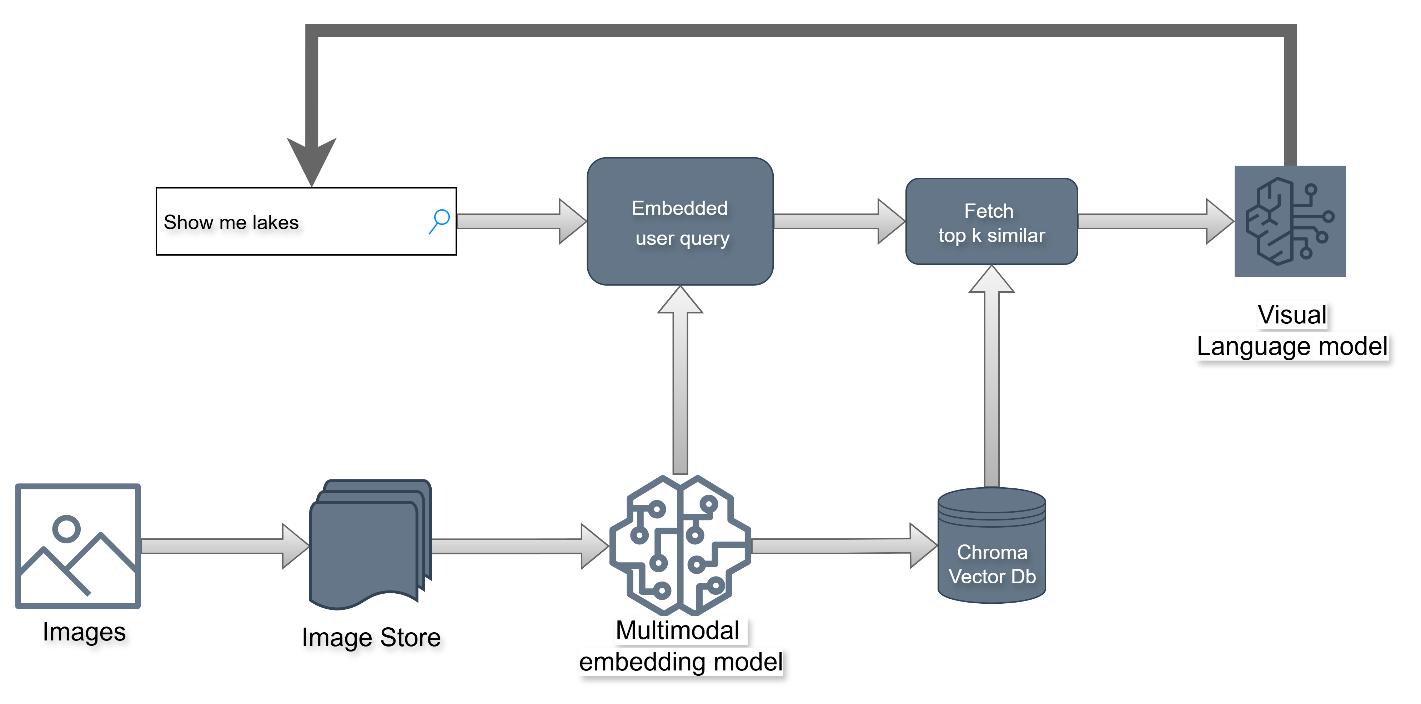
**System Architecture & Deployment Process**



|  |  |
| --- | --- |
| **Component** | **Role** |
| Image Downloader | Fetches images from URLs. |
| Image Store | Saves raw images for reference. |
| Multimodal Embedding Model | Converts images and text into vector representations. |
| Chroma Vector Database | Stores embeddings for fast similarity search (Top-K retrieval). |
| VLM (Vision-Language Model) | Generates detailed explanations for retrieved images. |
| Gradio App | Displays images with similarity score and explanation. |

**Deployment Steps for Multimodal Image Search System**

1. **Ingestion / Image Pipeline**

* **Scheduled Batch Job:** Use **Databricks** or another batch scheduler to download and preprocess images and ingest in vector db.
* **Embedding & Storage:**
  + Images stored temporarily or in blob storage.
  + Embeddings generated using **multimodal model** and stored in **Chroma DB** hosted as a service in **Azure**.

1. **Local Testing & Containerization**

* **Dockerize the App:** Build the FastAPI + Gradio app container:
  + docker build -t pbassignment:latest .
* **Push to Azure Container Registry (ACR):** For production deployment
* **Test Locally:**
  + docker run -p 8000:8000 pbassignment:latest

1. **Production Deployment**

* **Azure Container App / Web App for Containers:**
  + Deploy the container from **ACR**.
  + Configure environment variables:
    - MODEL\_PATH or API keys if required.
  + Create a hosted cromadb service
* **Authentication:**
  + Add **FastAPI authentication** (e.g., OAuth2, API Key) to secure endpoints.
* **Gradio App Deployment:**
  + Optionally, deploy Gradio separately as a lightweight container pointing to the same backend API.

1. **Scaling & Monitoring**

* **Auto-scaling:** Enable in Azure Container App based on CPU, memory, or request count.
* **Monitoring:**
  + Azure Monitor / Logs track request latency, errors, and system health.

1. **Optional Enhancements**

* Use **CD/CI pipelines** to automate build and deployment.

APP sample results

