1. Authors:

   Group nr.: 8

   - Aleksander Świniarski (309423)
   - Marta Sobol (318723)
   - Magdalena Kalińska (310242)

   Forked Repo

2. Follow all steps in README.md.

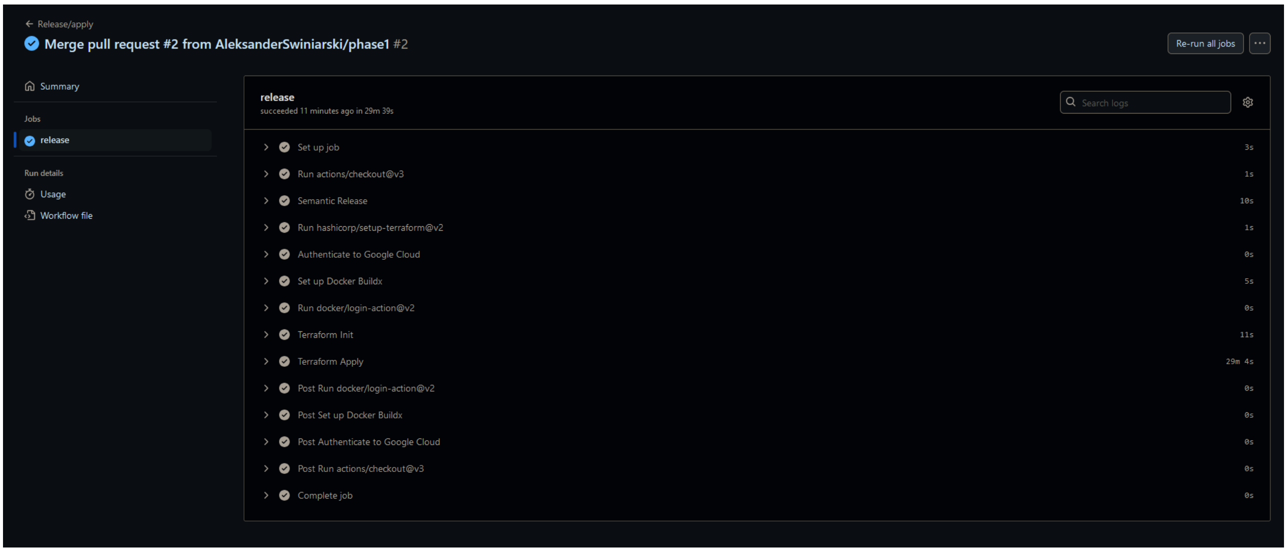
3. In boostrap/variables.tf add your emails to variable "budget_channels".

4. From avaialble Github Actions select and run destroy on main branch.

5. Create new git branch and:

   1. Modify tasks-phase1.md file.

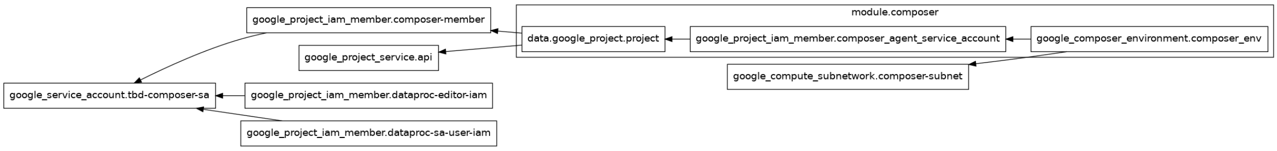
   2. Create PR from this branch to **YOUR** master and merge it to make new release.

   ```
   release
   succeeded 11 minutes ago in 29m 39s

   > Set up job                            3s
   > Run actions/checkout@v3               1s
   > Semantic Release                     10s
   > Run hashicorp/setup-terraform@v2      1s
   > Authenticate to Google Cloud          0s
   > Set up Docker Buildx                  5s
   > Run docker/login-action@v2            0s
   > Terraform Init                       11s
   > Terraform Apply                  29m 4s
   > Post Run docker/login-action@v2       0s
   > Post Set up Docker Buildx             0s
   > Post Authenticate to Google Cloud     0s
   > Post Run actions/checkout@v3          0s
   > Complete job                          0s
   ```

6. Analyze terraform code. Play with terraform plan, terraform graph to investigate different modules.

   Graf modułu:

   module.composer
google_project_iam_member.composer-member
data.google_project.project
google_project_iam_member.composer_agent_service_account
google_composer_environment.composer_env
google_project_service.api
google_compute_subnetwork.composer-subnet
google_service_account.tbd-composer-sa
google_project_iam_member.dataproc-editor-iam
google_project_iam_member.dataproc-sa-user-iam

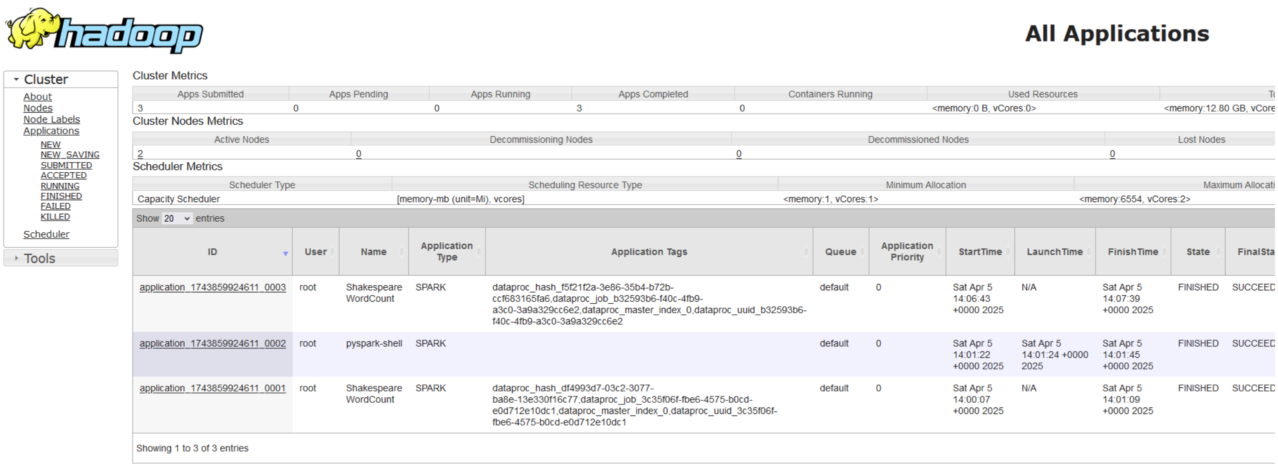
   Opis: Moduł Composer odpowiada za automatyczne utworzenie środowiska Cloud Composer 2 (czyli zarządzanego Airflowa) w Google Cloud Platform. W ramach działania tworzy dedykowane konto serwisowe, przypisuje mu niezbędne role IAM (w tym composer.worker, dataproc.editor i serviceAccountUser) oraz aktywuje wymagane API. Dodatkowo tworzy podsieć w ramach wskazanej sieci VPC, którą następnie przekazuje do modułu Composer jako środowisko sieciowe. Środowisko jest konfigurowane z parametrami dotyczącymi zasobów (CPU, RAM, storage) dla schedulera, webserwera i workerów.

7. Reach YARN UI

   Aby dostać się do konsoli YARN użyliśmy komendy:

   ```
   gcloud compute ssh tbd-cluster-m \
   --project=tbd-2025l-9921 \
   --zone=europe-west1-d \
   -- -L 8088:localhost:8088
   ```

   A następnie w przeglądarce weszliśmy na adres : `http://localhost:8088`



All Applications

Cluster Metrics

Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Used Resources	T
3	0	0	3	0	<memory:0 B, vCores:0>	<memory:12.80 GB, vCore

Cluster Nodes Metrics

Active Nodes	Decommissioning Nodes	Decommissioned Nodes	Lost Nodes
2	0	0	0

Scheduler Metrics

Scheduler Type	Scheduling Resource Type	Minimum Allocation	Maximum Allocati
Capacity Scheduler	[memory-mb (unit=Mi), vcores]	<memory:1, vCores:1>	<memory:6554, vCores:2>

Show 20 entries

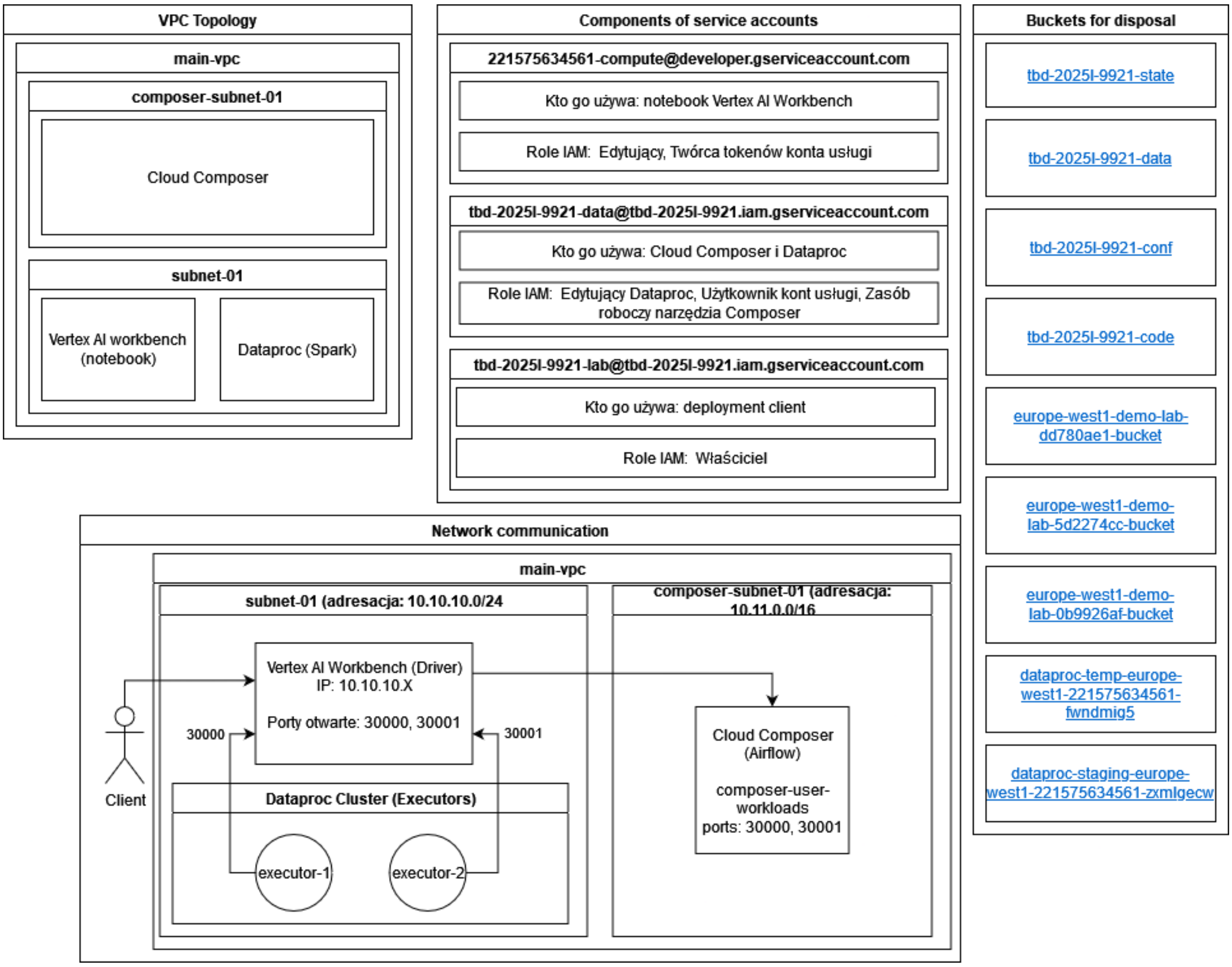
ID	User	Name	Application Type	Application Tags	Queue	Application Priority	StartTime	LaunchTime	FinishTime	State	FinalSta
application_1743859924611_0003	root	Shakespeare WordCount	SPARK	dataproc_hash_f5f21f2a-3e86-35b4-b72b-ccf683165fa6,dataproc_job_b32593b6-f40c-4fb9-a3c0-3a9a329cc6e2,dataproc_master_index_0,dataproc_uuid_b32593b6-f40c-4fb9-a3c0-3a9a329cc6e2	default	0	Sat Apr 5 14:06:43 +0000 2025	N/A	Sat Apr 5 14:07:39 +0000 2025	FINISHED	SUCCEED
application_1743859924611_0002	root	pyspark-shell	SPARK		default	0	Sat Apr 5 14:01:22 +0000 2025	Sat Apr 5 14:01:24 +0000 2025	Sat Apr 5 14:01:45 +0000 2025	FINISHED	SUCCEED
application_1743859924611_0001	root	Shakespeare WordCount	SPARK	dataproc_hash_df4993d7-03c2-3077-ba8e-13e330f16c77,dataproc_job_3c35f06f-fbe6-4575-b0cd-e0d712e10dc1,dataproc_master_index_0,dataproc_uuid_3c35f06f-fbe6-4575-b0cd-e0d712e10dc1	default	0	Sat Apr 5 14:00:07 +0000 2025	N/A	Sat Apr 5 14:01:09 +0000 2025	FINISHED	SUCCEED

Showing 1 to 3 of 3 entries

8. Draw an architecture diagram (e.g. in draw.io) that includes:

   1. VPC topology with service assignment to subnets
   2. Description of the components of service accounts
   3. List of buckets for disposal
   4. Description of network communication (ports, why it is necessary to specify the host for the driver) of Apache Spark running from Vertex AI Workbech

   Diagram:


VPC Topology
main-vpc
composer-subnet-01
Cloud Composer
subnet-01
Vertex AI workbench (notebook)
Dataproc (Spark)
Components of service accounts
221575634561-compute@developer.gserviceaccount.com
Kto go używa: notebook Vertex AI Workbench
Role IAM: Edytujący, Twórca tokenów konta usługi
tbd-2025l-9921-data@tbd-2025l-9921.iam.gserviceaccount.com
Kto go używa: Cloud Composer i Dataproc
Role IAM: Edytujący Dataproc, Użytkownik kont usługi, Zasób roboczy narzędzia Composer
tbd-2025l-9921-lab@tbd-2025l-9921.iam.gserviceaccount.com
Kto go używa: deployment client
Role IAM: Właściciel
Buckets for disposal
tbd-2025l-9921-state
tbd-2025l-9921-data
tbd-2025l-9921-conf
tbd-2025l-9921-code
europe-west1-demo-lab-dd780ae1-bucket
europe-west1-demo-lab-5d2274cc-bucket
europe-west1-demo-lab-0b9926af-bucket
dataproc-temp-europe-west1-221575634561-fwndmig5
dataproc-staging-europe-west1-221575634561-zxmlgecw
Network communication
main-vpc
subnet-01 (adresacja: 10.10.10.0/24
composer-subnet-01 (adresacja: 10.11.0.0/16
Vertex AI Workbench (Driver)
IP: 10.10.10.X
Porty otwarte: 30000, 30001
30000
30001
Client
Dataproc Cluster (Executors)
executor-1
executor-2
Cloud Composer (Airflow)
composer-user-workloads
ports: 30000, 30001

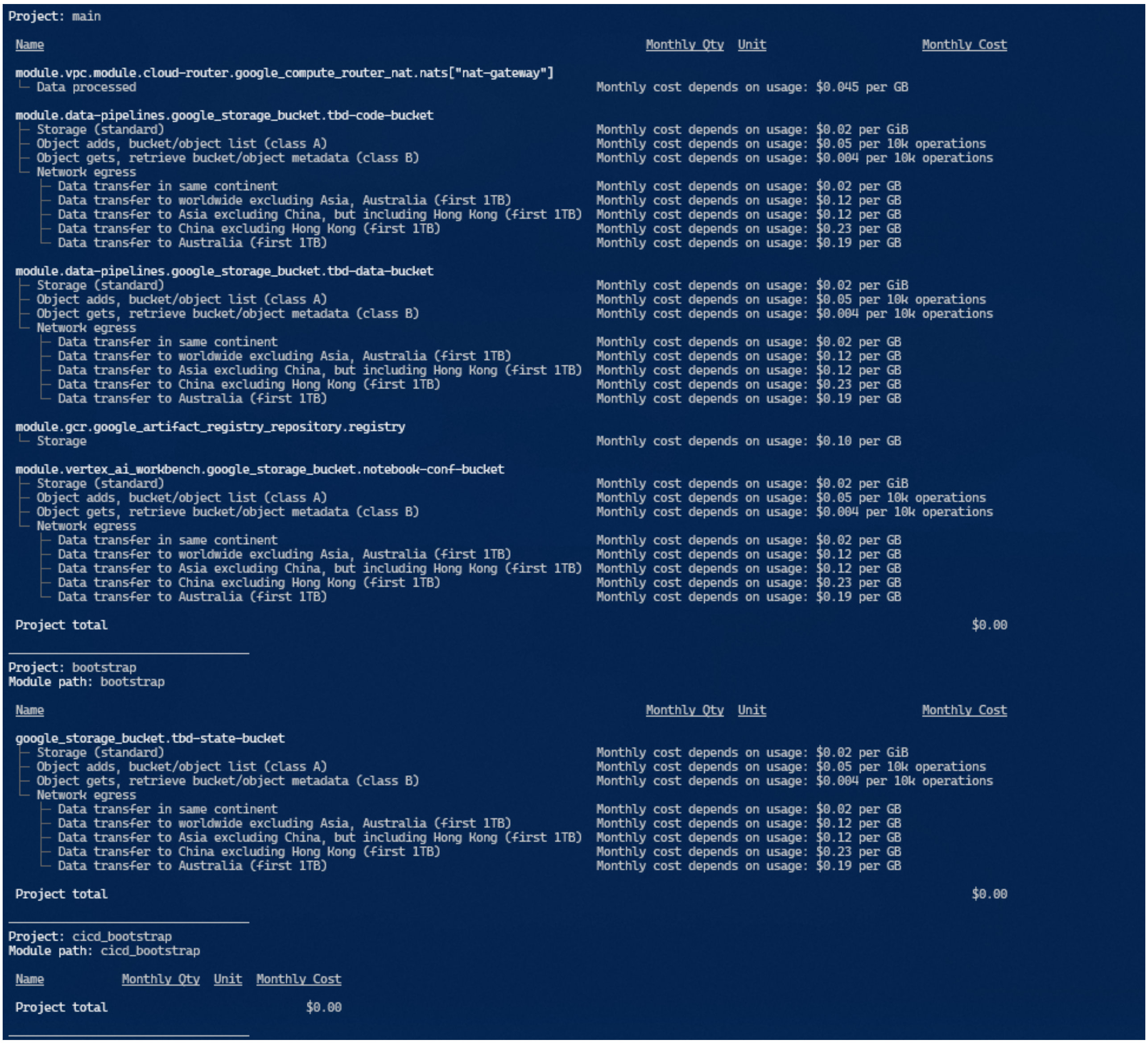
Why it is important to specify the host for the driver?: W trybie client Apache Spark, driver uruchamiany jest na instancji Vertex AI Workbench, a executory w klastrze Dataproc. Aby zapewnić poprawną komunikację, konieczne jest jawne ustawienie parametru spark.driver.host na wewnętrzny adres IP notebooka. W przeciwnym razie Spark może użyć adresu lokalnego niedostępnego dla executorów, co skutkuje błędami połączenia i niepowodzeniem joba.

9. Create a new PR and add costs by entering the expected consumption into Infracost For all the resources of type: `google_artifact_registry`, `google_storage_bucket`, `google_service_networking_connection` create a sample usage profiles and add it to the Infracost task in CI/CD pipeline. Usage file example

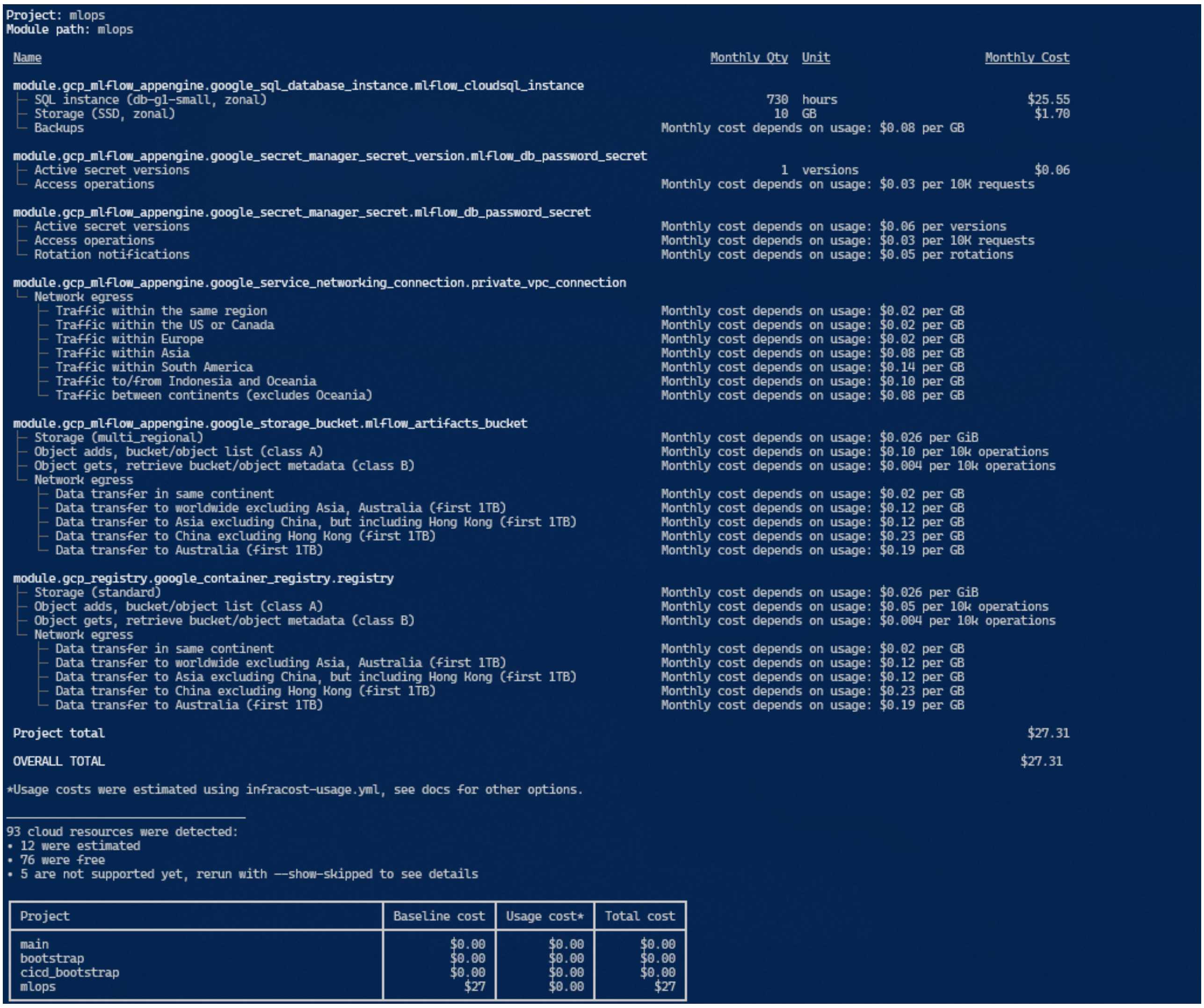
   Expected consumption:

```
 version: 0.1

 resource_usage:
 google_artifact_registry_repository.my_artifact_registry:
     storage_gb: 100                # Total data stored in the repository in GB
     monthly_egress_data_transfergb: # Monthly data delivered from the artifact registry repository in GB. You can specify any number of Google Cloud regions below, replacing - for  e.g.:
     europe_north1: 20             # GB of data delivered from the artifact registry to europe-north1.
     australia_southeast1: 30      # GB of data delivered from the artifact registry to australia-southeast1.
     china: 15                     # China excluding Hong Kong.

 google_storage_bucket.my_storage_bucket:
     storage_gb: 10                    # Total size of bucket in GB.
     monthly_class_a_operations: 100   # Monthly number of class A operations (object adds, bucket/object list).
     monthly_class_b_operations: 200   # Monthly number of class B operations (object gets, retrieve bucket/object metadata).
     monthly_data_retrieval_gb: 50     # Monthly amount of data retrieved in GB.
     monthly_egress_data_transfer_gb:  # Monthly data transfer from Cloud Storage to the following, in GB:
     same_continent: 30              # Same continent.
     worldwide: 125                  # Worldwide excluding Asia, Australia.
     asia: 15                        # Asia
     australia: 25                   # Australia.

 google_service_networking_connection.my_connection:
     monthly_egress_data_transfer_gb: # Monthly VM-VM data transfer from VPN gateway to the following, in GB:
     same_region: 25                # VMs in the same Google Cloud region.
     worldwide: 20                  # to a Google Cloud region on another continent.
```

Infracost breakdown output:



```
Project: main

 Name                                                                                     Monthly Qty  Unit                      Monthly Cost

 module.vpc.module.cloud-router.google_compute_router_nat.nats["nat-gateway"]
 └─ Data processed                                                                 Monthly cost depends on usage: $0.045 per GB

 module.data-pipelines.google_storage_bucket.tbd-code-bucket
 ├─ Storage (standard)                                                             Monthly cost depends on usage: $0.02 per GiB
 ├─ Object adds, bucket/object list (class A)                                      Monthly cost depends on usage: $0.05 per 10k operations
 ├─ Object gets, retrieve bucket/object metadata (class B)                         Monthly cost depends on usage: $0.004 per 10k operations
 └─ Network egress
    ├─ Data transfer in same continent                                             Monthly cost depends on usage: $0.02 per GB
    ├─ Data transfer to worldwide excluding Asia, Australia (first 1TB)            Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to Asia excluding China, but including Hong Kong (first 1TB)  Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to China excluding Hong Kong (first 1TB)                      Monthly cost depends on usage: $0.23 per GB
    └─ Data transfer to Australia (first 1TB)                                      Monthly cost depends on usage: $0.19 per GB

 module.data-pipelines.google_storage_bucket.tbd-data-bucket
 ├─ Storage (standard)                                                             Monthly cost depends on usage: $0.02 per GiB
 ├─ Object adds, bucket/object list (class A)                                      Monthly cost depends on usage: $0.05 per 10k operations
 ├─ Object gets, retrieve bucket/object metadata (class B)                         Monthly cost depends on usage: $0.004 per 10k operations
 └─ Network egress
    ├─ Data transfer in same continent                                             Monthly cost depends on usage: $0.02 per GB
    ├─ Data transfer to worldwide excluding Asia, Australia (first 1TB)            Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to Asia excluding China, but including Hong Kong (first 1TB)  Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to China excluding Hong Kong (first 1TB)                      Monthly cost depends on usage: $0.23 per GB
    └─ Data transfer to Australia (first 1TB)                                      Monthly cost depends on usage: $0.19 per GB

 module.gcr.google_artifact_registry_repository.registry
 └─ Storage                                                                        Monthly cost depends on usage: $0.10 per GB

 module.vertex_ai_workbench.google_storage_bucket.notebook-conf-bucket
 ├─ Storage (standard)                                                             Monthly cost depends on usage: $0.02 per GiB
 ├─ Object adds, bucket/object list (class A)                                      Monthly cost depends on usage: $0.05 per 10k operations
 ├─ Object gets, retrieve bucket/object metadata (class B)                         Monthly cost depends on usage: $0.004 per 10k operations
 └─ Network egress
    ├─ Data transfer in same continent                                             Monthly cost depends on usage: $0.02 per GB
    ├─ Data transfer to worldwide excluding Asia, Australia (first 1TB)            Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to Asia excluding China, but including Hong Kong (first 1TB)  Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to China excluding Hong Kong (first 1TB)                      Monthly cost depends on usage: $0.23 per GB
    └─ Data transfer to Australia (first 1TB)                                      Monthly cost depends on usage: $0.19 per GB

 Project total                                                                                                                          $0.00

──────────────────────────────────
Project: bootstrap
Module path: bootstrap

 Name                                                                                     Monthly Qty  Unit                      Monthly Cost

 google_storage_bucket.tbd-state-bucket
 ├─ Storage (standard)                                                             Monthly cost depends on usage: $0.02 per GiB
 ├─ Object adds, bucket/object list (class A)                                      Monthly cost depends on usage: $0.05 per 10k operations
 ├─ Object gets, retrieve bucket/object metadata (class B)                         Monthly cost depends on usage: $0.004 per 10k operations
 └─ Network egress
    ├─ Data transfer in same continent                                             Monthly cost depends on usage: $0.02 per GB
    ├─ Data transfer to worldwide excluding Asia, Australia (first 1TB)            Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to Asia excluding China, but including Hong Kong (first 1TB)  Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to China excluding Hong Kong (first 1TB)                      Monthly cost depends on usage: $0.23 per GB
    └─ Data transfer to Australia (first 1TB)                                      Monthly cost depends on usage: $0.19 per GB

 Project total                                                                                                                          $0.00

──────────────────────────────────
Project: cicd_bootstrap
Module path: cicd_bootstrap

 Name           Monthly Qty  Unit  Monthly Cost

 Project total                            $0.00

──────────────────────────────────
```

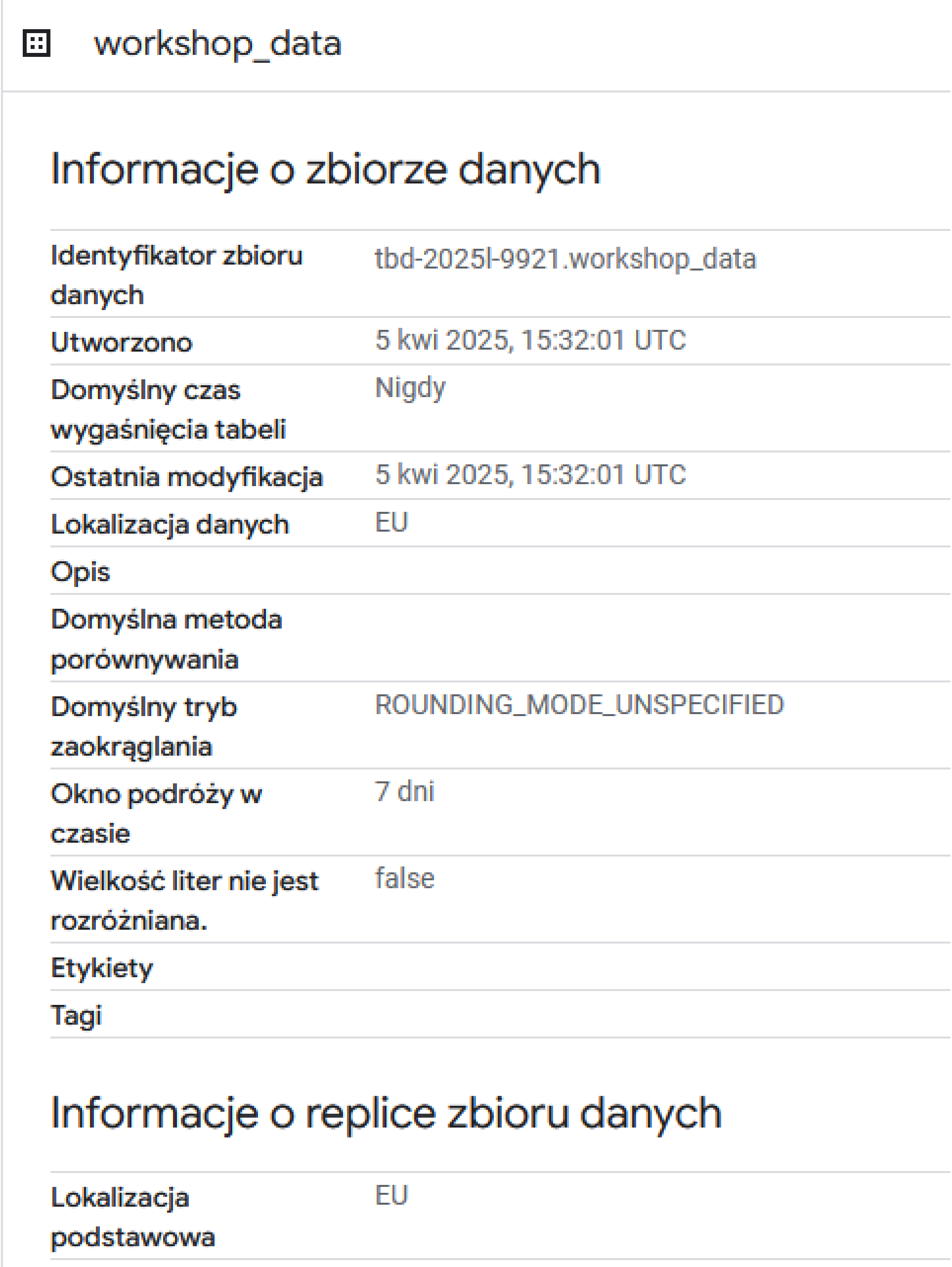


```
Project: mlops
Module path: mlops

 Name                                                                                  Monthly Qty  Unit                 Monthly Cost

 module.gcp_mlflow_appengine.google_sql_database_instance.mlflow_cloudsql_instance
 ├─ SQL instance (db-g1-small, zonal)                                                          730  hours                     $25.55
 ├─ Storage (SSD, zonal)                                                                        10  GB                         $1.70
 └─ Backups                                                              Monthly cost depends on usage: $0.08 per GB

 module.gcp_mlflow_appengine.google_secret_manager_secret_version.mlflow_db_password_secret
 ├─ Active secret versions                                                                       1  versions                   $0.06
 └─ Access operations                                                    Monthly cost depends on usage: $0.03 per 10K requests

 module.gcp_mlflow_appengine.google_secret_manager_secret.mlflow_db_password_secret
 ├─ Active secret versions                                               Monthly cost depends on usage: $0.06 per versions
 ├─ Access operations                                                    Monthly cost depends on usage: $0.03 per 10K requests
 └─ Rotation notifications                                               Monthly cost depends on usage: $0.05 per rotations

 module.gcp_mlflow_appengine.google_service_networking_connection.private_vpc_connection
 └─ Network egress
    ├─ Traffic within the same region                                    Monthly cost depends on usage: $0.02 per GB
    ├─ Traffic within the US or Canada                                   Monthly cost depends on usage: $0.02 per GB
    ├─ Traffic within Europe                                             Monthly cost depends on usage: $0.02 per GB
    ├─ Traffic within Asia                                               Monthly cost depends on usage: $0.08 per GB
    ├─ Traffic within South America                                      Monthly cost depends on usage: $0.14 per GB
    ├─ Traffic to/from Indonesia and Oceania                             Monthly cost depends on usage: $0.10 per GB
    └─ Traffic between continents (excludes Oceania)                     Monthly cost depends on usage: $0.08 per GB

 module.gcp_mlflow_appengine.google_storage_bucket.mlflow_artifacts_bucket
 ├─ Storage (multi_regional)                                             Monthly cost depends on usage: $0.026 per GiB
 ├─ Object adds, bucket/object list (class A)                            Monthly cost depends on usage: $0.10 per 10k operations
 ├─ Object gets, retrieve bucket/object metadata (class B)               Monthly cost depends on usage: $0.004 per 10k operations
 └─ Network egress
    ├─ Data transfer in same continent                                   Monthly cost depends on usage: $0.02 per GB
    ├─ Data transfer to worldwide excluding Asia, Australia (first 1TB)  Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to Asia excluding China, but including Hong Kong (first 1TB) Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to China excluding Hong Kong (first 1TB)            Monthly cost depends on usage: $0.23 per GB
    └─ Data transfer to Australia (first 1TB)                            Monthly cost depends on usage: $0.19 per GB

 module.gcp_registry.google_container_registry.registry
 ├─ Storage (standard)                                                   Monthly cost depends on usage: $0.026 per GiB
 ├─ Object adds, bucket/object list (class A)                            Monthly cost depends on usage: $0.05 per 10k operations
 ├─ Object gets, retrieve bucket/object metadata (class B)               Monthly cost depends on usage: $0.004 per 10k operations
 └─ Network egress
    ├─ Data transfer in same continent                                   Monthly cost depends on usage: $0.02 per GB
    ├─ Data transfer to worldwide excluding Asia, Australia (first 1TB)  Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to Asia excluding China, but including Hong Kong (first 1TB) Monthly cost depends on usage: $0.12 per GB
    ├─ Data transfer to China excluding Hong Kong (first 1TB)            Monthly cost depends on usage: $0.23 per GB
    └─ Data transfer to Australia (first 1TB)                            Monthly cost depends on usage: $0.19 per GB

 Project total                                                                                                            $27.31

 OVERALL TOTAL                                                                                                           $27.31

*Usage costs were estimated using infracost-usage.yml, see docs for other options.

──────────────────────────────────
93 cloud resources were detected:
• 12 were estimated
• 76 were free
• 5 are not supported yet, rerun with --show-skipped to see details

┏━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━┳━━━━━━━━━━━━┓
┃ Project                                             ┃ Baseline cost ┃ Usage cost* ┃ Total cost ┃
┣━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╋━━━━━━━━━━━━━━━╋━━━━━━━━━━━━━╋━━━━━━━━━━━━┫
┃ main                                                ┃         $0.00 ┃       $0.00 ┃      $0.00 ┃
┃ bootstrap                                           ┃         $0.00 ┃       $0.00 ┃      $0.00 ┃
┃ cicd_bootstrap                                      ┃         $0.00 ┃       $0.00 ┃      $0.00 ┃
┃ mlops                                               ┃           $27 ┃       $0.00 ┃        $27 ┃
┗━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┻━━━━━━━━━━━━━━━┻━━━━━━━━━━━━━┻━━━━━━━━━━━━┛
```

10. Create a BigQuery dataset and an external table using SQL

    Kod do stworzenia BigQuery dataset:

    ```
    CREATE SCHEMA IF NOT EXISTS `tbd-2025l-9921.workshop_data`
    OPTIONS (location = 'EU');
    ```



workshop_data

Informacje o zbiorze danych

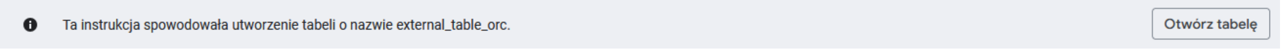
Identyfikator zbioru danych	tbd-2025l-9921.workshop_data
Utworzono	5 kwi 2025, 15:32:01 UTC
Domyślny czas wygaśnięcia tabeli	Nigdy
Ostatnia modyfikacja	5 kwi 2025, 15:32:01 UTC
Lokalizacja danych	EU
Opis	
Domyślna metoda porównywania	
Domyślny tryb zaokrąglania	ROUNDING_MODE_UNSPECIFIED
Okno podróży w czasie	7 dni
Wielkość liter nie jest rozróżniana.	false
Etykiety	
Tagi	

Informacje o replice zbioru danych

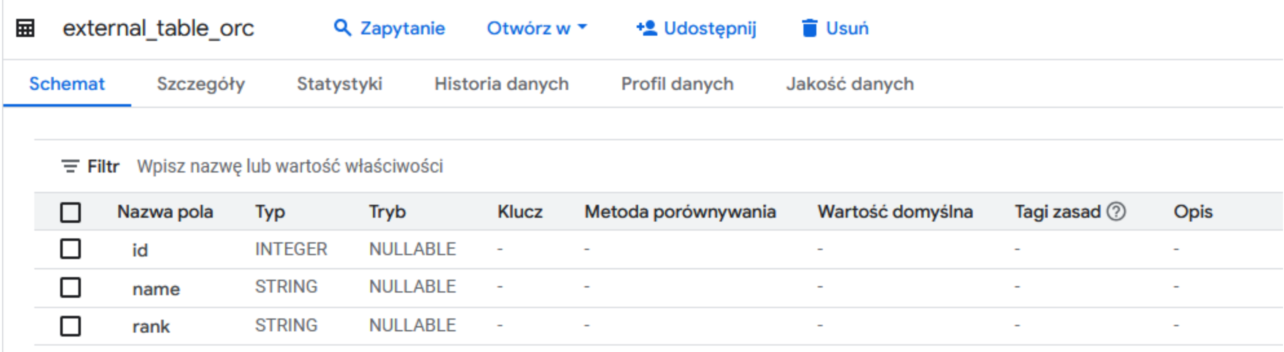
Lokalizacja podstawowa	EU

Kod do stworzenia external table:

```
CREATE OR REPLACE EXTERNAL TABLE `tbd-2025l-
9921.workshop_data.external_table_orc`
OPTIONS (
format = 'ORC',
uris = ['gs://tbd-2025l-9921-data/sample.orc']
);
```



Ta instrukcja spowodowała utworzenie tabeli o nazwie external_table_orc.



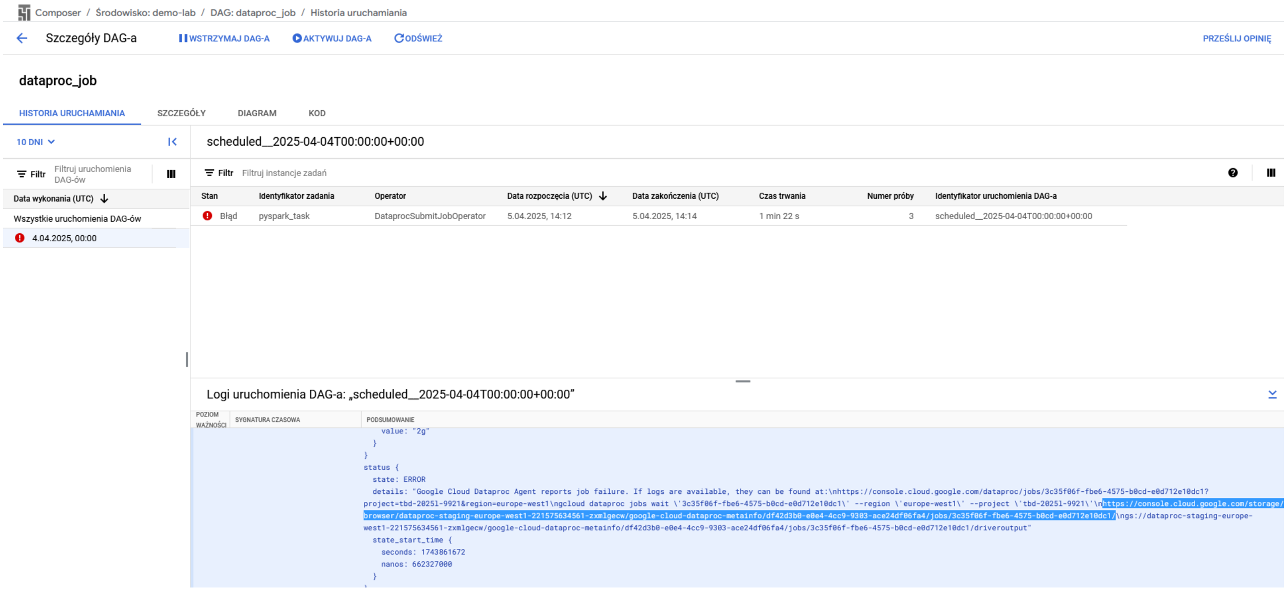
□	Nazwa pola	Typ	Tryb	Klucz	Metoda porównywania	Wartość domyślna	Tagi zasad	Opis
□	id	INTEGER	NULLABLE	-	-	-	-	-
□	name	STRING	NULLABLE	-	-	-	-	-
□	rank	STRING	NULLABLE	-	-	-	-	-

why does ORC not require a table schema?

ORC nie potrzebuje table schema ponieważ, zawiera metadane i schemat zapisane jest wewnątrz pliku (selfdescribing)

11. Find and correct the error in spark-job.py

    Jak znaleźć: Znaleźliśmy logi błędu w Dag'ach:



scheduled__2025-04-04T00:00:00+00:00

Stan	Identyfikator zadania	Operator	Data rozpoczęcia (UTC)	Data zakończenia (UTC)	Czas trwania	Numer próby	Identyfikator uruchomienia DAG-a
Błąd	pyspark_task	DataprocSubmitJobOperator	5.04.2025, 14:12	5.04.2025, 14:14	1 min 22 s	3	scheduled__2025-04-04T00:00:00+00:00

Logi uruchomienia DAG-a: „scheduled__2025-04-04T00:00:00+00:00”

POZIOM WAŻNOŚCI | SYGNATURA CZASOWA | PODSUMOWANIE

```
    value: "2g"
  }
}
status {
  state: ERROR
  details: "Google Cloud Dataproc Agent reports job failure. If logs are available, they can be found at:\nhttps://console.cloud.google.com/dataproc/jobs/3c35f06f-fbe6-4575-b0cd-e0d712e10dc1?
project=tbd-2025l-9921&region=europe-west1\ngcloud dataproc jobs wait \'3c35f06f-fbe6-4575-b0cd-e0d712e10dc1\' --region \'europe-west1\' --project \'tbd-2025l-9921\'\nhttps://console.cloud.google.com/storage/
browser/dataproc-staging-europe-west1-221575634561-zxmlgecw/google-cloud-dataproc-metainfo/df42d3b0-e0e4-4cc9-9303-ace24df06fa4/jobs/3c35f06f-fbe6-4575-b0cd-e0d712e10dc1/\ngs://dataproc-staging-europe-
west1-221575634561-zxmlgecw/google-cloud-dataproc-metainfo/df42d3b0-e0e4-4cc9-9303-ace24df06fa4/jobs/3c35f06f-fbe6-4575-b0cd-e0d712e10dc1/driveroutput"
  state_start_time {
    seconds: 1743861672
    nanos: 662327000
  }
```

Z logów dotarliśmy do pliku `google-cloud-dataproc-metainfo_df42d3b0-e0e4-4cc9-9303-ace24df06fa4_jobs_3c35f06f-fbe6-4575-b0cd-e0d712e10dc1_driveroutput` który wskazał nam błąd:

```
:
com.google.cloud.hadoop.repackaged.gcs.com.google.api.client.googleapis.json.GoogleJsonResponseException: 404 Not Found
POST https://storage.googleapis.com/upload/storage/v1/b/tbd-2025l-9900-data/o?ifGenerationMatch=0&uploadType=multipart
{
"code" : 404,
"errors" : [ {
    "domain" : "global",
    "message" : "The specified bucket does not exist.",
    "reason" : "notFound"
} ],
"message" : "The specified bucket does not exist."
}
```

Powód: Błędna nazwa bucket'a

Fix: Poprawa nazwy bucket'a i dodanie katalogu `shakespeare` do bucketa `gs://tbd-2025l-9921-data/data`

12. Add support for preemptible/spot instances in a Dataproc cluster

    place the link to the modified file and inserted terraform code

    Zmieniony plik

    Dokonana zmiana:

    ```
    preemptible_worker_config {
      num_instances = 2
    }
    ```