



프로그램가상화 솔루션(Docker)과 클라우드(AWS) 서비스 기반 웹&앱 개발자 양성A - 정소희

2021 SEMI-PROJECT



소켓(Socket) 통신을 활용한 피자가게 키오스크 (ver 0.1.0)

주제 선정하기

소켓(Socket) 통신을 활용한 피자가게 키오스크



condition 1

소켓 통신을 이용한 서버-클라이언트, 다수 클라이언트 간의 상호작용이 있을 것.

#카카오톡 #마피아게임 #키오스크 #야구게임 #오목게임 #빙고게임



condition 2

기간 안에 완료할 수 있을 것.

#키오스크 #야구게임 #오목게임 #빙고게임



condition 3

재미있게 만들 수 있는 것.

#카카오톡 #마피아게임 #키오스크 #야구게임 #오목게임 #빙고게임

프로그램 요구사항

소켓(Socket) 통신을 활용한 피자가게 키오스크



NO	구분	화면(단계)	상세내용	비고
1	클라이언트 	공통	[cancel] 버튼 클릭 시 해당 단계까지의 모든 동작 취소 후 초기 메뉴선택 화면으로 이동	
			단계에 따라 상단 텍스트 변경	
		메뉴선택	각 메뉴 클릭 시 새로운 창에서 옵션(사이즈, 수량) 선택 후 메뉴 담기	
			담은 메뉴를 하단 주문목록에 추가. 총 수량,과 가격도 함께 출력	
			선택된 메뉴가 없을 시 다음단계로 넘어가지 못하도록	
			추가된 메뉴에 대한 수량 변경, 삭제	
			이미 주문목록에 추가된 동일 메뉴(옵션) 선택 시 새로 추가X -> 담겨진 해당 메뉴의 수량 변경	
			사이즈 선택 시 추가비용 부과	
			메뉴 담기 시 해당메뉴와 연관된 추천메뉴 선택창	
3		주문확인	메뉴선택 단계에서 선택된 메뉴들에 대한 목록을 다시 한번 출력	
4		결제	금액을 입력 받아 최종 주문 금액과 조건검사 후 결제 진행	
5		결제완료	주문번호, 주문 시간, 주문 내역 출력	* 주문완료 시 초기화면으로 이동
			- 주문번호 : 각 키오스크 마다 고유 주문번호 부여 (ex. 키오스크 1번: A1, A2... 키오스크 2번: B1, B2...)	
			- 주문시간 : 서버가 주문을 전달받은 시간	
6			주방(서버)으로 결제된 주문내역 전송 및 화면 하단에 출력	

프로그램 요구사항

소켓(Socket) 통신을 활용한 피자가게 키오스크



NO	구분	화면(단계)	상세내용	비고
7	서버 Check	주방	전달받은 주문내역 출력 (주문번호, 주문내역, 호출버튼)	
			[호출] 버튼 클릭 시 해당 주문번호 콘솔창에 호출	
			호출된 주문은 다시 호출할 수 없도록 버튼 비활성화	
			전달받은 주문내역 텍스트파일로 자동 저장(주문시간, 주문번호, 주문내역) 7일간의 데이터만 각 텍스트파일로 저장. 7개 이상일 시 가장 앞 날짜의 파일 삭제하여 7개 유지	
			각 메뉴 별 수량 설정 후 해당 메뉴 품절 시 키오스크(클라이언트)에서 품절 표기	

초기 스케치

소켓(Socket) 통신을 활용한 피자가게 키오스크



1) 클라이언트 - 메뉴 선택

클라이언트 창 1) 메뉴선택 화면

- 1) 메뉴 클릭 -> 메뉴 수량, 사이즈 선택창
- 2) 담은 내역 없을 시 -> [주문하기] 버튼 비활성화
- 3) 각 메뉴 클릭 -> 메뉴담기 창 생성
- 4) [취소하기] -> 클라이언트 창 닫음

2) 클라이언트 - 메뉴 별 옵션 선택

클라이언트 창 2) 메뉴담기 화면

- 1) 사이즈 선택 (라디오버튼) / 수량선택 (텍스트필드)
- 2) 사이즈, 수량 중 하나라도 입력 안되었을 시 [담기]버튼 비활성화
- 3) [취소] -> 메뉴담기 창 닫음(메뉴선택화면 유지)

김민준, 홍은영

3) 클라이언트 - 담은 메뉴 목록 생성

클라이언트 창 3) 메뉴 담았을 시 메뉴선택 화면

- 1) 담은 메뉴에 대한 내역이 하단에 출력
- 2) [주문하기] 버튼 활성화
- 3) [주문하기] -> 결제화면으로 이동

4) 클라이언트 - 주문내역 확인 및 결제

클라이언트 창 4) 결제화면

- 1) 주문한 메뉴내역 출력
- 2) [결제하기] -> 주문번호 부여, 서버 측에 주문내역 전달
- 3) [취소하기] -> 이전화면으로 돌아가기 (담은 메뉴 그대로 유지)

5) 클라이언트 - 결제 완료

클라이언트 창 5) 결제완료 화면

- 1) 주문내역, 부여된 주문번호 출력
- 2) [확인] -> 클라이언트 창 닫기? 홈화면으로 돌아가기?

* 주문번호 부여 - 결제완료 순서대로 주문번호 생성, 본인한테 전송

6) 서버 - 주문 전송 및 완료 호출

서버 창 6) 서버 측 Frame?

- 1) 클라이언트가 주문한 내역 출력
- 2) [호출] 버튼 -> 해당 주문번호를 서버문송창에 호출 or 해당 클라이언트에게 알림? = 진동벨

주요기능 구현 계획

소켓(Socket) 통신을 활용한 피자가게 키오스크



메뉴 선택 (클라이언트)

메뉴 옵션 선택

메뉴 옵션 Dialog 창에서 사용자에게 입력 받은 메뉴, 수량, 사이즈를
"메뉴이름(사이즈)@개수!" 형태의 String으로 변환하여 저장

중복된 주문 구별

메뉴를 추가할 때마다 HashMap에 put("메뉴이름(사이즈)", 수량)
하여 주문하려는 메뉴를 저장. 계속 메뉴를 추가할 때 key값을 통해
조건검사 진행하여 이미 담긴 메뉴가 있는지 검사

주문목록 출력

선택된 메뉴를 중복검사 후 화면 하단 주문목록에 추가
메뉴가 담길 때 마다 Label을 add
JScrollPane, JPanel을 이용한 스크롤 박스 생성

주문내역 전송 (클라이언트)

클라이언트 -> 서버 : 주문내역

클라이언트 측에서 String으로 저장한 주문내역을 서버 측에 전달
split을 활용하여 String을 가공해 서버 측에 주문내역 출력

서버 -> 클라이언트 : 주문번호, 주문시간

클라이언트에게 주문내역을 전달받았을 때의 시간을 체크
주문번호는 각 클라이언트마다 고유이름을 부여 -> 초기 클라이언트
수 세팅 필요 후 설정 ex) A, B, DT, POS
클라이언트 별 주문수 별도로 카운팅 ex) A16, B30, C7,...

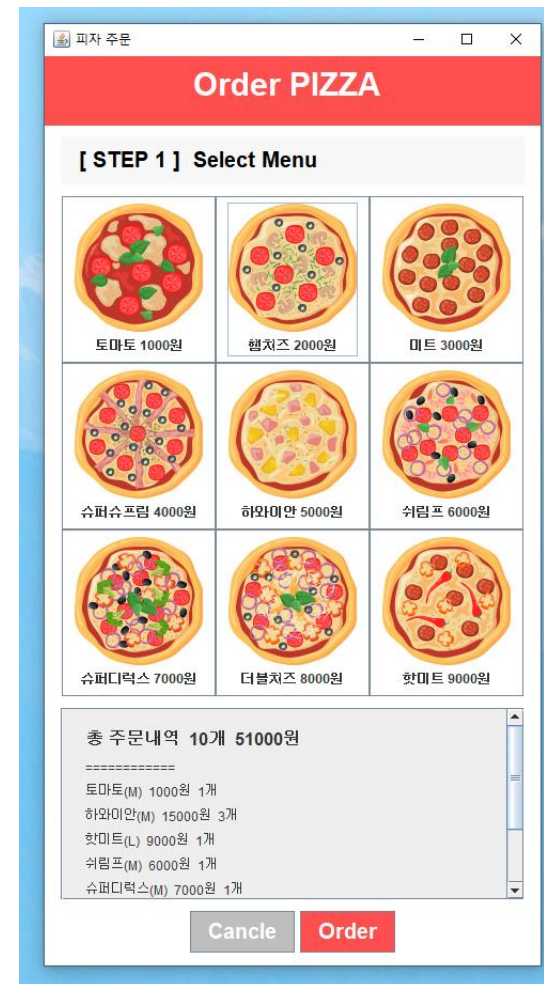
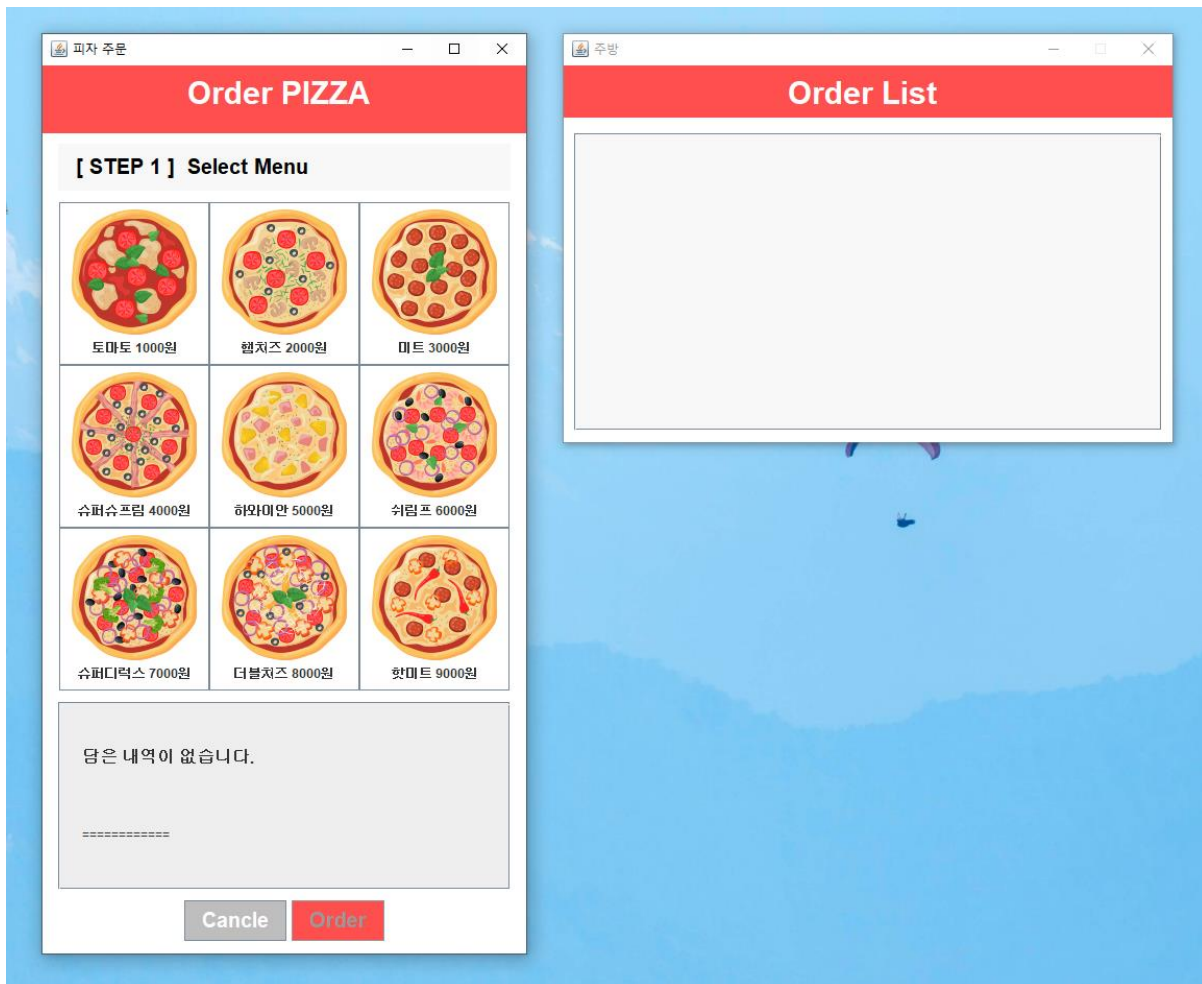
주문번호 호출 (서버)

[호출] 버튼

버튼 클릭 시 해당 주문번호 호출을 위해
버튼 생성시 버튼의 Label값을 주문번호로 설정.
버튼의 Label 텍스트를 폰트사이즈 0, 폰트컬러 투명으로 설정
-> "호출"이라는 텍스트는 이미지로 제작하여 버튼에 적용

프로그램 구현 화면

소켓(Socket) 통신을 활용한 피자가게 키오스크





피자 주문

- □ ×

Order PIZZA

[STEP 2] Confirm order

총 주문내역 10개 51000원
=====

토마토(M) 1000원 1개
하와이안(M) 15000원 3개
핫미트(L) 9000원 1개
워림프(M) 6000원 1개
슈퍼디럭스(M) 7000원 1개
더블치즈(M) 8000원 1개
미트(M) 3000원 1개
햄치즈(M) 2000원 1개

위 내역을 주문하시겠습니까?

CancePayment

피자 주문

Order PIZZA

[STEP 2] Confirm order

총 주문내역 10개 51000원

결제 진행중.

최종 결제 금액은 51000원 입니다.

결제금액을 입력해주세요 :)

결제

햄치즈(M) 2000원 1개

위 내역을 주문하시겠습니까?

Cancel Payment

피자 주문

Order PIZZA

[STEP 2] Confirm order

결제 진행중..

총 주문내역 0개 51000원

최종 결제 금액은 51000원 입니다.

결제금액을 입력해주세요 :)

51000

결제

결제금(M) 2000원 1개

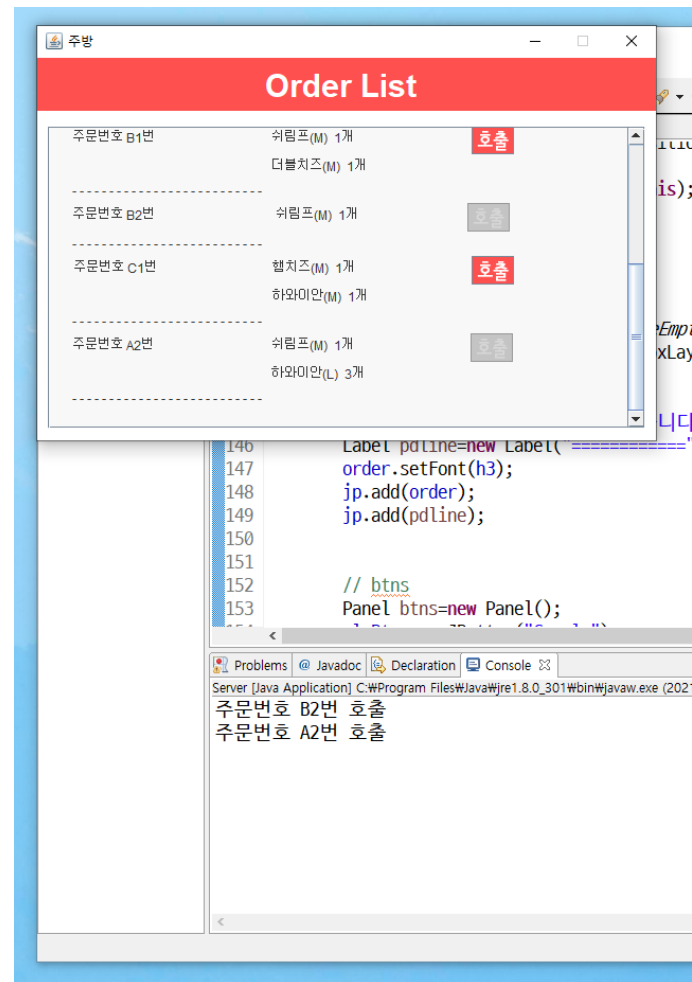
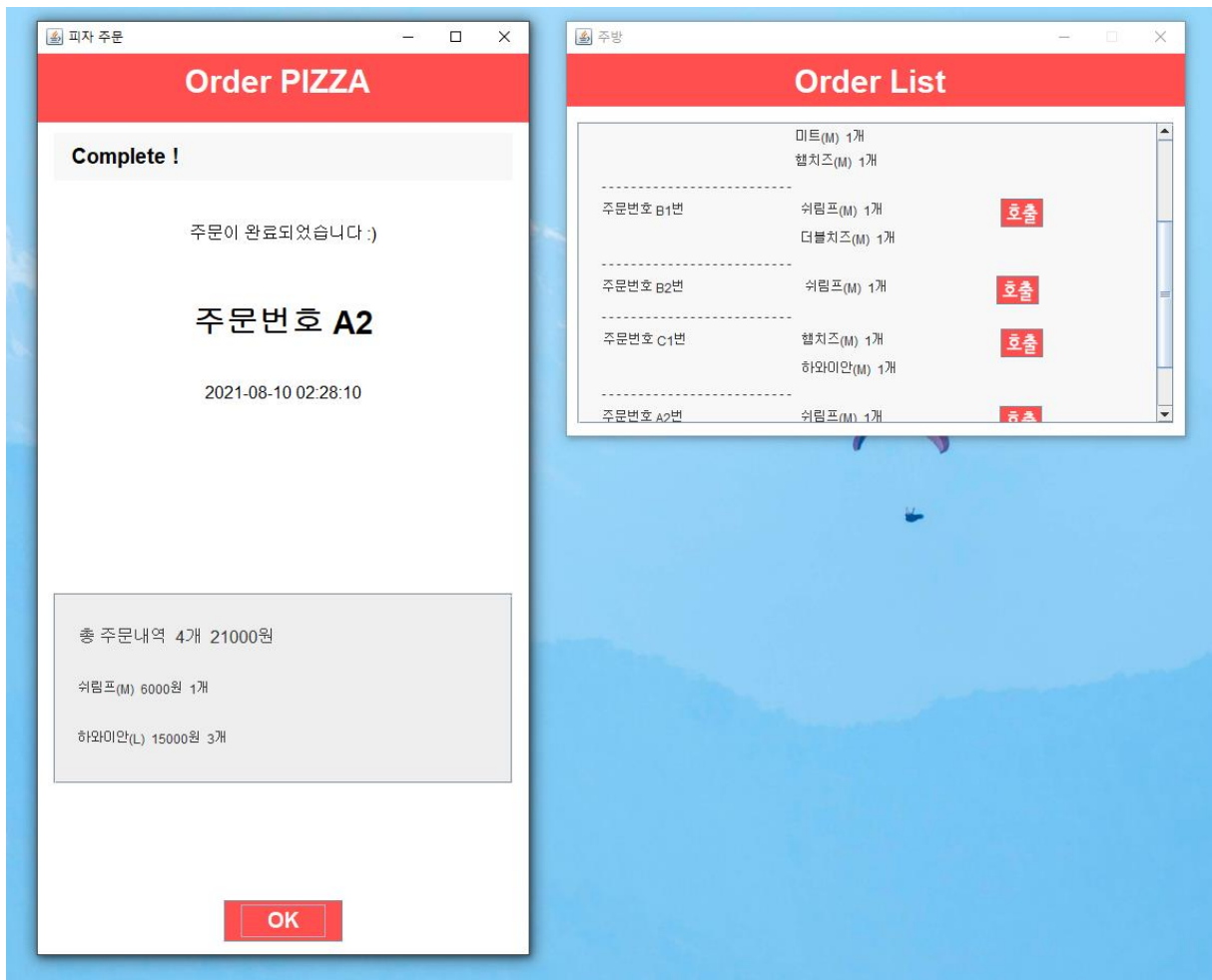
위 내역을 주문하시겠습니까?

Cancel

Payment

프로그램 구현 화면

소켓(Socket) 통신을 활용한 피자가게 키오스크



업데이트 예정 기능

소켓(Socket) 통신을 활용한 피자가게 키오스크



음료, 사이드 메뉴 탭 추가



추가된 메뉴에 대한 수량 변경, 삭제



메뉴 담기 시 해당메뉴와 연관된 추천메뉴 선택창 추가



사이즈 선택 시 추가비용 부과



각 메뉴 별 수량 설정 후 해당 메뉴 품절 시
키오스크(클라이언트)에서 품절 표기



7일간의 데이터만 각 텍스트파일로 저장
(7개 이상일 시 가장 앞 날짜의 파일 삭제하여 7개 유지)



영수증 출력



프로그램가상화 솔루션(Docker)과 클라우드(AWS) 서비스 기반 웹&앱 개발자 양성A - 정소희



- END -

