# HIMANSH RAJ

Applied AI Engineer | Agent Platforms, Inference & Systems

iamthehimansh@gmail.com | +91 6200483104 | Noida, India

himansh.in | github.com/iamthehimansh | linkedin.com/in/iamthehimansh

## Education

**IILM University, Greater Noida** 2023 – Expected 2027

B.Tech, Computer Science & Engineering (Artificial Intelligence & Machine Learning)

## Experience

**Colate | Applied AI Engineer** Jul 2025 – Present

Cana: agent platform spanning desktop, web and mobile | cana.build

- Designed a shared-IR LLM gateway across OpenAI, Anthropic and Gemini; added 108 unit/integration tests for streaming, tool calls and protocol compatibility, plus prompt caching and token accounting.
- Built a Python agent runtime with dependency-aware orchestration, batched tool execution and approvals, integrated with React and Rust/Tauri; added unit and end-to-end regression tests.
- Implemented gVisor-isolated cloud execution and a Redis-backed Kubernetes autoscaler; added persistent workspaces and sequence/ACK event replay to recover missed progress after disconnects.
- Shipped desktop/cloud handoff and phone/web remote control, including streamed history, reconnect reconciliation and permissioned agent actions.
- Added per-turn filesystem checkpoints and conversation rewind to undo agent edits when users revise earlier instructions.

**Indian Institute of Technology Delhi | Research Intern** May 2026 – Aug 2026

Machine Intelligence, Signals & Networks Lab | Advisor: Dr. Sandeep Kumar

- Built agentic recommendation pipelines combining retrieval, planning and tool use; developed evaluation methodology for failure modes and robustness.

## Open Source Contributions

- **Microsoft Semantic Kernel**: fixed empty tool results becoming `None` during XML prompt round trips; added regression tests. Merged PR #14686.
- **Google ADK**: made Agent Engine tests portable to macOS using GNU coreutils, resolving 13 failing test cases. Integrated upstream via Copybara.
- **AnyPS5**: six merged upstream PRs improving macOS builds, Mach-O signing layout, pooled-buffer handling, sleep timing and graphics tests. Merged contributions.

## Selected Product

**Cana Desktop, Web & Mobile** (Colate) Product walkthroughs

Rebuilt the React Native chat client for iOS/Android with streamed tool activity, remote approvals, attachments and notification deep links.

- Extended the typed Mods SDK with commands, shortcuts, MCP tools, model presets and scoped permissions; added multi-profile Chrome companion pairing.
- Integrated live meeting transcription across Python, Rust and the UI; added Windows executable signing to the release pipeline.

## HIMANSH RAJ

Engineering projects and research

### Open Source Projects

**Maya OS / Mayac** | TypeScript, Python, SQLite, MCP — Source / Agent

Built local llama.cpp inference, cloud model routing and PTY-backed agent teams; implemented permission-gated tools and persistent sessions.

- Built a provider-agnostic coding-agent loop with streaming tool execution, context compaction, interruption and session recovery; exposed it through a full-screen terminal interface.
- Added an independent workspace evaluator that checks generated files against task expectations and reports false-success claims separately from completed work.

**Kestrel** | Swift, Metal, gRPC — Source

Built a native Apple Silicon player around Google's Android Emulator with authenticated frame streaming, editable keymaps, bounded GPU uploads and nonblocking input delivery.

- Added emulator instance management, snapshots and keyboard/mouse mapping with a live editor; implemented regression coverage for session, input and rendering behavior.

**LlamaVision** | PyTorch, Transformers — Source

Trained a 6.3M-parameter, two-layer projector connecting frozen CLIP ViT-L/14 and Llama 3.2-1B for image-conditioned text generation; published model and inference code.

**H++** | C, Assembly, Compiler Design — Source

Implemented a statically typed language compiler with lexing, parsing, semantic checks, macros, modules and x86-64 NASM code generation; built an assembly-backed standard library.

**F75** | Python, PyTorch — Source

Implemented and trained an eight-layer character-level transformer from scratch, including tokenization, attention, the training loop and sampling; compared dropout and no-dropout runs.

**boxctl / BoxDeck** | Python, Swift, SSH, MCP — Source

Built a CLI and native macOS companion for a remote GPU workstation, with LAN/remote routing, tunnel and service management, live telemetry and an MCP interface.

### Research

**Low-resource machine translation: English / Prakrit**

Trained translation models and built synthetic-data generation and filtering workflows, including beam-candidate agreement filtering and CTranslate2 inference. Research collaboration with a PhD researcher at IIT Delhi; manuscript in preparation.

### Technical Skills

**Languages:** Python, TypeScript/JavaScript, C/C++, SQL, Rust, Swift

**AI & systems:** PyTorch, Hugging Face Transformers, MCP, RAG, llama.cpp, React, React Native, FastAPI, Node.js, Docker, Kubernetes, AWS, Redis, PostgreSQL, SQLite, WebSockets, Git, pytest

### Awards

**Winner**: HIVE, IIT Delhi (2024); FinAThon, IIT Delhi (2023).

**Top 10 internationally**: HERE Hackathon, IIT Bombay TechFest (2023).