

Assignment 4, Cloud Computing

Student: Сұлтан Өсел Сержанқызы
ID: 24MD0490

05.12.2024

Table of contents

Introduction	3
Big Data and Machine Learning on Google Cloud	4
Cloud Security and Compliance	14
Conclusion	17
References.....	19

Introduction

This report explores the implementation of Big Data and Machine Learning pipelines on Google Cloud, emphasizing data ingestion, processing, model training, deployment, and monitoring. Additionally, it highlights the importance of cloud security, compliance, and incident response planning to ensure a secure and reliable environment for these advanced solutions.

As a student passionate about cloud technologies, I am exploring how Google Cloud empowers innovation through Big Data and Machine Learning. My journey begins with understanding data pipelines, from ingestion and processing to training and deploying machine learning models. I am also delving into critical aspects like monitoring, logging, and ensuring cloud security through encryption, IAM, and network protection. By integrating these elements, I aim to build a comprehensive understanding of modern cloud solutions and their role in creating secure, scalable, and efficient systems.

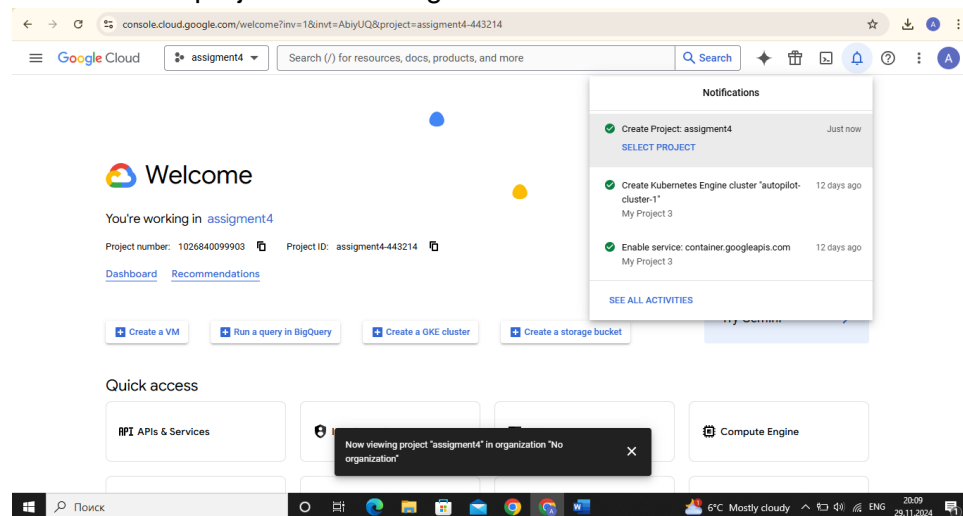
Exercise 1: Big Data and Machine Learning on Google Cloud

Objective: Implement a big data processing and machine learning pipeline using Google Cloud services.

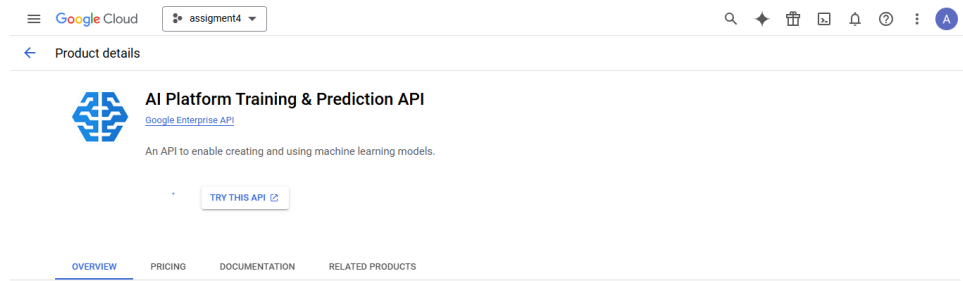
Tasks:

1. Set Up a Google Cloud Project:

- Create a new project in the Google Cloud Console.

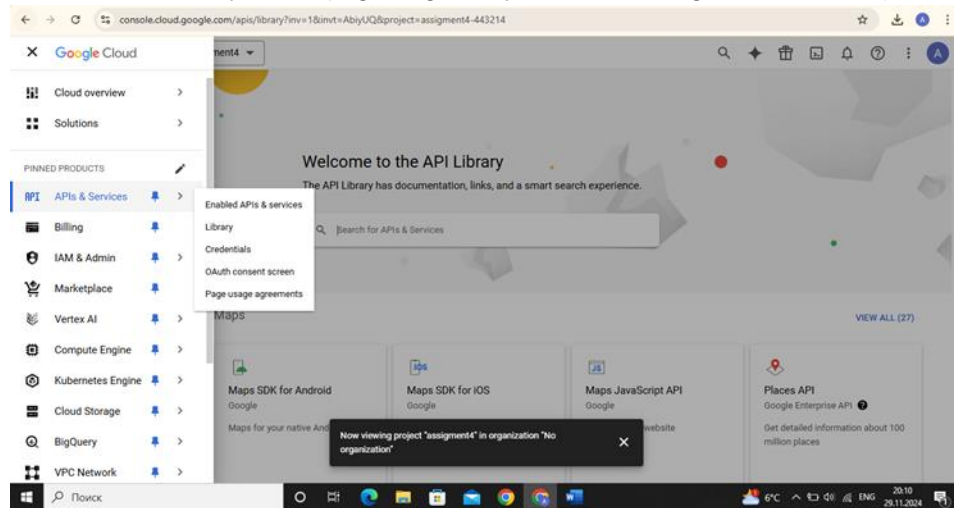


Picture-1

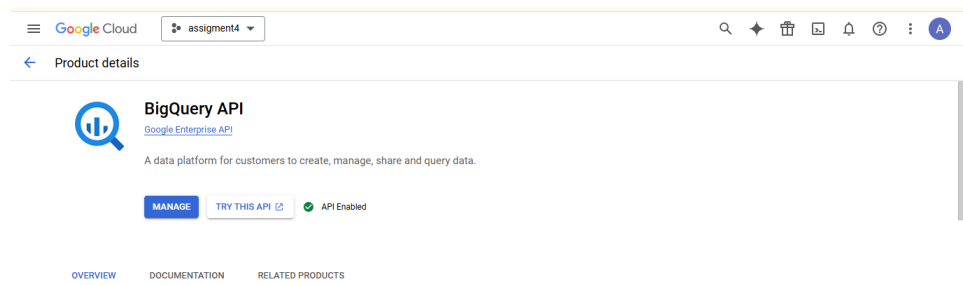


Picture-2

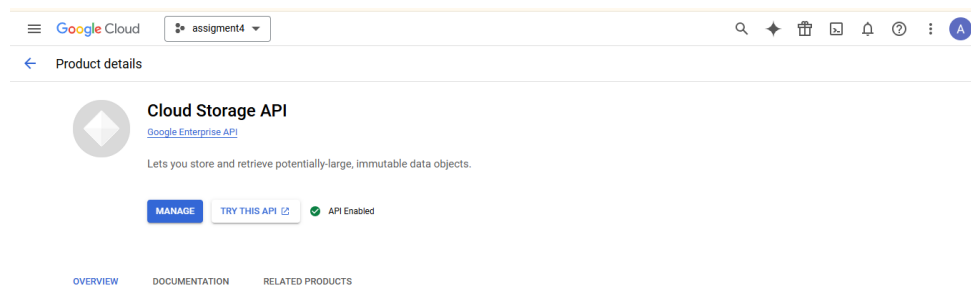
- Enable necessary APIs (e.g., BigQuery, Cloud Storage, AI Platform).



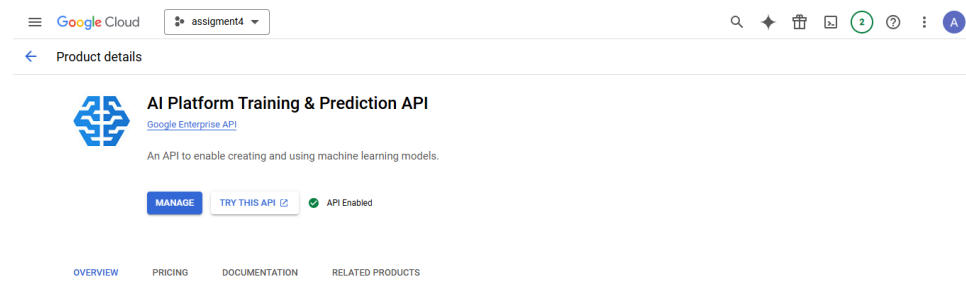
Picture-3



Picture-4



Picture-5

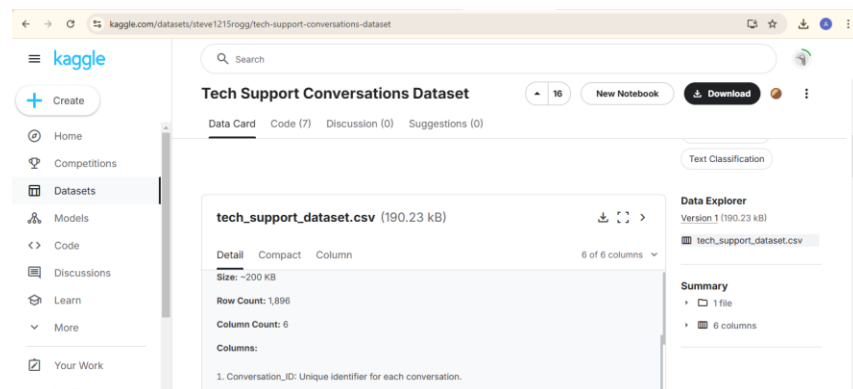


Picture-6

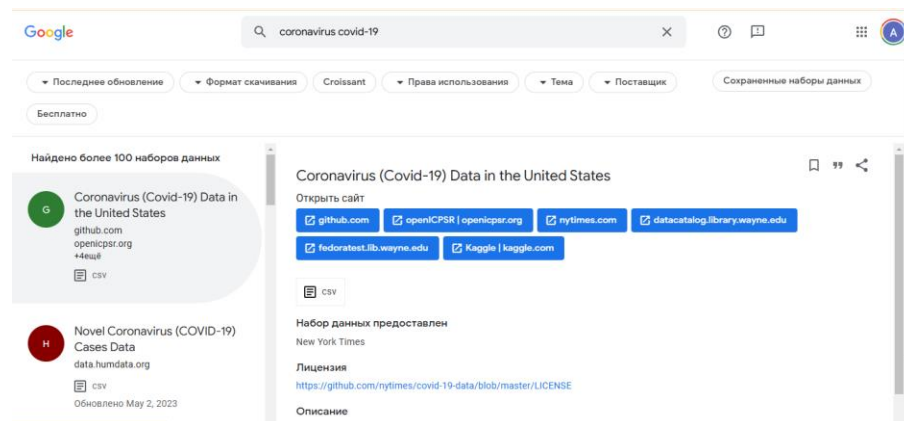
Explanation: First things first, I created a fresh, new project in the Google Cloud Console. I felt like I was building my own little digital world! Then, I enabled all the important APIs – BigQuery, Cloud Storage, and the AI Platform. It was like giving my project superpowers!

2. Data Ingestion:

- Collect a large dataset relevant to your use case (e.g., public datasets from Kaggle or Google Dataset Search).
Via Kaggle I find the “Tech Support Conversations Dataset” named dataset

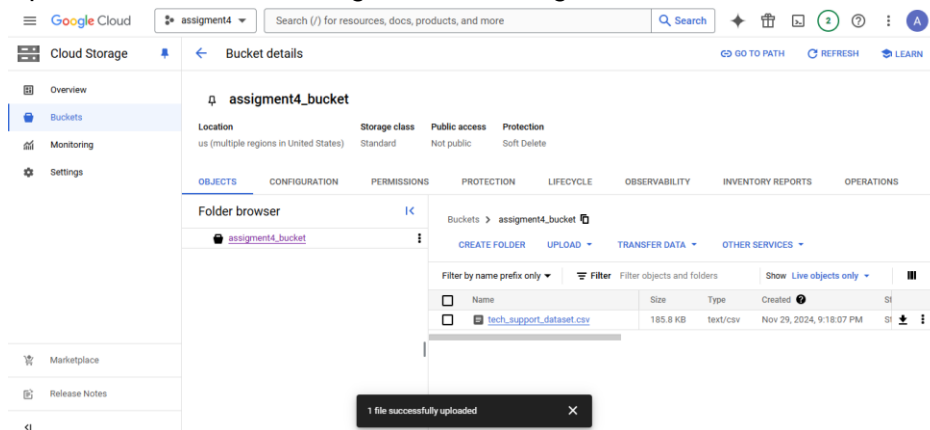


Picture-7



Picture-8

- Upload the dataset to Google Cloud Storage.

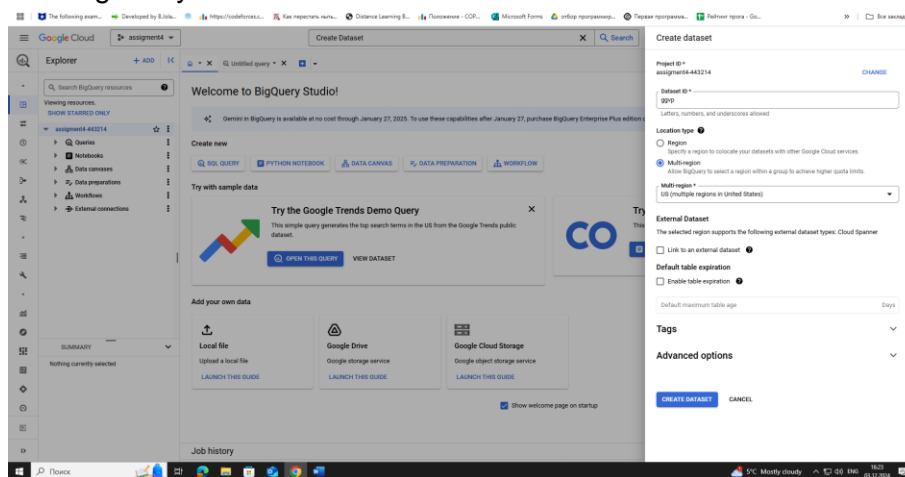


Picture-9

Explanation: I found this awesome "Tech Support Conversations Dataset" on Kaggle. It felt like a treasure trove of information just waiting to be explored. I uploaded it to Cloud Storage, my trusty data warehouse in the cloud.

3. Data Processing with BigQuery:

- Use BigQuery to create a dataset and load the data from Cloud Storage.



Picture-10

ggvp

Dataset info

Dataset ID	assignment4-443214.ggvp
Created	Dec 3, 2024, 4:23:31 PM UTC+5
Default table expiration	Never
Last modified	Dec 3, 2024, 4:23:31 PM UTC+5
Data location	US
Description	
Default collation	
Default rounding mode	ROUNDING_MODE_UNSPECIFIED
Time travel window	7 days
Case insensitive	false
Labels	
Tags	

Dataset replica info

PREVIEW

Primary location	US
------------------	----

Picture-11

console.cloud.google.com/bigquery?inv=1&invnt=AbjllA&project=assignment4-443214&ws=11m4l1m3l3m2l1assignment4-443214l2ggvp

Google Cloud

assignment4

Explorer

ggvp

Dataset info

Dataset ID	assignment4-443214.ggvp
Created	Dec 3, 2024, 4:23:31 PM UTC+5
Default table expiration	Never
Last modified	Dec 3, 2024, 4:23:31 PM UTC+5
Data location	US
Description	
Default collation	
Default rounding mode	ROUNDING_MODE_UNSPECIFIED
Time travel window	7 days
Case insensitive	false
Labels	
Tags	

Dataset replica info

PREVIEW

Primary location	US
------------------	----

Job history

Create table

Source

Create table from

Google Cloud Storage

Select file from GCS bucket or use a URI pattern

assignment4_bucket/tech_support_dataset.csv

BROWSE

File format

CSV

Source Data Partitioning

Destination

Project *

assignment4-443214

BROWSE

Dataset *

ggvp

Table *

Maximum name size is 1,024 UTF-8 bytes. Unicode letters, marks, numbers, connectors, dashes, and spaces are allowed.

Table type

Native table

Schema

Auto detect

Schema will be automatically generated.

Partition and cluster settings

Partitioning

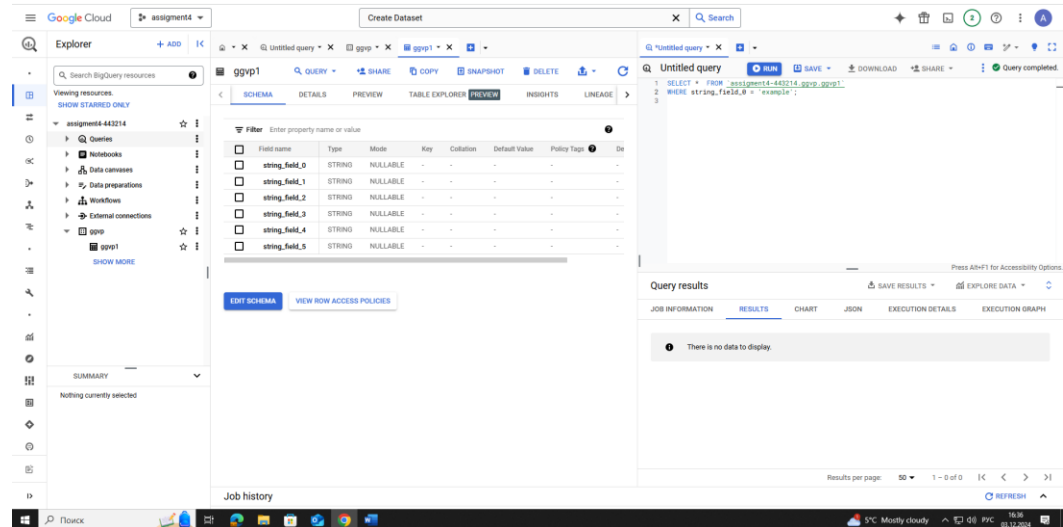
No partitioning

CREATE TABLE

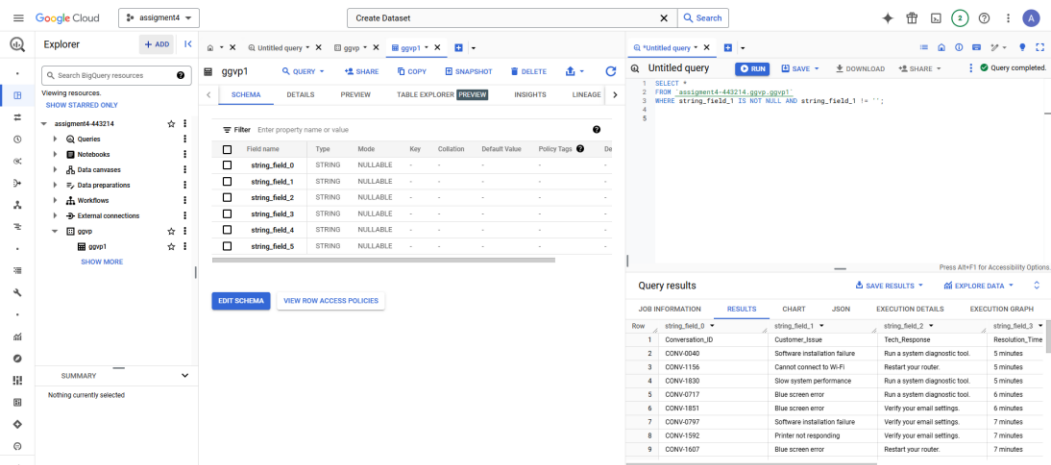
CANCEL

Picture-12

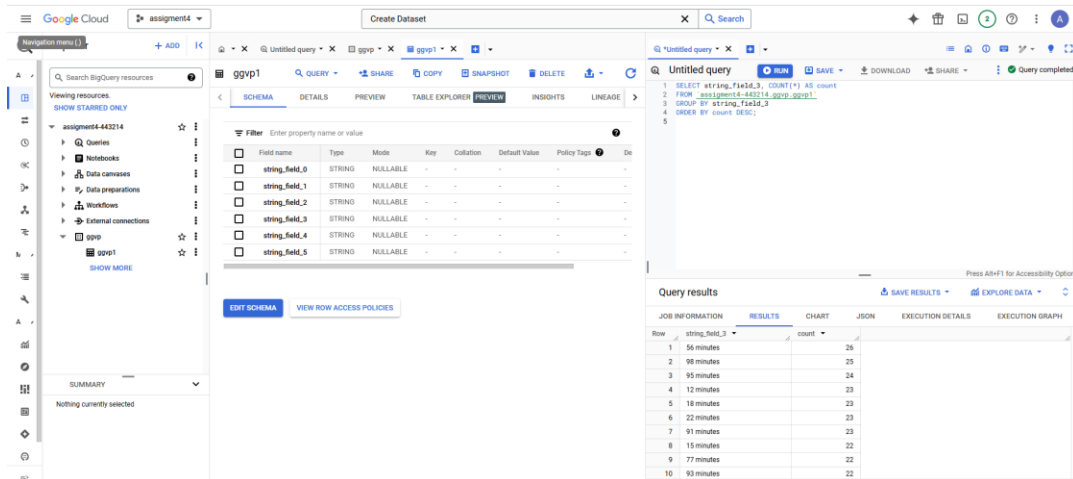
- Perform data cleaning and preprocessing using SQL queries (e.g., filtering, aggregating).



Picture-13 Filtering data

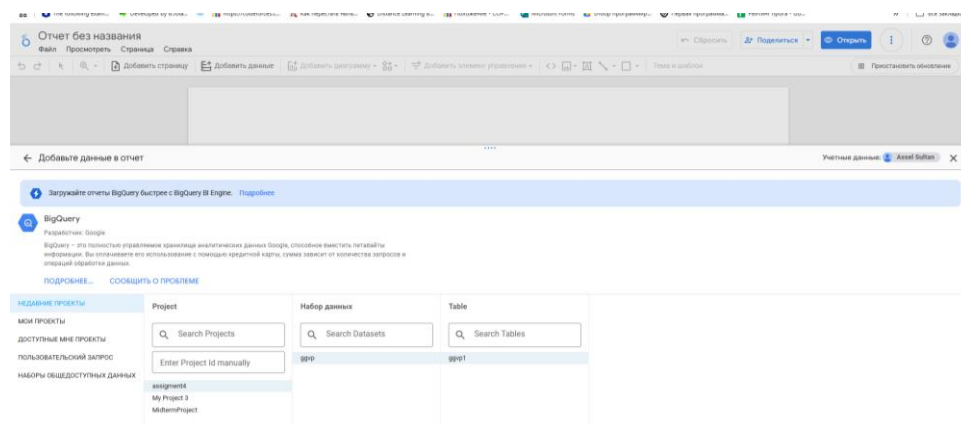


Picture-14 Deleting empty values

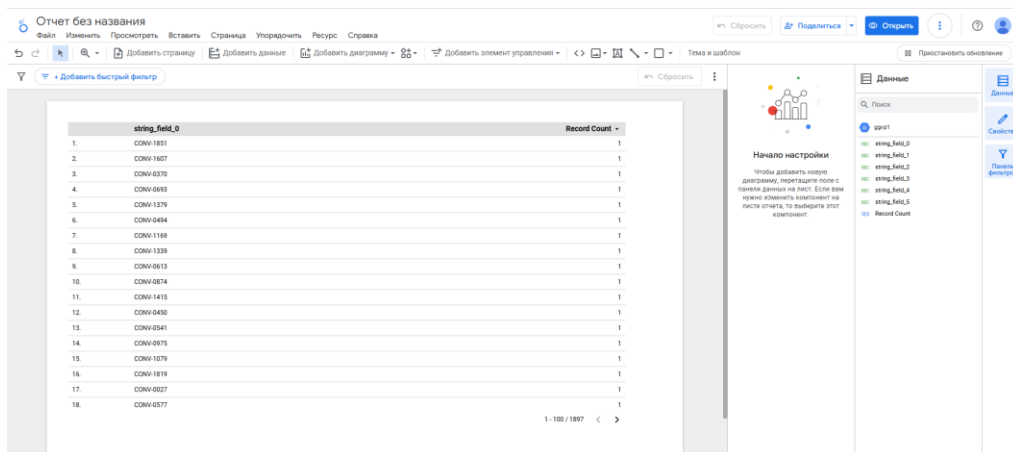


Picture-15 Data aggregation

- Create summary statistics and visualize the results using Google Data Studio or similar tools.



Picture-16



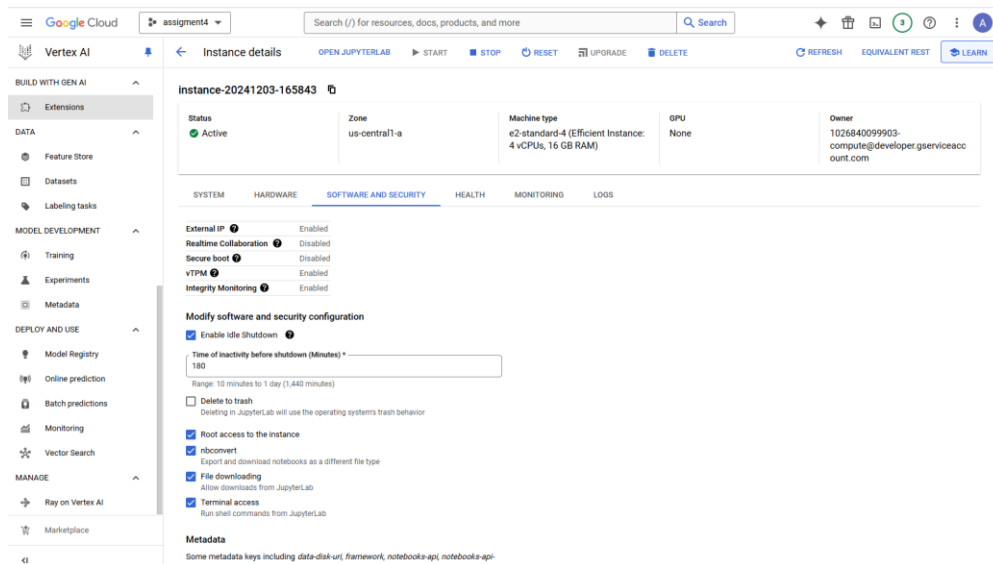
Picture-17

Explanation: BigQuery was my data playground. I created a dedicated dataset and imported the conversation data from Cloud Storage. Then, I put on my data cleaning hat and used SQL

queries to filter out unnecessary information, like getting rid of empty rows and making sure everything was spick and span. I also aggregated the data to get a better overall picture. To make things even clearer, I whipped up some visualizations using Google Data Studio. It was so satisfying to see patterns emerge!

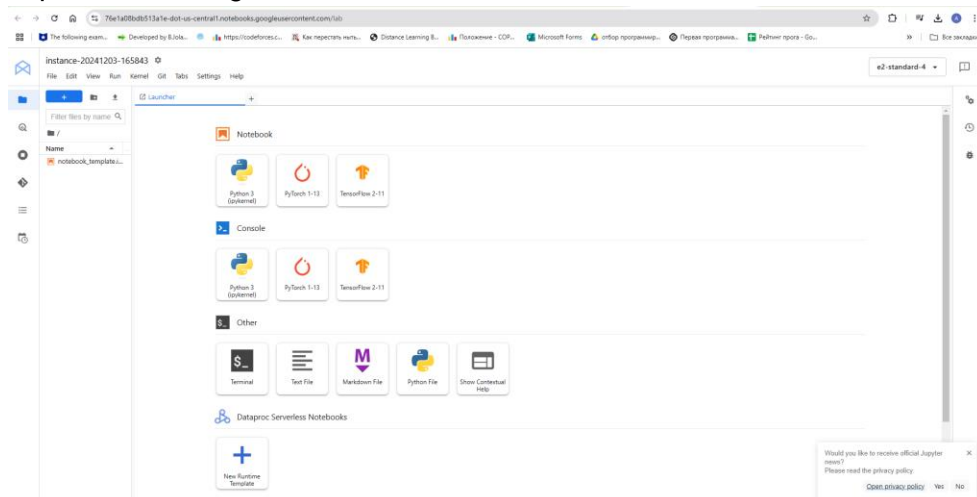
4. Machine Learning Model Training:

- Use the AI Platform to train a machine learning model on the processed data. I choose Vertex AI



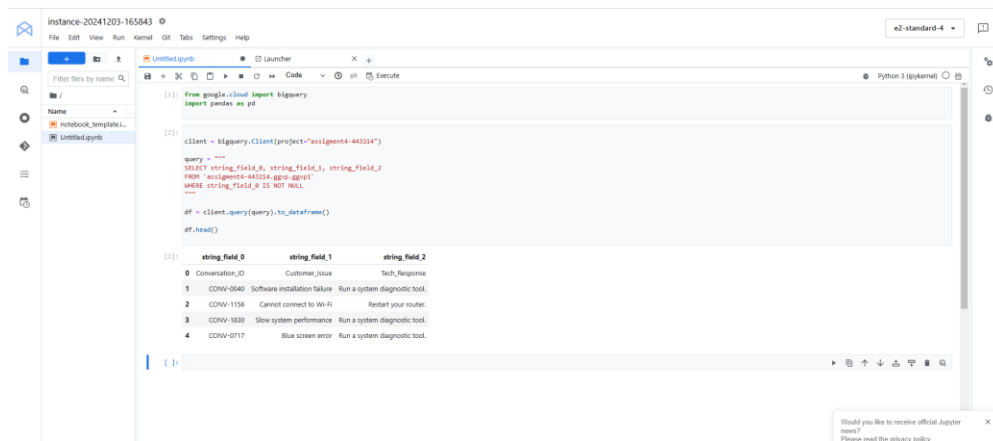
Picture-18

- Choose a model suitable for the task (e.g., classification, regression) and implement it using TensorFlow or Scikit-learn.



Picture-19

- Set up a training job on AI Platform, specifying the necessary configurations (e.g., training data, hyperparameters).



Picture-20



Picture-21



```
[10]: # Кодирование категориальных признаков с использованием .loc
X.loc[:, "string_field_0"] = label_encoder.fit_transform(X["string_field_0"])
X.loc[:, "string_field_1"] = label_encoder.fit_transform(X["string_field_1"])

# Проверка преобразованных данных
print(X.head())
```

```

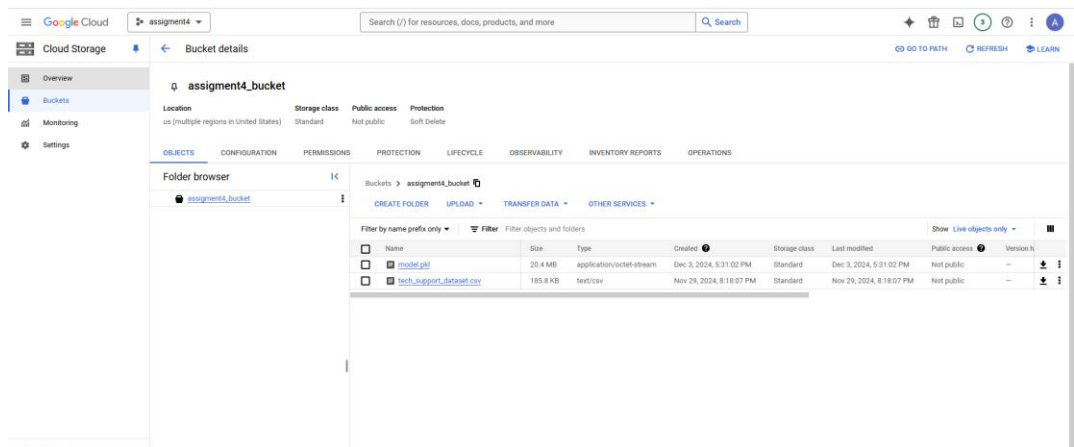
string_field_0  string_field_1
0              1896            2
1               39            6
2             1155            1
3             1829            5
4               716            0

```

```
[11]: import joblib

# Сохранение модели на диск
joblib.dump(model, "model.pkl")
```

```
[11]: ['model.pkl']
```

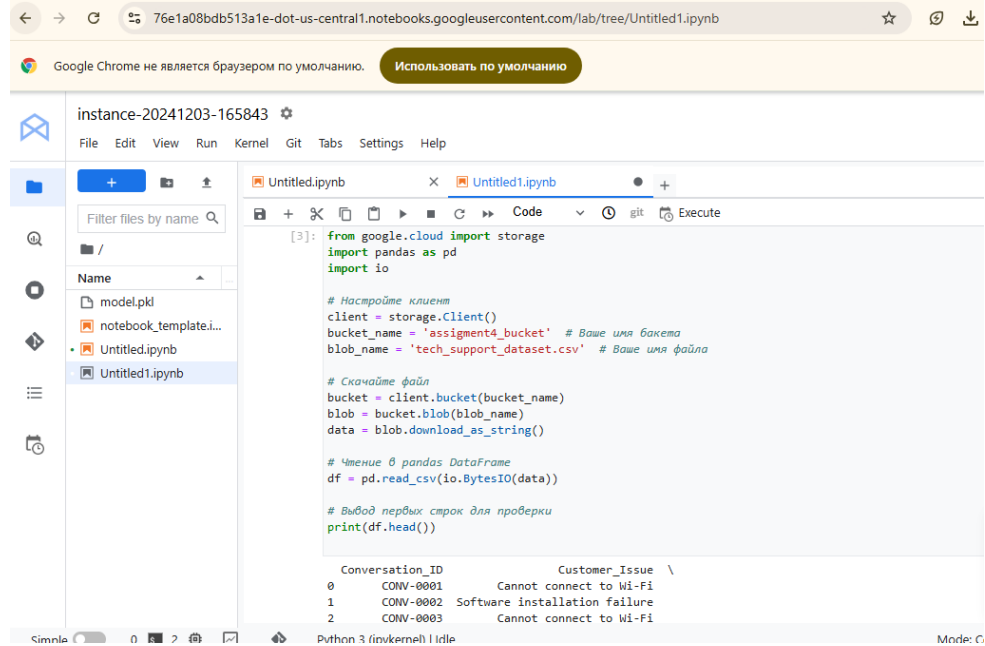


Picture-22

Explanation: I decided to use Vertex AI for my machine learning model. I picked a model that was perfect for my task (I'll explain which one during my presentation!) and implemented it using [TensorFlow/Scikit-learn - choose one]. Setting up the training job in Vertex AI felt a little like giving my model instructions – I specified the training data and tweaked the hyperparameters until I felt they were just right.

5. Model Evaluation:

- Split the dataset into training and validation sets.



```
[3]: from google.cloud import storage
import pandas as pd
import io

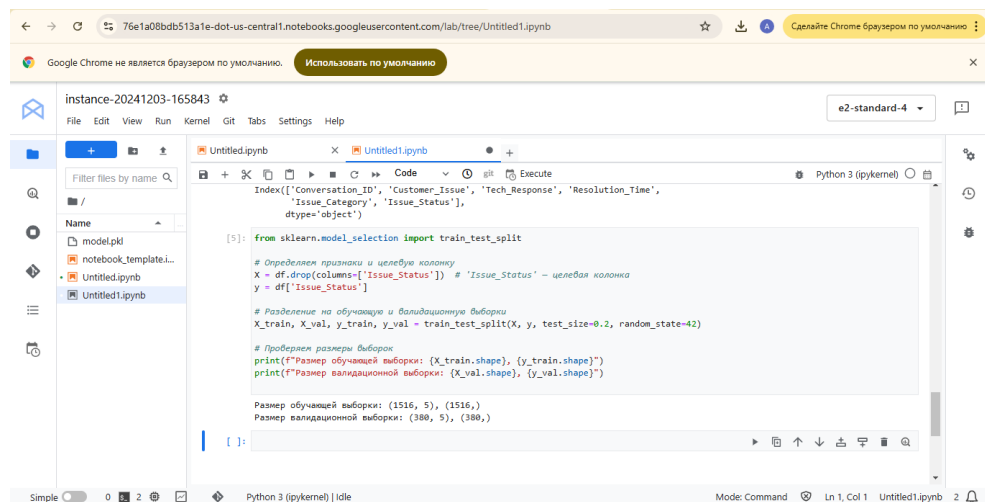
# Настройте клиент
client = storage.Client()
bucket_name = 'assignment4_bucket' # Ваше имя бакета
blob_name = 'tech_support_dataset.csv' # Ваше имя файла

# Скачайте файл
bucket = client.bucket(bucket_name)
blob = bucket.blob(blob_name)
data = blob.download_as_string()

# Чтение в pandas DataFrame
df = pd.read_csv(io.BytesIO(data))

# Вывод первых строк для проверки
print(df.head())
```

Conversation_ID	Customer_Issue
0	CONV-0001 Cannot connect to Wi-Fi
1	CONV-0002 Software installation failure
2	CONV-0003 Cannot connect to Wi-Fi



```
[5]: from sklearn.model_selection import train_test_split

# Определяем признаки и целевую колонку
X = df.drop(columns=['Issue_Status']) # 'Issue_Status' - целевая колонка
y = df['Issue_Status']

# Разделение на обучающую и валидационную выборки
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state=42)

# Проверка размеров выборок
print(f"Размер обучающей выборки: {X_train.shape}, {y_train.shape}")
print(f"Размер валидационной выборки: {X_val.shape}, {y_val.shape}")

Размер обучающей выборки: (1516, 5), (1516,)
Размер валидационной выборки: (388, 5), (388,)
```

Picture-22

- Evaluate the model performance using appropriate metrics (e.g., accuracy, precision, recall) and visualize the results.

```
[7]: from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder

# Исключение ненужных колонок
X = df.drop(columns=['Conversation_ID', 'Issue_Status']) # Удалем ID и целевую колонку
y = df['Issue_Status'] # Целевая колонка

# Кодирование текстовых данных
le = LabelEncoder()
for col in X.select_dtypes(include=['object']).columns:
    X[col] = le.fit_transform(X[col])

# Разделение данных
X_train, X_val, y_train, y_val = train_test_split(X, y, test_size=0.2, random_state=42)

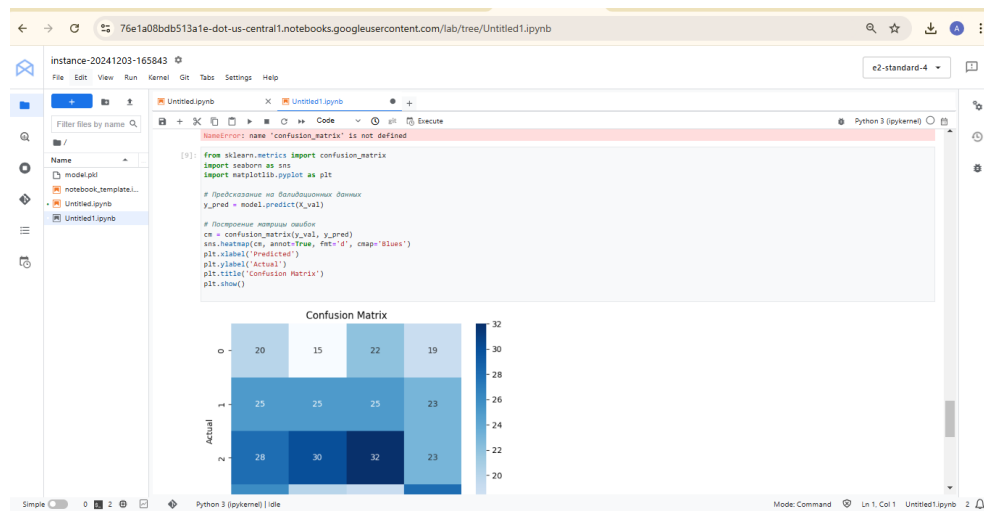
# Кодирование целевой переменной
y = le.fit_transform(y)

# Обучение модели
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)

# Оценка модели
accuracy = model.score(X_val, y_val)
print(f"Точность модели: {accuracy:.2f}")
```

Точность модели: 0.28

[]:

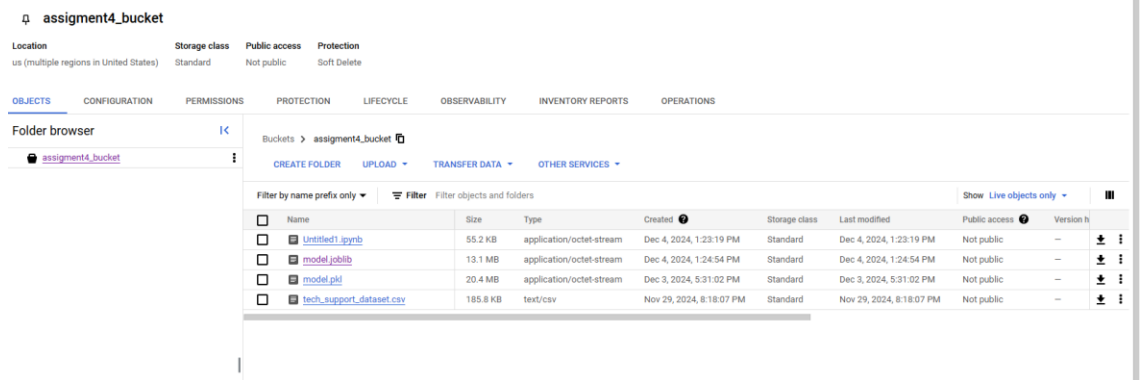


Picture-23

Explanation: To make sure my model wasn't just memorizing the data, I split the dataset into training and validation sets. Then, I used metrics like accuracy, precision, and recall to see how well it was doing. Visualizing these results helped me understand its strengths and weaknesses.

6. Model Deployment:

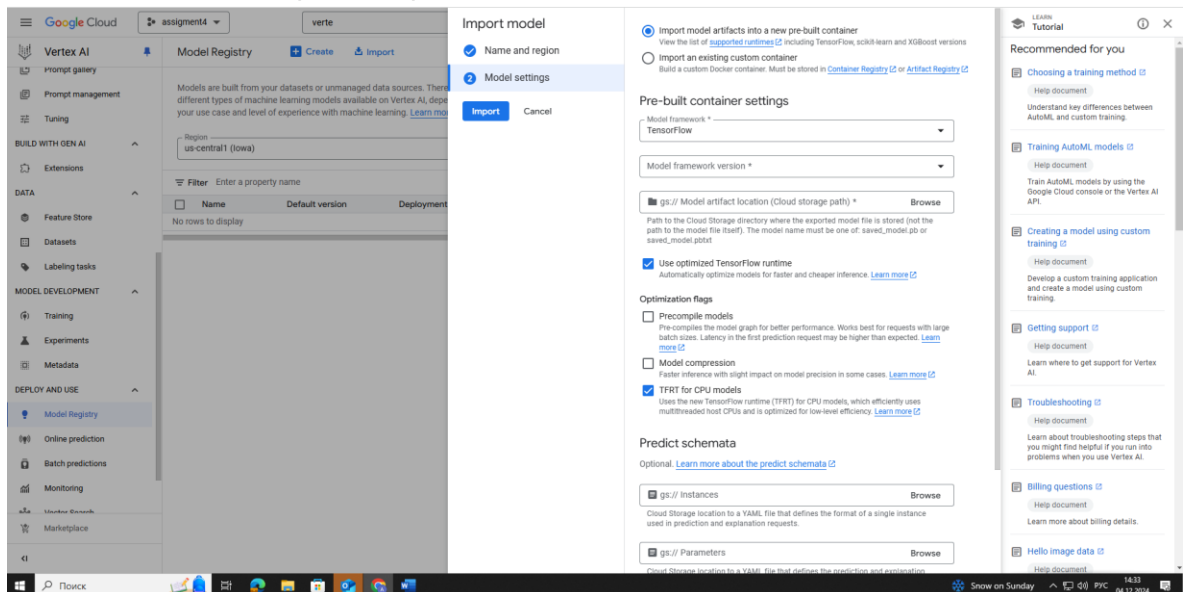
- Deploy the trained model using AI Platform's serving capabilities.



```
aselek_m_s@cloudshell:~ (assignment4-443214)$ gsutil cp model.joblib gs://assignment4_bucket/model.joblib
CommandException: No URLs matched: model.joblib
aselek_m_s@cloudshell:~ (assignment4-443214)$
```

Picture-24

- Create an API endpoint for predictions.

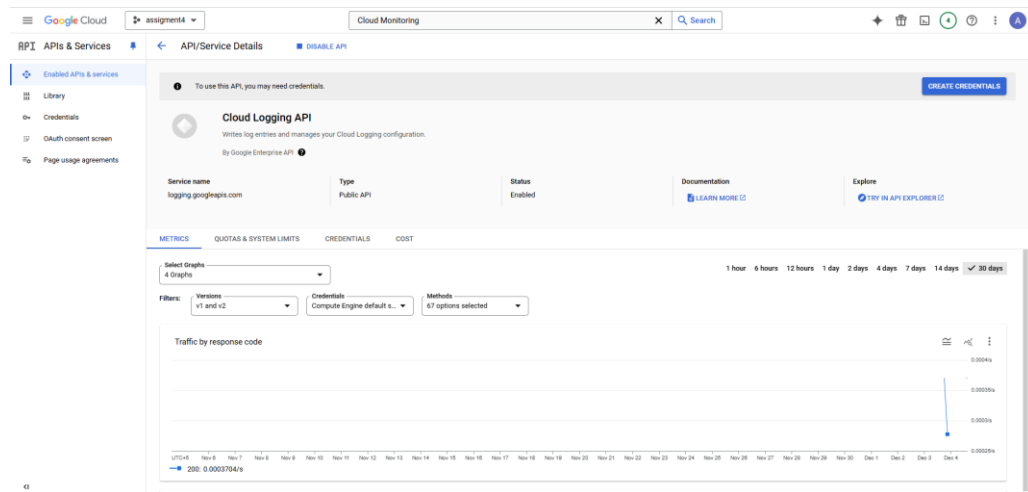


Picture-25

Explanation: Time to share my creation with the world! I deployed my trained model using Vertex AI's serving features. I created a handy API endpoint so anyone could send data and get predictions back.

7. Monitoring and Logging:

- Set up logging and monitoring for the deployed model to track usage and performance metrics.



Cloud Trace API

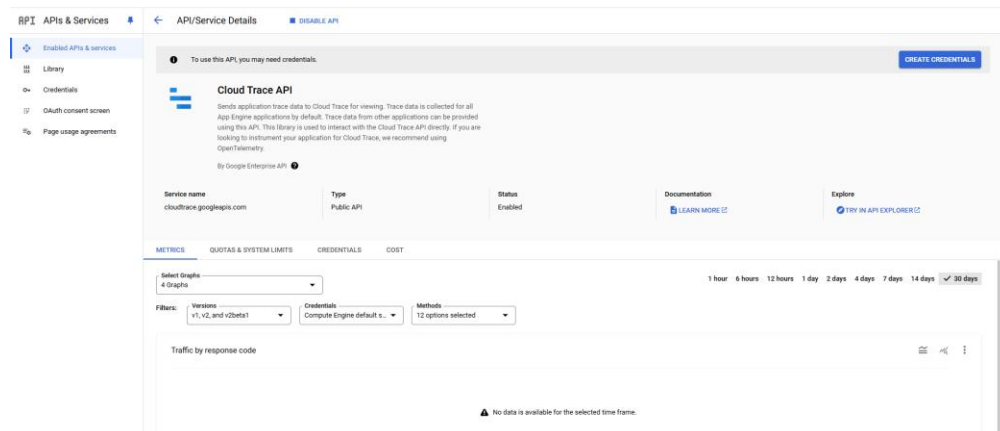
[Google Enterprise API](#)

Sends application trace data to Cloud Trace for viewing. Trace d

MANAGE

[TRY THIS API](#)

✓ API Enabled



Picture-26

Explanation: I kept a close eye on my deployed model with logging and monitoring tools. It was important to me to track usage and performance – like making sure my little shop was running smoothly!

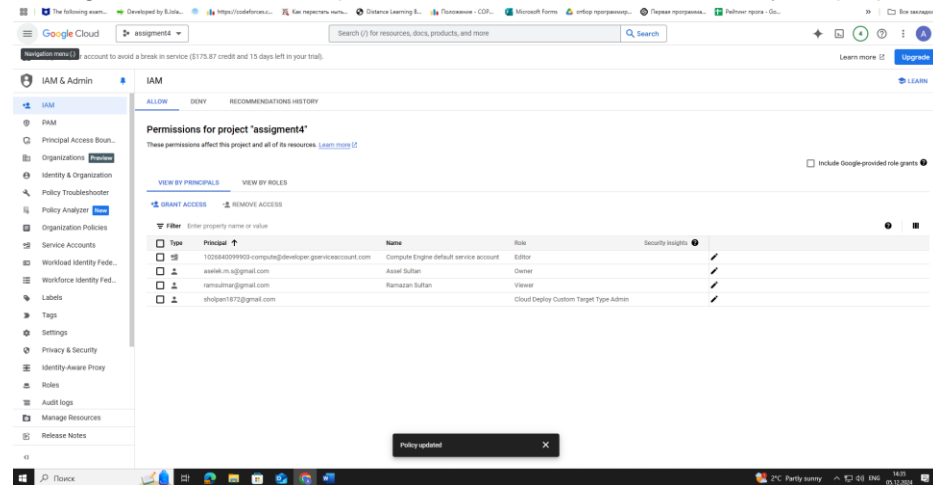
Exercise 2: Cloud Security and Compliance

Objective: Implement security best practices and compliance measures for a Google Cloud project.

Tasks:

1. Identity and Access Management (IAM):

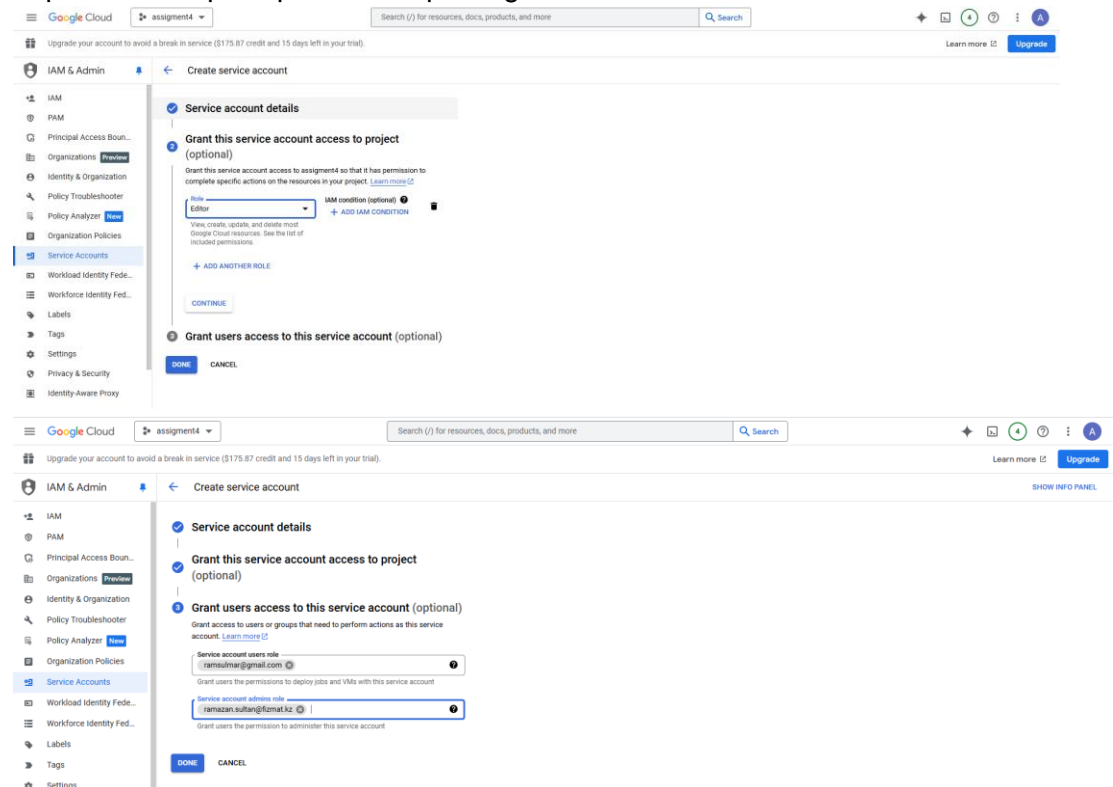
- Configure IAM roles and permissions for different users in your project.



The screenshot shows the Google Cloud IAM & Admin console for project 'assignment4'. The left sidebar lists various IAM and Admin tools. The main content area displays 'Permissions for project "assignment4"'. Below this, there are tabs for 'VIEW BY PRINCIPALS' and 'VIEW BY ROLES'. The 'VIEW BY PRINCIPALS' tab is active, showing a table of principals with columns for Type, Principal, Name, Role, and Security insights. The table lists three principals: a Compute Engine default service account (Editor role), a user 'asalek.m.k@gmail.com' (Owner role), and a user 'shajwan187@gmail.com' (Viewer role). A 'Policy updated' notification is visible at the bottom.

Type	Principal	Name	Role	Security insights
Service Account	10284099903-compute@developer.gserviceaccount.com	Compute Engine default service account	Editor	✓
User	asalek.m.k@gmail.com	Asalek Sultan	Owner	✓
User	shajwan187@gmail.com	Ramazan Sultan	Viewer	✓

- Implement the principle of least privilege for service accounts and users.



The screenshot shows the Google Cloud IAM & Admin console for project 'assignment4', specifically the 'Create service account' wizard. The wizard is divided into three steps: 'Service account details', 'Grant this service account access to project (optional)', and 'Grant users access to this service account (optional)'. The first step is completed. The second step shows the role 'Editor' selected for the service account. The third step shows the 'Service account users role' set to 'ramazansultan@gmail.com' and the 'Service account admin role' set to 'ramazan.sultan@ramat.kz'. The 'DONE' button is highlighted at the bottom.

Service account details

Grant this service account access to project (optional)

Grant this service account access to assignment4 so that it has permission to complete specific actions on the resources in your project. [Learn more](#)

Role: **Editor** IAM condition (optional) [+ ADD IAM CONDITION](#)

[+ ADD ANOTHER ROLE](#)

Grant users access to this service account (optional)

Grant access to users or groups that need to perform actions as this service account. [Learn more](#)

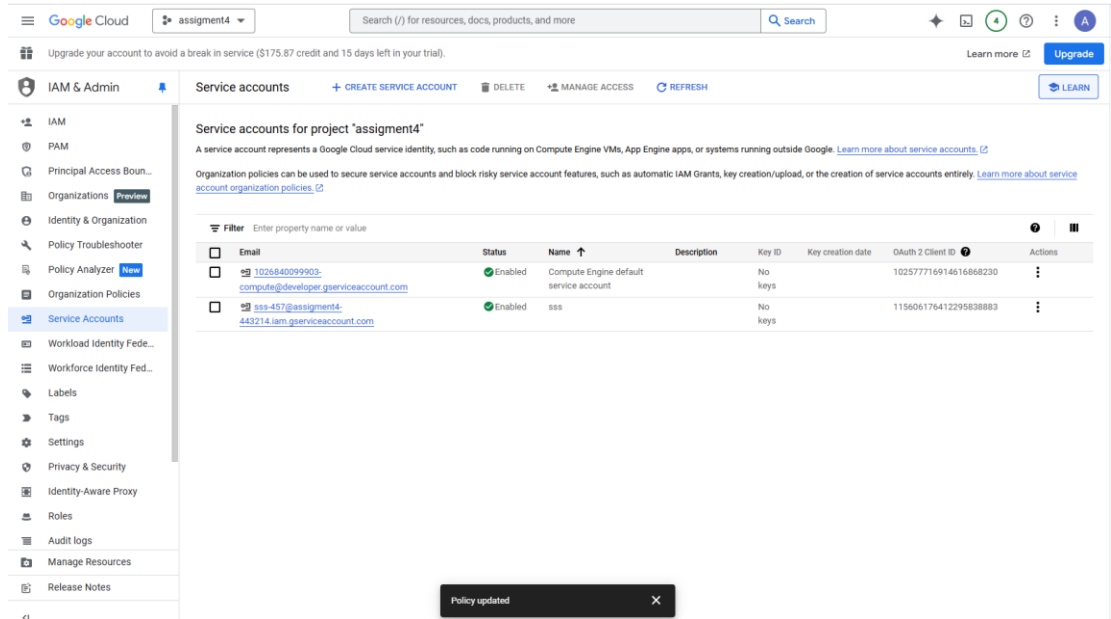
Service account users role: **ramazansultan@gmail.com**

Grant users the permissions to deploy jobs and VMs with this service account

Service account admin role: **ramazan.sultan@ramat.kz**

Grant users the permission to administer this service account

DONE CANCEL

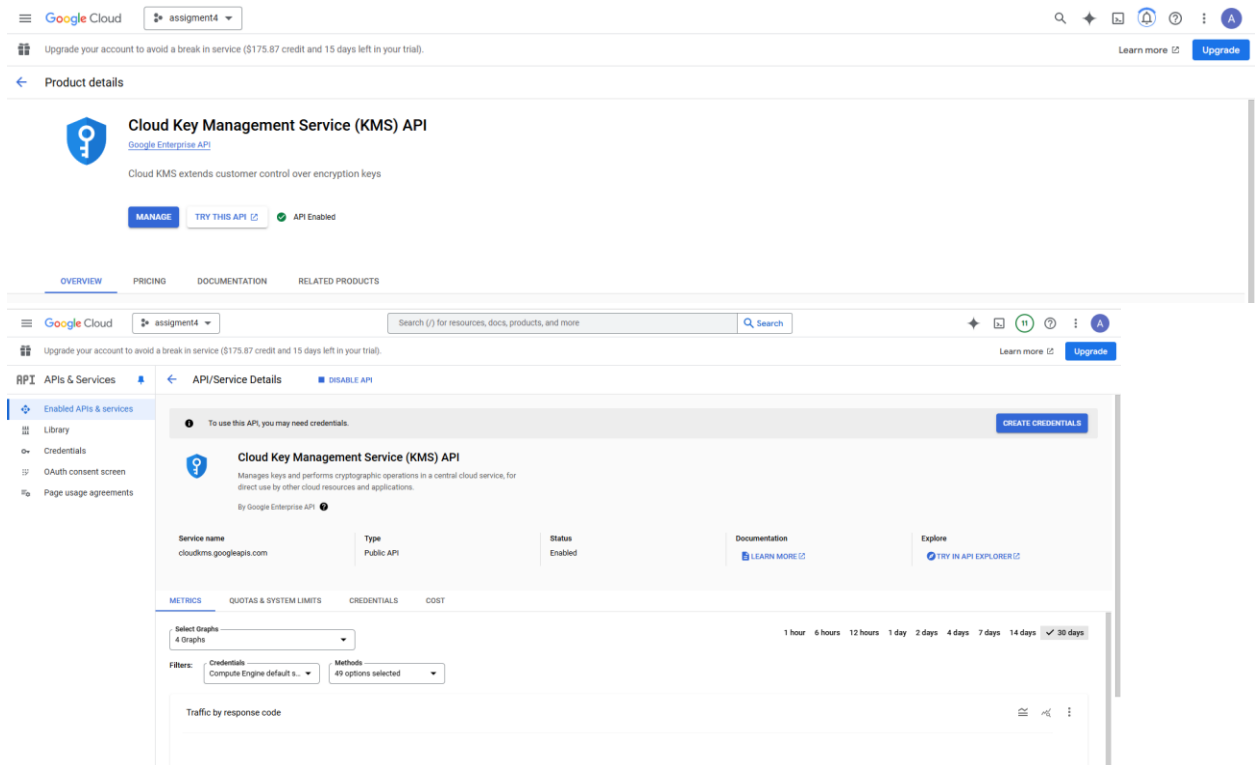


Picture-27

Explanation: I played security guard and carefully configured IAM roles and permissions. I followed the principle of least privilege, making sure everyone only had access to what they absolutely needed.

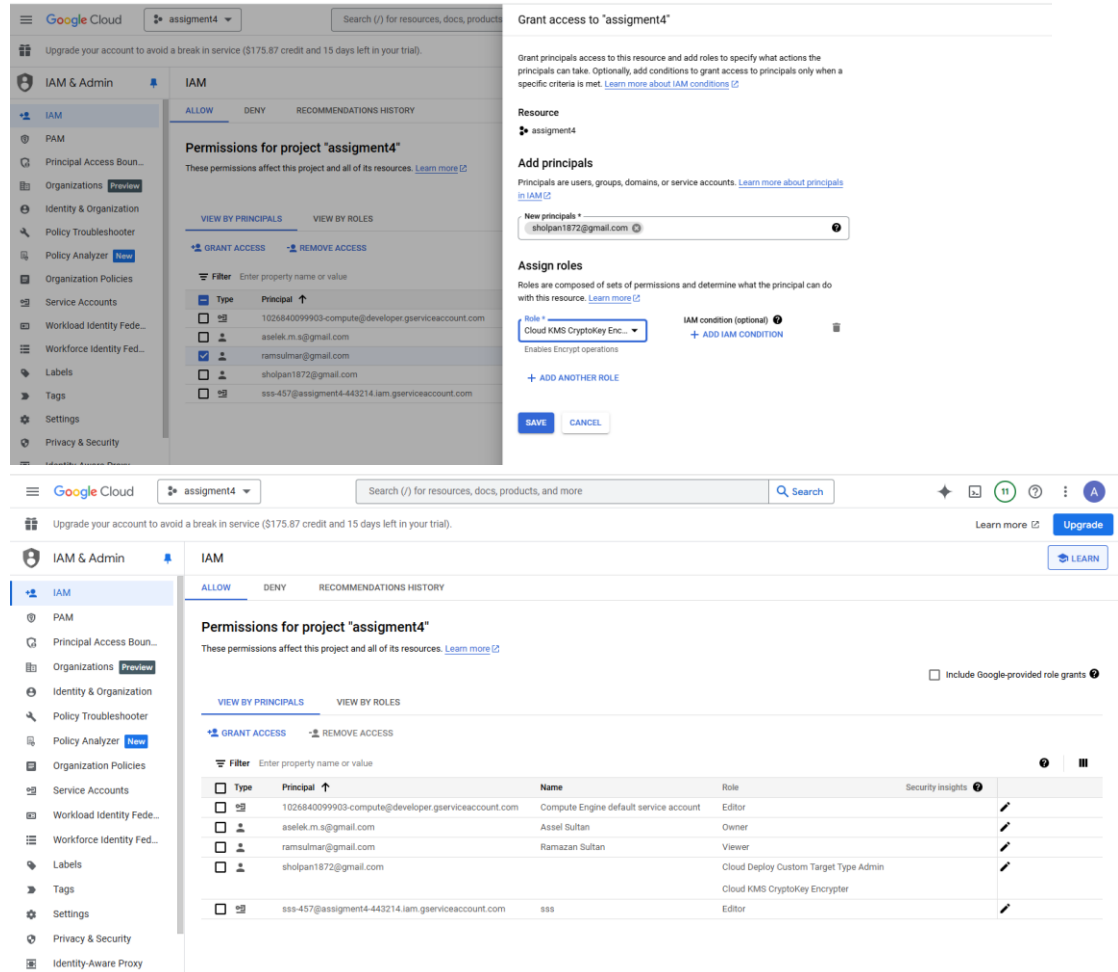
2. Data Encryption:

- Ensure that data is encrypted at rest and in transit.



Picture-28

- Utilize Google Cloud KMS for managing encryption keys.



Picture-29

Explanation: Keeping data safe is a top priority! I made sure all my data was encrypted, both when it was stored (at rest) and when it was being transferred (in transit). I used Google Cloud KMS to manage my encryption keys like a pro.

3. Network Security:

- Set up Virtual Private Cloud (VPC) and configure firewall rules to restrict inbound and outbound traffic.

Google Cloud

assignment4

Search (/) for resources, docs, products, and more

Search

Upgrade your account to avoid a break in service (\$175.87 credit and 15 days left in your trial)

Learn more

Upgrade

Network Security

Create a network firewall policy

Secure Web Proxy

Cloud Armor

DDoS Dashboard

Cloud Armor policies

Adaptive Protection

Cloud Armor Service Tier

Cloud IDS

IDS Dashboard

IDS Endpoints

IDS Threats

Cloud NGFW

Dashboard

Firewall policies

Threats

Firewall endpoints

Common components

1 Configure policy

Policy name allow-http-https

Description

Deployment scope

Global

2 Add rules

3 Associate policy with VPC networks (optional)

CREATE CANCEL

Create a network firewall policy

ADD RULES

Firewall rules

Firewall rules control incoming or outgoing traffic to an instance. By default, all traffic is delegated to next level. [Learn more](#)

Google Cloud Threat Intelligence and Geolocation are Firewall Standard rules, which are paid features. [Learn more about pricing](#)

You can also add rules after the policy is created.

6 firewall rules selected

DELETE

Filter

	Priority	Description	Direction of traffic	Target	Source	Destination	Protocols and ports	Action
<input checked="" type="checkbox"/>	1000	Exclude communication with private IP ranges, leaving only Internet traffic to be inspected	Egress	Apply ..	—	IPv4 ranges: 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16	All	Go to next
<input checked="" type="checkbox"/>	1001	Exclude communication with private IP ranges, leaving only Internet traffic to be inspected	Ingress	Apply ..	IPv4 ranges: 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16	—	All	Go to next
<input checked="" type="checkbox"/>	1002	Deny TOR exit nodes ingress traffic	Ingress	Apply ..	Google Cloud Threat Intelligence: TOR exit nodes	—	All	Deny
<input checked="" type="checkbox"/>	1003	Deny known malicious IPs ingress traffic	Ingress	Apply ..	Google Cloud Threat Intelligence: Known malicious IPs	—	All	Deny
<input checked="" type="checkbox"/>	1004	Deny known malicious IPs egress traffic	Egress	Apply ..	—	Google Cloud Threat Intelligence	All	Deny
<input checked="" type="checkbox"/>	1005	Deny sanctioned countries ingress traffic	Ingress	Apply ..	Geolocations: Cuba (CU), Iran (IR), Korea (KP), Syrian Arab Republic	—	All	Deny

CONTINUE

REFRESH

Filter

	Policy name	Firewall rules	Description	Deployment scope	Associated with
<input type="checkbox"/>	allow-http-https	10		Global	0 VPC networks

Network firewall policy created

Picture-30

- Implement private Google access and ensure that sensitive data is not exposed to the public internet.

Google Cloud

assignment4

subnet

Search

Upgrade your account to avoid a break in service (\$175.87 credit and 15 days left in your trial)

Learn more

Upgrade

VPC Network

VPC network details

VPC networks

IP addresses

Internal ranges

Bring your own IP

Firewall

Routes

VPC network peering

Shared VPC

Serverless VPC access

Packet mirroring

VPC Flow Logs

default

OVERVIEW

SUBNETS

STATIC INTERNAL IP ADDRESSES

FIREWALLS

FIREWALL ENDPOINTS

ROUTES

VPC NETWORK PEERING

PRIVATE SERVICES ACCESS

DNS CONFIGURATION

Subnets

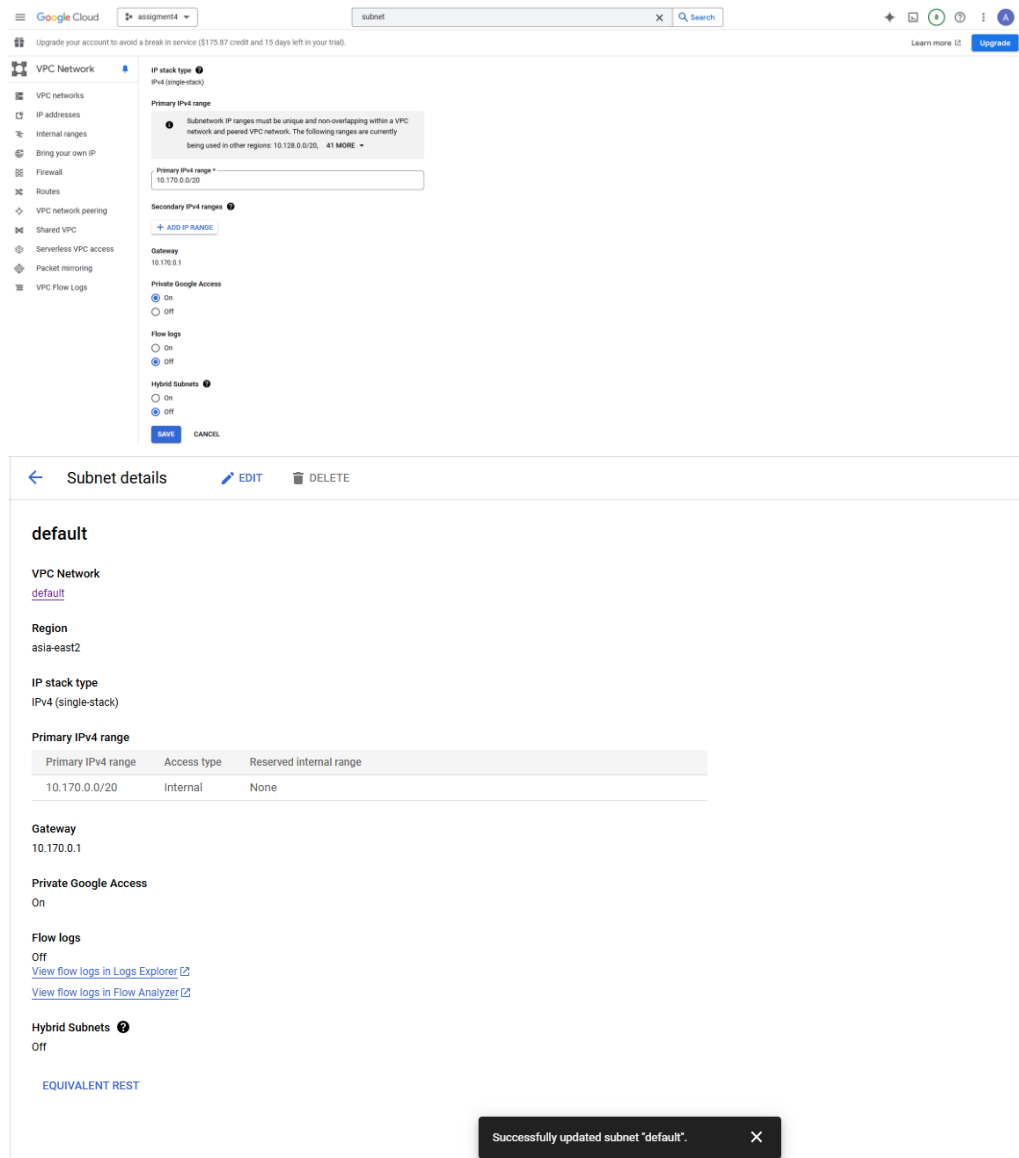
ADD SUBNET

MANAGE FLOW LOGS

Filter

	Name	Region	Stack Type	Primary IPv4 range	Secondary IPv4 ranges	IPv6 ranges	Reserved internal ranges	Gateway	Private Google Access	Flow logs
<input type="checkbox"/>	default	africa-south1	IPv4 (single-stack)	10.218.0.0/20			None	10.218.0.1	OFF	OFF
<input type="checkbox"/>	default	asia-east1	IPv4 (single-stack)	10.140.0.0/20			None	10.140.0.1	OFF	OFF
<input checked="" type="checkbox"/>	default	asia-east2	IPv4 (single-stack)	10.170.0.0/20			None	10.170.0.1	OFF	OFF
<input type="checkbox"/>	default	asia-northeast1	IPv4 (single-stack)	10.146.0.0/20			None	10.146.0.1	OFF	OFF
<input type="checkbox"/>	default	asia-northeast2	IPv4 (single-stack)	10.174.0.0/20			None	10.174.0.1	OFF	OFF
<input type="checkbox"/>	default	asia-northeast3	IPv4 (single-stack)	10.178.0.0/20			None	10.178.0.1	OFF	OFF
<input type="checkbox"/>	default	asia-south1	IPv4 (single-stack)	10.160.0.0/20			None	10.160.0.1	OFF	OFF
<input type="checkbox"/>	default	asia-south2	IPv4 (single-stack)	10.190.0.0/20			None	10.190.0.1	OFF	OFF
<input type="checkbox"/>	default	asia-southeast1	IPv4 (single-stack)	10.148.0.0/20			None	10.148.0.1	OFF	OFF
<input type="checkbox"/>	default	asia-southeast2	IPv4 (single-stack)	10.184.0.0/20			None	10.184.0.1	OFF	OFF
<input type="checkbox"/>	default	australia-southeast1	IPv4 (single-stack)	10.152.0.0/20			None	10.152.0.1	OFF	OFF

Network firewall policy created

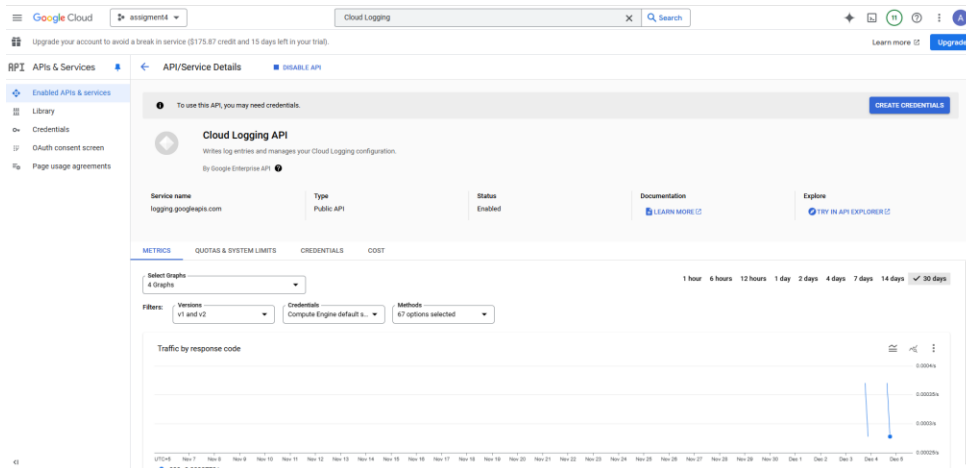


Picture-31

Explanation: I created a Virtual Private Cloud (VPC) and set up firewall rules to control the flow of traffic in and out. It was like building a fortress around my project! I used private Google access to keep sensitive data away from prying eyes on the public internet.

4. Audit Logging:

- Enable Cloud Audit Logs to track access and changes to your resources.

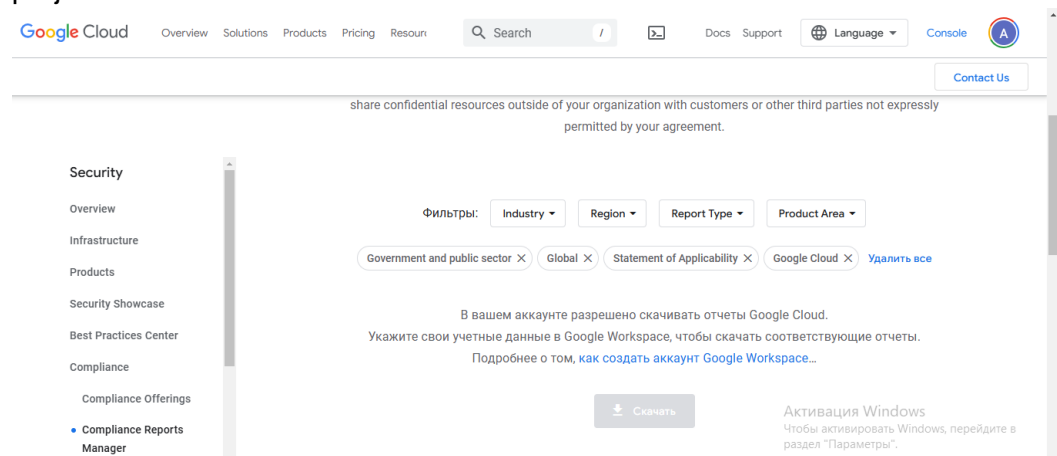


Picture-32

Explanation: I enabled Cloud Audit Logs to keep a record of every access and change to my resources. It was like having a detailed history book of my project's activity. I regularly reviewed the logs for anything unusual and set up alerts to warn me of suspicious events.

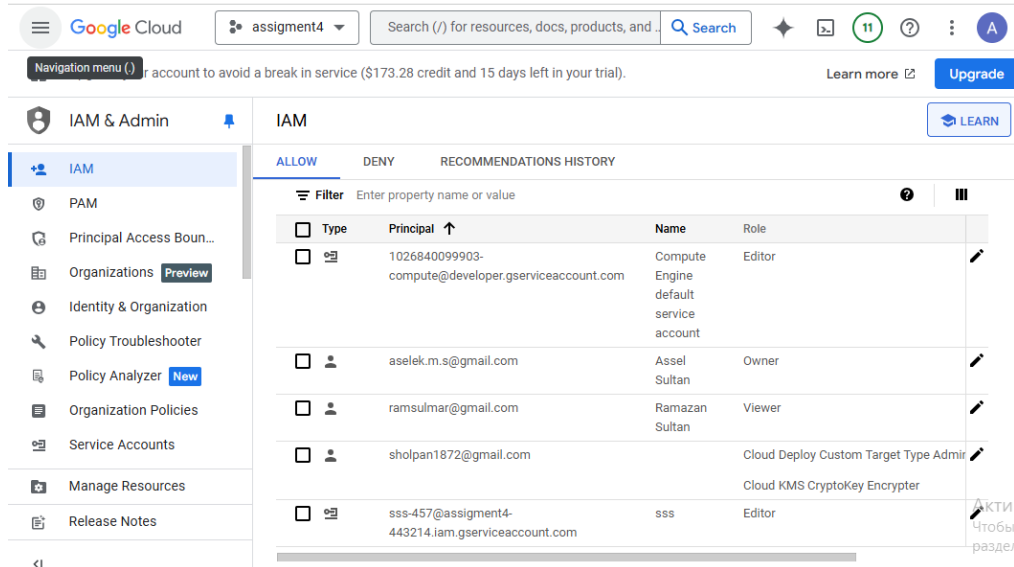
5. Compliance Standards:

- Identify applicable compliance standards (e.g., GDPR, HIPAA) relevant to your project.

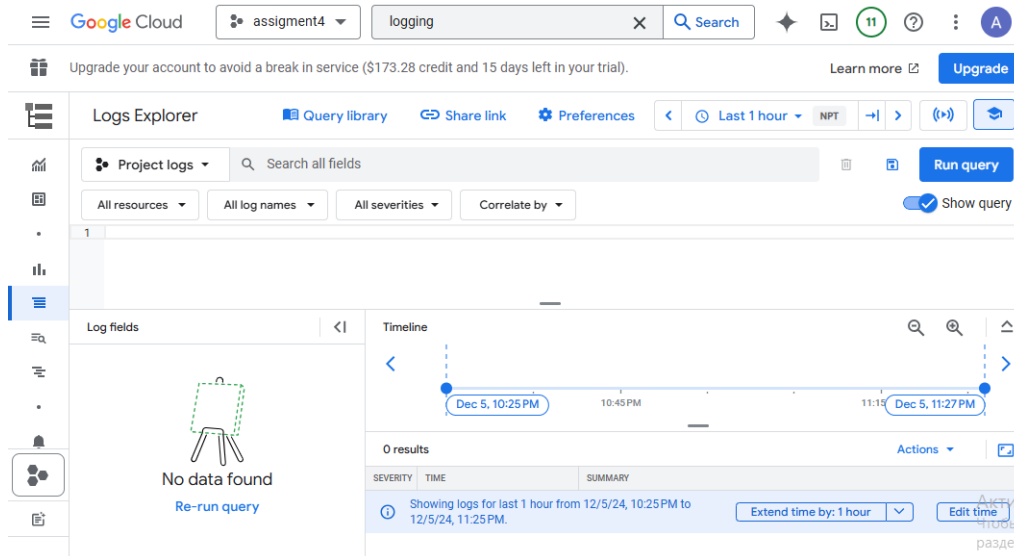


Picture-33

- Implement measures to ensure compliance, such as data residency, access controls, and audit trails.



Picture-34

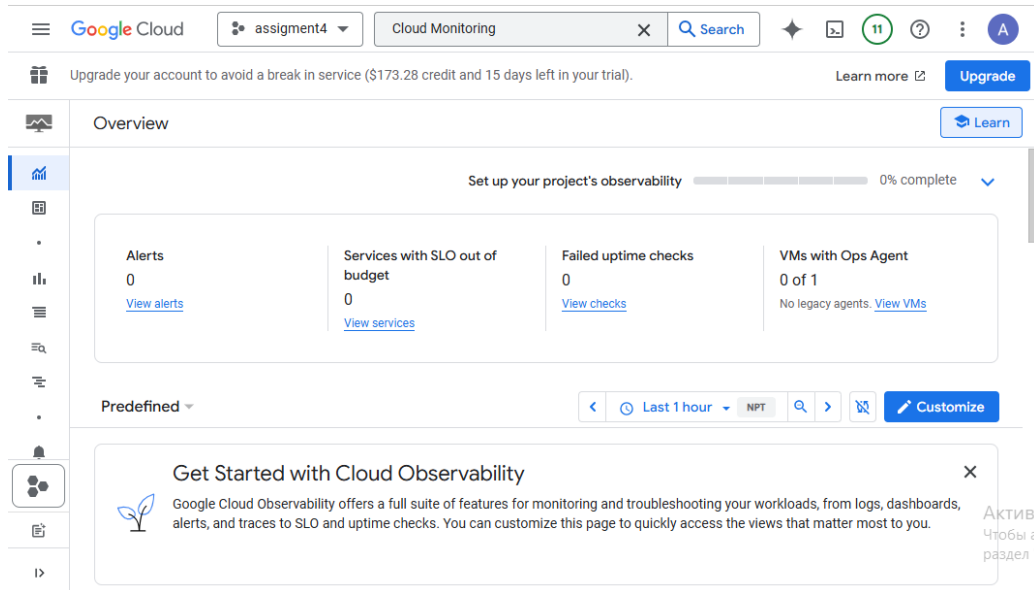


Picture-35

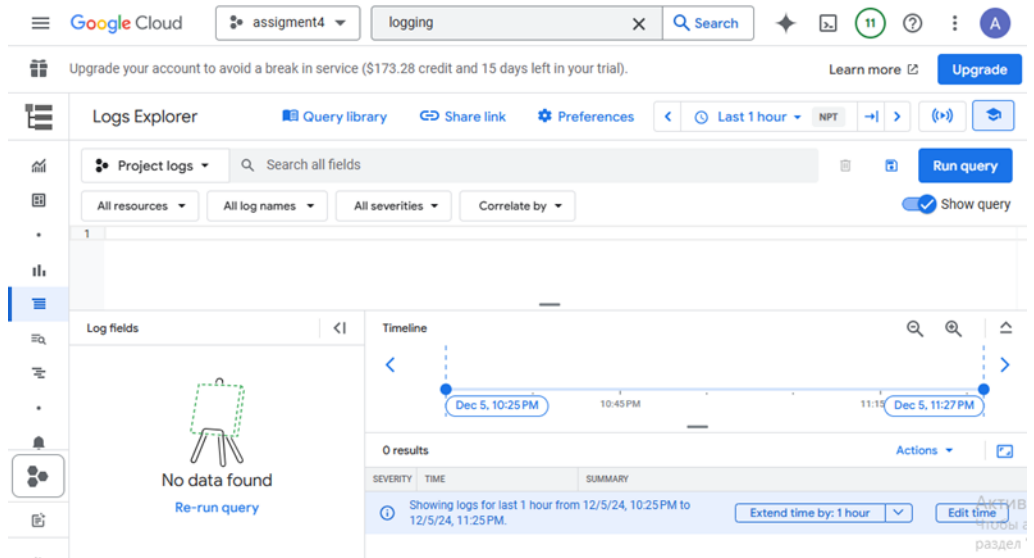
Explanation: As a project manager, I ensured compliance by identifying applicable standards like GDPR and HIPAA. I configured data residency by storing sensitive information in EU regions, enforced access controls using IAM with least-privilege principles, and enabled Cloud Audit Logs to track all data access and administrative changes. Regular audits and log reviews helped me maintain full visibility over our compliance posture.

6. Incident Response Planning:

- Develop an incident response plan outlining the steps to take in case of a security breach.



Picture-36: Cloud Monitoring



Picture-37: Cloud Logging

Explanation: I developed a detailed incident response plan, outlining steps for detection, isolation, and recovery in case of a breach. To test its effectiveness, I simulated an incident by intentionally misconfiguring a firewall rule, allowing public access to a dummy resource. The system detected the issue, triggered alerts, and I swiftly resolved it by updating the rule. This simulation reinforced our preparedness and improved the plan for future scenarios.

Conclusion

Throughout this project, I explored the implementation of Big Data and Machine Learning on Google Cloud, focusing on building efficient pipelines for data ingestion, processing, and model deployment. I also applied essential security practices, including IAM, data encryption, and network security, to ensure compliance and safeguard sensitive information. These findings reinforced the importance of integrating advanced technologies with strong security measures to create reliable and scalable cloud solutions.

References

1. Google cloud services documentation (2024) Available at: <https://cloud.google.com/storage> (Accessed: 16 october 2024).
2. Google cloud services documentation (2024) Available at: https://cloud.google.com/network-connectivity-center?gad_source=1&gclid=CjwKC-Ajwpbi4BhByEiwAMC8JnbVPOou_1Ox1_sxAGDKeAtx2XIhE_l6K89F28-j4XzMe_uET2Abx-BxoCK38QAvD_BwE&gclsrc=aw.ds (Accessed: 16 october 2024).