

Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ciencias y Sistemas  
Sistemas Operativos 2  
Sección A



## Manual Tecnico - Docker y Kubernetes

### GRUPO 4

Repositorio	<a href="https://github.com/Yaiza81/SO2_1S2022_P_G4">https://github.com/Yaiza81/SO2_1S2022_P_G4</a>
-------------	---

## Manual Técnico:

### Introducción:

En el siguiente escrito se tiene la finalidad de proporcionar al usuario la lógica utilizada para realizar la configuración e implementación del despliegue del sistema con Docker y Kubernetes, se documentaron los pasos empleados en la construcción del programa.

Este manual no está dedicado para enseñar sobre Prometheus y Grafana, ni las otras herramientas utilizadas para el desarrollo de la aplicación. Para conocer la guía respectiva a las otras dos herramientas se debe consultar el “Manual Técnico - Prometheus y Grafana”.

### Objetivo del manual:

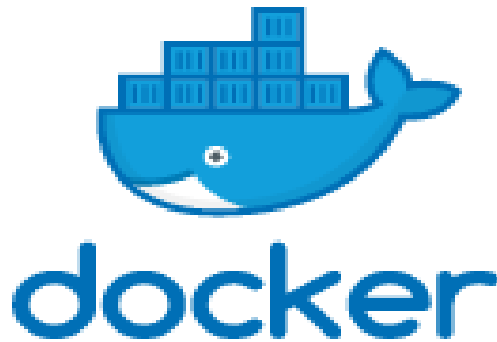
Se pretende dar a conocer las herramientas y la manera en que se utilizaron para realizar la implementación de la siguiente aplicación. Desde la configuración de los archivos “Dockerfile” hasta los pasos empleados para realizar el despliegue con Kubernetes.

## Aplicación:

### Descripción de herramientas utilizadas:

#### ➤ Docker

Docker es una plataforma de software que permite crear, probar e implementar aplicaciones rápidamente. Docker empaqueta software en unidades estandarizadas llamadas contenedores que incluyen todo lo necesario para que el software se ejecute, incluidas bibliotecas, herramientas de sistema, código y tiempo de ejecución. Con Docker, puede implementar y ajustar la escala de aplicaciones rápidamente en cualquier entorno con la certeza de saber que su código se ejecutará.



#### ➤ Kubernetes

Kubernetes es una plataforma portable y extensible de código abierto para administrar cargas de trabajo y servicios. Kubernetes facilita la automatización y la configuración declarativa. Tiene un ecosistema grande y en rápido crecimiento. El soporte, las herramientas y los servicios para Kubernetes están ampliamente disponibles.



## Docker:

### Configuración de Docker:

- Descargar Docker de la página oficial
- Instalación de Docker en windows 10 o en linux
- Reiniciar la computadora

Cuenta en DockerHub :

- Dirigirnos al sitio de DockerHub
- Registrar un nuevo usuario
- Activar la cuenta desde gmail

### Configuración de archivos Dockerfile:

Crear un archivo Dockerfile para el frontend y uno para el backend

Backend:

Se especifica la imagen a utilizar:

```
FROM node:10-alpine
```

Se crea un nuevo directorio y se dan los permisos al usuario

```
RUN mkdir -p /home/node/app/node_modules && chown -R node:node  
/home/node/app
```

Indica el directorio sobre el que se van a aplicar las instrucciones siguientes.

```
WORKDIR /home/node/app
```

Copiar los archivos especificados

```
COPY package*.json ./
```

Instalar las dependencias

```
RUN npm install
```

Indicar el puerto utilizado

```
EXPOSE 4000
```

Se indica los comandos

```
CMD ["npm", "run", "dev"]
```

Posteriormente utilizamos `docker build .` para poder construir el contenedor.

Se corre el contenedor

Se sube la imagen a la cuenta de DockerHub

## Kubernetes:

### Configuración de Kubernetes:

- Instalación de herramienta SDK de Google (gcloud)

Esta herramienta se instaló en la máquina virtual donde se estaba trabajando para poder utilizar los recursos de Google Cloud, ya que se utilizó el servicio de Kubernetes Engine para almacenar el clúster que despliega la aplicación.

Pasos:

Se realiza la instalación y configuración de la herramienta CLI, introduciendo los siguientes comandos en la terminal de la máquina virtual

- `curl -O https://dl.google.com/dl/cloudsdk/channels/rapid/downloads/google-cloud-sdk-297.0.0-linux-x86_64.tar.gz`
- `tar -zxvf google-cloud-sdk-297.0.0-linux-x86_64.tar.gz google-cloud-sdk`
- `./google-cloud-sdk/install.sh`

Se realiza la configuración de la cuenta de Google donde se va a trabajar, respondiendo lo solicitado en la terminal

- *Do you want to help improve the Google Cloud SDK (y/N)? N*
- *Modify profile to update your \$PATH and enable shell command completion? Do you want to continue (Y/n)? Y*
- *Enter a path to an rc file to update, or leave blank to use [/home/pablo/.bashrc]: <<ENTER>>*

Se inicializa la herramienta CLI y se seleccionan las opciones deseadas

- `./google-cloud-sdk/bin/gcloud init`

```
pablo@kube-vm:~/gcloud$ ls
google-cloud-sdk google-cloud-sdk-297.0.0-linux-x86_64.tar.gz
pablo@kube-vm:~/gcloud$ ./google-cloud-sdk/bin/gcloud init
Welcome! This command will take you through the configuration of gcloud.
```

- ```
Choose the account you would like to use to perform operations for
this configuration:
[1] 343270227939-compute@developer.gserviceaccount.com
[2] Log in with a new account
Please enter your numeric choice: 2
```

- ```
Your credentials may be visible to others with access to this
virtual machine. Are you sure you want to authenticate with
your personal account?

Do you want to continue (Y/n)? y
```

- Go to the following link in your browser:
- [https://accounts.google.com/o/oauth2/auth?client\\_id=32555940559\\_apps.googleusercontent.com&redirect\\_uri=urn%3Aietf%3Awww%3Aauth%3A2%03%3Aaccounts.google.com&response\\_type=code&scope=openid+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fuserinfo.email+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fappengine.admin+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcompute+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Faccounts.reauth&code\\_challenge=wsbN0YkLkx-gx5HQd3W10MH2XyRxopX0nwrbcURHw&code\\_challenge\\_method=S256&access\\_type=offline&response\\_type=code&prompt=select\\_account](https://accounts.google.com/o/oauth2/auth?client_id=32555940559_apps.googleusercontent.com&redirect_uri=urn%3Aietf%3Awww%3Aauth%3A2%03%3Aaccounts.google.com&response_type=code&scope=openid+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fuserinfo.email+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcloud-platform+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fappengine.admin+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcompute+https%3A%2F%2Fwww.googleapis.com%2Fauth%2Faccounts.reauth&code_challenge=wsbN0YkLkx-gx5HQd3W10MH2XyRxopX0nwrbcURHw&code_challenge_method=S256&access_type=offline&response_type=code&prompt=select_account)

- Enter verification code: 4/1AX4Xfwgomp3J3yzcBBPobCMBWegAvmhwgQJhZQnK0x0xo2HV1Ie7ZqsQs8s

- ```
Updates are available for some Cloud SDK components. To install them,
please run:
$ gcloud components update

You are logged in as: [pablopachecof.18@gmail.com].

Pick cloud project to use:
[1] flawless-psyche-341722
[2] modern-yeti-334017
[3] Create a new project
Please enter numeric choice or text value (must exactly match list
item): 1
```

```

Your current project has been set to: [flawless-psyche-341722].

Do you want to configure a default Compute Region and Zone? (Y/n)? y

Which Google Compute Engine zone would you like to use as project
default?
If you do not specify a zone via a command line flag while working
with Compute Engine resources, the default is assumed.
[1] us-east1-b
[2] us-east1-c
[3] us-east1-d
[4] us-east4-c
[5] us-east4-b
[6] us-east4-a
[7] us-central1-c
[8] us-central1-a
[9] us-central1-f
[10] us-central1-b
[11] us-west1-b
[12] us-west1-c
[13] us-west1-a
[14] europe-west4-a
[15] europe-west4-b
[16] europe-west4-c
[17] europe-west1-b
[18] europe-west1-d
[19] europe-west1-c

```

- *Seleccionar la zona deseada, en el presente caso se seleccionó la zona 7*

```

[49] asia-northeast2-c
[50] asia-northeast3-a
Did not print [45] options.
Too many options [95]. Enter "list" at prompt to print choices fully.
Please enter numeric choice or text value (must exactly match list
item): 7

Your project default Compute Engine zone has been set to [us-central1-c].
You can change it by running [gcloud config set compute/zone NAME].

Your project default Compute Engine region has been set to [us-central1].
You can change it by running [gcloud config set compute/region NAME].

```

- Instalar Kubectrl:

Herramienta que se utilizó para comunicarse con el clúster de Kubernetes que provee Google Kubernetes Engine

- Crear el nodo maestro que será albergado en GKE

## Configuración de archivos YAML:

- Se crea el Namespace que almacenará la aplicación
- Se introducen las configuraciones de nombres para la aplicación (tanto para backend como frontend)

- Se crean los servicios respectivos para comunicarse entre ambas apps (tipo LoadBalancer)
- Se configura el servicio de RollinUpdate

### Despliegue de la aplicación:

- `kubectl apply -f Namespace.yaml`
- `kubectl apply -f Backend.yaml`
- `kubectl apply -f Frontendv1.yaml`
- `kubectl get svc -n sopes2`
- `kubectl get deploy -n sopes2`