JikoXpress Pro - Menu & Dish Management System

Version: 1.0
Target Market: East African Restaurant Industry
System Type: Kitchen Management System with Component-Based Inventory

---

1. System Overview

JikoXpress Pro is a comprehensive restaurant kitchen management system designed specifically for the East African market. The system provides component-based inventory tracking, flexible menu scheduling, comprehensive waste management, and detailed profit analytics.

Core Philosophy: Track everything at the component (dish) level, regardless of how items are sold - standalone, in combos, or custom builds.

---

2. Core Concepts

2.1 Dish (Component)

The fundamental building block of the entire system. A dish is any individual food item that can be:

- Sold standalone
- Combined into menu schedules
- Tracked in inventory
- Used in custom builds

Dish Properties

Property	Type	Description
`id`	UUID	Unique identifier

Property	Type	Description
`name`	String	Display name (e.g., "Ugali", "Maharage")
`prepCost`	BigDecimal	Cost to prepare one portion
`sellingPrice`	BigDecimal	Standalone selling price
`currentInventory`	Integer	Available portions
`canSellStandalone`	Boolean	Whether item can be sold individually
`imageUrl`	String	Required dish photo
`thumbnailUrl`	String	Auto-generated thumbnail
`active`	Boolean	Whether dish is available

Dish Examples

```
Dish: Ugali
- prepCost: 100 TSH
- sellingPrice: 300 TSH
- canSellStandalone: true

Dish: Maharage (Beans)
- prepCost: 200 TSH
- sellingPrice: 400 TSH
- canSellStandalone: true

Dish: Chai (Tea)
- prepCost: 20 TSH
- sellingPrice: 100 TSH
- canSellStandalone: true
```

Important: Dishes do NOT have time-based visibility. Time visibility is handled at the Menu Schedule level only.

2.2 Menu Schedule

Pre-defined combinations of dishes with time-based visibility and promotional pricing.

Menu Schedule Properties

Property	Type	Description
`id`	UUID	Unique identifier
`name`	String	Display name (e.g., "Lunch Special")
`components`	List<Dish>	Dishes included in this menu
`componentCost`	BigDecimal	Auto-calculated from components
`sellingPrice`	BigDecimal	Manually set (can be promotional)
`category`	Enum	MAIN_FOOD, DRINKS, SNACKS, DESSERTS
`session`	Enum	BREAKFAST, LUNCH, DINNER, ALL_DAY
`scheduleType`	Enum	ONE_TIME, DAILY, WEEKLY, PERMANENT
`menuType`	Enum	COMBO, INDIVIDUAL_ITEM
`visibleFrom`	LocalTime	Start time for visibility
`visibleUntil`	LocalTime	End time for visibility
`activeDays`	Set<DayOfWeek>	For recurring schedules
`validFrom`	LocalDate	Start date validity
`validUntil`	LocalDate	End date validity
`active`	Boolean	Master on/off switch
`heroImageUrl`	String	Optional custom combo image

Key Concepts

1. Category vs Session:

```
Category (Product Type):     Session (Time Slot):
- MAIN_FOOD                  - BREAKFAST (6am-10am)
- DRINKS                     - LUNCH (12pm-3pm)
- SNACKS                     - DINNER (6pm-9pm)
- DESSERTS                   - ALL_DAY
```

2. Schedule Types:

```
ONE_TIME:   Special event menu (e.g., "New Year Special")
DAILY:      Repeats every day within time window
WEEKLY:     Repeats on specific days (e.g., Mon-Fri only)
PERMANENT:  Always available (within time window)
```

3. Menu Types:

```
COMBO:           Multiple components (e.g., Ugali + Maharage)
INDIVIDUAL_ITEM: Single component as menu item (e.g., standalone Chai)
```

Menu Schedule Examples

Combo Menu:

```
Name: "Ugali + Maharage Special"
Components: [Ugali, Maharage]
Category: MAIN_FOOD
Session: LUNCH
Schedule Type: DAILY
Menu Type: COMBO
Visible: 12:00 - 15:00
Component Cost: 300 TSH (100 + 200)
Selling Price: 600 TSH
Profit Margin: 300 TSH (50%)
```

Individual Item Menu:

```
Name: "Chai"
Components: [Chai]
Category: DRINKS
Session: BREAKFAST
Schedule Type: DAILY
Menu Type: INDIVIDUAL_ITEM
Visible: 06:00 - 10:00
Component Cost: 20 TSH
Selling Price: 100 TSH
Profit Margin: 80 TSH
```

2.3 Selling Price Strategy

Critical Design Decision: Selling prices exist at BOTH levels.

Where Prices Are Set

Level	Purpose	Example
Dish Level	Standalone/custom build sales	Ugali = 300 TSH
Menu Schedule Level	Combo/promotional pricing	"Ugali + Maharage" = 600 TSH

Price Usage Rules

```
Flow 1: Customer orders Menu Schedule
→ Use MENU price (600 TSH)
→ Allocate revenue proportionally to components

Flow 2: Customer orders standalone dish
→ Use DISH price (300 TSH)
→ Full revenue to that dish

Flow 3: Custom build
→ Sum of DISH prices (300 + 400 = 700 TSH)
→ Each component gets full price
```

Same Components, Different Prices

```
Scenario                          Price Source    Amount
──────────────────────────────────────────────────────────
"Ugali + Maharage Special"        Menu            600 TSH
Ugali standalone                  Dish            300 TSH
Maharage standalone               Dish            400 TSH
Custom: Ugali + Maharage          Sum of Dishes   700 TSH
```

2.4 Dish Pairing (Smart Sides)

Dish pairings power the Smart Sides popup in the counter app. They define which sides appear when a main dish is selected.

Pairing Properties

Property	Type	Description

`id`	UUID	Unique identifier
`mainDish`	Dish	The main dish (e.g., Ugali)
`sideDish`	Dish	The suggested side (e.g., Maharage)
`combo`	MenuSchedule	Linked combo if exists (nullable)
`source`	Enum	COMBO, BEHAVIOR, MANUAL
`orderCount`	Integer	Times ordered together
`savings`	BigDecimal	Discount if combo exists
`displayOrder`	Integer	Sort priority in popup
`active`	Boolean	Whether to show in popup

Pairing Sources

Source	Description	Badge in UI
COMBO	Derived from existing MenuSchedules	□□ SAVE X
BEHAVIOR	Learned from customer order patterns	□□ Popular
MANUAL	Admin-defined suggestions	(none)

How Pairings Are Generated

Source 1: From Existing Combos

System automatically scans all MenuSchedules and creates pairings.

```
MenuSchedule: "Ugali Maharage" (Ugali + Maharage)
→ Creates pairing: Ugali → Maharage (source: COMBO, savings: 100)

MenuSchedule: "Ugali Nyama" (Ugali + Nyama)
→ Creates pairing: Ugali → Nyama (source: COMBO, savings: 150)
```

Source 2: From Customer Behavior

System analyzes custom build orders to find frequent pairings.

```
Analysis finds:
- Ugali + Kachumbari ordered 47 times (no combo exists)
→ Creates pairing: Ugali → Kachumbari (source: BEHAVIOR, orderCount: 47)
```

```
- Chapati + Maharage ordered 32 times (no combo exists)
→ Creates pairing: Chapati → Maharage (source: BEHAVIOR, orderCount: 32)
```

Source 3: Manual Admin Entry

Admin can manually add pairings for strategic suggestions.

```
Admin adds:
- Chips → Kachumbari (always suggest salad with chips)
→ Creates pairing: Chips → Kachumbari (source: MANUAL)
```

Popup Display Priority

When showing the Smart Sides popup, pairings are sorted:

1. COMBO pairings first (sorted by savings, highest first)
2. BEHAVIOR pairings second (sorted by orderCount, highest first)
3. MANUAL pairings last (sorted by displayOrder)

2.5 Behavior Learning System

The system continuously learns from customer orders to improve suggestions.

What It Tracks

For every custom build order, the system records:

- Which main dish was selected
- Which sides were paired with it
- Date and time of order
- Restaurant location (for multi-location)

Learning Rules

Rule	Threshold	Action
Minimum orders	10+ times	Consider for pairing
Time window	Last 30 days	Recent behavior only
Already exists	Skip	Don't duplicate pairings

Rule	Threshold	Action
Combo created	Convert	Change source to COMBO

Admin Insights

The system surfaces learned pairings to admins with recommendations:

Suggested New Combos

```
□□Ugali + Kachumbari
   Ordered together 47 times this month
   No combo exists
   Potential monthly revenue if combo created: +12,000 TSH
   [Create Combo] [Ignore] [Hide]
```

Underperforming Combos

```
⚠□ Wali + Samaki
   Only 3 orders this month
   Combo exists but rarely used
   [Promote] [Deactivate] [Keep]
```

---

3. Order Flow & Sales Tracking

3.1 Order Entity Structure

```
Order
├── id: UUID
├── orderNumber: String (auto-generated)
├── orderType: MENU_SCHEDULE | CUSTOM_BUILD | STANDALONE
├── subtotal: BigDecimal
├── discountType: PERCENTAGE | FIXED_AMOUNT | PROMO_CODE | MANAGER_OVERRIDE
├── discountValue: BigDecimal
├── discountAmount: BigDecimal
├── total: BigDecimal
├── paymentMethod: CASH | MOBILE_MONEY
```

```
├── status: PENDING | COMPLETED | CANCELLED
├── createdBy: Staff
├── createdAt: LocalDateTime
└── items: List<OrderItem>

OrderItem
├── id: UUID
├── dish: Dish
├── menuSchedule: MenuSchedule (nullable)
├── quantity: Integer
├── sourceType: MENU_SCHEDULE | CUSTOM_BUILD | STANDALONE
├── priceType: ALLOCATED | FULL_PRICE
├── originalPrice: BigDecimal
├── allocatedRevenue: BigDecimal
├── discountAllocation: BigDecimal
├── netRevenue: BigDecimal
├── prepCost: BigDecimal
└── profit: BigDecimal
```

3.2 Three Sales Flows

Flow 1: Menu Schedule Order

Customer orders a pre-defined combo.

```
Customer: "I want the Lunch Special"

Order Created:
- orderType: MENU_SCHEDULE
- Menu: "Ugali + Maharage Special" (600 TSH)

OrderItems Created:
1. Ugali
   - sourceType: MENU_SCHEDULE
   - menuScheduleId: "lunch-special"
   - priceType: ALLOCATED
   - allocatedRevenue: 257.14 TSH (proportional)
   - prepCost: 100 TSH
```

```
   - profit: 157.14 TSH

2. Maharage
   - sourceType: MENU_SCHEDULE
   - menuScheduleId: "lunch-special"
   - priceType: ALLOCATED
   - allocatedRevenue: 342.86 TSH (proportional)
   - prepCost: 200 TSH
   - profit: 142.86 TSH

Inventory Impact:
- Ugali: -1
- Maharage: -1
```

Flow 2: Standalone Order

Customer orders a single dish.

```
Customer: "Just Chai please"

Order Created:
- orderType: STANDALONE

OrderItem Created:
1. Chai
   - sourceType: STANDALONE
   - menuScheduleId: null
   - priceType: FULL_PRICE
   - allocatedRevenue: 100 TSH (full price)
   - prepCost: 20 TSH
   - profit: 80 TSH

Inventory Impact:
- Chai: -1
```

Flow 3: Custom Build Order

Counter staff builds a custom combination.

```
Staff picks: Ugali + Maharage + Nyama

Order Created:
- orderType: CUSTOM_BUILD
- Total: 300 + 400 + 600 = 1,300 TSH

OrderItems Created:
1. Ugali
   - sourceType: CUSTOM_BUILD
   - priceType: FULL_PRICE
   - allocatedRevenue: 300 TSH
   - prepCost: 100 TSH
   - profit: 200 TSH

2. Maharage
   - sourceType: CUSTOM_BUILD
   - priceType: FULL_PRICE
   - allocatedRevenue: 400 TSH
   - prepCost: 200 TSH
   - profit: 200 TSH

3. Nyama
   - sourceType: CUSTOM_BUILD
   - priceType: FULL_PRICE
   - allocatedRevenue: 600 TSH
   - prepCost: 300 TSH
   - profit: 300 TSH

Inventory Impact:
- Ugali: -1
- Maharage: -1
- Nyama: -1
```

3.3 Revenue Allocation (Proportional)

When selling menu combos, revenue is allocated proportionally based on standalone prices.

Allocation Formula

```
Component Allocation = Menu Price × (Component Standalone Price / Total Standalone Prices)
```

Allocation Example

```
Menu: "Ugali + Maharage Special"
Selling Price: 600 TSH

Components:
- Ugali standalone: 300 TSH
- Maharage standalone: 400 TSH
Total standalone: 700 TSH

Allocation:
- Ugali: 600 × (300/700) = 600 × 0.4286 = 257.14 TSH
- Maharage: 600 × (400/700) = 600 × 0.5714 = 342.86 TSH
Total: 600 TSH ✓
```

Why Proportional?

Restaurant owners need to see profit/loss per item. Proportional allocation ensures:

- Each item gets same percentage discount as the combo offers
- Fair distribution based on relative value
- Accurate item-level profitability reporting

---

4. Discount System

4.1 Discount Structure

Design Decision: Order-level discounts only (not item-level).

Discount Types

Type	Description	Example
`PERCENTAGE`	Percentage off total	10% off
`FIXED_AMOUNT`	Fixed amount off	200 TSH off
`PROMO_CODE`	Promotional code	"LUNCH10"
`MANAGER_OVERRIDE`	Manual discount with reason	Customer complaint
`STAFF_MEAL`	100% discount for staff	Employee meal
`ROUNDING`	Cash rounding adjustment	Round to 50/100

4.2 Discount Application

Discounts are distributed proportionally to all items.

Example

```
Order:
- Ugali + Maharage Special: 600 TSH
- Wali + Nyama: 800 TSH
- Chai: 100 TSH
Subtotal: 1,500 TSH

Discount: 10% = 150 TSH

Distribution (same % to each):
- Ugali + Maharage: 600 - 60 = 540 TSH (10% off)
- Wali + Nyama: 800 - 80 = 720 TSH (10% off)
- Chai: 100 - 10 = 90 TSH (10% off)
Total: 1,350 TSH ✓
```

4.3 Counter App Discount Flow

```
┌──────────────────────────────────┐
│ Current Order                    │
├──────────────────────────────────┤
│ Ugali + Maharage Special   600   │
│ Chai                       100   │
```

```
│                                  │
│ Subtotal:                  700   │
│                                  │
│ [□□Apply Discount]               │
│ [□ Complete Order]               │
└──────────────────────────────────┘

         ↓ Tap Apply Discount

┌──────────────────────────────────┐
│ Apply Discount                   │
├──────────────────────────────────┤
│ Type:                            │
│ ○ Percentage (%)                 │
│ ○ Fixed Amount (TSH)             │
│ ○ Promo Code                     │
│                                  │
│ Value: [10] %                    │
│                                  │
│ Reason: [Customer complaint___]  │
│                                  │
│ Preview:                         │
│ Subtotal:         700            │
│ Discount:         -70            │
│ ═══════════════════              │
│ Total:            630 ✓          │
│                                  │
│ [Cancel]          [Apply]        │
└──────────────────────────────────┘
```

4.4 Cash Rounding (Tanzania Specific)

Tanzania uses cash denominations (50/100 TSH coins). System handles rounding.

Rounding Options

```
Calculated Total: 315 TSH
```

```
Options:
1. Round to nearest 50: 300 TSH (customer saves 15)
2. Round to nearest 100: 300 TSH (customer saves 15)
3. Staff adjustment: Apply -15 discount
4. Credit to wallet: 15 TSH for next order
```

Mobile Money

No rounding needed - exact amounts supported.

---

5. Inventory Management

5.1 Component-Level Tracking

Core Principle: All inventory is tracked at the dish (component) level, never at menu level.

How It Works

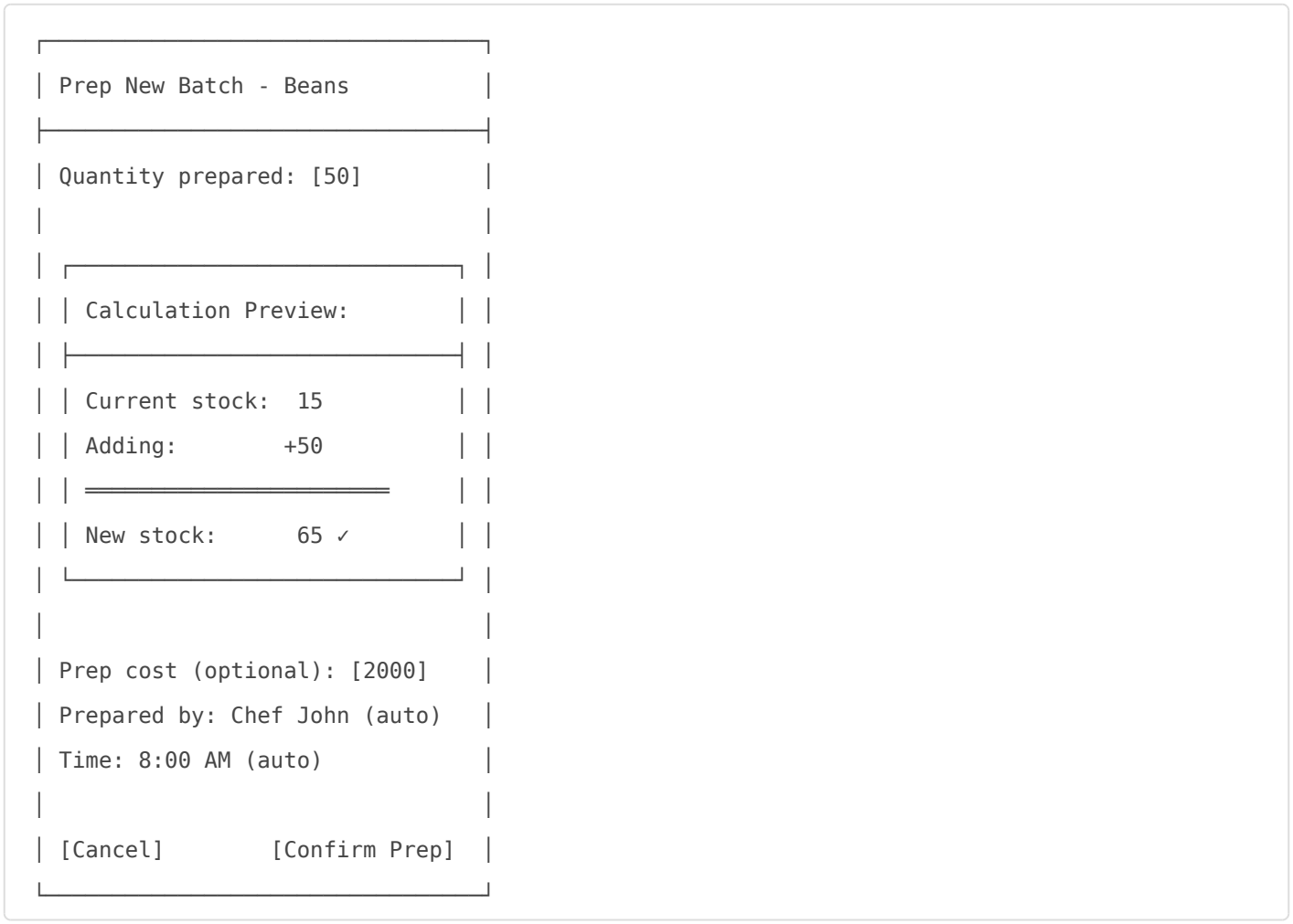
```
Sale: "Ugali + Maharage Special"

Inventory Update:
- Ugali: -1 portion
- Maharage: -1 portion

NOT:
- "Ugali + Maharage Special": -1 ← WRONG! No menu-level inventory
```

5.2 Stock Adjustment System

Three adjustment types with live calculation preview.

Action 1: Prep New Batch

Adds inventory when kitchen prepares food.

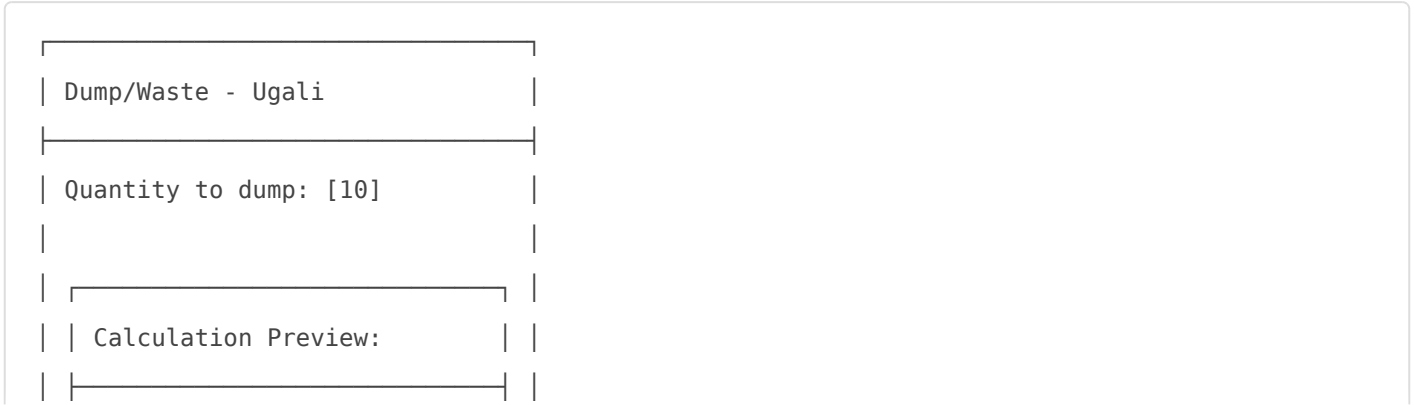


Records:

- Quantity prepared
- Prep cost (optional)
- Who prepared
- Timestamp

Action 2: Dump/Waste

Removes inventory with waste tracking.



```
| | Current stock:  65          | |
| | Removing:      -10          | |
| | ═══════════════════         | |
| | New stock:      55 ✓        | |
| | Waste value:   400 □□       | |
| └─────────────────────────────┘ |
|                                 |
| Reason:                         |
| ○ Spoiled/Expired               |
| ● Poor quality                  |
| ○ Overcooked                    |
| ○ Dropped/Accident              |
| ○ End of day dump               |
| ○ Other: [___________]          |
|                                 |
| Dumped by: Chef Mary (auto)     |
| Time: 2:30 PM (auto)            |
|                                 |
| [Cancel]        [Confirm Dump]  |
└─────────────────────────────────┘
```

Waste Reasons:

- Spoiled/Expired
- Poor quality
- Overcooked
- Dropped/Accident
- End of day dump
- Other (free text)

Action 3: Manual Adjust

General corrections for edge cases.

```
┌─────────────────────────────────┐
| Manual Adjust - Rice            |
├─────────────────────────────────┤
| Adjustment type:                |
| ● Add inventory                 |
| ○ Remove inventory              |
|                                 |
```

```
| Quantity: [5]                    |
|                                  |
| ┌──────────────────────────────┐ |
| | Calculation Preview:         | |
| ├──────────────────────────────┤ |
| | Current stock:  55           | |
| | Adding:         +5           | |
| | ═══════════════════          | |
| | New stock:      60 ✓         | |
| └──────────────────────────────┘ |
|                                  |
| Reason:                          |
| ● Recount correction             |
| ○ Found missing stock            |
| ○ System error                   |
| ○ Transfer from storage          |
| ○ Other: [____________]          |
|                                  |
| [Cancel]   [Confirm Adjustment]  |
└──────────────────────────────────┘
```

Adjustment Reasons:

- Recount correction
- Found missing stock
- System error
- Transfer from storage
- Other (free text)

5.3 Error Prevention

Cannot Remove More Than Available

```
Quantity to dump: [100]


┌──────────────────────────────────┐
| Calculation Preview:             |
├──────────────────────────────────┤
| Current stock:  60               |
```

```
| Removing:     -100              |
| ═══════════════════             |
| New stock:     -40 □            |
|                                 |
| ⚠□ Cannot remove more than      |
|    available!                   |
| Max: 60 portions                |
└─────────────────────────────────┘


[Confirm] ← DISABLED
```

Large Adjustment Warning

```
If dumping > 15 portions OR value > 500 TSH:


┌─────────────────────────────────┐
| ⚠□ Confirm Large Waste          |
├─────────────────────────────────┤
| Dumping: 50 portions            |
| Value: 2,000 TSH                |
| Reason: End of day dump         |
|                                 |
| This is a significant loss.     |
| Please confirm.                 |
|                                 |
| [Cancel]        [Yes, Dump]     |
└─────────────────────────────────┘
```

5.4 Permission Recommendations

Action	Kitchen Staff	Manager
Prep New Batch	✓	✓
Dump/Waste	✓	✓
Manual Adjust	✗	✓

---

6. Counter App Interface

6.1 Smart Sides Flow (Core UX)

The counter app uses a unified, intelligent flow that eliminates the need for staff to choose between "combos" or "custom builds". Staff simply taps what the customer orders, and the system handles combo detection automatically.

Design Philosophy

Old Approach	New Smart Sides Approach
Staff chooses: Combo OR Custom OR Standalone	One natural flow for everything
Must memorize which combos exist	System suggests and auto-applies
Three different paths to learn	"Tap main, pick sides, done"
Easy to pick wrong option	Cannot go wrong

The Flow

Step 1: Staff sees main dishes only

No combo clutter. Just the base items organized by category.

```
│                                                    │
└────────────────────────────────────────────────────┘
```

Step 2: Staff taps main dish → Smart Sides popup appears

System shows all possible sides, with combo savings highlighted.

```
┌────────────────────────────────────────────────────┐
│ Add Sides to Ugali?                          [x]   │
├────────────────────────────────────────────────────┤
│                                                    │
│ ┌──────────────────────────────────────────────┐   │
│ │ Maharage                 +300   □□SAVE 100 │ │
│ │                            Combo available! │ │
│ └──────────────────────────────────────────────┘   │
│                                                    │
│ ┌──────────────────────────────────────────────┐   │
│ │ Nyama Choma              +500   □□SAVE 150 │ │
│ │                            Combo available! │ │
│ └──────────────────────────────────────────────┘   │
│                                                    │
│ ┌──────────────────────────────────────────────┐   │
│ │ Kachumbari               +150   □□Popular  │ │
│ │                            47 orders        │ │
│ └──────────────────────────────────────────────┘   │
│                                                    │
│ ┌──────────────────────────────────────────────┐   │
│ │ Sukuma Wiki              +200               │ │
│ └──────────────────────────────────────────────┘   │
│                                                    │
│      [+ Custom Side]       [Skip → No Sides]       │
│                                                    │
└────────────────────────────────────────────────────┘
```

Step 3: System auto-detects and applies combo pricing

```
┌────────────────────────────────────────────────────┐
│ Cart                                               │
├────────────────────────────────────────────────────┤
│                                                    │
```

```
| Ugali Maharage                  x1        600   |
| ├─ Ugali                                        |
| └─ Maharage                    □□Saved 100!     |
|                                                 |
| ─────────────────────────────────────────────   |
| Subtotal:                                600   |
|                                                 |
| [+ Add More]                   [Checkout →]     |
|                                                 |
└─────────────────────────────────────────────────┘
```

6.2 Smart Sides Popup Logic

What Appears in the Popup

The popup shows sides from three sources, in priority order:

Priority	Source	Badge	Description
1	Existing Combos	□□ SAVE X	Sides that complete a predefined combo
2	Customer Behavior	□□ Popular	Frequently ordered together (learned)
3	Manual Pairings	(none)	Admin-defined suggestions

Popup Display Rules

- Show combo sides first (sorted by savings amount)
- Show popular/learned sides second (sorted by order count)
- Show manual pairings last
- Maximum 6-8 sides visible (scroll for more)
- Always show "Custom Side" and "Skip" options

6.3 Cart Intelligence

The system automatically determines the best pricing based on what's selected.

Scenario 1: Side Completes a Combo

```
Staff picks: Ugali → Maharage

Cart shows:
┌─────────────────────────────────────────────┐
│ Ugali Maharage                x1       600   │
│                               □□Saved 100!   │
└─────────────────────────────────────────────┘

Internally tracked as: MENU_SCHEDULE order
Revenue allocation: Proportional to components
```

Scenario 2: No Sides Selected (Skip)

```
Staff picks: Ugali → Skip

Cart shows:
┌─────────────────────────────────────────────┐
│ Ugali                         x1       300   │
└─────────────────────────────────────────────┘

Internally tracked as: STANDALONE order
Revenue: Full price to Ugali
```

Scenario 3: Sides Don't Match Any Combo

```
Staff picks: Ugali → Kachumbari (no combo exists)

Cart shows:
┌─────────────────────────────────────────────┐
│ Ugali                         x1       300   │
│ Kachumbari                    x1       150   │
└─────────────────────────────────────────────┘

Internally tracked as: CUSTOM_BUILD order
Revenue: Full standalone prices
```

Scenario 4: Multiple Mains

```
Customer: "Ugali na Wali, zote na maharage"

Staff flow:
1. Tap Ugali → Pick Maharage → Cart: "Ugali Maharage x1 - 600"
2. Tap Wali → Pick Maharage → Cart adds: "Wali Maharage x1 - 650"

Cart shows:
┌──────────────────────────────────────────────┐
│ Ugali Maharage                x1        600  │
│ Wali Maharage                 x1        650  │
│ ──────────────────────────────────────────   │
│ Subtotal:                              1,250  │
└──────────────────────────────────────────────┘

Each main is a separate cart item with its own sides flow.
```

Scenario 5: Drinks as Main

```
Customer: "Chai mbili tu"

Staff flow:
1. Tap Chai → Skip (or popup shows no relevant sides)
2. Adjust quantity to 2

Cart shows:
┌──────────────────────────────────────────────┐
│ Chai                          x2        200  │
└──────────────────────────────────────────────┘

Drinks are treated as main dishes. They can have their own
sides popup (e.g., Chai + Mandazi combo) or be skipped.
```

6.4 Quantity Handling

Quantity applies to the entire combo, not individual components.

```
Customer: "Ugali maharage mbili"

Cart shows:

┌──────────────────────────────────────────────┐
│ Ugali Maharage                  x2      1,200  │
│                                 □□Saved 200!  │
└──────────────────────────────────────────────┘

This means:
- 2x Ugali portions
- 2x Maharage portions
- Combo price × 2
```

6.5 Cart & Checkout

```
┌──────────────────────────────────────────────┐
│ Current Order                                │
├──────────────────────────────────────────────┤
│                                              │
│ Ugali Maharage                  x1        600  │
│                                 □□Saved 100   │
│                                              │
│ Wali Nyama                      x1        800  │
│                                 □□Saved 150   │
│                                              │
│ Chai                            x2        200  │
│                                              │
│ ──────────────────────────────────────────   │
│ Subtotal:                                1,600  │
│ Discount (10%):                           -160  │
│ ══════════════════════════════════════════   │
│ Total:                                   1,440  │
│                                              │
│ Total Saved from Combos:                   250  │
│                                              │
│ [□□Apply Discount]                           │
│                                              │
```

```
| Payment Method:                                  |
| [□□Cash]        [□□Mobile Money]              |
|                                                  |
| [□ Complete Order]                              |
|                                                  |
└──────────────────────────────────────────────────┘
```

6.6 Staff Training Summary

The entire training can be reduced to:

```
"Tap the main dish customer wants.
 Pick sides from the popup if they want any.
 If they want it plain, tap Skip.
 System handles the rest."
```

No need to memorize combos. No need to choose between different order types. Just tap what you hear.

---

7. Customer App Flow

7.1 Time-Based Menu Display

Customer app shows only menus currently visible based on time.

```
8:00 AM Customer View:


┌──────────────────────────────────────────────────┐
| □□Breakfast Menu                                |
├──────────────────────────────────────────────────┤
|                                                  |
| COMBO DEALS                                      |
| ┌──────────────────────────────────────────────┐ |
| | □□Mandazi + Chai                 200 TSH   | |
| |    □□Save 20!                               | |
```

```
| ├──────────────────────────────────────────┤ |
| | □□Ugali + Maharage              350 TSH   | |
| |    □□Save 50!                             | |
| └──────────────────────────────────────────┘ |
|                                              |
| INDIVIDUAL ITEMS                             |
| ┌──────────────────────────────────────────┐ |
| | Mandazi                          120 TSH   | |
| | Chai                             100 TSH   | |
| | Ugali                            200 TSH   | |
| └──────────────────────────────────────────┘ |
|                                              |
└──────────────────────────────────────────────┘
```

7.2 Standalone Item Solution

Problem: Customer wants just Mandazi, but only "Mandazi + Chai" combo exists.

Solution: Create both combo AND individual menu schedules.

```
Breakfast Menus (6:00 - 10:00):

1. "Mandazi + Chai" (combo)     = 200 TSH
2. "Mandazi" (individual)       = 120 TSH
3. "Chai" (individual)          = 100 TSH

Customer can choose:
- Combo (save 20 TSH)
- Just Mandazi
- Just Chai
```

---

8. Reports & Analytics

8.1 Daily Sales Report

```
═══════════════════════════════════════════════════════════
        DAILY SALES REPORT - January 15, 2025
═══════════════════════════════════════════════════════════

SUMMARY:
─────────────────────────────────────────────────────────

Total Orders:              45
Gross Sales:           50,000 TSH
Discounts Given:       -3,500 TSH (7%)
Net Sales:             46,500 TSH
Total Food Cost:       18,600 TSH
Gross Profit:          27,900 TSH
Profit Margin:            60%

DISCOUNT BREAKDOWN:
─────────────────────────────────────────────────────────

Manager Override:       2,000 TSH
Promo Codes:           1,000 TSH
Staff Meals:             500 TSH

PAYMENT METHODS:
─────────────────────────────────────────────────────────

Cash:                  30,000 TSH (64%)
Mobile Money:          16,500 TSH (36%)
```

8.2 Item Sales Report

```
═══════════════════════════════════════════════════════════
         ITEM SALES REPORT - January 15, 2025
═══════════════════════════════════════════════════════════

Item              Qty    Revenue    Cost      Profit    Margin
─────────────────────────────────────────────────────────────

Ugali              45     11,571    4,500     7,071     61.1%
Maharage           52     13,943    10,400    3,543     25.4%
Wali               35      8,750     3,500    5,250     60.0%
Nyama              28     15,736     8,400    7,336     46.6%
```

```
Chai               30     3,000       600    2,400     80.0%
──────────────────────────────────────────────────────────────────
TOTAL             190    53,000    27,400   25,600     48.3%

TOP PERFORMERS (by profit):
1. Nyama:     7,336 TSH (28 sold)
2. Ugali:     7,071 TSH (45 sold)  High volume
3. Wali:      5,250 TSH (35 sold)

ATTENTION NEEDED:
1. Maharage: 25.4% margin ⚠
   - High volume but lowest margin
   - Consider: raise prices or reduce prep cost
```

8.3 Item Deep Dive Report

```
═══════════════════════════════════════════════════════
           UGALI - DETAILED P&L REPORT
═══════════════════════════════════════════════════════

Quantity Sold: 45 portions

REVENUE BREAKDOWN:
──────────────────────────────────────────────────────

From Menu "Ugali + Maharage Special":
  25 orders × 257.14 allocated         =    6,428.50

From Menu "Ugali + Nyama Special":
  15 orders × 240.00 allocated         =    3,600.00

From Standalone:
  5 orders × 300.00 full price         =    1,500.00

From Custom Builds:
  (various allocations)                =       42.50
──────────────────────────────────────────────────────

Total Revenue:                              11,571.00
Total Cost:                                  4,500.00 (45 × 100)
```

```
═══════════════════════════════════════════════════════
Gross Profit:                                  7,071.00
Profit Margin:                                    61.1%

Average Revenue per Portion:                     257.13
Cost per Portion:                                100.00
Average Profit per Portion:                      157.13
```

8.4 Waste Report

```
═══════════════════════════════════════════════════════
           WASTE REPORT - January 15, 2025
═══════════════════════════════════════════════════════

BY COMPONENT:
─────────────────────────────────────────────────────

Beans: 5 portions (200 TSH cost)
  • Spoiled:       3 portions
  • Poor quality:  2 portions

Ugali: 12 portions (1,200 TSH cost)
  • End of day:   12 portions

Rice: 3 portions (150 TSH cost)
  • Dropped:       3 portions
─────────────────────────────────────────────────────

TOTAL WASTE: 20 portions
TOTAL COST:  1,550 TSH
% OF FOOD COST: 2.8%

WASTE BY REASON:
─────────────────────────────────────────────────────

End of day dump:     12 portions  (60%)
Spoiled:              3 portions  (15%)
Poor quality:         2 portions  (10%)
Dropped:              3 portions  (15%)
```

8.5 Prep Report

```
═══════════════════════════════════════════════════════════

          PREP REPORT - January 15, 2025

═══════════════════════════════════════════════════════════

TIME     STAFF        ITEM       QTY    COST
─────────────────────────────────────────────────────────
08:00    Chef John    Beans       50    2,000
08:30    Chef John    Ugali       30    1,500
09:00    Chef Mary    Rice        40    1,600
12:00    Chef John    Beans       25    1,000
14:30    Chef Mary    Ugali       20    1,000
─────────────────────────────────────────────────────────
TOTAL PREP COST:                        7,100

BY CHEF:
Chef John:   105 portions   4,500 TSH
Chef Mary:    60 portions   2,600 TSH
```

8.6 Menu Engineering Matrix

```
═══════════════════════════════════════════════════════════

              MENU ENGINEERING MATRIX

═══════════════════════════════════════════════════════════

              | High Profit          | Low Profit
──────────────┼──────────────────────┼──────────────────
High Volume   | □ STARS              | □□WORKHORSES
              | Nyama, Ugali         | Maharage
              | → Promote heavily    | → Fix pricing
──────────────┼──────────────────────┼──────────────────
Low Volume    | □□PUZZLES            | □□DOGS
              | Wali                 | (none currently)
              | → Test promotions    | → Consider removing
──────────────┴──────────────────────┴──────────────────
```

```
RECOMMENDATIONS:
──────────────────────────────────────────────────────────

 Nyama: Keep promoting - high margin, growing volume
 Ugali: Maintain quality - consistent performer
Maharage: Increase price to 450 or reduce prep cost
Wali: Run "Wali Specials" to boost volume
```

8.7 Audit Trail

```
═══════════════════════════════════════════════════════════

           ADJUSTMENT AUDIT - January 15, 2025

═══════════════════════════════════════════════════════════

TIME     STAFF        ACTION      ITEM     QTY   REASON
──────────────────────────────────────────────────────────
08:00    Chef John    PREP        Beans    +50   Morning prep
08:30    Chef John    PREP        Ugali    +30   Morning prep
14:30    Chef Mary    DUMP        Ugali     -3   Poor quality
16:00    Chef Mary    DUMP        Beans     -2   Spoiled
20:00    Manager      ADJUST      Beans     -2   Recount
20:30    Chef John    DUMP        Ugali    -12   End of day
──────────────────────────────────────────────────────────

LARGE ADJUSTMENTS (>15 portions or >500 TSH):
• 20:30 - Ugali dump: 12 portions (1,200 TSH)
  Approved by: Manager
  Reason: End of day - remaining portions
```

---

9. Image Management

9.1 Strategy

Entity	Image Requirement
Dish	**REQUIRED** - all dishes must have image

Entity	Image Requirement
Menu Schedule	**OPTIONAL** - custom hero image
Fallback	Use component images if no menu image

9.2 Image Sizes

```
Original:    1200 × 1200 (storage)
Large:       800 × 800 (desktop detail view)
Medium:      600 × 600 (mobile, listings)
Thumbnail:   200 × 200 (cart, small displays)
```

9.3 Image Requirements

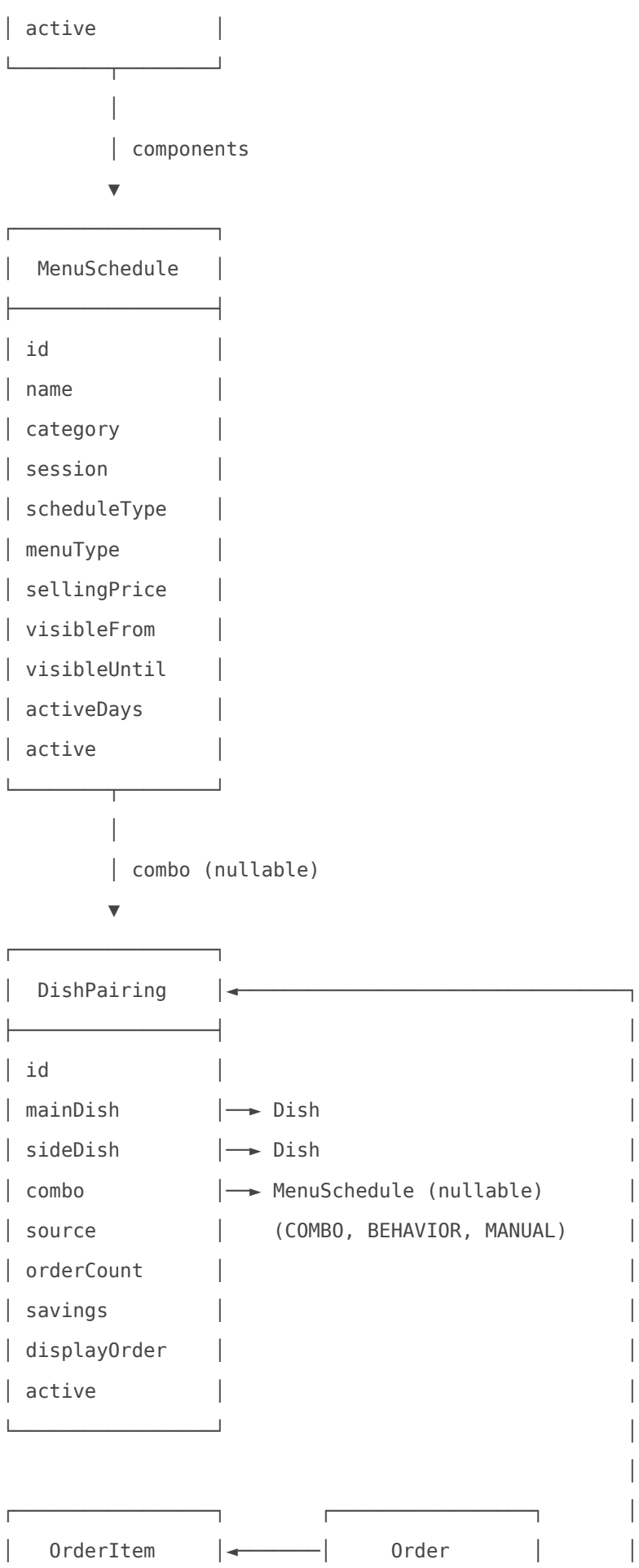
```
Format:      JPEG or PNG (WebP preferred)
Max Size:    5MB
Min Size:    400 × 400
Quality:     80-85% compression
```

---

10. Data Models Summary

10.1 Complete Entity Diagram

```
┌──────────────────┐
│      Dish        │
├──────────────────┤
│ id               │
│ name             │
│ prepCost         │
│ sellingPrice     │
│ currentInventory │
│ canSellStandalone│
│ imageUrl         │
```


active
components
MenuSchedule
id
name
category
session
scheduleType
menuType
sellingPrice
visibleFrom
visibleUntil
activeDays
active
combo (nullable)
DishPairing
id
mainDish
Dish
sideDish
Dish
combo
MenuSchedule (nullable)
source
(COMBO, BEHAVIOR, MANUAL)
orderCount
savings
displayOrder
active
OrderItem
Order

10.2 Enums

Menu & Scheduling

Enum	Values
MenuCategory	MAIN_FOOD, DRINKS, SNACKS, DESSERTS
FoodSession	BREAKFAST, LUNCH, DINNER, ALL_DAY
ScheduleType	ONE_TIME, DAILY, WEEKLY, PERMANENT

Enum	Values
MenuType	COMBO, INDIVIDUAL_ITEM

Orders & Sales

Enum	Values
OrderType	MENU_SCHEDULE, CUSTOM_BUILD, STANDALONE
SourceType	MENU_SCHEDULE, CUSTOM_BUILD, STANDALONE
PriceType	ALLOCATED, FULL_PRICE
OrderStatus	PENDING, COMPLETED, CANCELLED
PaymentMethod	CASH, MOBILE_MONEY

Discounts

Enum	Values
DiscountType	PERCENTAGE, FIXED_AMOUNT, PROMO_CODE, MANAGER_OVERRIDE, STAFF_MEAL, ROUNDING

Inventory & Adjustments

Enum	Values
AdjustmentType	PREP, DUMP, MANUAL_ADD, MANUAL_REMOVE
WasteReason	SPOILED_EXPIRED, POOR_QUALITY, OVERCOOKED, DROPPED_ACCIDENT, END_OF_DAY, OTHER
AdjustmentReason	RECOUNT_CORRECTION, FOUND_MISSING, SYSTEM_ERROR, TRANSFER_FROM_STORAGE, OTHER

Dish Pairings

Enum	Values
PairingSource	COMBO, BEHAVIOR, MANUAL

---

11. Key System Benefits

For Restaurant Owners

- Real-time inventory - always know what's available
- Item-level profitability - see profit per dish
- Waste tracking - understand where money is lost
- Staff accountability - track who does what
- Menu optimization - identify stars and dogs
- Behavior insights - see what customers actually want
- Combo suggestions - system recommends new combos based on data

For Kitchen Staff

- Simple prep tracking - one tap to log
- Easy waste logging - categorized reasons
- Live inventory - no surprise stockouts
- Clear audit trail - protect from blame

For Counter Staff

- One simple flow - tap main, pick sides, done
- No memorization - system suggests combos
- Auto pricing - combos detected automatically
- Flexible discounts - handle any situation
- Clear totals - live calculation preview
- Multiple payment - cash and mobile money

For Customers

- Faster service - staff doesn't fumble with options
- Always get best price - combos auto-applied
- Consistent experience - same flow every time

For System

- Component-level truth - single source for inventory
- Proportional allocation - fair revenue distribution
- Complete audit - every action logged
- Learning engine - improves suggestions over time
- Scalable - handles growth from 10 to 10,000 orders

12. Implementation Checklist

Phase 1: Core Setup

- ☐ Dish entity with inventory
- ☐ Menu Schedule entity
- ☐ Category and Session enums
- ☐ Basic CRUD operations

Phase 2: Smart Sides

- ☐ Dish Pairing entity
- ☐ Auto-generate pairings from combos
- ☐ Pairing display priority logic
- ☐ Smart Sides popup component

Phase 3: Orders

- ☐ Order entity
- ☐ OrderItem with source tracking
- ☐ Cart intelligence (combo detection)
- ☐ Revenue allocation service
- ☐ Discount application

Phase 4: Inventory

- ☐ Stock adjustment system
- ☐ Prep tracking
- ☐ Waste management
- ☐ Audit trail

Phase 5: Counter App

- [ ] Main dishes display
- [ ] Smart Sides popup
- [ ] Cart with combo badges
- [ ] Checkout flow
- [ ] Discount UI

Phase 6: Behavior Learning

- [ ] Order pattern tracking
- [ ] Pairing analysis service
- [ ] Admin insights dashboard
- [ ] Combo suggestions

Phase 7: Reports

- [ ] Daily sales summary
- [ ] Item profitability
- [ ] Waste reports
- [ ] Prep reports
- [ ] Menu engineering
- [ ] Pairing performance

Phase 8: Customer App

- [ ] Time-based menu display
- [ ] Cart functionality
- [ ] Payment integration
- [ ] Order history

Document Version: 1.0
Last Updated: December 2024
Author: JikoXpress Development Team

Revision #2
Created 22 December 2025 09:57:58 by Admin Qbit
Updated 22 December 2025 11:37:31 by Admin Qbit