Toranosuke Tanigaki

Software Engineer · Backend (Python / TypeScript) · WordPress & WooCommerce bug fixing and maintenance

founder@sphior.com · toraastrta.github.io · github.com/Toraastrta · linkedin.com/in/toraastrta · Japan (JST), remote

SUMMARY

I build software that people depend on and make it work reliably: payment systems where money cannot be lost to rounding, ERP data that stays consistent across six business areas, and AI features that run 24/7 without an operator. I built these as a contractor in Python and TypeScript, then took my own compliance products from an empty repository to seven marketplaces and run them in production alone.

Today I focus on code I did not write: I reproduce a bug, find its root cause, and ship a small, tested fix that maintainers can merge with confidence. My fixes have been merged in WooCommerce, Supabase, React Router, Zed and PostHog. I work async in written English, and from Japan I keep work moving while the Americas and Europe are offline.

EXPERIENCE

Founder / Engineer

Jul 2025 – Present

SPHIOR — B2B security & compliance SaaS · solo

- Took products from an empty repository to live listings on seven marketplaces on my own, passed each marketplace's review, and run them in production: architecture, API, infrastructure and billing.
- **SPHIOR CODE** (open source, MIT) — gives developers the same security findings in every tool they use, without their code leaving the machine. No AI in detection, so results are reproducible and defensible in an audit: 36 secret rules, 85 pattern rules mapped to CWE and OWASP, and 69 taint rules across 13 languages parsed with real grammars (tree-sitter, Babel). One engine powers the CLI (npm, Homebrew, SARIF), a GitHub Action, VS Code, Open VSX, JetBrains, and an MCP server that lets AI coding agents fix what it finds.
- **Security Diagnosis** — automated external testing of web applications, including pages behind the login, safe to run against live production sites. It never asks for a password: a Chrome extension hands over the login session, encrypted in the browser (AES-GCM with key wrapping) and valid for one scan. Excluded 280 of 7,509 test templates that could change the target and rate-limited every scan; new templates are proven against a local decoy site before they run (1,240 of 1,245 passed, zero false positives).
- Lets customers prove to an auditor that their evidence was never altered, without the auditor having to trust SPHIOR: records are sealed with SHA-256 hash chains and Merkle roots, anchored with OpenTimestamps, mapped to SOC 2, ISO 27001 and OWASP, and checkable with an open-source verifier.
- **Sphior Ledger for Jira** — gives Jira admins the change history and audit evidence Jira does not keep for its own configuration. Passed Atlassian's Marketplace security review; 800+ tests.
- Isolated tenants at the database row level, so a forgotten filter in application code cannot expose another customer's data.

Backend Engineer

Jan 2025 – Jun 2025

Booking & customer-service SaaS — contract · Python, Django, PostgreSQL, AWS

- Let a booking platform answer customer messages around the clock without an operator, by integrating an LLM and building the automation around it; it runs unattended, 24/7.
- Kept bookings correct by never letting the LLM decide: it only turns the customer's words into a structured request, while stock and scheduling checks stay in rule code.
- Lowered LLM cost without hurting reply quality, by routing requests by difficulty, summarising history and retrieving only the context each reply needs, and made every prompt or model change measurable with a fixed-case evaluation harness.

Backend Engineer

Feb 2024 – Dec 2024

Internal OS + ERP — contract · Python, FastAPI, PostgreSQL, TypeScript, React

- Brought accounting, inventory, HR, attendance, CRM and supply chain onto one shared data model, so the same person or item stopped being maintained six times, and added them as modules so the existing database never had to be migrated.
- Protected HR data and the books: role, row-level and org-hierarchy permissions, and inventory movements written in the same transaction as their accounting entries, so a failure cannot leave stock and books out of step.

Backend Engineer

Jan 2023 – Jan 2024

Crypto payments fintech — contract · Python, FastAPI, PostgreSQL, TypeScript, Google Cloud

- Let users hold crypto and fiat in one balance with no rounding errors, by keeping every amount at full precision instead of floating point.
- Let customers pay in crypto while the merchant receives fiat, converting through a stablecoin at transaction time; each stage is persisted, so a failed conversion resumes where it stopped instead of starting over.

SELECTED OPEN-SOURCE FIXES (2026, MERGED)

Supabase Storage — unsupported S3 operations overwrote or deleted data

#1722

- **Problem:** S3 writes for operations Storage does not implement (tagging, ACLs, deleting CORS settings) fell through to the upload and delete routes, replacing the object's contents or deleting the object or an empty bucket, and returned success.
- **Fix:** such requests are now rejected as unsupported before they reach the upload and delete handlers.

React Router — form encoding and history fixes

#15776, #15779

- **Problem:** `<Form encType="Multipart/Form-Data">` was submitted URL-encoded, so file inputs arrived as just their names. `enctype` keywords are case-insensitive; after confirming Chromium's behaviour, I fixed it in the one helper the form and the submit button share.
- **Second fix:** memory history `go()` clamped the index but reported the requested delta, so undoing a blocked navigation overshot. It now reports the distance actually moved, with regression tests for both ends of the stack.

Orca — mobile web app broke on older iPhones

#44941, #45192

- **Problem:** iOS 16.0–16.3 cannot parse regex lookbehind, and one literal anywhere in a loaded chunk stops the whole page. I replaced it with a character class under the `u` flag, which reads a valid surrogate pair as one code point and so matches only lone halves, and kept a replace-based approach because Android WebViews before Chromium 111 lack `toWellFormed()`. Merged the same day.
- **Follow-up:** on iOS before 17.4, picking a model in the mobile chat threw because `Promise.withResolvers` is missing; I replaced it with a plain `new Promise`.

WooCommerce

#70900, #71118

- Removed Mini-Cart footer state that was computed on every render but never read, and kept the now-unused protected method as deprecated rather than deleting it, because extensions can subclass the block and call it.
- Added `role="listitem"` to the Variation Selector rows inside its `role="list"` wrapper, so screen readers announce the list correctly.

Other fixes

- **PostHog #1265:** exception autocapture sent `null` for source lines over 1024 bytes; added a test that fails on the old code.
- **Zed #66974:** corrected the documented defaults of eleven settings. Merged within about an hour.
- **e2e #1837, #1917, #2005:** the replay cache treated growing Japanese, Chinese and Korean counts as fixed anchors, so the next replay failed.
- **Meta Box #1746, LifterLMS #3510:** an icon field showed "[object Object]" on WooCommerce admin screens, and post search results showed raw HTML entities.
- **Hugging Face Hub #5400, Tauri #3793:** fixed documented code examples that failed or did not compile.

SKILLS

Programming	Python, TypeScript, JavaScript, SQL; PHP (since 2026, through open-source WordPress and WooCommerce work)
WordPress	WooCommerce (HPOS, block checkout, Store API, additional checkout fields), plugin hooks and filters, block editor, wp-env, WP-CLI
Backend	FastAPI, Django, Node.js, PostgreSQL, MySQL
Frontend	React, Next.js, Tailwind
Testing & CI	PHPUnit, Jest, Playwright, pytest, GitHub Actions; failing test before the fix
Cloud	AWS, Google Cloud, Azure, Docker, Atlassian Forge
AI	Claude Code for daily development; OpenAI and Anthropic APIs; author of a published MCP server
Languages	Japanese (native), English (professional working proficiency)