Tarea #6  
Despliegue en GKE

línea corta



**PONDERACIÓN: 100**

**Horas Aproximadas: 6**

Universidad San Carlos de Guatemala

Facultad de ingeniería.

Ingeniería en ciencias y sistemas

# Introducción

# Introducción Esta tarea se centra en la utilización práctica de Google Kubernetes Engine (GKE), un servicio gestionado de Kubernetes en Google Cloud Platform. Los estudiantes aprenderán a configurar un clúster, desplegar una aplicación de prueba y exponerla utilizando un LoadBalancer. El objetivo es familiarizarse con el ciclo de vida completo de una aplicación en un entorno de Kubernetes en la nube, desde la creación del clúster hasta la limpieza de recursos, aplicando conocimientos sobre contenedores, orquestación y servicios en un proveedor cloud específico.

# Competencia(s)

Competencia(s) Al finalizar esta tarea, el estudiante será capaz de:

* Navegar y utilizar la consola de Google Cloud Platform para servicios de Kubernetes.
* Habilitar APIs necesarias para servicios en GCP.
* Crear y configurar un clúster de Kubernetes regional en GKE.
* Conectarse y gestionar un clúster de GKE utilizando Cloud Shell o gcloud CLI.
* Desplegar una aplicación contenerizada (ej. Nginx o una aplicación Go simple) en GKE utilizando manifiestos YAML.
* Exponer una aplicación en GKE al internet público utilizando un Service de tipo LoadBalancer.
* Validar el despliegue y acceso a la aplicación desplegada.
* Gestionar y eliminar recursos en GCP para evitar costos innecesarios.

# Contenido

**Objetivo:** Configurar un clúster de Kubernetes utilizando Google Cloud Platform (GCP), desplegar una aplicación de prueba y aplicar conocimientos sobre contenedores, orquestación y servicios.

**Pasos a seguir:**

1. **Inicia sesión en GCP (Google Cloud Console):** Accede a tu cuenta de Google Cloud.
2. **Crea un proyecto nuevo o usa uno existente:** Asegúrate de que la facturación esté habilitada para el proyecto.
3. **Habilita la API de Kubernetes Engine:** Busca "Kubernetes Engine API" en la barra de búsqueda del Cloud Console y habilítala para tu proyecto si aún no lo está.
4. **Crea un clúster regional con 3 nodos estándar:**
   * Utiliza la consola de GKE o el comando gcloud para crear un clúster.
   * Especifica que sea regional (ej. us-central1).
   * Configura 3 nodos. Puedes usar tipos de máquina estándar (ej. e2-medium o similar).
5. **Accede al clúster desde Cloud Shell o gcloud local:**
   * Obtén las credenciales para kubectl usando el comando: gcloud container clusters get-credentials NOMBRE\_DEL\_CLUSTER --region REGION\_DEL\_CLUSTER --project ID\_DEL\_PROYECTO
6. **Despliega una aplicación de prueba:**
   * Puedes usar una imagen pública como nginx:latest o una aplicación simple en Go que hayas contenerizado previamente y subido a un registro (como Google Container Registry - GCR, o Docker Hub).
   * Crea un archivo deployment.yaml para tu aplicación. Ejemplo para Nginx:
   * Aplica el deployment kubectl apply -f deployment.yaml
7. **Expón la app con un Service de tipo LoadBalancer:**

* Crea un archivo service.yaml:

1. **Valida la ejecución desde el navegador:**
   * Espera unos minutos a que el LoadBalancer obtenga una IP externa.
   * Obtén la IP externa con: kubectl get service mi-app-servicio (busca en la columna EXTERNAL-IP).
   * Abre un navegador web y navega a http://<EXTERNAL-IP>. Deberías ver tu aplicación (la página de bienvenida de Nginx o tu app Go).
2. **Limpia los recursos al finalizar:**
   * Elimina el servicio: kubectl delete service mi-app-servicio
   * Elimina el deployment: kubectl delete deployment mi-app-nginx
   * Elimina el clúster de GKE para evitar cargos: gcloud container clusters delete NOMBRE\_DEL\_CLUSTER --region REGION\_DEL\_CLUSTER --project ID\_DEL\_PROYECTO Confirma la eliminación cuando se te solicite.

# Recursos Adicionales

* Documentación oficial de Google Kubernetes Engine (GKE): <https://cloud.google.com/kubernetes-engine/docs>
* Quickstart de GKE: <https://cloud.google.com/kubernetes-engine/docs/quickstart>
* Creación de clústeres regionales en GKE: <https://cloud.google.com/kubernetes-engine/docs/how-to/creating-a-regional-cluster>
* Despliegue de una aplicación stateless en GKE: <https://cloud.google.com/kubernetes-engine/docs/how-to/deploying-stateless-application-deployment>
* Exposición de aplicaciones con Services en GKE: <https://cloud.google.com/kubernetes-engine/docs/how-to/exposing-apps>

# Forma de entrega

* Archivo PDF que incluya:
  + Capturas de pantalla de la Google Cloud Console mostrando: el proyecto seleccionado, la API de Kubernetes Engine habilitada, la lista de clústeres con tu clúster creado y sus detalles (ej. región, nodos).
  + Contenido completo de los archivos deployment.yaml y service.yaml utilizados.
  + Capturas de pantalla de los comandos clave ejecutados en Cloud Shell o terminal local (gcloud para crear clúster y obtener credenciales, kubectl apply ..., kubectl get pods, kubectl get service, gcloud ... delete cluster) y sus salidas relevantes.
  + Captura de pantalla del navegador accediendo a la aplicación a través de la IP externa del LoadBalancer, mostrando la aplicación funcionando.
  + Captura de pantalla que evidencie la eliminación de los recursos (ej. clúster ya no listado, o comando de eliminación exitoso).
* El archivo debe nombrarse Carnet\_T6.pdf.
* Entregar en Classroom con el Auxiliar correspondiente.

# Fecha de entrega

Fecha límite para realizar la entrega de la tarea: 7 días a partir de la asignación de la tarea.

# Rúbrica de Calificación

Establecer de una forma clara los aspectos que se calificarán

|  |  |  |
| --- | --- | --- |
| **Criterio** | **Descripción** | **Puntos Máximos** |
| **Configuración Proyecto y API en GCP** | Evidencia clara del proyecto configurado en GCP y la API de Kubernetes Engine habilitada. | **20** |
| **Creación del Clúster GKE** | Clúster regional con 3 nodos estándar creado correctamente en GKE, con evidencia de su configuración (capturas y/o comandos gcloud). | **10** |
| **Manifiesto Deployment YAML** | El archivo deployment.yaml está correctamente definido para la aplicación de prueba y se despliega sin errores. | **15** |
| **Manifiesto Service YAML (LoadBalancer)** | El archivo service.yaml es de tipo LoadBalancer, expone correctamente el deployment y se obtiene una IP externa funcional. | **25** |
| **Validación y Acceso a la Aplicación** | La aplicación es accesible correctamente a través de la IP externa del LoadBalancer y se demuestra su funcionamiento mediante captura. | **20** |
| **Limpieza de Recursos** | Se proporciona evidencia clara de la eliminación del servicio, deployment y, fundamentalmente, del clúster de GKE. | **10** |

# Valores

Valores Establecer de una forma clara que pasa si no se cumplen con los valores del laboratorio: El trabajo debe ser individual. Cualquier indicio de copia total o parcial resultará en la anulación de la tarea para todos los involucrados. Se espera que el estudiante realice el desarrollo de manera ética y profesional, consultando los recursos adicionales como guía y no como fuente de copia directa. Las dudas deben realizarse en el foro semanal. Recordar la importancia de eliminar los recursos en GCP para evitar costos inesperados.

guion corto