Project ZYBS™ – Multi-Model Voice Assistant (Spec v1)

> **Trademark: Project ZYBS™ by Carlos Hufano III — Scientist & Innovator (© 2025). © 2025. All rights reserved.**

1) Vision

A privacy-first, natural-voice AI that works **online or offline**, orchestrates multiple foundation models (Claude, ChatGPT, Gemini, local LLMs), and acts as a **day-by-day assistant** that can: compose emails, build & deploy websites, connect to servers, automate homes, integrate with wearables (smartwatches, smart glasses), manage social media bots, answer calls, perform real-world tasks, and place delivery orders.

---

2) MVP Scope (4–6 weeks)

Voice In/Out

- Streaming ASR: Whisper (server) + faster-whisper (edge). Offline fallback via whisper.cpp on desktop/mobile.
- TTS: Online (PlayHT/ElevenLabs/OpenAI TTS), offline fallback via Piper/Coqui.

Orchestration

- Model Router: routes tasks to Claude / ChatGPT / Gemini / Local LLM based on **capability** & **cost/latency** policy.
- Toolformer/Function-calling layer: Structured tool use with JSON schema.
- Short-term memory (conversation) + Long-term memory (user profile, preferences) via local encrypted vector DB (Qdrant/Chroma). Export/import allowed.

Core Skills (initial)

- Email assistant: Draft/reply, summarize threads (Gmail/Outlook OAuth).
- DevOps lite: SSH orchestration (read-only at first), GitHub actions trigger, deploy to Vercel/Netlify.
- Website starter: Generate Next.js or WordPress scaffold; one-click deploy to Vercel; DNS helper.
- Home automation: Integrate with Home Assistant / Matter via webhook + MQTT; safe-mode by default.
- Social media integration: Connect to Facebook Messenger bots, Instagram, Twitter/X, TikTok for messaging and post scheduling.
- Call handling: Integrate with VoIP/SIP or mobile APIs to answer, forward, and respond to calls.
- Task execution: Connect with delivery platforms (e.g., Grab, Foodpanda) to place and track orders.
- Daily Brief: calendar, tasks, reminders, email digest, weather, simple health summary if wearable is connected.

Apps & Wearables

- Companion app (Flutter): iOS/Android with background wake word, push-to-talk, and notification actions.
- Wearables: WearOS & watchOS complications; smart-glasses via phone bridge (camera captioning & voice notes).

Offline mode

- On-device small models for: ASR, intent classification, note summary, and simple Q&A (e.g., Llama-3 8B / Mistral-7B via llama.cpp). Auto-sync when back online.

Safety & Privacy (MVP)

- All credentials in encrypted vault (device Keystore/Secure Enclave + server KMS).
- Per-tool permissions & logs ("why this action").
- PH Data Privacy Act (RA 10173) alignment.

---

3) High-Level Architecture

```
[Mic/Camera] → ASR Stream   ┐
                        ┌───▼──────────────────────────┐
                        │ ZYBS Orchestration Gateway   │
                        │ - Intent & Task Planner      │
                        │ - Model Router               │
User Memory/Profile ────▶ - Tool/Skill Runner          │◀── App Integrations
(Gmail, Vercel, GitHub,
(Encrypted Vector DB)   │ - Policy & Safety Guard      │    Home Assistant,
Calendar, Drive,
                        └───▲───────────┬──────────────┘    Social Media APIs,
VoIP, Delivery APIs)
                            │           │
                   Online Models   Local Models (offline)
                 (Claude/ChatGPT/   (llama.cpp, faster-whisper,
                 Gemini Realtime)    Piper/Coqui)
                            │           │
                        ┌───▼───────────▼───┐
                        │ Response Synth     │ → TTS Stream → Speakers
                        └────────────────────┘
```

---

4) Project Proposal (SaaS)

Objective. Launch Project ZYBS™ as a **Software-as-a-Service** that delivers a natural-voice, multi-model AI assistant with online/offline modes, social bots, call handling, and delivery tasking.

In-Scope (Phase 1). Conversational core (ASR/TTS), model routing (Claude/ChatGPT/Gemini/local), email composer, website deployer (Vercel/WordPress), server connect (read-only SSH), Home Assistant scenes, **social bot connectors (Messenger/WhatsApp/IG/Telegram)**, **telephony inbound calls (IVR + voicemail)**, **ordering flow with approvals**, mobile companion app (Flutter), admin console, audit logs.

Out-of-Scope (Phase 1). Payments processing, deep CRM, outbound voice campaigns at scale, advanced analytics (moved to Phase 2), automatic order without explicit approval.

Success Metrics.

- Latency: < 600ms partial ASR, < 2.0s first TTS for online mode.
- Task completion rate: ≥ 85% for top-10 skills (email draft, deploy, server status, scene toggle, simple order).
- Call answer/containment: ≥ 70% without human transfer for FAQs.
- Cost per active user: within target per tier (see below) while meeting SLAs.
- NPS ≥ 40 after 30 days.

Deliverables. Mobile apps (iOS/Android), web console, orchestration gateway, skills microservices (email, deploy, SSH RO, HA scenes, bots, telephony, ordering), offline model pack, audit/consent system, documentation.

---

5) Business Blueprint (SaaS Model)

Tiers & Pricing (proposal).

- **Starter (Personal)** — ₱1,490/mo: Voice chat, email drafts, 1 social channel, 200 mins ASR/TTS, offline pack lite.
- **Pro (SMB)** — ₱4,990/mo: + 3 social channels, basic call answering (1 number), website deploy wizard, RPA ordering (allowlist vendors), 1,000 mins voice, unified inbox.
- **Enterprise** — Custom: SSO, on-prem gateway option, advanced audit, priority model routing, DPA/SCCs, dedicated numbers, SLAs (99.9%).

Ops & Compliance. PH RA 10173 alignment; opt-in/opt-out for messaging; call recording notices; encrypted vaults; export/delete on request; optional data residency.

Risks & Mitigations.

- Provider dependency → multi-model router + offline fallback.
- API rate limits → backoff, queuing, campaign pacing.
- Erroneous actions → explicit approvals (voice PIN/app confirm), budgets, allowlists, dry-run preview.

Next Steps. Confirm MVP feature set and tiers → provision cloud resources & OAuth apps → implement Week 1–6 plan → closed beta.

---

6) Workflows (Flowcharts)

6.1 Core Conversation & Tooling

```
flowchart TD
    U[User Speech/Touch] --> ASR[ASR Stream]
    ASR -->|text| Planner[Intent & Task Planner]
    Planner --> Router[Model Router]
    Router --> Models{Claude / ChatGPT / Gemini / Local}
    Models --> Planner
    Planner --> Tools[Tool/Skill Runner]
    Tools --> Integrations[Gmail • Vercel • GitHub • Home Assistant • Social APIs • VoIP • Delivery]
    Planner --> Safety[Policy & Safety Guard]
    Safety --> Approvals{Approval Needed?}
    Approvals -- Yes --> Confirm[Voice PIN / App Confirm]
    Confirm --> Execute[Execute Action]
    Approvals -- No --> Execute
    Execute --> Synth[TTS]
    Synth --> U
```

6.2 Inbound Call Answering

```
flowchart TD
    Caller --> IVR[Twilio/SIP IVR]
    IVR --> Transcribe[Realtime ASR]
    Transcribe --> CallAgent[Voice Agent]
    CallAgent --> Intent{Intent}
    Intent -- FAQ --> Answer
    Intent -- Schedule/Task --> Tools
    Tools --> Calendar/CRM
    Intent -- Human Needed --> Transfer[Warm Transfer]
    Answer --> TTS
    Tools --> TTS
    TTS --> Caller
    subgraph Compliance
    Notice["Recording Notice / Opt-out"]
    end
```

6.3 Delivery Order Flow

```
flowchart TD
    Request["User: Order X from Y"] --> Planner
    Planner --> PriceTime[Price & ETA Fetch]
    PriceTime --> Review[Summarize for Review]
    Review --> Approval{Approve within Budget?}
    Approval -- Yes --> Place[Place Order via API/RPA]
    Approval -- No --> Cancel
    Place --> Track[Track & Notify]
    Track --> Receipt[Send Receipt to Email/Sheets]
```

6.4 Mock Flowchart (All-in-One Overview)

```
flowchart LR
    Start([User says: "ZYBS, help"]) --> Detect[Wake word / Push-to-talk]
    Detect --> ASR[ASR]
    ASR --> NLU[Intent & Entities]
    NLU --> Choice{Task Type}
    Choice -->|Email| Draft[Draft Email]
    Choice -->|Website| Site[Scaffold & Deploy]
    Choice -->|Server| Srv[Check Server Status]
    Choice -->|Home| Home[Run Scene]
    Choice -->|Social| Social[Send Msg / Post]
    Choice -->|Order| Order[Delivery Quote]
    Draft --> Approve{Approval?}
    Site --> Approve
    Srv --> Approve
    Home --> Approve
    Social --> Approve
    Order --> Approve
    Approve -- Yes --> Exec[Execute Tool/API]
    Approve -- No --> Cancel((Cancel))
    Exec --> Notify[Notify & Log]
    Notify --> Memory[Update Memory]
    Memory --> TTS[TTS]
    TTS --> End([Done])
```

6.5 Mock Swimlanes (User • Orchestrator • Integrations)

```
flowchart TB
    subgraph User
      U1[Speak/Click]
      U2[Approve Action]
```

```
  end
  subgraph Orchestrator
    O1[ASR]
    O2[Planner]
    O3[Model Router]
    O4[Safety & Policy]
    O5[Tool Runner]
    O6[TTS]
  end
  subgraph Integrations
    I1[Gmail]
    I2[Vercel]
    I3[Home Assistant]
    I4[Social APIs]
    I5[VoIP]
    I6[Delivery APIs]
  end
  U1 --> O1 --> O2 --> O3 --> O4 --> O5
  O5 --> I1 & I2 & I3 & I4 & I5 & I6
  O5 --> O4
  O5 --> O6 --> U2
  U2 --> O4 --> O5
```

---

7) Trademark Notice

Project ZYBS™ — Trademark claim by **Carlos Hufano III — Scientist & Innovator**.\ © 2025. All rights reserved.