FOODORA — Product Requirements Document (PRD)

Product: FOODORA — Order & Pick-up (MVP)
Doc Type: Product Requirements Document (PRD)
Version: 1.0
Status: Final (Portfolio-ready)
Date: 2026-01-07
Owner: Business Analyst
Audience: Restaurant Owner, Product/BA, Designer, Developer, QA

0) Document Control

0.1 Version History

- v1.0 (2026-01-07): Initial PRD for Phase 1 MVP (Order & Pick-up)

0.2 Definitions (Power, Interest)

- **Power:** authority to approve scope/budget OR operational influence that can block go-live.
- **Interest:** frequency of interaction + impact of change on daily workflow.

1) Executive Summary

FOODORA is an **Order & Pick-up** digital solution for small/medium restaurants aiming to reduce dependency on third-party delivery platforms, improve operational efficiency, and enhance customer experience through clear ordering and real-time order status.

FOODORA includes:

- **Customer Mobile App:** browse menu, add to cart, select pickup time, place order, track status.
- **Restaurant Web Dashboard:** manage menu, confirm/reject orders, update status, view basic performance.

Phase 1 (MVP): Pickup only, **Pay at Counter**.
Phase 2 (Future): Online payment, loyalty/membership, advanced analytics (out of scope for Phase 1).

2) Background & Business Context

2.1 Current Situation (As-Is)

Restaurant XYZ:

- Relies on third-party delivery platforms (e.g., GrabFood, ShopeeFood)
- Pays **25–30% commission** per order
- Processes orders via manual notes and fragmented channels
- Does not own customer data (phone number, order history)
- Encounters frequent order confusion and delays during peak hours

2.2 Business Problem

- **Financial:** commission reduces profit margins
- **Operational:** manual + disconnected workflow causes errors and slows service
- **Data:** no direct customer data to support retention
- **CX:** unclear order status → customers repeatedly ask staff → more chaos during peak

2.3 Business Objectives

Short-term (MVP)

- Reduce platform dependency by shifting order share to FOODORA
- Standardize order processing workflow
- Reduce average order handling time during peak hours (baseline → target improvement)

Long-term

- Build a direct customer database and improve repeat purchases
- Enable future scalability (multi-branch readiness)

---

3) Success Metrics, Acceptance Criteria, and Exit Criteria

3.1 KPIs (Targets)

- **KPI-01 Order Share:** ≥ 30% of total orders placed via FOODORA within 8–12 weeks post-launch
- **KPI-02 Order Error Rate:** < 2%

- **KPI-03 Processing Time:** average < 5 minutes

3.2 KPI Definitions

- **Order Error:** wrong item / missing item / wrong quantity / incorrect pickup time recorded / status mismatch causing customer pickup failure.
- **Processing Time (primary):** Confirmed → Ready elapsed time (recommended operational metric).
- **Handling Time (secondary):** Placed → Confirmed elapsed time (front-desk responsiveness metric).

3.3 Release (Go-Live) Exit Criteria

- All **Must-have** user stories pass UAT (critical defects = 0 open)
- Owner signs off UAT
- Staff training completed; menu and pickup rules configured
- Monitoring/logging enabled; rollback plan documented

---

4) Scope

4.1 In Scope — Phase 1 (MVP)

Customer App

- Menu browsing (categories/items/item detail)
- Cart (add/remove/update qty)
- Pickup time selection
- Place order (pay at counter)
- Order tracking (status timeline)
- View order history (minimal)

Web Dashboard

- Staff authentication + role-based access
- Order queue + order details
- Confirm/reject with reason
- Status updates: Confirmed/Preparing/Ready/Completed

- Menu management + availability toggle
- Basic reporting (daily summary)

4.2 Out of Scope — Phase 1

- Delivery, courier management, delivery tracking
- Loyalty/membership program
- AI recommendations
- Online payment processing
- Multi-branch configuration and cross-branch reporting

---

5) Stakeholders & Users

5.1 Stakeholders

- **Restaurant Owner (Sponsor/Approver):** scope/budget decisions, UAT sign-off
- **Cashier/Front Desk (Primary Operator):** daily dashboard user; UAT execution
- **Kitchen Staff:** receives order info; cares about clarity and accuracy
- **Customers:** end users of mobile ordering and status tracking

5.2 Communication Strategy (Summary)

- Owner: weekly checkpoint + milestone demos; decision log and approvals
- Cashier: workshops + weekly backlog review; usability + edge case validation; UAT walkthrough
- Kitchen: short demos focused on workflow; peak-hour readiness
- Customers: onboarding copy, QR instructions, status meaning clarity

---

6) Personas (Condensed)

6.1 Customer Persona (An)

- Busy office worker, time-sensitive lunch ordering
- Needs fast ordering, reliable pickup time, clear “Ready” signal

6.2 Cashier Persona (Binh)

- Peak-hour multitasker, hates manual notes
- Needs minimal clicks, clear queue, quick status updates, low confusion

---

7) Assumptions, Constraints, Dependencies

7.1 Assumptions

- Restaurant has stable Wi-Fi in cashier/kitchen areas
- Staff can operate dashboard after training
- Customers have smartphones and can receive on-screen status updates

7.2 Constraints

- Limited budget
- MVP timeline ≤ 3 months
- Small IT team

7.3 Dependencies

- Backend/API hosting and database availability
- Notification strategy (MVP: in-app refresh; push notifications optional later)
- Device availability for dashboard (PC/tablet)

---

8) Information Architecture (Screens & Navigation)

8.1 Customer App Screens (MVP)

1. Home / Categories
2. Menu List (by category)
3. Item Detail
4. Cart
5. Pickup Time Selection
6. Checkout (Pay at Counter)
7. Order Confirmation
8. Order Tracking (Status timeline)

9. Order History (minimal)

8.2 Dashboard Screens (MVP)

1. Login
2. Order Queue (tabs by status)
3. Order Detail
4. Menu Management
5. Reporting (daily summary)

---

9) Core Flows (End-to-End)

9.1 Flow A — Customer Places Pickup Order (Happy Path)

1. Browse menu → add items → cart
2. Select pickup time slot
3. Checkout → confirm order (pay at counter)
4. Order created with Order ID
5. Restaurant confirms → preparing → ready
6. Customer picks up → completed

9.2 Flow B — Restaurant Processes Order

1. Order appears in “Placed” queue
2. Cashier confirms (or rejects with reason)
3. Kitchen prepares
4. Cashier marks Ready
5. Customer pickup → cashier marks Completed

9.3 Flow C — Out of Stock / Reject

- Restaurant rejects with reason before confirming
- Customer sees status Cancelled + reason; encouraged to reorder

9.4 Flow D — Customer Cancels Before Confirmation

- Customer cancels while status = Placed

- Status becomes Cancelled; dashboard reflects cancellation

---

10) Data Model (High-Level)

Entities (minimum viable):

- **User** (customer): id, phone, name(optional), createdAt
- **Restaurant**: id, name, address, openingHours
- **MenuCategory**: id, restaurantId, name, sortOrder
- **MenuItem**: id, restaurantId, categoryId, name, description, price, imageUrl(optional), isAvailable, updatedAt
- **Order**: id, restaurantId, customerId(optional), customerPhone, pickupTime, status, subtotal, total, notes(optional), createdAt, confirmedAt, readyAt, completedAt, cancelledAt, cancelReason(optional)
- **OrderItem**: id, orderId, menuItemId, nameSnapshot, priceSnapshot, qty

Audit (recommended):

- **OrderStatusLog**: id, orderId, fromStatus, toStatus, changedBy, changedAt

---

11) Functional Requirements (FR)

11A) Customer Mobile App

FR-APP-01 Menu Browsing

- View categories and items.
- View item detail (name, price, availability; optional image/description).

FR-APP-02 Search (Should)

- Search items by keyword.
- Display results quickly and handle empty results state.

FR-APP-03 Cart

- Add/remove/update quantity.
- Show subtotal/total.
- Validate availability at checkout time.

FR-APP-04 Pickup Time Selection

- Show available time slots.
- Prevent choosing unavailable/expired slots.
- Display chosen pickup time consistently (checkout + confirmation + tracking).

FR-APP-05 Place Order

- Collect minimum: phone number; optional name/note.
- Submit order and receive Order ID.
- Show confirmation with next steps (“Pay at counter”, pickup instructions).

FR-APP-06 Order Tracking

- Show current status and timestamps (where available).
- Show status descriptions (what each status means).

FR-APP-07 Cancellation

- Customer can cancel only before confirmation (Placed).
- If status is Confirmed or later, cancellation option is disabled with explanation.

FR-APP-08 Order History (Should)

- Customer can view past orders (basic list).

11B) Restaurant Web Dashboard

FR-DASH-01 Auth & Roles

- Login required.
- Roles:
  - **Owner/Admin:** full access
  - **Cashier/Operator:** orders + limited menu availability toggle

FR-DASH-02 Order Queue

- Tabs/filters by status (Placed/Confirmed/Preparing/Ready/Completed/Cancelled).
- Display key info: pickupTime, orderId, item count, customer phone, status.

FR-DASH-03 Order Detail

- Show items, quantities, notes, pickup time, timestamps.
- Show actions allowed by current status.

FR-DASH-04 Confirm / Reject

- Confirm transitions Placed → Confirmed.
- Reject requires reason and transitions to Cancelled (with reason).

FR-DASH-05 Status Management

- Allowed transitions:
  - Confirmed → Preparing → Ready → Completed
- Record timestamp + operator for each transition.
- Prevent invalid transitions (e.g., Ready → Preparing).

FR-DASH-06 Menu Management

- CRUD categories and items.
- Toggle availability quickly; changes reflected to customer app.

FR-DASH-07 Reporting

- Daily summary: orders count, total revenue, average processing time.
- View order history with status and timestamps.

FR-DASH-08 Settings (Minimal)

- Configure pickup slot interval (e.g., 10 minutes).
- Optional: max orders per slot (capacity cap).

---

12) Business Rules & Constraints

12.1 Order Status Definitions

- **Placed:** customer submitted order
- **Confirmed:** restaurant accepted order
- **Preparing:** kitchen started preparation
- **Ready:** ready for pickup
- **Completed:** handed to customer

- **Cancelled:** cancelled by customer (before confirm) or rejected by restaurant (with reason)

12.2 Cancellation Rules

- Customer cancellation only when status = Placed
- Restaurant cancellation allowed before Ready; must provide reason
- Once Ready, cancellation is restricted (optional operational rule)

12.3 Pickup Slot Rules

- Slots generated based on interval and restaurant business hours
- Slots cannot be in the past
- Optional capacity:
    - max orders per slot
    - if reached, slot becomes unavailable

12.4 Price Rules

- Total = sum(priceSnapshot × qty)
- Price snapshot is captured at time of order creation to avoid later menu price changes affecting old orders

12.5 Error Handling Rules (User-Friendly)

- Availability change during checkout → block submit; show which items are unavailable
- Slot becomes unavailable during submit → prompt customer to choose another slot
- Dashboard transition conflict (two operators) → lock or “last write wins” with warning (implementation choice)

---

13) User Stories + Acceptance Criteria + MoSCoW

Epic 1 — Menu Management

US-MENU-01 (Must) Create Category

- As an Admin, I want to create a menu category so that customers can browse items logically.
- **Acceptance Criteria**

    - Given I am logged in as Admin, when I create a category with a non-empty name, then the category is saved and appears in the category list.
    - Given a category name already exists (case-insensitive), when I try to create a duplicate, then the system prevents saving and shows an error message.
    - Given I enter a name longer than the allowed limit (e.g., 50 chars), then the system shows validation error and blocks saving.

US-MENU-02 (Must) Create Menu Item

- As an Admin, I want to create a menu item so that customers can order it.
- **Acceptance Criteria**
    - Given I am Admin, when I create an item with required fields (name, price, category), then the item appears in the menu list.
    - Given price ≤ 0, when I save, then the system blocks saving and shows validation error.
    - Given I leave name empty, when I save, then validation error is shown.
    - Given I set item availability to "Unavailable", then the customer app displays it as out-of-stock and prevents ordering.

US-MENU-03 (Must) Toggle Item Availability Quickly

- As a Cashier/Admin, I want to toggle item availability so that customers cannot order out-of-stock items during peak hours.
- **Acceptance Criteria**
    - Given an item is Available, when I toggle it to Unavailable, then the change is saved and reflected on the customer app.
    - Given a customer has the item in cart, when the item becomes Unavailable before checkout submission, then checkout is blocked and the customer is informed which item is unavailable.

US-MENU-04 (Should) Edit Menu Item

- As an Admin, I want to edit an item's price/description so that the menu stays accurate.
- **Acceptance Criteria**
    - Given an item exists, when I update price and save, then the new price is shown to customers for future orders.

    - Given an order already exists, item price changes do not change historical order totals (price snapshot rule).

Epic 2 — Customer Ordering (Mobile App)

US-APP-01 (Must) Browse Menu by Category

- As a Customer, I want to browse items by category so I can find what I want quickly.
- **Acceptance Criteria**
    - Given the menu exists, when I open the app, then categories are displayed and selectable.
    - When I select a category, then only items from that category are displayed.
    - Given a category has no items, then an empty state is shown (no crash).

US-APP-02 (Should) Search Menu Items

- As a Customer, I want to search items by keyword so I can find items faster.
- **Acceptance Criteria**
    - Given I type a keyword, when I search, then matching items are displayed.
    - Given no items match, then a “No results” state is shown.

US-APP-03 (Must) Add Items to Cart

- As a Customer, I want to add items to cart so I can build my order.
- **Acceptance Criteria**
    - Given an item is Available, when I tap Add, then it is added to cart and cart total updates.
    - Given an item is Unavailable, when I tap Add, then the system blocks adding and shows “Out of stock”.

US-APP-04 (Must) Update Cart Quantity

- As a Customer, I want to change quantities so I can adjust my order.
- **Acceptance Criteria**
    - When I increase/decrease quantity, then subtotal and total update immediately.
    - Given I reduce quantity to zero, then the item is removed from cart.

US-APP-05 (Must) Select Pickup Time Slot

- As a Customer, I want to select a pickup time slot so I know when to pick up my order.
- **Acceptance Criteria**
  - Given available slots exist, when I open pickup selection, then I see a list of available slots (not in the past).
  - When I select a slot, then it is displayed in checkout summary.
  - Given a slot is full/unavailable, when I tap it, then I cannot select it and I see an explanation.

US-APP-06 (Must) Place Pickup Order (Pay at Counter)

- As a Customer, I want to place my pickup order so the restaurant can prepare it.
- **Acceptance Criteria**
  - Given cart is not empty AND pickup time is selected AND phone number is provided, when I confirm order, then an order is created with an Order ID and status = Placed.
  - Then I see an order confirmation screen showing: Order ID, pickup time, items, total, and "Pay at counter".
  - Given any cart item becomes Unavailable at submission time, when I confirm, then the order is NOT created and the system shows which items must be removed/changed.
  - Given the selected pickup slot becomes unavailable at submission time, then submission is blocked and customer must re-select a slot.

US-APP-07 (Must) Track Order Status

- As a Customer, I want to track order status so I know when to come pick it up.
- **Acceptance Criteria**
  - Given I have an active order, when I open tracking, then I see current status and timestamps (if available).
  - Given restaurant updates status to Ready, when I refresh/open tracking, then status shows Ready.
  - Given order is Cancelled, then the tracking screen shows Cancelled with reason (if provided by restaurant).

US-APP-08 (Must) Cancel Before Confirmation

- As a Customer, I want to cancel my order before the restaurant confirms it.
- **Acceptance Criteria**
    - Given status = Placed, when I tap Cancel and confirm, then order becomes Cancelled and disappears from active queue.
    - Given status ≠ Placed (Confirmed/Preparing/Ready/Completed), then Cancel action is not available and the app explains cancellation is no longer possible.

US-APP-09 (Must) View Item Details

- As a customer, I want to view item details so I can decide before adding to cart.
- AC:
    - Item detail displays name, price, availability, and description/image (if provided).

US-APP-10 (Should) View Order History

- As a customer, I want to view past orders so I can reference previous purchases.
- AC:
    - History lists completed/cancelled orders with date, total, and status.

Epic 3 — Dashboard Order Operations

US-DASH-01 (Must) View New Orders Queue

- As a Cashier, I want to see new orders in a queue so I can process them quickly.
- **Acceptance Criteria**
    - Given a customer placed an order, when I open the dashboard queue, then the order appears under Placed.
    - Each order row shows: Order ID, pickup time, item count, customer phone, created time.
    - New orders are visually distinct (e.g., highlight) until opened/confirmed (implementation detail acceptable).

US-DASH-02 (Must) Confirm Order

- As a Cashier, I want to confirm an order so the kitchen can start preparing.
- **Acceptance Criteria**

  - Given status = Placed, when I click Confirm, then status becomes Confirmed and confirmed timestamp is recorded.
  - Given two staff attempt to confirm simultaneously, then only one confirmation succeeds and the other sees a "status already updated" message.

US-DASH-03 (Must) Reject Order with Reason

- As a Cashier, I want to reject an order with a reason so the customer understands.
- **Acceptance Criteria**
  - Given status = Placed, when I click Reject, then I must select/enter a reason.
  - When I submit rejection, then status becomes Cancelled, cancelled timestamp is recorded, and cancel reason is stored.
  - Customer sees Cancelled + reason in tracking.

US-DASH-04 (Must) Update Order Status (Preparing/Ready/Completed)

- As a Cashier, I want to update statuses so customers receive accurate pickup signals.
- **Acceptance Criteria**
  - Given status = Confirmed, when I set Preparing, then status becomes Preparing and timestamp is recorded.
  - Given status = Preparing, when I set Ready, then status becomes Ready and ready timestamp is recorded.
  - Given status = Ready, when I set Completed, then status becomes Completed and completed timestamp is recorded.
  - Invalid transitions are blocked (e.g., Ready → Preparing).

US-DASH-05 (Must) Audit Trail for Status Changes

- As an Owner/Admin, I want to know who changed an order status for accountability.
- **Acceptance Criteria**
  - Each status change records: fromStatus, toStatus, changedBy, changedAt.
  - Audit log is viewable in order detail (minimal list).

US-DASH-06 (Must) Staff Login

- As a staff member, I want to log in to the dashboard so that only authorized users can access orders and customer data.

- AC:
    - Given valid credentials, when I log in, then I can access the dashboard.
    - Given invalid credentials, when I log in, then I see an error and cannot access the dashboard.

US-DASH-07 (Must) Role-Based Access Control

- As an Owner/Admin, I want role-based access so that permissions are controlled.
- AC:
    - Admin can access Menu + Orders + Reporting + Settings.
    - Cashier can access Orders; menu editing is limited to availability toggle (if configured).

US-DASH-08 (Must) View Order Details

- As a cashier, I want to view full order details before confirming so I can avoid mistakes.
- AC:
    - Order detail shows items, qty, notes, pickup time, customer phone, timestamps.
    - Actions shown depend on current status.

US-DASH-09 (Should) Configure Pickup Slot Interval

- As an Admin, I want to configure pickup slot interval so that pickup time selection matches kitchen capacity.
- AC:
    - Admin can set slot interval (e.g., 10 minutes).
    - Customer app shows slots generated accordingly.

Epic 4 — Reporting (MVP)

US-REP-01 (Should) Daily Summary

- As an Owner, I want to view daily performance so I can evaluate results.
- **Acceptance Criteria**
    - Given a selected date, dashboard shows total orders, total revenue, orders by status.

- Dashboard shows average processing time (Confirmed → Ready).

---

14) Non-Functional Requirements (Measurable)

- **Performance (Dashboard):** order queue refresh ≤ 10 seconds (MVP acceptable); target ≤ 5 seconds later
- **Usability (Cashier):** confirm an order in ≤ 2 clicks from queue
- **Availability:** ≥ 99% during business hours (portfolio target)
- **Security:** RBAC; protect customer phone number; no unauthenticated access
- **Audit:** record status transitions with operator + timestamp
- **Data Privacy:** store minimum necessary customer data; restrict access to customer phone to authorized staff roles

---

15) Analytics & Reporting (MVP)

- Daily orders count
- Daily revenue (sum of totals)
- Orders by status
- Average processing time (Confirmed→Ready)
- Peak time distribution by pickup slot (optional)

---

16) Edge Cases (Must-Cover Scenarios)

- Item becomes unavailable during checkout → block submit; show impacted items
- Pickup slot becomes unavailable → require re-selection
- Restaurant rejects due to out-of-stock → customer sees Cancelled + reason
- Customer cancels before confirm → dashboard updates accordingly
- Duplicate clicks on confirm/status update → system prevents double transitions
- Network delay: customer sees stale status → refresh mechanism + last updated timestamp

---

17) UAT Overview (Portfolio-ready)

17.1 UAT Scope

- Customer ordering + tracking
- Dashboard confirmation + status transitions
- Out-of-stock rejection
- Customer cancel before confirmation
- Peak-hour multi-order queue handling

17.2 UAT Entry / Exit

- Entry: test environment ready; menu configured; test users; UAT plan approved
- Exit: all Must stories passed; critical defects resolved; owner sign-off obtained

---

18) Traceability (Simple Mapping)

- Objective: reduce dependency → Epics 2 & 4
- Objective: standardize ops → Epic 4
- Objective: improve CX/status clarity → Epics 2 & 4
- Objective: build customer database → Epic 2 (phone + history)

---

19) Risks (Operational) & Mitigation

- Staff resistance → training + minimal UI clicks + pilot run
- Low customer adoption → QR onboarding + counter signage + staff prompts
- Peak-hour load → queue performance targets + simple status model + capacity slots (optional)

---

20) Glossary

- **MVP:** Minimum Viable Product
- **UAT:** User Acceptance Testing
- **AC:** Acceptance Criteria

- **RBAC:** Role-Based Access Control