

Jarvis — back-end handleiding

Dit is de complete technische gebruiksaanwijzing van **Jarvis** (jarvis.haartips.nl).

Doel: altijd precies weten hoe Jarvis werkt en waar je moet kijken bij een storing.

Alle info komt uit de geheugen-bestanden, de sessie-archieven én de **live** config op de VPS (gecheckt 28 juni 2026).

1. Wat is Jarvis?

Jarvis is **Jamals persoonlijke AI-assistent** met stem en camera.

Je praat live met hem in je browser (Google Gemini Live, natuurlijke stem).

Jamals regel: "**Jarvis moet alles kunnen wat Conductor ook kan.**"

- **URL:** <https://jarvis.haartips.nl>
- **Huidige versie:** v19 (label op de pagina: "versie 19 · pauzeknop · 27 jun")
- **Stem:** Sulafat (was eerder Charon)

De 3 standen

Op de pagina kies je een knop bovenaan:

- **Persoonlijk** — Jamals eigen assistent. Kan huis bedienen, internet zoeken, berichten sturen, onthouden, taken delegeren, verslagen lezen.
- **Salon** — haaradvies voor klanten. Alleen haar. Off-topic wordt teruggeleid. Verwijst nooit naar andere AI of websites.
- **Sport** — fitness/houding-coach via de camera (squat, push-up, plank, lunge, crunch). Vegetarisch eiwit-advies. Geen arts/fysio.

2. Architectuur

ONDERDEEL	WAARDE
URL	<code>https://jarvis.haartips.nl</code>
Server	VPS <code>149.210.171.174</code> (user <code>jamalhaarvisie</code>)
Frontend	static <code>/var/www/jarvis-haartips/index.html</code> (v19)
Stem-relay (Live)	<code>jarvis-live.service</code> → <code>/opt/jarvis/jarvis-live.py</code> op <code>127.0.0.1:8749</code>
Tool-API (functies)	<code>jarvis-api.service</code> → <code>/opt/jarvis/jarvis-api.py</code> op <code>127.0.0.1:8748</code>
nginx route <code>/</code>	static HTML, <code>Cache-Control: no-store</code> (verse versie gegarandeerd)
nginx route <code>/api/</code>	proxy → <code>127.0.0.1:8748</code> (de tool-API)
nginx route <code>/live</code>	WebSocket-proxy → <code>127.0.0.1:8749</code> (Upgrade-headers, timeout 600s)
AI-model tekst/tools	<code>gemini-2.5-flash</code> (in <code>jarvis-api.py</code>)
AI-model stem (Live)	<code>gemini-2.5-flash-native-audio-preview-09-2025</code>
Stem-naam	Sulafat (<code>var VOICE</code> in <code>index.html</code>)
DNS	A-record <code>jarvis</code> → <code>149.210.171.174</code> (via TransIP API)
SSL	Let's Encrypt (certbot), ECDSA. Verloopt: 12 september 2026 (op 28-06 nog 75 dagen geldig, auto-renew)
Gemini-key	staat in <code>/opt/jarvis/env</code> als <code>GEMINI_KEY</code> (chmod 600) — waarde niet hier vermeld

nginx-routes (samengevat)

- `/` → static map, geen cache
- `/api/` → tool-API op 8748

- `/live` → WebSocket-relay op 8749
- Poort 80 → 301 redirect naar https

systemd-services

- `jarvis-api.service` — **actief/running** — Gemini-proxy + alle tool-endpoints (poort 8748)
- `jarvis-live.service` — **actief/running** — Gemini Live WebSocket-relay (poort 8749)

Beide draaien als user `jamalhaarvisie` , met `EnvironmentFile=/opt/jarvis/env` ,
`Restart=on-failure` .

3. Waar staat wat (exacte paden)

Op de VPS (`149.210.171.174` , user `jamalhaarvisie`):

- Frontend/ website: `/var/www/jarvis-haartips/index.html`
- Product-foto's (publiek): `/var/www/jarvis-haartips/producten/<hash>.jpg`
- Tool-API: `/opt/jarvis/jarvis-api.py`
- Stem-relay: `/opt/jarvis/jarvis-live.py`
- Env/ geheime sleutels: `/opt/jarvis/env` (keys: `GEMINI_KEY` , `SUPABASE_URL` , `SUPABASE_KEY` , `TELEGRAM_BOT_TOKEN` , `JAMAL_CHAT` , `JARVIS_REC_DIR`)
- Jarvis' geheugen (vaste feiten): `/opt/jarvis/jarvis-memory.md` (`chmod 666`, schrijfbaar)
- Gesprek-logging: `/opt/jarvis/conversations.jsonl`
- Opnames: `/opt/jarvis/recordings/`
- Backups: in `/opt/jarvis/` als `jarvis-api.py.bak-*` en `jarvis-live.py.bak-*`
- systemd-units: `/etc/systemd/system/jarvis-api.service` + `/etc/systemd/system/jarvis-live.service`
- nginx: `/etc/nginx/sites-enabled/jarvis.haartips.nl`
- SSL-certificaat: `/etc/letsencrypt/live/jarvis.haartips.nl/`

Op de MacBook (Conductor):

- Verzend-poller: `~/ .claude/scripts/jarvis-send-poller.py` (LaunchAgent `nl.haarvisie.jarvis-send-poller` , elke ~25s)
- Poller-log: `~/ .claude/logs/jarvis-send-poller.log`

LET OP: `index.html` , `jarvis-api.py` en `jarvis-live.py` zijn deels root-owned. Voor edits: `sudo tee /` `sudo mv` . De map `producten/` is van `jamalhaarvisie` .

4. Hoe draait het + veilig herstarten

De keten in het kort

1. Jamal opent `jarvis.haartips.nl` en kiest een stand.
2. Browser verbindt via `/live` (WebSocket) met `jarvis-live.py` (8749).
3. `jarvis-live.py` pijpt de audio door naar Gemini Live (key blijft serverside).
4. Als Jarvis een functie aanroept (bv. bericht sturen), gaat dat naar `/api/...` op `jarvis-api.py` (8748).
5. Berichten/foto's gaan via Supabase-queue → de **poller op de MacBook** → Beeper/Telegram.

Veilig herstarten

Herstart **alleen de Jarvis-service**, nooit breed killen:

```
# stem-relay (gesprek werkt niet / verbinding valt weg)
sudo systemctl restart jarvis-live
```

```
# tool-API (functies werken niet: bericht/zoek/product/onthoud)
sudo systemctl restart jarvis-api
```

```
# status / logs bekijken
systemctl status jarvis-live
journalctl -u jarvis-api -n 50 --no-pager
```

NOOIT een brede `kill python3` of iets dergelijks — op deze VPS draaien ook Coco, Appel en andere services op `python3` (8748-8753). Die zou je dan meeslepen.

Na een wijziging deployen

Edit HTML/py → opnieuw plaatsen → `sudo systemctl restart jarvis-api` (en/of `jarvis-live`).

De frontend heeft `Cache-Control: no-store` , dus een verse versie wordt gegarandeerd geserveerd. Toch kan een al-open tab de oude JS in geheugen houden → tab **echt sluiten en heropenen**.

Belangrijk (max 3 SSH/min): fail2ban op de VPS. Bundel commando's in 1 ssh-call en wacht ~25s tussen calls.

5. Tools die Jarvis heeft

Jarvis roept "functies" aan tijdens het gesprek. Deze zitten als declaraties in `index.html` en worden uitgevoerd via `/api/...` op `jarvis-api.py` :

- **bedien_rolluik** — rolluiken open/dicht.
- **bedien_apparaat** — lampen/stekkers aan/uit/dimmen. Loopt via `jarvis-api` → licht-dashboard (8770) → Supabase `licht_commands` → **Paco** voert fysiek uit (Paco moet online zijn).
- **zoek_internet** — web-zoeken. `/api/zoek` doet een grounded Gemini-call (`google_search` , model `gemini-2.5-flash`).
- **stuur_bericht** — WhatsApp/Telegram via Beeper. Jarvis leest ontvanger + tekst hardop terug, vraagt "klopt dat?", bij "ja" roept hij de functie écht aan → queue → poller (MacBook) → Beeper.
- **stuur_product_foto** — product opzoeken + foto sturen. Grounded Gemini (naam/tarief/link) → Bing image search → Gemini Vision-controle ("is dit echt product X?") → foto via Telegram (naar Jamal) of Beeper-attachment (naar contact).
- **onthoud_dit** — feit opslaan in `/opt/jarvis/jarvis-memory.md` (wordt bij opstart geïnjecteerd).
- **delegeer_taak** — taak naar Conductor (queue, marker `JARVIS-TAAK`). Voor alles wat Jarvis zelf niet kan (mail, bestanden, enz.).

- **lees_verslagen** — keyword-zoek in `/var/www/verslag-haartips/reports/*.md` → spreekt de inhoud uit. Bij "laatste/recente verslag" pakt hij het nieuwste bestand (mtime).

Extra functies (geen losse tool, wel ingebouwd)

- **Camera** — JPEG-frames naar Gemini → Jarvis ziet wat je laat zien (haar, oefening, product).
- **Live meeluisteren / monitoring** — <https://jarvis.haartips.nl/monitor.html> (PIN 2026). Per-bot ringbuffer in `jarvis-api.py`, polt `/api/live-feed?bot=` .
- **Bluf-detector** — in `jarvis-live.py`: als Jarvis zegt "verstuurd" maar de tool níet aanriep, krijgt Jamal een seintje op Telegram.
- **"Bespreek met Jarvis"-knop** — op elk verslag/podcast: opent `jarvis.haartips.nl/?verslag=<slug>` en injecteert de verslag-inhoud direct in de `systemInstruction`.
- **Gesprek-verslag** — na elk gesprek schrijft Jarvis een samenvatting naar Conductor (`queue, from_agent=jarvis`).

6. Changelog / upgrades (laatste ~14 dagen)

- **14-06** — v1 gebouwd: stem in/uit (Web Speech) + typen + camera-snapshot. Daarna meteen omgebouwd naar de **Live**-versie (Gemini native audio, stem Charon). Oude tekst-versie op `/tekst.html` .
- **20-06** — apparaat-bediening + rolluiken toegevoegd; gesprek-logging naar `conversations.jsonl` ; verslag-naar-Conductor na elk gesprek.
- **21-06** — tools-botsing gefixt (`googleSearch` + `functionDeclarations` samen → model liet functies vallen); web-zoeken nu als functie `zoek_internet` . NL-taallock. **Stap 3: berichten via stem** gebouwd (`stuur_bericht` → `queue` → `poller` → Beeper). Cache-Control no-store toegevoegd. Slimme naam-herkenning (fuzzy match). Deterministische Beeper-verzender (Telegram eruit). Label → "versie 8".
- **22-06** — **Stap 4**: écht geheugen (`onthoud_dit`), taken (`delegeer_taak`), verslagen (`lees_verslagen`). Geheugen-injectie bij opstart + laatste 6 samenvattingen.
- **23-06** — **Punt 1**: `stuur_product_foto` (grounded zoeken + Bing-foto + Vision-controle). Frontend **v13**. **Live monitoring** (`monitor.html`, PIN 2026). Berichten-bug echt gefixt (gearchiveerde Beeper-chats via `?query=` + KNOWN-aliassen).

- **24-06** — Bluf-detector in `jarvis-live.py`. Geheugen-fix (filter `GESPREEK` in de query, zodat `SEND`-jobs het geheugen niet vervuilen).
- **26-06** — **v17**: SPORT-COACH-stand toegevoegd (houding-feedback via camera).
- **26-06** — **v18**: "Bespreek met Jarvis"-knop op elk verslag (verslag-inhoud direct in `systemInstruction`).
- **27-06** — **v19**: pauzeknop toegevoegd. Jarvis-kraak gefixt (`AudioContext` op vaste `sampleRate 24000`). Stem inmiddels **Sulafat**.

7. Bekende problemen + oplossingen

PROBLEEM	OORZAAK	FIX
Jarvis "kan ineens minder" (geen huisbediening)	<code>googleSearch</code> + <code>functionDeclarations</code> botsten in <code>native-audio</code>	<code>googleSearch</code> eruit, web-zoeken als functie <code>zoek_internet</code> (21-06, gefixt)
Jarvis "bluft": zegt verstuurd maar doet niks	Live-model riep <code>stuur_bericht</code> niet écht aan (praatte er alleen over)	Anti-bluf prompt + bluf-detector → seintje naar Jamal (niet 100% te garanderen)
Oude pagina / functie "werkt niet"	Browser-cache; al-open tab houdt oude JS	<code>Cache-Control: no-store</code> staat live; tab echt sluiten + heropenen. Check versie-label op de pagina
Bericht naar contact mislukt stil	Beeper <code>/v1/chats</code> gaf alleen actieve chats; Asuman/Erik gearchiveerd	<code>poller</code> óók <code>?query=</code> + <code>KNOWN</code> -aliassen (<code>papa=70</code> , <code>asuman=75</code> , <code>eric/erik=97</code> , <code>gerrie=194</code>) + seintje bij mislukking
"papa" matchte verkeerd	Beeper-zoek gaf rare 'Pappa'-varianten	<code>KNOWN</code> -map → rechtstreeks naar <code>chatID 70</code> (Jamals eigen WhatsApp)
Transcriptie schakelt naar Arabisch	tv-/achtergrondgeluid bij onduidelijke audio	NL-lock in systeemprompt; wisselt alleen op expliciet verzoek van taal

PROBLEEM	OORZAAK	FIX
Verkeerde productfoto verstuurd	Bing-relevantie alleen is niet genoeg	Gemini Vision-controle per kandidaat ("is dit echt product X?")
Bing geeft slechte foto's	rate-limit bij veel snelle tests vanaf zelfde IP	niet hameren tijdens testen; trekt vanzelf bij
Lampen gaan niet écht aan	Paco (fysieke uitvoerder) offline	Paco online brengen; commando staat in Supabase <code>licht_commands</code>
Audio-opname komt niet binnen op Telegram	jarvis-live stuurde ruwe .wav (Telegram weigert stil)	wav→mp3 vóór <code>sendAudio</code> (in <code>appel-live</code> al gefixt; jarvis-live had dezelfde bug nog open — checken)
Geluid kraakt	per-chunk resampling	<code>AudioContext</code> op vaste <code>sampleRate 24000</code> (27-06, gefixt)
Monitor leeg na herstart	ringbuffer in geheugen	normaal; buffer vult weer na het eerste nieuwe gesprek

8. Storing-zoekgids ("Jarvis werkt niet")

Werk van boven naar beneden. Wacht ~25s tussen SSH-calls (fail2ban).

1. Site laadt niet?

- Check DNS/SSL: `curl -I https://jarvis.haartips.nl` (verwacht HTTP 200).
- SSL verlopen? `sudo certbot certificates | grep -A4 jarvis` (verloopt 12-09-2026, auto-renew).

2. Site laadt, maar geen gesprek / stem?

- Is de Live-relay up? `systemctl status jarvis-live`
- Zo nee: `sudo systemctl restart jarvis-live`
- Logs: `journalctl -u jarvis-live -n 50 --no-pager`
- Mic-rechten in de browser? Bij ingebedde Jarvis (hub) moet de `iframe` `allow="microphone; camera"` hebben.

3. Oude versie / functie ontbreekt?

- Check het versie-label op de pagina (nu "versie 19").
- Tab echt sluiten + heropenen (niet minimaliseren).

4. Functies werken niet (bericht/zoek/product/onthoud)?

- Is de tool-API up? `systemctl status jarvis-api`
- Zo nee: `sudo systemctl restart jarvis-api`
- Logs: `journalctl -u jarvis-api -n 50 --no-pager`
- Gemini-key aanwezig? Check dat `/opt/jarvis/env` de regel `GEMINI_KEY=` heeft (waarde niet tonen).

5. Bericht/foto komt niet aan?

- Draait er een `JARVIS-SEND` in Supabase `agent_messages` ? Zo nee → het model riep de functie niet aan (bluf) → opnieuw proberen.
- Zo ja → check de poller op de MacBook: `launchctl list | grep jarvis-send-poller` + `log ~/ .claude/logs/jarvis-send-poller.log` .
- Beeper-token nog geldig? (Keychain "Claude Code-credentials").

6. Lamp/rolluik reageert niet?

- Is Paco online? (fysieke uitvoerder). Commando staat in Supabase `licht_commands` .

7. Laatste redmiddel: alleen de Jarvis-services herstarten (zie sectie 4). **Nooit** breed killen — andere bots delen python3 op de VPS.

Bronnen: geheugen-bestanden (project_jarvis_assistent, reference_jarvis_v1, reference_jarvis_product_foto, project_email_assistent), sessie-archieven 20-28 juni, en live VPS-config (gecontroleerd 28 juni 2026).