



# Genesis

## Reference Manual

European funding workflow · by Solo Lattices

*Version 17 · July 2026*

*Author: Dr. Javier Torralba*

Reference manual · intensive use · for Javier, David and Carlos

Author: Dr. Javier Torralba · COO, Solo Lattices

Horizon Europe · Erasmus+ · Irish funders | English / Español | Research preview

<https://genesis-funding-app.vercel.app/manual.html>

Solo Lattices · Waterford, IE · 2026

**For whom:** Javier, David, Carlos and any Genesis operator. It covers each and every capability of version 17: the 11-stage flow, the V17 control plane and the enclave portal. The short applicant version is the "User Manual v17".

## CONTENTS

- |  |  |
|--|--|
| 1. What Genesis is                                       | 10. Stage 6: work plan, budget and risks               |
| 2. The boundary banner                                   | 11. Stages 7, 8 and 9: draft, LOACF audit and red-team |
| 3. Accounts, roles and responsibility                    | 12. Stage 10: the submission package                   |
| 4. Surfaces, modes and languages                         | 13. Provenance, autosave and accelerators              |
| 5. The method: 11 stages, 5 phases, gates                | 14. Notifications and automations                      |
| 6. Stage 0: opportunity, call and deadline               | 15. Control (V17): the command post                    |
| 7. Stage 1 (V17): official sources, matrix and documents | 16. The customer enclave portal (V17)                  |
| 8. Stages 2 and 3: fit, evidence and claims              | 17. Technical architecture and data                    |
| 9. Stages 4 and 5: concept and consortium                | 18. Operations, writing rules and roadmap              |

# 1. What Genesis is

Genesis is the Solo Lattices web application for preparing European funding proposals (Horizon Europe, Erasmus+ and Irish funders). It turns an overwhelming application into a guided 11-stage workflow (Stage 0 to 10) with structural checks (gates) at every step, and it ends in a submission package ready for the official portal.

Version 17 is the first to carry the enclave layer of the Geny engine: besides Studio (where a proposal is worked) and Control (oversight for David and Javier), there is now a customer enclave portal, in pilot, and a V17 control plane inside Control with Validation, Enclaves and Packet runs. All of it is documented in this manual.

## What it certifies and what it does not

**The rule that governs everything:** Genesis structures and checks a proposal. It does not certify fundability, it does not promise scores, and it does not declare anything production ready. Official sources, the gates and a human signature decide.

That rule is not a disclaimer: it is wired into the interface. The boundary banner (section 2) shows the live values of the gate report on every screen, and they will stay negative until signed human evidence exists. In Genesis, negative is the designed state, not an error.

## The pieces at a glance

PIECE	WHO USES IT	WHAT IT DOES
<b>Studio</b>	Applicants (and us)	Runs the Stage 0 to 10 flow: call, sources, evidence, concept, consortium, work plan, draft, audit, red-team, submission.
<b>Control</b>	owner/admin	Read-only portfolio oversight, with a chart panel and the V17 control plane (Validation, Enclaves, Packet runs).
<b>Enclave portal</b>	Enclave contract customer (pilot)	Isolation verification: Seal Check, egress ledger, certificates. Read only, one identity one enclave.

## 2. The boundary banner

In V17, every operator screen (Studio and Control) carries a slim band under the header with three values read from the v10 gate report ( `controls/v10_trust_layer_boundary.yaml` ), rendered verbatim, never paraphrased:

VALUE	TODAY	MEANING
<code>boundary_state</code>	<code>human_validation_ready_not_release_ready</code>	The machinery is ready for validation with humans; it is not ready for release.
<code>release_ready</code>	<code>false</code>	No signed, verified human approval exists yet.
<code>compliance_ready</code>	<code>false</code>	Nothing may be described as compliant or certified.

The band is centered, never hidden, and does not soften when the values are negative. Hovering shows the source file. The values will change the day the approver key registry stops being empty and verified Ed25519 approvals exist (section 15): until then, negative by design.

**Banned across the product:** success language (win rates, "improve your chances"), absolute anti-hallucination promises, and the words "compliance-ready" or "release-ready" outside the gate values. Where a statement is needed, the declared measure is used: claim extraction has a boundary measured by the B9 benchmark.

## 3. Accounts, roles and responsibility

Genesis uses Supabase Auth with email. Every new account starts with the `user` role. The `admin` and `owner` roles see Control; everyone else does not know it exists.

## Role pre-authorization

The system owner is Javier Torralba ( `javier@sololattices.com` ). The `role_grants` table assigns a role in advance: insert the email and the role, and when that person signs up the system applies it automatically. That is how David ( `dalarco@sololattices.com` , admin) was set up before his first login. Nobody on the team handles anyone else's password: each person creates their own.

## Data isolation (RLS)

Every table carries Row Level Security: a user sees their proposals and only theirs; owner/admin read everything. Attachments live in the `proposal-documents` bucket under the same rules.

## The responsibility log

A Genesis principle: there is always a named human behind every confirmation. Every relevant action writes an event with `actor = "Name · email"`: marking a source reviewed, selecting a call, passing a gate, exporting. Gate passes also record the decision with its reviewer. Automatic source preloading is logged as `System · auto-load`, because a human did not do it and the log does not lie. This log feeds the "Responsibility log" card on each proposal and the Evolution section of the Control panel.

---

# 4. Surfaces, modes and languages

**Studio** is what everyone sees on login: My proposals, New proposal, Guide, manuals. **Control** appears in the sidebar only for owner/admin. The **enclave portal** is a separate surface (different page, different access system): an operator never browses inside an enclave from the console, and a customer never sees operator navigation.

## Simple and Expert modes

The toggle lives at the top. In Simple, the 11 stages group into 5 phases in plain words ("Worth it?", "Prepare", "Write", "Review", "Send"), one clear action per phase and per-step help cards. In Expert you see the stages under their technical names, the full Stage 0 to 10 rail and all the machinery. The choice is remembered per user.

## Languages

The full interface exists in EN, ES, IT, FR and DE. The selector sits in the top bar and the preference is saved to the profile. The operator console and the enclave portal are English first, per the project rule; Spanish versions arrive with the commercial site.

## Built-in help

Every field, artefact and gate carries an ⓘ icon with an explanation and an example. The Guide ( `#/guide` ) summarises the 5 steps for non-experts. This manual and its short companion download from the sidebar.

## 5. The method: 11 stages, 5 phases, gates

The engine is an 11-stage method. Each stage requires inputs, produces artefacts and ends in a structural gate. A stage never shows complete while required artefacts are missing or blockers stand unresolved or without an explicit waiver: that rule comes from v8.1.1 and remains intact in V17.

PHASE (SIMPLE)	STAGES	WHAT HAPPENS
1 · Worth it?	S0	Opportunity intake, call and deadline.
2 · Prepare	S1 to S6	Sources and matrix, fit, evidence and claims, concept, consortium, work plan.
3 · Write	S7	Draft from approved artefacts, no unsupported upgrades.
4 · Review	S8 and S9	LOACF audit and evaluator red-team.
5 · Send	S10	Multi-format submission package.

States always use colour plus label plus icon, never colour alone:

✓ pass

♦ in progress

⊘ blocked

△ needs refresh

○ waived

A waiver is always a recorded human decision, never a silent shortcut.

## 6. Stage 0: opportunity, call and deadline

Stage 0 captures the opportunity: name, framework (Horizon, Erasmus+, Ireland, other), programme, idea, lead applicant, likely partners, strategic fit and deadline.

### Programme selector

The "App / route" field is a per-framework dropdown: 12 Horizon programmes (clusters, ERC, MSCA, EIC...), 11 for Erasmus+ (KA1, KA2, COVE, Mundus...), 9 for Ireland (Research Ireland, EI, EPA, SEAI, DTIF, HRB...). Picking a programme makes Genesis look up the next call and its deadline.

### Call finder (SEDIA)

The finder queries the official Funding & Tenders portal through the `call-lookup` function. Three modes: by keywords, by exact topic code, and idea-to-call (describe the idea, get candidate calls). Selecting a call writes code, deadline and status, and the deadline is flagged as automatic.

## Deadline traffic light

STATE	SIGNAL
open with a future deadline	Green banner with the date and days left.
forthcoming	Amber banner: planned, not yet open.
closed or no upcoming call	Red banner: no reachable deadline.
unknown	Warning chip: verify by hand.

A manually set deadline is never overwritten automatically. Buttons: re-check deadline (repeats the lookup) and download an .ics calendar reminder. Decentralised Erasmus+ (national agencies) and Ireland have no per-topic API: there the deadline is set by hand and the banner says so.

## 7. Stage 1 (V17): official sources, matrix and documents

Nothing gets written without fresh official sources. On first entry to stage 1, Genesis preloads the canonical catalogue of the chosen framework (F&T portal, programme guides, AGA, agencies, Irish funders) as "pending human review": preloads do not count as reviewed until a human marks them.

### The source register, column by column (V17 additions marked)

COLUMN	WHAT IT IS
Title / URL / Type	The source and its document class (portal, guide, legal...).
<b>Capture (V17)</b>	<code>capture_class</code> : <code>portal</code> (preloaded official catalogue), <code>api</code> (live lookup) or <code>human-supplied</code> (added by hand), plus who captured it and when.
<b>Route (V17)</b>	The source's framework. If the framework has no official domain list (the "other" framework), the amber chip <i>Route blocked (this framework only)</i> appears: the block applies to that route only, never globally.
Reviewed	Date of the last human look.
<b>Freshness (V17)</b>	Freshness window: 14 days in general, tightening to <b>3 days when the call deadline is 30 days away or less</b> . A source reviewed longer ago than the window shows "Needs refresh" and blocks the gate.

**Why the window tightens:** near the deadline, funders publish corrigenda and FAQs often. Genesis forces a look at the sources every 3 days in that stretch because that is when they change most.

## Rule matrix and documents

The matrix turns the call into checkable rows (eligibility, budget, deadlines, evaluation criteria, annexes), each with a backing status: sourced, partial, inferred, stale, missing or conflicting. Call documents upload next to the sources. **Gate 1:** at least one official source and none pending refresh.

## 8. Stages 2 and 3: fit, evidence and claims

**Stage 2 (fit).** Before investing weeks: an idea-to-call fit report and a differentiation map against alternatives. Gate 2: both filled in.

**Stage 3 (evidence and claims).** The anti-smoke heart of Genesis. The evidence table gathers data and references; the claim manifest lists every relevant claim in the text with its class:

CLASS	MEANING
grounded	Anchored to a registered source, with a location.
inferred	Reasonably derived from evidence, with a derivation note.
generated	Produced during drafting; pending anchoring or removal.
unsupported	No backing: blocks.
prohibited	Of the kind never promised (outcome guarantees): blocks.

**Gate 3:** evidence table filled, at least one claim, and none unsupported or prohibited. Classification has a declared, measured boundary (the B9 benchmark); that is why human review and stage 8 exist.

## 9. Stages 4 and 5: concept and consortium

**Stage 4 (concept).** Three artefacts: objective tree, golden thread (need, activity, result, impact) and intervention logic. The golden thread is a gate and a diagnostic tool: it checks the chain does not break. It is not a score promise. Gate 4: tree and thread filled.

**Stage 5 (consortium).** Topic-based partner search across the official portals (CORDIS, F&T Partner Search, Enterprise Europe Network, EPALE-OTLAS), with per-call direct links. Genesis suggests a consortium shape per framework (for Horizon: coordinating university, RTO, innovative SME, industrial demonstration partner) with each role's non-substitutability. Every partner carries a state: suggested, candidate, confirmed or declined. The capacity gap list records what is still missing. Gate 5: minimum shape covered and gaps documented.

## 10. Stage 6: work plan, budget and risks

Work packages with a lead and months (the Gantt draws itself), budget lines per partner and category, and a risk register with likelihood, impact and mitigation.

Per-framework starter templates create the typical structure (management, technical work, dissemination and exploitation, quality) with all amounts at zero: Genesis does not invent figures, it lays the skeleton and you decide the numbers. Gate 6: WP matrix, budget logic and risk register present.

## 11. Stages 7, 8 and 9: draft, LOACF audit and red-team

**Stage 7 (draft).** Evaluator-facing sections are written from the approved artefacts, without introducing unsupported upgrades: if the draft says something absent from evidence or matrix, that sentence is born as a "generated" claim and must be anchored or removed.

**Stage 8 (LOACF).** Coherence audit: Length, Organisation, Anchoring, Consistency and Format against the call rules. It produces an actionable repair list: "Gate failed: 2 claims missing anchors. Repair list below." Failure is useful information, never an apology.

**Stage 9 (red-team).** Adversarial read with evaluator eyes: weak points per criterion, uncomfortable questions, gaps. The output feeds the final draft pass. Gates 8 and 9: audit and red-team present and their blockers resolved or explicitly waived.

## 12. Stage 10: the submission package

The end of the flow is a complete package: application text, source register, matrix, claim manifest, consortium, work plan, budget and a submission checklist.

FORMAT	WHEN
PDF	Recommended for Horizon Europe.
Word (.docx)	Recommended for Erasmus+ and Irish funders.
Markdown / HTML / JSON	Reuse, review and archive.

Every export is recorded (who, when, format) and Control flags the proposal as export ready. The actual submission is always done by a person on the official portal: Genesis prepares, it does not submit.

## 13. Provenance, autosave and accelerators

**Provenance chips (V17).** Every artefact carries two non-editable headers: `produced_by_model` and `produced_by_runtime`. In today's Studio, everything you write is recorded as `human · studio`. When an external model takes part (the stage 7 drafting channel in enclaves), its chip renders distinct in purple, for example `claude · external`. The chip is the base of the AI disclosure story: always visible, always accurate, never editable.

**Autosave.** Artefacts save themselves as you type (with a status indicator) and when leaving the field. The Save button remains for those who want the gesture.

**Readiness card.** On each proposal overview: stage progress bar, what the current gate needs, sources to refresh and claim risks. One glance says what is missing.

**Duplicate proposal.** Clones sources, matrix, artefacts, claims, partners, WPs, budget and risks; it does not clone history or gates: the copy starts unpassed, as the audit requires.

**Templates, tooltips, .ics, deadline re-check:** described in their stages; all recorded in the responsibility log.

## 14. Notifications and automations

Genesis notifies inside the application, no email. The bell in the top bar shows unread notifications.

SCHEDULED JOB	WHEN	WHAT IT DOES
Deadline reminders	Daily, 06:00	Alerts at T-30, T-14, T-7 and T-1 for each deadline.
Source freshness	Monday, 06:00	Flags sources unreviewed for a long time (besides the live V17 window of 14/3 days acting in stage 1).
Portfolio digest	Daily, 06:00	Status digest for owner/admin.

The three jobs run in the database (`pg_cron`) and write to the notifications table; the interface only displays them. Nobody outside the team receives anything.

## 15. Control (V17): the command post

Control is read only and owner/admin only. Since V17 it has four tabs.

## Portfolio

The chart panel: KPI cards (proposals with cumulative curve, active, **gates passed X/total** with its weekly chart, open P1/P2, sources to refresh, claim risks, exports ready), the status legend, and the portfolio table where each proposal shows its **11-dot rail** (green passed, blue in progress, red blocked, grey pending) with the current stage as text, split P1/P2 chips, source count and deadline. Below, **Evolution**: gates passed, source updates and issues opened per week (last 8), computed from the responsibility log. Charts only draw series that really exist: a KPI without a history renders as a number, no invented trend.

## Validation (V17)

Three cards: the **release boundary** with the three v10 gate values verbatim; the Ed25519 **approvals panel** against the pinned key registry ( `controls/approver_keys.yaml` ), today showing its exact empty state: *"No approver keys registered. Release remains closed."* and no signed approvals. The panel observes, it does not act: no interface path creates keys or approvals. And **cross-model agreement**: the record of comparison runs against the frozen human yardstick, today with no runs recorded.

## Enclaves (V17)

The enclave registry ( `controls/enclave_registry.yaml` ) as metadata only: id, contract, tier, status, branch, evidence. Today it is empty by decision D-0006 and shows its expected state: *"No enclaves provisioned. Hardware validation E0.2 pending (SETU)."* Hard rule: the console shows metadata and evidence file names, never enclave content. There is no "open enclave" button.

## Packet runs (V17)

The scheduler heartbeat when the agent runs unattended: reverse-chronological view of `runtime/packet_runs.log` with a status filter. Today, no runs recorded.

---

# 16. The customer enclave portal (V17)

The product face of V17: every customer contract has a sealed enclave, and this portal lets the customer verify their isolation in two clicks, knowing nothing about our internals.

## What an enclave is

A protected execution environment (IBM Secure Execution as the trust root) serving the engine to a single contract. Two tiers: `standard_se_guest` and `dedicated_lpar` . Lifecycle: declared, provisioned, sealed, active, closing, shredded (tombstone). The key moments are witnessed human ceremonies: sealing pins the engine weights hash and issues a certificate; destruction requires dual control and issues the destruction certificate.

## Access

Passkey (WebAuthn) first; authenticator code (TOTP) as fallback. There is no password. One identity maps to exactly one enclave: no organisation picker, no URL that resolves across enclaves. The session always

shows id, tier and state, plus the standing line: *"release\_ready remains false; this report proves isolation, not release readiness."*

## The Seal Check

One button: Run Seal Check. It compares the SHA-256 hash of the engine serving you now against the hash pinned at your sealing ceremony, and renders the report with a downloadable JSON twin. Four states:

STATE	PRESENTATION
Running	Quiet progress, no percentage theater.
✓ Seal intact (EQUAL)	Calm confirmation, both hashes side by side. The report's sentence: the engine serving you is byte-identical to the one sealed into your contract.
⚠ Seal mismatch (NOT EQUAL)	Unmissable red alarm; the full report still renders. Report copy: raise this with your provider contact immediately; do not proceed.
⚠ Report refused	If any input is missing, the generator writes nothing: the reason shows verbatim (for example "seal check refused: missing inputs: attestation"). Never a partial or cached report in its place; historical reports are labeled with their generation date.

## Egress ledger and certificates

The egress ledger lists everything that has left the enclave (time, destination, payload hash, stage, gate approval), with the allow-list beside it: you see the complete universe of permitted destinations next to the record of actual ones. An empty ledger is good news and says so: *"Nothing has left this enclave."*

Certificates (sealing and, after contract end, destruction) download from the portal; humans issue them in ceremonies, the portal only serves the files.

## What the portal does not do

No document editing, no proposal browsing, no chat with the model, no admin superview, and not a single tracker: the page makes no external requests. The About page carries the contract's mandatory disclosure paragraph, verbatim.

**Pilot state:** until hardware validation E0.2 (SETU) closes, no real enclave exists. The portal builds and demos entirely from fixture data behind the `?pilot=1` flag; without it, it shows the expected closed state. That is the designed launch state, not an error.

## 17. Technical architecture and data

**Frontend.** Studio and Control live in a single self-contained `index.html` (no build), with supabase-js from CDN. The enclave portal is `portal.html`, also self-contained with zero external requests. Hosted on Vercel with caching disabled so every deploy shows on reload.

**Backend.** Supabase project in the EU: 24+ tables under RLS (proposals, stages, sources, matrix, artefacts, claims, partners, WPs, budget, risks, issues, gates, exports, events, notifications, role\_grants...), a documents bucket, email Auth.

**V17 schema changes** (migration `v17_source_capture_and_artefact_provenance`):

`sources.capture_class` and `sources.captured_by`; `artefacts.produced_by_model` and `artefacts.produced_by_runtime`.

**Functions and jobs.** Edge function `call-lookup` (SEDIA finder); `geny-assist` written but not deployed (the Geny/OpenClaw engine connection is deferred to the end by decision); three `pg_cron` jobs for alerts.

**The V17 tree.** The `geny_v17_core` package (227 audited files of the V16 core plus the V17 enclave layer) is the source of truth for the control plane: enclave registry, approver keys, v9/v10 boundaries, fixtures to build the portal without real data, validators and per-stage skill contracts. The console mirrors it read only.

## 18. Operations, writing rules and roadmap

### Operating

Deploy: `bash deploy.sh` in the app folder (Vercel, dePie team). Database changes go as migrations. This manual's PDFs regenerate with WeasyPrint from `manual.html` and `manual_breve.html` (ES as is; EN by adding `class="show-en"` to the body).

### Project writing rules (the ones that bind the interface)

No em or en dashes in any text: periods, commas, colons, parentheses. No banned words (seamless, robust, leverage, unlock and friends). No "not X, but Y" reframes. Buttons are verbs. Evidence over promises: "12 sources registered, 2 need refresh" and never "your sources are looking great!". The honest negative is part of the product.

### Roadmap

MILESTONE	WHAT IT UNBLOCKS
E0.2 (hardware validation, SETU)	First real enclave can be provisioned; the registry stops being empty.
Legal review of certificates	The annex moves from DRAFT to final; production portal launch with the first sealed enclave (Phase 3).
Approver keys + Ed25519 approvals	The boundary banner values change; the release path opens.
Geny connection (OpenClaw)	In-flow model accelerators, last by decision; <code>geny-assist</code> is ready to be pointed.

## Changelog

VERSION	DATE	WHAT IT BROUGHT
v1.0 (Pre-Geny)	2026-06-20	Full S0 to S10 flow, basic Control, SEDIA finder, in-app automations, manuals.
<b>v17</b>	2026-07-06	Boundary banner; source register with capture, route and the 14/3 window; provenance chips; Control with chart panel and the Validation, Enclaves and Packet runs tabs; customer enclave portal in pilot (Seal Check, egress, certificates); schema migration; full copy sweep to the project rules.

---

Genesis · Reference Manual · v17 · July 2026. Written and maintained by Dr. Javier Torralba. Solo Lattices, Waterford (IE) and Valencia (ES). This manual describes the real state of the product: where something is pending, it says so.