

# MediaMarkt Hackathon - Challenge Solution

---

Indice:

- [MediaMarkt Hackathon - Challenge Solution](#)
  - [Requisitos](#)
  - [Registrar el Contenedor generado por el DockerFile con Cloud Build / Artifacts](#)
    - [Post-data](#)
  - [Generar un fichero YAML para Docker Compose](#)
  - [Generar los ficheros Terraform para tener la infraestructura como código y poder desplegar con Kubernetes](#)
  - [Extra: Desplegar con Terraform los recursos Kubernetes definidos en un fichero yaml](#)
  - [Pregunta sobre IAM](#)
    - [Respuesta:](#)

## Requisitos

Para realizar esta práctica he necesitado las siguiente hereramientas:

- docker
- gcloud CLI
- terraform
- vscode

Ya tenía instalada la mayoría de estas excepto gcloud CLI. Para instalarla he seguido los siguientes pasos:

1. He descargado el instalador de gcloud CLI desde la [página oficial](#).
2. He ejecutado el instalador en bash y he seleccionado la opción de instalar la CLI de gcloud.
3. He ejecutado el comando `gcloud init` para iniciar sesión en mi cuenta de Google Cloud.
4. He seleccionado el proyecto para esta práctica.

## Registrar el Contenedor generado por el DockerFile con Cloud Build / Artifacts

1. Primero, He realizado un análisis del [repositorio con la aplicación](#).
2. Me he asegurado de que el Dockerfile construido funciona desplegando en local con los siguiente comandos: `docker build -t foo .` `docker run -it -p 3000:3000 foo`

# Welcome to the Nuwe's vite vue skeleton

Here you would find a Vue bootstrapped webapp where you could start coding without taking care about configuring so many things, and just focus on the code!

You could start modifying the `src/App.vue` file to see any change in the webapp.

3. Luego, he creado un archivo de configuración de Cloud Build ([cloudbuild.yaml](#)) que especifica las acciones que deben realizarse para compilar y registrar el contenedor en el Registry.
4. He ejecutado el comando `gcloud builds submit --config cloudbuild.yaml` para enviar la compilación a Cloud Build.

Aquí tuve una incidencia ya que google no me permitía habilitar el servicio de Cloud Build. Parecía un problema con la facturación de la cuenta.

The screenshot shows the Google Cloud Billing 'Perfil de pagos' (Payment profile) page. The page is in Spanish and displays the following information:

- Perfil de pagos:** Temporal User, Persona física, ID de perfil de pagos: 9489-5024-4289.
- Información del cliente:** Tipo de cuenta: Persona física; Información fiscal: Estado fiscal: Personal.
- Accede a todos los productos de Cloud Platform:** Obtén todo lo que necesitas para compilar y ejecutar tus apps, sitios web y servicios, incluidos Firebase y la API de Google Maps.
- Obtén \$300 en crédito gratis:** Comienza a usar Google Cloud con \$300 de crédito para invertir en los próximos 90 días.
- Sin cargos automáticos cuando finaliza la prueba gratuita:** Te pedimos tu tarjeta de crédito para asegurarnos de que no eres un robot. No se te cobrará a menos que se actualice a una cuenta de pagos de forma manual.

An error message is displayed: 'Se produjo un error. No se pudo completar la acción. Vuelve a intentarlo más tarde. [OR-CBAT-15]'. A modal dialog box is open with the title 'Comunícate con nosotros para comenzar' (Contact us to get started). The modal text says: 'No pudimos establecer tu perfil de pagos. Para terminar de configurar tu prueba gratuita, solicita ayuda de nuestro equipo de Asistencia al cliente.' (We couldn't set up your payment profile. To finish configuring your free trial, request help from our Customer Support team.) There is a button labeled 'IR A ASISTENCIA' (Go to support).

Con este problema he perdido alrededor de 1 hora y no he tenido resultados positivos. Así que no he podido continuar con este punto.

```
Step #1 - "Push container image to Artifact Registry": d21707d42a97: Retrying in 10 seconds
Step #1 - "Push container image to Artifact Registry": b37eb609be8f: Retrying in 10 seconds
Step #1 - "Push container image to Artifact Registry": 146a7f289aca: Retrying in 10 seconds
Step #1 - "Push container image to Artifact Registry": d6de8055d22f: Retrying in 10 seconds
Step #1 - "Push container image to Artifact Registry": 07e43b9577c8: Retrying in 7 seconds
Step #1 - "Push container image to Artifact Registry": d21707d42a97: Retrying in 9 seconds
Step #1 - "Push container image to Artifact Registry": b37eb609be8f: Retrying in 9 seconds
Step #1 - "Push container image to Artifact Registry": 146a7f289aca: Retrying in 9 seconds
Step #1 - "Push container image to Artifact Registry": d6de8055d22f: Retrying in 9 seconds
Step #1 - "Push container image to Artifact Registry": 07e43b9577c8: Retrying in 6 seconds
Step #1 - "Push container image to Artifact Registry": d21707d42a97: Retrying in 8 seconds
Step #1 - "Push container image to Artifact Registry": b37eb609be8f: Retrying in 8 seconds
Step #1 - "Push container image to Artifact Registry": 146a7f289aca: Retrying in 8 seconds
Step #1 - "Push container image to Artifact Registry": d6de8055d22f: Retrying in 8 seconds
Step #1 - "Push container image to Artifact Registry": 07e43b9577c8: Retrying in 5 seconds
Step #1 - "Push container image to Artifact Registry": d21707d42a97: Retrying in 7 seconds
Step #1 - "Push container image to Artifact Registry": b37eb609be8f: Retrying in 7 seconds
Step #1 - "Push container image to Artifact Registry": 146a7f289aca: Retrying in 7 seconds
Step #1 - "Push container image to Artifact Registry": d6de8055d22f: Retrying in 7 seconds
Step #1 - "Push container image to Artifact Registry": 07e43b9577c8: Retrying in 4 seconds
Step #1 - "Push container image to Artifact Registry": d21707d42a97: Retrying in 6 seconds
Step #1 - "Push container image to Artifact Registry": b37eb609be8f: Retrying in 6 seconds
Step #1 - "Push container image to Artifact Registry": 146a7f289aca: Retrying in 6 seconds
Step #1 - "Push container image to Artifact Registry": d6de8055d22f: Retrying in 6 seconds
Step #1 - "Push container image to Artifact Registry": 07e43b9577c8: Retrying in 3 seconds
Step #1 - "Push container image to Artifact Registry": d21707d42a97: Retrying in 5 seconds
Step #1 - "Push container image to Artifact Registry": b37eb609be8f: Retrying in 5 seconds
Step #1 - "Push container image to Artifact Registry": 146a7f289aca: Retrying in 5 seconds
Step #1 - "Push container image to Artifact Registry": d6de8055d22f: Retrying in 5 seconds
Step #1 - "Push container image to Artifact Registry": 07e43b9577c8: Retrying in 2 seconds
Step #1 - "Push container image to Artifact Registry": d21707d42a97: Retrying in 4 seconds
Step #1 - "Push container image to Artifact Registry": b37eb609be8f: Retrying in 4 seconds
Step #1 - "Push container image to Artifact Registry": 146a7f289aca: Retrying in 4 seconds
Step #1 - "Push container image to Artifact Registry": d6de8055d22f: Retrying in 4 seconds
Step #1 - "Push container image to Artifact Registry": 07e43b9577c8: Retrying in 1 second
Step #1 - "Push container image to Artifact Registry": d21707d42a97: Retrying in 3 seconds
Step #1 - "Push container image to Artifact Registry": b37eb609be8f: Retrying in 3 seconds
Step #1 - "Push container image to Artifact Registry": 146a7f289aca: Retrying in 3 seconds
Step #1 - "Push container image to Artifact Registry": d6de8055d22f: Retrying in 3 seconds
Step #1 - "Push container image to Artifact Registry": unknown: Project 'project:mediamarkt-hackathon' not found or deleted.
Finished Step #1 - "Push container image to Artifact Registry"
ERROR
ERROR: build step 1 "gcr.io/cloud-builders/docker" failed: step exited with non-zero status: 1
-----
BUILD FAILURE: Build step failure: build step 1 "gcr.io/cloud-builders/docker" failed: step exited with non-zero status: 1
ERROR: (gcloud.builds.submit) build 7022abfc-bce5-4c57-9f98-09d71b84a292 completed with status "FAILURE"
```

Lo único que quedaba es comprobar la compilación y el registro en la consola de google.

## Post-data

El motivo del problema era porque no tenía el proyecto adecuado en google cloud. Tenía un nuevo proyecto y no el proyecto que había sido designado para mí. Una vez que he cambiado el proyecto, he podido habilitar el servicio de Cloud Build.

También estaba intentado subir el proyecto con un mecanismo antiguo, ahora ya no se usa Cloud Build y pronto quedará deprecado. Ahora se utiliza Artifact Registry.

Para pushear la imagen la construí en local con un nombre adecuado:

```
docker build . -t services/vite-app:v1.0.0
```

Tagué la imagen:

```
docker tag b1faa276127a  
gcr.io/aqvwa13jfqszux6hm4qihaeb4mklhk/services/vite-app:v1.0.0
```

Me autentiqué en el proyecto:

```
gcloud auth login  
gcloud config set project NAME_PROJECT
```

Cree el repositorio en el registry de google cloud:

```
gcloud artifacts repositories create services --repository-format=docker --  
location=us-central1 --description="Docker repository for services" --  
project=mediamarkt_hackathon
```

Después hice el push al registry de google cloud:

```
docker push gcr.io/aqvwa13jfqszux6hm4qihaeb4mklhk/services/vite-app:v1.0.0
```

## Generar un fichero YAML para Docker Compose

1. He creado el fichero [docker-compose.yaml](#).
2. He comprobado su despliegue en local.

 Despliegue local docker compose

## Generar los ficheros Terraform para tener la infraestructura como código y poder desplegar con Kubernetes

1. He creado los siguientes ficheros Terraform:

├─ [deploy.tf](#) │ ── [main.tf](#) │ ── [service.tf](#) │ ── [cluster.tf](#) │ ── [default/default.tfvars](#) ── [variables.tf](#)

1. Una vez creados los ficheros, he ejecutado el comando `terraform init` para inicializar el directorio de trabajo.
2. Para comprobar el aprovisionamiento de la infraestructura, he ejecutado el comando `terraform plan` para realizar las comprobaciones necesarias.
3. Una vez comprobado el aprovisionamiento, he ejecutado el comando `terraform apply` para aplicar los cambios.

Extra: Desplegar con Terraform los recursos Kubernetes definidos en un fichero yaml

Para aplicar desde Terraform los recursos Kubernetes definidos un fichero yaml, por ejemplo `deployment.yaml`, es necesario utilizar el recurso `null_resource` en Terraform y ejecutar el comando `kubectl apply`. Ejemplo:

```
resource "null_resource" "apply_deployments" {  
  provisioner "local-exec" {  
    command = "kubectl apply -f ${path.module}/deployments.yaml"  
  }  
}
```

## Pregunta sobre IAM

*"MediaMarkt quiere almacenar información sensible en Google Cloud Platform (GCP) y utiliza el principio de mínimo privilegio en la asignación de roles. Supongamos que eres el encargado de asignar roles en GCP, admin de la organización. Tu tarea es decidir qué rol sería el adecuado para cada grupo de personas: el equipo de DevOps para la creación de clusters en Kubernetes y el equipo de Finanzas en la gestión de la facturación en GCP. Detalla qué roles deben aplicar y los pasos que aplicaron en la Consola IAM GCP."*

### Respuesta:

En este caso, se puede asignar el siguiente rol a cada equipo:

- **Equipo de DevOps:** para la creación de clusters en Kubernetes, se puede asignar el rol Kubernetes Engine Developer o Kubernetes Engine Admin. Estos roles permiten la creación y gestión de clústeres de Kubernetes, pero no proporcionan acceso a otros servicios de GCP.
- **Equipo de Finanzas:** para la gestión de la facturación en GCP, se puede asignar el rol Billing Account Administrator o Billing Account Viewer. Estos roles permiten la gestión de la facturación y el uso de los recursos de GCP asociados a la cuenta de facturación, pero no proporcionan acceso a otros servicios de GCP.