

WhatsApp + Groq AI Multi-Client Platform

Full architecture, workflow, database design, and step-by-step implementation plan.

Platform owner

Super Admin: You, the developer who sells and manages the platform.

Client users

Each company receives its own private dashboard, staff accounts, WhatsApp setup, and AI configuration.

Scope: Welcome message, location, brochure, Groq auto-replies, per-chat AI control, human handover, secure login, and full client isolation.

1. Platform roles and ownership

You own the complete platform and create separate accounts for every business that buys it.

Role	What they can do
Super Admin (you)	Create, edit, suspend, and support all client companies. Create client admins, set plans and expiry, manage errors and global usage.
Client Admin	Manage only their company: chats, team members, WhatsApp settings, brochure, welcome/location content, and Groq settings if you permit it.
Agent	View and reply only to their company chats. AI toggle and human handover can be allowed by permission.

Account hierarchy

Super Admin (you) → Client A: Client Admin → Agents A1/A2; Client B: Client Admin → Agent B1; Client C: Client Admin.

Important access rule

Every API request must derive the client from the logged-in user. Never accept a client_id from the browser as proof of access. The backend verifies it for every query and action.

2. End-to-end message flow

Incoming WhatsApp messages must be routed to the correct company before any data is read or sent.

Customer sends WhatsApp message

↓

Meta WhatsApp webhook reaches your server

↓

Read phone_number_id and find the matching whatsapp_configs record

↓

Obtain client_id; save the incoming message under that client

↓

Find or create the lead with (client_id, phone)

↓

First message? Send welcome → location → brochure

↓

AI enabled for this lead and human takeover OFF? Call Groq

↓

Validate, send, and save the Groq reply; otherwise show chat to staff only.

First-contact rules

- New lead defaults to AI Auto Reply ON unless the client chooses otherwise.
- Send welcome message, WhatsApp location pin, and brochure only once per client-contact pair.
- Set each sent flag only after WhatsApp confirms the message was accepted.
- Use webhook/message IDs and events to prevent duplicate sends when Meta retries a webhook.
- After the welcome flow, Groq may answer the customer's actual first message if AI is allowed.

WhatsApp rule

Free-form WhatsApp replies are normally allowed during the 24-hour customer-service window after a customer message. Outside that window, use Meta-approved message templates.

3. Database structure

Each company must be isolated using client id. Existing Someshwar data becomes the first client.

Relationship map

```
clients
  ■■■ users
  ■■■ whatsapp_configs
  ■■■ ai_settings
  ■■■ whatsapp_leads
  ■ ■■■ whatsapp_messages
  ■ ■■■ conversation_events
  ■■■ uploaded brochures/media
```

Table	Purpose / key data
clients	Company name, slug, plan, active/suspended status, expiry.
users	Login users: super_admin, client_admin, agent. Password hash and client_id.
whatsapp_configs	One config per client: WhatsApp number, phone_number_id, encrypted access token, welcome, location, brochure.
ai_settings	Per-client encrypted Groq key, model, system prompt, business information, FAQ, language and handover keywords.
whatsapp_leads	Per-contact data, first-flow flags, AI toggle, human takeover state, unread count.
whatsapp_messages	All customer, AI, human, and system messages; includes client_id, lead_id and sender type.
conversation_events	Audit record for AI changes, handover, first-contact sends and failures.

Required uniqueness

UNIQUE(phone_number_id) in whatsapp_configs. UNIQUE(client_id, phone) in whatsapp_leads. This lets the same customer number exist safely for different companies.

4. Per-chat AI and human handover

AI control should be made separately for every contact, not just globally for the client.

Chat header example

Rahul Patel | Status: AI handling
[AI Auto Reply: ON] [Take Over Chat]

After staff takes over:
Rahul Patel | Status: Human handling
[AI Auto Reply: OFF] [Resume AI]

Lead field	Meaning
ai_auto_reply_enabled	Whether Groq may reply to this lead.
human_handover	Whether a staff member has taken ownership of the chat.
human_handover_at / by	When and which staff member started takeover.
ai_paused_reason	staff, customer_stop, ai_error, or other safe reason.

Rules

- Take Over Chat: set human_handover = true and ai_auto_reply_enabled = false immediately.
- Resume AI: clear the takeover fields and enable AI again.
- A manual staff reply should normally start human takeover automatically, avoiding two replies at once.
- Keywords such as agent, human, person, staff, call me, or owner should pause AI and notify staff.
- STOP, unsubscribe, no bot, or similar opt-out requests must pause AI until staff intentionally resumes it.
- If Groq fails, times out, or returns an unusable reply, do not send an invented fallback. Flag the chat for staff.

5. Groq AI design

Groq is used as the model provider. Keep its model name configurable so the platform can move to a different available model later without a code rewrite.

Groq request content

- The current client's system prompt and business identity.
- Client-specific FAQ and business information.
- The last 10-20 messages for this lead only.
- The new customer message.
- Reply language, maximum length, refusal rules, and human-handover instructions.

Call Groq only when all checks pass

Client is active → client AI is enabled → lead AI toggle is ON → human handover is OFF → customer has not opted out → message is not duplicate.

Outgoing AI message record

Save every successful AI reply in `whatsapp_messages` with `direction = outgoing`, `sender_type = ai`, `client_id = current client`, and `lead_id = current lead`. Then send it through that same client's WhatsApp Cloud API account.

Security

Encrypt Groq and WhatsApp tokens before database storage. Decrypt only on the backend while making the provider request. Configuration APIs must return only a masked value or "configured", never the actual secret.

6. Implementation order

Build in this order so that chat data is secure before the platform is sold to multiple businesses.

1. Backup and baseline

Back up PostgreSQL, commit current code, create secure environment variables for session secret and encryption key.

2. Database migration

Create clients, users, ai_settings, conversation_events; add client_id and AI/handover fields to current tables; assign existing records to Someshwar AI.

3. Authentication

Add login, logout, password hashing with bcrypt, secure HTTP-only sessions, roles, and route protection.

4. Super Admin dashboard

Add client list, create/edit/suspend client, create Client Admin, reset password, plans and expiry.

5. Tenant isolation

Scope every database query and upload by client_id. Check client ownership in backend middleware.

6. Multi-client WhatsApp routing

Resolve client by incoming phone_number_id. Pass the resolved client configuration to every WhatsApp send/download operation.

7. First-contact service

Move welcome, location, brochure logic into an idempotent service with separate success flags and event records.

8. Groq service

Load only current client settings, use limited lead history, validate output, send/save AI response, report errors.

9. Chat controls and UI

Add AI toggle, takeover, resume AI, status label, staff/team management, and protected configuration pages.

10. Test and deploy

Test two clients, webhook retries, cross-client access, AI/handover rules, then deploy with HTTPS, backups, logs, and rate limits.

7. API and testing checklist

Core API groups

```
POST /api/auth/login | POST /api/auth/logout | GET /api/auth/me
GET/POST/PUT /api/clients | GET/POST/PUT /api/users
GET/PUT /api/whatsapp-config | GET/PUT /api/ai-settings
GET /api/messages | POST /api/messages/send | POST /api/messages/send-file
PUT /api/leads/:id/ai-auto-reply | POST /api/leads/:id/takeover | POST /api/leads/:id/resume-ai
GET/POST /webhook/whatsapp
```

Must-pass tests before selling

- Client A cannot see, query, upload to, or change any Client B chat or setting.
- Each incoming WhatsApp number uses the correct client token, brochure, welcome flow, and Groq configuration.
- A new lead receives welcome, location, brochure, and then (when enabled) one AI reply.
- A repeat lead never receives the first-contact sequence again.
- AI ON replies; AI OFF does not; takeover stops instantly; resume re-enables it.
- Human-request keywords and STOP commands pause AI correctly.
- Duplicate Meta webhook deliveries never create duplicate incoming/outgoing messages.
- A failed Groq or WhatsApp call creates a useful error event without affecting another client.
- Suspended clients cannot log in or send AI replies.

Production checklist

- HTTPS, secure HTTP-only cookies, rate limiting, and Meta webhook signature verification.
- Encrypted API credentials, database backups, audit logs, and error monitoring.
- Per-plan limits for agents, chats, AI replies, and storage, plus scheduled expiry/suspension checks.

Recommended build order: Database → Login → Client isolation → Multi-client WhatsApp routing → First-contact flow → Groq → Chat controls → Dashboards → Testing and security.