

API Reference & Developer Guide

Technical Guide for Developers Integrating CohrenzAI

CohrenzAI

Noida, Uttar Pradesh, India

info@cohrenzai.com | +91 8273597975

Overview

CohrenzAI provides a REST API that powers all chatbot interactions. While most users integrate CohrenzAI using our one-script widget, developers who want deeper integration can use our API directly. This guide covers all available endpoints, authentication, request/response formats, and advanced customization options.

Authentication

All API requests require authentication using two credentials:

- API URL: Your unique API endpoint provided during onboarding
- Public Key: Your unique public key included in the X-Public-Key header

Both credentials are provided when you sign up for a CohrenzAI account. Keep your credentials secure and never expose them in client-side code that is publicly accessible. For the chatbot widget, the public key is embedded in the script tag data attributes.

Base URL

Your API base URL is provided during onboarding. All endpoints are relative to this base URL. Example: `https://api.cohrenzai.com/v1`

API Endpoints

POST /chat - Send a Chat Message

Send a visitor message to the chatbot and receive an AI-powered response.

Request Headers:

- Content-Type: application/json
- X-Public-Key: your_public_key

Request Body (JSON):

- message (string, required): The visitor's message text
- session_id (string, required): Unique session identifier for conversation tracking

Response Body (JSON):

- answer (string): The AI-generated response from the chatbot

Status Codes:

- 200: Successful response

- 400: Bad request (empty message)
- 500: Server error

Example: Send a POST request to /chat with the body containing message set to 'What are your pricing plans?' and session_id set to 'session_12345'. The response will contain an answer field with the chatbot's response about pricing plans based on your knowledge base.

GET /chats/{session_id} - Retrieve Chat History

Retrieve the conversation history for a specific session.

Path Parameters:

- session_id (string, required): The session ID to retrieve chats for

Query Parameters:

- k (integer, optional, default: 10): Number of recent messages to retrieve

Response Body (JSON):

- chats (array): List of chat objects, each containing 'message' (string) and 'sender' (string, either 'user' or 'ai')

Status Codes:

- 200: Successful response with chat history
- 500: Server error

GET /predict_intent/{session_id} - Get Intent Analysis

Get an AI-generated intent analysis for a specific conversation session. This endpoint analyzes all user messages in the session and produces a structured report about the visitor's interests, requirements, and behavioral signals.

Path Parameters:

- session_id (string, required): The session ID to analyze

Response Body (JSON):

- intent_summary (string): Structured intent analysis report covering user interest, segments of interest, actual requirements, unmet needs, matched offerings, and behavioral signals

Status Codes:

- 200: Successful response with intent analysis
- 500: Server error

Session Management

Sessions are the foundation of conversation tracking in CohrenzAI. Here is how they work:

- Each conversation is identified by a unique session_id
- The chatbot widget generates session IDs automatically in the format: session_{timestamp}_{random_string}
- Session IDs are stored in the browser's localStorage for continuity across page navigations
- If you are building a custom integration, you can generate your own session IDs using any unique string format
- To start a new conversation, generate a new session_id
- All messages sent with the same session_id are treated as part of the same conversation
- Conversation history is used for context in AI responses (last 20 messages)

Lead Capture Flow

Understanding the lead capture flow is important for developers building custom integrations. The flow works as follows:

Lead Detection Triggers

The system detects lead opportunities through three triggers:

- 1. Opportunistic: Visitor shares email or phone number during conversation
- 2. Intent Signal: Visitor mentions keywords like pricing, demo, trial, contact, consultation, services, or partnership
- 3. Engagement: After 4 or more user messages in a session

Lead Capture State Machine

Once triggered, the chatbot follows a state machine to collect information:

- NONE: Normal conversation mode, no lead capture active
- ASKED_NAME: Chatbot has asked for the visitor's name. Waiting for name input.
- ASKED_EMAIL: Chatbot has asked for email. Validates email format before accepting.
- ASKED_PHONE: Chatbot has asked for phone. Validates phone format before accepting.
- COMPLETED: All information collected. Lead is exported to Google Sheets.

The system is smart about data collection. If a visitor provides their email or phone early in the conversation before the lead flow starts, the system remembers it and skips asking for that information later. This prevents redundant questions and creates a smoother experience.

Input Validation

The system validates all lead inputs:

- Names: Detected from natural phrases like 'My name is Rahul' or 'I am Priya' or just the name by itself
- Emails: Validated using standard email format patterns
- Phone Numbers: Validated for Indian phone number format (10 digits with optional country code)
- Casual messages like 'hi', 'ok', 'thanks' are recognized and not mistaken for data input

Building a Custom Chat Widget

If you want to build your own chat interface instead of using our default widget, here is what you need:

- Generate unique session IDs for each conversation
- Send POST requests to /chat with the message and session_id
- Display the 'answer' field from the response to the user
- Handle the conversation UI, typing indicators, and message history in your frontend

- Optionally use GET /chats/{session_id} to restore conversation history if the user returns
- Optionally use GET /predict_intent/{session_id} for analytics

The API handles all AI processing, lead capture, and intent analysis on the server side, so your custom widget only needs to handle the user interface.

Webhook Integration (Premium)

Premium plan users can configure webhooks to receive real-time notifications when events occur:

- New Lead Captured: Triggered when a visitor completes the lead capture flow with name, email, and phone
- High-Intent Conversation: Triggered when intent analysis detects a high-value prospect
- Custom Events: Configure triggers based on specific keywords or conversation patterns

Webhooks send HTTP POST requests to your specified endpoint with JSON payloads containing event details.

Rate Limiting

To ensure platform stability and fair usage, API requests are subject to rate limits:

- Free Plan: 100 requests per hour
- Standard Plan: 1,000 requests per hour
- Premium Plan: 10,000 requests per hour (custom limits available)

If you exceed the rate limit, the API will return a 429 status code. Implement exponential backoff in your integration to handle rate limiting gracefully.

Error Handling

The API returns standard HTTP status codes:

- 200: Success - Request processed successfully
- 400: Bad Request - Invalid or missing parameters (e.g., empty message)
- 401: Unauthorized - Invalid or missing authentication credentials
- 429: Too Many Requests - Rate limit exceeded
- 500: Internal Server Error - An unexpected error occurred on our side

Error responses include a 'detail' field with a human-readable error message. For production integrations, always handle error responses gracefully and provide fallback behavior for your users.

Best Practices for Developers

- Always generate unique session IDs to prevent conversation mixing
- Implement error handling and retry logic for API calls
- Use HTTPS for all API communications
- Do not expose your API credentials in publicly accessible code

- Implement a typing indicator in your UI while waiting for API responses
- Cache conversation history client-side to reduce API calls
- Use the intent analysis endpoint to build custom analytics dashboards

Technical Support for Developers

Need help with your integration? Our technical team is available to assist:

- Email: info@cohrenzai.com
- Phone: +91 8273597975 / +91 8770278713
- Include your account ID and session IDs when reporting issues for faster resolution