Inya.ai Buildathon 2025 by Gnani.ai

Title: Inbound Banking Support Agent

1) Executive summary

Your mission is to design an AI agent that handles real-world inbound banking scenarios across Voice and Chat. Call the onbound phone numbers of Bank of Baroda, Union Bank of India, and Central Bank of India. Start by researching each bank's IVR flows to map common intents and information needs. Implement an agent that can resolve those intents using mock data or safe public datasets. Do not use real customer data at any point.

2) Problem statement

Inbound banking teams face heavy call volumes, repeated questions, strict compliance, identity checks, and many status lookups. Customers expect quick answers and a graceful path to a human when needed.

The core challenge is to detect intent with high precision, capture the right entities, provide a correct and helpful answer, recover from errors, and escalate to a live agent when appropriate. Your agent must behave consistently for both Voice and Chat.

Constraints apply. Do not connect to production systems. Use mock data or public datasets. Protect privacy at all times. Log interactions in a structured way so judges can reproduce results.

3) Solution scope and required scenarios

Support all the scenarios that appear on bank IVRs and websites:

1. Account balance and recent transactions
2. Card hotlist or replacement
3. Dispute or chargeback initiation
4. Complaint registry and ticket status
5. Branch locator
6. ATM locator
7. KYC or document status
8. Cheque status
9. Fixed deposit information
10. Loan status and payment options

For every scenario include intent detection, entity capture, confirmation, correct response, error recovery, and optional escalation through a warm transfer to a live agent via a stub endpoint.

4) Functional expectations

Intents and entities

Canonical intents: account_info, tx_history, card_block, raise_dispute, complaint_new, complaint_status, locate_branch, locate_atm, kyc_status, cheque_status, fd_rate_info, loan_status, speak_to_agent.
Canonical entities: account_number, last4, transaction_date, amount, branch_city, pincode, ifsc, ticket_id, language_pref, channel.

Teams can add more intents and entities if justified by research.

Dialogue behavior

Use a supportive, calm, solution oriented tone. Use plain language. Ask only for missing entities and confirm critical details such as account identifiers or action confirmations like card blocking. Handle interruptions, repeats, silence, and accents. Offer English and Hindi and allow mid conversation language switch without losing context.

Human escalation

Trigger escalation on repeated ASR or NLU failure, explicit user request, or flows with higher risk. Provide a compact handoff summary with user identity info from mock data, detected intent, captured entities, last bot message, and current disposition.

5) Data and APIs

Data policy

Only mock data or public datasets. No real PII.

Acceptable sources and approaches

Use bank websites or open government portals for branch and ATM lists. For locator features you may use a simple place search API or a static CSV with branch_name, address, city, pincode, ifsc, lat, lon. Use published IFSC lists for basic metadata. For balances, transactions, complaints, and tickets use a local mock service or JSON files that simulate realistic responses and error cases.

Required artifacts

Provide a data schema for each scenario, an API wrapper or stub with input and output shapes, and fallback behavior on timeout or error.

Example schema snippets

```
// Request to mock balance service
{ "account_number": "123456789012", "last4": "9012" }

// Response from mock balance service
{ "account_number": "******9012", "balance": 15430.55, "currency": "INR", "as_of": "2025-09-12T10:00:00+05:30" }

// Branch record
{ "branch_name": "BOB Fort", "city": "Mumbai", "pincode": "400001", "ifsc": "BARB0FORTXX", "lat": 18.931, "lon": 72.833 }
```

6) Best practices

Design NLU with varied utterances and synonyms. Cover code mixing for English and Hindi. Keep the entity set minimal and well documented. Use explicit confirmation for sensitive actions such as card blocking or dispute creation. For error handling, clarify and rephrase once, then route to a human after a finite number of failures. Add accessibility features such as slower speech, repeat last answer, and optional SMS or chat transcript. Instrument structured logs that record intent, entities, latency, API outcome, errors, and handoff reason.

7) Edge cases to test

Ambiguous requests like “my card issue” without details. Number ASR confusion for account and pincode. Locator results with many branches in the same city that require disambiguation. Dispute raised for a transaction that does not exist in the mock ledger. API timeout or 500 with a graceful retry and fallback. Language switch mid turn. Escalation when the user asks for a human or expresses frustration.

8) Testing checklist

Provide at least 20 scripts that cover happy paths and edge cases. Keep them executable and reproducible.

#	User utterance	Expected intent	Required entities	Mock API	Expected response	Escalation
1	What is my balance	account_info	account_number	balance	Speak balance with as_of	No
2	Last five transactions	tx_history	account_number	tx	List 5 with dates	No
3	Block my card	card_block	last4	card_block	Confirm action then success	No
4	Block card ending 9012	card_block	last4=9012	card_block	Success message	No

5	Raise a dispute for 1200 on 5 Aug	raise_dispute	amount, transaction_date	dispute	Ticket id returned	No
6	Status of my complaint C123	complaint_status	ticket_id	ticket	Current status	No
7	I want to complain	complaint_new	none	ticket_new	Ask details, create ticket	No
8	Nearest branch in Andheri	locate_branch	branch_city	branch	Top 3 with distance	No
9	ATM near pincode 560001	locate_atm	pincode	atm	Top 3 with address	No
10	KYC status	kyc_status	account_number	kyc	Status and next steps	No
11	Cheque status for 345678	cheque_status	cheque_number	cheque	Cleared or pending	No
12	FD rates	fd_rate_info	none	fd	Current rates table	No
13	Loan status for home loan 7788	loan_status	loan_id	loan	EMI due and date	No
14	Speak to agent	speak_to_agent	none	none	Warm transfer summary	Yes
15	My card problem	raise_dispute	none	dispute	Clarify missing fields	No
16	Balance please in Hindi	account_info	account_number, language_pref=hi	balance	Reply in Hindi	No
17	Branch in Fort Mumbai two options	locate_branch	branch_city	branch	Disambiguate choice	No
18	Dispute for txn not found	raise_dispute	amount, date	dispute	Explain not found and next steps	No
19	API timeout on balance	account_info	account_number	balance	Retry then fallback message	No
20	You are useless agent now	speak_to_agent	none	none	De escalate and transfer	Yes

9) Deliverables and submission format

Submit one PDF named exactly **Team Name_Inya_Final**. The PDF must include the executive summary, an architecture diagram, data schemas, links to all YouTube demo videos for each scenario, setup and run instructions, API stubs or mock server documentation, and a test results summary with pass or fail counts. Provide a code repository link with a clear README and environment steps. Include a small architecture diagram that shows NLU, dialog manager, API layer, mock datastore, and the escalation module.

10) Timeline and checkpoints

Masterclass and use case deep dive on **13 September 2025, 15:00 IST**.
Open Q and A on **19 September 2025, 15:00 IST**.
Final submission due **30 September 2025, 23:59 IST**.

11) Compliance and ethics

Do not use real customer data. Do not call live bank agents during demos. Redact any accidental PII in logs and videos. Disclose all datasets and API sources in the README. Keep all credentials for any third party APIs out of the repo.