Documentation

Problem Statement:

Build a Conversational AI System for Handling Dynamic Banking Interactions like Loan Applications, Card Blocking, and Account Queries with Multi-Turn Understanding and Real-Time Task Execution.

Quick Start:

Prereqs

- Node.js **18+**
- (Backend) Gemini **API key** from Google AI Studio

How It's Built (30-sec overview):

- **Frontend (Next.js + TS)**: Chat UI, suggested reply chips, **Debug drawer** (shows `memory` + last 20 request/response logs).
- **Backend (Fastify + TS)**: Single endpoint **POST `/chat`** returns `{ conversationId, response, memory}`.
- **LLM (Gemini)**:
  - **Extraction:** Intent + slots → **strict JSON** (validated by **Zod**).
  - **Reply:** Short, safe message from tool results.
- **Policy & Tools:** Policy controls **ASK / CONFIRM / CALL_TOOL / FALLBACK**. Tools are mocked: `blockCard`, `getTransactions`, `checkLoanPreEligibility`.
- **Memory:** Per-conversation slot store (in-process Map; replaceable with Redis + TTL).

Folder Map:

ai-banking-frontend/ # Next.js chat UI

ai-banking-backend/ # Fastify + Gemini + policy + tools

docs/ # Diagrams & notes (optional)

Demo Script (2 minutes):

1. Open UI → send **"Block my card ending 1234"** → click **Yes** → show reference ID.
2. Mini Statement → **SB-001** → **5** → show list.
3. Open **Debug drawer** → point out changing `memory` (cardLast4, confirmed, accountId, limit).
4. Mention guardrails: confirmations, allow-lists, redaction, Zod.