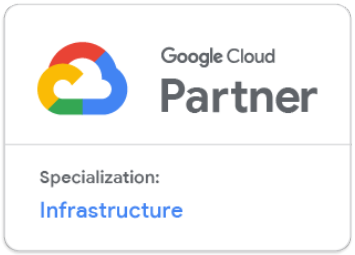
TSMC Career Hack 2025 User Guide (ICSD)

Date: 2025-2-4



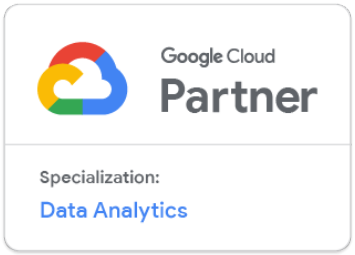


Table of contents

1. User Accounts

每位學生有一個 account，由題目組提供。
範例：AAID 第一組學生
AAID User 01 TSMC aaid-user01.tsmc@hackathon.cloudmile.vip
AAID User 02 TSMC aaid-user02.tsmc@hackathon.cloudmile.vip
AAID User 03 TSMC aaid-user03.tsmc@hackathon.cloudmile.vip
AAID User 04 TSMC aaid-user04.tsmc@hackathon.cloudmile.vip
以此類推

2. 使用時數限制

- 2/7 - 2/15 每項資源共可使用 60 小時
- 請注意 dashboard [CareerHack Resource Timing Dashboard (Student View)]

機器不使用時，請記得關機。以下所列資源每組上限使用 60 小時。

AAID

project_id	GCE	Workbench-0	Workbench-1	Workbench-2
aaid-test	0:00:00	0:16:35	5:41:00	0:00:00
aaid-1	0:00:00	0:00:00	0:00:00	0:00:00
aaid-2	0:00:00	0:00:00	0:00:00	0:00:00
aaid-3	0:00:00	0:00:00	0:00:00	0:00:00
aaid-4	0:00:00	0:00:00	0:00:00	0:00:00
aaid-5	0:00:00	0:00:00	0:00:00	0:00:00
aaid-6	0:00:00	0:00:00	0:00:00	0:00:00

ICSD

project_id	GCE	SQL-vector	GKE
icsd-test	0:00:00	0:31:12	2:01:57
icsd-1	0:00:00	0:00:00	0:00:00
icsd-2	0:00:00	0:00:00	0:00:00
icsd-3	0:00:00	0:00:00	0:00:00
icsd-4	0:00:00	0:00:00	0:00:00
icsd-5	0:00:00	0:00:00	0:00:00
icsd-6	0:00:00	0:00:00	0:00:00

BSID

project_id	GCE	SQL-relational	SQL-vector	Vector Search
bsid-test	0:00:00	0:00:00	0:00:00	6:00:16
bsid-1	0:00:00	0:00:00	0:00:00	0:00:00
bsid-2	0:00:00	0:00:00	0:00:00	0:00:00
bsid-3	0:00:00	0:00:00	0:00:00	0:00:00
bsid-4	0:00:00	0:00:00	0:00:00	0:00:00
bsid-5	0:00:00	0:00:00	0:00:00	0:00:00
bsid-6	0:00:00	0:00:00	0:00:00	0:00:00

TSID

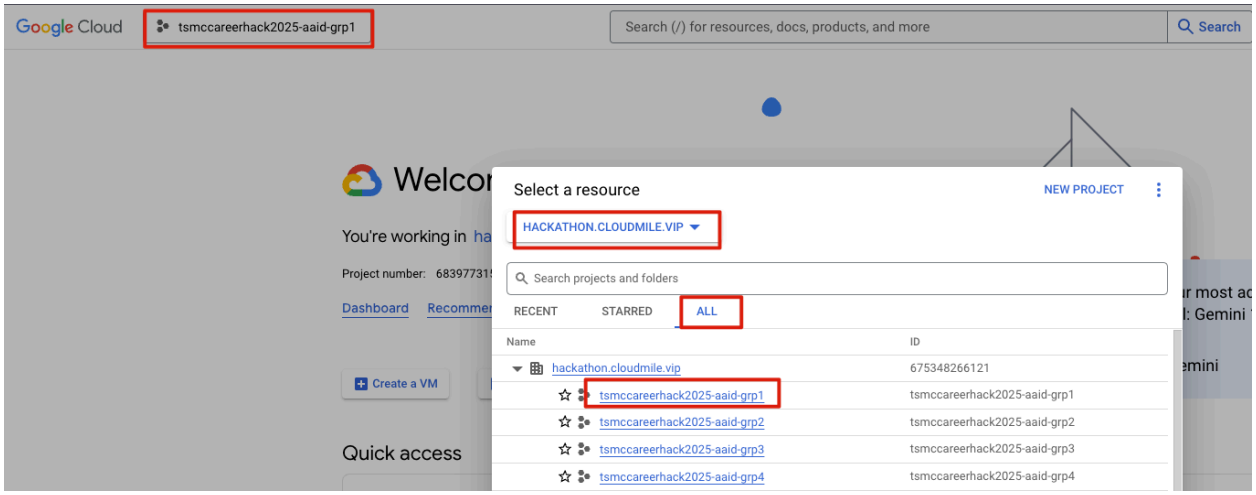
project_id	GCE	GKE
tsid-test	0:01:15	0:30:51
tsid-1	0:00:00	0:00:00
tsid-2	0:00:00	0:00:00
tsid-3	0:00:00	0:00:00
tsid-4	0:00:00	0:00:00
tsid-5	0:00:00	0:00:00
tsid-6	0:00:00	0:00:00

3. Login GCP Console

https://console.cloud.google.com
(Need to reset password at the first login)

4. Select Project

Project ID: tsmccareerhack2025-icsd-grp#
（下圖僅供參考，請選擇自己實際所屬組別的 folder 和 project。）

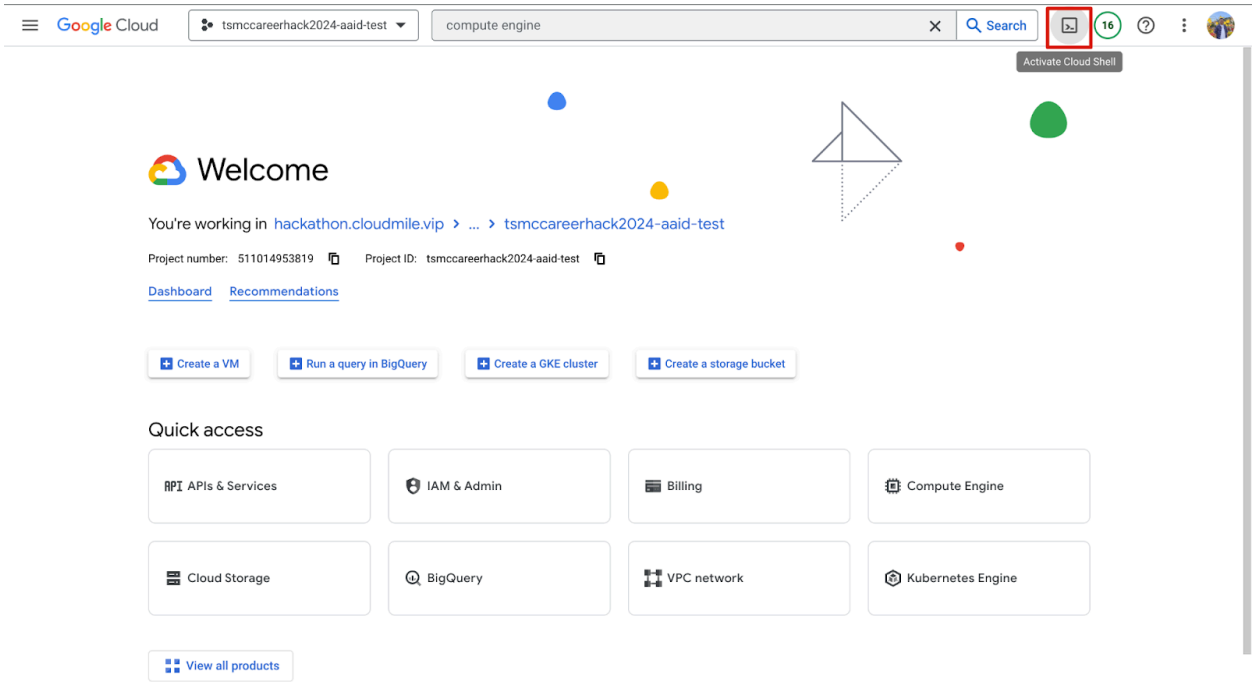


5. Virtual Machine

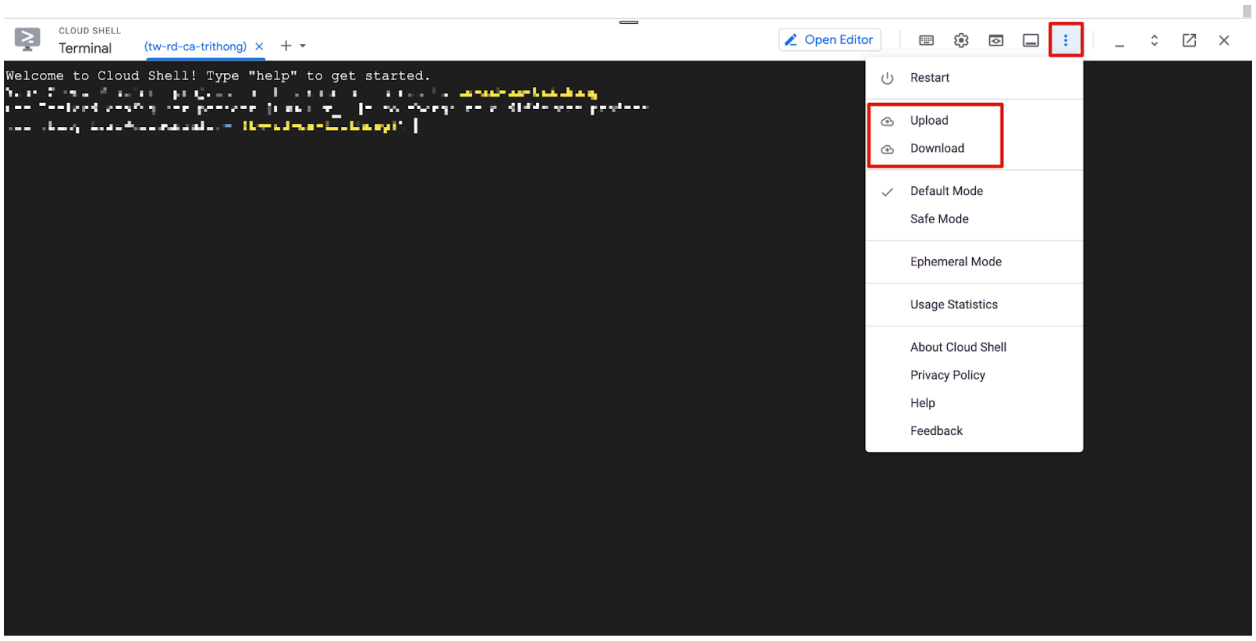
- Cloud Shell

  Cloud Shell 供使用者在 GCP Console 上快速簡單使用 Linux 和 Google Cloud SDK 指令操作。

- 開啟 Cloud Shell：



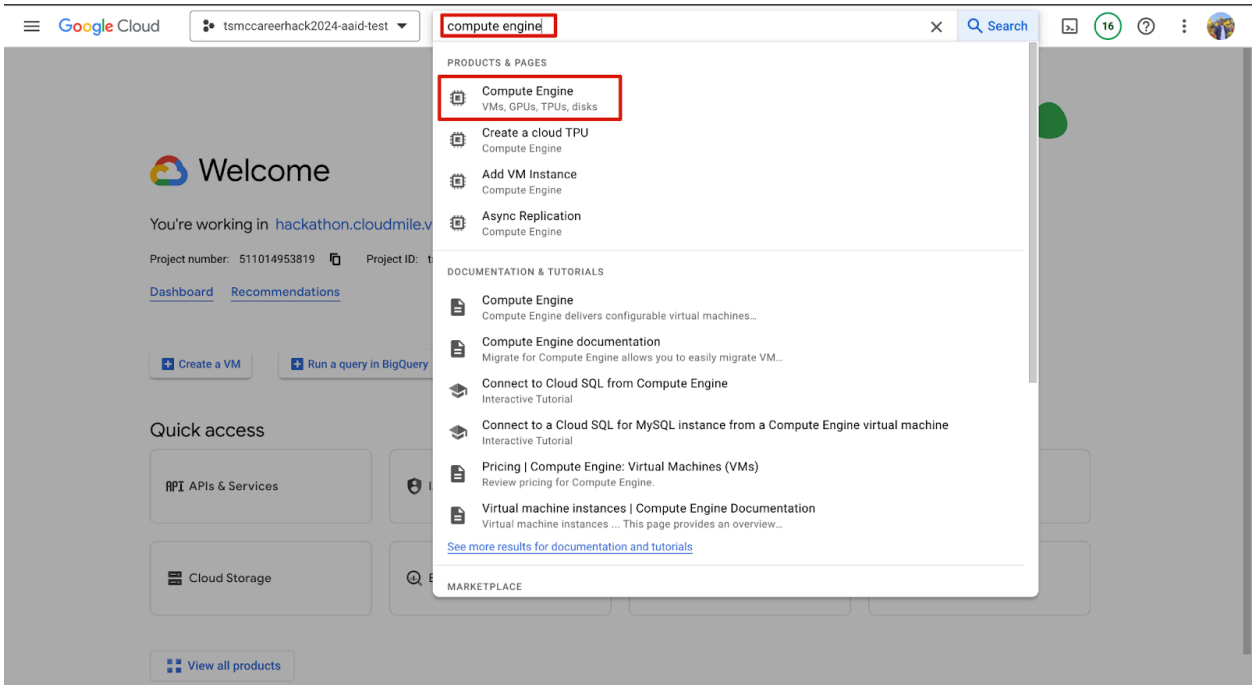
- 可透過右上角功能選單從虛擬主機下載檔案或上傳本地檔案到虛擬主機：



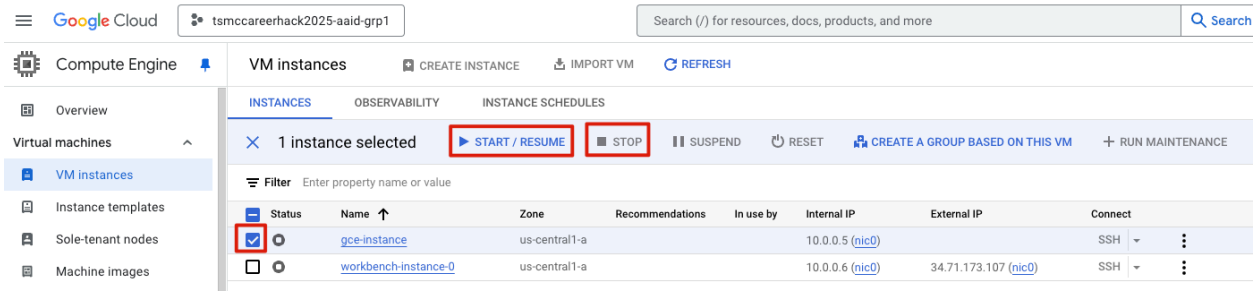
○ 虛擬主機 (Google Compute Engine, GCE)

虛擬主機供參賽組別公用操作環境，可使用 Google Cloud SDK、可部署應用程式。使用虛擬主機的操作步驟如下：

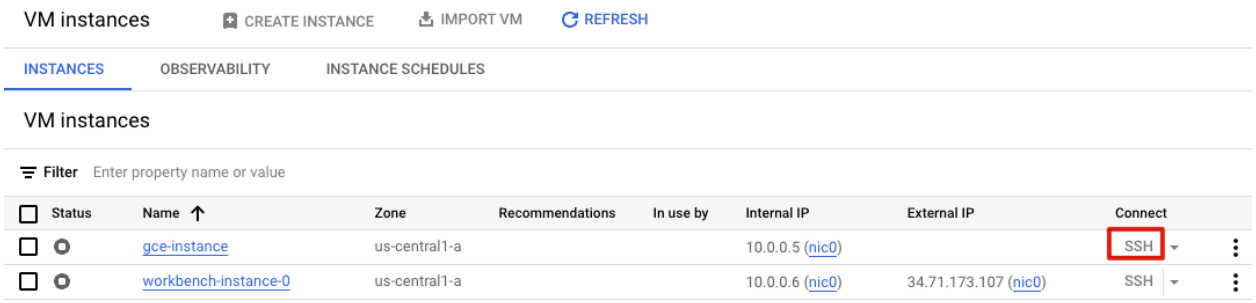
- 進入 Compute Engine 服務頁面：



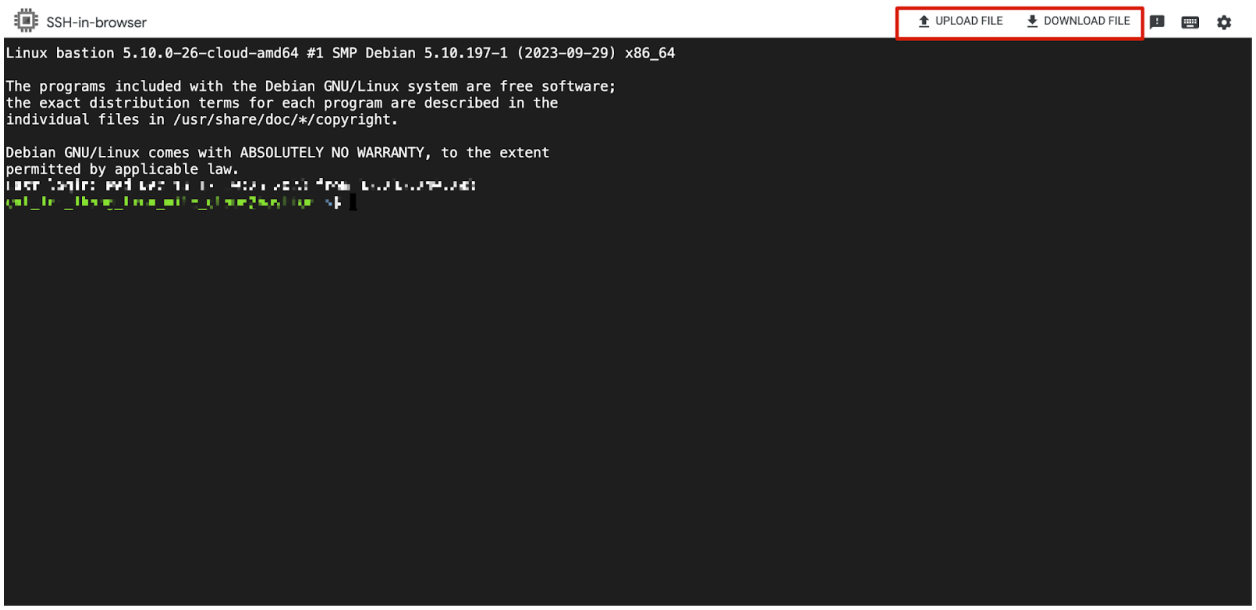
- 勾選機器，點擊 Start 開機、Stop 關機（**請記得不使用時要關機！**）：



- 開機後點擊右端 “SSH” 按鈕進行 SSH 連線：



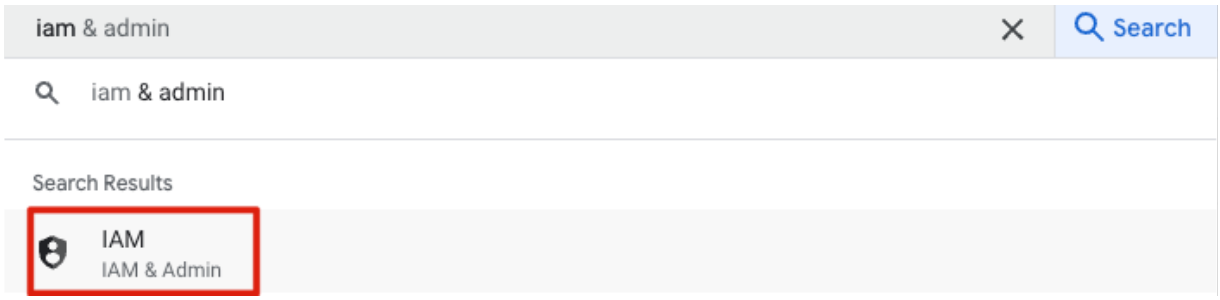
- 可透過右上角功能選單從虛擬主機下載檔案或上傳本地檔案到虛擬主機：



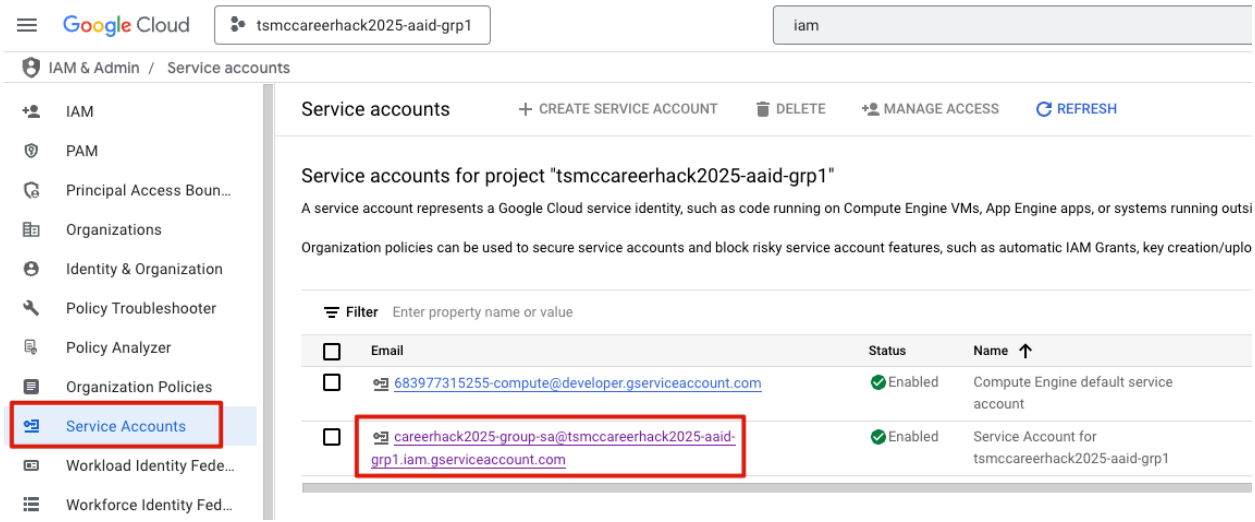
更多連線方式，請參考：https://cloud.google.com/compute/docs/connect/ssh-using-iap

6. Service Account

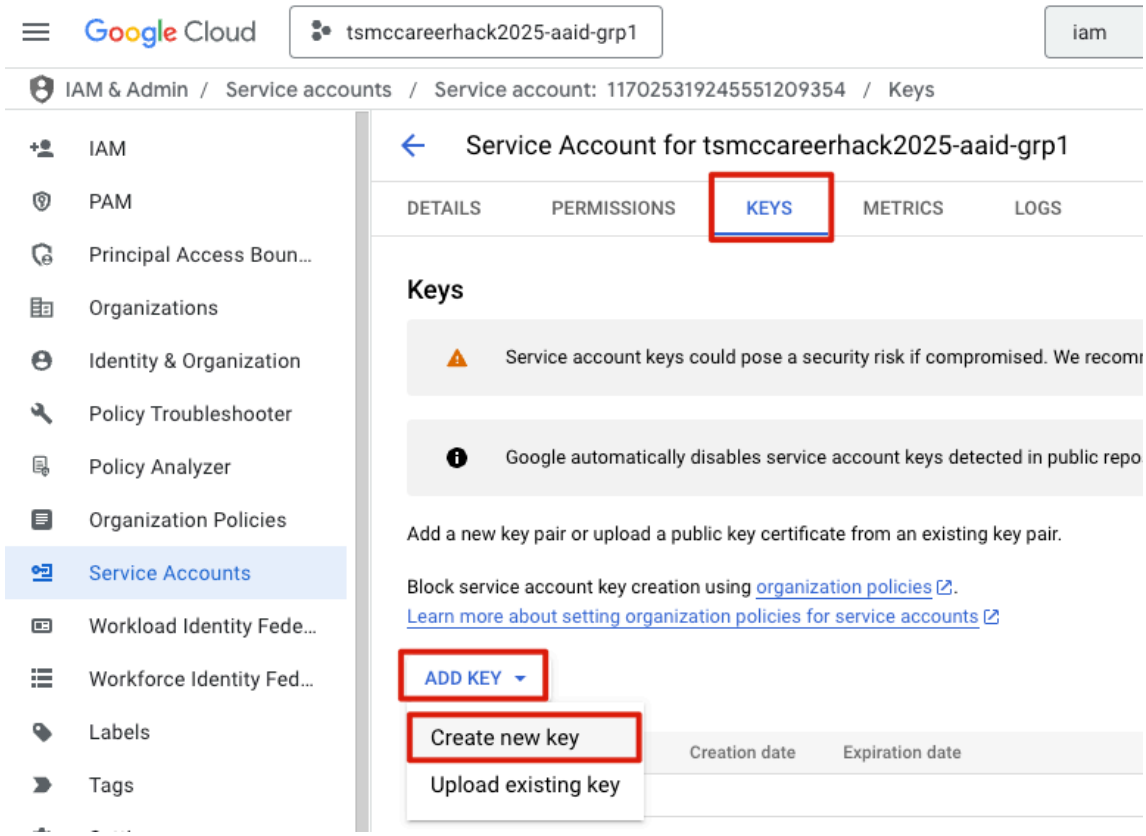
若有使用 service account key 的需求，請參考以下步驟
請勿將 key 上傳至公開 repository
進入 IAM 頁面



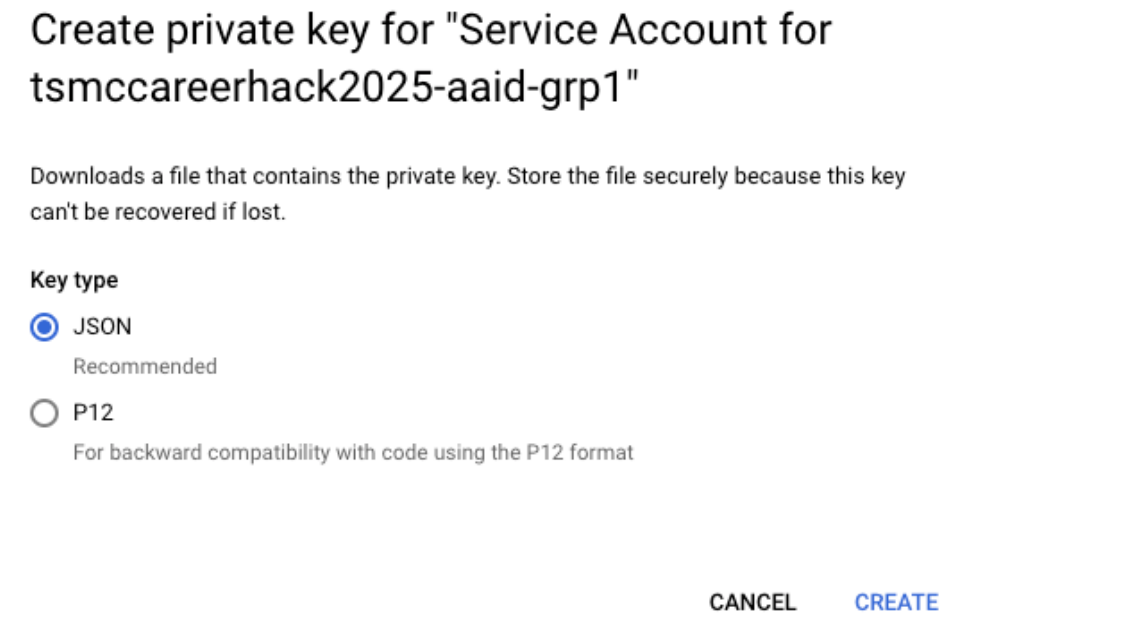
點擊 Service Accounts，選擇小組的 service account:
careerhack2025-group-sa@tsmccareerhack2025-icsd-grp#.iam.gserviceaccount.com



點擊 KEYS，按下 ADD KEY → Create new key 以下載 key 的 json 檔。
請勿將 key 上傳至公開 repository



請勿將 key 上傳至公開 repository

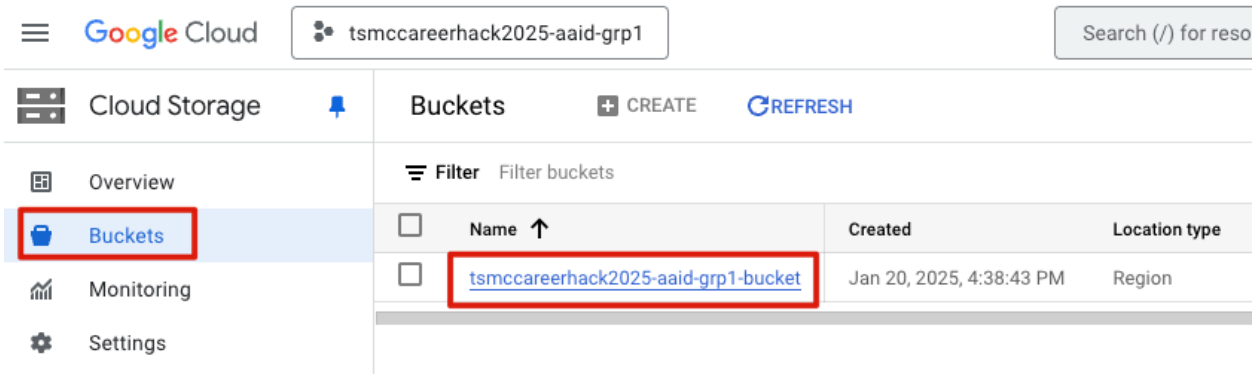


7. Cloud Storage

Bucket name:

- 競賽題目的資源下載：careerhack2025-icsd-resource-bucket
  - 此 bucket 位於 tsmccareerhack2025-icsd-test project 中，學生無法從 UI 存取，需要下指令
- 供參賽者存放資料：tsmccareerhack2025-icsd-grp#-bucket

UI 操作：



指令操作：

- 下列操作指令可在上述 Cloud Shell 和 GCE VM 環境執行以存取 Google Cloud Storage 的 bucket。
- 基本操作指令：
  - gcloud storage ls
  - gcloud storage cp
  - gcloud storage mv
  - gcloud storage rm
  - …

  完整指令操作手冊：https://cloud.google.com/sdk/gcloud/reference/storage
- 常用指令範例：
  - 列出目錄中的物件：

```
Unset
gcloud storage ls gs://BUCKET_NAME
```

  - 下載檔案到當前本機目錄：

```
Unset
gcloud storage cp gs://BUCKET_NAME/OBJECT_NAME .
```

- 下載資料夾到當前本機目錄：

```
Unset
gcloud storage cp -r gs://BUCKET_NAME/FOLDER_NAME .
```

- 上傳檔案：

```
Unset
gcloud storage cp LOCAL_FILE_NAME gs://BUCKET_NAME
```

- 上傳資料夾：

```
Unset
gcloud storage cp -r LOCAL_DIR_NAME gs://BUCKET_NAME
```

8. Gemini API

請留意各 model 的 rate limit

Token per minute

Base model	Tokens per minute
base_model: gemini-1.5-flash (version 001)	4M (4,000,000)
base_model: gemini-1.5-pro (version 001)	4M (4,000,000)

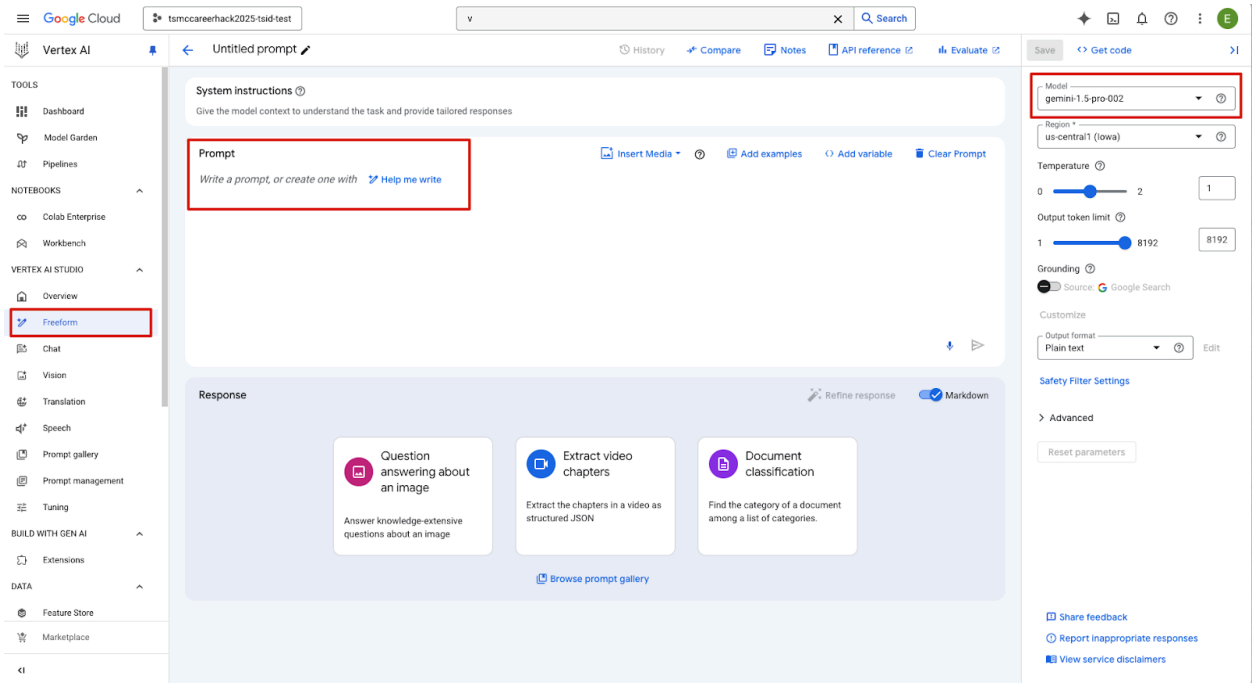
Request per minute

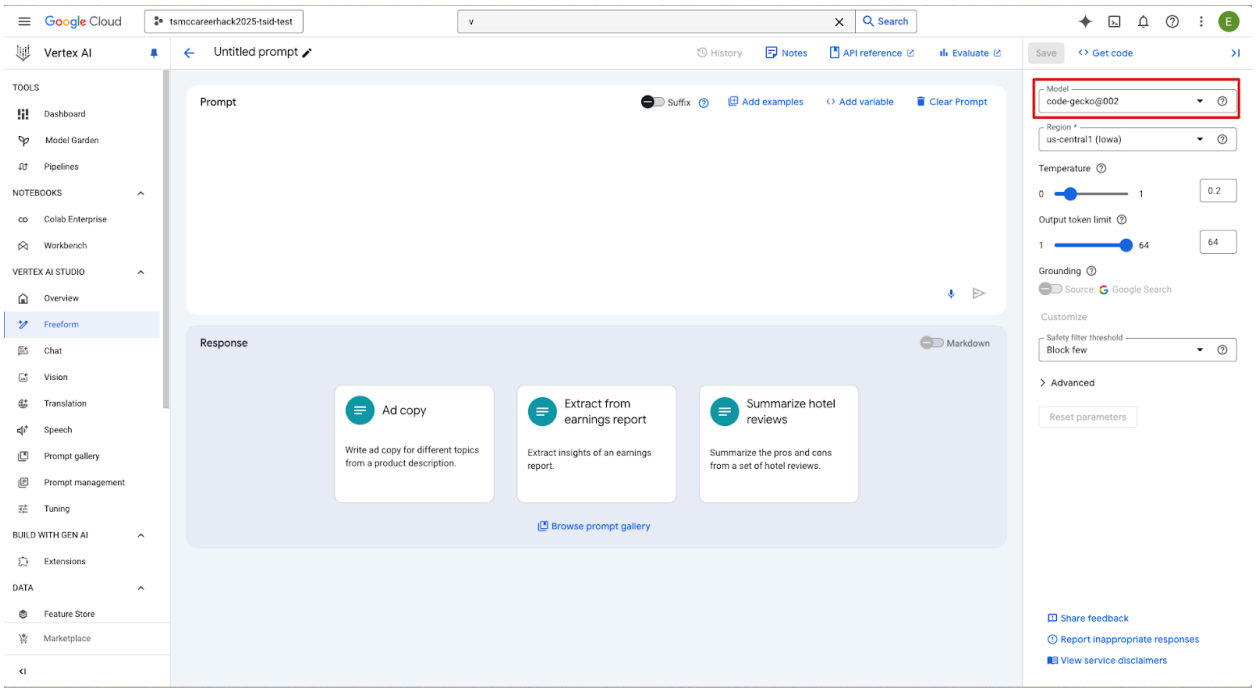
Base model	Requests per minute
base_model : gemini-1.5-flash	200
base_model : gemini-1.5-pro	60

請參考：https://cloud.google.com/vertex-ai/generative-ai/docs/quotas

- Vertex AI Studio UI 介面

Gemini 1.5 Pro for Text





- curl
- 若使用 Cloud Shell 或所提供的虛擬主機操作，請跳過此步驟。若欲從使用者本地機器操作，請先下載並安裝 **Google Cloud SDK**。安裝完成後需要執行下列指令登入帳號取得授權：

```
Unset
gcloud auth application-default login
```

curl 指令範例：

- 範例 request.json：

```
Unset
{
  "contents": [{
    "role": "user",
    "parts": [{
      "text": "Tell me how to win a hackathon"
    }]
  }]
}
```

- 範例 input：

```
Unset
export PROJECT_ID=YOUR_PROJECT_ID
export MODEL_ID="gemini-1.5-pro-002"

curl \
  -X POST \
  -H "Authorization: Bearer $(gcloud auth application-default
print-access-token)" \
  -H "Content-Type: application/json" \
  -d @request.json \

https://us-central1-aiplatform.googleapis.com/v1/projects/${PROJECT_ID}/locatio
ns/us-central1/publishers/google/models/${MODEL_ID}:generateContent
```

- 範例輸出（注意：usageMetadata 回傳了該 prompt 的 input 和 output token count）：

Unset


```
{
  "candidates": [
    {
      "content": {
        "role": "model",
        "parts": [
          {
            "text": "Winning a hackathon isn't just about writing the most technically brilliant code, it's about creating a well-rounded project that solves a problem and presents well.  Here's a breakdown of how to increase your chances:\n\n**Before the Hackathon:**\n\n* **Team Up (Optional but Recommended):**  A diverse team with complementary skills (coding, design, business, marketing) is a huge advantage.  Find teammates beforehand if possible.\n* **Brainstorm Ideas:** Don't wait until the hackathon starts.  Having a few project ideas in your back pocket saves valuable time.  Consider current events, trending technologies, and personal passions.\n* **Practice:** Familiarize yourself with the technologies you plan to use.  The more fluent you are, the more time you can spend on building, not learning.\n* **Gather Resources:**  Identify APIs, libraries, and frameworks you might need.  Knowing where to find resources can save you hours of searching during the event.\n* **Pack Smart:**  Charger, laptop stand, external monitor (if allowed), headphones, snacks, water bottle – the essentials for a long haul.\n\n**During the Hackathon:**\n\n* **Scope Realistically:**  Ambition is good, but a finished, functional project is better than an unfinished masterpiece.  Focus on a core feature and expand if time allows.\n* **Divide and Conquer:**  If you're in a team, leverage everyone's strengths.  Clear roles and responsibilities prevent overlap and maximize efficiency.\n* **Version Control (Git):**  Essential for team projects.  Use branches and merge requests to keep the codebase organized and avoid conflicts.\n* **Focus on the User:**  Keep the end-user in mind throughout the process.  A user-friendly interface and intuitive design are crucial.\n* **Seek Feedback:**  Talk to mentors and other participants.  External perspectives can highlight blind spots and spark new ideas.\n* **Document Everything:**  Keep track of your progress, decisions, and challenges.  This will be invaluable for your presentation and future development.\n* **Don't Neglect the Presentation:**  A compelling presentation can make or break your project.  Practice your pitch and highlight the problem, solution, and impact of your creation.\n* **Rest (Seriously):**  Pulling an all-nighter might seem productive, but fatigue leads to errors and poor decision-making.  Short breaks and power naps can boost your performance.\n\n**Key Ingredients for a Winning Project:**\n\n* **Originality/Innovation:**  A fresh perspective or a novel application of existing technology.\n* **Impact/Usefulness:**  Addresses a real-world problem
```

```
or provides a valuable service.\n* **Technical Execution:**  Clean,
well-documented code that functions as intended.\n* **Design/User Experience:**
An intuitive and aesthetically pleasing interface.\n* **Presentation:**  A
clear, concise, and engaging pitch that highlights the project's
strengths.\n\n**After the Hackathon:**\n\n* **Network:**  Connect with other
participants, judges, and sponsors.  Hackathons are excellent networking
opportunities.\n* **Learn from Feedback:**  Even if you don't win, gather
feedback and use it to improve your skills for future events.\n* **Iterate and
Improve:**  Don't abandon your project after the hackathon.  Continue
developing and refining it – you might have a viable product on your
hands.\n\n**Common Pitfalls to Avoid:**\n\n* **Feature Creep:**  Trying to
implement too many features and ending up with nothing finished.\n* **Ignoring
the User:**  Building something technically impressive but impractical or
difficult to use.\n* **Poor Presentation:**  Failing to communicate the value
and impact of your project effectively.\n* **Lack of Teamwork:**  Inefficient
collaboration and communication within the team.\n\n\nWinning a hackathon is a
combination of skill, preparation, and a bit of luck.  By focusing on these
tips, you'll significantly increase your chances of success and have a
rewarding experience.  Good luck!\n"
          }
        ]
      },
      "finishReason": "STOP",
      "avgLogprobs": -0.20640207265878652
    }
  ],
  "usageMetadata": {
    "promptTokenCount": 8,
    "candidatesTokenCount": 847,
    "totalTokenCount": 855
  },
  "modelVersion": "gemini-1.5-pro-002"
}
```


更多範例請參考：[Generate content with the Gemini Enterprise API | Generative AI on Vertex AI | Google Cloud]

- Python SDK
- Python 版本需求：Python 3.8 or higher
- 安裝 Vertex AI Python SDK：

Mac/Linux:

```
Unset
pip install virtualenv
virtualenv <your-env>
source <your-env>/bin/activate
<your-env>/bin/pip install google-cloud-aiplatform
```

Windows:

```
Unset
pip install virtualenv
virtualenv <your-env>
<your-env>\Scripts\activate
<your-env>\Scripts\pip.exe install google-cloud-aiplatform
```

更多資訊請參考 Vertex AI Python SDK.
Python client library | Google Cloud
Install the Vertex AI client libraries | Google Cloud

- 本地開發登入授權，本地開發需先在本地安裝 gcloud CLI：

```
Unset
gcloud auth application-default set-quota-project <PROJECT_ID>
gcloud auth application-default login
```

- 範例程式：

```
Python
import vertexai
from vertexai.generative_models import GenerativeModel

# TODO(developer): Update below line
PROJECT_ID = "PROJECT_ID"

REGION = "us-central1"
vertexai.init(project=PROJECT_ID, location=REGION)

model = GenerativeModel("gemini-1.5-pro-002")

response = model.generate_content(
```

```
    "Tell me how to win a hackathon"
)

print(response.text)
```

9. Vertex AI Embedding API model

text-multilingual-embedding-002
Text embeddings API | Generative AI on Vertex AI | Google Cloud

Embeddings for Text | Model Garden

- curl

  curl 指令範例：

- 範例 request.json：

```
Unset
{
  "instances": [
    {
      "task_type": "RETRIEVAL_DOCUMENT",
      "title": "document title",
      "content": "I would like embeddings for this text!"
    }
  ],
  "parameters": {
    "outputDimensionality": 256
  }
}
```

- 範例 input：

```
Unset
MODEL_ID="text-multilingual-embedding-002"
PROJECT_ID=PROJECT_ID


curl \
-X POST \
-H "Authorization: Bearer $(gcloud auth print-access-token)" \
-H "Content-Type: application/json" \
-d @request.json \
https://us-central1-aiplatform.googleapis.com/v1/projects/${PROJECT_ID}/locatio
ns/us-central1/publishers/google/models/${MODEL_ID}:predict
```

- 範例輸出：

```
Unset
{
  "predictions": [
    {
      "embeddings": {
        "statistics": {
          "token_count": 14,
          "truncated": false
        },
        "values": [
          0.039897624403238297,
          -0.011670297011733055,
          0.03811752051115036,
          0.024373488500714302,
          0.065585844218730927,
          0.013257214799523354,
          0.0053430162370204926,
          0.041858211159706116,
          ...
          ...
        ]
      }
    }
  ],
  "metadata": {
    "billableCharacterCount": 45
```

```
  }
}
```

10. Speech-to-Text V2 API

詳細步驟請參考官方文件： Transcribe speech to text by using the command line

- 若使用 Cloud Shell 或所提供的虛擬主機操作，請跳過此步驟。若欲從使用者本地機器操作，請先下載並安裝 **Google Cloud SDK**。安裝完成後需要執行下列指令登入帳號取得授權：

```
Unset
gcloud auth application-default login
```

- Curl
  - 修改檔案路徑 `/full/path/to/audio/file.wav`

```
Unset
echo "{
  \"config\": {
    \"auto_decoding_config\": {},
    \"language_codes\": [\"en-US\"],
    \"model\": \"long\"
  },
  \"content\": \"$(base64 -w 0 /full/path/to/audio/file.wav | sed 's/+/-/g;
s/\//_/g')\"
}" > /tmp/data.txt
```

  - 範例 input：

```
Unset
PROJECT_ID=PROJECT_ID

curl -X POST -H "Content-Type: application/json; charset=utf-8" \
    -H "Authorization: Bearer $(gcloud auth print-access-token)" \
    -d @/tmp/data.txt \
```

```
https://speech.googleapis.com/v2/projects/${PROJECT_ID}/locations/global/recognizers/_:recognize
```

- 範例輸出：

```
Unset
{
    "results": [
      {
        "alternatives": [
          {
            "transcript": "how old is the Brooklyn Bridge",
            "confidence": 0.98267895
          }
        ]
      }
    ]
  }
```

- Python SDK
  - 請參考前一章節安裝 Vertex AI Python SDK

```python
Python
import os

from google.cloud.speech_v2 import SpeechClient
from google.cloud.speech_v2.types import cloud_speech

PROJECT_ID = os.getenv("GOOGLE_CLOUD_PROJECT")


def quickstart_v2(audio_file: str) -> cloud_speech.RecognizeResponse:
    """Transcribe an audio file.
    Args:
        audio_file (str): Path to the local audio file to be transcribed.
```

```
    Returns:
        cloud_speech.RecognizeResponse: The response from the recognize
request, containing
        the transcription results
    """
    # Reads a file as bytes
    with open(audio_file, "rb") as f:
        audio_content = f.read()

    # Instantiates a client
    client = SpeechClient()

    config = cloud_speech.RecognitionConfig(
        auto_decoding_config=cloud_speech.AutoDetectDecodingConfig(),
        language_codes=["en-US"],
        model="long",
    )

    request = cloud_speech.RecognizeRequest(
        recognizer=f"projects/{PROJECT_ID}/locations/global/recognizers/_",
        config=config,
        content=audio_content,
    )

    # Transcribes the audio into text
    response = client.recognize(request=request)

    for result in response.results:
        print(f"Transcript: {result.alternatives[0].transcript}")

    return response
```

11. Artifact Registry

- 本活動為每一組準備好一個 private docker repository，可用於存放 docker images：tsmccareerhack2025-icsd-grp#-repository

- 使用說明：
    - 上傳 docker image：
        - 授權：於 Cloud Shell 或 GCE VM 執行下列指令

```
Unset
sudo gcloud auth configure-docker us-central1-docker.pkg.dev
```

- tag 本地 image

```
Unset
sudo docker tag SOURCE-IMAGE
us-central1-docker.pkg.dev/PROJECT-ID/REPOSITORY/IMAGE:TAG
```

- 將 image 推到 Artifact Registry：

```
Unset
sudo docker push us-central1-docker.pkg.dev/PROJECT-ID/REPOSITORY/IMAGE:TAG
```

更多說明請參考官方文件：
https://cloud.google.com/artifact-registry/docs/docker/pushing-and-pulling#push-tagged

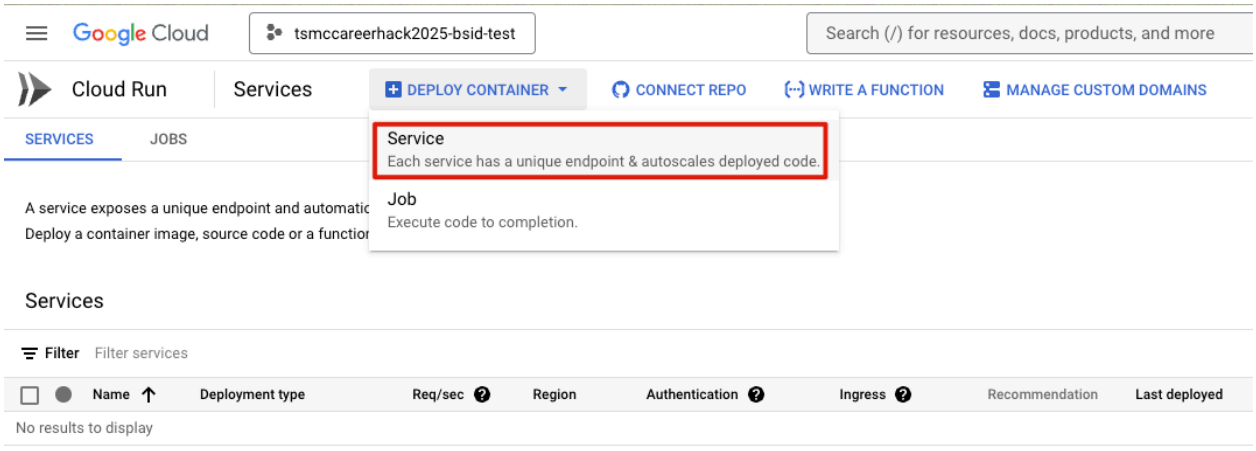
- 從 Artifact Registry 拉 image 到本機 docker：

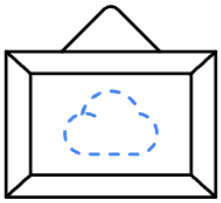
```
Unset
sudo docker pull
us-central1-docker.pkg.dev/PROJECT-ID/REPOSITORY/IMAGE@IMAGE-DIGEST
```

更多說明請參考官方文件：
https://cloud.google.com/artifact-registry/docs/docker/pushing-and-pulling#pulling

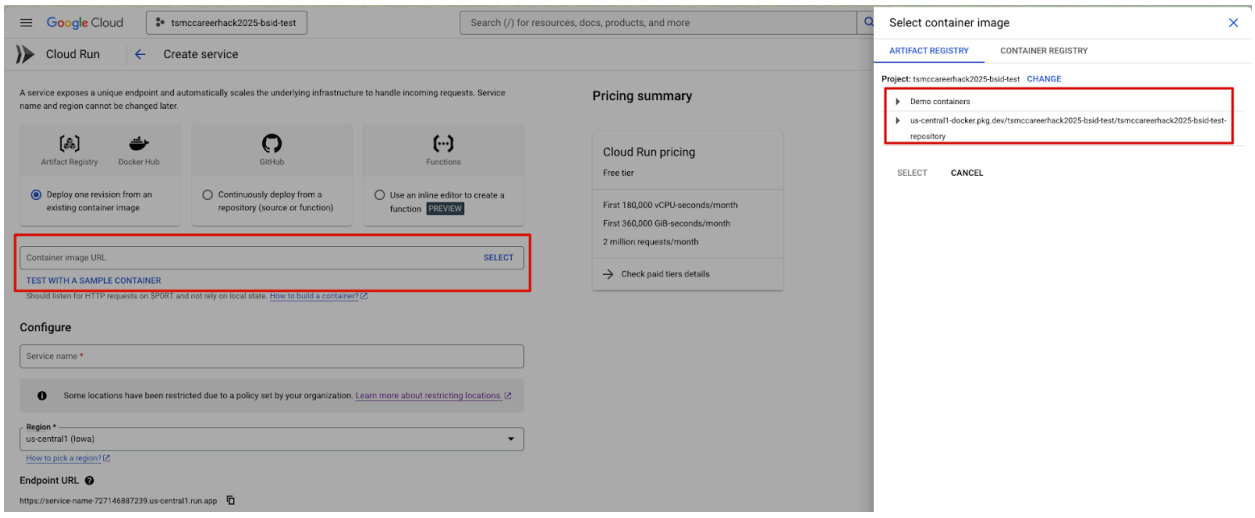
12. Cloud Run

- Cloud Run 是一個完全託管的運算環境，用於部署和擴展無伺服器 HTTP 容器。
- 從選單或搜尋找到 Cloud Run，點擊 CREATE SERVICE 創建一個 Cloud Run 服務。

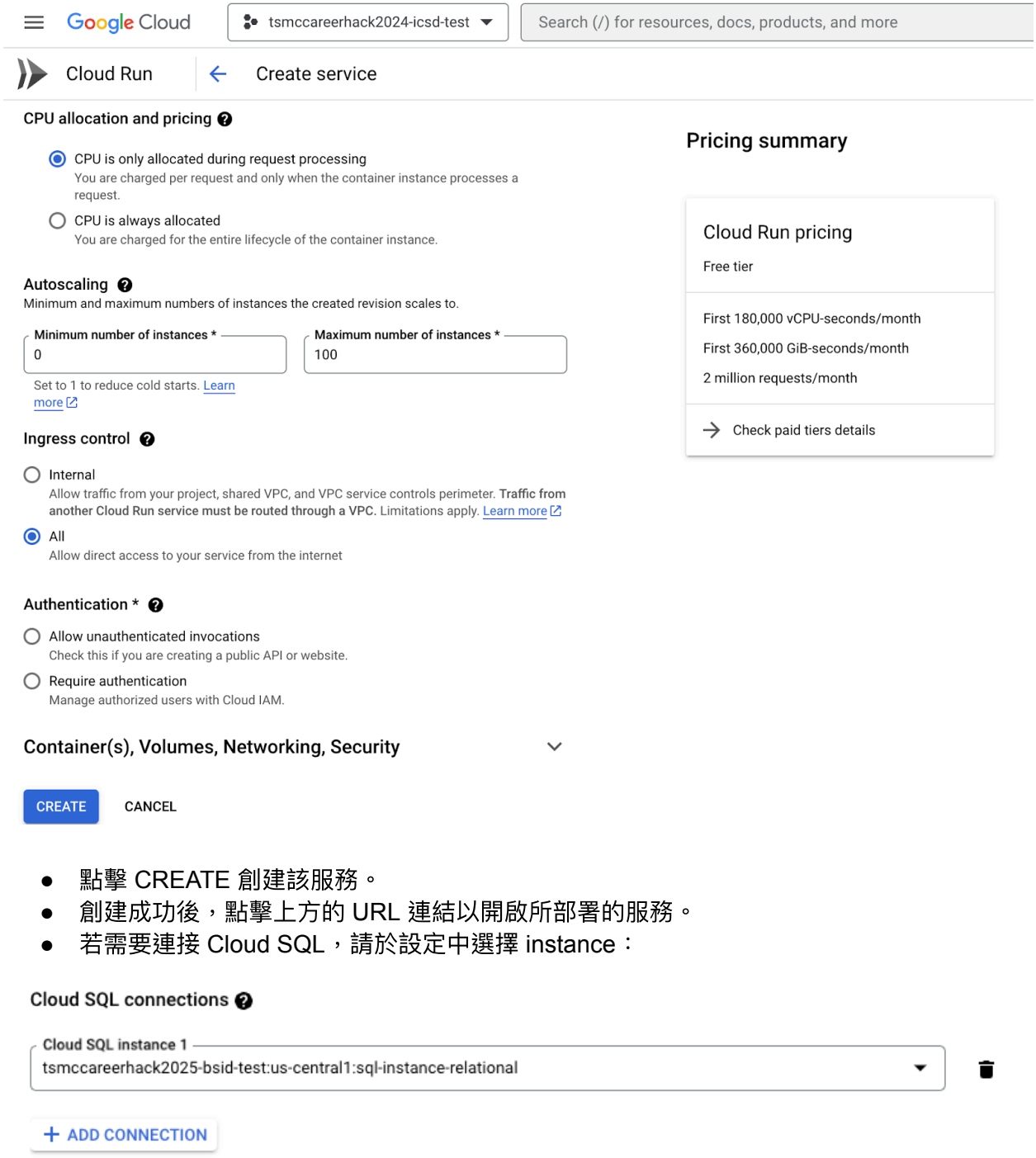




- 選擇 container image：
  - 快速選用範例 container image：點擊 TEST WITH A SAMPLE CONTAINER
  - 選用自備的 container image：點擊 SELECT，從 Artifact Registry 中選擇欲使用的 image。



- 其他設定：根據實際需求設定參數：

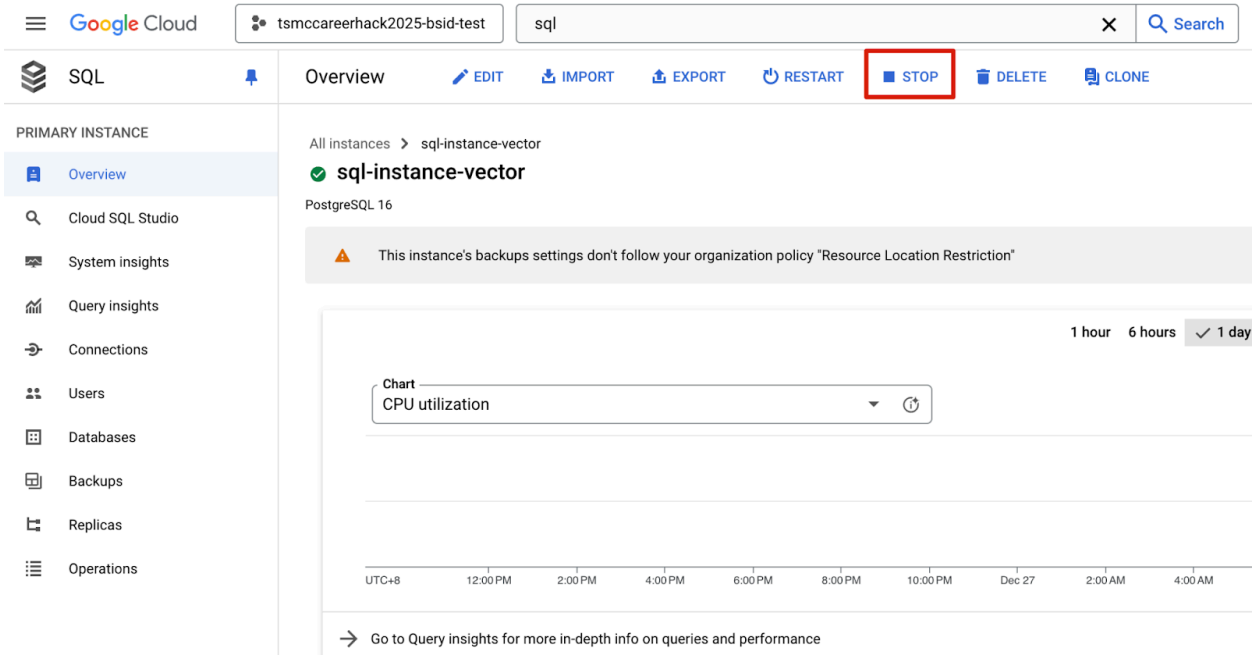


- 點擊 CREATE 創建該服務。
- 創建成功後，點擊上方的 URL 連結以開啟所部署的服務。
- 若需要連接 Cloud SQL，請於設定中選擇 instance：

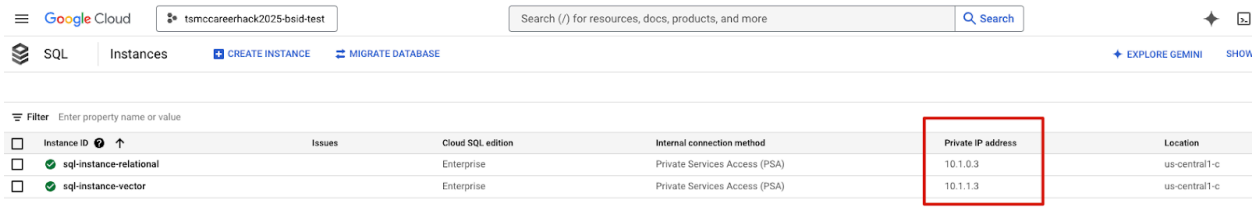
詳細設定請參考官方文件：[Quickstart: Connect to Cloud SQL for PostgreSQL from Cloud Run | Google Cloud]

13. Cloud SQL

- Vector DB : sql-instance-vector
- 點擊 Stop 關機（**請記得不使用時要關機！**）：



- 連線方式：
  - 查看 instance 的 private IP



  - 從提供的 GCE VM 中使用 postgres client

```
@gce-instance:~$ psql -h 10.1.1.3 -U postgres
Password for user postgres:
psql (15.10 (Debian 15.10-0+deb12u1), server 16.6)
WARNING: psql major version 15, server major version 16.
         Some psql features might not work.
SSL connection (protocol: TLSv1.3, cipher: TLS_AES_256_GCM_SHA384, compression: off)
Type "help" for help.

postgres=>
```

- Vector DB 使用方式請參考官方文件：[Work with vector embeddings | Cloud SQL for PostgreSQL]

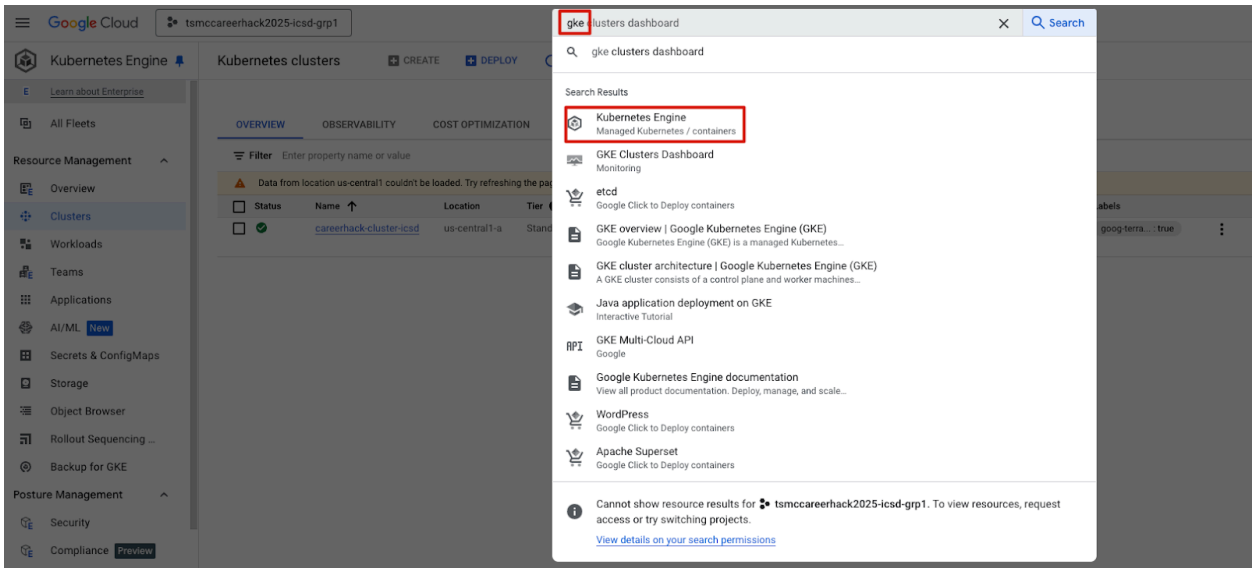
- 使用 psql 連接 DB 後，需建立 EXTENSION

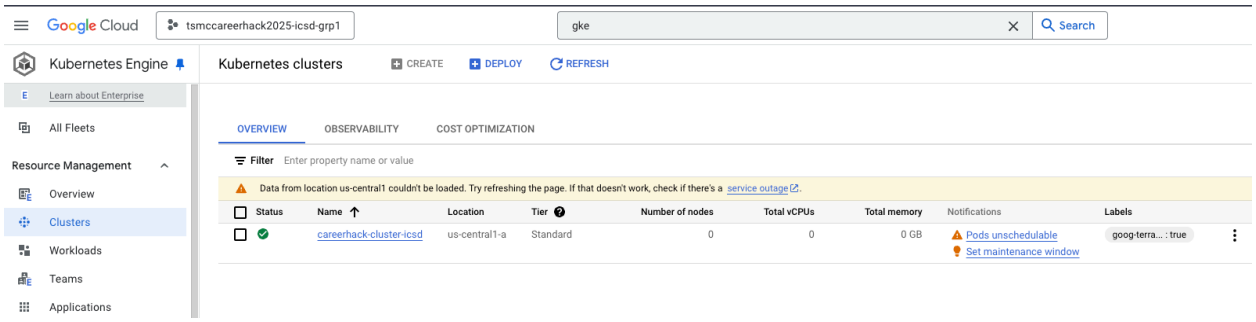
```
Unset
\c DB_NAME

GRANT EXECUTE ON FUNCTION embedding TO USER_NAME;
```

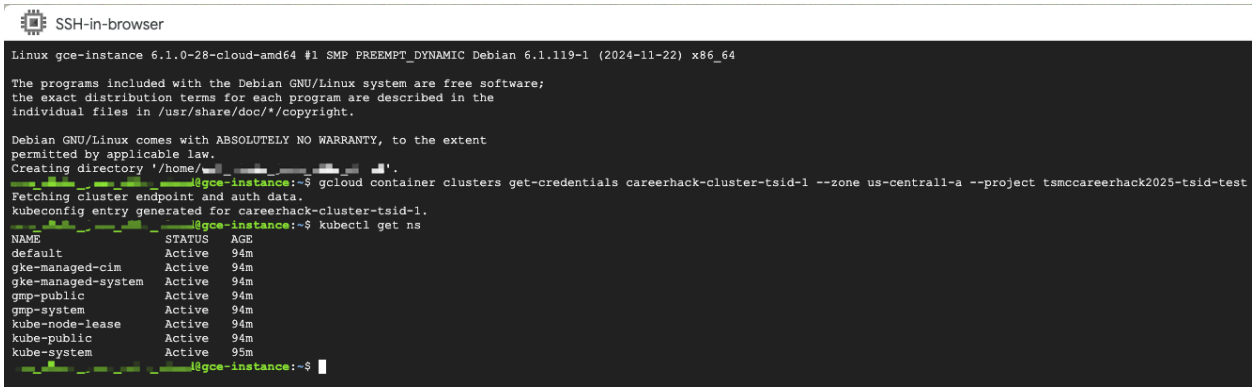
14. GKE

- Cluster: careerhack-cluster-icsd





連線方式：
需於 GCE VM 中操作 (請參考前面章節連線至 GCE VM)



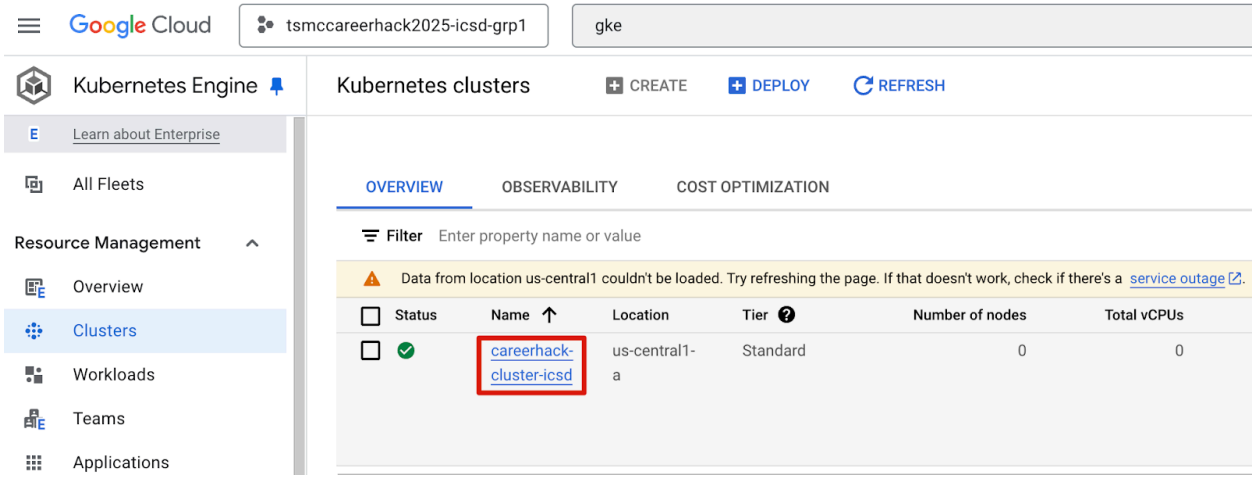
取得 gke credentials

```
Unset
gcloud container clusters get-credentials careerhack-cluster-icsd --region
us-central1 --project tsmccareerhack2025-icsd-test
```

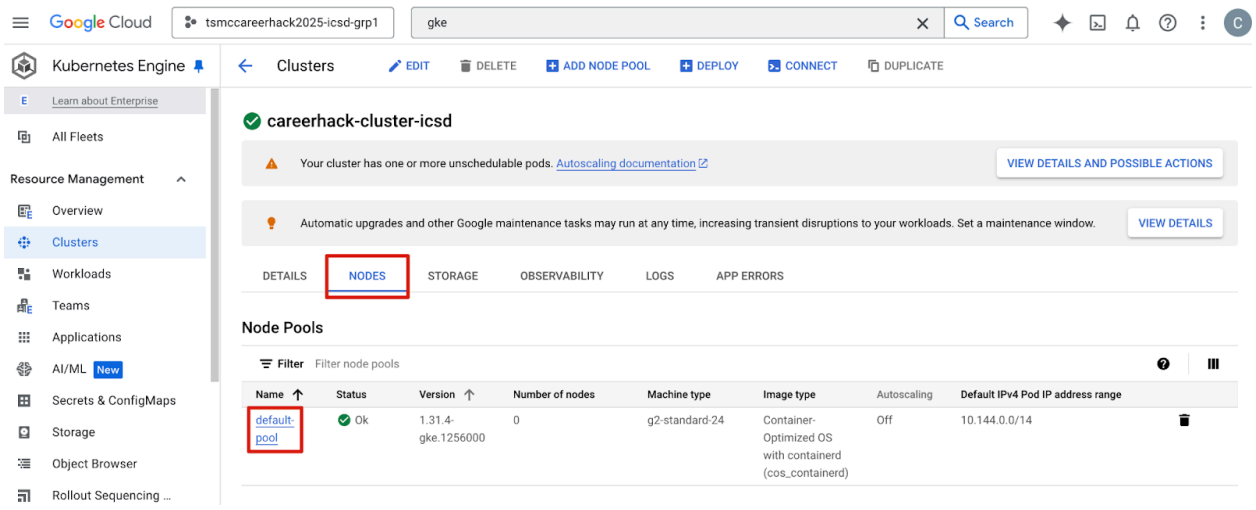
取得 credentials 後可使用 kubectl 操作 cluster (請參考上圖範例)

Resize Node Pool:
開機 - 將 worker node resize to 1
關機 - 將 worker node resize to 0（**請記得不使用時要關機！**）

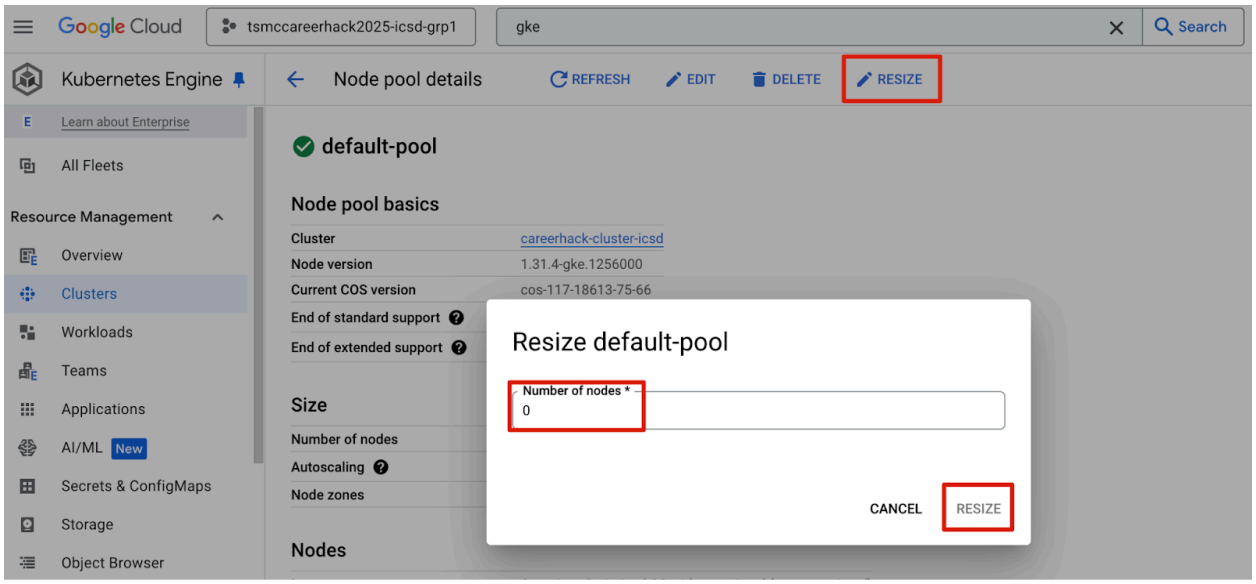
點擊 cluster: careerhack-cluster-tsid



點擊 NODES → Node Pools → default-pool

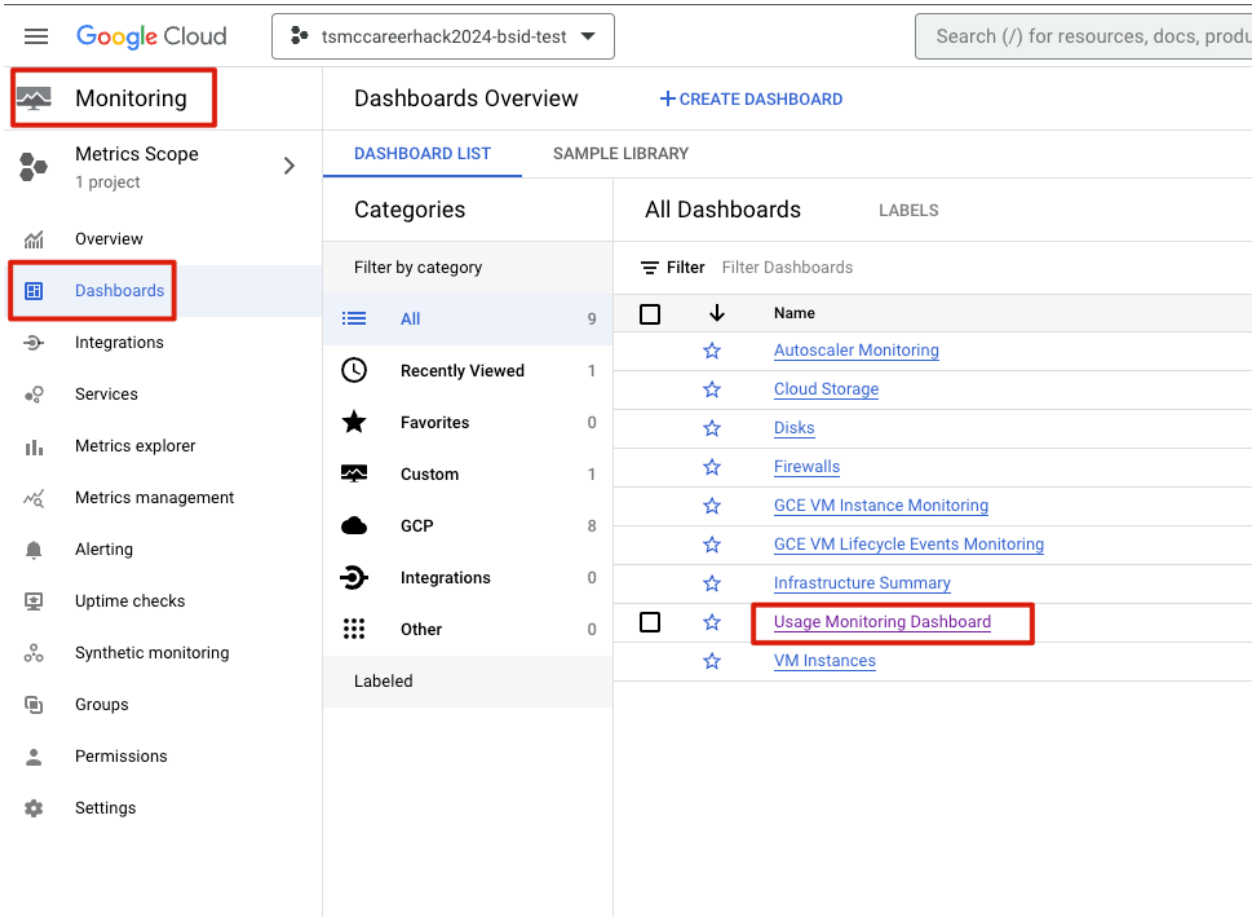


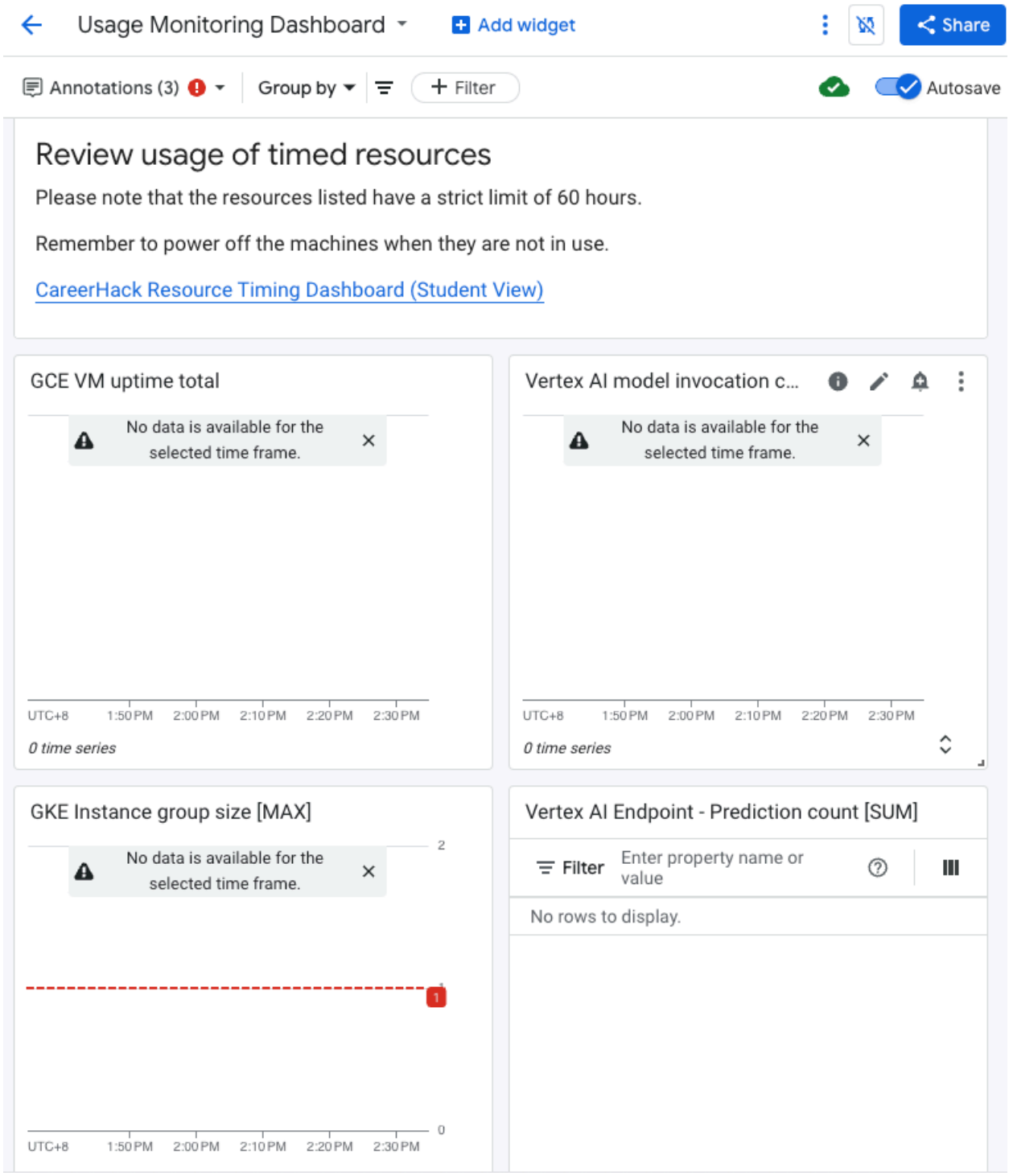
開機 - 將 worker node resize to 1
關機 - 將 worker node resize to 0（**請記得不使用時要關機！**）



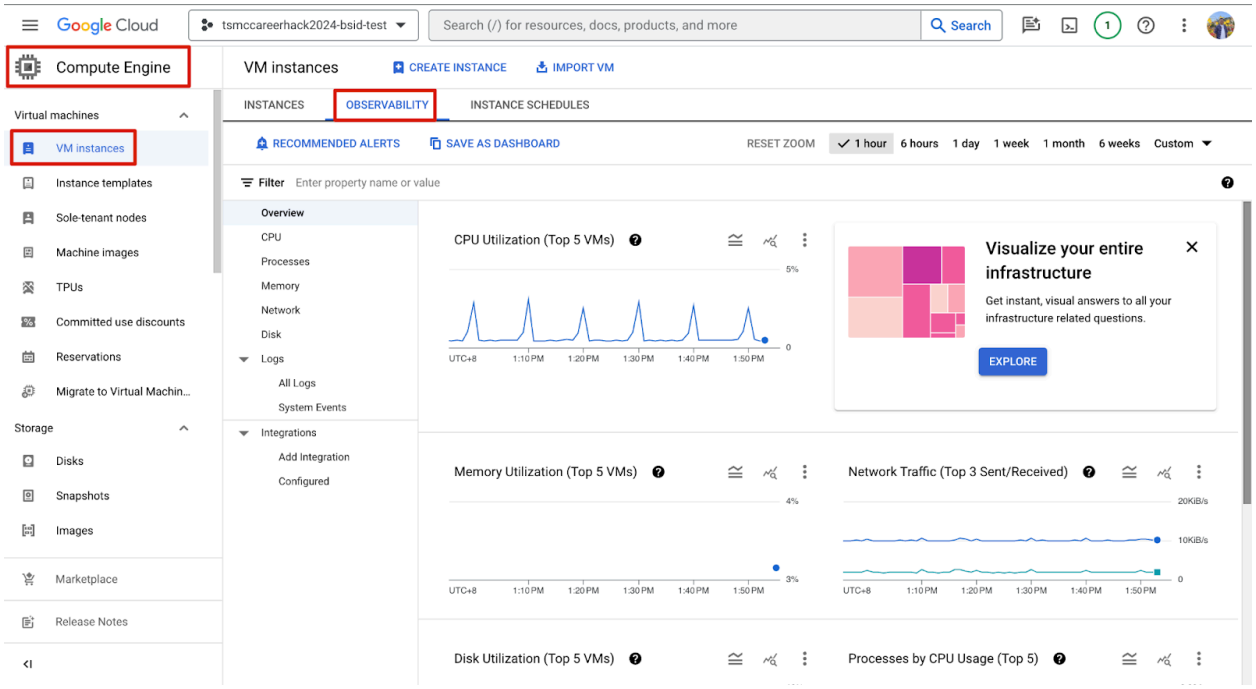
15. Monitoring

- 進入 Monitoring 服務的 Dashboards，打開 Usage Monitoring Dashboard 查看 GCE VM 的使用時數以及 gemini 使用次數。



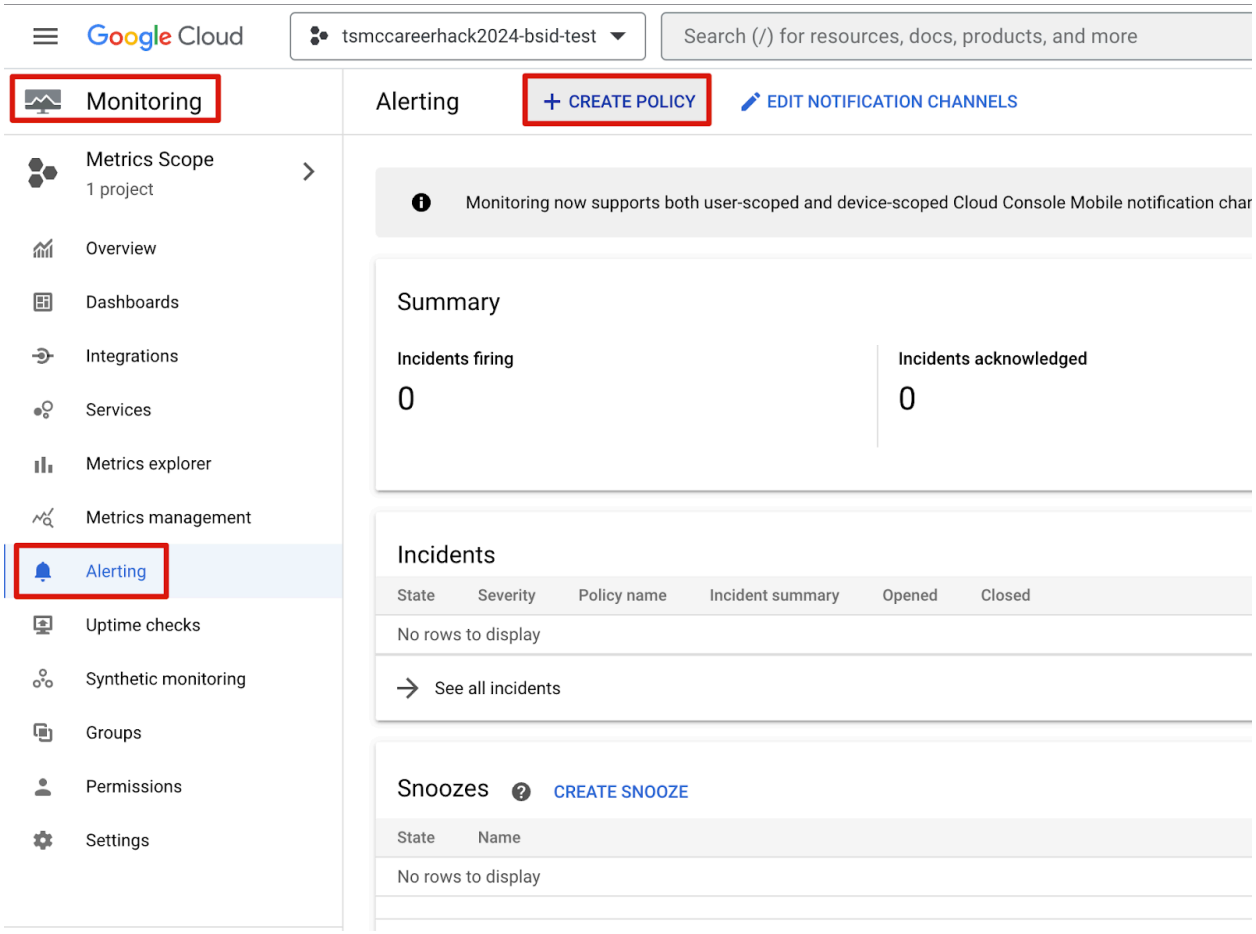

Usage Monitoring Dashboard
Add widget
Share
Annotations (3)
Group by
Filter
Autosave
Review usage of timed resources
Please note that the resources listed have a strict limit of 60 hours.
Remember to power off the machines when they are not in use.
CareerHack Resource Timing Dashboard (Student View)
GCE VM uptime total
No data is available for the selected time frame.
UTC+8
1:50 PM
2:00 PM
2:10 PM
2:20 PM
2:30 PM
0 time series
Vertex AI model invocation c...
No data is available for the selected time frame.
UTC+8
1:50 PM
2:00 PM
2:10 PM
2:20 PM
2:30 PM
0 time series
GKE Instance group size [MAX]
No data is available for the selected time frame.
2
1
0
UTC+8
1:50 PM
2:00 PM
2:10 PM
2:20 PM
2:30 PM
Vertex AI Endpoint - Prediction count [SUM]
Filter
Enter property name or value
No rows to display.

- 若欲查看 GCE VM 的 metrics (CPU utilization, memory utilization, disk operations…)，可透過在 Console 點擊進入 Compute Engine 的 OBSERVABILITY 頁面查看。

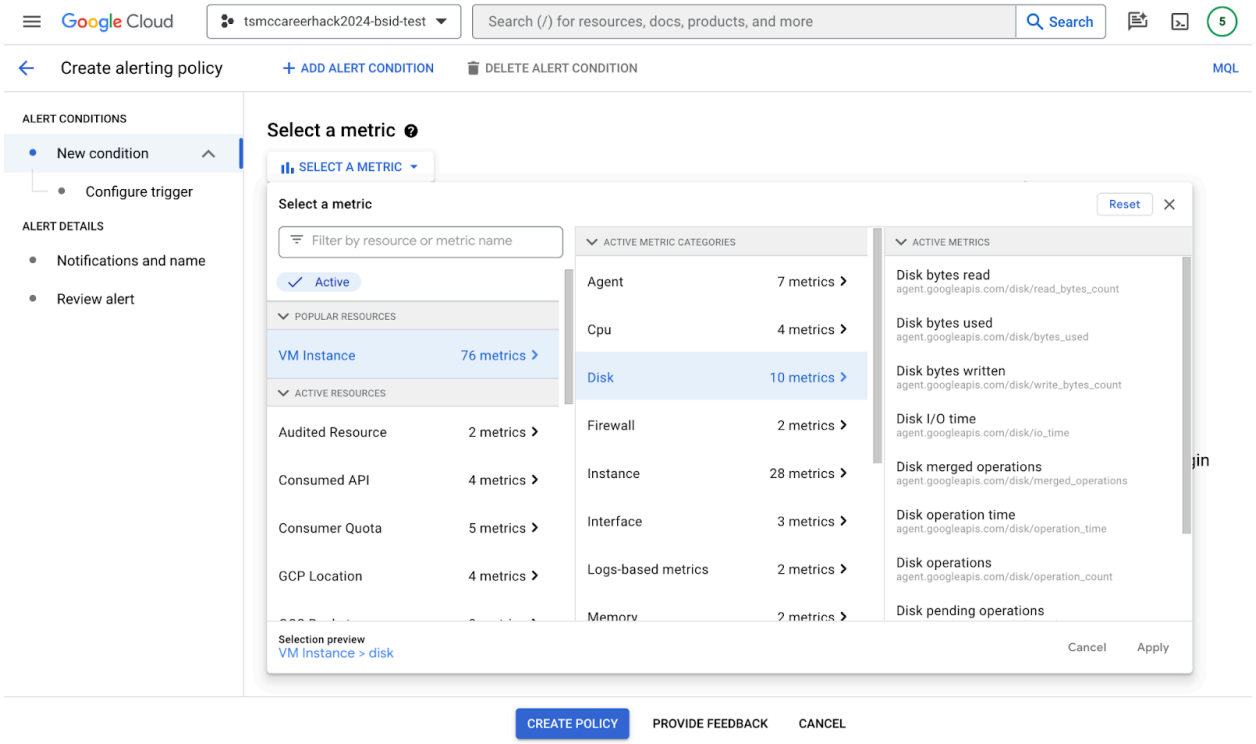


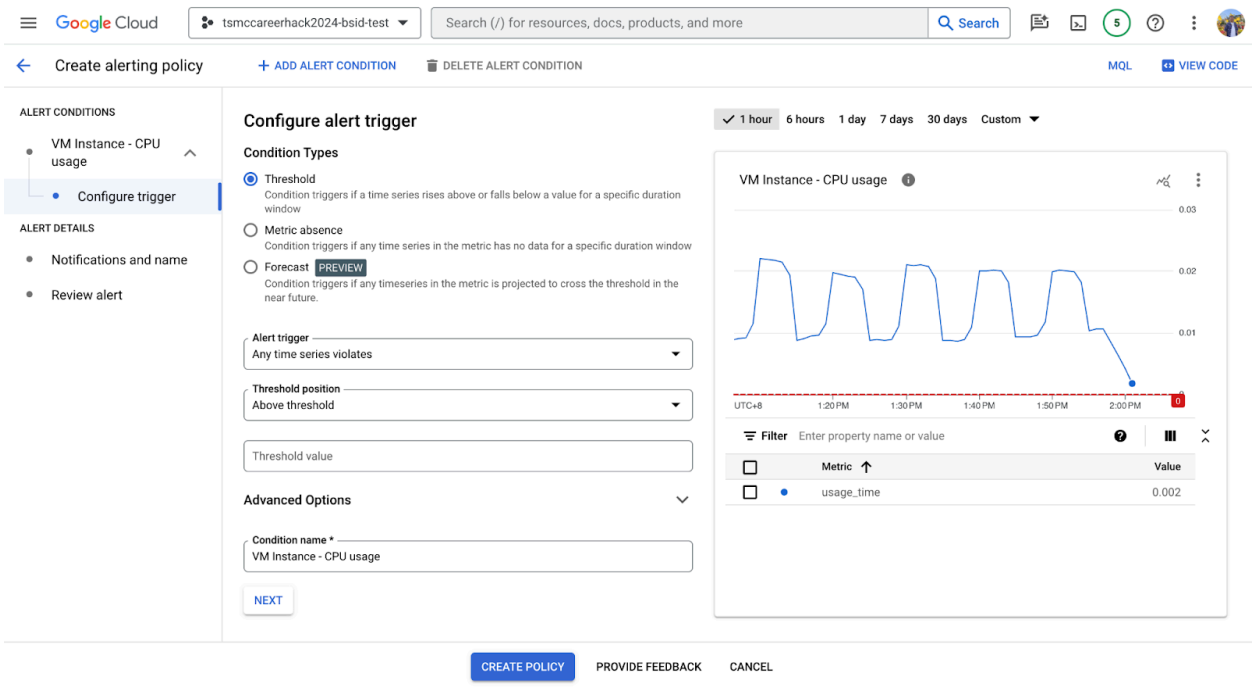
16. Alerting

- 若有需要獲取一些監控 metric 的告警 (alerts)，可在 Monitoring → Alerting 設定。

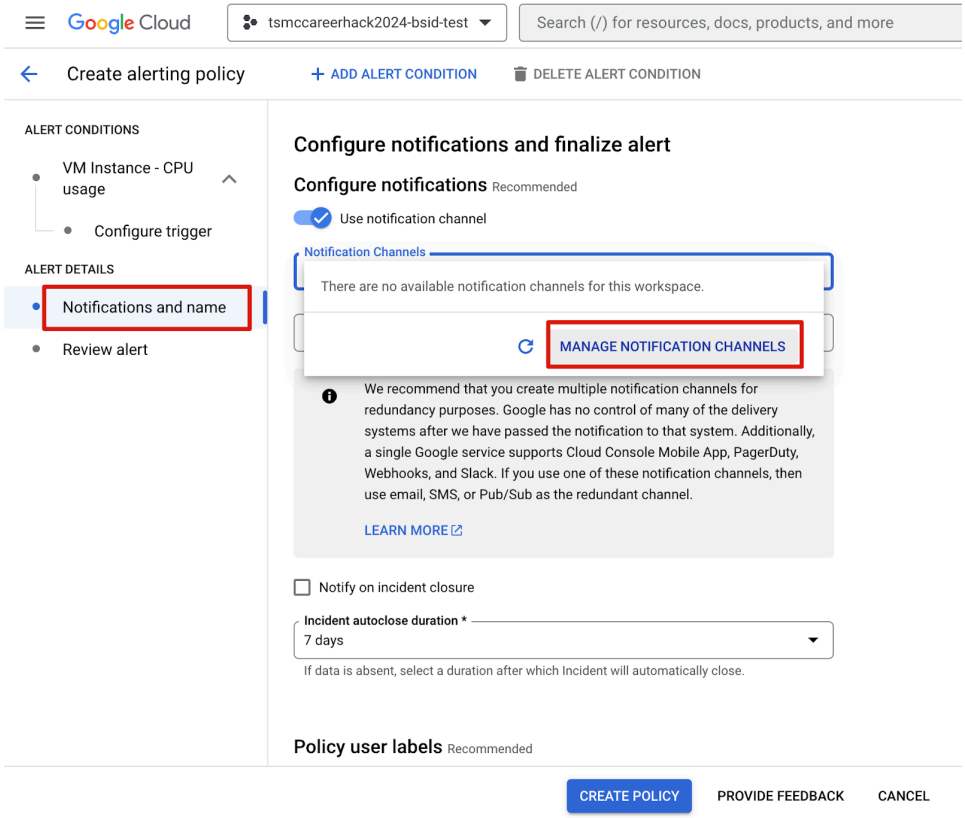


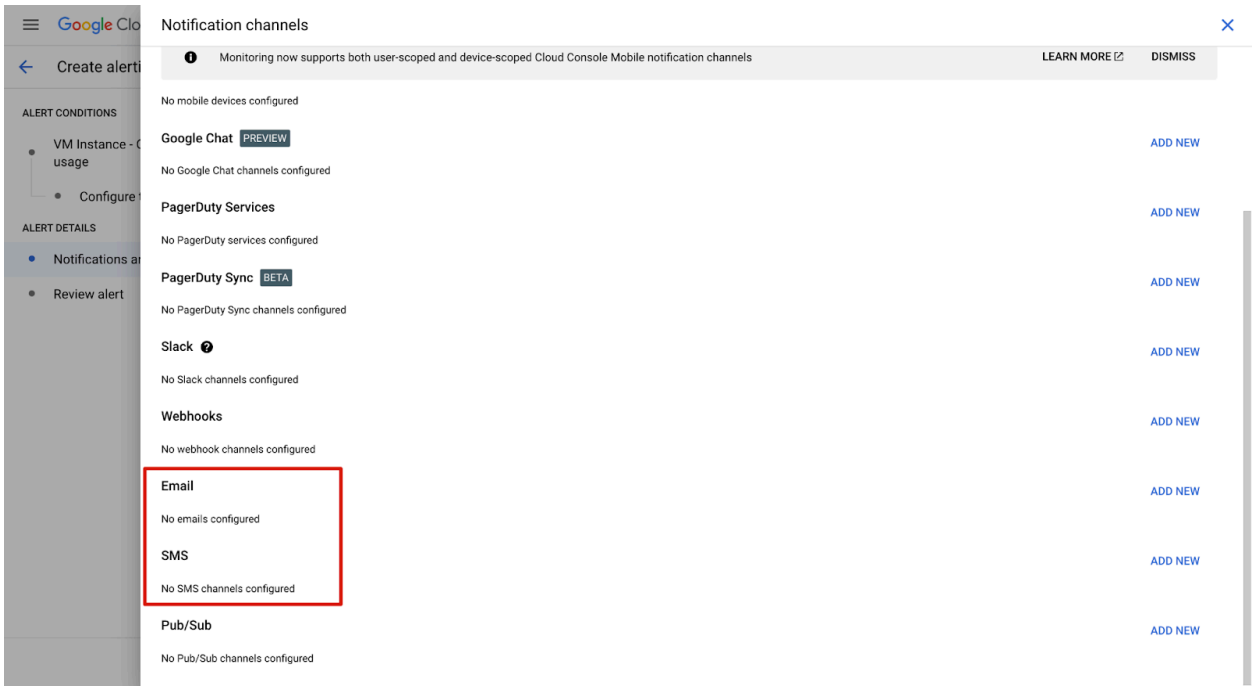
- 選定特定 metric 作為觸發告警的條件及其門檻值（請根據各自需求設定各參數，更多詳細說明請參考官方說明：https://cloud.google.com/monitoring/alerts/using-alerting-ui）





- 設定接收告警的管道





17. Review 後點擊 CREATE POLICY 創建該 alert policy。