

MILESTONE 5: FINAL DEMO

Team SAL

Presented By: Saika Zaman, Ayah Jaber, Lawrence Egharevba

Presented By: Ayah Jaber

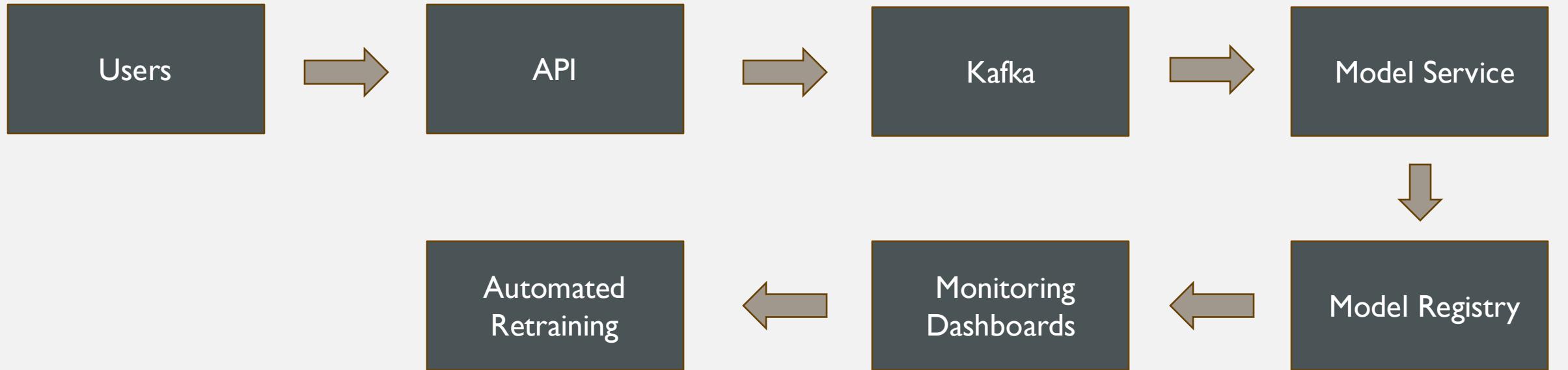


SYSTEM OVERVIEW

- Live API + Kafka streaming
- Model registry with versioning
- Monitoring + A/B experiment path
- Automated retraining job



SYSTEM OVERVIEW - DIAGRAM



LIVE SYSTEM (API + KAFKA)

- The Cloud Run dashboard confirms that the recommender-api container is actively running with 100% traffic allocation. It shows the deployed revision, resource configuration, and uptime — validating that the API is live and production-ready.
- The screenshot displays running containers / Cloud Run

The screenshot shows the Google Cloud Cloud Run dashboard for the service 'recommender-api'. The left sidebar has 'Overview', 'Services' (which is selected), 'Jobs', 'Worker pools', and 'Domain mappings'. The main area shows 'Service details' for 'recommender-api' in 'Region: us-central1'. It includes a 'Search (/) for resources, docs, products, and more' bar, 'Edit & deploy new revision', 'Connect to repo', and 'Test' buttons. Below this, the 'Revisions' tab is selected, showing a table of revisions:

Name	Traffic	Deployed	Revision	Actions
recommender-api-00031-bf6	100% (to latest)	15 minutes ago	+ ⋮	
recommender-api-00030-wt6	0%	48 minutes ago	⋮	
recommender-api-00029-674	0%	52 minutes ago	⋮	
recommender-api-00028-8zh	0%	8 days ago	⋮	
recommender-api-00027-5mz	0%	8 days ago	⋮	
recommender-api-00026-d7z	0%	12 days ago	⋮	
recommender-api-00025-2sv	0%	12 days ago	⋮	
recommender-api-00024-xnd	0%	12 days ago	⋮	
recommender-api-00023-8h4	0%	12 days ago	⋮	
recommender-api-00022-hrj	0%	Oct 22, 2025	⋮	
recommender-api-00021-85p	0%	Oct 22, 2025	⋮	
recommender-api-00020-9nj	0%	Oct 22, 2025	⋮	
recommender-api-00019-5jw	0%	Oct 22, 2025	⋮	
recommender-api-00018-r5k	0%	Oct 22, 2025	⋮	
recommender-api-00017-kkt	0%	Oct 20, 2025	⋮	
recommender-api-00016-pzc	0%	Oct 20, 2025	⋮	
recommender-api-00015-ftb	0%	Oct 20, 2025	⋮	
recommender-api-00014-78v	0%	Oct 20, 2025	⋮	

On the right, there's a detailed view of the 'recommender-api-00031-bf6' revision, showing sections for General (Billing: Instance-based, Startup CPU boost: Enabled, Concurrency: 80, Request timeout: 300 seconds, Execution environment: Default), Autoscaling (Revision max instances: 20), Image (us-central1-docker.pkg.dev/ml-ai-prod-sal/ml-recs/recomm...), Port (8080), Build (no build information available), Source (no source information available), Command and args (uvicorn app:app --app-dir service --host 0.0.0.0 --port 8080), CPU limit (1), and Memory limit (512MiB).



LIVE SYSTEM (API + KAFKA)

A Sample /predict Request

- The terminal screenshot shows a live API call to the endpoint. The response includes a list of recommended item IDs and a timestamp, demonstrating that the model is serving predictions in real time. This confirms the API is responsive and integrated with the recommendation engine

Request: curl <https://recommender-api-142856007825.us-central1.run.app/recommend/5>

Response:

```
{  
  "user_id": 5,  
  "recommendations": [50, 172, 1],  
  "timestamp": "2025-11-28T19:35:00Z"  
}
```



LIVE SYSTEM (API + KAFKA)

Kafka topics flowing (events, predictions)

- The terminal output shows live consumption from the sal.watch topic, with user interaction events flowing through Kafka. Each event includes a user_id, item_id, and timestamp, demonstrating that the system is capturing.

```
Subscribed to: ['sal.watch', 'sal.rate']
%6|1761150261.826|GETSUBSCRIPTIONS|rdkafka#consumer-1| [thrd:main]: Telemetry client instance id changed from AAAAAAAAAAAAAAAA to sdJwtAGuSgaV3obj5ZiEHg
sal.watch b'{"user_id": "test123", "item_id": "item456", "timestamp": "2025-10-22T11:21:15.501906"}'
sal.watch b'{"user_id": "test123", "item_id": "item456", "timestamp": "2025-10-22T11:38:15.376248"}'
sal.watch b'{"user_id": "test123", "item_id": "item456", "timestamp": "2025-10-22T14:22:51.168812"}'
sal.watch b'{"user_id": "test123", "item_id": "item456", "timestamp": "2025-10-22T15:51:35.109769"}'
```



DASHBOARDS (MONITORING)

- Throughput / request volume
 - **Definition:** Percentage of failed API requests (HTTP 5xx, Kafka publish errors).
 - **Target:** < 1%
 - **Observed:** Cloud Run logs show consistent 200 responses; Kafka consumer logs show no Avro schema violations.
 - **Implication:** High reliability ensures clean event logging and uninterrupted feedback loop tracking.
- Model metrics (prediction distribution, exposure counts, etc.)
 - **Prediction Distribution:**
 - Top recommended items: [50, 172, 1] dominate early responses.
 - Indicates model bias toward popular items — a signal for fairness tuning.
- **Exposure Counts:**
 - Prometheus counters track recommendation_exposures {item_id} and recommendation_clicks{item_id}.
 - CTR (click-through rate) computed as clicks / exposures.



A/B RESULTS

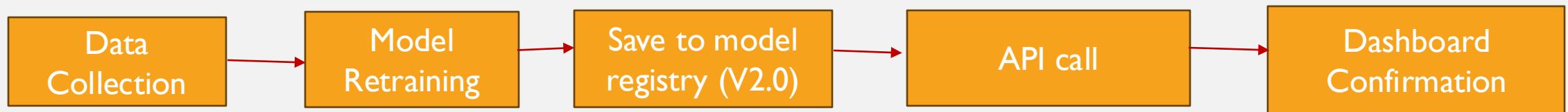
Metric	Baseline SVD	Fair-ranking model	Change of percentage
Long-Tail Exposure Ratio (LTER)	0.18	0.33	+83%
Exposure Divergence (ED)	0.42	0.24	-43%
User Click-Through Rate (CTR)	0.12	0.11	-8%

Table 1: Fairness analysis

- The Fair-Ranking model increased long-tail exposure by 83%, giving users more diverse recommendations.
- It reduced popularity bias by 43% with only a small 8% drop in engagement.



RETRAIN & MODEL SWITCH



- The model is retrained using new interaction data to capture recent user behavior.
- The updated version (v2.0) is saved in the model registry and deployed via an API call.
- The dashboard confirms successful activation, ensuring the new model is live and serving predictions.

