JikoXpress Pro: Checkout & Order Management Architecture

Document Information

Property	Value
Version	2.1
Status	Draft
Created	December 2025
Updated	January 2026
Purpose	Development Guide & Requirements Reference

Executive Summary

Vision

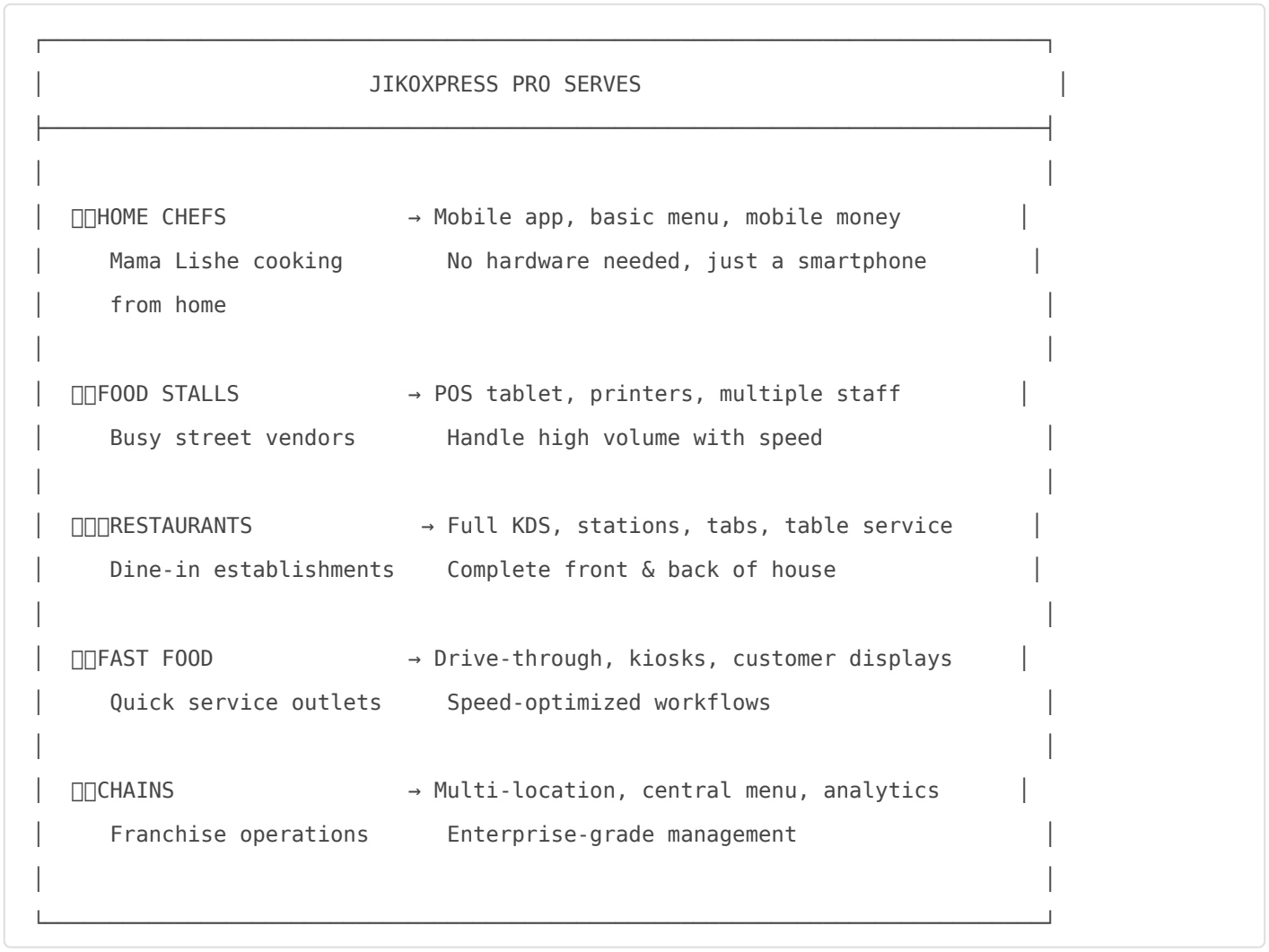
JikoXpress Pro is a unified restaurant management platform designed for the East African market, enabling food businesses of any size to accept orders, manage kitchen operations, and grow their business - all from a single system.

What is JikoXpress Pro?

JikoXpress Pro is a complete checkout and order management system that handles:

- **Order Capture** - From 7 different sales channels (POS, Kiosk, Mobile App, WhatsApp, Table QR, Drive-Through, Direct Counter)
- **Payment Processing** - Mobile money (USSD), cash, card, digital wallets, and pay-later tabs
- **Kitchen Operations** - Station-based routing, KDS displays, ticket printing, and expeditor control
- **Delivery Management** - Multi-provider integration with real-time tracking
- **Business Intelligence** - Sales reports, customer insights, and operational metrics

Who Is It For?



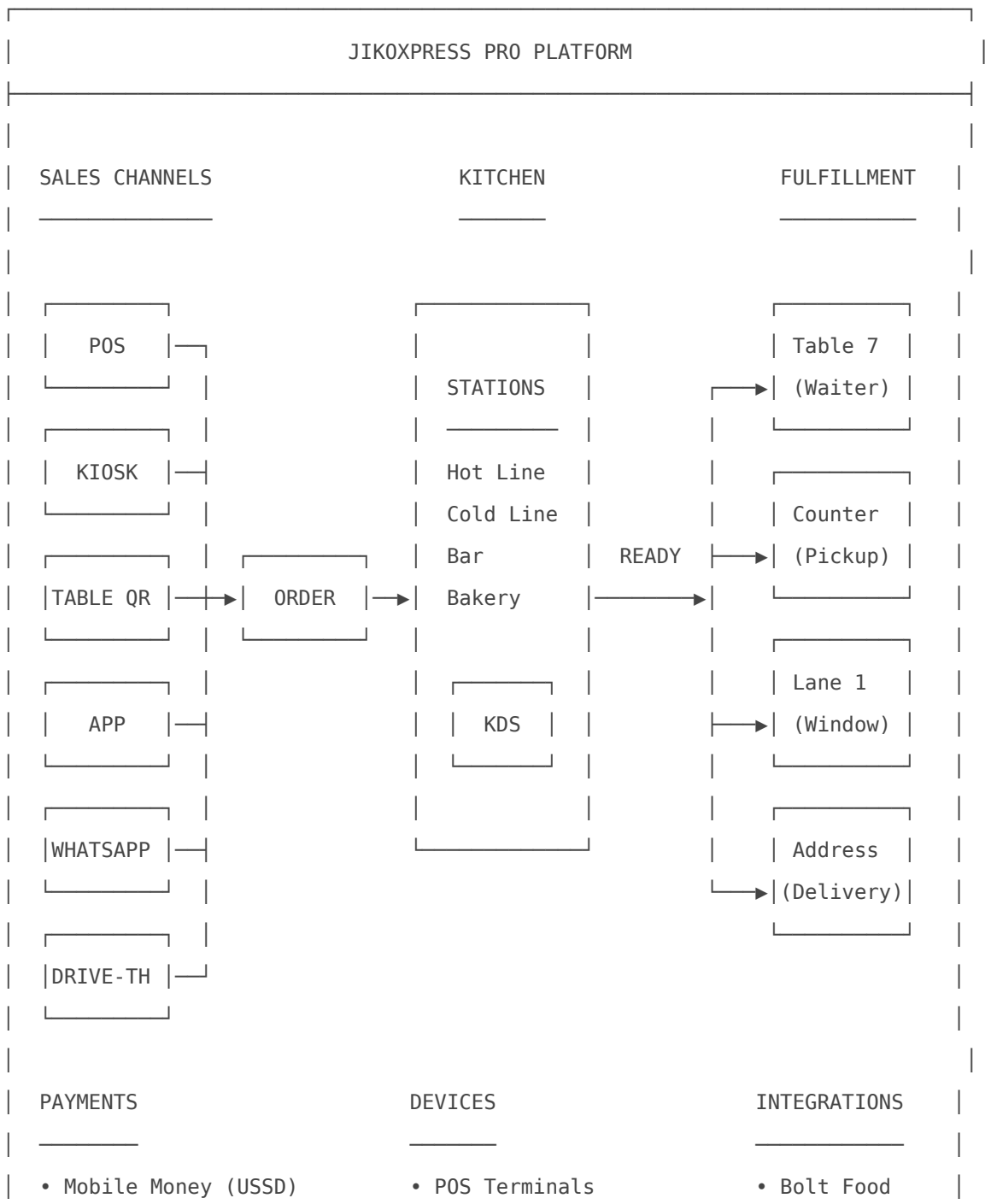
Core Architecture Principles

The architecture is built on five core principles:

Principle	Description
Single Source of Truth	All channels (POS, Kiosk, App, WhatsApp, etc.) create the same Order entity. Kitchen sees one unified queue regardless of where orders originate.
Progressive Feature Unlocking	Start free with basic features. Unlock POS, stations, KDS, drive-through, and more as your business grows. Same platform from day one to 100 locations.
Zero-Config Defaults	Works immediately after signup. Set your menu, start taking orders. No complex setup required. Customize only what you need.

Principle	Description
Device-Agnostic Operations	Register any device (phone, tablet, dedicated terminal) as POS, Kiosk, or KDS. Configure each device's role and capabilities independently.
Channel-Flexible Menu	One menu, configurable per channel. Set different prices for delivery, hide items from kiosks, restrict items by fulfillment type.

System Overview



```
│  • Cash                    • Kiosks                 • Uber Direct   │
│  • Card                    • KDS Displays           • SafeBoda      │
│  • App Wallet              • Waiter Tablets         • JikoXpress    │
│  • Kitchen Wallet          • Customer Displays        Dashers       │
│  • Tabs (Pay Later)        • Printers                               │
│                                                                     │
└─────────────────────────────────────────────────────────────────────┘
```

Subscription Tiers

Tier	Target User	Key Features	Limits
STARTER	Home chefs, vendors	Mobile app, basic menu, mobile money	20 items, 100 orders/mo
GROWING	Small restaurants	+ POS, Kiosk, printers, card payments	1,000 orders/mo, 3 staff
PROFESSIONAL	Full restaurants	+ KDS, stations, tabs, drive-through	Unlimited orders, 15 staff
ENTERPRISE	Chains, franchises	+ Multi-location, API, white-label	Unlimited everything

Key Differentiators

1. **East Africa First** - Built for M-Pesa, Tigo Pesa, Airtel Money. Designed for local restaurant workflows and trust models.
2. **Grow Without Migration** - A home chef on STARTER uses the same platform as a 50-location chain on ENTERPRISE. Upgrade your plan, not your system.
3. **Offline-Capable** - POS works offline, queues orders, syncs when connected. Critical for areas with unreliable internet.
4. **Multi-Channel by Default** - Not an add-on. Every kitchen can receive orders from POS, app, WhatsApp, and kiosks simultaneously from day one.
5. **Kitchen-Centric Design** - Built around how real kitchens work. Stations, expeditor control, parallel preparation, and clear handoff points.

Document Scope

This architecture document covers:

- **Part 1-2**: Platform tiers, subscription plans, and sales channels
- **Part 3-5**: Order lifecycle, tabs, checkout flows
- **Part 6**: Payment architecture
- **Part 7-10**: Kitchen operations, stations, printing, configuration presets

- **Part 11-12**: Device management, menu channel settings
- **Part 13**: Data models and entity relationships
- **Part 14-15**: Unified system architecture, delivery integration
- **Part 16-18**: Operational scenarios, requirements, APIs
- **Appendices**: Glossary, configuration checklists, decision log

---

Part 1: Platform Tiers & Progressive Unlocking

1.1 The Vision: One Platform, All Scales

JikoXpress Pro is designed to grow with the business. A home chef starts with the simplest possible setup and progressively unlocks features as their operation scales.

```
STARTER          GROWING           PROFESSIONAL      ENTERPRISE
(Home Chef)      (Small Shop)      (Restaurant)      (Chain)

    □□     →      □□□□    →        □□□□□    →       □□□□

• Mobile app     • + POS access    • + KDS           • + Multi-location
• Basic menu     • + Kiosk         • + Stations      • + Central menu
• Order list     • + Table QR      • + Drive-through • + Analytics
• Accept/Ready   • + Basic stats   • + Staff roles    • + API access
• Mobile money   • + More payments • + Tabs          • + Integrations
                 • + Printer       • + Inventory     • + White-label
```

1.2 Subscription Plans

STARTER Plan

Tagline: "Perfect for home chefs & small vendors"

Price: Free or minimal fee

Features Included:

- Basic menu (up to 20 items)
- Order management (receive, accept, prepare, complete)
- Mobile app only

- Mobile money payments (USSD)
- Operating hours scheduling
- Order history (30 days)
- Table QR (1 table/location)
- Basic push notifications

Limits:

- 20 menu items
- 100 orders/month
- 1 staff account (owner only)
- 1 location

Ideal For: Home kitchens, street food vendors, side hustle chefs, testing the platform

---

GROWING Plan

Tagline: "For busy kitchens ready to scale"

Price: Mid-tier monthly fee

Features Included:

- Everything in STARTER
- Unlimited menu items
- POS access (desktop/tablet)
- Kiosk channel
- Table QR (unlimited)
- WhatsApp bot ordering
- Receipt printer support
- Kitchen printer support
- Card payments
- Cash handling
- Basic sales reports
- Order history (1 year)

Limits:

- 1,000 orders/month
- 3 staff accounts
- 1 location

Ideal For: Small restaurants, busy food stalls, cafes, food trucks

---

PROFESSIONAL Plan

Tagline: "Full-featured restaurant management"

Price: Higher-tier monthly fee

Features Included:

- Everything in GROWING
- Kitchen stations
- KDS (Kitchen Display System)
- Drive-through support
- Tabs (pay-later for dine-in)
- Table management
- Up to 15 staff accounts
- Roles & permissions
- Discount limits by role
- Advanced reports
- Customer insights
- Priority support

Limits:

- Unlimited orders
- 15 staff accounts
- 1 location

Ideal For: Full-service restaurants, fast food outlets, bars & lounges, hotel restaurants

---

ENTERPRISE Plan

Tagline: "For chains & franchises"

Price: Custom pricing

Features Included:

- Everything in PROFESSIONAL
- Multi-location support
- Central menu management
- Consolidated cross-location reports
- API access for integrations
- Inventory management
- White-label option
- Dedicated support
- Custom integrations
- SLA guarantees

Limits:

- Unlimited everything

Ideal For: Restaurant chains, franchises, hotel groups, cloud kitchens with multiple brands

---

1.3 Smart Onboarding & Segmentation

When a new kitchen signs up, 4 simple questions determine the recommended plan:

Question 1: Business Type

- Home Kitchen / Catering → Score: 0
- Food Stall / Vendor → Score: 1
- Restaurant / Cafe → Score: 2
- Chain / Multiple Locations → Score: 3

Question 2: Expected Orders Per Day

- Just starting (1-10/day) → Score: 0
- Getting busy (10-50/day) → Score: 1
- High volume (50-200/day) → Score: 2
- Very high (200+/day) → Score: 3

Question 3: Team Size

- Just me → Score: 0
- 2-5 people → Score: 1
- 6-15 people → Score: 2
- 15+ people → Score: 3

Question 4: Service Styles (multi-select)

- Delivery only → Score: 0
- Pickup only → Score: 0
- Dine-in → Score: +1
- Drive-through → Score: +2

Recommendation Logic:

- Score 0-2: STARTER
- Score 3-5: GROWING
- Score 6-8: PROFESSIONAL
- Score 9+: ENTERPRISE

1.4 Feature Modules (Unlockable)

Each feature is a module that can be toggled on/off based on subscription:

```
Kitchen Feature Modules:

CORE (Always On - STARTER):
├── basic_menu
├── order_management
├── mobile_notifications
├── basic_payments (mobile money)
├── operating_hours
└── order_history

CHANNELS (GROWING+):
├── pos_access
├── kiosk_channel
├── table_qr
└── whatsapp_bot

PAYMENTS (GROWING+):
├── card_payments
├── kitchen_wallet
├── cash_handling
└── custom_payment_methods

HARDWARE (GROWING+):
├── receipt_printer
├── kitchen_printer
└── cash_drawer

KITCHEN OPS (PROFESSIONAL+):
├── stations
├── kds_display
├── expeditor_mode
└── drive_through

DINE-IN (PROFESSIONAL+):
├── tabs
```

```
├── table_management
└── reservations (future)

TEAM (PROFESSIONAL+):
├── staff_accounts
├── roles_permissions
├── discount_limits
└── shift_management

INSIGHTS (PROFESSIONAL+):
├── sales_reports
├── popular_items
├── peak_hours
└── customer_insights

SCALE (ENTERPRISE):
├── multi_location
├── central_menu
├── consolidated_reports
├── api_access
└── white_label

INVENTORY (ENTERPRISE):
├── ingredient_tracking
├── auto_deduction
├── low_stock_alerts
└── supplier_orders
```

1.5 Usage-Based Upgrade Triggers

The platform monitors usage and suggests upgrades:

STARTER → GROWING Triggers:

- Orders this month > 80 (approaching 100 limit)
- Menu items > 15 (approaching 20 limit)
- Customer requested card payment
- Multiple login attempts (wants staff accounts)

GROWING → PROFESSIONAL Triggers:

- Orders this month > 800 (approaching 1,000 limit)
- Staff accounts maxed at 3
- Searched for: tabs, dine-in, stations, KDS
- Average prep time > 15 minutes

PROFESSIONAL → ENTERPRISE Triggers:

- Asked about second location
- Searched for: multi-location, franchise, API
- Orders this month > 5,000

Part 2: Sales Channels

2.1 Channel Overview

JikoXpress Pro supports seven distinct sales channels, each with unique characteristics.

POS (Point of Sale)

The primary channel for counter staff and waiters.

Characteristics:

- Staff-operated, requires speed and efficiency
- Supports both immediate payment and pay-later (tab) scenarios
- Full access to all payment methods
- Primary channel for dine-in customers
- Can lookup existing customers or create new ones
- Generates QR codes for customer self-payment via app

Availability: GROWING plan and above

Typical Flow:

1. Staff enters items into cart
2. Identifies customer (optional)
3. Selects fulfillment type (dine-in with table number)
4. Chooses payment timing: Pay Now or Pay Later (Tab)
5. Processes payment or opens tab
6. Kitchen receives order based on configuration

Kiosk

Self-service terminals for customer-driven ordering.

Characteristics:

- Customer-operated, must be intuitive
- Session-based cart storage
- Limited payment options (QR, USSD, Cash, Card Swipe)
- Primarily for dine-in and pickup customers
- Anonymous or optional login
- Prints order stub for customer

Availability: GROWING plan and above

Typical Flow:

1. Customer browses menu on touchscreen
2. Adds items to cart with modifications
3. Proceeds to checkout
4. Selects payment method
5. Completes payment (or receives cash payment slip)
6. Receives printed stub with order number
7. Waits for number to be called

Table QR

Self-ordering from customer's own device by scanning QR code at table.

Characteristics:

- Customer scans QR code placed on their table
- Opens in App (if installed) or Web browser (no app needed)
- No login required - anonymous ordering
- Table number automatically identified from QR
- Payment: USSD, Cash (pay at counter), App Wallet (if logged in)
- Dine-in only

Availability: STARTER plan (1 table), GROWING+ (unlimited)

QR Code Content:

```
https://menu.jikoxpress.com/{kitchenSlug}?table=5
or
jikoxpress://menu/{kitchenId}?table=5 (deep link for app)
```

Web vs App Experience:

Aspect	Web (No App)	App
Login	Not required	Optional
Cart	Browser session	Persistent
Payment	USSD, Cash	USSD, Cash, Wallet
Order History	Not available	Available
Push Notifications	Not available	Available

Mobile App

Full-featured mobile application for registered users.

Characteristics:

- Requires user account
- Server-side persistent cart
- Supports delivery, pickup, and dine-in
- Preferred payment: App Wallet, USSD
- Push notifications for order status
- Order history and reordering

Availability: All plans (customer-facing)

WhatsApp

Conversational ordering via chatbot integration.

Characteristics:

- Customer identified by phone number
- Chatbot-guided ordering process
- Cart maintained in conversation state
- Payment via USSD or payment link
- Supports delivery and pickup only

Availability: GROWING plan and above

Drive-Through

Vehicle-based ordering and pickup.

Characteristics:

- Designed for speed - time SLA is critical
- Lane-based queue management
- Speaker/mic or digital menu board ordering
- Payment and pickup at windows
- Vehicle tracking by order number

Availability: PROFESSIONAL plan and above

Typical Flow:

1. Customer enters lane
2. Places order at order point (speaker or screen)
3. Proceeds to payment window
4. Proceeds to pickup window
5. Receives order and exits

Direct Counter Service

Simplified model for small operations where counter is kitchen.

Characteristics:

- Same person takes order and prepares food
- No kitchen routing needed
- Immediate handoff to customer
- May only need receipt, no kitchen ticket

Availability: All plans (default for STARTER)

2.2 Channel Comparison Matrix

Aspect	POS	Kiosk	Table QR	App	WhatsApp	Drive-Through
Operator	Staff	Customer	Customer	Customer	Customer	Staff/Screen
Cart Storage	Memory	Session	Session/App	Server	Conversation	Memory
Customer ID	Optional	Anonymous	Anonymous	Required	Phone	Optional
Pay Later (Tab)	Yes	No	No	No	No	No
Fulfillment	All	Dine-in, Pickup	Dine-in	All	Delivery, Pickup	Drive-Through
Speed Priority	Critical	High	Normal	Normal	Normal	**Critical**

Aspect	POS	Kiosk	Table QR	App	WhatsApp	Drive-Through
Time SLA	Normal	Normal	Normal	Normal	Normal	**Critical**
Min Plan	GROWING	GROWING	STARTER	All	GROWING	PROFESSIONAL

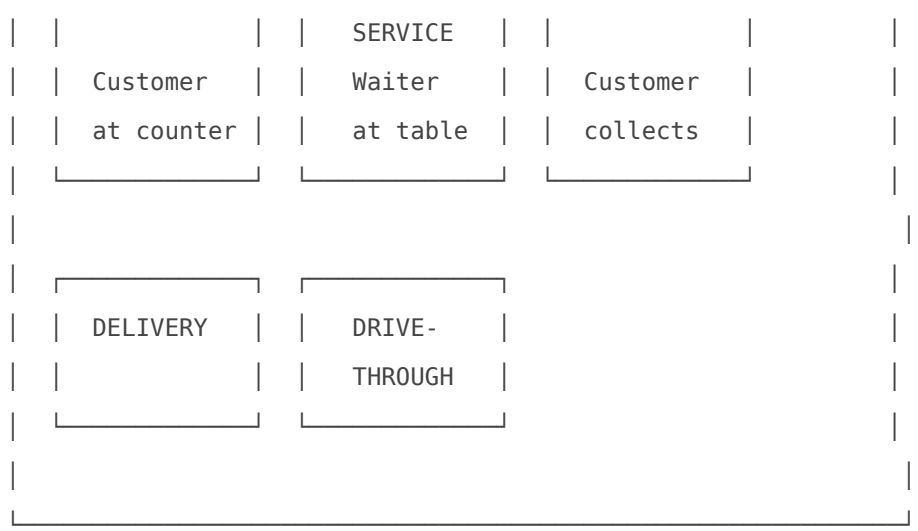
2.3 POS Service Modes

POS devices can operate in different service modes depending on the restaurant setup:

```
POS Service Modes:

COUNTER_SERVICE:
  description: "Staff behind counter, customer faces them"
  flow:
    - Customer approaches counter
    - Staff takes order on POS
    - Customer waits at counter or takes number
  used_for: Fast food, cafes, takeaway

TABLE_SERVICE:
  description: "Waiter goes to customer's table with tablet"
  flow:
    - Customer seated at table
    - Waiter approaches with tablet
    - Waiter takes order at table
    - Table number captured
    - Food delivered to table
  used_for: Restaurants, fine dining, casual dining
```

Service Mode Selection in POS:

```
┌─────────────────────────────────────────────────────────┐
│  New Order                                              │
├─────────────────────────────────────────────────────────┤
│                                                         │
│  Service Type:                                          │
│                                                         │
│  ┌──────────────┐  ┌──────────────┐  ┌──────────────┐   │
│  │  COUNTER     │  │   TABLE      │  │   PICKUP     │   │
```



Part 3: Order & Tab Architecture

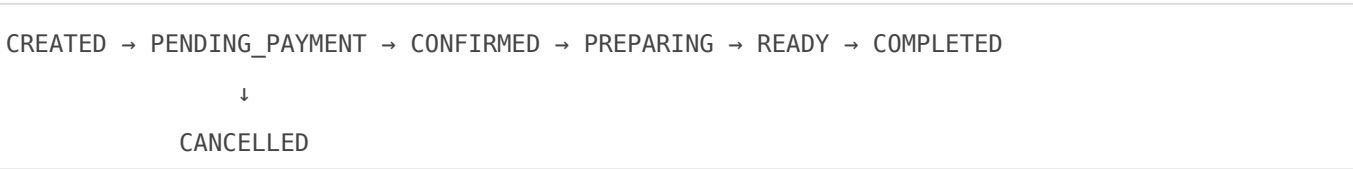
3.1 Core Concept: Single Source of Truth

All channels ultimately create Orders. The Order is the central entity that:

- Tracks what was purchased
- Connects to payment records
- Drives kitchen operations
- Generates receipts and reports

Key Principle: Cart is an unpersisted Order draft. Checkout persists the Order and initiates payment.

3.2 Order Lifecycle



State Definitions:

Status	Description	Kitchen Visible	Payment Status
CREATED	Order recorded but not yet submitted	No	Not started

Status	Description	Kitchen Visible	Payment Status
PENDING_PAYMENT	Awaiting async payment confirmation	Depends on config	In progress
CONFIRMED	Payment complete or tab opened	Yes	Paid or Deferred
PREPARING	Kitchen actively working on order	Yes	N/A
READY	Order complete, waiting for pickup	Yes	N/A
COMPLETED	Customer received order	No	N/A
CANCELLED	Order cancelled before completion	No	Refunded if paid

3.3 Payment Status (Separate from Order Status)

Orders track payment independently:

Payment Status	Description
UNPAID	No payment received (tab scenario)
PENDING	Async payment initiated, awaiting confirmation
PARTIAL	Some payment received (future feature)
PAID	Full payment received
REFUNDED	Payment returned to customer

3.4 Tab System (Pay Later)

Tabs enable the "eat now, pay later" model common in dine-in scenarios.

Availability: PROFESSIONAL plan and above

What is a Tab?

- A container for one or more orders at a table
- Remains open until customer is ready to pay
- Aggregates totals across all orders
- Single payment closes the entire tab

Tab Lifecycle:

```
[No Tab] → OPEN → CLOSED
              ↑
    (orders can be added while OPEN)
```

Tab Rules:

- Customer can add more orders to open tab
- Each order goes to kitchen independently
- Tab shows running total of all orders
- Discounts can be applied at tab level before closing
- Single payment closes tab and marks all orders as PAID

Why Multiple Orders per Tab:

- Kitchen already received and is preparing original order
- New items create new kitchen tickets
- Each order has clean, independent lifecycle
- Simplifies kitchen display and ticket management

3.5 Fulfillment Types

Type	Description	Queue Identifier	Handoff Point
DINE_IN	Customer eats at restaurant	Table Number	Table
PICKUP	Customer collects order	Order Number	Counter
DELIVERY	Order delivered to customer	Address	Door
DRIVE_THROUGH	Vehicle-based pickup	Lane + Order #	Window

Part 4: Drive-Through Architecture

4.1 Overview

Drive-through is modeled as both a **channel** and a **fulfillment type**, allowing it to integrate seamlessly with the existing order system while supporting its unique requirements.

4.2 Drive-Through Flow

```
LANE_ENTRY → ORDER_POINT → PAYMENT_WINDOW → PICKUP_WINDOW → EXIT
     ↓            ↓              ↓               ↓
  (detected)  (order created) (payment)    (order handed)
```

4.3 Configuration Options

```
Drive-Through Settings:
  enabled: false (default)
  numberOfLanes: 1-4

  orderPointType:
    - SPEAKER_MIC        # Staff takes order via speaker
    - DIGITAL_MENU_BOARD # Customer self-orders on screen
    - BOTH               # Screen with staff backup

  windowConfiguration:
    - SINGLE_WINDOW      # Pay and pickup at same window
    - SEPARATE_WINDOWS   # Payment window, then pickup window

  vehicleTracking:
    - ORDER_NUMBER       # Customer given number, called at window
    - MANUAL             # Staff tracks position manually
    - TIMER_BASED        # System estimates based on avg times
```

4.4 Drive-Through Order Properties

```
Order (when fulfillment = DRIVE_THROUGH):
  laneNumber: 1-4
  vehicleDescription: "Red Toyota" (optional)
  orderPointTimestamp: when order was taken
  paymentWindowTimestamp: when payment completed
  pickupWindowTimestamp: when order handed off
```

4.5 Kitchen Ticket for Drive-Through

```
================================
        ORDER #47
```

```
     Lane 2 | DRIVE-THROUGH
   12:34 PM | 03-Jan-2026
--------------------------------
>> HOT LINE <<

1x Burger
   - No onions
   - Extra cheese

2x Fries
   - Large


--------------------------------
⚡ DRIVE-THROUGH - SPEED PRIORITY
================================
```

4.6 Drive-Through Metrics

Due to the critical nature of speed in drive-through:

Tracked Metrics:

- Average time per vehicle
- Time at each station (order, payment, pickup)
- Orders per hour per lane
- Peak hour analysis

Alerts:

- Order taking > 2 minutes
- Payment processing > 1 minute
- Pickup wait > 3 minutes
- Lane backup detected

Part 5: Checkout Flow

5.1 Universal Checkout Principles

Regardless of channel, checkout always involves:

1. **Identify Items** - What is being ordered
2. **Identify Customer** - Who is ordering (optional for some channels)
3. **Identify Fulfillment** - How order will be fulfilled
4. **Calculate Totals** - Items + taxes + fees - discounts
5. **Select Payment** - How customer will pay
6. **Process Payment** - Execute payment or defer
7. **Create/Confirm Order** - Finalize and notify kitchen

5.2 Default Flow (Zero Configuration)

If a kitchen sets nothing, this is what happens:

```
Default Configuration:
  fulfillmentTypes: [PICKUP]
  paymentTiming: ON_PAYMENT_COMPLETE
  orderRouting: DISPLAY_ONLY
  kitchenNotification: DISPLAY
  printOnConfirm: true
  customerStub: true
  tabsEnabled: false
```

Default Experience:

1. Kitchen signs up
2. Sets their menu
3. Immediately can take orders via mobile app
4. Customer pays → Kitchen sees order on screen → Prepares → Calls number → Customer picks up

No stations, no tabs, no printers required. Just works.

5.3 Channel-Specific Checkout Flows

POS Checkout (Pay Now)

```
1. Staff finalizes cart
2. System creates checkout session
3. Staff selects customer (optional) and fulfillment
4. System calculates totals
5. Staff selects payment method
6. System processes payment
7. Order created with status CONFIRMED
```

```
8. Kitchen notified
9. Receipt printed
```

POS Checkout (Pay Later / Tab)

```
1. Staff finalizes cart
2. Staff selects "Pay Later" and enters table number
3. System finds/creates Tab for table
4. Order created with status CONFIRMED, paymentStatus UNPAID
5. Order linked to Tab
6. Kitchen notified immediately
7. Tab remains open for additional orders

[Later, when customer ready to pay]
8. Staff retrieves Tab for table
9. Staff applies discounts if needed
10. Staff processes payment
11. Tab closed, all orders marked PAID
12. Receipt printed
```

Drive-Through Checkout

```
1. Customer arrives at order point (lane assigned)
2. Staff/screen takes order
3. Order created with fulfillment DRIVE_THROUGH
4. Customer proceeds to payment window
5. Payment processed
6. Order status → CONFIRMED, sent to kitchen
7. Customer proceeds to pickup window
8. Kitchen prepares (priority queue)
9. Order handed to customer
10. Order status → COMPLETED
```

Kiosk / Table QR Checkout

```
1. Customer finalizes cart
2. System creates checkout session
3. Customer selects fulfillment
4. Customer selects payment method:
   - USSD: Receives prompt, confirms
```

```
   - Cash: "Pay at counter" displayed
5. On payment confirmation:
   - Order confirmed
   - Kitchen notified
   - Stub printed (kiosk) or shown on screen (Table QR)
```

5.4 Checkout Session

A checkout session provides:

- Temporary holding state during checkout
- Protection against abandoned checkouts
- Support for async payment flows

Session States:

Status	Description
ACTIVE	Checkout in progress
COMPLETED	Payment successful, order created
EXPIRED	Session timed out (15 min default)
CANCELLED	User or system cancelled

Part 6: Payment Architecture

6.1 Payment Method Categories

Immediate Settlement

Method	Description	Channels
Cash	Physical currency	POS, Kiosk
Card Swipe	Card terminal	POS, Kiosk
Kitchen Wallet	Kitchen-issued balance	POS
App Wallet	Platform wallet	POS (QR), App

Asynchronous Settlement

Method	Description	Channels

Mobile Money (USSD)	Customer confirms on phone	All
QR Scan (App Wallet)	POS displays QR	POS
Payment Link	Customer clicks link	WhatsApp

Deferred Settlement

Method	Description	Channels
Pay Later (Tab)	Opens tab, pay before leaving	POS only

6.2 Payment Method by Plan

Method	STARTER	GROWING	PROFESSIONAL	ENTERPRISE
Mobile Money (USSD)	✓	✓	✓	✓
Cash	-	✓	✓	✓
Card	-	✓	✓	✓
Kitchen Wallet	-	✓	✓	✓
App Wallet	✓	✓	✓	✓
Pay Later (Tab)	-	-	✓	✓
Custom Methods	-	-	✓	✓

6.3 Kitchen Wallet

A prepaid balance system for loyal customers.

Characteristics:

- Issued and managed by individual kitchen
- Not platform-wide (unlike App Wallet)
- Can be topped up via cash at counter
- Used for quick payment without cash handling

Use Cases:

- Regular customers with running credit
- Corporate accounts
- Staff meals
- VIP customers

6.4 Discount Handling

Discount Types:

- Percentage: Reduce total by X%
- Fixed Amount: Reduce total by fixed value

Staff Discount Limits (PROFESSIONAL+):

- Each staff role has maximum discount percentage
- Exceeding limit requires manager approval (PIN)
- All discounts recorded with who applied and approved

Part 7: Kitchen Operations

7.1 Operation Models

Model A: Direct Counter Service (Default for STARTER)

```
Order Confirmed → Same person prepares → Immediate handoff
```

No kitchen routing, no printers, no complexity.

Model B: Paper Ticket Kitchen

```
Order Confirmed → Print ticket → Kitchen works from paper
```

Traditional, reliable, no technology in kitchen.

Model C: Kitchen Display System (KDS)

```
Order Confirmed → Appears on screen → Kitchen marks done
```

Real-time visibility, paperless, item-level tracking.

Model D: Hybrid

```
Order Confirmed → Screen AND paper
```

Best of both, redundancy if one fails.

Model E: Expeditor Controlled (Fine Dining)

```
Order Confirmed → Expeditor screen → Expeditor fires items
```

Pacing control for multi-course meals.

7.2 Kitchen Stations (PROFESSIONAL+)

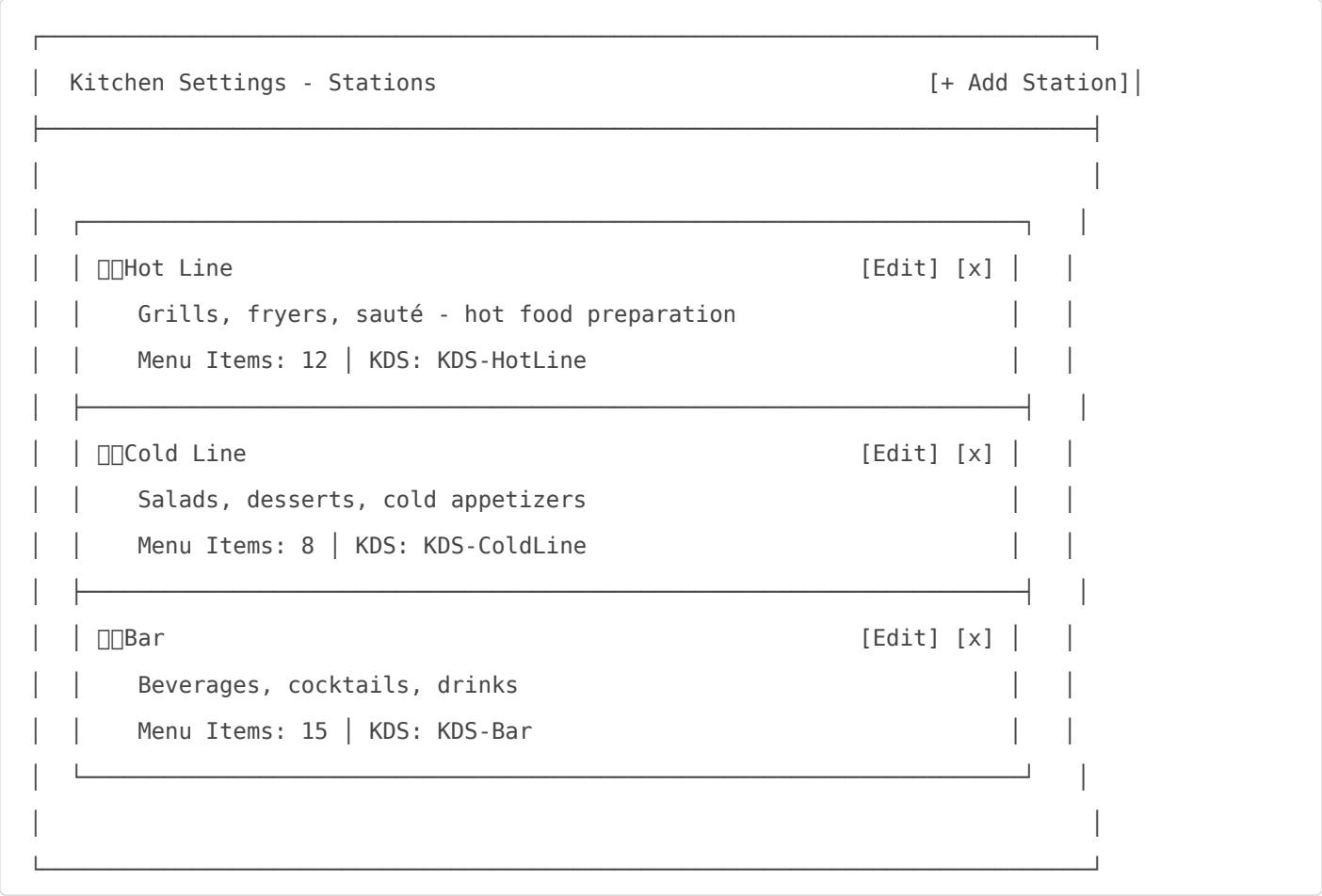
Larger kitchens have multiple stations. Each menu item is pre-assigned to a station during menu setup, and the system automatically routes order items to the appropriate stations.

Common Stations:

- Hot Line: Grills, fryers, sauté
- Cold Line: Salads, desserts
- Bar: Beverages, cocktails
- Bakery: Bread, pastries
- Expeditor: Final assembly

7.2.1 Station Creation

Stations are created in Kitchen Settings by the admin:



7.2.2 Station Data Model

```
Station:
  id: uuid
  kitchenId: uuid

  # Basic info
  name: "Hot Line"
  description: "Grills, fryers, sauté - hot food preparation"
  icon: "□□"
  color: "#FF5733"  # For KDS display

  # Configuration
  displayOrder: 1  # Sort order on KDS
  printerId: uuid (nullable)  # Dedicated printer for this station

  # Status
  isActive: true

  # Timestamps
  createdAt: datetime
  updatedAt: datetime
```

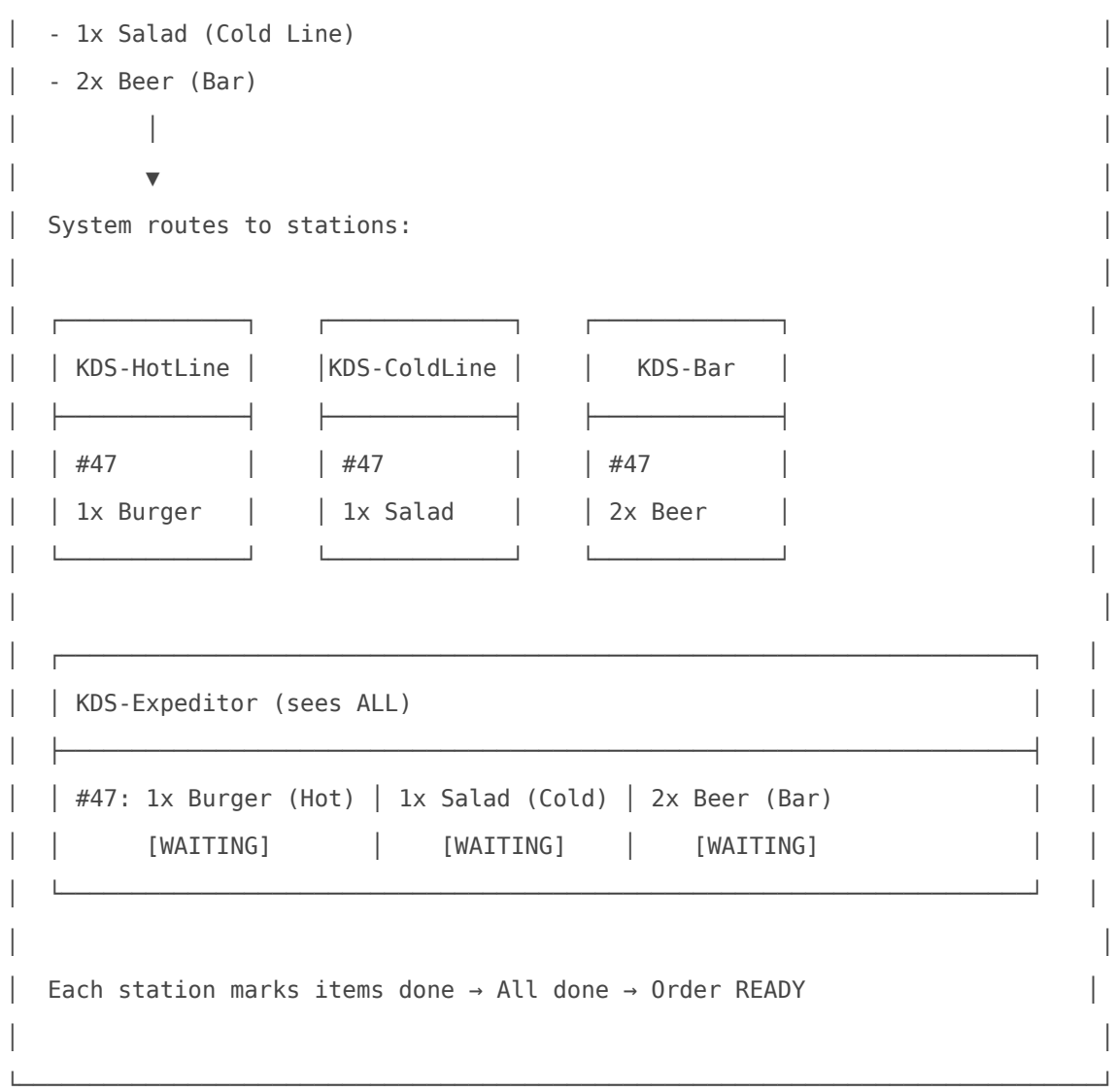
7.2.3 Menu Item Station Assignment

Each menu item is assigned to a station during creation/editing:

```
MenuItem:
  id: uuid
  name: "Burger"
  price: 12000
  category: "Main Course"
  stationId: uuid  # ← Pre-defined assignment to station
```

7.2.4 Station Routing Flow

```
┌─────────────────────────────────────────────────────────────────────────┐
│                        STATION ROUTING FLOW                             │
├─────────────────────────────────────────────────────────────────────────┤
│                                                                         │
│  Order #47 comes in:                                                    │
│  - 1x Burger (Hot Line)                                                 │
```



7.2.5 Station Presets

For quick setup, kitchens can choose from presets:

Preset	Stations Created
Fast Food	Hot Line, Cold Line, Drinks
Full Restaurant	Hot Line, Cold Line, Bar, Bakery, Expeditor
Cafe / Coffee Shop	Barista, Kitchen, Display
Pizza Place	Pizza Station, Grill, Salads, Bar
No Stations	All orders on single screen

Station Routing Example:

```
Order #47:
├── Burger → Hot Line
```

```
├── Caesar Salad → Cold Line
├── Beer → Bar
└── Cheesecake → Cold Line
```

7.3 Kitchen Timing Configuration

When does kitchen receive the order?

Setting	Description	Use Case
ON_ORDER_CREATED	Before payment	Trusted dine-in
ON_PAYMENT_COMPLETE	After payment	Kiosk, delivery
MANUAL	Staff triggers	Special events

Default by Channel:

- POS Dine-in: ON_ORDER_CREATED
- POS Pickup: ON_PAYMENT_COMPLETE
- Kiosk: ON_PAYMENT_COMPLETE
- Table QR: ON_PAYMENT_COMPLETE
- App: ON_PAYMENT_COMPLETE
- WhatsApp: ON_PAYMENT_COMPLETE
- Drive-Through: ON_PAYMENT_COMPLETE

7.4 Configuration Summary

```
Kitchen Operation Settings:
  orderRouting:
    - DIRECT_TO_STATIONS
    - EXPEDITOR_CONTROLLED
    - DISPLAY_ONLY
    - NONE (counter service - default)

  notificationMethod:
    - DISPLAY (KDS)
    - PRINTER
    - BOTH
    - NONE (counter service - default)

  timingByChannel:
    pos_dine_in: ON_ORDER_CREATED | ON_PAYMENT_COMPLETE
```

```
    pos_pickup: ON_PAYMENT_COMPLETE
    kiosk: ON_PAYMENT_COMPLETE
    table_qr: ON_PAYMENT_COMPLETE
    app: ON_PAYMENT_COMPLETE
    whatsapp: ON_PAYMENT_COMPLETE
    drive_through: ON_PAYMENT_COMPLETE

  kioskCashFlow:
    - KITCHEN_FIRST (prepare before cash confirmed)
    - PAYMENT_FIRST (wait for cash - default)
```

Part 8: Printing Architecture

8.1 Print Types

Print Type	When	Where	Purpose
Kitchen Ticket	Order to kitchen	Kitchen printer	What to prepare
Customer Receipt	Payment completed	POS printer	Proof of purchase
Bill/Check	Customer requests	POS printer	Amount due
Order Stub	Kiosk order	Kiosk printer	Order number
Void Notice	Item voided	Kitchen printer	Cancel notification

8.2 Kitchen Ticket Example

```
================================
        ORDER #47
     Table 5 | Dine-in
   12:34 PM | 03-Jan-2026
--------------------------------
>> HOT LINE <<

1x Burger
   - No onions
   - Extra cheese
```

```
2x Fish & Chips
   - One mild spice

--------------------------------
Server: John
================================
```

8.3 Customer Receipt Example

```
================================
       JIKOXPRESS KITCHEN
    123 Main Street, Dar
      Tel: +255 123 456

      TAX ID: 12345678
================================
Receipt #: R-20260103-0047
Date: 03-Jan-2026 13:45
Table: 5
Server: John
--------------------------------
ITEMS:
1x Burger              12,000
   Extra cheese         1,500
2x Fish & Chips        30,000
--------------------------------
Subtotal:              43,500
Discount (10%):        -4,350
VAT (18%):              7,047
--------------------------------
TOTAL:                 46,197
================================
PAYMENT:
Mobile Money           46,197
Ref: MP12345678
================================
      Thank you!
================================
```

8.4 Print Architecture

Model: Client Pull (Recommended)

```
1. Order confirmed
2. Backend creates PrintJob (status: PENDING)
3. POS app polls: GET /print-jobs?status=PENDING
4. POS app claims job
5. POS app prints locally
6. POS app marks complete
```

Works with local USB/Bluetooth printers, supports offline queuing.

Part 9: Numbering Systems

9.1 Order Number

- **Purpose:** Kitchen identification, customer reference
- **Format:** Simple integer, resets daily
- **Example:** #47

9.2 Receipt Number

- **Purpose:** Accounting, tax compliance
- **Format:** Prefix-Date-Sequence
- **Example:** R-20260103-0047
- Never resets (continuous with date prefix)

9.3 Internal IDs

- All entities have UUID primary keys
- Used for database relationships and API references
- Display numbers are separate from internal IDs

Part 10: Configuration Presets

10.1 Available Presets

Home Kitchen Preset (STARTER Default)

```
orderRouting: NONE
kitchenNotification: NONE
sendToKitchen: immediate (same person)
customerStub: false
customerDisplay: false
tabsEnabled: false
```

Fast Food Preset

```
orderRouting: DIRECT_TO_STATIONS
kitchenNotification: DISPLAY
sendToKitchen: ON_PAYMENT_COMPLETE
customerStub: true
customerDisplay: true
tabsEnabled: false
```

Casual Dining Preset

```
orderRouting: DIRECT_TO_STATIONS
kitchenNotification: PRINTER
sendToKitchen: ON_PAYMENT_COMPLETE
customerStub: optional
customerDisplay: false
tabsEnabled: true
```

Fine Dining Preset

```
orderRouting: EXPEDITOR_CONTROLLED
kitchenNotification: BOTH
sendToKitchen: ON_ORDER_CONFIRMED
customerStub: false
customerDisplay: false
tabsEnabled: true (default for dine-in)
```

Drive-Through Preset

```
orderRouting: DIRECT_TO_STATIONS
kitchenNotification: DISPLAY
sendToKitchen: ON_PAYMENT_COMPLETE
customerStub: false
customerDisplay: true (shows order numbers)
tabsEnabled: false
driveThrough:
  enabled: true
  lanes: 1
  combinedWindows: true
```

Food Truck Preset

```
orderRouting: DISPLAY_ONLY
kitchenNotification: DISPLAY
sendToKitchen: ON_PAYMENT_COMPLETE
customerStub: true
customerDisplay: single screen
tabsEnabled: false
```

Part 11: Device Management

11.1 Device Types & Roles

Each app type knows its responsibilities:

```
Device Types:

# ═══════════════════════════════════════════════════════════════════════
# ORDER ENTRY DEVICES
# ═══════════════════════════════════════════════════════════════════════

POS:
  role: "Staff order entry & payment processing"
  capabilities:
    - Create orders
    - Process payments
```

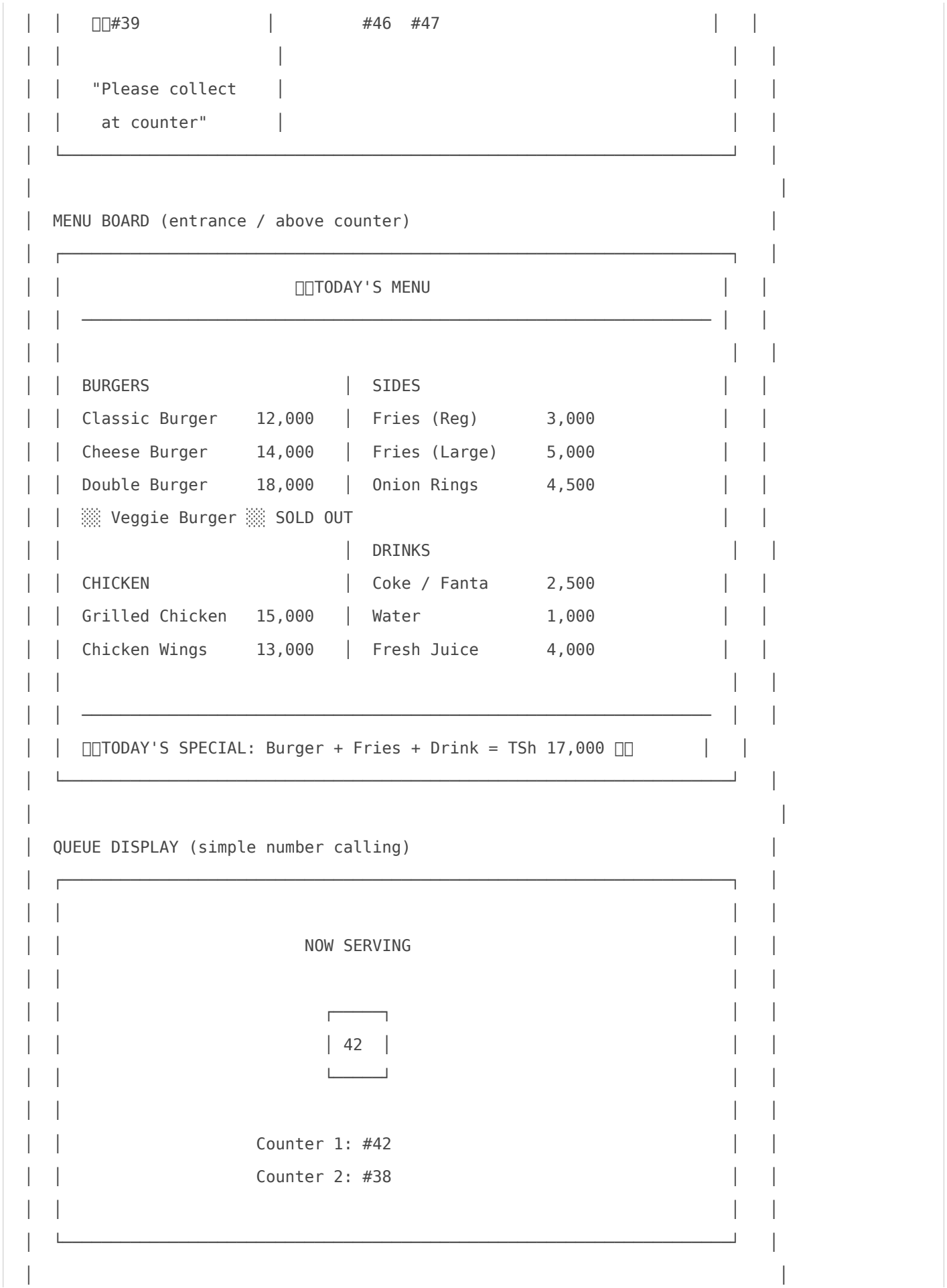
```
    - Manage tabs
    - Apply discounts
    - Print receipts
    - View order history
  used_by: Cashiers, Waiters, Counter Staff

KIOSK:
  role: "Customer self-service ordering"
  capabilities:
    - Browse menu
    - Create orders
    - Process payments (limited methods)
    - Print stubs
  used_by: Customers (unattended)

WAITER_TABLET:
  role: "Mobile order entry at tables"
  capabilities:
    - Same as POS
    - Optimized for table service
    - Quick table selection
  used_by: Waiters, Servers

# ═══════════════════════════════════════════════════════════════════════
# KITCHEN DEVICES
# ═══════════════════════════════════════════════════════════════════════

KDS (Kitchen Display System):
  role: "Kitchen order display & management"
  capabilities:
    - View incoming orders
    - Filter by station (optional)
    - Update item status
    - Mark orders ready
    - Bump orders
  used_by: Kitchen Staff, Line Cooks, Expeditor

# ═══════════════════════════════════════════════════════════════════════
# CUSTOMER-FACING DISPLAYS (View Only)
```

```
# ═══════════════════════════════════════════════════════════════════════

CUSTOMER_FACING_SCREEN:
  role: "Show current order being entered at POS"
  capabilities:
    - Mirror POS cart in real-time
    - Show items as they're added
    - Show running total
    - Show payment confirmation
  location: Counter (facing customer)
  used_by: Customers (view only, paired with POS)

ORDER_STATUS_DISPLAY:
  role: "Show order progress to waiting customers"
  capabilities:
    - Display "Now Preparing" orders
    - Display "Ready for Pickup" orders
    - Call order numbers
    - Show estimated wait time
  location: Waiting area, Counter
  used_by: Customers (view only)

MENU_BOARD:
  role: "Digital menu display"
  capabilities:
    - Show menu items with prices
    - Show item availability (live)
    - Show daily specials
    - Show promotions
    - Auto-hide sold out items
  location: Entrance, Above counter, Drive-through order point
  used_by: Customers (view only)

DRIVE_THROUGH_DISPLAY:
  role: "Order confirmation at drive-through"
  capabilities:
    - Show current order being taken
    - Show order total
    - Confirm items to customer
```

```
  location: Drive-through order point
  used_by: Customers in vehicle (view only)

QUEUE_DISPLAY:
  role: "Show queue/ticket numbers"
  capabilities:
    - Display current serving number
    - Display numbers being prepared
    - Play sound/chime when number called
    - Show multiple counters if applicable
  location: Waiting area
  used_by: Customers (view only)
```

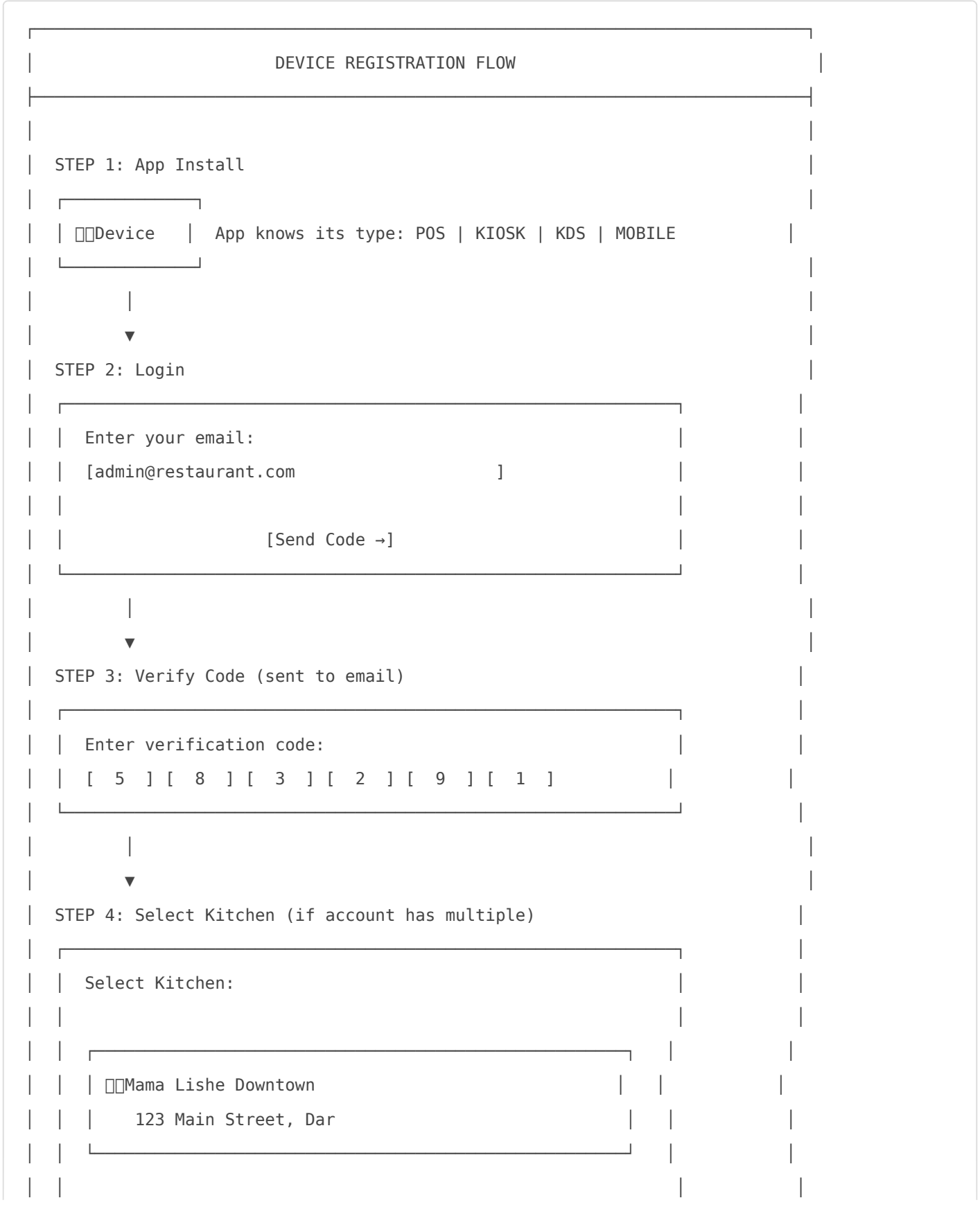
11.1.1 Display Device Examples

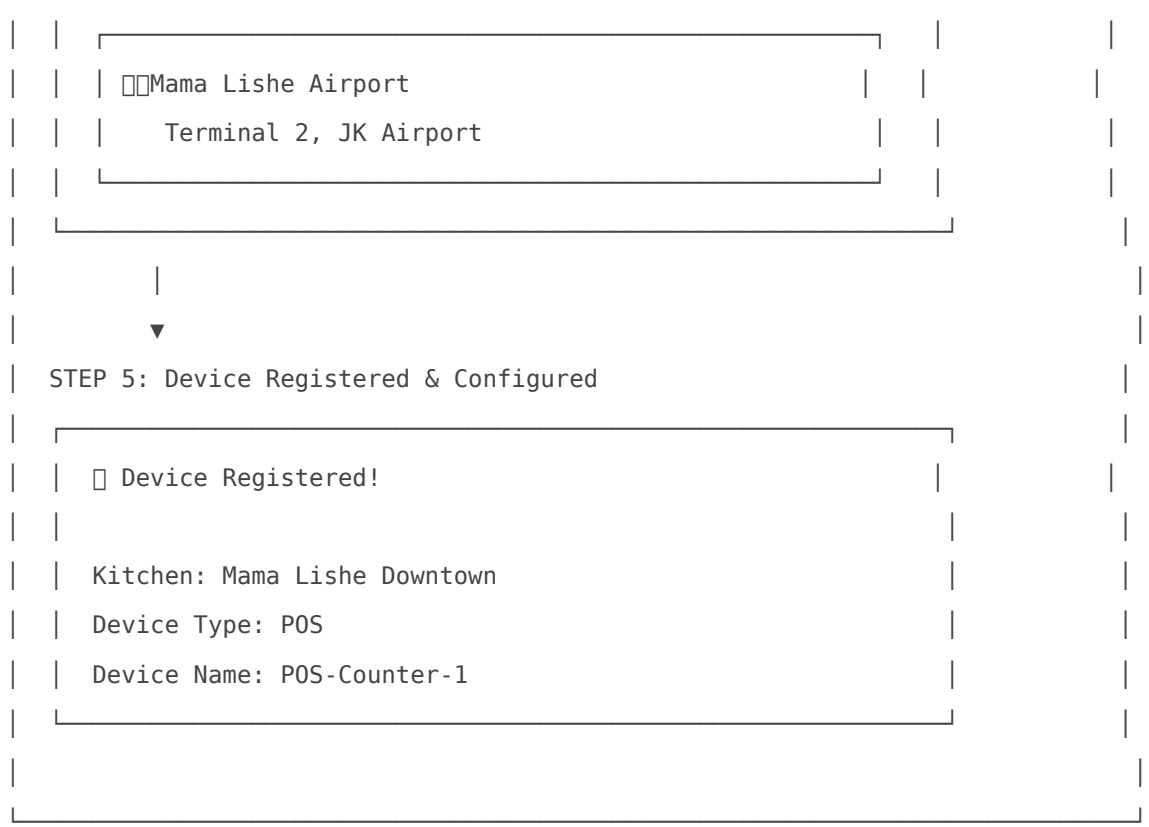
```
┌──────────────────────────────────────────────────────────────────────────┐
│                        CUSTOMER-FACING DISPLAYS                          │
├──────────────────────────────────────────────────────────────────────────┤
│                                                                          │
│  CUSTOMER FACING SCREEN (at POS)                                         │
│  ┌──────────────────────────────────────────────────────────────────┐    │
│  │                        Your Order                                │    │
│  │  ──────────────────────────────────────────────────────────────  │    │
│  │  1x Burger                                    TSh 12,000         │    │
│  │  1x Fries (Large)                             TSh  5,000         │    │
│  │  1x Coke                                      TSh  2,500         │    │
│  │  ──────────────────────────────────────────────────────────────  │    │
│  │                                    Total:     TSh 19,500         │    │
│  │                                                                  │    │
│  │             "Please confirm your order with the cashier"         │    │
│  └──────────────────────────────────────────────────────────────────┘    │
│                                                                          │
│  ORDER STATUS DISPLAY (waiting area)                                     │
│  ┌──────────────────────────────────────────────────────────────────┐    │
│  │                                                                  │    │
│  │   NOW READY          |         PREPARING                       │    │
│  │   ──────────         |         ─────────                       │    │
│  │                      |                                         │    │
│  │   □□#42              |         #43  #44  #45                   │    │
```


□□#39
#46 #47
"Please collect
at counter"
MENU BOARD (entrance / above counter)
□□TODAY'S MENU
BURGERS
Classic Burger 12,000
Cheese Burger 14,000
Double Burger 18,000
▒ Veggie Burger ▒ SOLD OUT
CHICKEN
Grilled Chicken 15,000
Chicken Wings 13,000
SIDES
Fries (Reg) 3,000
Fries (Large) 5,000
Onion Rings 4,500
DRINKS
Coke / Fanta 2,500
Water 1,000
Fresh Juice 4,000
□□TODAY'S SPECIAL: Burger + Fries + Drink = TSh 17,000 □□
QUEUE DISPLAY (simple number calling)
NOW SERVING
42
Counter 1: #42
Counter 2: #38

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

11.2 Device Registration Flow





Key Rules:

- One device operates for ONE kitchen at a time
- Must logout to switch kitchens
- Device appears in dashboard after registration
- Heartbeat every 30s to show online status

11.3 Device Data Model

```
Device:
  id: uuid

  # Device identification
  deviceType:
    # Order Entry
    - POS
    - KIOSK
    - WAITER_TABLET
    # Kitchen
    - KDS
    # Customer-Facing Displays
    - CUSTOMER_FACING_SCREEN
```

```
  - ORDER_STATUS_DISPLAY
  - MENU_BOARD
  - DRIVE_THROUGH_DISPLAY
  - QUEUE_DISPLAY

deviceId: string (hardware ID or generated UUID)
deviceName: "POS-Counter-1" | "Kiosk-Entrance" | "KDS-HotLine" | "Menu-Entrance"

# Assignment
kitchenId: uuid (currently assigned kitchen)
accountId: uuid (owner account)

# For KDS - which station does it show?
stationId: uuid (nullable - null means ALL stations)

# For POS/KIOSK - which fulfillment types allowed?
allowedFulfillmentTypes: [DINE_IN, PICKUP, DELIVERY, DRIVE_THROUGH]

# For POS - which service modes allowed?
allowedServiceModes: [COUNTER, TABLE_SERVICE]

# For CUSTOMER_FACING_SCREEN - which POS is it paired with?
pairedDeviceId: uuid (nullable)

# For MENU_BOARD - display settings
menuBoardSettings:
  showPrices: true
  showSoldOut: true
  showSpecials: true
  categories: [uuid] (which categories to display, null = all)
  rotationInterval: 30 (seconds between slides, if multiple)

# For ORDER_STATUS_DISPLAY / QUEUE_DISPLAY
displaySettings:
  showPreparing: true
  showReady: true
  playSoundOnReady: true
  maxOrdersShown: 10
```

```
  # Status
  status: ACTIVE | INACTIVE | OFFLINE
  lastSeenAt: datetime

  # Metadata
  registeredAt: datetime
  registeredBy: staffId
  appVersion: string
  osVersion: string
```

11.4 KDS Station Assignment

When setting up a KDS device, admin chooses what it displays:

```
┌─────────────────────────────────────────────────────────┐
│  KDS Setup - "KDS-Kitchen-1"                            │
├─────────────────────────────────────────────────────────┤
│                                                         │
│  This display will show orders for:                     │
│                                                         │
│  ○ All Stations (shows everything)                      │
│                                                         │
│  ● Specific Station:                                    │
│    ┌─────────────────────────────────────────────────┐  │
│    │ ☑ Hot Line                                      │  │
│    │ □ Cold Line                                     │  │
│    │ □ Bar                                           │  │
│    │ □ Bakery                                        │  │
│    └─────────────────────────────────────────────────┘  │
│                                                         │
│  ○ Expeditor View (all items, controls firing)          │
│                                                         │
└─────────────────────────────────────────────────────────┘
```

11.5 POS Fulfillment Configuration

Each POS device can be restricted to specific fulfillment types:

```
┌──────────────────────────────────────────────────────────┐
│  POS Setup - "POS-Counter-1"                             │
├──────────────────────────────────────────────────────────┤
│                                                          │
│  This POS can accept orders for:                         │
│                                                          │
│  ☑ Dine-In (tables)                                      │
│  ☑ Pickup (customer collects)                            │
│  ☐ Delivery (send to address)                            │
│  ☐ Drive-Through                                         │
│                                                          │
│  Service Modes:                                          │
│                                                          │
│  ☑ Counter Service (customer at counter)                 │
│  ☑ Table Service (waiter at table)                       │
│                                                          │
└──────────────────────────────────────────────────────────┘
```

Example Device Configurations:

Device	Fulfillment Types	Service Modes	Location
POS-Counter-1	Dine-In, Pickup	Counter	Main counter
POS-Counter-2	Pickup only	Counter	Takeaway window
POS-DriveThru	Drive-Through only	Counter	DT Window
Tablet-Waiter-1	Dine-In only	Table Service	Floor
Kiosk-Entrance	Dine-In, Pickup	N/A	Entrance

11.6 Dashboard - Device Overview

Admin dashboard shows all registered devices:

```
┌──────────────────────────────────────────────────────────────────────────┐
│  Devices - Mama Lishe Downtown                                [+ Add]    │
├──────────────────────────────────────────────────────────────────────────┤
│                                                                          │
│  ORDER ENTRY DEVICES                                                     │
│  ═══════════════════                                                     │
│                                                                          │
│  POS (3)                                                                 │
```


POS-Counter-1 | Dine-In, Pickup | Counter | [Edit]
POS-Counter-2 | Pickup | Counter | [Edit]
POS-DriveThru | Drive-Through | Counter | [Edit]
WAITER TABLETS (2)
Tablet-Waiter-1 | Dine-In | Table Service | [Edit]
Tablet-Waiter-2 | Dine-In | Table Service | [Edit]
(offline 15 min)
KIOSKS (2)
Kiosk-Entrance-1 | Dine-In, Pickup | - | [Edit]
Kiosk-Entrance-2 | Dine-In, Pickup | - | [Edit]
KITCHEN DEVICES
KDS DISPLAYS (4)
KDS-HotLine | Station: Hot Line | | [Edit]
KDS-ColdLine | Station: Cold Line | | [Edit]
KDS-Bar | Station: Bar | | [Edit]
KDS-Expeditor | All Stations | | [Edit]
CUSTOMER-FACING DISPLAYS
CUSTOMER FACING SCREENS (2)
Screen-Counter-1 | Paired: POS-Counter-1 | [Edit]
Screen-Counter-2 | Paired: POS-Counter-2 | [Edit]
ORDER STATUS DISPLAYS (1)

```
│  ┌──────────────────────────────────────────────────────────────────────┐   │
│  │ □□Status-WaitingArea │ Preparing + Ready  │ Sound: On    │ [Edit] │   │
│  └──────────────────────────────────────────────────────────────────────┘   │
│                                                                             │
│  MENU BOARDS (2)                                                            │
│  ┌──────────────────────────────────────────────────────────────────────┐   │
│  │ □□Menu-Entrance      │ All Categories     │ Live         │ [Edit] │   │
│  │ □□Menu-DriveThru     │ DT Menu Only       │ Live         │ [Edit] │   │
│  └──────────────────────────────────────────────────────────────────────┘   │
│                                                                             │
│  QUEUE DISPLAYS (1)                                                         │
│  ┌──────────────────────────────────────────────────────────────────────┐   │
│  │ □□Queue-Counter      │ Now Serving        │ Sound: On    │ [Edit] │   │
│  └──────────────────────────────────────────────────────────────────────┘   │
│                                                                             │
│  DRIVE-THROUGH DISPLAYS (1)                                                 │
│  ┌──────────────────────────────────────────────────────────────────────┐   │
│  │ □□DT-OrderConfirm    │ Order Confirmation │ Lane 1       │ [Edit] │   │
│  └──────────────────────────────────────────────────────────────────────┘   │
│                                                                             │
│  ═══════════════════════════════════════════════════════════════════════   │
│  Summary: 18 devices │ 17 online │ 1 offline                                │
└─────────────────────────────────────────────────────────────────────────────┘
```

Part 12: Menu Channel Settings

12.1 Channel-Specific Visibility & Pricing

Each menu item can have different visibility and pricing per channel:

```
MenuItem:
  id: uuid
  name: "Burger"
  basePrice: 12000
  stationId: uuid  # Hot Line

  # Channel settings
```

```
  channelSettings:
    POS:
      visible: true
      price: 12000  # Same as base
    KIOSK:
      visible: true
      price: 12000
    APP:
      visible: true
      price: 13000  # +1000 for delivery convenience
    WHATSAPP:
      visible: true
      price: 13000
    TABLE_QR:
      visible: true
      price: 12000
    DRIVE_THROUGH:
      visible: true
      price: 12000

  # Fulfillment restrictions
  fulfillmentSettings:
    DINE_IN: available
    PICKUP: available
    DELIVERY: available
    DRIVE_THROUGH: unavailable  # e.g., soup not good for DT
```

12.2 Menu Item Setup UI

```
┌─────────────────────────────────────────────────────────┐
│  Edit Menu Item - Burger                                │
├─────────────────────────────────────────────────────────┤
│                                                         │
│  Name: [Burger                    ]                     │
│  Base Price: [12,000              ]                     │
│  Station: [Hot Line           ▼]                        │
│                                                         │
│  ─────────────────────────────────────────────────────  │
│  CHANNEL SETTINGS                                       │
```

```
┌──────────────────────────────────────────────────────────────┐
│                                                              │
│  Channel      | Visible | Price    |                        │
│  |————————————|—————————|————————|                        │
│  | POS          |   ☑     | 12,000   | (base)              |
│  | Kiosk        |   ☑     | 12,000   |                     │
│  | Table QR     |   ☑     | 12,000   |                     │
│  | App          |   ☑     | 13,000   | (+1,000)            |
│  | WhatsApp     |   ☑     | 13,000   | (+1,000)            |
│  | Drive-Thru   |   ☑     | 12,000   |                     │
│                                                              │
│  ————————————————————————————————————————————————————————  │
│  FULFILLMENT AVAILABILITY                                    │
│  ————————————————————————————————————————————————————————  │
│                                                              │
│  ☑ Dine-In                                                   │
│  ☑ Pickup                                                    │
│  ☑ Delivery                                                  │
│  ☐ Drive-Through (not suitable)                              │
│                                                              │
└──────────────────────────────────────────────────────────────┘
```

12.3 Use Cases

Scenario	Configuration
Alcohol only on POS	Beer: visible on POS only, hidden on Kiosk/App
Delivery markup	All items: +1000 on App/WhatsApp channels
Soup not for drive-through	Soup: fulfillment DRIVE_THROUGH = unavailable
Happy hour pricing	Beer: different price on POS during specific hours
Dine-in exclusive	Special dish: fulfillment = DINE_IN only

Part 13: Data Model Overview

13.1 Core Entities

Kitchen

- Has many staff members
- Has many menu items
- Has configuration settings
- Has subscription tier and enabled features
- Has many orders, tabs, customers

Customer

- Can have App Wallet (platform-wide)
- Can have Kitchen Wallet (kitchen-specific)
- Can have order history
- Identified by phone number or account

Tab (PROFESSIONAL+)

- Belongs to a kitchen
- Associated with a table number
- Has many orders
- Has status (OPEN, CLOSED)
- Can have tab-level discounts

Order

- Belongs to a kitchen
- Optionally belongs to a tab
- Optionally linked to a customer
- Has many order items
- Has order status and payment status
- Has channel source
- Has fulfillment type
- Has order number (daily reset)
- Has delivery status (for delivery orders)

Order Item

- Belongs to an order
- References a menu item
- Has quantity, modifiers, notes
- Has item status (for kitchen tracking)
- Has line total
- Has station assignment (from menu item)

Payment

- Linked to an order or tab
- Has payment method
- Has amount and status

- Has reference/confirmation code

Receipt

- Generated on payment completion
- Has unique receipt number
- Contains snapshot of transaction
- Immutable (only voided, not modified)

13.2 Subscription & Features

```
Kitchen:
  id: uuid
  name: string

  subscription:
    plan: STARTER | GROWING | PROFESSIONAL | ENTERPRISE
    status: ACTIVE | TRIAL | PAST_DUE | CANCELLED
    started_at: datetime
    current_period_end: datetime

  usage:
    orders_this_month: integer
    menu_items_count: integer
    staff_accounts_count: integer
    locations_count: integer

  features: # Derived from plan
    pos_enabled: boolean
    kiosk_enabled: boolean
    table_qr_enabled: boolean
    whatsapp_enabled: boolean
    tabs_enabled: boolean
    stations_enabled: boolean
    kds_enabled: boolean
    drive_through_enabled: boolean
    team_enabled: boolean
    multi_location_enabled: boolean

  profile: # From onboarding
```

```
    business_type: HOME_KITCHEN | FOOD_STALL | RESTAURANT | CHAIN
    expected_orders_per_day: string
    team_size: string
    service_styles: [DELIVERY, PICKUP, DINE_IN, DRIVE_THROUGH]
```

13.3 Entity Relationships

```
Account
├── has many → Kitchens
└── has many → Devices (registered under this account)

Kitchen
├── belongs to → Account
├── has many → Staff
├── has many → Menu Items
├── has many → Customers
├── has many → Orders
├── has many → Tabs
├── has many → Payments
├── has many → Receipts
├── has one → Kitchen Settings
├── has one → Subscription
├── has many → Payment Methods
├── has many → Kitchen Stations
├── has many → Kitchen Printers
└── has many → Devices (assigned to this kitchen)

Station
├── belongs to → Kitchen
├── has many → Menu Items (assigned)
├── has many → KDS Devices (showing this station)
└── has one → Printer (optional dedicated printer)

Device
├── belongs to → Account
├── assigned to → Kitchen (one at a time)
├── assigned to → Station (for KDS, nullable)
└── has → Configuration (fulfillment types, service modes)
```

```
Tab
├── belongs to → Kitchen
├── has many → Orders
└── has one → Payment (on close)

Order
├── belongs to → Kitchen
├── belongs to → Tab (optional)
├── belongs to → Customer (optional)
├── has many → Order Items
├── has many → Payments
├── has → Delivery Info (for delivery orders)
└── generates → Receipt

MenuItem
├── belongs to → Kitchen
├── belongs to → Category
├── assigned to → Station
└── has → Channel Settings (visibility, pricing per channel)
```

13.4 Order with Delivery Status

For delivery orders, the Order tracks both preparation and delivery status:

```
Order:
  id: uuid
  orderNumber: 47
  kitchenId: uuid

  # Core status
  status: CREATED | PENDING_PAYMENT | CONFIRMED | PREPARING | READY | COMPLETED | CANCELLED
  paymentStatus: UNPAID | PENDING | PAID | REFUNDED

  # Channel & Fulfillment
  channel: POS | KIOSK | TABLE_QR | APP | WHATSAPP | DRIVE_THROUGH
  fulfillmentType: DINE_IN | PICKUP | DELIVERY | DRIVE_THROUGH

  # For DINE_IN
  tableNumber: string (nullable)
```

```
# For DRIVE_THROUGH
laneNumber: integer (nullable)
vehicleDescription: string (nullable)

# For DELIVERY - Shipping Status
deliveryStatus:
  - PENDING           # Waiting for driver assignment
  - DRIVER_ASSIGNED   # Driver accepted
  - DRIVER_ARRIVED    # Driver at restaurant
  - PICKED_UP         # Driver has food
  - IN_TRANSIT        # On the way to customer
  - ARRIVED           # Driver at customer location
  - DELIVERED         # Handed to customer
  - FAILED            # Delivery failed

deliveryProvider: JIKOXPRESS_DASHERS | BOLT | UBER_DIRECT | null
deliveryExternalId: string (provider's order ID)

driverInfo:
  id: uuid
  name: string
  phone: string
  vehicle: string
  photo: url
  currentLocation: {lat, lng}
  eta: datetime

deliveryAddress:
  street: string
  building: string
  floor: string
  instructions: string
  coordinates: {lat, lng}
```

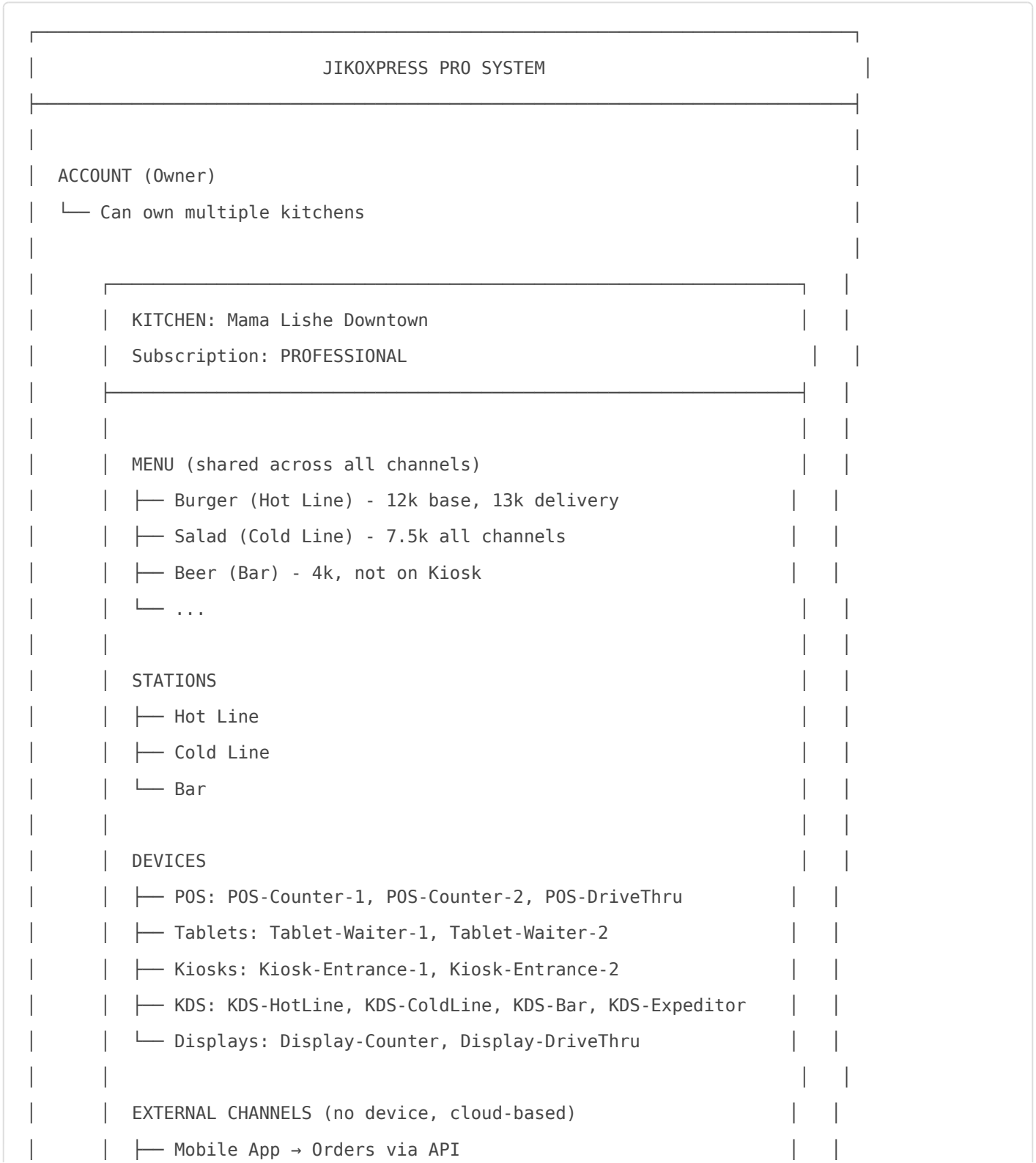
Two Parallel Statuses for Delivery Orders:

- `status` → Kitchen/preparation status (PREPARING → READY)
- `deliveryStatus` → Shipping/logistics status (PICKED_UP → IN_TRANSIT → DELIVERED)

Part 14: Unified System Architecture

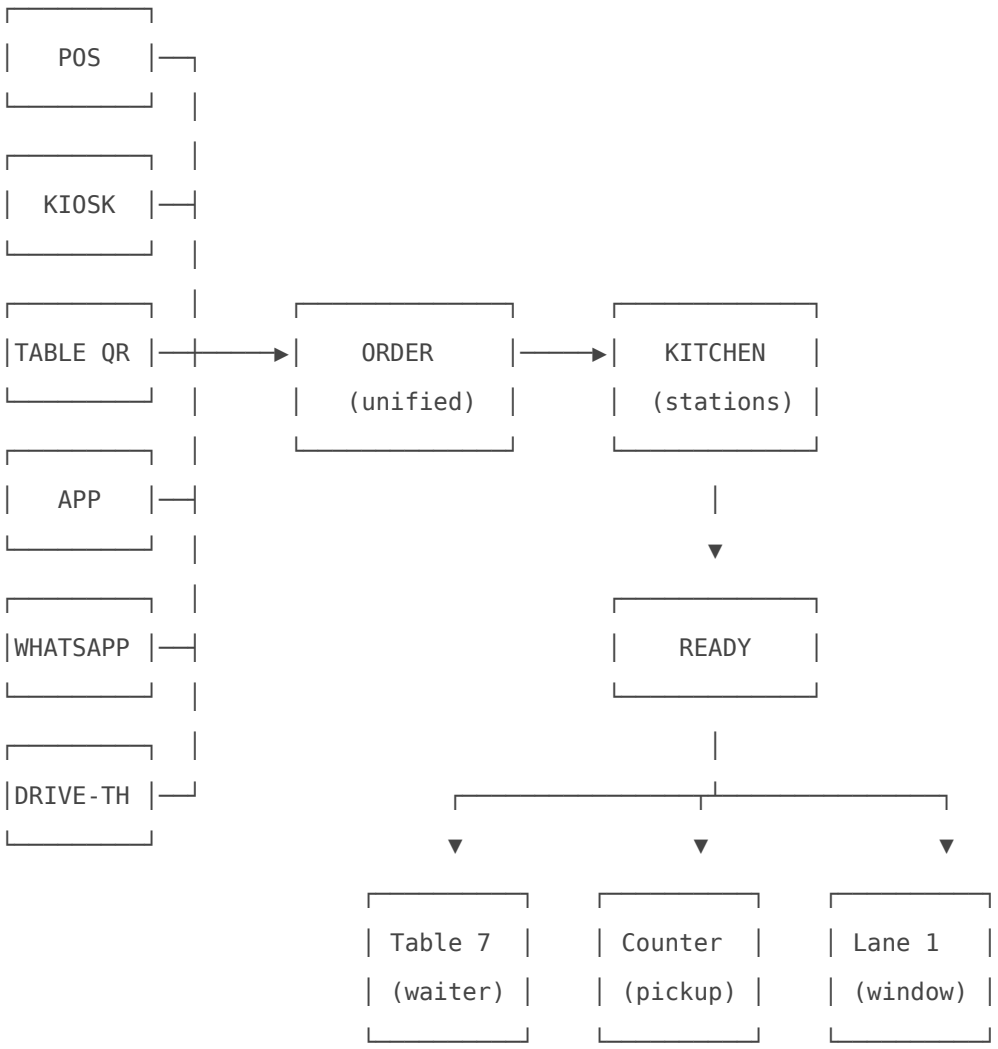
14.1 All Channels, One Kitchen

A single restaurant can enable all channels while sharing the same kitchen and menu:



```
│      │  ├── WhatsApp Bot → Orders via webhook                          │   │
│      │  └── Table QR → Orders via web/app                              │   │
│      │                                                                 │   │
│      └─────────────────────────────────────────────────────────────────┘   │
│                                                                            │
└────────────────────────────────────────────────────────────────────────────┘
```

14.2 Order Flow - All Channels to One Kitchen



14.3 Kitchen Unified Queue

Kitchen sees ONE unified queue with orders from all channels:

```
┌────────────────────────────────────────────────────────────────────────────┐
│  KITCHEN DISPLAY (All Orders)                                  2:34 PM     │
├────────────────────────────────────────────────────────────────────────────┤
```

```
┌──────────────────────────────────────────────────────────────────────────┐
│                                                                          │
│  #41          │  #42          │  #43          │  #44          │  #45     │
│  □□APP        │  □□□KIOSK     │  □□POS        │  □□DRIVE      │  □□WHATS │
│  DELIVERY     │  Table 12     │  Table 7      │  Lane 1       │  PICKUP  │
│  3 min ago    │  2 min ago    │  1 min ago    │  Just now ⚡  │  Just now │
├───────────────┼───────────────┼───────────────┼───────────────┼──────────┤
│  1x Burger    │  1x Burger    │  1x Burger    │  1x Burger    │  1x Fish │
│  1x Salad     │  1x Fries     │  1x Salad     │  1x Fries     │          │
│               │               │  1x Beer      │  1x Coke      │          │
│               │               │               │               │          │
│  [START]      │  [WORKING]    │  [START]      │  [PRIORITY]   │  [START] │
└──────────────────────────────────────────────────────────────────────────┘
```

Key Points:

- Same menu shared across all channels (with per-channel visibility/pricing)
- Same kitchen queue (channel shown as badge)
- Same stations (items route based on menu setup)
- Different handoff points (table, counter, window, delivery)
- Drive-through flagged as priority (⚡)

Part 15: Delivery Integration (Future)

15.1 Multiple Delivery Providers

JikoXpress Pro will support multiple delivery providers:

Planned Providers:

- JIKOXPRESS_DASHERS (own fleet)
- BOLT_FOOD
- UBER_DIRECT
- SAFEBODA (East Africa specific)

15.2 Kitchen Configuration

```
deliverySettings:
  enabledProviders: [JIKOXPRESS_DASHERS, BOLT_FOOD]
  preferredProvider: JIKOXPRESS_DASHERS
```

```
  fallbackProvider: BOLT_FOOD
  autoAssign: true | false
```

15.3 Order Delivery Properties

```
Order (when fulfillment = DELIVERY):
  deliveryProvider: string (nullable)
  deliveryExternalId: string (provider's order ID)
  deliveryStatus: FINDING_DRIVER | ASSIGNED | PICKED_UP | DELIVERING | DELIVERED
  driverInfo:
    name: string
    phone: string
    vehicle: string
    eta: datetime
```

Part 16: Operational Scenarios

16.1 Home Chef Scenario (STARTER)

```
Context: Mama Lishe cooking from home, delivery only

Setup:
- Downloaded mobile app
- Added 10 dishes with photos
- Set operating hours: 11am - 8pm
- Enabled mobile money payments

Daily Flow:
1. Customer finds Mama Lishe on JikoXpress app
2. Customer orders Ugali + Samaki
3. Customer pays via M-Pesa
4. Mama Lishe gets push notification □□
5. Mama Lishe taps "Accept"
6. Mama Lishe cooks, taps "Ready"
7. Customer picks up (or delivery arranged separately)
8. Order completed
```

```
No POS, no printer, no complexity. Just phone and cooking.
```

16.2 Busy Food Stall Scenario (GROWING)

```
Context: Popular food stall, high lunch traffic

Setup:
- Tablet with POS app
- Kitchen ticket printer
- Receipt printer
- 2 staff accounts

Daily Flow:
1. Customer orders at counter
2. Staff enters order on tablet POS
3. Customer pays cash or mobile money
4. Kitchen ticket prints automatically
5. Receipt prints for customer
6. Customer waits for number
7. Order called when ready
```

16.3 Restaurant with Dine-In (PROFESSIONAL)

```
Context: Full restaurant, tables, bar

Setup:
- POS terminals
- KDS in kitchen
- Multiple stations (hot line, cold line, bar)
- Table QR codes on all tables
- Tabs enabled

Scenario A - Waiter Service:
1. Customers seated at Table 7
2. Waiter takes order on POS
3. Selects "Pay Later" → Tab created
4. Orders split to stations on KDS
```

```
5. Customers order more drinks (added to tab)
6. Customers request bill
7. Waiter prints bill
8. Customers pay
9. Tab closed, receipt printed

Scenario B - Table QR:
1. Customer scans QR at table
2. Orders on phone (no app needed)
3. Pays via USSD
4. Kitchen receives order with "Table 7"
5. Food delivered to table
```

16.4 Drive-Through Scenario (PROFESSIONAL)

```
Context: Fast food with drive-through

Setup:
- Speaker/mic at order point
- POS at order window
- KDS showing drive-through orders with priority
- Single window (pay + pickup)

Flow:
1. Car enters Lane 1
2. Customer orders via speaker
3. Staff enters on POS: "Lane 1, Order #23"
4. Customer drives to window
5. Staff processes payment
6. Kitchen already preparing (priority queue)
7. Order ready, handed to customer
8. Car exits, time logged

Metrics tracked:
- Order time: 45 seconds
- Payment time: 30 seconds
- Pickup wait: 2 minutes
```

```
- Total time: 3:15 ✓ (target: < 4 min)
```

16.5 Multi-Location Chain (ENTERPRISE)

```
Context: 5 restaurant locations

Setup:
- Central menu management
- Each location has own POS, KDS, printers
- Consolidated reporting dashboard
- Staff managed per location with central oversight

Daily Operations:
- Menu update pushed to all 5 locations
- Each location operates independently
- HQ sees real-time sales across all locations
- End of day: consolidated report generated
- Inventory synced across locations
```

Part 17: Non-Functional Requirements

17.1 Performance

- Checkout session creation: < 200ms
- Kitchen notification: < 500ms from trigger
- Print job creation: < 200ms
- KDS refresh: Real-time (< 2s)
- Drive-through order entry: < 30 seconds target

17.2 Reliability

- Print job retry on failure
- Payment webhook idempotency
- Offline POS capability (queue and sync)
- KDS fallback to printer if display fails

17.3 Scalability

- Support 100+ concurrent orders per kitchen
- Support 10,000+ orders per day per kitchen (enterprise)
- Support 100+ locations per enterprise account

17.4 Security

- Payment data encryption
- Staff authentication
- Manager PIN for sensitive operations
- Audit log for all transactions
- Receipt data immutability

Part 18: API Structure Overview

18.1 Checkout APIs

- Create checkout session
- Process payment
- Cancel session
- Tab management (find, create, close)

18.2 Order APIs

- Create order
- Get order details
- Update order status
- Cancel order
- Void item

18.3 Kitchen APIs

- Get pending orders (by station)
- Update item/order status
- Fire order (expeditor)

18.4 Payment APIs

- Get payment status
- Confirm manual payment

- Handle webhooks
- Request refund

18.5 Print APIs

- Get pending print jobs
- Claim/complete/fail print job
- Manual reprint triggers

18.6 Subscription APIs

- Get current plan
- Get feature flags
- Get usage stats
- Upgrade/downgrade plan
- Unlock feature module

18.7 Device APIs

- Register device
- Get device configuration
- Update device settings
- Device heartbeat
- List devices for kitchen

18.8 Station APIs

- Create station
- Update station
- Delete station
- Assign menu items to station
- Assign KDS to station

Appendix A: Glossary

Term	Definition
Tab	Container for unpaid orders at a dine-in table
Stub	Small printed slip with order number
KDS	Kitchen Display System - digital order screens

Term	Definition
Expeditor	Senior staff controlling order flow
Station	Specific preparation area in kitchen
Bump	Mark order complete on KDS
Fire	Send order to kitchen for preparation
Void	Cancel an item from an order
Kitchen Wallet	Prepaid balance issued by kitchen
App Wallet	Platform-wide customer wallet
Lane	Drive-through vehicle queue
Device	Registered hardware (POS, Kiosk, KDS, Tablet, Display)
Channel	Source of order (POS, Kiosk, App, WhatsApp, etc.)
Fulfillment	How order is delivered (Dine-in, Pickup, Delivery, Drive-through)
Service Mode	POS operation style (Counter Service, Table Service)
Delivery Status	Shipping progress for delivery orders
Account	Owner entity that can have multiple kitchens
Customer Facing Screen	Display at POS showing cart to customer
Menu Board	Digital display showing menu items and prices
Order Status Display	Screen showing preparing/ready orders
Queue Display	Screen showing "Now Serving" numbers

Appendix B: Configuration Checklist

STARTER Setup

- ☐ Download mobile app
- ☐ Set business name and logo
- ☐ Add menu items (up to 20)
- ☐ Set operating hours
- ☐ Enable mobile money

GROWING Setup (+ STARTER)

- [ ] Set up POS on tablet/desktop
- [ ] Configure kitchen printer
- [ ] Configure receipt printer
- [ ] Add staff accounts (up to 3)
- [ ] Enable additional payment methods
- [ ] Generate Table QR codes

PROFESSIONAL Setup (+ GROWING)

- [ ] Define kitchen stations
- [ ] Set up KDS displays
- [ ] Configure station assignments for menu items
- [ ] Enable tabs
- [ ] Set staff roles and discount limits
- [ ] Configure drive-through (if applicable)

ENTERPRISE Setup (+ PROFESSIONAL)

- [ ] Add additional locations
- [ ] Configure central menu
- [ ] Set up consolidated reporting
- [ ] Configure API access
- [ ] Set up inventory tracking

Appendix C: Decision Log

Decision	Choice	Rationale
Credit system (Kwa Deni)	Removed	Complexity vs V1 scope
Tab creation	Auto on first pay-later	Less friction for staff
Partial payments	Defer to V2	Complexity vs actual need
Drive-through	Channel + Fulfillment	Unified model, same order entity
Feature unlocking	Subscription-based	Clear value tiers, simple UX
Default config	Zero-config works	Reduce onboarding friction

Decision	Choice	Rationale
Print architecture	Client pull	Local printer support, offline
Order number	Daily reset	Easier for kitchen communication
Platform scale	One codebase all tiers	Same APIs, feature flags gate access
Device registration	Email + code + kitchen select	Secure, supports multi-kitchen accounts
One device one kitchen	Must logout to switch	Simplifies operations, prevents confusion
Station assignment	Pre-defined in menu setup	Automatic routing, no manual selection
Menu channel settings	Per-channel visibility/pricing	Flexibility for delivery markup, restrictions
Delivery status	Separate from order status	Two parallel tracking (kitchen vs shipping)
POS service modes	Counter vs Table service	Same device, different workflows

---

Document History

Version	Date	Changes
1.0	December 2025	Initial document
2.0	January 2026	Removed Kwa Deni, added Drive-Through, added Progressive Feature Unlocking, added Subscription Tiers, added Default Configuration, noted Multi-Provider Delivery
2.1	January 2026	Added Device Management (registration, types, configuration), expanded Station architecture (creation, assignment, routing), added Menu Channel Settings (per-channel visibility/pricing), added POS Service Modes (counter vs table service), added Order Delivery Status tracking, added Unified System Architecture diagrams

---

End of Document

Revision #6
Created 29 December 2025 19:01:46 by Admin Qbit
Updated 2 January 2026 22:52:29 by Admin Qbit