

Workshop: Building a Recommendation System with H&M Data on GCP

Overview

In this workshop, you will build an end-to-end recommendation system using the H&M Personalized Fashion Recommendations dataset. You will leverage Google Cloud's **Vertex AI** for model training (Two-Tower and LightGBM) and **Cloud Run** for deploying the backend and frontend services.

What you will learn:

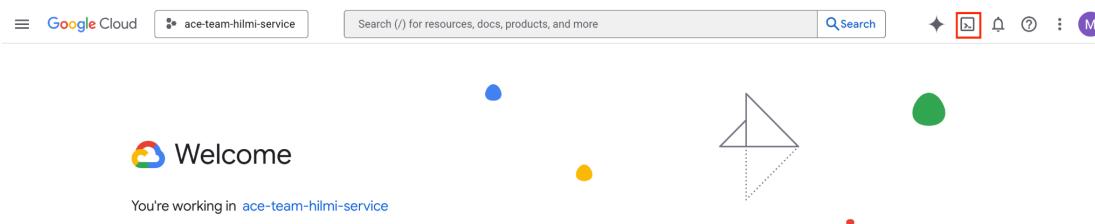
- Setting up a High-Performance Data Science environment on GCP.
- Training a Two-Tower Retrieval model using GPUs on Vertex AI Colab Enterprise.
- Training a LightGBM Ranking model and deploying a serving API.
- Deploying a Streamlit frontend to visualize recommendations.

Task 1: Environment Setup

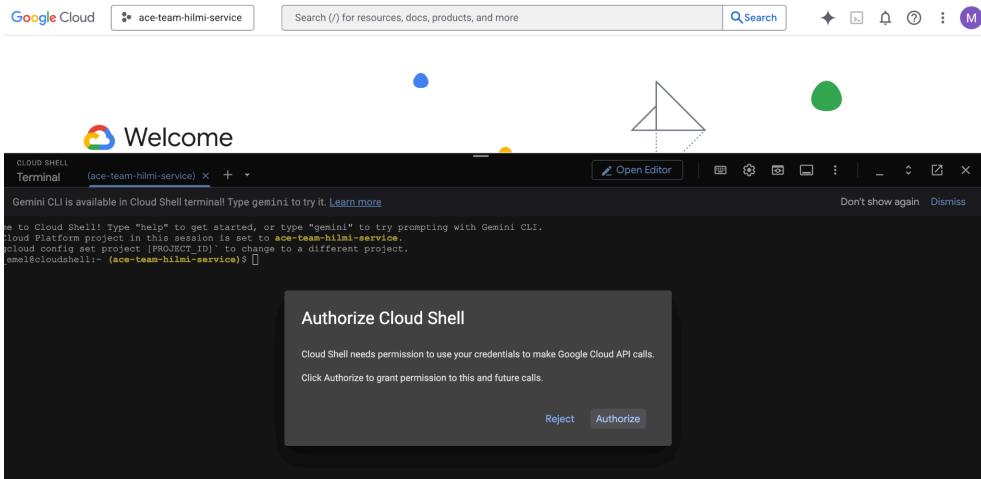
Activate Google Cloud Shell

Google Cloud Shell is a virtual machine loaded with development tools. It offers a persistent 5GB home directory and runs on the Google Cloud. It provides command-line access to your Google Cloud resources.

1. In the Cloud Console, on the top right toolbar, click the **Open Cloud Shell** button.



2. Click **Authorize**.



It takes a few moments to provision and connect to the environment. When you are connected, you are already authenticated, and the project is set to your **PROJECT_ID**.

3. Verify your project ID by running the following command (optional):

```
Shell  
gcloud config get-value project
```

Download and Run the Setup Script

We have prepared an automated script to provision the necessary infrastructure.

1. Run the following command in Cloud Shell to download the setup script:

```
Shell  
wget  
https://raw.githubusercontent.com/hilmi-collab/HM\_Recommendation\_System\_on\_GCP/main/setup\_workshop.sh
```

2. Verify that the file has been downloaded successfully:

```
Shell  
ls
```

You should see `setup_workshop.sh` in the list.

```
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ ls  
README-cloudshell.txt  setup_workshop.sh  
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ █
```

3. **What does this script do?** Before running it, here is a summary of the resources it creates:

- **Enable APIs:** Activates Vertex AI, Dataform, Compute Engine, Cloud Run, and Cloud Build APIs.
- **IAM Permissions:** Grants necessary roles (Storage Admin, Vertex AI User, etc.) to your account and the Compute Service Account.
- **Runtime Templates:** Creates two high-performance Vertex AI Runtime templates:
- **hm-retrieval-gpu-template** (Optimized for RAM/GPU)
- **hm-ranking-gpu-template** (Optimized for CPU/RAM)
- **Cloud Storage:** Creates a bucket (`hm-workshop-[PROJECT_ID]`) and uploads the training notebooks.
- **Frontend Files:** Downloads the necessary Python scripts and Dockerfiles for the Streamlit UI.

4. Make the script executable and run it:

Shell

```
chmod +x setup_workshop.sh  
./setup_workshop.sh
```

Wait for the script to finish. You should see a "READY (HIGH PERFORMANCE)" message at the end.

```
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ chmod +x setup_workshop.sh  
.setup_workshop.sh  
=====  
HAM Workshop Setup (High Performance Mode)  
=====  
Your active configuration is: [cloudshell-11500]  
Your active configuration is: [cloudshell-11500]  
[Info] Project: ace-team-hilmi-service  
[Info] Work Bucket: hm-workshop-ace-team-hilmi-service  
[Step 1/5] Enabling APIs...  
[Step 2/5] Configuring IAM...  
[Step 3/5] Creating High-Performance Runtime Templates...  
Creating template: hm-retrieval-gpu-template (n1-standard-8)...  
✓ hm-retrieval-gpu-template created.  
Creating template: hm-ranking-gpu-template (n1-standard-16)...  
✓ hm-ranking-gpu-template created.  
[Step 4/5] Setting up User Bucket & Notebooks...  
✓ Bucket created: hm-workshop-ace-team-hilmi-service  
  -> hm_tower_training.ipynb uploaded.  
  -> hm_ranking_lightgbm_training.ipynb uploaded.  
[Step 5/5] Downloading Frontend Files...  
=====  
⚡ **READY (HIGH PERFORMANCE)**  
Notebooks: gs://hm-workshop-ace-team-hilmi-service/notebooks  
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ █
```

Verify Created Resources

1. **Check Runtime Templates:** Navigate to **Vertex AI > Colab Enterprise** and click on **Runtime templates** on the left sidebar. You should see the two templates created by the script.

The screenshot shows the Vertex AI Colab Enterprise interface. The left sidebar has sections like Dashboard, Model Garden, Vertex AI Studio, GenAI Evaluation, Tuning, Agent Builder, Notebooks, and Colab Enterprise. Under Colab Enterprise, there are sub-options like Workbench, Model development, Feature Store, Provisioned Throughput, Pipelines, and Tutorials. The main area is titled "Runtime templates" and shows a table with three rows. The columns are Template name, ID, Machine type, GPU type, Data disk type, Network, and Actions. The first row is "hm-ranking-gpu-template" with ID 547171961562726400, machine type n1-standard-16, GPU type NVIDIA_TESLA_T4 x 1, 100 GB Standard Disk (pd-standard), and default network. The second row is "hm-retrieval-gpu-template" with ID 3777378794294214656, machine type n1-standard-8, GPU type NVIDIA_TESLA_T4 x 1, 100 GB Standard Disk (pd-standard), and default network. The third row is "Default" with ID 5069887698093735936, machine type e2-standard-4, no GPU, 100 GB Standard Disk (pd-standard), and default network.

Template name	ID	Machine type	GPU type	Data disk type	Network	Actions
hm-ranking-gpu-template	547171961562726400	n1-standard-16	NVIDIA_TESLA_T4 x 1	100 GB Standard Disk (pd-standard)	default	
hm-retrieval-gpu-template	3777378794294214656	n1-standard-8	NVIDIA_TESLA_T4 x 1	100 GB Standard Disk (pd-standard)	default	
Default	5069887698093735936	e2-standard-4	—	100 GB Standard Disk (pd-standard)	default	

2. **Check Cloud Storage:** Navigate to **Cloud Storage > Buckets**. You will see a bucket named **hm-workshop- [PROJECT_ID]**. (*Example: hm-workshop-ace-team-hilmi-service*)

The screenshot shows the Google Cloud Cloud Storage interface. The left sidebar has sections like Overview, Buckets, Monitoring, Settings, Storage Intelligence, Insights datasets, and Configuration. The main area is titled "Buckets" and shows a table with one row. The columns are Name, Created, Location type, Location, Default storage class, and Last modified. The single row is "hm-workshop-ace-team-hilmi-service" with Created date Dec 10, 2025, 4:21:07PM, Region location type, us-central1 location, Standard storage class, and Last modified date Dec 10, 2025, 4:21:07PM.

Name	Created	Location type	Location	Default storage class	Last modified
hm-workshop-ace-team-hilmi-service	Dec 10, 2025, 4:21:07PM	Region	us-central1	Standard	Dec 10, 2025, 4:21:07PM

3. **Check Notebooks:** Click on the bucket name, then open the **notebooks** folder. You will see **hm_two_tower_training.ipynb** and **hm_ranking_lightgbm_training.ipynb**.

4. **Check Frontend Files:** Return to Cloud Shell and click the **Open Editor** button to view your workspace. You should see a folder named `hm_frontend` containing `streamlit_app.py`, `Dockerfile`, and `requirements.txt`.

```

CLOUD SHELL Terminal (ace-team-hilmi-service) + - Open Editor Don't show again Dismiss
Gemini CLI is available in Cloud Shell terminal! Type gemini to try it. Learn more
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ chmod +x setup_workshop.sh
./setup_workshop.sh
=====
[HWM Workshop Setup (High Performance Mode)]
Your active configuration is: [cloudshell-11500]
Your active configuration is: [cloudshell-11500]
[Info] Project: ace-team-hilmi-service
[Info] Work Bucket: hm-workshop-ace-team-hilmi-service
[Info] Using existing API...
[Step 2/5] Configuring IAM...
[Step 3/5] Creating High-Performance Runtime Templates...
Creating template: hm-retrieval-gpu-template (n1-standard-8)...
Creating template: hm-ranking-gpu-template (n1-standard-16)...
✓ hm-ranking-gpu-template created.
[Step 4/5] Setting up User Bucket & Notebooks...
✓ Bucket hm-workshop-ace-team-hilmi-service
  - hm_ranking_lightgbm_training.ipynb uploaded.
  - hm_two_tower_training.ipynb uploaded.
-> hm_ranking_lightgbm_training.ipynb uploaded.
[Step 5/5] Downloading Frontend Files...
✓ **READY (HIGH PERFORMANCE)**
Notebooks: gs://hm-workshop-ace-team-hilmi-service/notebooks
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ ls
hm_frontend README-cloudshell.txt setup_workshop.sh
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$

```

The screenshot shows a Cloud Shell Editor interface. On the left, the Explorer sidebar displays a project structure with a folder named 'HILMI...' containing 'hm_frontend', 'Dockerfile', 'requirements.txt', and 'streamlit_app.py'. The 'streamlit_app.py' file is currently selected and shown in the main editor area. The code in 'streamlit_app.py' is a Python script for a Streamlit application, defining imports, settings, helper functions, and data loading logic. The status bar at the bottom indicates 'Ln 1, Col 1' and 'Python 3.12.3'.

```

CLOUD SHELL
Editor
File Edit Selection View Go Run Terminal ... ← → ⌂ hilm_eme1
EXPLORER
HILMI... hm_frontend Dockerfile requirements.txt streamlit_app.py README-cloudshell... $ setup_workshop.sh
OUTLINE
TIMELINE
Ln 1, Col 1 Spaces: 4 UTF-8 LF {} Python 3.12.3

```

```

streamlit_app.py
1 import streamlit as st
2 import pandas as pd
3 import requests
4 import math
5 import os
6
7 # =====
8 # 1. SETTINGS
9 #
10 st.set_page_config(page_title="HGM AI Shop", layout="wide", page_icon="🛍️")
11
12 # =====
13 # 2. HELPER FUNCTIONS
14 #
15 def get_image_url(article_id):
16     # Whatever the ID is (int or str), convert to string first
17     # Then strip whitespace
18     # Then pad with zeros to ensure 10 digits
19     article_id = str(article_id).strip().zfill(10)
20
21     base_url = "https://repo.hops.works/dev/dowling/h-and-m/images"
22     folder = article_id[:3]
23     return f"{base_url}/{folder}/{article_id}.jpg"
24
25 def sigmoid(x):
26     try: return 1 / (1 + math.exp(-x))
27     except: return 0.0
28
29 # =====
30 # 3. LOAD DATA
31 #
32 @st.cache_data
33 def load_data():
34     ...

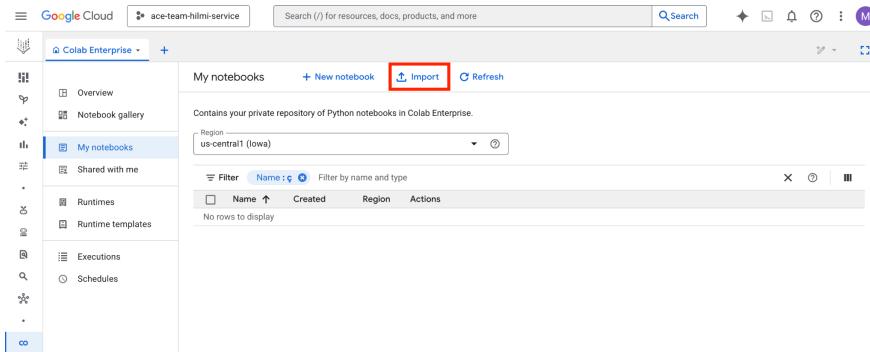
```

Task 2: Training the Two-Tower (Retrieval) Model

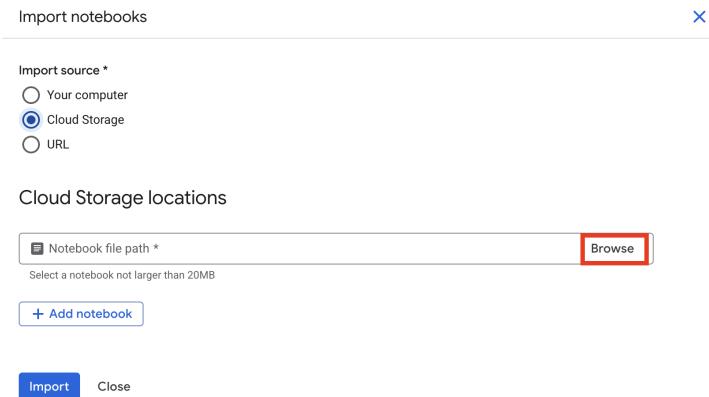
The Two-Tower model is responsible for "Retrieval." Instead of ranking every single item for a user (which is slow), this model finds a smaller set of relevant candidate items based on user-item similarity.

Import Notebooks to Vertex AI

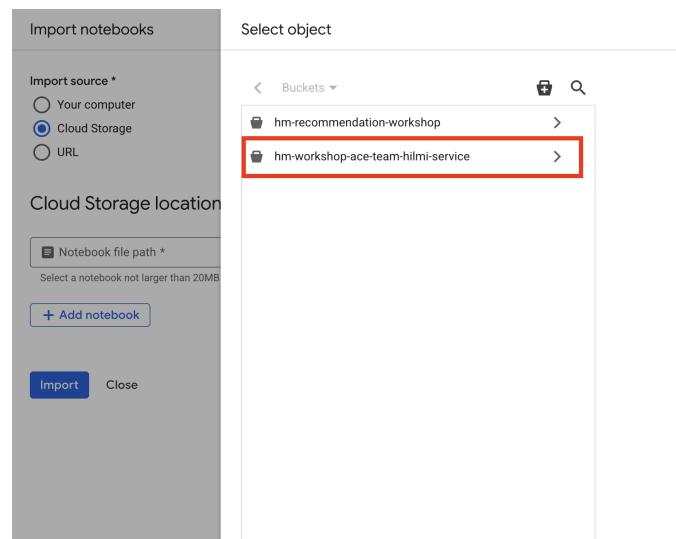
1. Navigate to **Vertex AI > Colab Enterprise > My Notebooks**.
2. Click **Import**.

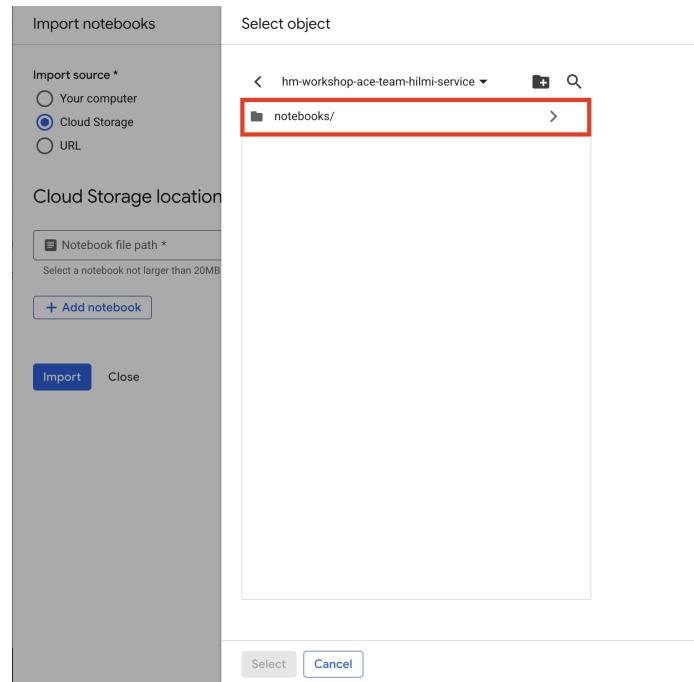


3. Select **Cloud Storage** as the source. Click **Browse** under "Cloud Storage locations".

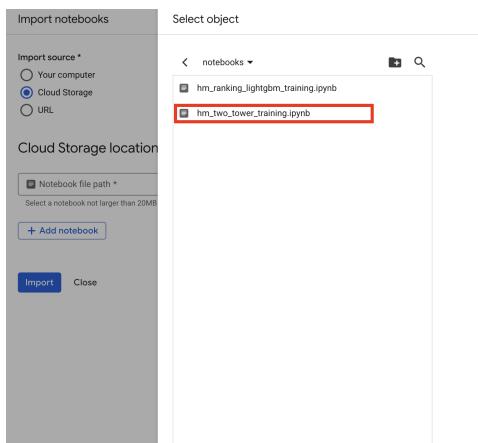


4. Select your bucket (`hm-workshop-[PROJECT_ID]`) > notebooks.

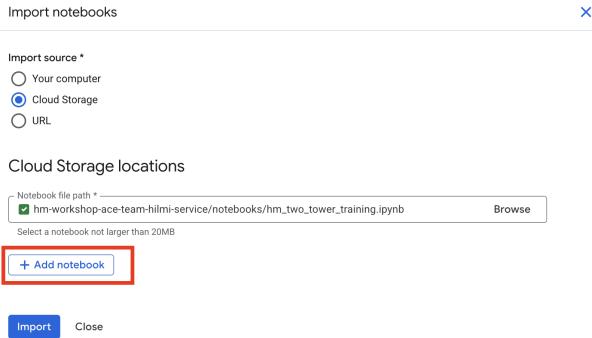




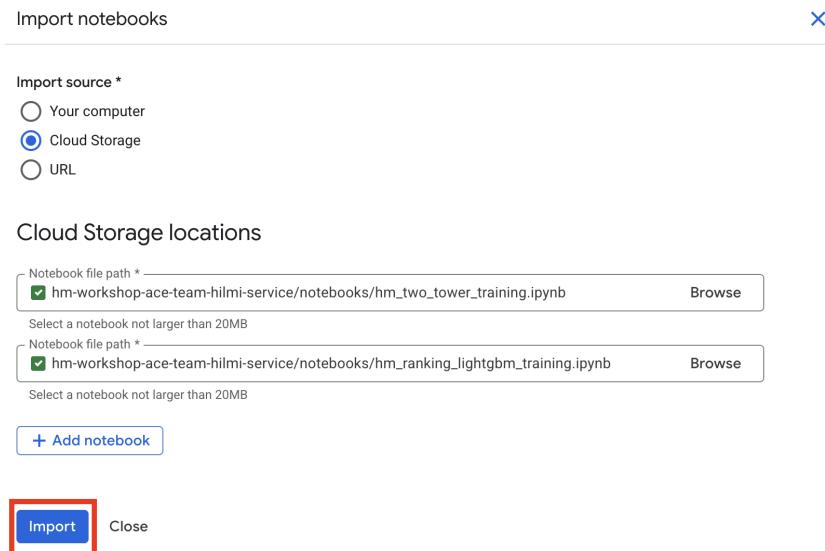
5. Select **hm_two_tower_training.ipynb** and click **Select**.



6. Click **+ Add notebook** and repeat the process for **hm_ranking_lightgbm_training.ipynb**.



7. Click Import.



Train the Model

1. Click on `hm_two_tower_training.ipynb` to open it.
2. **Update Project ID:** In the "Project Settings" cell, replace `PROJECT_ID` with your actual Google Cloud Project ID.

```

# -- coding: utf-8 --
# title @ Workshop Configuration
# markdown Please enter your Project ID.

import os

# markdown ## Project Settings
PROJECT_ID = "YOUR_PROJECT_ID" # @param {type:"string"}
REGION = "us-central1" # @param {type:"string"}

# 1. PUBLIC DATA BUCKET (Read-Only)
# Raw data (CSV) will be read from here.
DATA_BUCKET_NAME = "hm-recommendation-workshop"
DATA_GCS_PATH = f"gs://{DATA_BUCKET_NAME}"

# 2. PRIVATE WORK BUCKET (Write)
# Trained models will be saved here.
WORK_BUCKET_NAME = f"hm-workshop-{PROJECT_ID}"
WORK_GCS_PATH = f"gs://{WORK_BUCKET_NAME}"

# markdown ## Model Hyperparameters
EMBEDDING_DIM = 64 # @param {type:"integer"}
LEARNING_RATE = 0.1 # @param {type:"number"}

```

Project Settings

- PROJECT_ID:
- REGION: us-central1

Model Hyperparameters

- EMBEDDING_DIM: 64
- LEARNING_RATE: 0.1
- EPOCHS: 5

3. Connect to Runtime:

- Click the arrow next to the **Connect** button (top right).
- Select **Connect to a runtime**.

Please enter your Project ID.

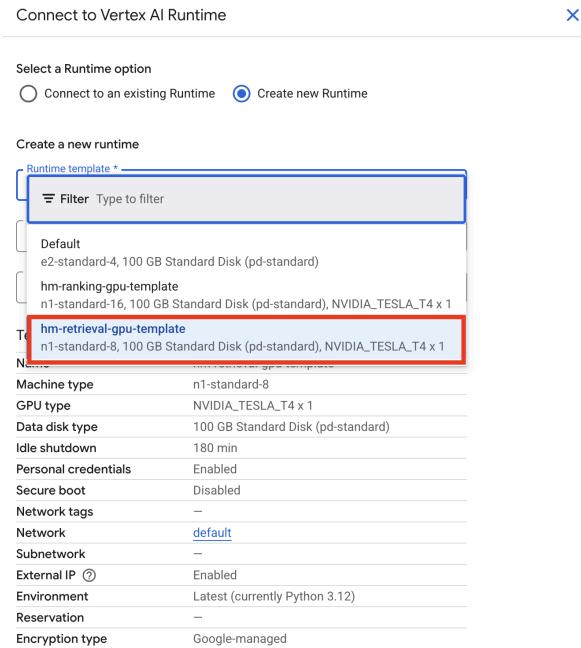
Project Settings

- PROJECT_ID: ace-team-hilmi
- REGION: us-central1

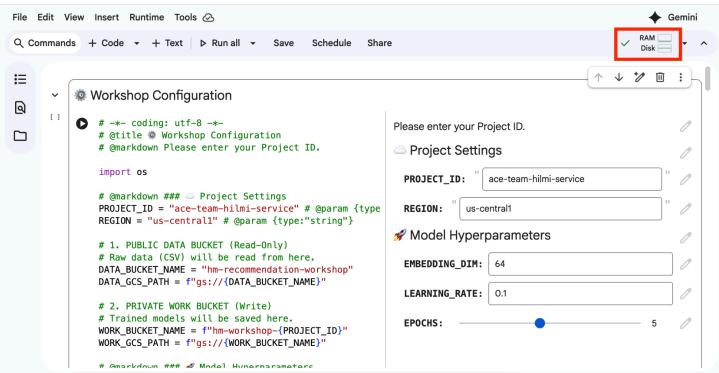
Model Hyperparameters

- EMBEDDING_DIM: 64
- LEARNING_RATE: 0.1
- EPOCHS: 5

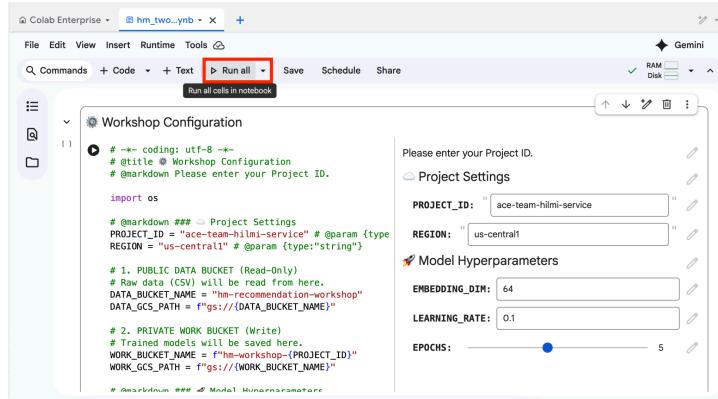
- Select **Create new Runtime**.
- **Crucial:** Under "Runtime template", select **hm-retrieval-gpu-template**.



- Click Connect.
4. Wait for the status bar at the bottom to show "Connected".



5. Run the Training: Click the Run all button in the toolbar.
- Note:** This process involves downloading data, preprocessing, and training the model. It may take approximately **30 minutes** to complete.
- You can monitor progress by watching the execution indicators next to the cells.



6. Verify Output: Once finished, the model is saved to your Cloud Storage bucket. You can verify this by going to **Cloud Storage > [Your Bucket] > models > two_tower_model**.

Name	Size	Type	Created	Storage class	Last modified	Public a
assets/	—	Folder	—	—	—	⋮
fingerprint.pb	97 B	File	Dec 11, 2025, 1:32:25PM	Standard	Dec 11, 2025, 1:32:25PM	Not pub ⚙ ⋮
saved_model.pb	101.7 MB	File	Dec 11, 2025, 1:32:23PM	Standard	Dec 11, 2025, 1:32:23PM	Not pub ⚙ ⋮
variables/	—	Folder	—	—	—	⋮

Clean Up Retrieval Runtime

To save costs, we will delete the runtime we just used, as we need a different machine configuration for the next step.

1. Go to Vertex AI > Colab Enterprise.

Name	Created	Region	Project
hm_two_tower_training.ynb	Dec 11, 2025, 1:07:05PM	us-central1	ace-team-hilmi-service
hilmi.eml	(Dec 2, 2025, 2:03:28PM)	us-central1	ace-team-hilmi-service

2. Click the three dots (options) next to your active runtime.
3. Select **Delete** and confirm.

The screenshot shows the 'Runtimes' section of the Colab Enterprise interface. A list of runtimes is displayed, including their names, IDs, statuses, templates, users, creation dates, and upgrade options. A context menu is open over the first runtime, with the 'Delete' option highlighted in red. Other options in the menu include Start, Stop, View Logs, and Upgrade.

Task 3: Training the Ranking Model (LightGBM) & Deploying Backend

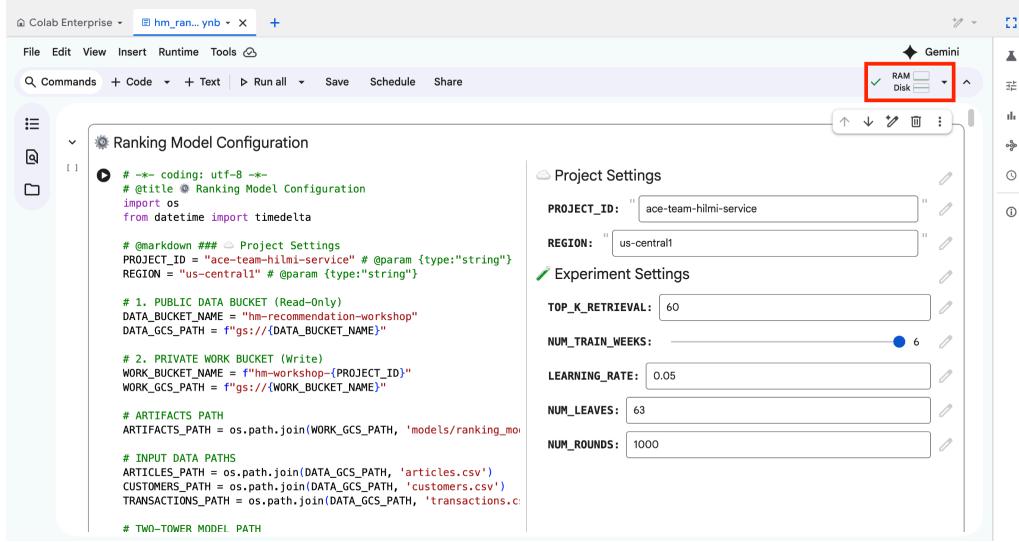
Now we will train the Ranking Model. This model takes the candidates from the Two-Tower model and re-orders them to find the absolute best recommendations.

1. Go back to **My Notebooks** and open `hm_ranking_lightgbm_training.ipynb`.

The screenshot shows a Google Colab notebook titled "hm_ranking_lightgbm_training.ipynb". The code cell contains Python code for a "Ranking Model Configuration". The code includes variables like PROJECT_ID, REGION, DATA_BUCKET_NAME, WORK_BUCKET_NAME, ARTIFACTS_PATH, and INPUT_DATA_PATHS. To the right of the code cell, there are two panels: "Project Settings" and "Experiment Settings". The "Project Settings" panel has fields for "PROJECT_ID" and "REGION". The "Experiment Settings" panel has fields for "TOP_K_RETRIEVAL", "NUM_TRAIN_WEEKS" (set to 6), "LEARNING_RATE", "NUM_LEAVES", and "NUM_ROUNDS".

2. **Update Project ID:** Replace the `PROJECT_ID` variable in the code with your project ID.
3. **Connect to Ranking Runtime:**
 - Click **Connect > Connect to a runtime**.
 - Select **Create new Runtime**.
 - **Crucial:** This time, select **hm-ranking-gpu-template**. (This template has high CPU/RAM required for processing 6 weeks of transaction data).

- Click Connect.



4. Run All: Click the Run all button.

Note: This notebook trains the LightGBM model, builds a backend API docker image, and deploys it to Cloud Run. This process may take up to **1 hour**.

5. Important Outputs: Once execution finishes, scroll through the output to find two key pieces of information:

- **User ID for Testing:** The notebook lists users with successful recommendations. Note down the ID of the top user (e.g., **499f785...**).

```
>>> Loading product names for human-readable report...
>>> Analyzing hits...
...
===== DEMO REPORT: At least 1 hit for a total of 5293 users! =====
=====
TOP 10 PREDICTION EXAMPLES:
Customer: 499f785cb034c252c3cc6e15c90@0e13602a3fe45611193c221ec7ac4794463
Hit Count: 7
Known Products: Tanya mockneck LS (Top), Joey HW contrast. TRS (Trousers), Tanya mockneck LS (Top), Pia PU TRS (Trousers), Victorvi
Customer: e3a37c99824ced2c752efc7b2bd1f6119821bb8ef97949a6ef9c358cfda3cce68
Hit Count: 6
Known Products: Raven blazer (Blazer), Mariette Blazer (Blazer), Volpino slacks (Trousers), Emma Roll Neck TVP (Sweater), Lia Slack
Customer: 5062031b31f3f104a300a4292424285a392e01c54fdc5f086a9b949ea1960a9f4c
Hit Count: 6
Known Products: Furry Roll Neck (Top), CASSIM SLIM ROLLNECK (Sweater), RICHIE HOOD (Hoodie), RICHIE HOOD (Hoodie), RICHIE HOOD (Hoo
Customer: 7afbd505e749984f6409229ffc0159876bf75a63bfeeed0a48bf024260b2404e
Hit Count: 5
Known Products: NILS LS CHECK (Shirt), Uno Hood (1) (Hoodie), Pink HW barrel (Trousers), Michael Hood 24.99:- (Hoodie), Becka hoodie
Customer: d2e5e07dadff8a95928c7458c8dca5a71165fb11e60aeaf4f6557c5eca214eb8
Hit Count: 5
Known Products: Tilly (1) (T-shirt), Alisa (Sweater), Flirt mini (Sweater), Sandy (Sweater), Scout (Top) (ID: 0610776001, 086864100
Customer: aa917a0741ef844743de6e6527b72d31c88c343eb081edba8ab99b82750d8da
Hit Count: 5
Known Products: Lucy blouse (Shirt), Roxy Blouse. (Blouse), Roxy Blouse. (Blouse), Lucy blouse (Shirt), Lucy blouse (Shirt) (ID: 07
Customer: 6e877f26b9b464ff1244f1fb4edcca219658df3d70452cbeec3ec5a25b7a1a14
```

- **Backend Service URL:** The final cell deploys the API. Copy the **Cloud Run Service URL** printed in the output.

```

900652cccd93a: Preparing
d95b894f0630: Preparing
8f1dd470896e: Preparing
9e1dd70bb39: Preparing
5f70bf18a086: Preparing
68410d957429: Preparing
7c8a8be9d47f: Preparing
7c8a8be9d47f: Layer already exists
68410d957429: Layer already exists
5f70bf18a086: Layer already exists
900652cccd93a: Pushed
ee6f9d60b32d: Pushed
d95b894f0630: Pushed
8f1dd470896e: Pushed
9e1dd70bb39: Pushed
f2f1c6f2360b: Pushed
latest: digest: sha256:e377abfea31956af3badda9a94c9019a4bbde4645c8a590dc56240a3f0f8abb size: 2201
DONE
ID          CREATE_TIME      DURATION SOURCE
68f36975-5f10-458f-9d55-574786ded6e3 2025-12-11T11:17:14+00:00 5M40S gs://ace-team-hilmi-service_cloudbu
Deploying container to Cloud Run service [hm-recommender-service] in project [ace-team-hilmi-service] region [us-central1]
Service [hm-recommender-service] revision [hm-recommender-service-00001-hkd] has been deployed and is serving
Service URL: https://hm-recommender-service-404135610173.us-central1.run.app
Deployment Complete! Check the URL.

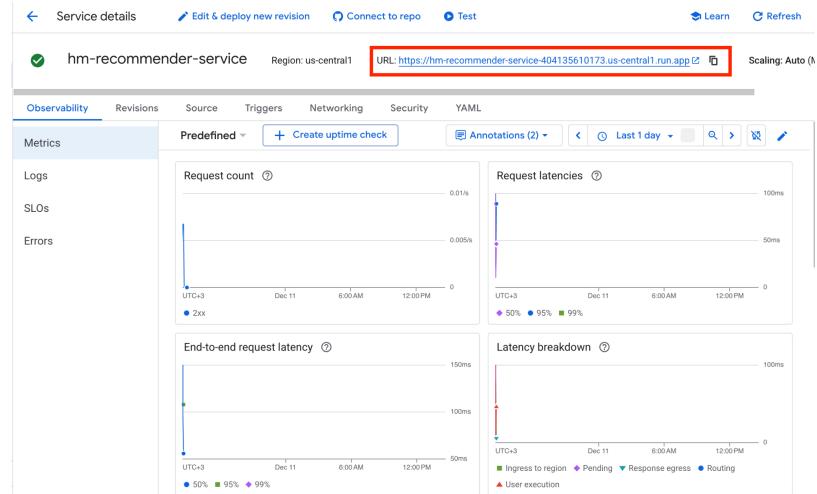
```

6. Verify Backend Deployment:

- Go to **Cloud Run** in the console.
- Click on the service named **hm-recommender-service**.

Name	Region	Type	Last updated
hm-recommender-service	us-central1	Service	10 minutes ago

- Ensure the URL matches the one you copied.



7. **Clean Up Runtime:** Return to Vertex AI > Colab Enterprise and **Delete** the ranking runtime to stop billing.

Runtime name	ID	Status	Template	User	Created	Upgrade
hilmi.emel-20251211-134734	736631009168064512	Healthy	8571460597630107648	hilmi.emel@acedemand.com	Dec 11, 2025, 1:49:18 PM	Delete

Task 4: Deploying the Frontend Application

We will now deploy a Streamlit user interface to visualize our recommendations.

1. Return to **Google Cloud Shell**.
2. If the session has timed out, click **Reconnect**.
3. Navigate to the frontend directory:

```
Shell
cd ~/hm_frontend
```

Verify you are in the correct folder:

```
Shell  
ls
```

```
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ ls  
hm_frontend README-cloudshell.txt setup_workshop.sh  
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ █
```

Download Model Artifacts

The frontend needs specific feature files to display product details. We will download these from the ranking model output in Cloud Storage.

1. Run the following commands:

```
Shell  
  
export PROJECT_ID=$(gcloud config get-value project)  
export BUCKET_NAME="hm-workshop-${PROJECT_ID}"  
  
echo "📦 Downloading artifacts..."  
gsutil cp gs://$BUCKET_NAME/models/ranking_model/app_customers_features.parquet .  
  
gsutil cp gs://$BUCKET_NAME/models/ranking_model/app_articles_features.parquet .  
  
gsutil cp gs://$BUCKET_NAME/models/ranking_model/val_truth.parquet .
```

2. Verify the files are present:

```
Shell  
ls -lh *.parquet
```

```
hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ cd ~/hm_frontend  
  
export PROJECT_ID=$(gcloud config get-value project)  
export BUCKET_NAME="hm-workshop-${PROJECT_ID}"  
  
echo "📦 files are installing..."  
gsutil cp gs://$BUCKET_NAME/models/ranking_model/app_customers_features.parquet .  
gsutil cp gs://$BUCKET_NAME/models/ranking_model/app_articles_features.parquet .  
gsutil cp gs://$BUCKET_NAME/models/ranking_model/val_truth.parquet .  
  
ls -lh *.parquet  
Your active configuration is: [cloudshell-4623]  
📦 files are installing...  
Copying gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/app_customers_features.parquet...  
| [1 files] | 86.8 MiB / 86.8 MiB  
Operation completed over 1 objects/86.8 MiB.  
Copying gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/app_articles_features.parquet...  
| [1 files] | 1.9 MiB / 1.9 MiB  
Operation completed over 1 objects/1.9 MiB.  
Copying gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/val_truth.parquet...  
| [1 files] | 5.7 MiB / 5.7 MiB  
Operation completed over 1 objects/5.7 MiB.  
-rw-rw-r-- 1 hilmi_emel hilmi_emel 2.0M Dec 11 11:48 app_articles_features.parquet  
-rw-rw-r-- 1 hilmi_emel hilmi_emel 87M Dec 11 11:48 app_customers_features.parquet  
-rw-rw-r-- 1 hilmi_emel hilmi_emel 5.8M Dec 11 11:48 val_truth.parquet  
hilmi_emel@cloudshell:~/hm_frontend (ace-team-hilmi-service)$ █
```

Build and Deploy to Cloud Run

1. Run the build and deploy script:

```
Shell
```

```
echo "🔨 Building Streamlit Image..."  
gcloud builds submit --tag gcr.io/$PROJECT_ID/hm-streamlit-app .  
  
echo "🚀 Deploying to Cloud Run..."  
gcloud run deploy hm-streamlit-ui \  
--image gcr.io/$PROJECT_ID/hm-streamlit-app \  
--platform managed \  
--region us-central1 \  
--allow-unauthenticated \  
--memory 4Gi
```

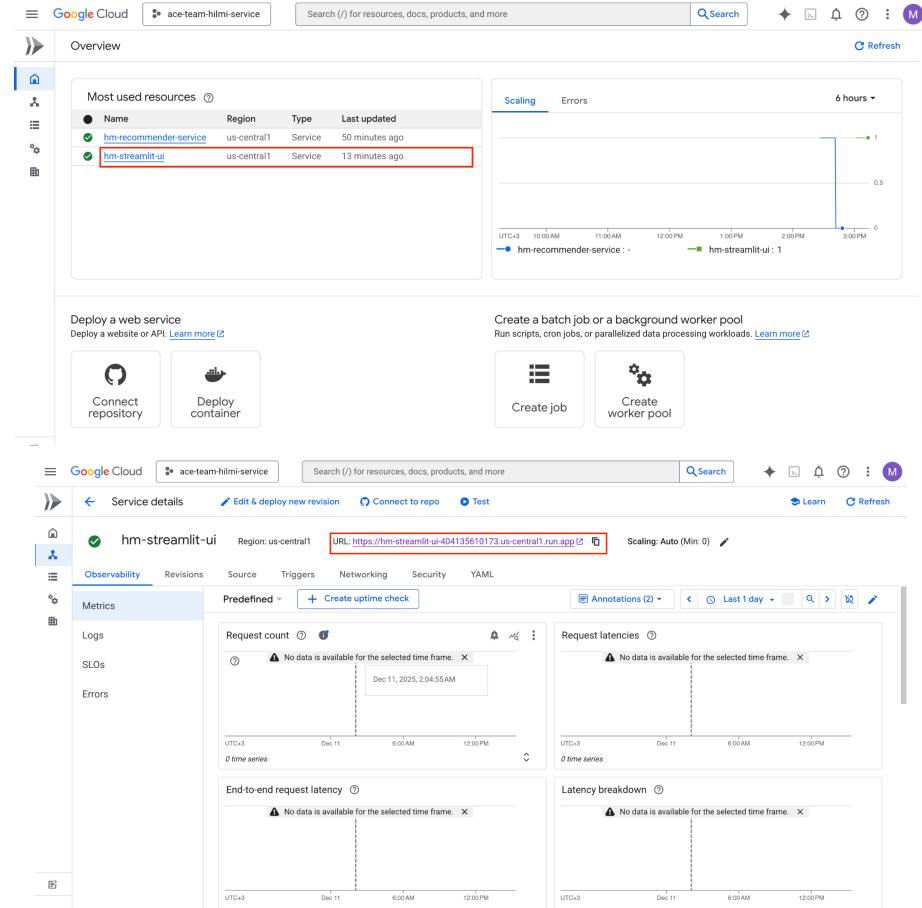
Note: `--allow-unauthenticated` makes the application publicly accessible for the purpose of this workshop.

2. Once complete, look for the **Service URL** in the output.

```
DONE  
-----  
ID: 1bd2ff069-8c0b-44ca-a61a-41300eaf0c5a  
CREATE_TIME: 2025-12-11T11:57:56+00:00  
DURATION: 1M36S  
SOURCE: gs://ace-team-hilmi-service_cloudbuild/source/1765454253.466572-0194ef0075d648edabeacddf9134791d.tgz  
IMAGES: gcr.io/ace-team-hilmi-service/hm-streamlit-app (+1 more)  
STATUS: SUCCESS  
🚀 Deploying.  
Deploying container to Cloud Run service [hm-streamlit-ui] in project [ace-team-hilmi-service] region [us-central1]  
Deploying new service...  
  Setting IAM Policy...done  
  Creating Revision...done  
  Routing traffic...done  
Done.  
Service [hm-streamlit-ui] revision [hm-streamlit-ui-00001-hwh] has been deployed and is serving 100 percent of traffic.  
Service URL: https://hm-streamlit-ui-404135610173.us-central1.run.app  
✔ Frontend Deployment Complete!  
hilmi_emel@cloudshell:~/hm_frontend (ace-team-hilmi-service)$ █
```

Task 5: Testing the Recommendation System

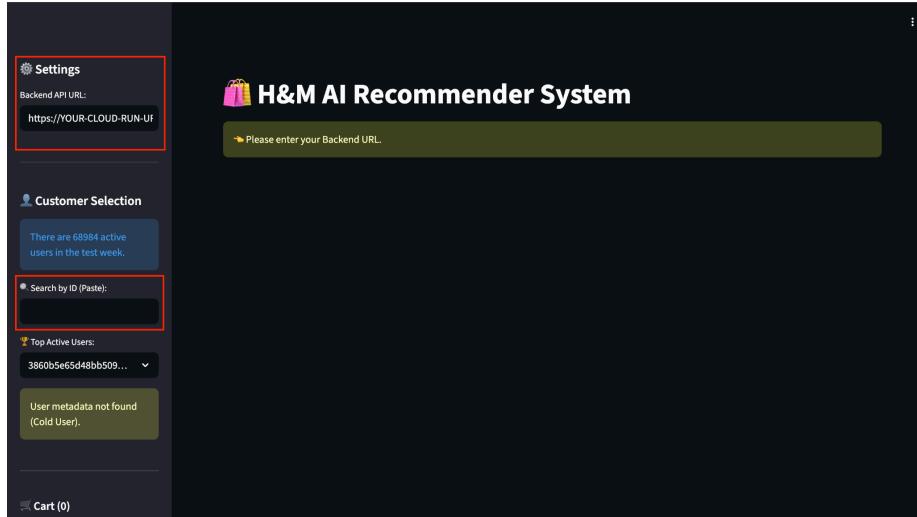
1. Click the **Service URL** generated in the previous step (or go to Cloud Run > `hm-streamlit-ui` and click the URL there).



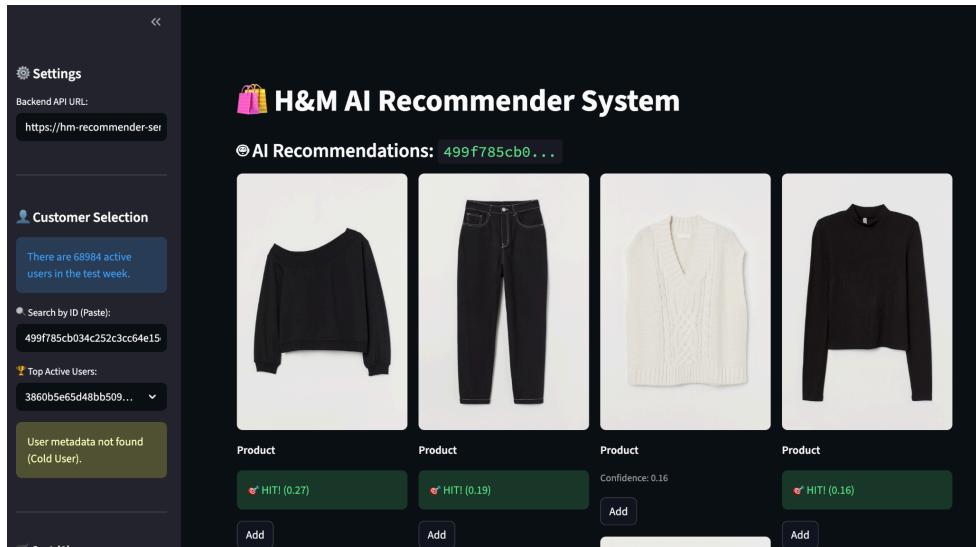
2. Wait 1-2 minutes for the app to initialize.

3. Configure the App:

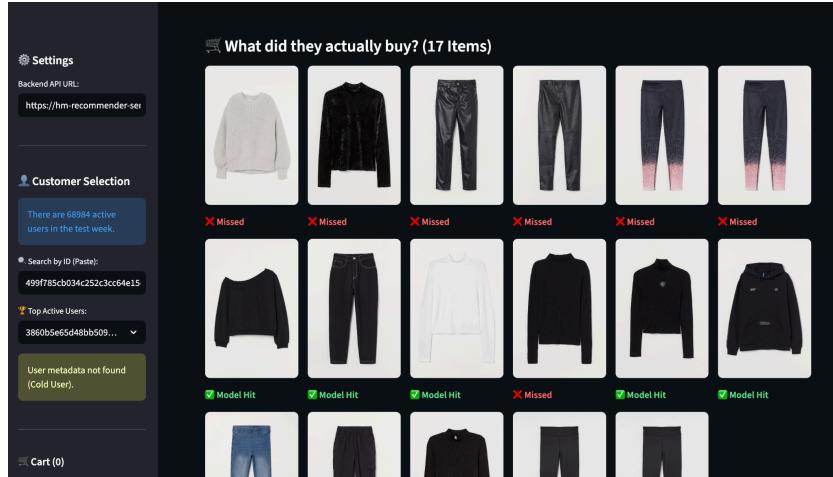
- **Backend API URL:** Paste the URL of the **hm-recommender-service** (from Task 3). (e.g., <https://hm-recommender-service-xyz.us-central1.run.app>)
- **Search by ID:** Paste the User ID you noted down earlier. (e.g., 499f785cb034c252c3cc64e15c90a9e13602a3fe45611193c221ec7ac4794463)



4. **Note:** If the backend service has been idle, you might experience a "Cold Start" delay of 10-20 seconds for the first request.
5. **View Results:**
 - **Top:** You will see the Top 12 Recommended Items. Items that the user actually purchased in the validation week are marked with a green "HIT" tag.



- **Bottom:** You will see the actual items purchased by the user.



Congratulations! You have successfully built and deployed a Two-Tower recommendation system on Google Cloud.

Task 6: Cleanup

To avoid incurring charges to your Google Cloud account for the resources used in this tutorial, verify that you delete all generated resources.

1. Return to **Cloud Shell**.
2. Navigate to the home directory:

```
Shell  
cd ..
```

3. Download and run the cleanup script:

```
Shell  
wget https://raw.githubusercontent.com/hilmi-collab/HM_Recommendation_System_on_GCP/main/cleanup_workshop.sh  
chmod +x cleanup_workshop.sh  
./cleanup_workshop.sh
```

4. Type **y** when prompted to confirm deletion. Wait until you see the confirmation message indicating resources have been deleted.

```
2025-12-11 12:28:39 (37.1 MB/s) - 'cleanup_workshop.sh' saved [4927/4927]

=====
H&M Workshop - RESOURCE CLEANUP
=====

INFO: This process will NOT delete your GCP Project. It only cleans up workshop resources.

Your active configuration is: [cloudshell-4623]
Project to Process: ace-team-hilmi-service (Project will be PRESERVED)
Bucket to Delete: hm-workshop-ace-team-hilmi-service

Do you want to clean up the resources? (y/n): █

[4/5] Cleaning Cloud Storage Bucket...
Removing gs://hm-workshop-ace-team-hilmi-service/models/#1765449133801990...
Removing gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/app_articles_features.parquet#1765451820888884...
Removing gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/app_customers_features.parquet#1765451824266209...
Removing gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/app_stats_int.parquet#1765451827529345...
Removing gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/app_user_history_int.parquet#1765451825990178...
Removing gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/article_map.pkl#1765451816144517...
Removing gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/customer_map.pkl#1765451819283521...
Removing gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/model.model#1765451831695235...
Removing gs://hm-workshop-ace-team-hilmi-service/models/ranking_model/val_truth.parquet#1765451829283315...
Removing gs://hm-workshop-ace-team-hilmi-service/models/two-tower-model/#1765449134105342...
Removing gs://hm-workshop-ace-team-hilmi-service/models/two-tower-model/assets#1765449140995467...
Removing gs://hm-workshop-ace-team-hilmi-service/models/two-tower-model/fingerprint.pb#1765449145268284...
Removing gs://hm-workshop-ace-team-hilmi-service/models/two-tower-model/saved_model.pb#1765449143177067...
Removing gs://hm-workshop-ace-team-hilmi-service/models/two-tower-model/variables/#1765449134394717...
Removing gs://hm-workshop-ace-team-hilmi-service/models/two-tower-model/variables/variables.data-00000-of=00001#1765449139986641...
Removing gs://hm-workshop-ace-team-hilmi-service/models/two-tower-model/variables/variables.index#1765449140175387...
Removing gs://hm-workshop-ace-team-hilmi-service/notebooks/hm_ranking_lightgbm_training.ipynb#1765447507612007...
Removing gs://hm-workshop-ace-team-hilmi-service/notebooks/hm_two_tower_training.ipynb#1765447502713281...
/ [18/18 objects] 100% Done
Operation completed over 18 objects.
Removing gs://hm-workshop-ace-team-hilmi-service/...
✓ Bucket and all contents deleted.
[5/5] Cleaning local files...
✓ Cloud Shell working directory cleaned.

===== CLEANUP COMPLETED (PROJECT ACTIVE) ✓ =====

hilmi_emel@cloudshell:~ (ace-team-hilmi-service)$ █
```