I’ve looked at your adk\_hackathon\_streamlit repo, and the good news is you already have a strong ADK‑based multi‑agent pipeline with Streamlit UI, BigQuery integration, and modular agents for toxicity, bias, and misinformation.

For the **GKE Turns 10 Hackathon**, the main work is **not rewriting your logic**, but **wrapping it so it runs as a containerized microservice inside their required GKE demo app** (Bank of Anthos or Online Boutique) and connects via the ADK layer.

Here’s how to adapt your existing project for the hackathon with minimal extra cost and time:

**1️⃣ Choose the base app**

* Pick **Bank of Anthos** (simpler API surface) or **Online Boutique**.
* Deploy it to **GKE Autopilot** in the cheapest region.
* Keep it stock — don’t modify their core services.

**2️⃣ Containerize your Streamlit app**

* Create a Dockerfile in adk\_hackathon\_streamlit/:

dockerfile

FROM python:3.11-slim

WORKDIR /app

COPY requirements.txt .

RUN pip install -r requirements.txt

COPY . .

CMD ["streamlit", "run", "main.py", "--server.port=8080", "--server.address=0.0.0.0"]

* Build & push to **Artifact Registry**:

bash

gcloud builds submit --tag us-central1-docker.pkg.dev/<PROJECT\_ID>/<REPO\_NAME>/emakia-agent:latest

**3️⃣ Wrap it as an “agentic” microservice**

* In the hackathon’s context, your Streamlit app becomes **one new service** in the GKE cluster.
* Add a **REST endpoint** (e.g., /analyze) in main.py or a small FastAPI/Flask sidecar that:
  + Accepts text or URL from the base app.
  + Runs your ADK agent chain (toxicity → bias → misinformation).
  + Returns JSON with results.

**4️⃣ Connect to the base app via ADK**

* Use the **Agent Development Kit** to register your service as an agent that can be called by the base app’s orchestrator.
* Example: In Bank of Anthos, you could have the “transactions” service send transaction notes or messages to your agent for moderation.

**5️⃣ Deploy to GKE**

* Create a Kubernetes Deployment + Service YAML for your container.
* Apply it to the same namespace as the base app.
* Expose it internally so the base app can call it.

**6️⃣ Keep costs low**

* Use **Autopilot** and scale your deployment to replicas: 0 when idle.
* Only run the cluster during dev/testing and demo recording.
* Mock Gemini calls during dev; make a few real calls for the final demo.

**7️⃣ Hackathon deliverables**

* **Demo video**: Show Bank of Anthos → your agent → moderation results.
* **Architecture diagram**: Update your existing diagram.png to show:
  + Base app
  + Your Emakia agent service
  + ADK connection
  + Optional BigQuery
* **Repo**: Keep your current repo, add a /gke\_integration folder with Dockerfile, YAML manifests, and integration notes.

💡 **Why this works for you**

* You reuse 90% of your existing code.
* You meet the hackathon’s “add an agentic AI layer” requirement.
* You keep infra costs under ~$20 by running small and short‑lived.
* You showcase Emakia’s moderation pipeline in a Google Cloud + GKE context.

If you want, I can **write the exact Kubernetes YAML + minimal FastAPI wrapper** so your Streamlit/ADK pipeline can be dropped into Bank of Anthos in one shot — that would save you a lot of integration time.

Do you want me to prepare that next?