CampusVoice

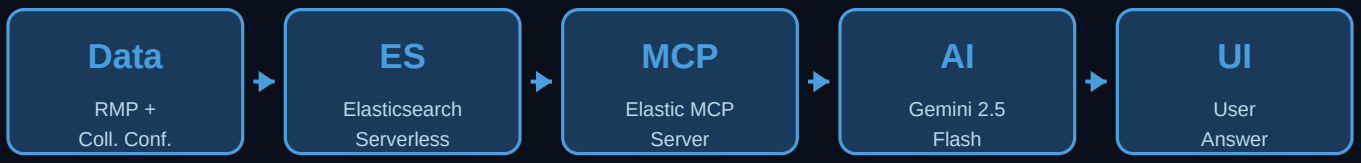
Tech Stack & Architecture

AI-powered student sentiment agent | Google Cloud Rapid Agent Hackathon 2026

What It Does

CampusVoice lets students, counselors, and administrators ask plain-English questions about university professors and courses — and get synthesized answers with real quotes drawn from **13,192 student reviews** — 6,773 from Rate My Professors and 6,419 from College Confidential — across 7 US universities.

Data Flow Pipeline



A user question triggers Gemini 2.5 Flash, which calls the Elastic MCP Server as a tool. The MCP server queries Elasticsearch and returns matching reviews. Gemini synthesizes a final answer with verbatim quotes — all in a single agentic loop.

Tech Stack

Layer	Technology
LLM	Google Gemini 2.5 Flash — reasoning, search planning, answer synthesis
Search & Storage	Elasticsearch Serverless on Elastic Cloud — vector + keyword search
Tool Layer	@elastic/mcp-server-elasticsearch — MCP server over stdio JSON-RPC
Agent Framework	Custom Python agent loop with Gemini function calling (no LangChain)
Web Framework	Flask + Gunicorn — REST API serving the HTML/JS frontend
Deployment	Google Cloud Run — containerized, auto-scales to zero, 120s timeout
CI/CD	Google Cloud Build — auto-deploys from GitHub main on every push
Data Collection	Rate My Professors (GraphQL API) + College Confidential (Discourse API) — 13,192 total

Container	Docker — Python 3.11-slim + Node.js 20 + Elastic MCP server bundled

Agent Loop (per request)

Step	What happens
1. Receive	Flask receives POST /api/ask with question + optional school filter
2. Prompt	Gemini receives question + full search strategy in system prompt
3. Tool call	Gemini generates an Elasticsearch query via function calling
4. Execute	MCPClient sends tools/call to the Elastic MCP subprocess over stdio
5. Results	Elasticsearch returns matching reviews; MCP passes text back to Gemini
6. Synthesize	Gemini reads review comments, identifies patterns, quotes verbatim
7. Respond	Final answer returned as JSON and rendered in the chat UI

3-Tier Search Fallback

To ensure the agent never returns empty results, every search follows a 3-tier fallback strategy:

Tier	Query type	When used
1 - Primary	school_tag term + keyword	First attempt for every question
2 - Fallback	school name match + keyword	If tier 1 returns 0 hits
3 - Broad	Keyword only, no school filter	If tier 2 also returns 0 hits

Universities Covered

Tag	University	Reviews
utk	University of Tennessee Knoxville	~950
vanderbilt	Vanderbilt University	~850
gatech	Georgia Institute of Technology	~1,100
uf	University of Florida	~1,200
umich	University of Michigan	~1,000
ucla	UCLA	~850

duke	Duke University	~820

Live: https://campus-voice-agent-420887396772.us-east1.run.app | **GitHub:** https://github.com/srivi19/campus-voice-agent | **Devpost:** https://devpost.com/software/campusvoice