

Falase Femi

Fullstack Engineer Location: Remote — International (open to relocation) | Email: femifalase228@gmail.com | Phone: 07013329953

[GitHub](#) | [Portfolio](#) | [npm](#) — [@pharlase/paystack-effect](#)

Summary

Fullstack engineer with professional experience since 2021, building type-safe, fault-tolerant systems end-to-end. Currently at Vatebra, where I rewrote the in-house HR application used by up to 150 staff. Author of a published Effect-TS SDK for the Paystack payment API, builder of production-grade backends (auth, payments, rate limiting, geospatial search) and fast, accessible frontends (React 19, Next.js, TanStack Start). I work across TypeScript/Bun and Go, and ship automated AI-powered CI quality gates.

Technical Skills

Languages: TypeScript, JavaScript (ES2024), Go, SQL, HTML5, CSS3, Python (basic) **Frontend:** React 19, Next.js 16 (App Router), TanStack Start/Router/Query v5, Vite, Tailwind CSS 4, shadcn/ui, Radix UI, Framer Motion/Motion, Leaflet **Backend:** Bun, Node.js, Effect-TS (v3/v4), Hono **Databases:** PostgreSQL (incl. PostGIS), MySQL, Redis; Drizzle ORM, pgx, @effect/sql-pg; SQL migrations **Auth & Security:** JWT (rotation, reuse detection, refresh flows), argon2/BCrypt, RBAC, Clerk **Payments & APIs:** Stripe Checkout + webhooks, Paystack (SDK author), REST API design, OpenAPI/Swagger, typed API clients, Effect Schema validation, SSE/EventSource, retry/backoff **AI/LLM:** Groq API integration, structured JSON output, jittered retries/backoff **DevOps & Tooling:** Docker, GitHub Actions (reusable workflows), CI quality gates, Vercel, Koyeb, Render, Railway, Supabase Storage, UploadThing, Cloudinary, Biome, ESLint, Prettier, pnpm **Testing:** Vitest, Bun test, Go testing

Experience

Frontend Developer — Vatebra Limited | Jun 2025 – Present

- Rewrote the company's in-house HR application supporting up to 150 staff members, overhauling core workflows (employee records, leave, approvals).
- Delivered responsive, accessible frontend features in React/TypeScript for enterprise clients, meeting WCAG and cross-browser standards.

Frontend Developer — Normdek | Apr 2022 – Jan 2023

- Built and maintained customer-facing web interfaces with React and TypeScript, turning design mockups into responsive, production-ready components.

Frontend Developer Intern — Sterling Bank | Jan – Apr 2021

- Contributed to frontend development within an enterprise banking team, building UI components and fixing accessibility and cross-browser issues.

Featured Projects

EasyRent — Fullstack Rental Platform (Backend + Frontend) [Backend](#) · [Frontend](#) · [Live API docs](#) · [Live demo](#)

- Backend (Effect-TS v4 on Bun, Postgres/PostGIS, Redis, Drizzle ORM, Koyeb): JWT auth with argon2, refresh-token rotation with reuse-detection, Redis rate limiting (429 with computed `retryAfter`) and caching, PostGIS geospatial search; 5 Vitest/Effect test suites, Dockerized.
- Frontend (TanStack Start SSR + React 19, migrated from Next.js App Router): typed Effect-TS API client with automatic token refresh on 401, exponential-backoff retries, and Effect Schema-validated responses; map-based discovery (lazy-loaded Leaflet, debounced search, facet filters), TanStack Query v5 with optimistic updates, favorites, responsive galleries in Tailwind CSS 4 + shadcn/ui.

paystack-effect — Published Effect-TS SDK for the Paystack API github.com/Falasefemi2/paystack-effect · [npm](#)

- Authored an Effect-native TypeScript SDK for the Paystack payment API and published it to npm; covers 20+ Paystack resources as typed, schema-validated services (~6,500 LOC).
- Built a declarative `HttpApi` spec with request/response schemas, error tags, and security descriptors, exposed as a Swagger/OpenAPI documentation server runnable on Bun.

local-deploy — Fullstack Local-First Deploy Platform (API + Dashboard) github.com/Falasefemi2/local-deploy

- Backend (Effect-TS v4 on Bun): Vercel-style pipeline — point at a local git repo or GitHub URL, detects project type, builds, archives and pushes to Supabase Storage under a versioned path; metadata + `production` aliases in Postgres (`@effect/sql-pg`) with swappable Effect services and in-memory test layers; typed REST + SSE API (`GET /api/projects`, `/api/projects/:project/deploy`, `/api/deploy/:id`, `POST /api/deploy/:id/promote|redploy`, `POST /api/deploy`, `GET /api/deploy/:id/logs|events` as `text/event-stream`); Effect CLI (`deploy`, `list`, `promote`, `logs`, `serve`); 42 Vitest tests.
- Frontend (Next.js 16 App Router + React 19, shadcn/ui, Tailwind CSS 4, Motion): thin routes (`/`, `/deploys/[deployId]`, `/new`) mounting feature modules; all server state in TanStack Query v5 only (`queryOptions`, `portalKeys`, `staleTime: 30s`) via typed fetch client + Next.js proxy handlers (`app/api/*` → `PORTAL/api/*`); live status/logs over SSE (`useDeployEvents/useBuildLogStream`) in a terminal-chrome viewer with rAF auto-scroll, optimistic mutations with rollback.

CI Quality-Gate Bot — ai-code-review-bot / bot-review github.com/Falasefemi2/bot-review

- Designed an automated CI quality-gate bot running 18 deterministic checks (typecheck, lint/format, coverage gate, gitleaks secret scan, Semgrep SAST, dependency audits) plus advisory LLM review, shipped as a reusable GitHub Action on Effect-TS v4.

daily-report / golang-update — Privacy-First Productivity Daemon github.com/Falasefemi2/daily-report · github.com/Falasefemi2/golang-update

- Designed a privacy-first daemon/CLI that tracks app usage and generates a daily Groq-LLM narrative with data staying on-device; implemented twice — in Effect-TS/Bun and Go (only dependency: `pgx`) — with cross-platform window sampling and jittered retries, unit-tested in both languages.

file-storage — Google Drive Clone file-storage-plum.vercel.app · github.com/Falasefemi2/file-storage

- Built a Google Drive–style file storage app in Next.js with Clerk authentication, file/folder management, UploadThing uploads, and Drizzle ORM over Postgres; styled with Tailwind CSS and deployed on Vercel.
-

Education

BSc Mathematics — Lagos State University | Sep 2019 – Mar 2024