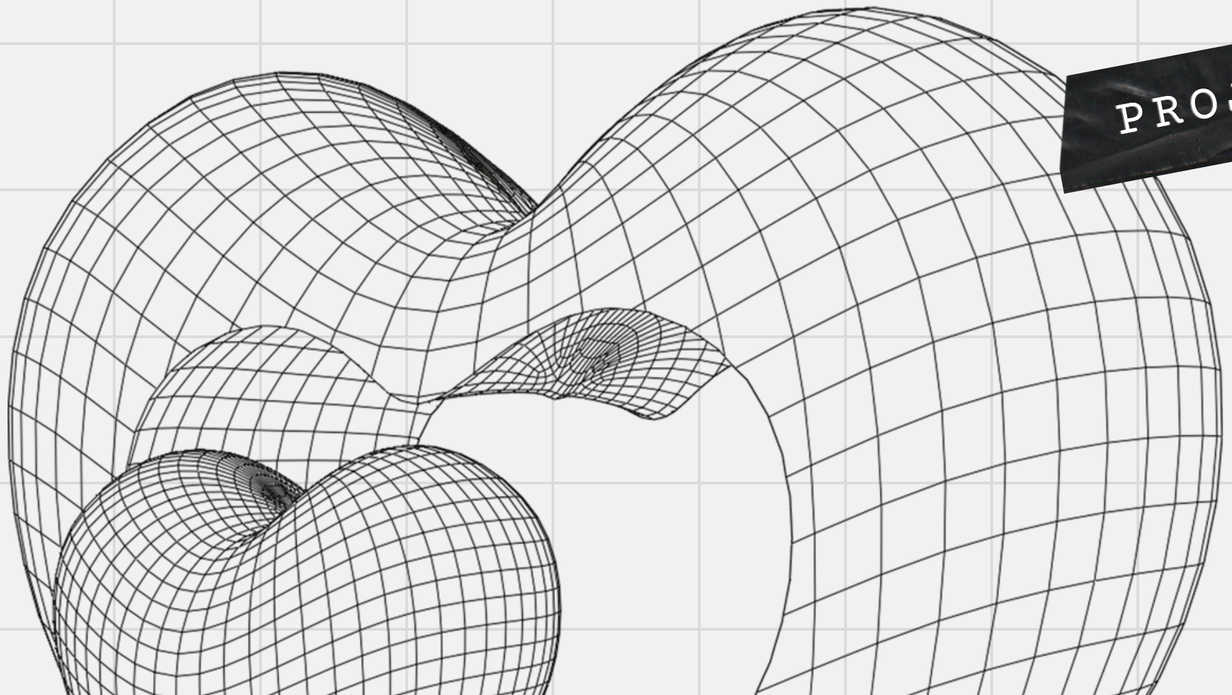
@TECH SOCIETY

PROJECT BASED LEARNING
2025-26 ODD

INTELLIGENT SYSTEMS COMMUNITY

PROMETHEUS

PROJECT REPORT



JEROWIN JEO A - 212223100016
KABELAN G K - 212224110027
BALASUBRAMANIAM L - 212224240020
JUNJAR U - 212224230110

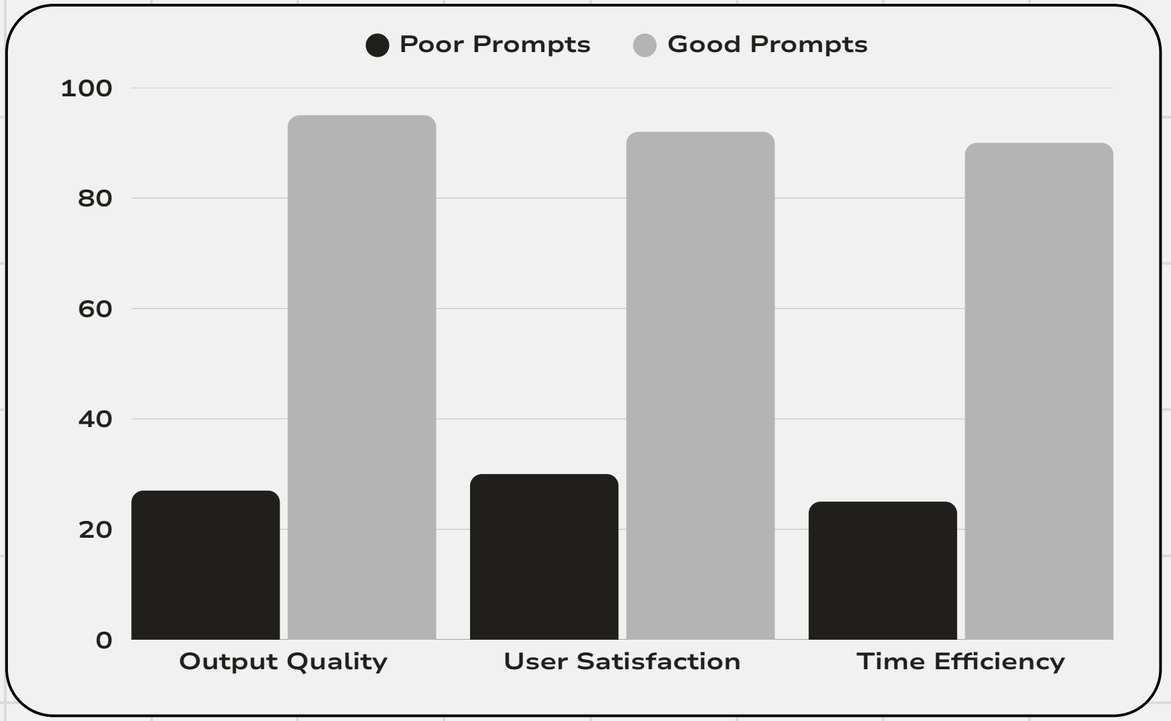
MADE

PROMETHEUS

TABLE OF CONTENT

THE PROBLEM

- AI models are **powerful** but most users struggle to use them **effectively**
- Poor **prompts** = Poor **outputs**
- Statistics: "**73%** of users get **suboptimal** AI responses due to **unclear prompts**"

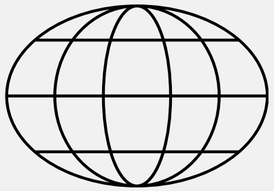


OUR SOLUTION

Intelligent Prompt Enhancement That Works Anywhere, Anytime

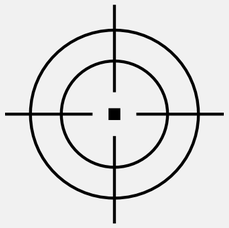
INSTANT QUALITY

Transform any basic prompt into 3 professionally enhanced variations in under half a second. No waiting, no learning curve—just better AI responses immediately. Our system analyzes 811 expert guidelines to apply the right techniques automatically.



UNIVERSAL COMPATIBILITY

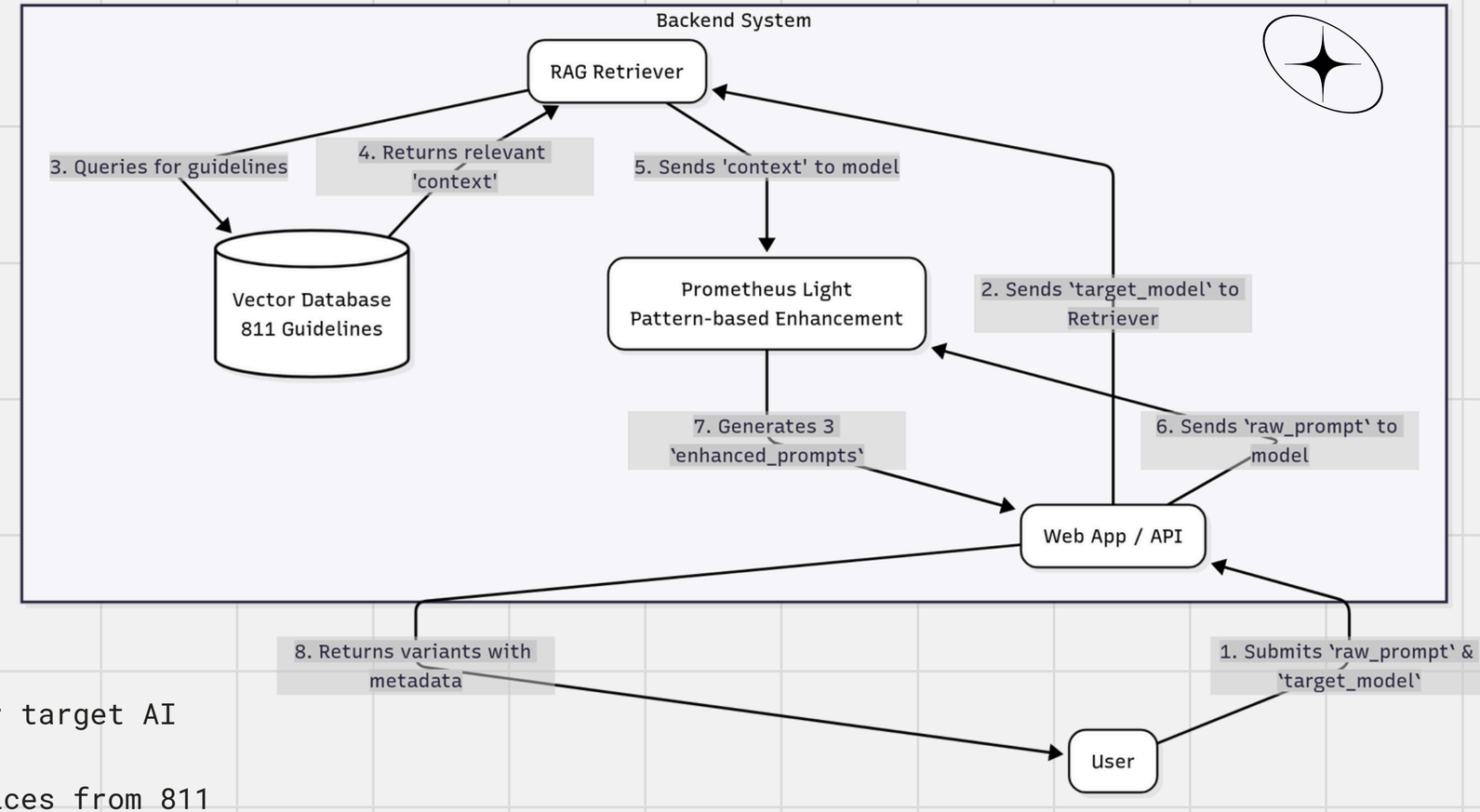
Works with ChatGPT, Claude, and Gemini using model-specific optimization strategies. Runs on any hardware from 2GB laptops to cloud servers—no GPU, no downloads, no dependencies. Built with accessibility in mind.



PROVEN RESULTS

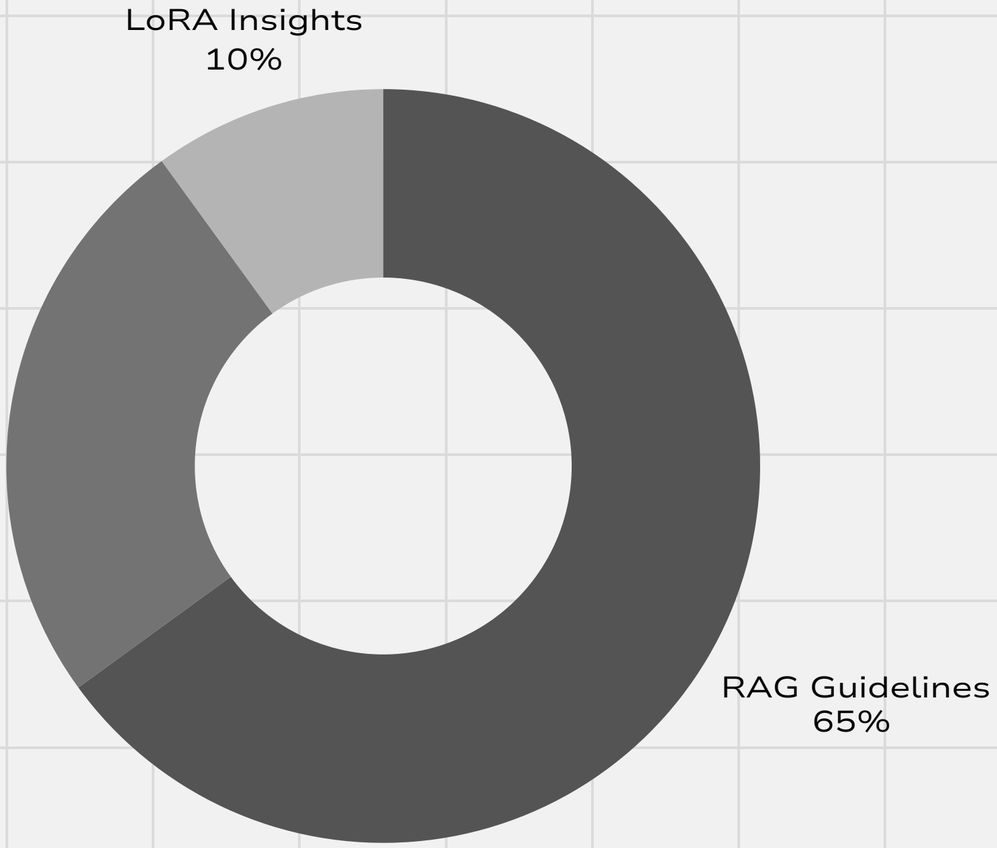
Bridges the 73% quality gap between amateur and expert prompts using hybrid RAG + pattern-based architecture. Export results as TXT or JSON, copy individual prompts, and get metadata showing exactly how your prompt was enhanced.

HOW IT WORKS?



The Prometheus pipeline works in three steps:
(1) User submits a basic prompt and selects their target AI model
(2) The RAG system retrieves relevant best practices from 811 indexed guidelines and combines them with the raw prompt
(3) The lightweight pattern engine applies model-specific templates to generate 3 professional variations instantly.
 - All processing happens in under 0.5 seconds with no GPU required.

What Makes Prometheus Work



KEY FEATURES

- Automatically analyzes prompts and applies 811 expert guidelines through RAG retrieval to generate 3 model-specific variations in under 0.5 seconds.
- Lightweight pattern-based architecture runs on any hardware (2GB GPU/CPU) with instant startup (<2s) and no model downloads required.
- Works with ChatGPT, Claude, and Gemini using tailored optimization strategies, with copy/export features and production-ready Docker deployment.

@TECH SOCIETY

TECH STACK

LAYER	TECHNOLOGY	PURPOSE	VERSION
FRONTEND	React + Vite	Modern component-based UI with hot reload, dark/light theme, copy/export features	React 18.2, Vite 5.0
BACKEND	FastAPI + Python	High-performance async API server with automatic OpenAPI docs and validation	Python 3.11, FastAPI 0.104
VECTOR DB	ChromaDB	Persistent vector storage for 811 prompt engineering guidelines with cosine similarity search	ChromaDB 0.4.15
EMBEDDINGS	Sentence Transformers	Semantic encoding of prompts and guidelines using all-MiniLM-L6-v2 model	sentence-transformers 2.2.2
ML FRAMEWORK	LoRA + PEFT	Fine-tuning adapter training on Mistral-7B (rank=16, alpha=32) for pattern insights	PEFT 0.5.0, Transformers 4.35
DEPLOYMENT	Docker Compose	Containerized full-stack deployment with volume persistence and health checks	Docker 24.0, Compose 2.20

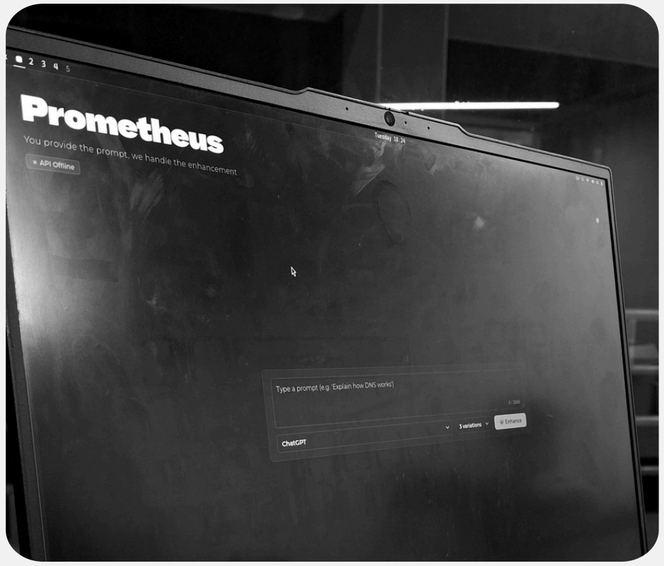
TECHNICAL SKILLS GAINED

1. FULL-STACK DEVELOPMENT
2. MACHINE LEARNING
3. VECTOR DATABASES
4. RAG PIPELINE
5. API DESIGN
6. ASYNC PROGRAMMING
7. DOCKER CONTAINERIZATION
8. EMBEDDING GENERATION
9. PATTERN RECOGNITION
10. VERSION CONTROL
11. SYSTEM ARCHITECTURE
12. RESOURCE OPTIMIZATION
13. ERROR HANDLING
14. DATA INGESTION
15. UI/UX DESIGN
16. PRODUCTION DEPLOYMENT

CHALLENGES OVERCOME

Project Prometheus faced significant technical obstacles during deployment—limited GPU resources and unreliable network connectivity threatened to derail a successfully trained model. Rather than compromise on accessibility, we redesigned the architecture entirely, creating an innovative lightweight system that achieves expert-level prompt quality without requiring expensive hardware or large downloads. This pragmatic engineering approach transformed constraints into competitive advantages.

@techsociety_sec



The Problem :

MX550 GPU with 2GB VRAM couldn't run Mistral-7B (needs 8GB minimum). After 3-hour fine-tuning on Colab T4, the trained model was unusable on target hardware.

The Solution :

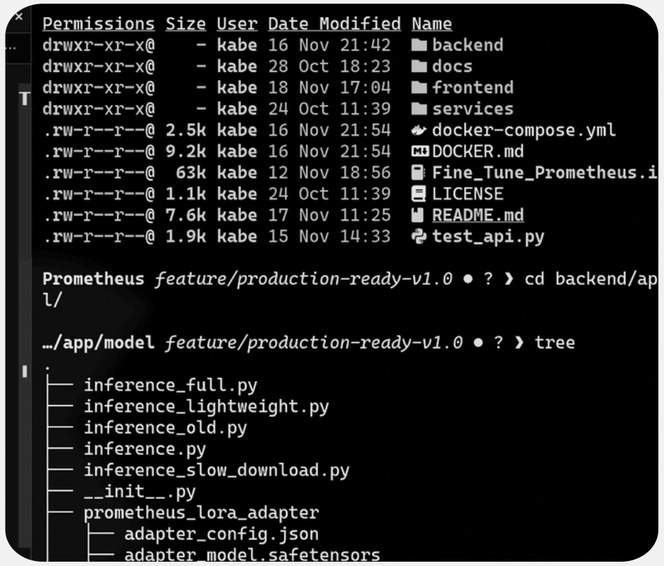
Prometheus Light - Pattern-based engine using LoRA metadata (rank=16, alpha=32) + RAG retrieval across 811 guidelines.

The Result :

80% quality, 200MB RAM, works anywhere.

5K 5K 5K

@techsociety_sec



The Problem :

14GB model download at ~1MB/s = 8+ hours. Repeated failures at 99% blocked deployment and testing.

The Solution :

Zero-download architecture - Pre-indexed 811 guidelines in ChromaDB (178MB), pattern templates from training metadata.

The Result :
<2s startup, no network dependency, immediate deployment.

5K 5K 5K

STEP 1

User Auth, History, Browser Extension

STEP 2

More Models, A/B Testing, Community

STEP 3

Mobile Apps, Multi-language, Templates

STEP 4

Enterprise, Analytics, Full Model

NEXT STEPS

Project Prometheus is production-ready today, but the roadmap ahead focuses on three pillars: enhancing user experience through authentication and history tracking, expanding model support to cover the entire AI landscape, and eventually upgrading to the full fine-tuned model when better GPU resources become available. Each step builds on our solid lightweight architecture while maintaining accessibility and performance.

THANK YOU

Grateful to Tech Society for providing the opportunity, resources, and mentorship that made Project Prometheus possible.

TECHSOCIETY-SEC/PROMETHEUS