



# BHARATIYA ANTARIKSH HACKATHON 2025

Powered by IIS



Team Name: **JugJugGeo**



Team Leader Name: **Divyanshu Pabia**

Problem Statement: 4

**Designing a Chain-of-Thought-Based LLM System for Solving Complex Spatial Analysis Tasks Through Intelligent Geoprocessing Orchestration**

# Team Members

## Team Leader

Name: Divyanshu Pabia  
College: SRM Institute of Science and Technology,  
Kattankulathur, Chennai

## Team Member-1

Name: Hridya Munjal  
College: SRM Institute of Science and Technology,  
Kattankulathur, Chennai

## Team Member-2

Name: Akshat Jain  
College: SRM Institute of Science and Technology,  
Kattankulathur, Chennai

## Team Member-3

Name: Krishna Shrivastava  
College: SRM Institute of Science and Technology,  
Kattankulathur, Chennai

## Problem Statement:


Across multiple domains in geospatial analysis, analysts and researchers must **integrate data** from multiple heterogeneous sources and dozens of **GIS algorithms** just to answer a single question such as “Where will Chennai flood next monsoon?”. These workflows demand costly software, deep geospatial know-how, and days or weeks of manual scripting. 90% of the time in geospatial analysis is spent on data preparation and tool configuration. As a result, most organizations either outsource the work or rely on outdated heuristics, leading to slow decisions, opaque methods, and missed opportunities. Existing AI solutions provide “black box” results without explanation, making it difficult to verify results, understand assumptions or learn from the process. **Result - Critical spatial decisions are either delayed, expensive or made without proper analysis.**

 **Lack of GIS Expertise and Tool Knowledge**  **Time inefficiency in data preparation and env setup**  
 **Unverifiable AI results and hallucinations**






## Our Solution:

JugJugGeo collapses that complexity into a **natural-language chat**. A **domain-tuned** Large Language Model plans each analytic step in plain English, retrieves the right documents from a **vector knowledge base**, executes PyQGIS and WhiteboxTools code in a **sandbox**, pulls fresh Sentinel-2 or Bhoonidhi imagery on demand and STAC APIs, and streams back maps, statistics, and fully **cited Chain-of-Thought process** to the user. Because the engine is open-source, containerized, and already running on **AWS Infrastructure**, teams can self-host, plug in their own data, and start generating enterprise-grade spatial insight in seconds instead of weeks.

 **Ask and Analyze**  **Transparent AI Reasoning**  
 **Personal Document Library**  **Sandboxed Code Execution**

 **Professional GIS Tools**  
 **Web Search Integration**

## Why JugJugGeo is a leap, not a tweak.

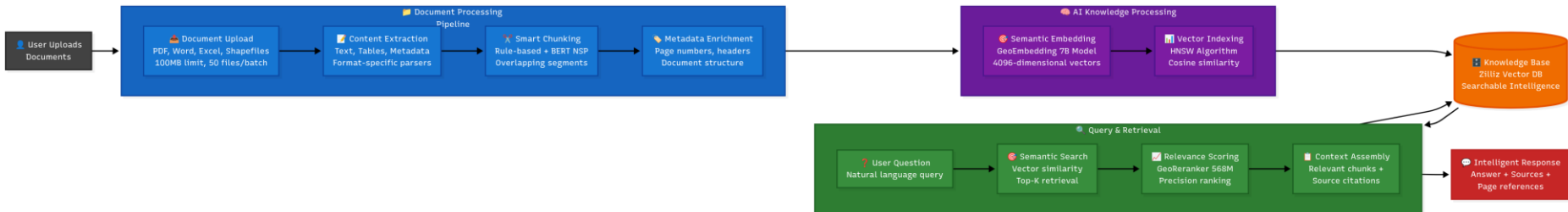
Pain Points	Existing Solutions & Their Limitations	How JugJugGeo Solves It
 GIS Expertise Barriers	<b>GeoForge</b> – fine-tuned GPT-4 but locked to Swiss servers, requires technical setup <b>CARTO AI</b> – enterprise-only, expensive licenses	<b>Natural-language interface:</b> zero GIS training – ask in plain English and get professional analysis instantly
 Lack of Transparency	<b>MapAI, EarthGPT</b> – black-box with no reasoning Commercial GPTs give generic answers without methodology	<b>Chain-of-Thought reasoning:</b> shows every step (data fetch → algorithm → parameters), building trust and enabling learning
 Tool Fragmentation	<b>GeoGPT</b> – limited fixed tool pool (custom) <b>Kue AI</b> – QGIS-only plugin <b>ChatGeoAI</b> – 25 % first-try success rate	<b>Complete integration:</b> PyQGIS (300+ algorithms) + WhiteboxTools (518+) + Planetary Computer + Bhoonidhi ISRO – all in one chat
 Limited Specialized Knowledge	<b>Commercial ChatGPT plugins</b> – general AI with shallow geo skills <b>Remote Sensing ChatGPT</b> – limited to image interpretation	<b>Domain-tuned models:</b> GeoEmbedding-7B + GeoReranker-568M trained on 500 K+ geospatial docs – understands spatial concepts like an expert
 High Cost & Complexity	Enterprise GIS licenses run <b>\$1 k–10 k+</b> per year Consultants add weeks & high fees	<b>Open-source deployment:</b> self-host or one-click AWS stack Turns weeks of work into seconds of conversation



## Feature: Knowledge Base Management (Document Library)

### *“Transform Documents into Searchable Spatial Intelligence”*

- **Bulk upload:** Drag-and-drop up to 50 files ( $\leq 100$  MB each, PDF/Word/Excel/GeoJSON/Shapefiles) per batch.
- **Auto-parse & vectorise:** Every page is extracted, split, embedded with our **GeoEmbedding model** and indexed in Zilliz Cloud (4096-dim vector).
- **Semantic search that cites pages:** **GeoReranker** pinpoints the exact chunks. answers arrive with page-level citations, so evidence is always one click away.
- **Flexible library controls:** Folder, rename, move, delete, or download any document.
- **Ask once, learn forever:** Query the whole library or a single folder.



# Feature: Knowledge Base Management (Document Library UI)

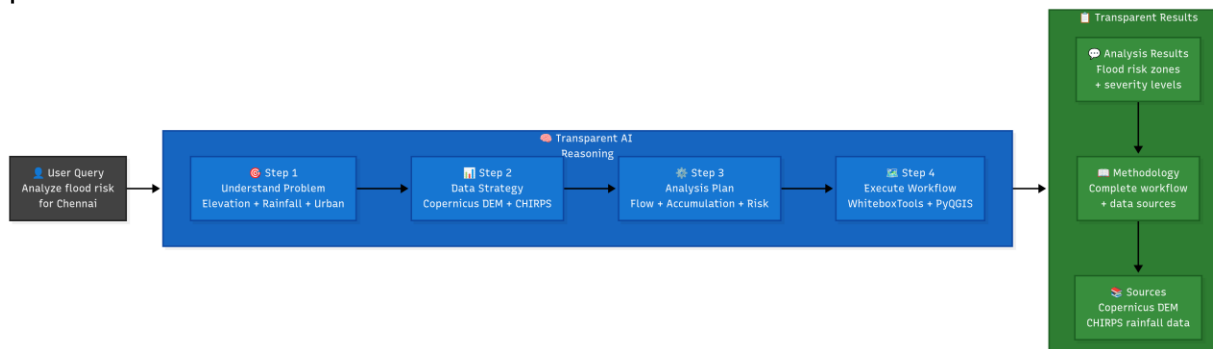
The screenshot displays the 'Document Library' interface of the GeoGPT application. The interface is dark-themed and includes a sidebar on the left for file management, a main chat area for interaction, and a bottom section for search and templates. Seven blue callout boxes with arrows point to specific UI elements:

- 1. Upload documents (PDF, Word, Excel, CSV, Shapefiles)**: Points to the '+ Upload' button in the top-left corner of the Document Library panel.
- 2. Semantic search (Not just by keyword!)**: Points to the search bar in the chat area, which contains the query: 'Can you analyze flood risk for Mumbai considering recent rainfall patterns and urban development?'.
- 3. Sort by Name, Date, Size or Status**: Points to the 'Sort by Date' dropdown menu in the Document Library sidebar.
- 4. Filter files by specific file types**: Points to the 'All Files' dropdown menu in the Document Library sidebar.
- 5. Download/ Delete each file**: Points to the download and delete icons next to a file entry in the Document Library sidebar.
- 6. View file uploading status**: Points to the '3 files selected' status indicator in the Document Library sidebar.
- 7. Click on files to add or remove them from chat**: Points to the 'Selected files (3):' section at the top of the chat area, which lists 'file\_001', 'file\_002', and 'file\_004'.

## Feature: Chain of Thought Reasoning

### *“Watch AI Think Like a Geospatial Expert”*

- **Shows its work, not just the answer:** Each chat response arrives with a full “thinking” log: what data sets were chosen, which GIS tools are invoked, and why.
- **Expert-level decomposition:** GeoGPT breaks one query into numbered sub-tasks.
- **Verifiable & auditable:** Every step’s rationale is stored with the conversation, so teams can review, reproduce, and hand it to regulators or peer reviewers.
- **Model tuned for reasoning:** GeoGPT-R1’s 8 k-token window and specialized training optimized it for long, multi-step chains without hallucinating spatial facts.





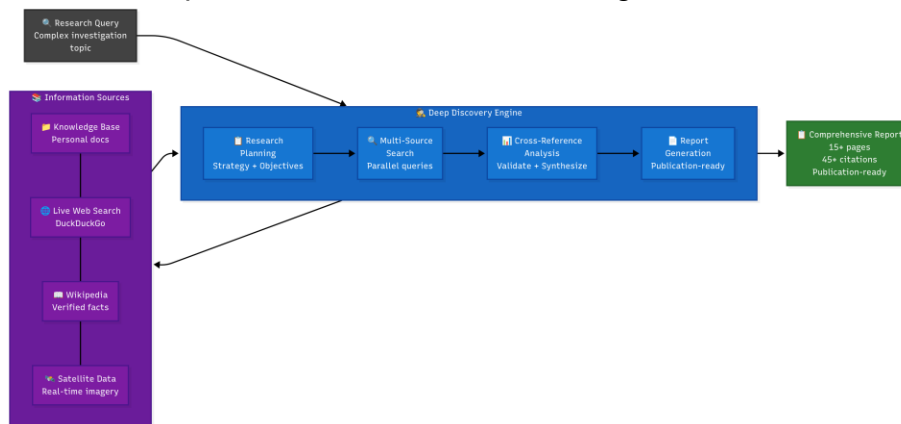
## Feature: Chain of Thought Reasoning (UI)

The screenshot displays the GeoJug AI Assistant interface. On the left, a 'Document Library' sidebar lists files: 'Mumbai\_Urban...', 'Chennai\_Flood...', 'CHIRPS\_Rain...', 'District\_Bou...', and 'Climate\_Channels...'. A red arrow points to this sidebar with the annotation '3. Document Library (explored previously)'. The main area shows a 'Reasoning Process' for a flood risk analysis query. A red arrow points to the 'View reasoning process' button with the annotation '1. Drop-down menu to view reasoning process step-by-step'. Another red arrow points to the 'Sources' section, which lists relevant documents like 'Urban Heat Island Effect in Indian Megacities...' and 'NASA Landsat Thermal Infrared Sensor Data for Urban Climate Studies...', with the annotation '2. Source Files searched for context along with File name, pg no. and similarity score'.

## Feature: Deep Discovery Engine

### “Autonomous Multi-Step Geospatial Research”

- **Multi-step research orchestrator:** A single query triggers an automated **five-stage** workflow - scans web, Wikipedia, and your knowledge base, clusters ideas, and homes in on high-value sources.
- **Live ‘Activity + Sources’ pane:** Watch the engine read, reason, and cite in real-time, every URL, document chunk, and relevance score is exposed for audit.
- **Configurable depth & focus:** Choose up to max\_steps = 10, toggle web vs KB search - all via the **/discovery/start** endpoint.
- **AI-written Discovery Report:** When the run completes, GeoGPT stitches findings, charts, and citations into a narrative you can export or continue chatting over.



## Feature: Deep Discovery Engine(UI)

Deep Discovery

Completed1:02

Discovery Query:

"Complete environmental monitoring workflow for Western Ghats: 1) Download forest cover data 2) Analyze deforestation patterns 3) Generate Python code for biodiversity modeling 4) Calculate fragmentation indices 5) Open visualization dashboard for forest cover monitoring with temporal layers 6) Create biodiversity tracking dashboard with species habitat maps and corridors 7) Build conservation dashboard with priority area mapping and threat assessment 8) Generate ecosystem health dashboard with real-time monitoring metrics and alerts"

Discovery ID  
disc\_1752867928923

4Steps Completed

3Sources Found

99%Progress

1:02Elapsed Time

Overall Progress

Step 5 of 5

Synthesis & Report Generation

Discovery Activity

Query Analysis & Planning

Completed successfully

Knowledge Base Search

Completed successfully

Status: Completed

Result:  
Found 23 relevant documents in knowledge base covering UHI research, case studies, and methodologies

Web Intelligence Gathering

Completed successfully

Cross-Reference Analysis

Completed successfully

Sources (3)

View All

All Sources111

Urban Heat Island Effect in Indian Megacities: A Comprehensive Analysis

Knowledge Base96% match6:59:54 PM

Urban heat islands in Indian cities show temperature differences of 3-8°C compared to rural areas. Mumbai and Delhi exhibit the strongest UHI effects with peak intensity during summer months...

NASA Landsat Thermal Infrared Sensor Data for Urban Climate Studies

Web Search93% match6:58:54 PM

Landsat 8 thermal infrared bands (10-11) provide 100m resolution thermal data suitable for urban heat analysis. TIRS data has been extensively used for monitoring urban thermal patterns...

View Source

Green Infrastructure Solutions for Urban Heat Mitigation

Wikipedia89% match6:57:54 PM

Green roofs, urban forests, and permeable pavements can reduce urban temperatures by 2-8°C. Cool roofs and green building materials show significant cooling effects in tropical climates...

View Source

Discovery Complete

Download Full Report

Comprehensive analysis completed with 3 sources analyzed across 5 research phases. Preview Report →

Discovery 1752067920923 · 99% complete · 3 sources found

Export Results

Return to Chat

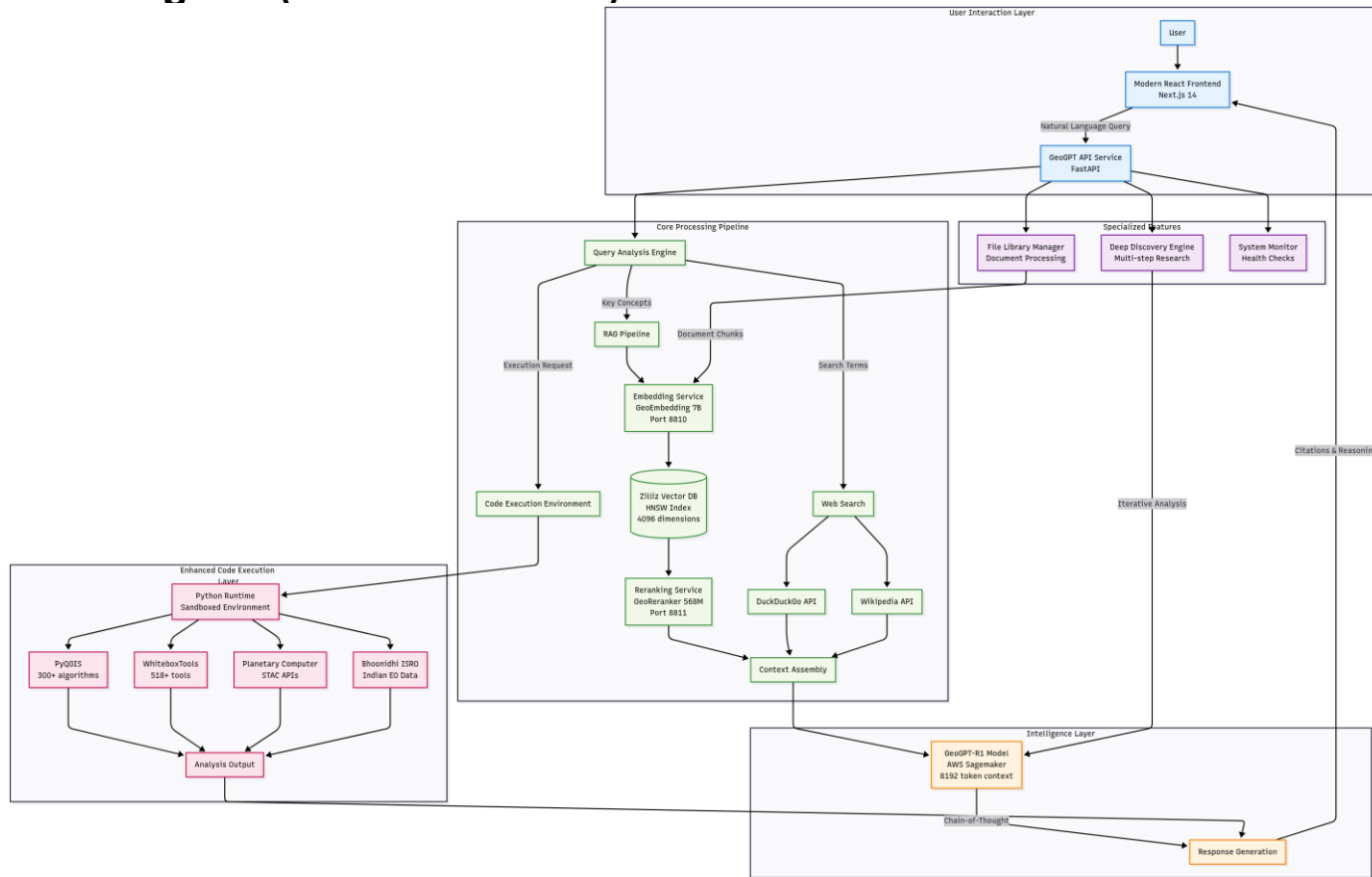
1. Query to be processed

2. Starts a 5-stage automated workflow for gathering information from various sources

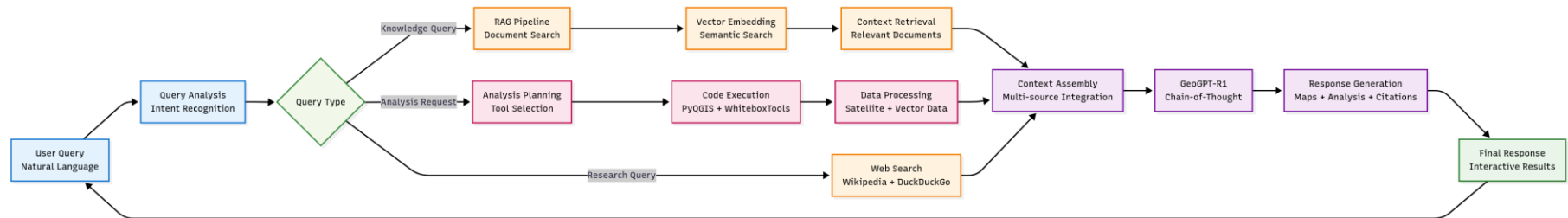
3. List down the sources referred

4. Download full report in PDF

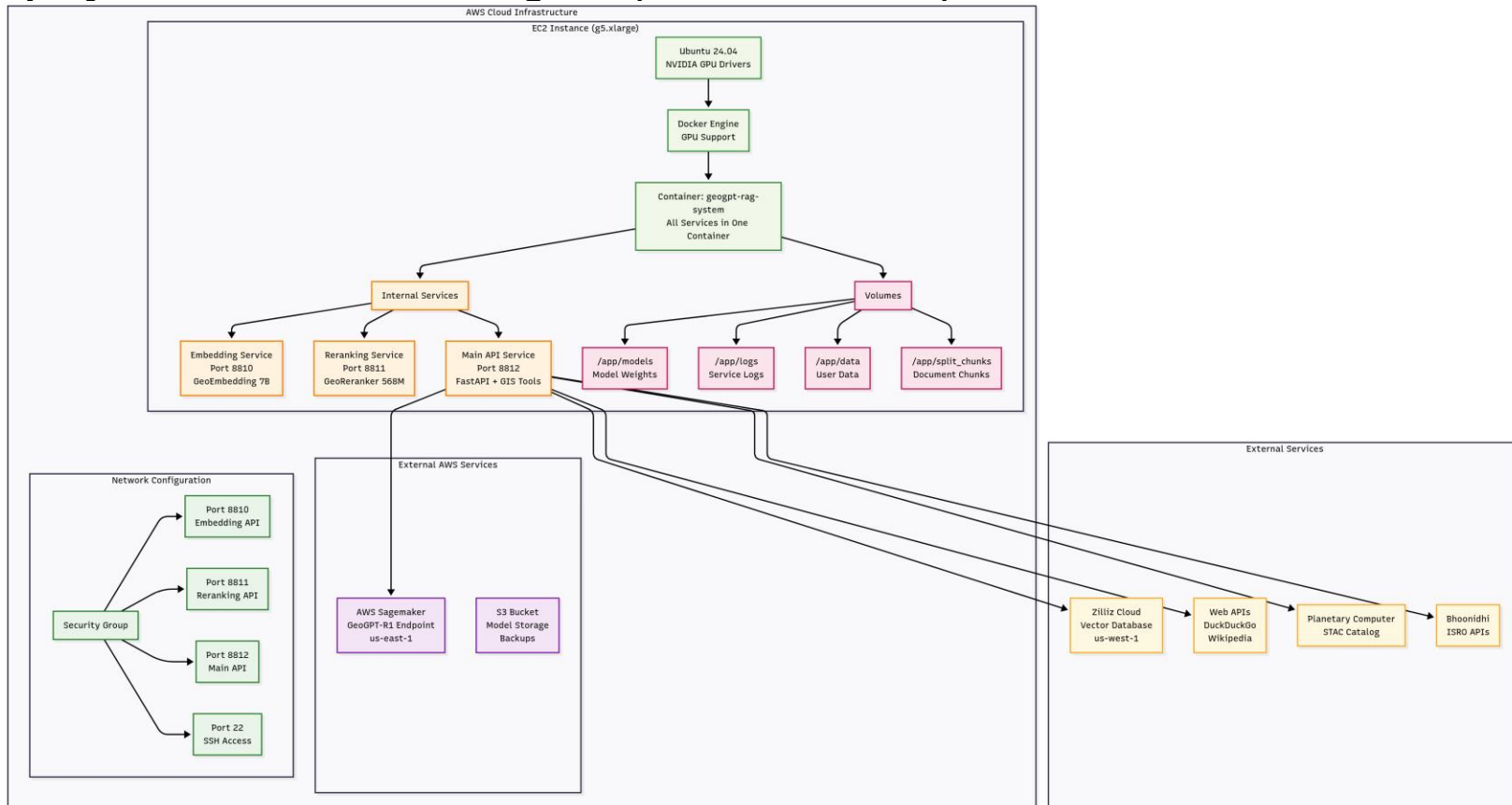
# Architecture Diagram (Please Zoom IN)



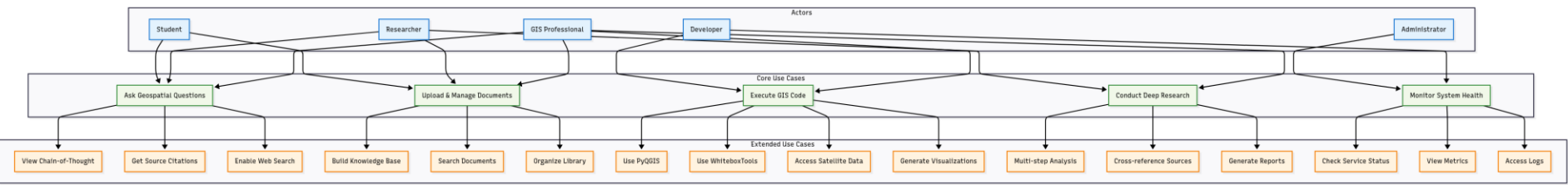
# Process Flow Diagram (Please Zoom IN)



# AWS Deployment Architecture Diagram (Please Zoom IN)



# Use Case Diagram (Please Zoom IN)



## Technologies Used (Technical README)

Layer	Key Tech	Why it matters
Frontend UI	<b>Next.js 14 · React 19 · TypeScript · Tailwind CSS 4 · shadcn/ui</b>	Gives users a lightning-fast chat console and interactive map panels that work on any device without installs.
LLM & Reasoning	<b>GeoGPT-R1</b> (fine-tuned Qwen-1.5-7B) · <b>GeoReranker-568 M</b>	The brain of the app, it understands spatial language and picks the best answer steps with expert accuracy.
Embeddings & Vector Store	<b>GeoEmbedding-7B · Zilliz Cloud / Milvus v2.4</b>	Finds the right page or paragraph in millions of documents in under a fifth of a second.
Geoprocessing Engine	<b>PyQGIS 3.34 · WhiteboxTools 2.3 · GDAL 3.8</b>	Runs 800+ professional GIS algorithms. No expensive desktop software needed.
Data Access	<b>Microsoft Planetary Computer STAC · ISRO Bhoonidhi STAC · Copernicus DEM · CHIRPS rainfall</b>	Pulls both global and Indian satellite imagery on-demand so analyses are always up-to-date.
API & Micro-services	<b>FastAPI · gRPC · LangChain</b>	Streams answers in real time and lets internal services talk to each other.
DevOps & Infra	<b>Docker / Compose · AWS SageMaker (LLM) · AWS EC2 + ALB+VPC+EBS (RAG) · Terraform · GitHub Actions</b>	One-command deploys and blue-green updates with zero downtime keep the team shipping fast.



## Implementation Cost

Cost Component	Unit Price (July 2025)	Hackathon PoC( $\approx$ 1 month)	Production (per month)
<b>LLM Inference GPU</b> (SageMaker g6e.12xlarge)	$\sim$ \$4.34 / hr	<b>10 hr testing <math>\rightarrow</math> \$43.40</b>	720 hr (24 $\times$ 7) $\rightarrow$ \$3124.8
<b>RAG / API Node</b> (EC2 g5.xlarge)	$\sim$ \$0.97 / hr	<b>1 hr / day <math>\times</math> 30 d <math>\rightarrow</math> \$29.10</b>	24 $\times$ 7 uptime $\rightarrow$ \$698
<b>Vector Store</b> (Zilliz Cloud)	Free tier 1 M vectors · Standard \$99 / mo	Free tier $\rightarrow$ \$0	Standard tier $\rightarrow$ \$99
<b>Object Storage</b> (Amazon S3 Standard)	\$0.023 / GB / mo	<b>10 GB <math>\rightarrow</math> \$0.23</b>	1 TB $\rightarrow$ \$23
<b>Satellite Imagery Access</b>	Planetary Computer = \$0 Bhoonidhi $\approx$ \$2 / scene	<b><math>\approx</math> 12 scenes <math>\rightarrow</math> \$24</b>	$\sim$ 50 scenes $\rightarrow$ \$100
<b>Monthly Total</b>	–	<b><math>\approx</math> \$97</b>	<b><math>\approx</math> \$4 050</b>

Costs can further be cut by utilizing smaller computer for endpoint inference, essentially deploying a smaller model. The project is currently compatible with **Llama3.1-70B**, **Qwen2.5-72B** and **Deepseek-R1-Preview**. All these aforementioned models are trained on geoscience-specific subset of CommonCrawl.

# Implementation Cost

Cost and usage breakdown (17)			<a href="#">Download as CSV</a>
<input type="text" value="Find cost and usage data"/>			< 1 >
Service	Service total	July 2025*	
Total costs	\$87.94	\$74.53	
SageMaker	\$57.47	\$57.47	
Tax	\$13.95	\$3.23	
EC2-Instances	\$16.02	\$13.72	
Route 53	\$0.50	\$0.10	
EC2-Other	\$0.01	\$0.01	
S3	\$0.00	\$0.00	
CloudShell	\$0.00	\$0.00	
Glue	\$0.00	\$0.00	
Key Management Service	\$0.00	\$0.00	
Secrets Manager	\$0.00	\$0.00	
Service Catalog	\$0.00	\$0.00	
Elastic File System	\$0.00	\$0.00	
SNS	\$0.00	\$0.00	
SQS	\$0.00	\$0.00	
VPC	\$0.00	\$0.00	
CloudWatch	\$0.00	\$0.00	

# Video Demonstration Link

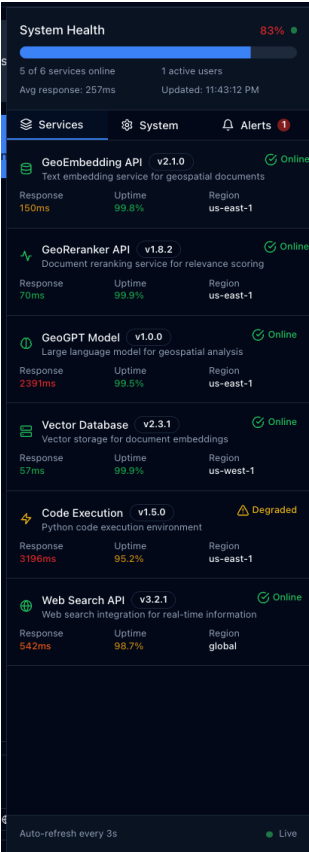
Youtube [ <https://youtu.be/6BRW9duBAfY> ]

# Github Repository Link

[Rekklesssss/geogpt-rag](https://github.com/Rekklesssss/geogpt-rag)  
(<https://github.com/Rekklesssss/geogpt-rag>)

Forgot to mention about system health monitoring and health service  
Each port from the pipeline is added with a /health check to monitor the status of the APIs  
Video demonstration doesn't include this too! (Completely Forgot)

# LIVE MONITORING OF SERVICES



## Mindmap (Bullets)

1. Selected an open source model (GeoGPT-R1-Preview) due to its high accuracy and COT process.
2. Deployed the model on an AWS Sagemaker Endpoint after creating a Huggingface TGI Container.
3. Hosted an EC2 Instance for RAG pipeline and Middleware CORS.\
4. Implemented basic API calls for /add\_file and /query for basic RAG file processing, this is where I added the two models for AI Intelligence inside RAG (GeoEmbedding and GeoReranker).
5. Created a Next.js frontend with the planned features and chat interface.
6. Integrated the pipeline with another Docker container for Sandboxed code execution and tool flow (PyQGIS, WhiteboxTools)
7. Added STAC APIs and Bhoonidhi Satellite imagery data in the pipeline processing.
8. Pipeline monitoring, deployment scripts, documentation

# BHARATIYA ANTARIKSH HACKATHON 2025

# THANK YOU

