AI Calling Platform – Complete Build Plan & Workflow

This document provides a complete, end-to-end execution plan for building an India-first AI Calling Platform from scratch. It covers architecture, workflow, development phases, and operational strategy. The goal is to build a single core engine that supports multiple bots (appointments, payments, reminders, leads) using configurable rules and scheduling, not separate systems.

1. What We Are Building (Product Clarity)

We are building a managed AI calling system that answers inbound calls and makes outbound calls on behalf of businesses. The system includes the AI bot, call management, scheduling, retry logic, and reporting as one integrated service. Customers do not host anything; all infrastructure is managed centrally.

- Inbound calls: customers call a business number, AI answers.
- Outbound calls: system calls customers based on rules (reminders, follow-ups, payments).
- Single engine, multiple bot templates via configuration.

2. High-Level Architecture

The platform is built as a layered system. Voice and AI are plug-ins; the core value lies in orchestration and rules.

- Call Orchestration Core – tracks call lifecycle.
- Scheduler – decides when calls should happen.
- Rules Engine – controls retries, stop conditions, outcomes.
- AI Conversation Layer – handles speech and intent during calls.
- Outputs & Reporting – what customers see.

3. Core Concepts

Bots are not separate systems. A bot is defined as a combination of triggers, rules, and outcomes. The same engine executes all bots with different configurations.

- Trigger: when a call should happen (event or time-based).
- Retry Policy: how many times and when to retry.
- Conversation Goal: what the AI tries to achieve in one call.
- Stop Conditions: when the system should stop calling.

4. Development Roadmap

Week 1 – Call Orchestration Core

- Set up Node.js backend and Postgres database.

- Create core tables: organizations, users, calls.
- Inbound webhook endpoint to log incoming calls.
- Outbound call API to create call tasks.
- Basic scheduler to trigger pending calls.

Week 2 – Scheduler & Rules Engine

- Implement retry logic and next-action scheduling.
- Define rule configs per bot type.
- Handle provider events (answered, completed, no-answer).
- Ensure call lifecycle stability.

Week 3 – AI Conversation Layer

- Integrate speech-to-text, LLM, and text-to-speech.
- Constrain AI to predefined outcomes.
- Map AI outcomes to rule engine actions.

Week 4 – Outputs & Customer Experience

- Call logs and outcomes view.
- Daily/weekly summaries via WhatsApp or email.
- Escalation alerts for human intervention.

5. Appointment Bot – End-to-End Workflow

The appointment bot uses fixed availability data provided by the clinic. The AI does not calculate availability; it reads predefined slots.

- Booking call – offer available slots.
- Confirmation immediately after booking.
- Reminder call (previous day or morning).
- Final confirmation call 1 hour before appointment.
- No-show handling and slot recovery via waitlist.

6. Payment & Reminder Bots

Payment and reminder bots reuse the same engine with different triggers and retry policies.

- Trigger based on due date or event.
- Multiple retries with delays.
- Stop on payment confirmation or promise.

7. Custom Bot Strategy

Users are not allowed to build bots from scratch. Custom bots are handled as managed requests.

- User submits custom request form.
- Team evaluates if it fits existing templates.
- If needed, create a private configuration.
- Charge premium for custom logic.

8. Key Principles to Follow

- Build one engine, reuse everywhere.
- Sell outcomes, not technology.
- Automate only after manual success.
- Keep users away from core logic.

This plan is designed to get you from zero to a working, revenue-ready AI calling platform without over-engineering. Follow the order strictly.