

# Jeonghun Kim

ML Engineer / AI Researcher

portfolio.jhkim3ispark.work



Software Engineer, Machine Learning Engineer (Agentic AI)

bandlagom0927@gmail.com

+82 10-9559-9471 · Seoul

github.com/thuygom



## Cover Letter

---

Dear FuriosaAI Hiring Team,

I am applying for the Software Engineer, Machine Learning Engineer (Agentic AI) position (Job ID 4005371201). I bring direct experience building the parts an agent needs beyond generation: validated tool execution, reusable context, and recoverable jobs. At ISPARK, I develop TieMing's core AI features and take responsibility for their implementation through deployment. Before ISPARK, I served one year as an information-systems NCO in the Republic of Korea Air Force (Dec 2021–Dec 2022), maintaining Linux servers through shell scripting and log analysis.

I built a system that turns selected coding-session records into searchable knowledge and reusable wikis. Engineering decisions were scattered across sessions, so I preserved links to original records and added incremental ingestion instead of repeatedly collecting the same history. Tenant-, department-, and user-level access controls determine which records are visible. I also delivered an installer and developer-tool integration, taking the work beyond a local prototype into a workflow another user could adopt.

I built a CRM query workflow covering schema retrieval, SQL validation, and controlled execution. I addressed incorrect table selection by enriching database context and combining semantic and keyword retrieval. I enforced read-only transactions and query limits, then designed and validated PostgreSQL Row Level Security (RLS) to restrict each user's row access. I also checked that document permission changes altered the evidence available to the answer. Execution checks and data access controls are part of the system, independently of the model's output.

At FuriosaAI, I can contribute to agent execution and memory through permission-aware tools, traceable context, and explicit job state. I would use real workflows to validate how these components behave together, then turn recurring failure cases into focused improvements.

Sincerely,

Jeonghun Kim