

MOHIT SAGAR

+91-9555424388 | mohitsagar378@gmail.com | [LinkedIn](#) | [GitHub](#)

EXPERIENCE

Hyperjump JSON Schema ([hyperjump-io/json-schema](#))

Oct 2026 – Present

Open Source Contributor

Remote

- Found and reported (issue #133, reproduced on `main` and npm v1.18.0) that this JavaScript validator (~366K weekly npm downloads) rejected an invalid schema only on its *first* validation; later calls silently used it or crashed with a `TypeError`.
- Fixed the root cause in [merged PR #134](#): kept the “validated” flag set *before* checking, because a meta-schema validates against itself (setting it afterwards crashed Node with infinite recursion), and cleared it on any load, compile or validation failure.
- Added 2 Vitest regression tests that fail on `main` (full suite: 15,001 passing; lint and `tsc` clean) and flagged a remaining race between concurrent `validate` calls, proposing a promise-based fix instead of widening the change unasked.

Internet Archive – Open Library ([internetarchive/openlibrary](#))

Sep 2026 – Present

Open Source Contributor

Remote

- Rebuilt the author-merge confirmation in [merged PR #13751](#) so librarians see exactly which record is kept and which are redirected into it (with record keys and work counts), replacing a generic sentence that forced them to read the page hidden behind the dialog.
- Shipped it in JavaScript and web.py templates with no backend change, reusing values the server had already rendered and localised (`gettext`); extended the shared `confirmFromTemplate` helper with backward-compatible overrides; stopped empty merges client-side, before the server round-trip.

PROJECTS

PRism – Precision-First AI Code Reviewer (GitHub Action + CLI)

Tech: TypeScript, Node.js, GitHub Actions, Octokit, OpenRouter, Zod, js-tiktoken, Vitest

- Built a pull-request reviewer that targets the false-positive noise that makes developers ignore AI review: one model proposes defects, a second (by default from a different provider) tries to **refute** each, and only survivors of severity and confidence gates are posted.
- Treated PR content as untrusted input: the workflow checks out only the base branch, never installs or executes PR head code, runs from a prebuilt `ncc` bundle, and holds least-privilege tokens (`contents: read, pull-requests: write`).
- Packed diffs into each model’s context window using `c1100k` token estimation, with OpenRouter model-fallback chains, bounded concurrency (3 diff groups, 4 verify calls in flight) and per-response cost accounting from provider usage data.
- Posted findings as inline comments anchored to changed lines, fell back to a summary comment when the GitHub API rejected an anchor, and de-duplicated across pushes with SHA-256 fingerprints.
- Wrote **87 Vitest tests** and a token-free plumbing harness; in a field test on a file with 3 planted bugs, it anchored **3/3** comments to the correct line and found 1 real bug I had not planted; recorded weak cross-push de-duplication (1/3) as the next fix.

CodeFang – Coding Agent with Approval-Gated Edits

Tech: TypeScript, Bun, Vercel AI SDK (tool-calling agent loop), OpenRouter, Telegraf, Zod

- Built a terminal coding agent that cannot touch the disk without consent: file writes and deletes are staged in an in-memory overlay and shell commands are queued; the user reviews a unified diff and approves changes individually or together.
- Designed one shared tool executor behind Agent, read-only Plan and Ask modes plus a Telegram bot (single-owner auth), so the agent reads back its own staged edits and Plan mode hands off only user-selected steps for execution.
- Enforced a workspace jail on every path, blocked `.env*`, `.git`, `node_modules`, build output and logs, and capped reads at 1 MB (checked with `stat` before reading); covered the executor, review grouping and streaming with **81 passing Bun tests**.

EDUCATION

Scaler School of Technology

Aug 2024 – Present

Undergraduate Program in Computer Science (Expected 2028)

Bengaluru, India

- CGR: **7.94**

BITS Pilani (Online Program)

Aug 2024 – Present

Bachelor of Science (B.Sc.) (Expected 2027)

Remote

- CGPA: **6.89**

TECHNICAL SKILLS

Languages	TypeScript, JavaScript, Python, SQL; working knowledge of Go and Java
Backend & Frameworks	Node.js, Bun, Express, REST APIs, Server-Sent Events, Vercel AI SDK
Cloud & DevOps	GitHub Actions, Docker, Firebase
Databases	PostgreSQL, Firestore, Pinecone (vector database)
AI & Data	LLM tool-calling agents, OpenRouter model fallback, Zod structured output, RAG
Core Concepts	Data structures & algorithms, OOP & design patterns, HTTP, concurrency, secure input handling
Tools	Git, Vitest, Bun test, GitHub REST API (Octokit), Claude Code, Cursor