

CASE STUDY 01 OF 04 | ONE SALE | B2B FOOD ORDERING PLATFORM

Agent Account System & Field Operations Model

How the platform was built around the agent as the primary actor, not the end consumer

#B2B #AgentModel #FieldOps #AccountManagement

Overview

One Sale is a B2B food ordering platform built around a model that does not exist in consumer apps: the sales agent. In One Sale, the person placing the order is not buying food for themselves. They are a field sales representative placing bulk food orders on behalf of the shops and businesses they manage in their territory. Every design decision in the platform flows from this distinction.

This case study covers how the agent account system was designed, what the agent's operational workflow looks like within the app, and why building around the agent as the primary actor rather than the end customer required a fundamentally different approach to account structure, data ownership, and session management than a standard food ordering platform.

Project Snapshot

Project	One Sale — B2B Food Ordering Platform
Feature Area	Agent Account System and Field Operations Model
Role	Senior Fullstack Engineer
Primary User	Field sales agents managing multiple customer shops
Business Model	B2B: agent places orders on behalf of shop customers
Outcome	An agent-centric account system supporting territory-based multi-customer operations

The B2B Context

In a consumer food app, the user ordering is always the person consuming. In One Sale, this relationship is broken deliberately. The agent is a distribution intermediary: they visit shops in their assigned territory, understand what each shop needs, and place orders through the platform on the shop's behalf. The shop owner may never interact with the One Sale app directly.

This creates a data model that is meaningfully different from a consumer platform. An agent account is not just a user account. It is the operational hub for a portfolio of customer businesses. Everything the agent does, from browsing the catalogue to placing an order, happens in the context of a specific customer relationship they are managing.

What Was Built

Agent Account Registration and Onboarding

Agent accounts are created through an admin-approved registration flow rather than open self-signup. An agent submits their details and territory information, and an admin approves or rejects the application. This gatekeeping is intentional: agents represent the business in the field, and each agent account carries significant operational responsibility for the orders and payments associated with their customer portfolio.

On approval, the agent receives credentials and completes a structured onboarding: adding their first customers, configuring their profile details, and verifying their identity. The onboarding is designed to be completable in under 15 minutes on a mobile device, since many agents are doing this between field visits.

Customer Management Under an Agent Account

Each agent maintains a portfolio of customer records within their account. A customer in One Sale represents a shop or business that the agent serves. Each customer record holds:

- Business name, address, and contact details for the shop
- Delivery address which may differ from the business address
- Order history specific to that customer
- Payment status and outstanding balance for that customer
- Notes field for agent-specific context about the customer's preferences or requirements

The customer list is the agent's primary navigation surface. When the agent starts a new session to place an order, they select the customer they are ordering for before anything else. Every subsequent action in the session, the catalogue they browse, the cart they build, the order they place, is scoped to that customer.

Territory and Quota Visibility

Agents work within assigned territories. The agent's dashboard surfaces territory-level information alongside their individual activity: total orders placed this period, total value ordered, number of active customers, and any performance benchmarks set by the platform administrators. This gives field agents situational awareness about their own territory performance without requiring them to pull reports.

Wishlist as a Field Planning Tool

The wishlist in One Sale functions differently from a consumer wishlist. An agent's wishlist is a pre-ordering planning surface. When an agent visits a shop and discusses what the shop needs, they can add items to the wishlist tagged against that customer. Later, when they are ready to convert the visit notes into an actual order, the wishlist gives them a structured starting point rather than forcing them to rebuild the selection from scratch.

A single agent can maintain separate wishlists for different customers simultaneously, so field visits to multiple shops can all be captured in context before any orders are formally placed.

Technical Architecture

Agent-Customer Data Hierarchy The data model places agents at the top of the hierarchy, with customers as child entities owned by an agent. Orders, carts, and wishlists all carry both an agent ID and a customer ID, making it possible to filter activity by either dimension independently.	Approval-Gated Registration Agent registration uses a two-step flow: application submission creates a pending account record, and admin approval activates it. Pending accounts cannot authenticate. The approval queue is surfaced in the admin dashboard with agent details and territory information for review.
Session Scoping by Customer The application enforces an active customer context at the session level. Cart contents, wishlist views, and order flows all require a selected customer to be active. Switching customer context resets the cart but preserves draft state in the previous customer's wishlist.	Multi-Device Agent Support Agents work across devices: mobile during field visits, desktop for order review and reporting. Account state (customers, wishlists, draft orders) is synchronized across devices through the server-side session model. There is no device-specific state that could cause inconsistency.
Customer Notes and History Customer notes are stored as a timestamped log rather than a single editable field. Each note entry records what was written and when, giving agents a running field journal for each customer rather than a single overwritten note.	Territory Performance Aggregation Dashboard metrics are pre-aggregated on a scheduled basis rather than computed live on page load. A background job rolls up order counts, values, and customer activity per agent at regular intervals, keeping the dashboard fast regardless of order volume.

Key Challenges

Designing for Field Use on Mobile

Field agents are often using the app standing in a shop, on a slow mobile connection, with one hand occupied. Every workflow in the agent-facing interface was evaluated against this scenario. Customer selection is a single-tap from the agent's home screen. Wishlist additions require two taps. The cart is accessible from any screen. Forms use large touch targets and auto-advance between fields where possible. Loading states degrade gracefully on slow connections rather than blocking interaction.

Customer Portfolio Scale

A senior agent might manage 80 or more customer accounts. Displaying all of them in a flat scrollable list becomes unusable at that scale. The customer list supports search by name, address, and notes content, and allows the agent to pin their most frequently visited customers to the top of the list. Recently ordered-from customers are surfaced in a quick-access section, since agents typically follow repeating visit cycles across their territory.

Preventing Cross-Customer Order Errors

Placing an order intended for one customer against a different customer's account is a serious operational error in a B2B context: it generates incorrect invoicing, misattributed payment records,

and fulfilled orders that go to the wrong account. The platform enforces customer context at multiple points in the order flow, showing the active customer name prominently throughout the cart and checkout, and requiring explicit confirmation before an order is submitted with a summary that makes the customer identity unmistakable.

Outcome

The agent account system gave field sales representatives a purpose-built operational tool rather than a consumer app they were expected to adapt to their B2B workflow. The customer portfolio model let agents manage large numbers of shop relationships within a single account without losing context. The wishlist-as-field-journal approach connected the physical act of visiting a shop to the digital act of placing the order, bridging a gap that had previously been handled with paper notes and memory. Mobile-optimized interactions made the app practical to use in the exact conditions field agents actually work in.

Architecture Highlights

- Approval-gated registration keeping the agent network quality-controlled
- Customer-scoped session model preventing cross-account order attribution errors
- Per-customer wishlist functioning as a structured field visit journal
- Pre-aggregated territory dashboard metrics keeping performance views instant at any scale