

Notify

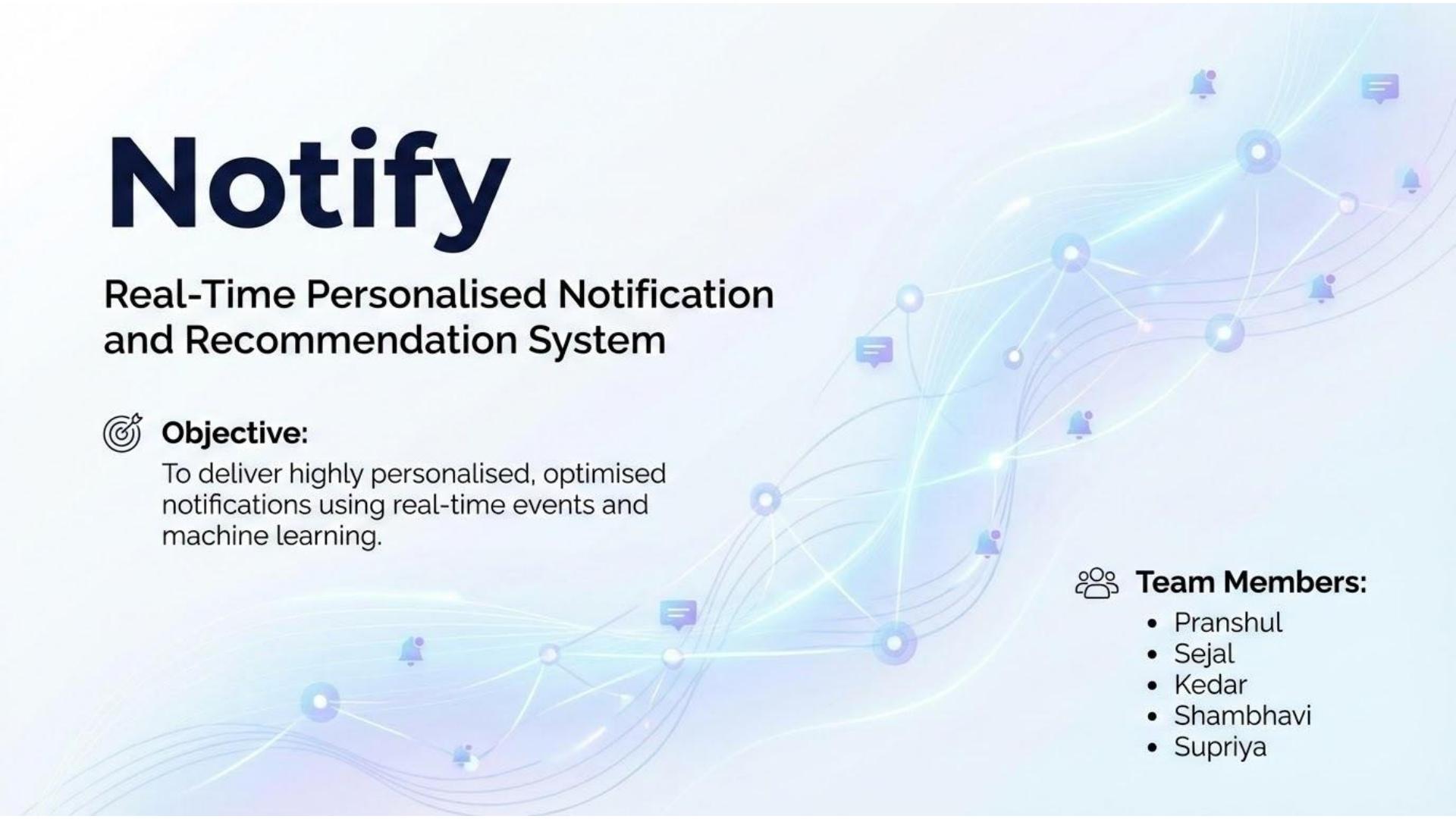
Real-Time Personalised Notification
and Recommendation System

🎯 Objective:

To deliver highly personalised, optimised notifications using real-time events and machine learning.

👥 Team Members:

- Pranshul
- Sejal
- Kedar
- Shambhavi
- Supriya



Problem Statement

-  Users receive generic and poorly timed notifications
-  No real-time personalisation
-  Rule-based targeting does not scale
-  No feedback-driven learning loop

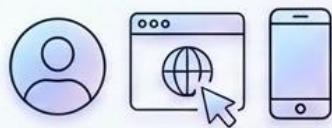


Goal:

Build a **scalable, ML-driven, real-time recommendation and notification system** with optimised delivery.

Frontend & Event Ingestion

User interacts via
Web/App frontend



Events
generated:

- Orders
- Clicks
- Campaign views

Kafka Topics:

- Orders Topic
- Events Topic

Stored in:

- `purchase_history`
- `notify_clickstream`

Batch Ingestion

Target Tables

-  **notify_purchase_history** – Historical transactions
-  **notify_user_features** – Enriched user profiles & behaviour
-  **notify_products** – Core product catalogue
-  **notify_product_features** – Product performance metrics
-  **notify_campaigns** – Campaign metadata
-  **notify_user_product_matrix** – User–product interaction matrix

Tech Stack:



**Amazon
DynamoDB**

Why DynamoDB:

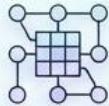


Low-latency at scale



Serverless auto-scaling

Core & Feature Table Schemas



1. notify_user_product_matrix (Behavior Matrix)

user_id (PK)
product_id
last_purchase_date
purchased_before
times_purchased

Purpose: Captures user-product interaction strength for recommendation and similarity.



2. notify_purchase_history (Transactions)

purchase_id
user_id (PK)
product_id
purchase_date
price
quantity
total_amount

Purpose: Stores raw historical transactions for analytics and embedding generation.



3. notify_campaigns (Campaign Metadata)

campaign_id (PK)
created_at
name
description
notification_type
priority
start_date, end_date
status
template_short,
template_long

Purpose: Defines all notification campaigns and delivery content.



4. notify_products (Core Product Catalog)

product_id (PK)
articleType
baseColour
gender
image_url
masterCategory
price
productDisplayName
season
subCategory
usage
year

Purpose: Master product reference table for the system.

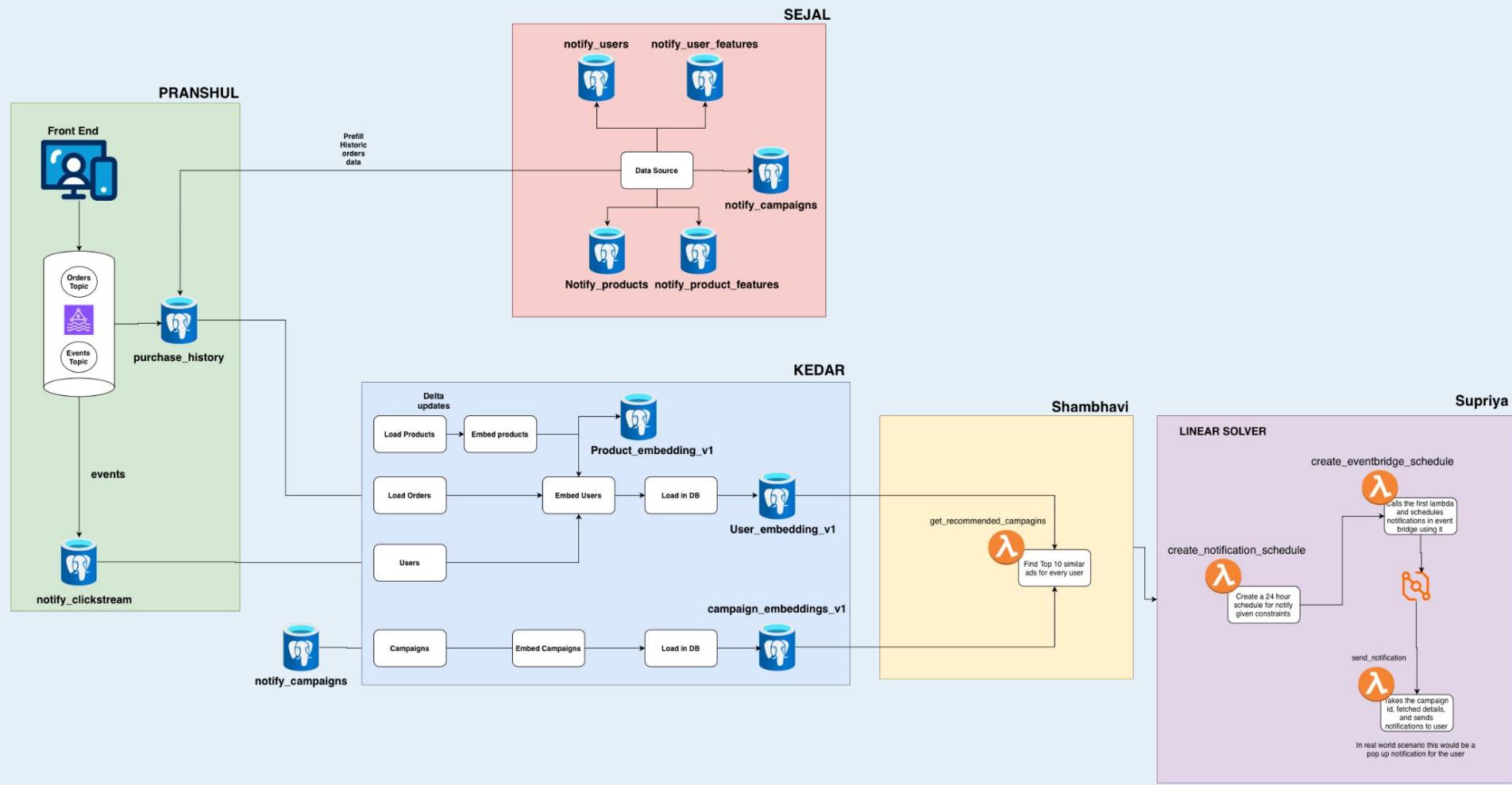


5. notify_product_features (Product Feature Store)

product_id (PK)
articleType
avg_price
baseColour
gender
image_url
last_sold_date
masterCategory
price
productDisplayName
revenue
season
subCategory
total_sales
usage
year

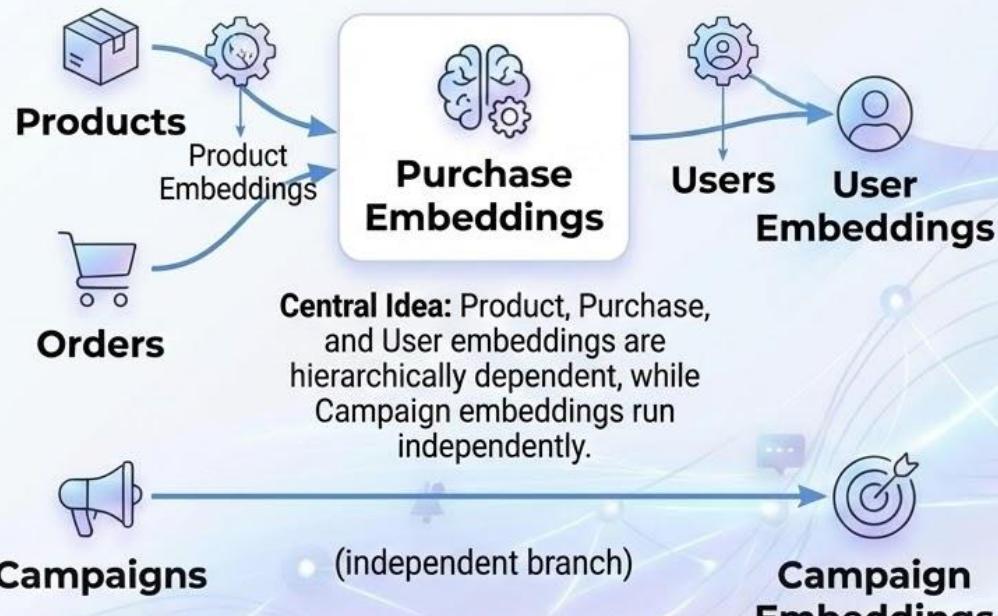
Purpose: Stores derived product performance features used for embeddings and targeting.

High-Level System Overview



ML Pipeline Orchestration in Airflow

Flow Logic



Execution Design

- Product and order pipelines converge at **Purchase Embeddings**
- User Embeddings are built downstream of **Purchase Embeddings**
- Product Embeddings are a first-class output
- Campaign embeddings run in **parallel**
- Ensures temporal consistency and maximum pipeline parallelism

Live System Validation: AWS MWAA



Items | Amazon DynamoDB Management Console | DynamoDB | us-east-1

du-mids (8814-9221-2023) Admin_role/kv133

DynamoDB > Explore items > notify_purchase_embeddings_v2

Autopreview View table details

notify_purchase_embeddings_v2

▼ Scan or query items

Scan Query

Select a table or index Table - notify_purchase_embeddings_v2 Select attribute projection All attributes

▶ Filters - optional

Run Reset

⚠️ Stopped - Items returned: 50 - Items scanned: 50 - Efficiency: 100% - RCU consumed: 25

Retrieve next page

Table: notify_purchase_embeddings_v2 - Items returned (50)

Actions Create item

Scan started on December 05, 2025, 20:51:46

	purchase_id (String)	embeddi...	updated_at
<input type="checkbox"/>	b9a88db1-1c97-4403-b6...	[{"N": "0.1..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	d5521ff3-23e2-4f61-a43...	[{"N": "0.8..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	5e2b97ad-6f7b-43f1-9c...	[{"N": "0.1..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	2e9bd95-25d8-4e14-8...	[{"N": "0.6..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	b3e42350-07e0-4ad9-af...	[{"N": "0.3..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	baa0d23e-46ad-4c25-ad...	[{"N": "0.5..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	d7158fff-6a77-4d80-9b...	[{"N": "0.2..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	81ba87b8-469a-4ec4-9b...	[{"N": "0.8..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	1cb8ae03-b019-4336-8e...	[{"N": "0.9..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	57e8b50d-8bc7-4696-83...	[{"N": "0.3..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	c517f051-b9f3-4ac6-baa...	[{"N": "0.8..."}]	2025-12-06T01:51:20.505585
<input type="checkbox"/>	f4057c51-a11c-45d0-b1...	[{"N": "0.9..."}]	2025-12-06T01:51:20.505585

Items | Amazon DynamoDB Management Console | DynamoDB | us-east-1

du-mids (8814-9221-2023) Admin_role/kv133

DynamoDB > Explore items > notify_campaign_embeddings_v2

Autopreview View table details

notify_campaign_embeddings_v2

▼ Scan or query items

Scan Query

Select a table or index Table - notify_campaign_embeddings_v2 Select attribute projection All attributes

▶ Filters - optional

Run Reset

⚠️ Stopped - Items returned: 50 - Items scanned: 50 - Efficiency: 100% - RCU consumed: 25

Retrieve next page

Table: notify_campaign_embeddings_v2 - Items returned (50)

Actions Create item

Scan started on December 05, 2025, 20:53:15

	campaign_id (String)	embeddi...	updated_at
<input type="checkbox"/>	C44	[{"N": "0.3..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C53	[{"N": "0.5..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C50	[{"N": "0.9..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C37	[{"N": "0.5..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C13	[{"N": "0.9..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C7	[{"N": "0.2..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C45	[{"N": "0.1..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C5	[{"N": "0.1..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C43	[{"N": "0.0..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C3	[{"N": "0.3..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C23	[{"N": "0.0..."}]	2025-12-06T01:56:34.254611
<input type="checkbox"/>	C27	[{"N": "0.3..."}]	2025-12-06T01:56:34.254611

Items | Amazon DynamoDB Management Console | DynamoDB | us-east-1

du-mids (8814-9221-2023) Admin_role/kv133

DynamoDB > Explore items > notify_user_embeddings_v2

Autopreview View table details

notify_user_embeddings_v2

▼ Scan or query items

Scan Query

Select a table or index Table - notify_user_embeddings_v2 Select attribute projection All attributes

▶ Filters - optional

Run Reset

⚠️ Stopped - Items returned: 50 - Items scanned: 50 - Efficiency: 100% - RCU consumed: 44

Retrieve next page

Table: notify_user_embeddings_v2 - Items returned (50)

Actions Create item

Scan started on December 05, 2025, 20:53:21

	user_id (String)	embeddi...	updated_at
<input type="checkbox"/>	U0023	[{"N": "0.7..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0083	[{"N": "0.7..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0079	[{"N": "0.3..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0062	[{"N": "0.3..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0057	[{"N": "0.8..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0053	[{"N": "0.8..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0031	[{"N": "0.5..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0076	[{"N": "0.1..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0009	[{"N": "0.7..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0020	[{"N": "0.7..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0058	[{"N": "0.9..."}]	2025-12-06T01:51:45.506972
<input type="checkbox"/>	U0001	[{"N": "0.9..."}]	2025-12-06T01:51:45.506972

CloudShell Feedback

Privacy Terms Cookie preferences

© 2025, Amazon Web Services, Inc. or its affiliates.

CloudShell Feedback

Privacy Terms Cookie preferences

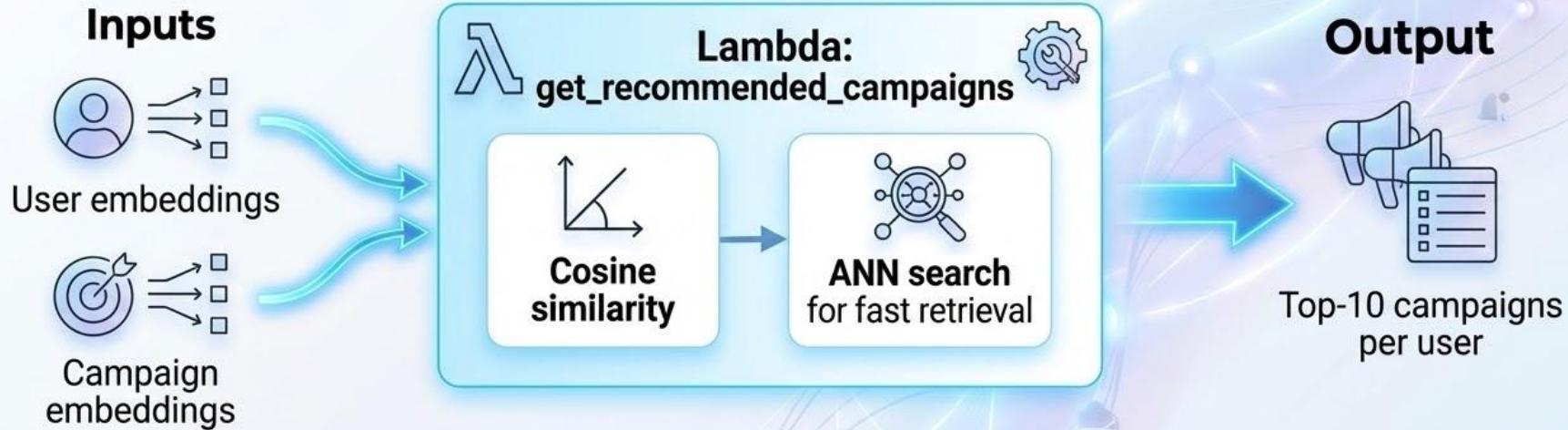
© 2025, Amazon Web Services, Inc. or its affiliates.

CloudShell Feedback

Privacy Terms Cookie preferences

© 2025, Amazon Web Services, Inc. or its affiliates.

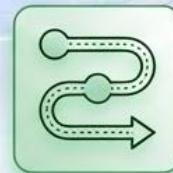
Real-Time Recommendation Engine



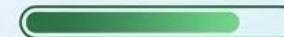
Complexity



Naive $O(n)$



Optimised $O(\log n)$ with ANN



aws [Lambda](#) > [Functions](#) > **sagemaker-inference-test**

Function overview

[Export to Infrastructure Composer](#) [Download](#)

template-diagram control

Diagram

+ Add destination

+ Add trigger

Last run: 4 minutes ago

Function code: 1,381 bytes

Function configuration: 0 bytes

Code SHA-256: 15WSlWmMC54giEqgy2NQcy+rR4cMT22lVH4EzuzSKc=

Test event

[Info](#) [Delete](#) [CloudWatch Logs](#) [Live Tail](#) [Save](#) [Test](#)

To invoke your function without saving an event, modify the event, then choose Test. Lambda uses the modified event to invoke your function, but does not overwrite the original event until you choose Save.

Feedback [Console Mobile App](#) Privacy Terms Cookie preferences
© 2025, Amazon Web Services, Inc. or its affiliates.

aws [Lambda](#) > [Functions](#) > **sagemaker-inference-test**

Executing function: succeeded (logs [View](#))

Details

```
{ "statusCode": 200, "body": "{\"user_id\": \"U0023\", \"top_recommendation\": {\"campaign_id\": \"C004\", \"similarity_score\": 0.1353, \"name\": \"New Feature\", \"description\": \"Notify users about a new feature release.\", \"campaign_data\": {\"campaign_name\": \"New Feature\", \"created_at\": \"2025-11-15T14:00:00Z\", \"end_date\": \"2026-11-20\", \"start_date\": \"2025-11-20\", \"campaign_embedding\": [0.1442, -0.0118, -0.0993, 0.0422, 0.2022, 0.0022, 0.0911, -0.0777, 0.1881, -0.0053, -0.2111, 0.1321, -0.0188, 0.0577], \"status\": \"active\", \"priority\": 1.0, \"description\": \"Notify users about a new feature release.\", \"campaign_id\": \"C004\", \"notification_type\": \"update\", \"template_long\": \"We\\u2019ve added something new! Log in and\"}}
```

Summary

Code SHA-256: 15WSlWmMC54giEqgy2NQcy+rR4cMT22lVH4EzuzSKc= Execution time: 4 minutes ago

Function version: \$LATEST Request ID: 335509e4-0b1f-48c7-a178-65bc7242b438

Duration: 84.51 ms Billed duration: 85 ms

Resources configured: 128 MB Max memory used: 82 MB

Log output

The area below shows the last 4 KB of the execution log. [Click here](#) to view the corresponding CloudWatch log group.

```
START RequestId: 335509e4-0b1f-48c7-a178-65bc7242b438 Version: $LATEST
END RequestId: 335509e4-0b1f-48c7-a178-65bc7242b438
```

Feedback [Console Mobile App](#) Privacy Terms Cookie preferences
© 2025, Amazon Web Services, Inc. or its affiliates.

Scheduling & Optimisation Layer



Notification Delivery System



[Option+S] Account ID: 3814-9221-2823 ▾
mids_27_supriya

schedule_notifications

PROBLEMS OUTPUT CODE REFERENCE LOG TERMINAL Execution Results

Response:

```
{ "statusCode": 200, "body": "\"Daily schedule created successfully\""} 
```

The area below shows the last 4 KB of the execution log.

Function Logs:

```
START RequestId: 3f3a971b-efab-4187-8cb6-d8e1402a6282 Version: $LATEST
[✓] FINAL SCHEDULE for U0023:
C004 → 2025-12-05T00:00:00Z
C001 → 2025-12-05T04:48:00Z
C005 → 2025-12-05T09:36:00Z
C002 → 2025-12-05T14:24:00Z
C003 → 2025-12-05T19:12:00Z
[✓] FINAL SCHEDULE for U001:
C003 → 2025-12-05T00:00:00Z
C005 → 2025-12-05T04:48:00Z
C002 → 2025-12-05T09:36:00Z
C004 → 2025-12-05T14:24:00Z
C001 → 2025-12-05T19:12:00Z
[✓] FINAL SCHEDULE for U003:
C004 → 2025-12-05T00:00:00Z
C001 → 2025-12-05T04:48:00Z
C005 → 2025-12-05T09:36:00Z
C003 → 2025-12-05T14:24:00Z
C002 → 2025-12-05T19:12:00Z
END RequestId: 3f3a971b-efab-4187-8cb6-d8e1402a6282
REPORT RequestId: 3f3a971b-efab-4187-8cb6-d8e1402a6282 Duration: 927.00 ms Billed Duration: 1462 ms Memory Size: 128 MB Max Memory Used: 90 MB Init Duration: 534.70 ms
```

Request ID: 3f3a971b-efab-4187-8cb6-d8e1402a6282

Success [create_notification_schedule]

Ln 85, Col 34 Spaces: 4 UTF-8 LF Python Lambda Layout: U.S.

Daily schedule successfully generated with timestamps — capped at 5 notifications per user.

[Option+S]

United States (N. Virginia) Account ID: 3814-9221-2823

mids_27_supriya

Execution Results

PROBLEMS OUTPUT CODE REFERENCE LOG TERMINAL

Response:

```
{  
  "statusCode": 200,  
  "body": "{\"user_id\": \"U001\", \"campaign_id\": \"C002\", \"title\": \"Welcome Offer\", \"short_text\": \"\\ud83c\\udf89 Welcome! Enjoy 10% off!\",  
  \"long_text\": \"Welcome aboard! As a thank you for joining, here's an exclusive 10% off on your first purchase.\", \"priority\": 4.0, \"notification_type\":  
  \"welcome\", \"sent_epoch\": 1764966787, \"sent_utc\": \"2025-12-05T20:33:07Z\"}"  
}
```

The area below shows the last 4 KB of the execution log.

Function Logs:

```
START RequestId: 63b1e656-d6b4-4494-a241-dda8dc716c31 Version: $LATEST  
=====  
⚠ send_notification invoked  
=====  
⭐ RAW EVENT - {"user_id": "U001"}  
⚠ Date: 2025-12-05 | ⚡ Now (epoch): 1764966787  
⚠ Schedule Loaded: [{"sent": true, "campaign_id": "C003", "timestamp": '2025-12-05T00:00:00Z'}, {"sent": true, "campaign_id": "C005", "timestamp": '2025-12-05T04:48:00Z'}, {"sent": false, "campaign_id": "C002", "timestamp": '2025-12-05T09:36:00Z'}, {"sent": false, "campaign_id": "C004", "timestamp": '2025-12-05T14:24:00Z'}, {"sent": false, "campaign_id": "C001", "timestamp": '2025-12-05T19:12:00Z'}]  
⌚ MATCH → campaign=C002 | scheduled=2025-12-05T09:36:00Z  
✓ Notification marked as sent  
⌚ FINAL PAYLOAD:  
{"user_id": "U001",  
 "campaign_id": "C002",  
 "title": "Welcome Offer",  
 "short_text": "\ud83c\udf89 Welcome! Enjoy 10% off!",  
 "long_text": "Welcome aboard! As a thank you for joining, here's an exclusive 10% off on your first purchase.",  
 "priority": 4.0,  
 "notification_type": "welcome",  
 "sent_epoch": 1764966787,  
 "sent_utc": "2025-12-05T20:33:07Z"}  
}
```

[create_eventbridge_schedule]

Lambda identifies the next pending notification and prepares the delivery payload.

AWS Search [Option+S] United States (N. Virginia) Account ID: 3814-9221-2823 mids_27_supriya

Amazon EventBridge > Schedules > daily_create_notification_schedule

Schedule detail

Schedule name daily_create_notification_schedule	Status Enabled	Schedule start time -	Flexible time window -
Description -	Schedule ARN arn:aws:scheduler:us-east-1:381492212823:schedule/default/daily_create_notification_schedule	Schedule end time -	
Schedule group name default	Action after completion NONE	Execution time zone Asia/Calcutta	

Schedule Target Retry policy Dead-letter queue Encryption

Schedule

Cron expression [Info](#)

0	0	*	*	?	*
Minutes	Hours	Day of month	Month	Day of week	Year

[Copy cron expression](#)

Next 10 trigger dates
Date and time are displayed in the selected time zone for which this schedule is set in UTC

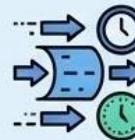
CloudShell Feedback Console Mobile App © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

EventBridge automation triggers notification delivery every minute —
fully serverless and hands-free.

Technology Stack



Frontend
Python



Real Time Data
Kafka



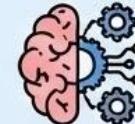
Batch Ingestion
Glue



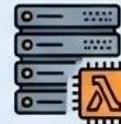
Databases
DynamoDB



Orchestration
Amazon Managed Airflow



ML
BERT Embeddings, ANN Search



Compute
Lambda, EC2



Scheduling
EventBridge



THANK YOU!



Batch Ingestion

Target Tables

- `purchase_history` – Historical orders
- `notify_users` – User profiles
- `notify_products` – Product catalogue
- `notify_campaigns` – Campaign metadata

Tech Stack:

- Amazon DynamoDB

Why DynamoDB:

- Low-latency at scale
- Serverless auto-scaling

Outcome:

Stable historical backbone for all ML pipelines.

Core & Feature Table Schemas

1. notify_user_product_matrix (Behavior Matrix)

```
• user_id (PK)  
• product_id  
• last_purchase_date  
• purchase_before  
• times_purchased
```

Purpose: Captures user-product interaction strength for recommendation and similarity.

2. notify_purchase_history (Transactions)

```
• purchase_id  
• user_id (PK)  
• product_id  
• purchase_date  
• price  
• quantity  
• total_amount
```

Purpose: Stores raw historical transactions for analytics and embedding generation.

3. notify_campaigns (Campaign Metadata)

```
• campaign_id (PK)  
• created_at  
• name  
• description  
• notification_type  
• priority  
• start_date, end_date  
• status  
• template_short, template_long
```

Purpose: Defines all notification campaigns and delivery content.

4. notify_products (Core Product Catalog)

```
• product_id (PK)  
• articleType  
• baseColour  
• gender  
• image_url  
• masterCategory  
• price  
• productDisplayName  
• season  
• subCategory  
• usage  
• year
```

Batch Ingestion

Target Tables

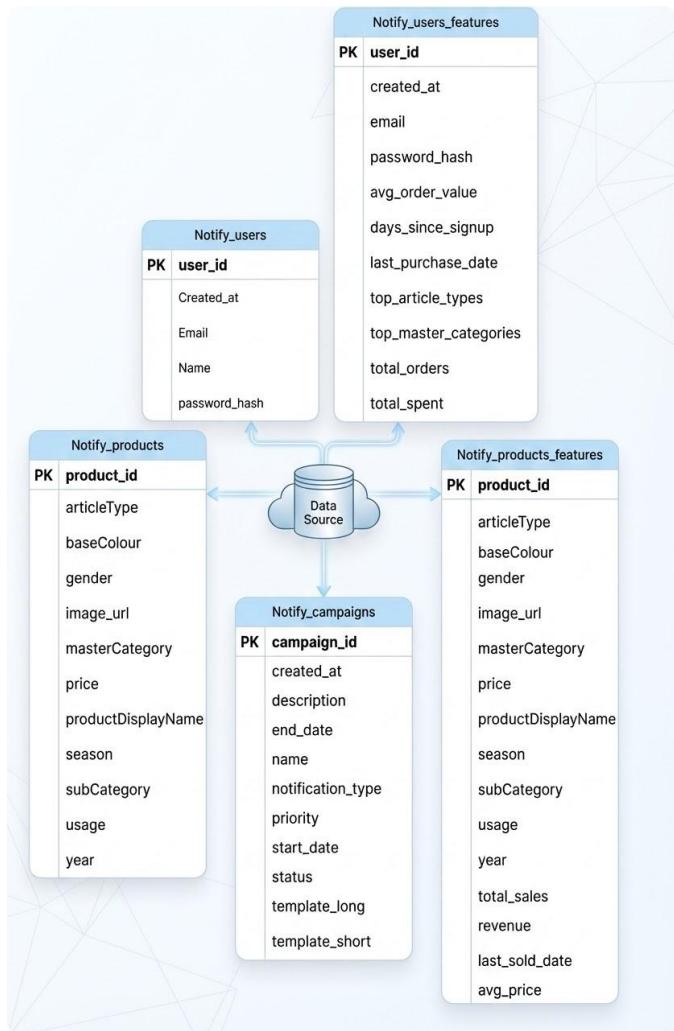
-  **notify_purchase_history** – Historical transactions
-  **notify_user_features** – Enriched user profiles & behaviour
-  **notify_products** – Core product catalogue
-  **notify_product_features** – Product performance metrics
-  **notify_campaigns** – Campaign metadata
-  **notify_user_product_matrix** – User–product interaction matrix

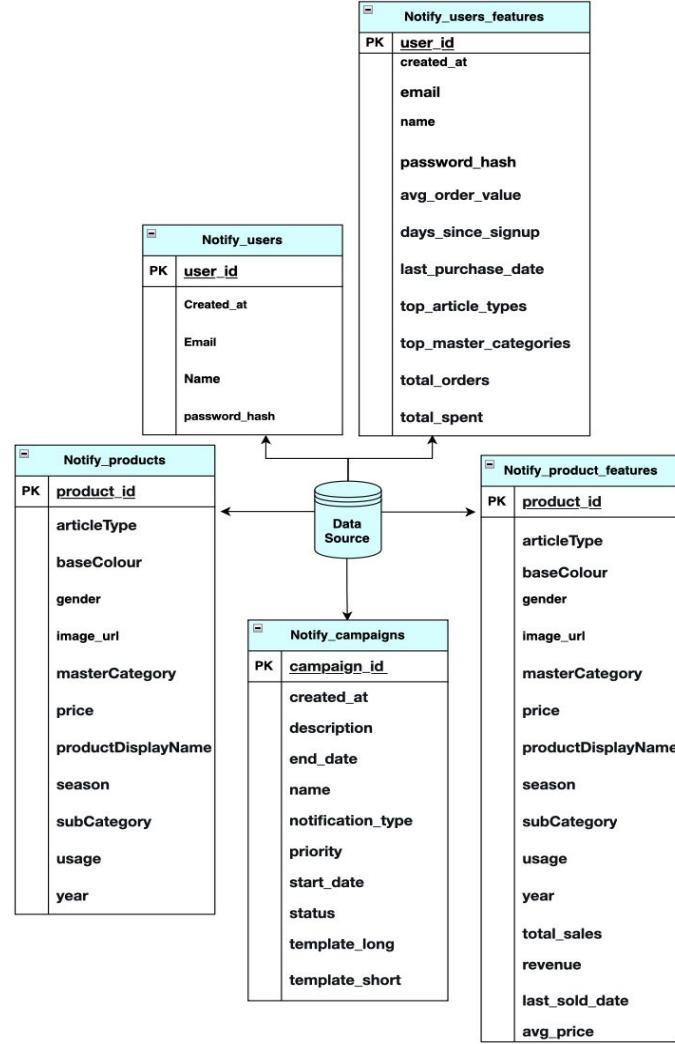
Tech Stack

Amazon DynamoDB

Why DynamoDB

-  Low-latency at scale
-  Serverless auto-scaling





Notification Delivery System

Lambda: `send_notification`

- Pulls scheduled campaign
- Fetches user context
- Sends notification
- User interaction flows back to clickstream

Result: Closed-loop learning system.

Notify_users (Core User Table)

Primary Key: user_id

Fields:

- created_at - account creation timestamp
- email - user email
- name - user name
- password_hash - encrypted password

Purpose:
Stores core authentication and identity information for each user.

Notify_users_features (User Feature Store)

Primary Key: user_id

Fields:

- created_at
- email
- password_hash
- avg_order_value
- avg_order_group
- last_purchase_date
- top_article_types
- top_product_categories
- total_orders
- total_spent

Purpose:
Stores derived behavioural features used for:

- User profiling
- Embedding generation
- Recommendation logic

Notify_products (Core Product Table)

Primary Key: product_id

Fields:

Scheduling & Optimisation Layer

Key Lambdas:

- `create_eventbridge_schedule`
- `create_notification_schedule`

Constraints handled:

- User fatigue
- Time windows
- Campaign priority
- Engagement optimisation

Output: 24-hour optimised notification plan.

Real-Time Recommendation Engine

Lambda: `get_recommended_campaigns`

- Inputs:
 - User embeddings
 - Campaign embeddings
- Processing:
 - Cosine similarity
 - ANN search for fast retrieval
- Output:
 - Top-10 campaigns per user

Complexity:

Naive $O(n)$ → Optimised $O(\log n)$ with ANN

Technology Stack

Frontend: Python

Real Time Data: Kafka

Batch Ingestion: Glue

Databases: DynamoDB

Orchestration: Amazon Managed Airflow

ML: BERT Embeddings, ANN Search

Compute: Lambda, EC2

Scheduling: EventBridge

ML Pipeline Orchestration in Airflow

Central Idea

Product, Purchase, and User embeddings are **hierarchically dependent**, while Campaign embeddings run independently.

Flow Logic

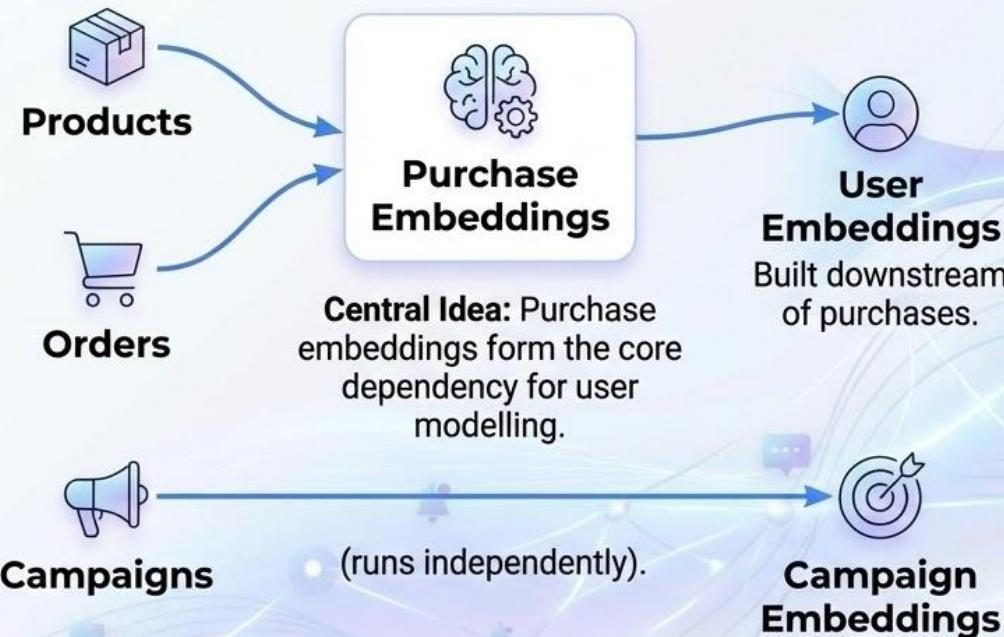
- Products → **Product Embeddings**
- Orders + Product Embeddings → **Purchase Embeddings**
- Users + Purchase Embeddings → **User Embeddings**
- Campaigns → **Campaign Embeddings** (independent branch)

Execution Design

- Product and order pipelines converge at **Purchase Embeddings**
- **User Embeddings are built downstream of Purchase Embeddings**
- **Product Embeddings are a first-class output**
- Campaign embeddings run in **parallel**
- Ensures **temporal consistency and maximum pipeline parallelism**

ML Pipeline Orchestration in Airflow

Flow Logic



Execution Design

- Product and order pipelines converge at purchase embeddings
- User embeddings are built downstream of purchases
- Campaign embeddings run in parallel
- Ensures temporal consistency and maximum pipeline parallelism

Frontend & Event Ingestion

User interacts via Web/App frontend

Events generated:

- Orders
- Clicks
- Campaign views

Kafka Topics:

- Orders Topic
- Events Topic

Stored in:

- `purchase_history`
- `notify_clickstream`

Objective:

To deliver highly personalised, optimised notifications using real-time events and machine learning.

Notify

Real-Time Personalised Notification and Recommendation System

Team Members:

Pranshul, Sejal, Kedar, Shambhavi, Supriya

Problem Statement

Users receive generic and poorly timed notifications

No real-time personalisation

Rule-based targeting does not scale

No feedback-driven learning loop

Goal:

Build a **scalable, ML-driven, real-time recommendation and notification system** with optimised delivery.

aws | Lambda | Functions | sagemaker-inference-test | Account ID: 381... | mids_27_shambhav...

Function overview | Info | Export to Infrastructure Composer | Download

template-diagram control

Diagram

+ Add destination

+ Add trigger

Code | Test | Monitor | Configuration | Aliases | >

Test event | Info | Delete | CloudWatch Logs Live Tail | Save | Test

To invoke your function without saving an event, modify the event, then choose Test. Lambda uses the modified event to invoke your function, but does not overwrite the original event until you choose Save.

Feedback | Console Mobile App | Privacy | Terms | Cookie preferences

© 2025, Amazon Web Services, Inc. or its affiliates.

aws | Lambda | Functions | sagemaker-inference-test | Account ID: 3814-9221-2823 | mids_27_shambhav...

Executing function: succeeded (logs)

Details

```
{ "statusCode": 200, "body": "{\"user_id\": \"U0023\", \"top_recommendation\": {\"campaign_id\": \"C004\", \"similarity_score\": 0.1353, \"name\": \"New Feature\", \"description\": \"Notify users about a new feature release.\", \"campaign_data\": {\"campaign_name\": \"New Feature\", \"created_at\": \"2025-11-15T14:00:00Z\", \"end_date\": \"2026-11-20\", \"start_date\": \"2025-11-20\", \"campaign_embedding\": [0.1442, -0.0118, -0.0993, 0.0422, 0.2022, 0.0022, 0.0311, -0.0777, 0.1881, -0.0053, -0.2111, 0.1321, -0.0188, 0.0577], \"status\": \"active\", \"priority\": 1, \"description\": \"Notify users about a new feature release.\", \"campaign_id\": \"C004\", \"notification_type\": \"update\", \"template_long\": \"We\\u2019ve added something new! Log in and\"}}
```

Summary

Code SHA-256 15WSIWmMC54g!Eggy2NQcy+rR4cMT22lVH4EzuzS+Kc=	Execution time 4 minutes ago
Function version \$LATEST	Request ID 335509e4-0b1f-48c7-a178-65bc7242b438
Duration 84.51 ms	Billed duration 85 ms
Resources configured 128 MB	Max memory used 82 MB

Log output

The area below shows the last 4 KB of the execution log. [Click here](#) to view the corresponding CloudWatch log group.

```
START RequestId: 335509e4-0b1f-48c7-a178-65bc7242b438 Version: $LATEST
END RequestId: 335509e4-0b1f-48c7-a178-65bc7242b438
```

CloudShell | Feedback | Console Mobile App | Privacy | Terms | Cookie preferences

© 2025, Amazon Web Services, Inc. or its affiliates.