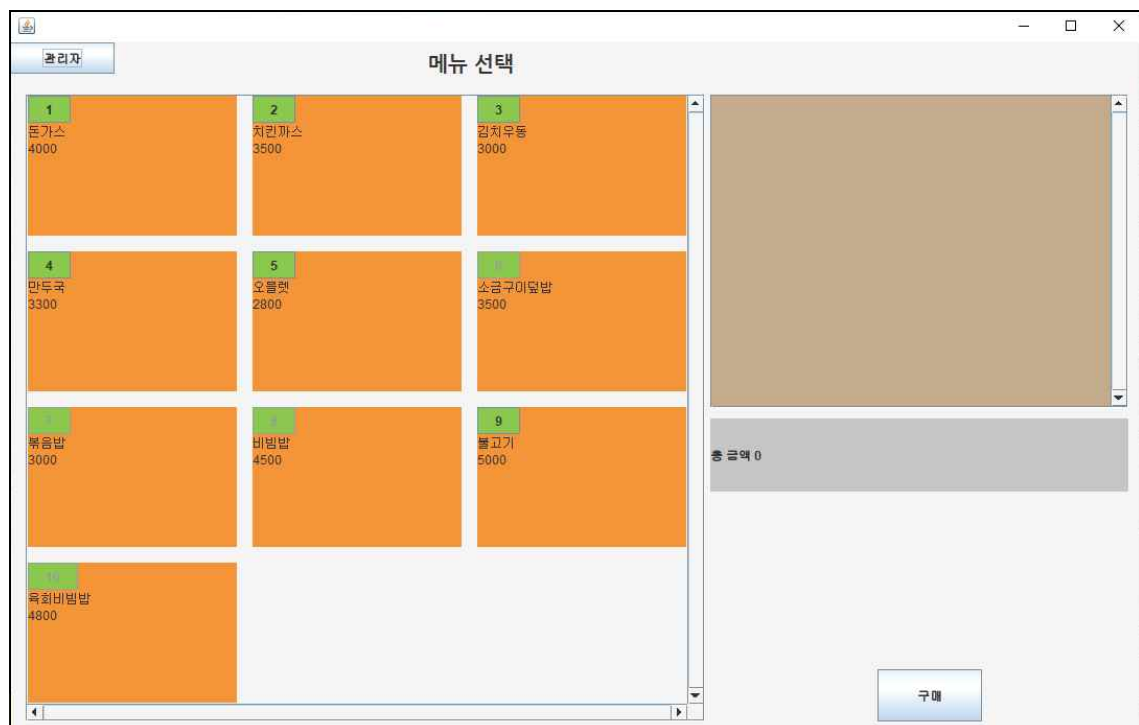


< 각 제품 버튼을 누른 후 장바구니 담긴 모습 >

< 제품이 장바구니에 담긴 후 총 금액이 변하는 모습>



< 제품의 수량을 늘린 후에 재료 부족으로 6,7,8,10 번 제품을 품절처리 한 화면 >
 <제품의 수량을 늘리면 총 금액이 변하는 모습>



< 구매 버튼 클릭 후 품절이 유지, 장바구니 및 총 금액 초기화 모습 >

2. 실행 화면 [Admin]



< Admin View 최초 실행 화면 >



< 제품 재고 현황 버튼 클릭시 나타나는 화면 >

Manage

총 매출

상품번호	제품명	제품가격	판매 갯수	총액
1	돈가스	4000	1	4000
2	소금구이덮밥	3500	3	7500

총 매출 14500

뒤로가기

< 제품 판매 총액 버튼 클릭시 나타나는 화면 >

3. Q & A에서 받은 질문들

※Q) 다중 접속을 했을 때 동시에 같은 물품을 구매하면 처리가 어떻게 되나요?

우선은 단일 클라이언트 모델로 구현되어 있습니다. 하지만 영화표 구매처럼 선택은 되더라도 구매가 안되는 코드로 작성이 되어있음.

※Q) 서버에 접속 안 되었을 때 로컬 데이터를 갖고 오나요?

인스타그램 예제는 Repository Pattern의 예시로써 Client는 서버로 연결을 위한 RemoteDataSource만 존재하고 서버에서 Repository Pattern을 적용하여 DB로 데이터를 가져올 것인지, 클라이언트에게 데이터 연결을 할 것인지 분기처리에 적용되어 있습니다.

※Q) 제품 별 재료 데이터는 어디에 저장 및 참조가 되나요?

product Table 속 PrIngredient 라는 데이터에서 각 product 별 써야하는 재료가 lgCode[Ingredient Code] - lgNumber [필요한 개수] 로 데이터베이스 속에 갖고 있습니다.

※Q) UML은 어디에 있나요?

핵심기술 설명으로 대체했습니다.

※Q) 제품 품절이 되는 메뉴들은 미리 값이 들어가는 건가요?

아닙니다. DataBase의 최신화를 통해서 하나의 값이 선택 혹은 취소될 때마다 품절 처리를 도와줍니다.

4. 개인 의견

조재영)

평소 디자인패턴에 관심은 많았고 혼자서 작업을 했던 경우는 많지만 각 부분에 대해 완벽히 분리하여 협업했던 경험은 처음이었습니다. 의존성을 최대한 분리한 결과 모듈 단위 테스트가 가능하였고, 깃허브 협업을 통해 팀원들과 조금 더 다양한 트러블 슈팅에 대처하는 방법을 배웠습니다. 비록 시간이 짧아 하고 싶었던 욕심을 다 부리지는 못했지만, 모두가 많이 배워가는 수업이었다고 생각합니다.

정유라)

MVC 패턴을 공부하면서 프레임워크 패턴에 대해 알게 되었고 팀 프로젝트에 적용하여 개발해 보면서 파트 간의 의존성이 낮을 수록 협업이 쉽다는 것을 몸소 느끼게 되었습니다. 이번 프로젝트에서 View파트를 맡았었는데 남은 방학 동안 View파트를 좀 더 보완하고 다른 팀원들이 작업한 Model파트와 Controller파트도 코드리뷰 및 추가구현해보며 공부해볼 예정입니다. 기회가 된다면 MVP나 MVVM등 다른 프레임워크를 사용한 프로그램 개발도 해보고 싶습니다.

최인호)

이번 프로젝트를 진행하면서 MVC 패턴이 무엇인지 직접적으로 알 수 있어서 좋았습니다. 이전에 진행했던 팀 프로젝트는 구현에 급급했지만, 이번에는 설계단계가 얼마나 중요한지 알게 되었습니다. 또 다른 팀들의 발표내용을 보면서 어떻게 클래스를 나누는지 다양한 시각을 볼 수 있어서 좋았습니다. 개인적으로 아쉬운 것이 있다면 데이터를 효율적으로 관리하는 측면에서 많이 부신했다고 생각합니다. 실제 코딩하기 전에 팀원들과 데이터를 주고받는 것에 대해 많은 이야기를 해야겠다고 생각했습니다. 팀장이 Git을 많이 알려줘서 2학기에 진행했던 자바 팀 프로젝트에서 썼던 것은 정말 잘 못 사용한 케이스라는 걸 깨달았고 또 이제는 어떻게 쓰는지 알게 되어서 좋았습니다. 다음에 팀 프로젝트를 진행한다면 어떻게 깃을 써야 할지 알 수 있어서 좋았습니다.

5. 실행 환경

SDK	13.0.1
DataBase	MySQL
추가한 라이브러리	Gson, mySQL Connector
Java IDE	IntelliJ

- 코틀린 코드도 있기 때문에 IntelliJ에서 실행하기를 권합니다.