Omri Ben Zvi

Selected Projects / Agent-Assisted Engineering

01 / PRODUCT DEVELOPMENT

Profile

I work in information security, with a background in communications infrastructure and military leadership. Alongside my professional work, I explore how AI coding agents can turn ideas into working products, practical tools, and useful open-source contributions.

These projects were implemented with coding agents, primarily Codex, with Claude Code also used. My role centers on defining goals, investigating options, guiding implementation, checking behavior, and refining the systems as I learn.

Hachozeh

Hebrew-language prediction market platform

My first substantial project: an Israeli prediction market using a closed virtual currency. It became a practical learning environment for architecture, testing, security, and deployment, from the initial idea through a public cloud deployment.

The platform includes market discovery, trading, portfolios, authentication, and community features. Its architecture separates transaction processing from market creation, evidence collection, and outcome resolution.

I guided the project through a move to Astro server-side rendering, Preact components, and a Fastify backend. Deployment work included managed PostgreSQL, domain configuration, monitoring, and documented recovery checks.

Agent continuity was a central design goal. Entry guides, task records, and explicit subsystem boundaries help another agent understand the project and continue working. In its maintenance phase, I delegate periodic checks and fixes to an agent.

hachozeh.com · GitHub / Source snapshot

Contact

omribz156@gmail.com

omribz.me

LinkedIn / omri-ben-zvi

GitHub / omribz156

System structure

Trading core

Pricing, quotes, balances, ledger entries, and transaction integrity.

Seer

Source intake, market proposals, and creation drafts.

Oracle & Horizon

Evidence and resolution workflows, plus scheduled market closing.

Technology

TypeScript
Astro & Preact
Fastify & PostgreSQL
Docker & Render
Cloudflare

Current stage

Deployed; now in maintenance with agent-led checks and fixes.

02 / AGENT WORKFLOWS & RESEARCH SYSTEMS

OSS Supporter

Reusable open-source contribution workflows

I started OSS Supporter to direct unused AI capacity toward useful community work. Working with an agent, I developed a portable framework for finding suitable issues, implementing focused changes, verifying results, and following up with maintainers.

Three workflows cover scouting, contribution execution, and review follow-up. Screening rules reject inactive repositories, duplicate work, unsuitable issues, and projects that prohibit AI-assisted contributions.

Agents perform much of the ongoing execution. I guide project selection and refine the framework based on experience. Friends also tried it with their own agents and environments, providing feedback on portability.

GitHub / oss-supporter

Accepted contributions

60

merged pull requests

33

external repositories

Verified September 23, 2026. Contributions from my account; personal repositories excluded.

Selected examples

Checkmarx / KICS

Bounded file-analysis workers.

Vercel / Streamdown

Streaming Markdown emphasis fix.

Datadog / Saluki

Autodiscovery event delivery and backpressure.

Security Research Harness

Experimental research infrastructure for Codex

An experimental project exploring how structured tooling can support agent-led security research. The goal was to automate repeatable preparation and information gathering so Codex could focus on analyzing evidence and testing hypotheses.

Developed with coding agents, the system includes a Python control plane, an MCP interface, research-tool integrations, persistent campaign state, and evidence tracking. It incorporates authorization boundaries, reproducibility checks, and evaluation fixtures.

A key lesson concerned agent behavior: providing tools and structured workflows did not ensure that Codex would consistently use them. The project exposed the gap between building research infrastructure and integrating it into an agent's workflow.

GitHub / Source snapshot · Engineering case study

Research components

Python control plane
MCP interface
Tool orchestration
Evidence & campaign state
Reproducibility checks

Omri Ben Zvi

Selected Projects / Agent-Assisted Engineering

03 / PRACTICAL TOOLS & WEBSITES

Runtime Radar

Local development visibility for macOS

A compact Swift menu bar application and CLI, created in response to a need I encountered while building Hachozeh. It brings together listening ports, local services, and tmux-based agent sessions, grouped by project.

A shared detection core powers both interfaces. The tool has a published release and reads local runtime metadata without sending telemetry.

GitHub / RuntimeRadar

Alon Porat

Law office website delivered to production

A Hebrew, right-to-left website for a family member's law practice. I guided an agent through implementation and deployment, covering responsive presentation, contact-form integration, search metadata, security headers, and domain configuration.

aporat.co.il

OmriBZ.me

Bilingual personal website

My personal website, built by adapting an existing Astro-based foundation with coding-agent assistance. Work included content and layout customization, Hebrew and English routes, deployment configuration, and security headers. Upstream attribution is preserved.

omribz.me

GitHub / omribz.me

How I work

01 Define

Start with a practical need and clarify the intended outcome.

02 Investigate

Explore options and trade-offs with the agent.

03 Direct

Guide implementation and revisit decisions as the system grows.

04 Verify

Check behavior, question assumptions, and use agent review and tests.

05 Continue

Keep decisions, task context, and handoffs in the repository.

Reusable practices

I turn lessons from projects into reusable agent instructions for session handoffs, focused code cleanup, browser verification, and project context.

GitHub / vault

Primary tools

Codex & Claude Code