Based on your original concept of an API finder, here's a redesigned project that fully complies with all hackathon requirements while maintaining your core vision:

### Project Name: \*\*API Genius - Intelligent API Discovery & Vulnerability Assistant\*\*

#### 🎯 Core Concept

An AI-powered platform that:

1. \*\*Analyzes public API datasets\*\* using Google Cloud AI

2. \*\*Recommends APIs\*\* using semantic search (MongoDB Vector Search)

3. \*\*Detects security vulnerabilities\*\* in API implementations

4. \*\*Generates secure integration code\*\* with Gemini

---

### ✅ How It Satisfies Requirements

| \*\*Requirement\*\* | \*\*Implementation\*\* |

|----------------|-------------------|

| \*\*Google Cloud AI\*\* | Vertex AI, Gemini Pro, Vision AI |

| \*\*MongoDB Features\*\* | Vector Search, Atlas Search |

| \*\*Public Dataset\*\* | APIs.guru, ProgrammableWeb, GovData |

| \*\*AI Analysis\*\* | Security scoring, semantic analysis |

| \*\*AI Generation\*\* | Code generation, vulnerability reports |

| \*\*Partner Integration\*\* | MongoDB Vector Search for semantic matching |

---

### 🔧 Technical Implementation

\*\*1. Data Pipeline (Google Cloud)\*\*

```python

from google.cloud import bigquery

from vertexai.preview.language\_models import TextEmbeddingModel

# Load public API datasets

dataset = bigquery.Client().dataset("public\_apis")

# Generate embeddings

model = TextEmbeddingModel.from\_pretrained("textembedding-gecko@003")

embeddings = model.get\_embeddings([api["description"] for api in apis])

```

\*\*2. MongoDB Vector Search\*\*

```javascript

// Atlas Vector Search Index

{

"fields": [

{ "type": "vector", "path": "embedding", "dimensions": 768 },

{ "type": "filter", "path": "category" }

]

}

// Semantic search query

db.apis.aggregate([

{

$vectorSearch: {

queryVector: [0.12, -0.05, ...],

path: "embedding",

limit: 10

}

}

])

```

\*\*3. AI Features\*\*

- \*\*Vulnerability Scanner\*\* (Vision AI + Gemini)

- \*\*Code Generator\*\* (Gemini Pro)

- \*\*Security Auditor\*\* (Vertex AI)

---

### 📽️ Demo Flow (3-min Video)

1. \*\*User Authentication\*\* (Google OAuth)

2. \*\*Natural Language Search\*\*: "Find weather APIs with free tier and OAuth support"

3. \*\*Vector Search Results\*\*: Visualize semantic matches

4. \*\*Security Audit\*\*: Upload screenshot → AI detects exposed keys

5. \*\*Code Generation\*\*: "Generate Python client for this API with security best practices"

---

### 🛠️ Tech Stack

| \*\*Component\*\* | \*\*Technology\*\* |

|--------------|---------------|

| Frontend | React + Material UI + Tailwind |

| Backend | Node + Express |

| Database | MongoDB Atlas (Vector Search) |

| AI Services | Vertex AI, Gemini Pro, Vision AI |

| Auth | Firebase Authentication |

| Hosting | Google Cloud Run |

---

### 📝 Submission Elements

1. \*\*Video Demo\*\*: Showing semantic search, vulnerability detection, and code generation

2. \*\*Public Repo\*\*: MIT-licensed on GitHub

3. \*\*Live Demo\*\*: Hosted on Cloud Run

4. \*\*Dataset\*\*: Public API datasets with enriched embeddings

5. \*\*Documentation\*\*: Explaining AI/vector search implementation

---

### ⚠️ Key Compliance Points

1. Uses \*\*Google Cloud AI\*\* as primary AI tool

2. Leverages \*\*MongoDB Vector Search\*\* for core functionality

3. Processes \*\*public datasets\*\* (APIs.guru)

4. Implements \*\*AI generation\*\* (code/vulnerability reports)

5. Avoids prohibited technologies (no competing cloud services)

This transforms your API finder concept into an AI-powered solution that directly addresses the MongoDB challenge requirements while adding unique value through security features and code generation - significantly increasing your chances of winning.

graph TD

A[User Query] --> B(Vertex AI Embedding)

B --> C[MongoDB Vector Search]

C --> D[Relevant APIs]

D --> E(Gemini-Powered Analysis)

E --> F[Security Audit & Code Gen]

APISage/

├── backend/

│ ├── routes/

│ ├── controllers/

│ ├── models/

│ ├── services/

│ ├── config/

│ ├── utils/

│ └── server.js

├── frontend/

│ ├── public/

│ └── src/

│ ├── assets/

│ ├── components/

│ ├── pages/

│ ├── contexts/

│ ├── services/

│ ├── styles/

│ ├── App.jsx

│ └── index.js

├── ml-models/

│ ├── embedding.js

│ └── vulnerability-checker.js

├── scripts/

│ └── data-ingestion.js

├── .env

└── README.md