PERSONAL ORB ASSISTANT — FULL SYSTEM SPEC

Your agent is:
A unified personal AI that lives on your PC + phone, orchestrates multiple frontier models, understands your life, watches your tools, manages your data, and assists in every aspect of your work and health.

Below is the **end-to-end specification** of what it will do, how it works, and what modules exist.

---

1. SYSTEM ARCHITECTURE

1.1 Local Orchestrator (PC)

- Runs a **FastAPI backend**.
- Stores all persistent data in **SQLite** (later Postgres if needed).
- Handles:
    - Routing between GPT / Claude / Gemini
    - Local tools execution (files, logs, git, graphs)
    - Auth, permissions, memory
    - Conversation session management

This is the **brainstem**, not the “thinking brain”.

1.2 Cloud Models

Each model has a **role**:

GPT – Communication & Research Engine

- Talks to you naturally (your voice + tone).
- Organises your thoughts into clean tasks/specs.
- Performs online research.
- Writes summaries, explanations, emails, documentation.
- First model in the funnel for all voice input.

Claude – Engineering & High-Reasoning Engine

- Architecture and system design.
- Deep planning, long-horizon reasoning.
- Security reviews & threat modelling.
- Code creation/refactoring/migrations.
- Processes bug reports + crash logs.
- Generates design docs and implementation plans.

This is the **planner brain**.

Gemini 3 – Vision & Multimodal Engine

- Reads screenshots, UI states, documents, web pages.
- Analyses form videos (fitness).
- Analyses your app UI via phone screen-mirroring.
- Handles image/video interpretation tasks.

This is the **eyes**.

1.3 Front-End Interfaces

Phone Orb App

- Voice input (push to talk + wake word “Orb”).
- Reads replies out loud automatically.
- Shows transcript + context.
- Quick commands (log weight, log food, log tasks).
- Can send bug reports, screenshots, or camera photos to backend.

Desktop Orb App

- Same voice in/out + transcript.
- **Screen-Tutor mode**:
    - Window capture
    - Full-screen capture (optional mode)
    - Cursor tracking
- Panels for:
    - Tasks & projects
    - Fitness dashboard
    - Calendar & reminders
    - Logs/history

Optional Web Dashboard (later)

- Read-only or admin dashboard for tasks, fitness, calendar.

---

2. CORE CAPABILITIES (ALL-DAY ASSISTANT)

2.1 Conversational Brain

- Continuous conversation across devices.
- Remembers:
    - Priorities
    - Preferences
    - Routines
    - Style of communication
- Can be asked:
    - “Orb, organise today’s thoughts.”
    - “Orb, what were yesterday’s ideas?”
    - “Orb, summarise this week’s plans.”

2.2 Memory & Structured Knowledge

Backend stores:

- Notes
- Ideas
- Bugs
- Tasks

- Calendar events
- Fitness logs
- App logs
- Dev architecture docs
- Personal routines & preferences

This becomes the agent's **long-term brain**.

---

3. FITNESS MODULE (HIGH PRIORITY)

3.1 Input Methods

- Voice logging (food, workouts, weight).
- Screenshots:
    - Garmin stats
    - Scale app
- Direct reading via Health Connect (if compatible).

3.2 Logs & Models

- `nutrition_entries`
- `body_metrics`
- `workouts`
- `sleep`

- `training_load`

3.3 Features

- Daily macros & calories.
- Weekly summaries.
- Graphs:
    - Weight trend
    - Sleep quality
    - Workout frequency
    - Macros over time
- Smart workout generation:
    - “Orb, give me a workout based on this week.”
- Injury-aware advice (not medical)
- Pattern detection:
    - “Your sleep is low on high-mileage days.”
    - “Protein has dipped below target this week.”

---

4. SCREEN-TUTOR MODULE (HIGH PRIORITY)

4.1 Capture Options

- **Window-specific capture** (default)
- **Full-screen capture** (manual opt-in)
- Frame capture 1–2 fps (adaptive)

4.2 Cursor Awareness

- Local app tracks cursor coords.
- Model receives cursor + frame:
    - “Your cursor is on the wrong menu.”
    - “Move slightly right.”
    - “Click there.”

4.3 Instructor Mode

You say:

- “Orb, help me configure this server panel.”
- “Orb, help me in Premiere.”
- “Orb, show me where I’m supposed to click next.”

The agent:

- Analyzes frame via Gemini.
- Describes step-by-step exactly what to click.
- Acts like a tutor watching over your shoulder.

4.4 Phone-Mirroring Support

Using scrcpy or similar:

- Phone screen appears as a window.
- Orb sees the window exactly like desktop apps.
- Can watch you run Driver Co-Pilot or PT app.
- Can evaluate UX flow, find friction points, confirm behaviour.

---

5. ENGINEERING & DEVELOPMENT MODULE

5.1 Input

- Voice (“Orb, fix the parcel count bug.”)
- Screenshots
- Crash logs
- Driver app bug reports (structured)
- Logs from ADB or manually uploaded

5.2 Claude’s Tasks

- Analyse bugs
- Produce reproducible steps
- Suggest fixes
- Generate code for:
    - Jetpack Compose

    - Kotlin
    - Android Room
    - Backend code
    - Website code
- Review security (GDPR, encryption, retention)
- Identify bad patterns in architecture

5.3 Output

- Dev plans
- File diffs
- Migration scripts
- Red flags for security
- High-level architecture diagrams
- Nightly “dev summary” report

---

6. ADMIN & SECRETARY MODULE

6.1 Calendar System (Custom)

- Agent stores your own events:
    - workouts
    - weigh-ins

    - dev sessions
    - admin tasks
    - delivery work
- Voice control:
    - “Orb, add a 2-hour dev block tonight.”
    - “Orb, move tomorrow’s workout to the afternoon.”

6.2 Reminders & Routines

- Recurring:
    - weigh-ins
    - fitness summaries
    - business finance check
    - development review
- Smart suggestions:
    - “You missed two sleep logs; log now?”
    - “Want to schedule a gym session based on your last one?”

6.3 Document/Browser Assistance

Using screen-tutor:

- Help filling legal forms
- Navigate tax portals
- DNS setup guidance

- New software onboarding

6.4 Email/Communication

- Drafts:
    - business emails
    - client messages
    - refunds
    - release notes
    - investor/partner communication

GPT handles:

- Tone
- Formatting
- Clarity

---

7. DRIVER CO-PILOT INTEGRATION

7.1 Input

- End-of-day summaries (OCR or structured)
- Parcels/stops/mileage
- Fuel usage
- Daily logs from your phone app

- Bug reports, crash logs, images of UI issues

7.2 Agent Tasks

- Analyse earnings vs hours vs miles
- Compare weeks & months
- Spot anomalies
- Flag data that looks off
- Suggest UI improvements
- Fix code issues (via Claude)
- Connect driver workload with fitness output:
    - High-mileage days → recovery advice
    - Hard shifts → adjusted gym plan

---

8. FUTURE FEATURES (PHASED, NOT CORE)

8.1 Gym Scanner

- AR-based mapping of gym layout
- Equipment type detection
- Position mapping to generate efficient workout flows

8.2 Form-checking

- Short video recording → Gemini
- Basic technique cues
- Injury warnings
- Disclaimers included

8.3 Multi-user PT App (commercial)

- Orb's fitness logic → packaged product
- Multi-tenant DB
- User-specific coaching

8.4 Business AI layer

- Client management
- Automated onboarding
- Invoice generation
- Website integration
- Order handling

---

9. PRIORITY ROADMAP — WHAT TO BUILD FIRST

Phase 0 — Local backend foundation

- FastAPI + SQLite

- `/chat` endpoint
- Basic task storage
- Voice in/out on desktop & phone

Phase 1 — Fitness module

- Log nutrition, weight, workouts
- Daily summaries
- Graphs + insights

Phase 2 — Screen-tutor module

- Window capture
- Full-screen mode (opt-in)
- Cursor tracking
- Tutor instructions

Phase 3 — Dev/planner engine (Claude)

- Bug ingestion
- Architecture advice
- Code generation
- Nightly dev plans

Phase 4 — Admin & Calendar

- Custom calendar

- Reminders
- Document guidance

Phase 5 — Driver Co-Pilot integration

- Structured ingestion from your delivery app
- Earnings analysis
- UX review through phone-mirror

Phase 6 — Gym scanning, PT packaging, business layer

- AR scanning
- Form check
- Commercial PT product
- Website integration