Product Requirements Document (PRD): FastFood AI Ordering & Management Platform

Version: 1.0

Author: Omar Embarki

Date: 13-Jan-2026

Status: Draft / Planning

1. Executive Summary

FastFood AI is an all-in-one restaurant SaaS designed for the Algerian market. It bridges the gap between digital and physical ordering by combining QR code table service, waiter-assisted entries, and an AI-powered Telegram chatbot for deliveries. The platform includes a real-time driver tracking system and a centralized management dashboard to streamline operations and reduce errors.

2. Target Audience & Users

- **Dine-in Customers:** Speed-oriented users who want to order via QR without waiting for a waiter.
- **Delivery Customers:** Users on Telegram/WhatsApp who prefer chatting over calling.
- **Restaurant Staff:** Waiters needing a simple interface for manual orders and kitchen staff tracking progress.
- **Delivery Drivers:** Mobile users requiring GPS tracking and order status updates.
- **Restaurant Owners:** Decision-makers looking for analytics and centralized control.

3. Functional Requirements

3.1 Ordering Channels

- **QR Code Table Ordering:** Dynamic menu access via table-specific QR codes. Offline-first caching for unstable internet.
- **Waiter Manual Input:** A "Point of Sale" (POS) style interface for staff to enter walk-in orders.
- **AI Chatbot (Telegram/Gemini):** NLP-based ordering. Extracts items, quantities, and addresses from natural language.
- **Phone Orders:** Dashboard interface to manually log telephonic orders and assign them

to the delivery flow.

3.2 Delivery & Logistics

- **Driver Management:** Status tracking (Active/Inactive), assignment logs, and performance metrics.
- **Real-time Tracking:** Map-based view for owners and status updates for customers (Picked up -> En Route -> Delivered).
- **Driver App:** Lightweight React Native app for order management and location pings.

3.3 Centralized Dashboard

- **Live Order Feed:** Real-time updates via WebSockets/Socket.io.
- **State Machine Management:** Ability to move orders through states (Pending, Preparing, Ready, En Route, Completed).
- **Reporting:** CSV/Excel export for accounting.

4. Technical Stack

- **Backend:** Node.js, TypeScript, Express.
- **Database:** PostgreSQL (Relational data for orders/menu) or MongoDB.
- **Frontend (Web):** React.js + Tailwind CSS (Dashboard & QR Menu).
- **Mobile:** React Native (Driver App).
- **AI:** LangChain + Google Gemini API for parsing unstructured text.

5. Success Metrics

- **Accuracy:** 90%+ parsing accuracy for the AI chatbot.
- **Efficiency:** 70% reduction in manual data entry errors.
- **Growth:** 50+ daily orders per active restaurant within 3 months.