

# 테이블오더 포트폴리오 주제 선정 및 발표 정리

## 1. 프로젝트 배경

- 학습 기술 스택: Flutter, FastAPI, MySQL, SQLite
- 이전 세미 프로젝트: 신발가게 앱 (6인 팀 프로젝트)
  - 기본 상품/주문 기능 구현
- 일부 관리자 화면 구현
- **한계:** 관리자 제품 등록, 수주, 발주, 발주 결제 등 핵심 관리자 기능을 모두 구현하지 못함
  - 팀 프로젝트 특성상 전체 기능을 완성하지 못한 **미완성 상태**로 종료
  - 실시간 처리 영역까지 확장하지 못한 아쉬움 존재

## 2. 이번 포트폴리오 주제: 테이블오더

### 주제 선정 이유

- 테이블오더는 현재 시장에서 **다수의 업체가 경쟁적으로 도전 중인 서비스 영역**
- 실제 매장 도입이 계속 늘어나고 있는 구조
- 포트폴리오 주제로 선택했을 때 **관심도가 높을 수 있는 도메인**이라고 판단

핵심 관점: 이미 완성된 서비스를 그대로 만드는 것이 아니라, 성장 중인 서비스 구조를 이해하고 재현하는 데 목적이 있음

## 3. 포트폴리오로서의 목표

### 3-1. 역할이 분리된 서비스 구조

- 점주 화면
- 손님(테이블) 주문/결제 화면
- 주방 화면(KDS)

→ 단순 CRUD 앱이 아닌, 실제 서비스 흐름을 가진 구조

### 3-2. 이전 프로젝트와의 확장성

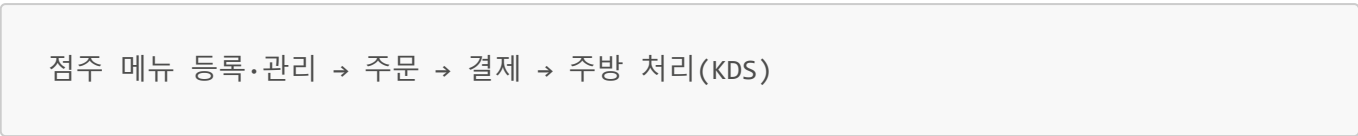
- 기존: 상품/주문 중심
- 이번 프로젝트:
  - 점주 메뉴 등록 시스템
  - 카테고리 및 메뉴 옵션 구조
  - 주문 → 결제 → 주방 처리 흐름

## 4. 구현 범위에 대한 명확한 기준

### 구현 목표

- 상용 서비스 수준의 완벽한 구현 ✕
- 제한된 포트폴리오 범위 내에서 핵심 흐름 구현 ○

핵심 흐름



확정 구현 영역

- 점주 메뉴 등록 (카테고리 / 옵션)
- 점주 화면
- 테이블 주문 화면
- 주방(KDS) 주문 처리
- MQTT 기반 실시간 주문 전달
- 토스페이먼츠 결제 연동 (샌드박스)
- UUID 기반 자동 로그인(세션)

추가 구현 목표

- 손님 테이블 예약 기능
- 예약 상태 변경에 따른 손님 대상 FCM 알림 전송

구현 제외 / 축소 범위

- 프린터 연동
- 대규모 통계 및 정산 시스템
- 복잡한 권한/조직 관리
- 실제 상용 배포 수준 기능

5. 기술 스택 정리

영역	기술	역할
프론트엔드	Flutter	점주 / 손님 / 주방 화면 구현
백엔드	FastAPI	API 제공 및 비즈니스 로직 처리
메인 DB	MySQL	최종 데이터 검증 및 저장 (Source of Truth)
로컬 스토리지	Key-Value Storage	세션 / 설정 / 소규모 캐시
실시간 처리	MQTT	주문·상태 변경 이벤트 전파
푸시 알림	Firebase Cloud Messaging (FCM)	예약·상태 변경 알림
결제	토스페이먼츠 (Sandbox)	결제 플로우 연동