📌 Odoo Cafe POS

---

1. Project Overview

This hackathon project is a Restaurant POS (Point of Sale) system called **"Od**
designed to handle:

- Restaurant table-based ordering (**Fl**
- Fast billing & checkout
- Multiple payment methods (**Cash, C**
- **Kitchen Display** integration (send c
- **Customer Display**
- POS backend configuration + report
- Optional additions: **Self/Online Ord**

---

ews dashboard and reports
andles self/online orders received in POS

---

dules / Features Breakdown

OS Backend (Configuration Area)

Authentication (Login / Signup)

OS users can create an account using **Signup**
xisting users can access the system using **Login**
ter login, the user can open POS session and access back

Product Management

al Info: Name, Category, Price, Unit, Tax, Product Descript

ts: Attribute (example: Pack), Values (6 / 12 items), Extra p

ars on the Payment page
screen exists after QR scan

Management

(example: “Ground Floor”)
ables Back-end
able Number, Seats, Active, Appointment Resou

---

nal Setup + Sessions

ted from POS Settings and includes:

sion
ale amount

→ Opens POS terminal

📌 Reporting Filters (Purpose)

- **Period:** Used to view sales/ord
  range)

💰 B4) Payment Screen

Payment screen includes:

- Total amount (example: **$580**)
- Payment methods list:
  - Cash
  - Digital / Card
  - UPI QR

After payment method selection:

- Validate payment
- Confirmation screen

📱 B5) UPI QR Payment Flow (Special Flow)

When UPI is selected:

- Show QR code screen
- Display:
  - Amount
  - “UPI QR” label
- Buttons:
  - Confirmed
  - Cancel

After confirmation:

- Payment confirmation screen appears
- Clicking anywhere dismisses it
- User returns to Floor View automatically

🖥️ B6) Customer Display

Separate display screen for customer view:

- Shows order info
- Shows payment status (paid/unpaid)
- Useful for transparency

👨‍🍳 B7) Kitchen Display

Kitchen Display shows:

- Only the products/categories that are configured to be sent to the kitchen
- Items list with quantity and item names
- Orders coming in real-time from POS “Send” action

🧾 Order Stages

- To Cook→ newly received orders
- Preparing → items currently being prepared
- Completed → ready orders

🖱️ Kitchen Actions

- Clicking a ticket/card moves the order to the next stage
- Clicking a product item marks it as prepared (strike-through)
- Ticket number is same as the Order number

---

5. Complete Flow (End-to-End)

- POS user signs up (first time) or logs in to access the system
- POS user configures POS (products, payment methods, floor/tables, displays)
- POS user opens session and selects a table
- Order is created either:
    - from POS manually, **or**
    - via mobile/self ordering using token (auto creates Order Number)
- Order is sent to Kitchen Display for preparation
- Kitchen updates order status (To Cook → Preparing → Completed)
- POS user completes payment (Cash / Digital / UPI QR)
- Reports are reviewed using filters (Period / Session / Responsible / Product)

---

6. Why This Hackathon Problem is Important

- **Real-world ERP workflow:** Shows how a complete restaurant flow works end-to-end (POS → Kitchen → Payment → Reports).
- **Business logic focus:** Teaches handling real operational problems like order lifecycle, table management, and status tracking not just UI.
- **Industry-ready system thinking:** Builds a production-like solution with real-time coordination, multi-payment support, and reporting insights.

🔗 Mockup Link

Mockup: https://link.excalidraw.com/l/65VNwvy7c4X/23T6FEnXS2l