

Practical Question 2 — Real-Time Event Ingestion & Analytics

Objective

To build a real-time analytics pipeline using **Google Cloud Pub/Sub**, **Cloud Functions**, and **BigQuery** to ingest user activity events, process them in real time, store valid records in BigQuery, and route invalid records to a Dead Letter Queue (DLQ).

Architecture Overview

1. **Pub/Sub Topic** receives user activity events in JSON format.
2. **Cloud Function (Gen2)** is triggered by the Pub/Sub topic.
3. The function:
 - o Validates incoming JSON messages
 - o Inserts valid events into BigQuery
 - o Logs errors and publishes invalid messages to a DLQ topic
4. **BigQuery** stores processed events for analytics.

Step 1: Create Pub/Sub Topics and Create Pub/Sub Topics

```
.gcloud pubsub topics create user-activity  
gcloud pubsub topics create user-activity-dlq-sujatha
```

Step 2: Create BigQuery Dataset and Table

```
3. bq mk realtime_sujatha  
And creating table with required schema
```

```
4. bq mk \ --table \ realtime_sujatha.user_events_sujatha \  
event:STRING,user:STRING,ts:TIMESTAMP
```

Step 3: Develop Cloud Function

Function Responsibilities

- Subscribe to user-activity topic
- Parse JSON messages
- Validate required fields: event, user, ts
- Insert valid records into BigQuery
- Send invalid messages to DLQ and log errors

Step 4: Deploy Cloud Function

```
gcloud functions deploy ingest_user_activity --gen2 --runtime=python311 --region=us-central1 --  
source=. --entry-point=ingest_user_activity --trigger-topic=user-activity --allow-unauthenticated
```

Step 5: Publish Test Messages

Publish a valid message

```
gcloud pubsub topics publish user-activity --message "{\"event\":\"LOGIN\",\"user\":\"sujatha\",\"ts\":\"2025-01-01T10:00:00Z\"}"
```

Publish an invalid message (missing user field)

```
gcloud pubsub topics publish user-activity --message "{\"event\":\"LOGIN\",\"ts\":\"2025-01-01T10:00:00Z\"}"
```

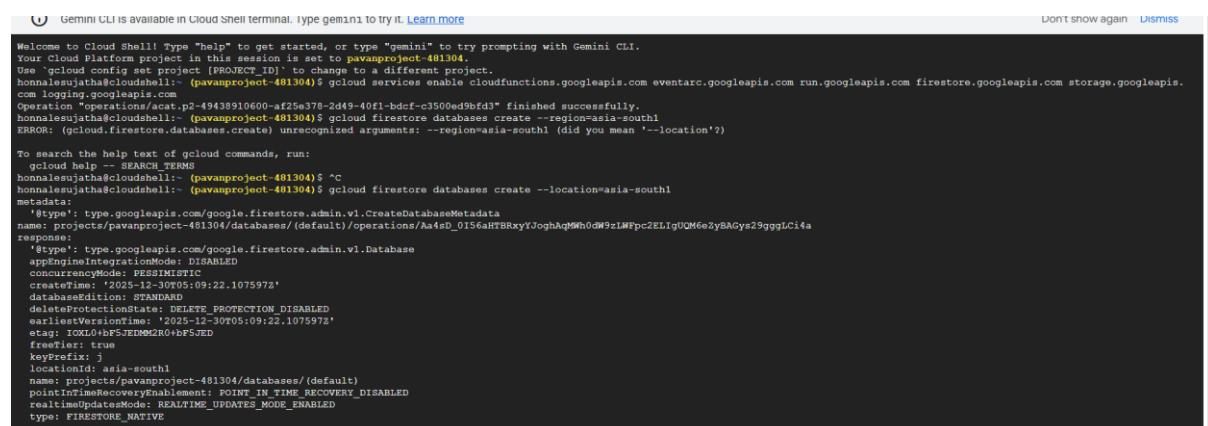
Step 6: Verify BigQuery Data

Step 7: Verify Dead Letter Queue (DLQ)

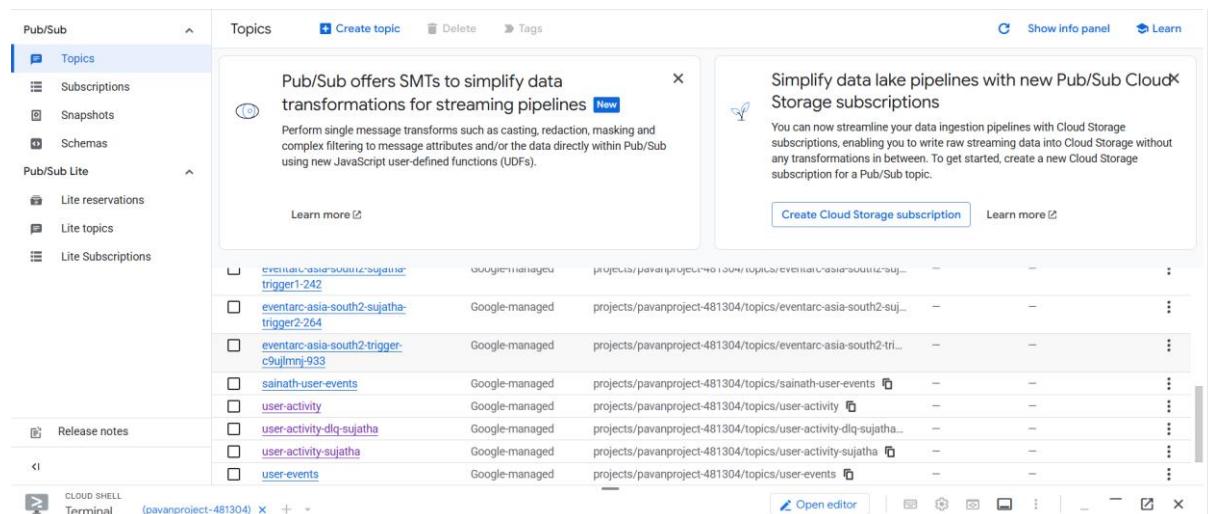
```
gcloud pubsub subscriptions pull user-activity-dlq-sub --auto-ack --limit=5
```

Step 8: View Logs

```
gcloud functions logs read ingest_user_activity --region=us-central1 --limit=10
```



```
Gemini CLI is available in Cloud Shell terminal. Type gemini to try it. Learn more Don't show again Dismiss
Welcome to Cloud Shell! Type "help" to get started, or type "gemini" to try prompting with Gemini CLI.
Your cloud Platform project in this session is set to pavanproject-481304.
Use "gcloud config set project [PROJECT_ID]" to change to a different project.
honnaleesujatha@cloudshell:~ (pavanproject-481304)$ gcloud services enable cloudfunctions.googleapis.com eventarc.googleapis.com run.googleapis.com firestore.googleapis.com storage.googleapis.com logging.googleapis.com
Operation completed successfully.
honnaleesujatha@cloudshell:~ (pavanproject-481304)$ gcloud firestore databases create --region=asia-south1
ERROR: (gcloud.firestore.databases.create) unrecognized arguments: --region=asia-south1 (did you mean '--location'?)
To search the help text of gcloud commands, run:
gcloud help --SEARCH_TERMS
honnaleesujatha@cloudshell:~ (pavanproject-481304)$ * C
honnaleesujatha@cloudshell:~ (pavanproject-481304)$ gcloud firestore databases create --location=asia-south1
metadata:
  *type: type.googleapis.com/google.firebaseio.admin.v1.CreateDatabaseMetadata
  name: projects/pavanproject-481304/databases/(default)/operations/Aa4sD_0I56aHT8RxyYJoghAqMWh0dW9zLWFpc2ELIqUQM6e2yBAGys29gggLC14a
  response:
    *type: type.googleapis.com/google.firebaseio.admin.v1.Database
    name: projects/pavanproject-481304/databases/(default)
    createTimestamp: '2025-12-30T05:09:22.107597Z'
    databaseEdition: STANDARD
    deleteProtectionState: DELETE_PROTECTION_DISABLED
    earliestVersionTime: '2025-12-30T05:09:22.107597Z'
    etag: IXOI0+bF5TEMM2R0+bF5JED
    freeTier: true
    kind: database
    locationId: asia-south1
    name: projects/pavanproject-481304/databases/(default)
    pointInTimeRecoveryEnabled: POINT_IN_TIME_RECOVERY_DISABLED
    realtimeUpdatesMode: REALTIME_UPDATES_MODE_ENABLED
    type: FIRESTORE_NATIVE
```



The screenshot shows the Google Cloud Pub/Sub interface. On the left, there's a sidebar with navigation links: Pub/Sub (Topics, Subscriptions, Snapshots, Schemas), Pub/Sub Lite (Lite reservations, Lite topics, Lite Subscriptions), and Release notes. The main area has two panels. The left panel, titled 'Topics', lists several topics: 'Pub/Sub offers SMTs to simplify data transformations for streaming pipelines' (with a 'Learn more' link), 'eventarc-asia-south2-sujatha-trigger1-242' (Google-managed), 'eventarc-asia-south2-sujatha-trigger2-264' (Google-managed), 'eventarc-asia-south2-trigger-9uljmj-933' (Google-managed), 'sainath-user-events' (Google-managed), 'user-activity' (Google-managed), 'user-activity-dlq-sujatha' (Google-managed), 'user-activity-sujatha' (Google-managed), and 'user-events' (Google-managed). The right panel, titled 'Simplify data lake pipelines with new Pub/Sub Cloud Storage subscriptions', contains a 'Create Cloud Storage subscription' button and a 'Learn more' link. At the bottom, there's a 'CLOUD SHELL Terminal' tab with '(pavanproject-481304)' and a toolbar with various icons.

Google Cloud PavanProject topics

Pub/Sub / Topics / Topic: user-activity-dlq-sujatha

user-activity-dlq-sujatha

- Topics
- Subscriptions
- Snapshots
- Schemas

Topic name: projects/pavanproject-481304/topics/user-activity-dlq-sujatha

Tags:

Export to BigQuery: Export data to a BigQuery table.

Export to Cloud Storage: Export data to a text or Avro file in Cloud Storage.

Export data

Subscriptions: Subscriptions, Snapshots, Metrics, Details, Messages

Only subscriptions attached to this topic are displayed. A subscription captures the stream of messages published to a given topic. You can also stream messages to BigQuery or Cloud Storage by creating a subscription from a Cloud Dataflow job. [Learn more](#)

Create subscription, Export

Release notes

Subscription ID: user-activity-dlq-sub, Subscription name: user-activity-dlq-sub, Project: pavanproject-481304

CLOUD SHELL Terminal (pavanproject-481304)

Google Cloud PavanProject

Pub/Sub / Topics / Topic: user-activity-sujatha

user-activity-sujatha

- Topics
- Subscriptions
- Snapshots
- Schemas

Topic name: projects/pavanproject-481304/topics/user-activity-sujatha

Tags:

Export to BigQuery: Export data to a BigQuery table.

Export to Cloud Storage: Export data to a text or Avro file in Cloud Storage.

Export data

Subscriptions: Subscriptions, Snapshots, Metrics, Details, Messages

Only subscriptions attached to this topic are displayed. A subscription captures the stream of messages published to a given topic. You can also stream messages to BigQuery or Cloud Storage by creating a subscription from a Cloud Dataflow job. [Learn more](#)

Create subscription, Export

Release notes

Subscription ID: user-events-sujatha, Subscription name: user-events-sujatha, Project: pavanproject-481304

Search BigQuery resources

pavanproject-481304 / Datasets / realtime_sujatha / Tables / user_events_sujatha

Schema

Field name	Type	Mode	Description	Key	Collation	Default value	Policy tags	Data policies
event	STRING	NULLABLE	-	-	-	-	-	-
user	STRING	NULLABLE	-	-	-	-	-	-
ts	TIMESTAMP	NULLABLE	-	-	-	-	-	-

Edit schema, View row access policies

BigQuery Resources

- pavanproject-481304
 - Repositories
 - Queries
 - Notebooks
 - Data canvases
 - Data preparations
 - Pipelines
 - Connections
 - analytics.ds
 - dataflow.ds
 - realtime_sujatha

Schema Details Preview Table explorer Insights Lineage Data profile Data Quality

Preview may not show all data being streamed to BigQuery. If you want to check for such data, run a SELECT statement over the table.

Row	event	user	ts
1	LOGIN	sujatha	2025-01-01 10:00:00 UTC
2	LOGOUT	sujatha	2025-01-01 13:00:00 UTC

Cloud Run Services

Overview Services Jobs Worker pools Domain mappings

Deploy container Connect repo Write a function Refresh

A service exposes a unique endpoint and automatically scales the underlying infrastructure to handle incoming requests. Deploy a container image, source code or a function to create a service.

Name	Deployment type	Req/sec	Region	Authentication	Ingress	Last deployed	Deployed by
ingest-user-activity	Function	0	us-central1	Public access	All	20 minutes ago	Cloud Run funct
sainath	Function	0	asia-south2	Require authentication	All	13 minutes ago	reddysainath741
sujatha	Function	0.05	asia-south2	Public access	All	35 minutes ago	pavanpann16@g

File Edit Selection View Go Run Terminal Help ⏎ ↻ ⌂ AZURE CHAT

OPEN EDITORS

EXPLORER

pyenv\cfg venv requirements.txt

FINAL ASSESSMENT

main.py requirements.txt

TERMINAL

```
availableMemory: 250M
environmentVariables:
  LOG_EXECUTION_ID: 'true'
  ingressSettings: ALLOW_ALL
maxCancelRequestConcurrency: 1
revision: ingest-user-activity_09001_x6b
service: projects/pavanproject-481304/locations/us-central1/services/ingest-user-activity
serviceAccountEmail: 49438916608-compute@developer.gserviceaccount.com
timeoutSeconds: 60
uri: https://ingest-user-activity-7ve7v0smua-uc.a.run.app
state: ACTIVE
updateTime: '2025-12-30T05:45:11.819455884Z'
url: https://us-central1-pavanproject-481304.cloudfunctions.net/ingest_user_activity

(venv) C:\Users\SUJATHA\Downloads\Final Assement>
(venv) C:\Users\SUJATHA\Downloads\Final Assement>gcloud pubsub topics publish user-activity --message "{\"event\": \"LOGIN\", \"user\": \"sujatha\", \"ts\": \"2025-01-01T10:00:00Z\"}"
messageId:
- '17562595112241815'

(venv) C:\Users\SUJATHA\Downloads\Final Assement>
(venv) C:\Users\SUJATHA\Downloads\Final Assement>bq query --use_legacy_sql=false "SELECT * FROM realtime_sujatha.user_events_sujatha ORDER BY ts DESC LIMIT 5"
+-----+
| event | user   | ts          |
+-----+
| LOGIN | sujatha | 2025-01-01 10:00:00 |
+-----+



(venv) C:\Users\SUJATHA\Downloads\Final Assement>
(venv) C:\Users\SUJATHA\Downloads\Final Assement>gcloud functions logs read ingest_user_activity --region=us-central1 -limit=10
LEVEL NAME EXECUTION_ID TIME UTC LOG
I ingest-user-activity 2025-12-30 05:58:04.443
I ingest-user-activity 2025-12-30 05:58:03.973
I ingest-user-activity 2025-12-30 05:58:03.083
I ingest-user-activity 2025-12-30 05:46:15.062
I ingest-user-activity 2025-12-30 05:45:04.334 Default STARTUP TCP probe succeeded after 1 attempt for container "worker" on port 8080.
I ingest-user-activity 2025-12-30 05:45:01.280 Starting new instance. Reason: DEPLOYMENT_ROLLOUT
- Instance started due to traffic shifting between revisions due to deployment, traffic split adjustment, or deployment

```

Build with Agent

AI responses may be inaccurate. Generate Agent Instructions to onboard AI onto your codebase.

main.py +

Describe what to build next

Agent Auto

The screenshot shows the Google Cloud Functions terminal interface. The terminal window displays the deployment of a function named 'ingest-user-activity' to the 'us-central1' region. It shows the command used to publish a message to a Pub/Sub topic and the resulting message being pulled from a subscription.

```

triggerRegion: us-central1
labels:
deployment-tool: cli-gcloud
name: projects/pavanproject-481304/locations/us-central1/functions/ingest_user_activity
satisfiesP2l: true
serviceConfig:
allTrafficContainsRevision: true
availableCpu: '0.1666'
availableMemory: 256M
environmentVariables:
LOG_EXECUTION_ID: 'true'
ingressSettings: ALLOW_ALL
maxInstanceCount: 12
maxInstanceRequestConcurrency: 1
revisionId: ingest-user-activity-00002-hoc
serviceAccountEmail: 49458916609-compute@developer.gserviceaccount.com
timeoutSeconds: 60
uri: https://ingest-user-activity-7ve7vomsua-uc.a.run.app
state: ACTIVE
updateTime: '2025-12-30T06:40:57.459963842Z'
url: https://us-central1-pavanproject-481304.cloudfunctions.net/ingest_user_activity

(venv) C:\Users\SUJATHA\Downloads\Finals Assement>gcloud pubsub topics publish user-activity --message "{\"event\":\"LOGIN\", \"ts\":\"2025-01-01T10:00:00Z\"}"
messageId:
- '17553284450460227'

(venv) C:\Users\SUJATHA\Downloads\Finals Assement>gcloud pubsub subscriptions pull user-activity-dlq-sub --auto-ack --limit=5
+-----+
| DATA | MESSAGE_ID | ORDERING_KEY | ATTRIBUTES | DELIVERY_ATTEMPT | A
+-----+
| {"event": "LOGIN", "ts": "2025-01-01T10:00:00Z"} | 1756534592303817 | | | | S
+-----+
| SUCCESS | | | | | |
+-----+

```

The screenshot shows the Google Cloud Functions terminal interface again. This time, it displays the results of a BigQuery query against a table named 'realtime_sujatha.user_events_sujatha'. The query selects data ordered by timestamp.

```

(venv) C:\Users\SUJATHA\Downloads\Finals Assement>gcloud pubsub topics publish user-activity --message "{\"event\":\"LOGOUT\", \"user\":\"sujatha\", \"ts\":\"2025-01-01T13:00:00Z\"}"
messageId:
- '17562523213424780'

(venv) C:\Users\SUJATHA\Downloads\Finals Assement>bq query --use_legacy_sql=false "SELECT * FROM realtime_sujatha.user_events_sujatha ORDER BY ts DESC LIMIT 5"
+-----+
| event | user | ts |
+-----+
| LOGOUT | sujatha | 2025-01-01 13:00:00 |
| LOGIN | sujatha | 2025-01-01 10:00:00 |
+-----+
(venv) C:\Users\SUJATHA\Downloads\Finals Assement>

```

The screenshot shows the Google Cloud Platform Cloud Run service details page for the 'ingest-user-activity' service. It provides an overview, service details, and observability logs. The logs section shows recent activity, including deployment logs and Cloud Run events.

Severity	Default
Logs	2025-12-30 11:28:04.443 IST GET 200 292 B 3 ms Chrome 101 https://ingest-user-activity-7ve7vomsua-uc.a.run.app/robots.txt 2025-12-30 11:34:42.797 IST POST 200 138 B 86 ms APIs-Google: (https://ingest-user-activity-7ve7vomsua-uc.a.run.app/ingest-user-activity)
SLOs	2025-12-30 11:34:42.810 IST Invalid message: Expecting value: line 1 column 1 (char 0)
Errors	2025-12-30 12:18:56.543 IST Cloud Run ReplaceInternalService ingest-user-activity service=494389166000gcf-admin-robot.i_ (@type: type.googleapis.com/google.cloud.error.LocationInfo) 2025-12-30 12:19:31.440 IST Cloud Run ReplaceInternalService ingest-user-activity service=494389166000gcf-admin-robot.i_ (@type: type.googleapis.com/google.cloud.error.LocationInfo) 2025-12-30 12:19:37.678 IST Starting new instance. Reason: DEPLOYMENT_ROLLOUT - Instance started due to traffic shifting between revisions due to deployment

Practical Question 3 — Event-Driven Data Ingestion

Objective

To implement a real-time, event-driven data ingestion pipeline on Google Cloud Platform that captures user activity events using Pub/Sub, processes them with Cloud Functions (Gen 2), and stores the data in BigQuery for analytics.

Architecture

Application → Pub/Sub → Cloud Functions (Gen 2) → BigQuery

Step 1: Create Pub/Sub Topic

A Pub/Sub topic is created to receive real-time user activity events.

```
gcloud pubsub topics create honnalestath_user_events
```

Step 2: Create BigQuery Dataset

A BigQuery dataset is created to store analytics data.

```
bq mk honnalestath_analytics_ds
```

Step 3: Create BigQuery Table

A table is created inside the dataset to store user activity events.

Schema:

- event → STRING
- user → STRING
- ts → TIMESTAMP

```
bq mk --table honnalestath_analytics_ds.honnalestath_user_activity \
event:STRING,user:STRING,ts:TIMESTAMP
```

Step 4: Develop Cloud Function (Gen 2)

A Cloud Function is written in Python to:

- Receive messages from Pub/Sub
- Decode Base64 data
- Parse JSON message
- Insert records into BigQuery

Step 5: Develop Cloud Function (Gen 2)

The Cloud Function is deployed with Pub/Sub trigger.

```
gcloud functions deploy honnalestath_user_function \
--gen2 \
--runtime=python311 \
--region=us-central1 \
--trigger-topic=honnalestath_user_events \
```

```
--entry-point=pubsub_to_bigquery
```

Step:6 Publish Test Message

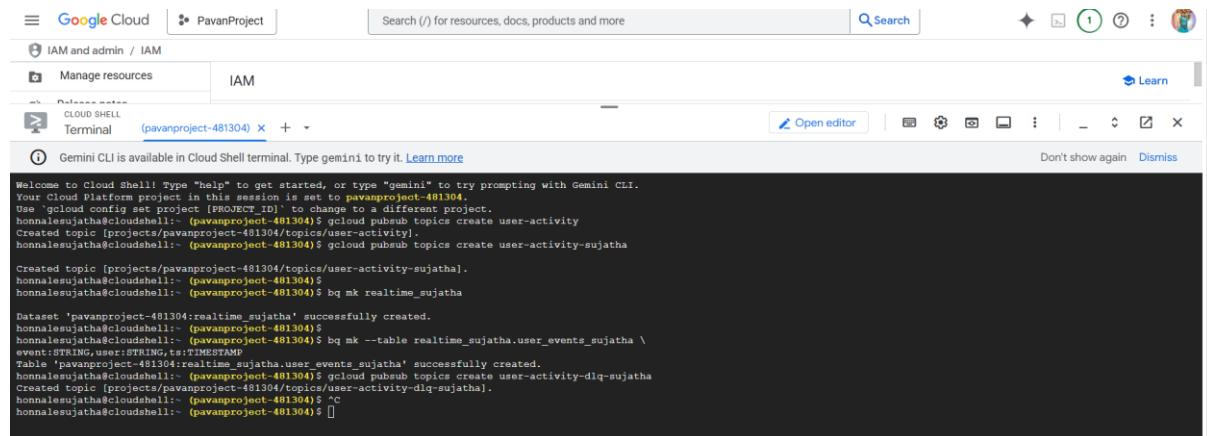
```
gcloud pubsub topics publish honnalesujatha_user_events \
--message='{"event":"LOGIN","user":"honnalesujatha","ts":"2025-01-01T10:00:00Z"}'
```

Step 7: Verify Data in BigQuery

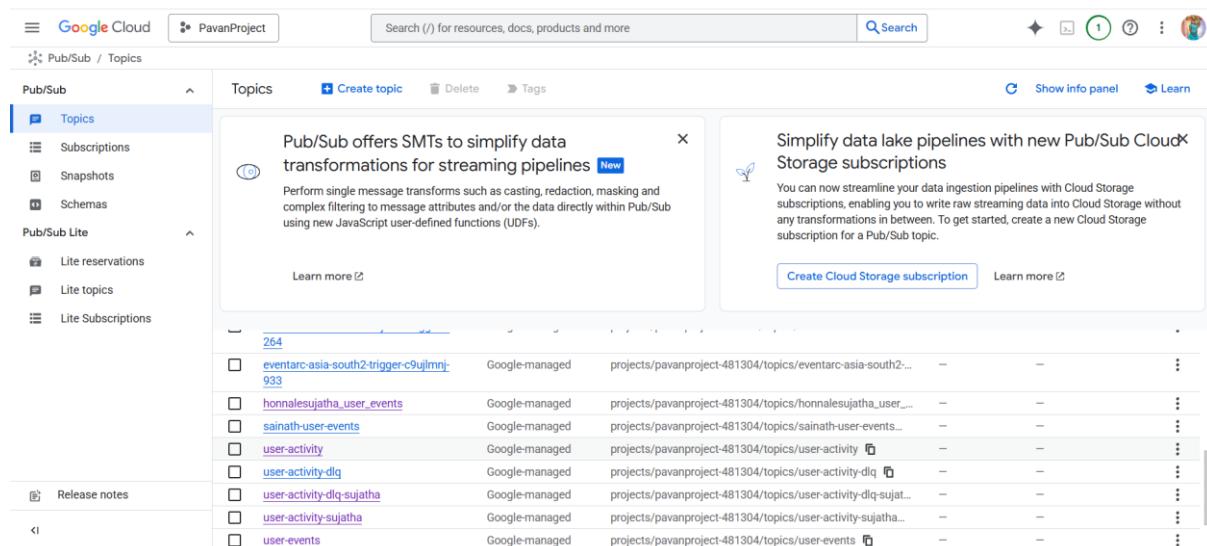
```
bq query --use_legacy_sql=false \
"SELECT * FROM `pavanproject-481304.honnalesujatha_analytics_ds.honnalesujatha_user_activity` \
ORDER BY ts DESC LIMIT 1"
```

Step 8: Verify Cloud Function Logs

```
gcloud functions logs read honnalesujatha_user_function --gen2 --region=us-central1 --limit=20
```



```
Welcome to Cloud Shell! Type "help" to get started, or type "gemini" to try prompting with Gemini CLI.
Your Cloud Platform project in this session is set to pavanproject-481304.
Use `gcloud config set project [PROJECT_ID]` to change to a different project.
honnalesujatha@cloudshell: ~(pavanproject-481304)$ gcloud pubsub topics create user-activity
Created topic (projects/pavanproject-481304/topics/user-activity)
honnalesujatha@cloudshell: ~(pavanproject-481304)$ gcloud pubsub topics create user-activity-sujatha
Created topic [projects/pavanproject-481304/topics/user-activity-sujatha].
honnalesujatha@cloudshell: ~(pavanproject-481304)$
honnalesujatha@cloudshell: ~(pavanproject-481304)$ bq mk realtime_sujatha
Dataset 'pavanproject-481304:realtime_sujatha' successfully created.
honnalesujatha@cloudshell: ~(pavanproject-481304)$
honnalesujatha@cloudshell: ~(pavanproject-481304)$ bq mk --table realtime_sujatha.user_events_sujatha \
event:STRING,user:STRING,ts:TIMESTAMP
Table 'pavanproject-481304:realtime_sujatha.user_events_sujatha' successfully created.
honnalesujatha@cloudshell: ~(pavanproject-481304)$ gcloud pubsub topics create user-activity-dlq-sujatha
Created topic (projects/pavanproject-481304/topics/user-activity-dlq-sujatha)
honnalesujatha@cloudshell: ~(pavanproject-481304)$
honnalesujatha@cloudshell: ~(pavanproject-481304)$
```



The screenshot shows the Google Cloud Pub/Sub Topics page. On the left, there's a sidebar with sections for Pub/Sub (Topics, Subscriptions, Snapshots, Schemas), Pub/Sub Lite (Lite reservations, Lite topics, Lite Subscriptions), and Release notes. The main area has a header with 'Topics', 'Create topic', 'Delete', and 'Tags'. Below the header, there's a callout box about Pub/Sub offers SMTs for streaming pipelines. The main list shows the following topics:

Topic	Type	Project	Last modified	Actions
eventarc-asia-south2-trigger-c9ujlmnj-933	Google-managed	projects/pavanproject-481304/topics/eventarc-asia-south2...	264	⋮
honnalesujatha_user_events	Google-managed	projects/pavanproject-481304/topics/honnalesujatha_user...	—	⋮
sainath-user-events	Google-managed	projects/pavanproject-481304/topics/sainath-user-events...	—	⋮
user-activity	Google-managed	projects/pavanproject-481304/topics/user-activity	—	⋮
user-activity-dlq	Google-managed	projects/pavanproject-481304/topics/user-activity-dlq	—	⋮
user-activity-dlq-sujatha	Google-managed	projects/pavanproject-481304/topics/user-activity-dlq-sujat...	—	⋮
user-activity-sujatha	Google-managed	projects/pavanproject-481304/topics/user-activity-sujatha...	—	⋮
user-events	Google-managed	projects/pavanproject-481304/topics/user-events	—	⋮

VS Code interface showing the Explorer, Editor, and Terminal panes.

EXPLORER pane:

- OPEN EDITORS: main.py (selected), requirements.txt, pyvenv.cfg
- QUESTION3 folder contains:
 - venv folder containing main.py and requirements.txt

TERMINAL pane:

```
Error in query string: Error processing job
'pavanproject-481304:bqjob_r3e43fa291fb177a7_0000019b6e751f9d_1': Syntax error:
Unexpected string literal 'SELECT * FROM `pavanproject-481304.honnale...`'
at [1:1]

(venv) C:\Users\SUJATHA\Downloads\Question3>bq query --use_legacy_sql=false --format=prettyjson "SELECT * FROM `pavanproject-481304.honnale...` ORDER BY ts DESC LIMIT 1"
[{"event": "LOGIN", "ts": "2025-01-01 10:00:00", "user": "honnale..."}]
```

Bottom status bar: Ln 32, Col 1 (1007 selected) Spaces: 4 UTF-8 CRLF {} Python

BigQuery UI - Datasets Overview for project pavanproject-481304.

Left sidebar: Shows project structure and resources like Repositories, Queries, Notebooks, Data canvases, Data preparations, Pipelines, Connections, and various datasets (analytics_ds, dataflow_ds, honnale..._analyt, honnale..._us, realtime_sujatha, user_events_sujath).

Current view: honnale..._ds / honnale..._analyt

Tables tab:

Table ID	Type	Create time	Expiry time	Label
honnale..._user_activity	Table	30 Dec 2025, 13:51:00 UTC+...	None	None

Rows per page: 50 | 1 - 1 of 1 | < > >>

BigQuery UI - Tables / honnale..._user_activity

Left sidebar: Same as the previous screenshot.

Current view: honnale..._user_activity

Preview tab:

Field name	Type	Mode	Description	Key	Collation	Default value	Policy tags	Data policies
event	STRING	NULLABLE	-	-	-	-	-	-
user	STRING	NULLABLE	-	-	-	-	-	-
ts	TIMESTAMP	NULLABLE	-	-	-	-	-	-

Schema tab:

Field name	Type	Mode	Description	Key	Collation	Default value	Policy tags	Data policies
event	STRING	NULLABLE	-	-	-	-	-	-
user	STRING	NULLABLE	-	-	-	-	-	-
ts	TIMESTAMP	NULLABLE	-	-	-	-	-	-

Buttons: Edit schema, View row access policies

The image displays three screenshots of Google Cloud Platform (GCP) interfaces:

- BigQuery Screenshot:** Shows the BigQuery interface with a dataset named "honnalestjatha_analytics_ds" and a table named "honnalestjatha_user_activity". The table has columns "event", "user", and "ts". A single row is visible: "LOGIN", "honnalestjatha", "2025-01-01 10:00:00 UTC".
- Cloud Run Services Screenshot:** Shows the Cloud Run services page for the "PavanProject" project. It lists several services:

Name	Deployment type	Req/sec	Region	Authentication	Ingress	Last deployed	Deployed by
honnalestjatha-user-function	Function	0	us-central1	Require authentication	All	1 hour ago	Cloud
ingest-user-activity	Function	0	us-central1	Public access	All	3 hours ago	Cloud
sainath	Function	0	asia-south2	Require authentication	All	2 hours ago	reddy
sainathmyapp	Function	0	asia-south2	Public access	All	1 hour ago	reddy
sujatha	Function	0.04	asia-south2	Public access	All	3 hours ago	pavan
- Cloud Run Logs Screenshot:** Shows the logs for the "honnalestjatha-user-function" service. The logs table has columns: Severity, Timestamp, and Summary. The logs show deployment events and a traceback for a recent call.

Conclusion

The event-driven data ingestion pipeline was successfully implemented using Google Cloud managed services. User activity events were ingested in real time using Pub/Sub, processed using Cloud Functions (Gen 2), and stored in BigQuery for analytics and querying.

Practical Question 1 — Event-Driven File Processing

Objective

To implement an automated event-driven system that processes files uploaded to Google Cloud Storage and stores file metadata in a database.

Architecture Used

- **Cloud Storage** – File upload trigger
- **Cloud Functions (Gen 2)** – Event processing
- **Cloud Firestore (Native Mode)** – Metadata storage
- **Cloud Logging** – Execution logs

Step 1: Create Cloud Storage Bucket

```
gsutil mb -l us-central1 gs://file-upload-lab
```

Step 2: Enable Required APIs

```
gcloud services enable cloudfunctions.googleapis.com eventarc.googleapis.com  
cloudbuild.googleapis.com firestore.googleapis.com
```

Step 3: Create Firestore Database

```
gcloud firestore databases create --region=us-central
```

Step 4: Deploy Cloud Function (Gen 2)

```
gcloud functions deploy file_upload_trigger `  
--gen2 `  
--runtime=python311 `  
--region=us-central1 `  
--source=. `  
--entry-point=file_upload_trigger `  
--trigger-event-filters="type=google.cloud.storage.object.v1.finalized" `  
--trigger-event-filters="bucket=file-upload-lab"
```

Step 5: Upload File to Trigger Function

```
"Test file upload" | Out-File test.txt
```

```
gsutil cp test.txt gs://file-upload-lab/
```

Step 6: Verify Cloud Logging

```
gcloud functions logs read file_upload_trigger --region=us-central1
```

The screenshot shows the 'Bucket details' page for the 'sujatha-file-upload-lab' bucket. The left sidebar includes links for Overview, Buckets (which is selected), Monitoring, Settings, Storage Intelligence, Insights datasets, Configuration, Marketplace, and Release notes. The main content area displays the bucket's location (asia-south2 (Delhi)), storage class (Standard), public access (Not public), protection (Soft delete), and hierarchical namespace status. Below this, there are tabs for Objects, Configuration, Permissions, Protection, Lifecycle, Observability, Inventory Reports, and Operations. The 'Objects' tab is active, showing a 'Folder browser' for 'sujatha-file-upload-lab'. It lists two objects: 'actor.csv' (547 B, text/csv, created 30 Dec 2025, 11:34:00) and 'banglore.csv' (938 KB, text/csv, created 30 Dec 2025, 11:38:35). There are also buttons for Create folder, Upload, Transfer data, and Other services.

The screenshot shows the Firestore Database interface. The top navigation bar includes a star icon, 'All databases' (selected), 'Database sujatha' (with a help icon), and a three-dot menu. Below this, a sidebar has an 'Up' arrow icon and a 'Panel view' tab (selected) with a 'Query builder' link. The main content area shows a database path: '/' > 'uploaded_files' > 'esf9y5rvvgeH2mE4tLL'. The 'uploaded_files' collection contains one document, 'esf9y5rvvgeH2mE4tLL', which is expanded. The document details are: bucket: "sujatha-file-upload-lab", file_name: "actor.csv", and file_size: 547.

The screenshot shows the Google Cloud Run console interface. On the left, there's a sidebar with navigation links: Overview, Services (which is selected), Jobs, Worker pools, and Domain mappings. The main area displays a service named 'sujatha' with the following details:

- Deploying revision:** Building source (see logs) - Completed
- Updating service:** Completed
- Creating revision:** Completed
- Routing traffic:** Completed

The service is currently running in the **asia-south2** region, with a URL provided. The scaling is set to Auto (min: 0, max: 3). Below this, there's a section for **Observability** with tabs for Revisions, Source, Triggers, Networking, Security, and YAML. The Logs tab is selected, showing a list of log entries. The logs include entries for API requests, errors, and Cloud Storage events related to file uploads and finalization.

Conclusion

This practical demonstrates a complete **event-driven file processing pipeline** in Google Cloud using **Cloud Storage, Cloud Functions (Gen 2), Firestore, and Cloud Logging**. The system automatically processes uploaded files and stores metadata reliably.

GitHub link

<https://github.com/HonnaleSujatha/GCP-Assessment>