



Universidad
Tecmilenio®





Contenerización y servicios con Kubernetes

Contenerización con
Kubernetes

Semana 10



Para trabajar con Kubernetes no solo es necesario conocer la forma en cómo estos se encuentran estructurados, sino también, en su integración y la forma en cómo se pueden maximizar sus funcionalidades a través de la integración con otros servicios.

Es de vital importancia conocer cómo funcionan las integraciones entre los mismos clústeres para obtener el máximo provecho de ellos, ya que se comparten recursos del sistema, así como capacidades de almacenamiento y de procesamiento, por lo que saber realizar configuraciones exitosas para obtener su máximo provecho es un tema de vital importancia para los equipos de trabajo de TI.



Integración de Kubernetes

Los Kubernetes son una herramienta de gestión de recursos, implementación y operación para ayudar a los usuarios en la implementación correcta de las aplicaciones desarrolladas, por lo que es importante que se considere la integración de estos de manera natural para lograr mejores resultados.



Redes de Kubernetes

Para ejecutar una tarea, es necesario verificar si se encuentra asociada con una oferta; en caso de no ser así, el programador arrojará un error, ya que cuando la tarea se esté ejecutando deberá tener recursos de una oferta determinada para ejecutarse.

Kubernetes está compuesto por nodos en los cuales corren contenedores y pods que se interrelacionan con el fin de llevar a cabo operaciones para las cuales han sido programados.



Redes de Kubernetes

De acuerdo con Unzue (2020), los tipos de comunicación de red en un clúster de Kubernetes son los siguientes:

- Contenedor a contenedor.
- Pod a pod.
- Pod a servicio.
- Tráfico externo a servicio.

Se combinan con las siguientes reglas:

1	Todos los pods de un clúster de Kubernetes pueden comunicarse con otros pods de otros nodos.
2	Los elementos del sistema pueden tener comunicación directa con otros pods en ese mismo nodo.
3	Los contenedores tienen la facilidad de comunicarse directamente entre sí.
4	Sin excepción, todos los nodos pueden comunicarse con todos los contenedores.
5	La dirección IP es la misma para todos los contenedores.

Arquitectura de Kubernetes

Para entender lo anterior, es como un modelo de red de la virtualización con máquinas virtuales, donde cada pod tiene una IP dinámica y esto facilita la asignación de puertos, equilibrio de cargas y llevar una aplicación a Kubernetes.

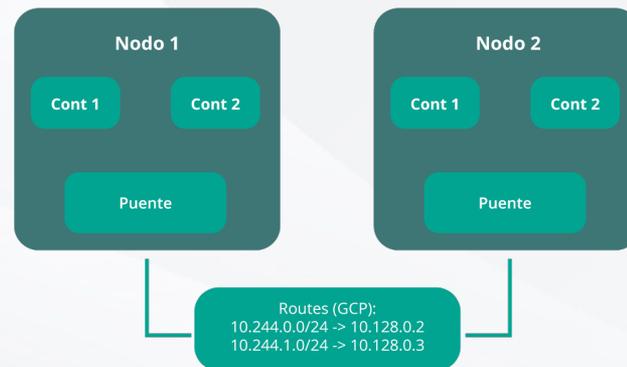


Figura 1. Redes de Kubernetes.

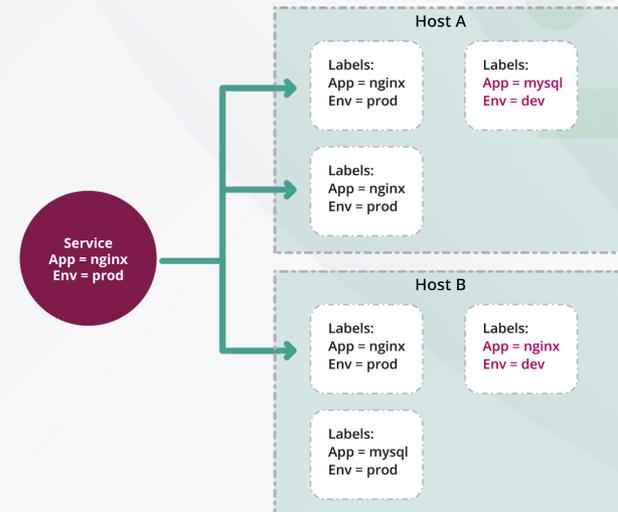


Figura 2. Conectividad de conectores.

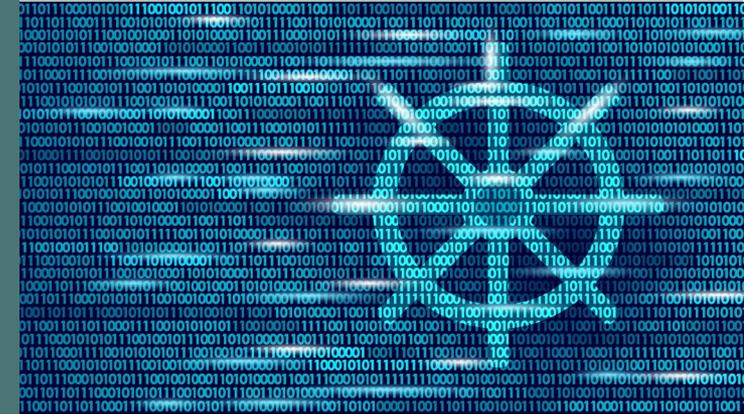
Las comunicaciones en un clúster de Kubernetes se dan a través de un canal seguro, como HTTPS en el puerto 443, con la autenticación de clientes, y para que exista una conexión segura, los nodos deben conectar con el API Server en conjunto con credenciales válidas de autenticación.

El concepto general es asignar subredes a cada host para posteriormente configurar enrutamientos entre ellos y así enviar el tráfico del contenedor de forma correcta.

Almacenamiento de Kubernetes

Los pods son temporales por definición, por lo que cuando se destruyen, se pierde toda la información que estaba en ellos. Es importante conocer la forma de conservar todos estos datos

Volumen: se puede entender como un directorio principal donde tienen acceso todos los contenedores de un pod; estos volúmenes pueden soportar todos los contenedores dentro del pod de Kubernetes (awselasticblockstore, azuredisk, azurefile, cephfs y cinder).



Explicación



Almacenamiento persistente

Kubernetes también ofrece la facilidad de conservar los datos a través del almacenamiento persistente, o bien, almacenamiento definitivo en bases de datos, discos duros, unidades externas, etc., a través del uso de PersistentVolume y PersistentVolumeClaims.

1. Investiga en Internet tres ejemplos de configuración de redes de Kubernetes que son utilizadas hoy en día por equipos de trabajo de TI para un desarrollo exitoso (puedes tomar como referencia las tablas del tema).

Tipo de red	Ejemplo de configuración de red	Donde se usa
Clúster a Master		
Master a Clúster		
Túneles SSH		

2. Con base en el tema de Volumen, selecciona tres de los volúmenes en Kubernetes mencionados y busca un ejemplo de la escritura del código para su correcta configuración en un proyecto.

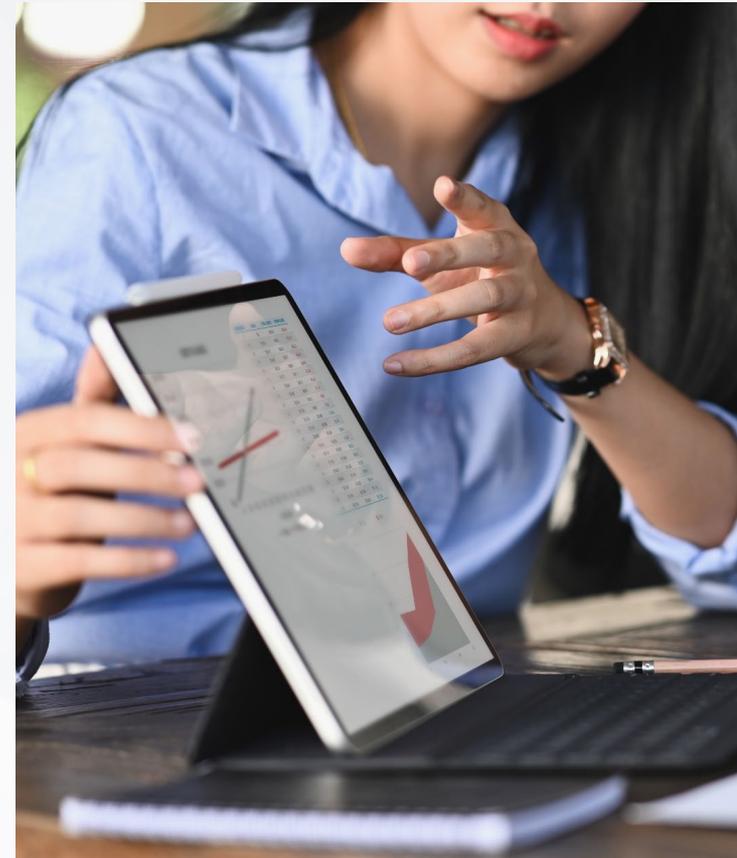
Tipos de volúmenes en Kubernetes:
 Awselasticblockstore, azuredisk, azurefile, cephfs, cinder, configmap, downwardapi, emptydir, fc, flocker, gcepersistentdisk, gitrepo, entre otros.



¿Cuál debería ser la configuración más adecuada para trabajar Kubernetes en red?

¿Qué consideraciones se deben tener para evitar la pérdida de datos y obtener una configuración apropiada de almacenamiento de Kubernetes?

Como se ha visto a lo largo de este tema, es muy importante conocer la estructura y el funcionamiento de los Kubernetes en el trabajo del día con día, saber que pueden existir diferentes tipos de integraciones con otras plataformas para realizar trabajos más completos y estructurados, así como conocer las configuraciones esenciales de las redes de Kubernetes, con el fin de obtener el mejor provecho de ellas y, finalmente, saber las opciones disponibles para salvaguardar la información de los pods y nodos antes de que se pierdan por completo, ya que el uso de estos recursos y desarrollos constituirá un ahorro de tiempo y recursos para la ejecución de trabajos futuros y permitirá la mejora continua a través de las lecciones aprendidas en proyectos iniciales.



Bibliografía



Unzue, R. (2020). *Kubernetes: Networking*. Recuperado de <https://www.maquinasvirtuales.eu/kubernetes-networking/>



Contenerización y servicios con Kubernetes

Servicios manejados de
Kubernetes - Parte 1

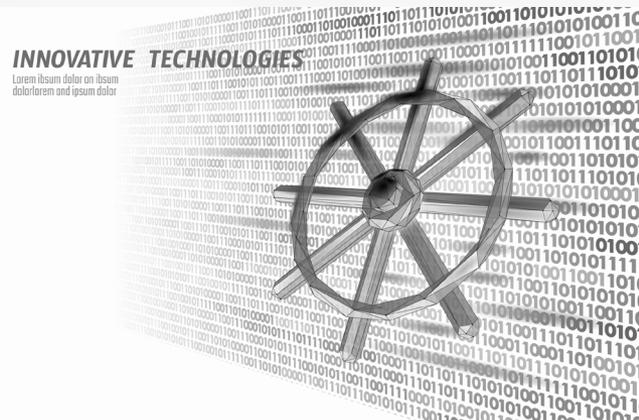
Semana 10



Existen servicios manejados de Kubernetes que pueden apoyar a la administración de los contenedores en lo referentes a los clústeres de dichos Kubernetes.

Los servicios se enfocan en brindar facilidades en cuanto a su manejo, gestión, optimización y seguridad, permitiendo, a través de procesos automatizados o semiautomatizados, una mejoría en tiempos y costos por aplicación desarrollada.

La necesidad de contar con estos servicios en la nube es que puedan ofrecer sus servicios de manera fácil y transparente, por ello, la mayoría de las soluciones tienen dicha ventaja, además proveen infraestructura de soporte para facilitar el trabajo de desarrolladores de aplicaciones.



AWS ELASTIC KUBERNETES SERVICE (EKS)

La plataforma AWS (Amazon Web Services) perteneciente a Amazon, que se considera una solución para ejecutar aplicaciones en la nube, también ayuda en la ejecución de Kubernetes de una manera amigable, ofreciendo un marco para operar en la nube. Amazon EKS es un servicio de contenedores administrado para ejecutar y escalar aplicaciones de Kubernetes en la nube o en las instalaciones.

Funciones de Amazon EKS				Ventajas:
Escalamientos automáticos de las estancias del plano de control.	Alta disponibilidad garantizada por la ejecución y escalamiento de control de Kubernetes.	Integración con varios servicios que le permiten mayor escalabilidad y seguridad a las aplicaciones.	Ejecución de versiones actualizadas del software de Kubernetes con el fin de acceder a complementos y herramientas de la comunidad.	

- Control de las instancias.
- Permite el acceso a código abierto sin bloqueos.
- Portabilidad.
- Cloudbursting.

Azure AKS

Azure Kubernetes Service (AKS), de acuerdo con Techtarget (2021), es un servicio de gestión de contenedores basado en el sistema de Kubernetes de código abierto, el cual está habilitado en la nube pública de Microsoft Azure; permite gestionar funciones como implementación, escalamiento, administración de contenedores y aplicaciones basadas en contenedores.



Incluye componentes de seguridad para los siguientes aspectos:

- Brindar autenticación y autorización.
- Protección de aplicaciones a través de Azure Policy.
- Uso de Microsoft Defender.
- Permite que el clúster de AKS ejecute las actualizaciones de seguridad del SO.
- Brindar un tráfico de pod seguro.

Contiene recursos que le permiten:

- Interacción con herramientas de gestión.
- Mantenimiento de la configuración del clúster.
- Programación y especificación de los nodos.
- Supervisión de las réplicas de Kubernetes.

Google GKE



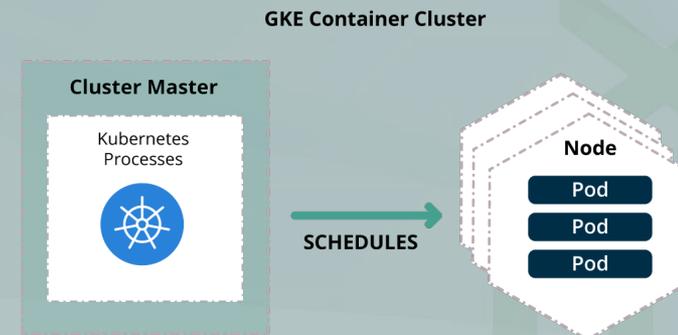
Plataforma que brinda un ambiente de trabajo gestionado para implementar, administrar y escalar aplicaciones en contenedores por medio de la infraestructura de Google. Trabaja sobre un entorno de varias máquinas agrupadas en un clúster. A través del uso de comandos de Kubernetes se pueden administrar e implementar las aplicaciones, establecer políticas, así como revisar el estado de las cargas de trabajo; permite, además, la administración automática, la supervisión de la capacidad de funcionamiento de los contenedores de aplicaciones, actualizaciones progresivas, etc.

Beneficios

- Balanceo de cargas.
- Grupos de nodos.
- Ajuste automático de escala.
- Reparación automática de nodos.

Modos de operación

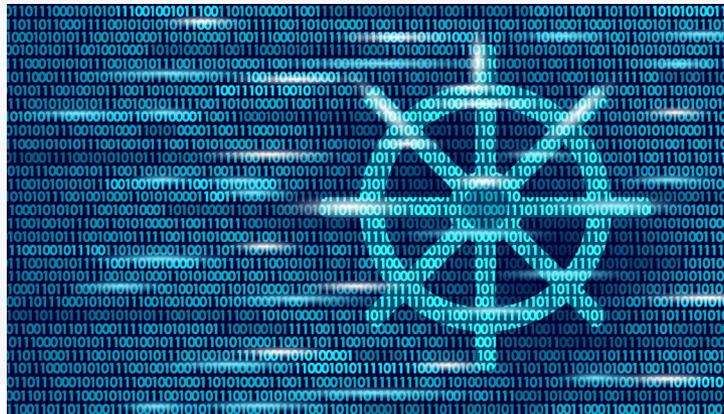
- Autopilot.
- Estándar.



Evidencia del uso de Kubernetes

Un ejemplo del uso de Kubernetes es el de Principal Financial Group, que es una empresa que nació en el año de 1879 con el fin de brindar servicios financieros y actualmente lo hace a nivel global. En todos estos años ha pasado por diversas reestructuras para acoplarse a los cambios evolutivos naturales de sus clientes. En el año 2020, toma la decisión de migrar sus servicios más significativos a la nube para agilizar más sus procesos dirigidos al cliente final.

De esta manera, podemos ver que AWS de Amazon no solo provee la infraestructura técnica para realizar proyectos con clúster de Kubernetes, sino que proporciona un ambiente de aprendizaje integral que permite a los colaboradores conocer a detalle, y de acuerdo con gaps de aprendizaje específicos, las mejores prácticas del uso de su plataforma para obtener el mejor provecho de esta.





1. Completa la información de la siguiente tabla con respecto a los servicios de Kubernetes vistos en el tema, enunciando los elementos solicitados para cada uno de ellos.

Servicio	Característica principal	Desventaja principal	Proyectos a los que aplica:
AWS EKS			
Azure AKS			
Google GKE			

2. Realiza un pequeño cuadro comparativo entre los tres servicios, donde incluyas elementos como seguridad, automatización, tipo de nube, costo y ventajas de la marca.

¿Es necesario contar con un servicio administrado de Kubernetes para operar un clúster?

¿Qué consideraciones se deben tener en cuenta para tomar la mejor decisión en este tipo de servicios?



Existe la necesidad de contar con plataformas que ayuden a optimizar tiempos y esfuerzos de los equipos de TI. Estos servicios tienen características y ventajas diferentes al usarse, pero tienen en común que permiten automatizar la administración de clústeres de Kubernetes, facilitando así su uso, gestión y desempeño en ejecución. Por lo anterior, será importante considerar todos los elementos que ofrecen y analizar cuál sería el más adecuado para una posible implementación en un proyecto dado.



Amazon. (2022). *¿Qué es Amazon EKS?* Recuperado de https://docs.aws.amazon.com/es_es/eks/latest/userguide/what-is-eks.html

Amazon. (2022). *Principal Financial Group acelera la innovación en AWS al capacitar a 1650 empleados en menos de 8 meses.* Recuperado de https://aws.amazon.com/es/solutions/case-studies/principal-financial-group-case-study/?did=cr_card&trk=cr_card

Google Cloud. (2022). *GKE overview | Kubernetes Engine Documentation.* Recuperado de https://cloud.google.com/kubernetes-engine/docs/concepts/kubernetes-engine-overview?hl=es_419

Microsoft. (2022). *Conceptos de seguridad de las aplicaciones y los clústeres en Azure Kubernetes Service (AKS).* Recuperado de <https://docs.microsoft.com/es-es/azure/aks/concepts-security>



Contenerización y servicios con Kubernetes

Servicios manejados de Kubernetes - Parte 2



Semana 10



Existen diversas plataformas que ayudan a implementar servicios de clústeres de forma física, virtual y también híbrida. Continuando en esa línea, se presenta información de otro servicio utilizado mayormente para proyectos grandes que requieren grandes cantidades de almacenamiento y procesamiento de datos, el cual se verá a detalle en el contenido de este tema.

Los servicios en la nube cada vez son más requeridos por las organizaciones, no solo por las ventajas y facilidades en su gestión, sino por la facilidad de acceso, portabilidad y transparencia en sus operaciones.



Openshift (Red Hat)

Al igual que los ejemplos anteriores, es una plataforma de Kubernetes empresarial que permite diseñar, implementar y ejecutar aplicaciones de forma estandarizada, ya sea en la nube, en el lugar de trabajo o en la red, ofreciendo elementos de autoservicio para los equipos de trabajo de TI y ayudarles a agilizar las ideas de la etapa de desarrollo a la producción de forma eficiente.



Ventajas, de acuerdo con S&M Services (2020):

- Multicloud.
- Nube híbrida.
- Monitorización y automatización sencilla.



Es importante mencionar que Openshift puede alojarse en diferentes entornos, incluyendo máquinas *bare metal* (servidores dedicados con alto rendimiento y alta seguridad), nubes públicas y privadas, así como dispositivos *edge*.



Levantar un clúster de Kubernetes y despliegue de una aplicación

De acuerdo con Amazon EKS (2022), para comenzar, se deben tener algunas consideraciones iniciales:

- Contar con un clúster de Kubernetes. Para más detalles de este punto, favor de revisar la información **Amazon EKS**.
- Para implementar en Windows es necesario tener **compatibilidad habilitada** y al menos un nodo de Windows de Amazon EC2.
- Contar con Kubectl instalado y configurado para comunicarse con el clúster.

Ejemplo

1. Creación de un espacio de nombres, el cual permitirá agrupar los recursos de Kubernetes.

```
kubectl create namespace eks-sample-app
```

Tabla 1. Ejemplo para ejecutar el valor.

2. Posteriormente, se realiza la implementación de Kubernetes que extrae una imagen del contenedor y hace tres réplicas o pods individuales en su clúster.

Creación de un recurso en Kubernetes

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: eks-sample-windows-deployment
  namespace: eks-sample-app
  labels:
    app: eks-sample-windows-app
spec:
  replicas: 3
  selector:
    matchLabels:
      app: eks-sample-windows-app
  template:
    metadata:
      labels:
        app: eks-sample-windows-app
    spec:
      affinity:
        nodeAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            nodeSelectorTerms:
              - matchExpressions:
                  - key: beta.kubernetes.io/arch
                    operator: In
                    values:
                      - amd64
      containers:
        - name: windows-server-iis
          image: mcr.microsoft.com/windows/servercore:ltsc2019
          ports:
            - name: http
              containerPort: 80
          imagePullPolicy: IfNotPresent
          command:
            - powershell.exe
            - -command
              - "Add-WindowsFeature Web-Server; Invoke-WebRequest -
                UseBasicParsing -Uri
                'https://dotnetbinaries.blob.core.windows.net/servicemonitor/2
                .0.1.6/ServiceMonitor.exe' -OutFile 'C:\\ServiceMonitor.exe';
                echo '<html><body><br/><br/><marquee><H1>Hello
                EKS!!<H1><marquee></body><html>' >
                C:\\inetpub\\wwwroot\\default.html; C:\\ServiceMonitor.exe
                'w3svc'; "
```

Creación de un recurso en Kubernetes

3. Se da de alta un servicio.
4. Consultar todos los recursos existentes.
5. Ejemplo de consulta de los detalles del servicio con la siguiente línea de código.

```
apiVersion: v1
kind: Service
metadata:
  name: eks-sample-windows-service
  namespace: eks-sample-app
  labels:
    app: eks-sample-windows-app
spec:
  selector:
    app: eks-sample-windows-app
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

Tabla 4. Ejemplo para ejecutar la instrucción.

```
kubectl get all -n eks-sample-app
```

Tabla 6. Ejemplo para consultar recursos existentes.

```
kubectl -n eks-sample-app describe service eks-sample-windows-service
```

Tabla 7. Ejemplo para ejecutar el código.

```
Name:          eks-sample-linux-service
Namespace:     eks-sample-app
Labels:        app=eks-sample-linux-app
Annotations:   <none>
Selector:      app=eks-sample-linux-app
Type:          ClusterIP
IP Families:   <none>
IP:            10.100.74.8
IPs:           10.100.74.8
Port:          <unset> 80/TCP
TargetPort:    80/TCP
Endpoints:     192.168.24.212:80,192.168.50.185:80,192.168.63.93:80
Session Affinity: None
Events:        <none>
```

Imagen 2. Ejemplo de salida del servicio implementado.

Creación de un recurso en Kubernetes

6. Se pueden consultar los detalles de los pods con la siguiente línea de código:
7. Para desconectarse, solo es necesario escribir “exit”.
8. Al terminar el ejercicio se puede borrar todo lo realizado con la siguiente instrucción:

```
kubectl -n eks-sample-app describe pod eks-sample-windows-deployment-65b7669776-m6qxz
```

Tabla 8. Ejemplo para ejecutar el código.

```
kubectl delete namespace eks-sample-app
```

Tabla 9. Ejemplo para ejecutar la instrucción.

Beneficios de utilizar plataformas para clústeres de Kubernetes:

Los beneficios son muchos, ya que no solo proporcionan facilidad en su uso (con capacitación en línea y en diferentes niveles), sino que también:

- Dan flexibilidad para adaptarse a cada proyecto dependiendo de su tamaño y necesidades.
- Permiten que el proyecto pueda tener un ahorro considerable de gastos, ya que se puede contar con herramientas que ayuden a hacer el producto más atractivo y rentable para el cliente final.
- Se puede tener rastreo de operaciones por usuario, lo que se traduce en una mayor confianza a lo largo del ciclo de vida del proceso.
- Se puede garantizar mayor escalabilidad al enlazarse con proyectos a futuro.
- Mejor desempeño a través de la infraestructura en la nube.
- Mayor seguridad al correr en entornos previamente comprobados en diferentes combinaciones de plataformas: virtuales, físicas o híbridas.

1. Investiga en Internet las características generales de cada una de las diferentes plataformas de servicios para realizar una toma de decisiones más efectiva al momento de implementar un proyecto de clúster de Kubernetes.

Servicio	Costo por licencia o de implementación	Tipo de empresas para las que aplica.	Tipo de infraestructura (física, virtual, híbrida).	Facilidad de implementación.
AWS EKS				
Azure AKS				
Google GKE				
RedHat				

2. Lee y resuelve el siguiente caso: La empresa OrangeTech ofrece servicios financieros en más de ocho países y tiene una cartera de más de 500,000 clientes. Es una empresa tradicional que ha ido evolucionando a través del tiempo y se encuentra en una fase de migración a servicios en la nube.

Sus principales necesidades a resolver son las siguientes:

- Proporcionar servicios bancarios en la nube.
- Implementar la nueva app móvil.
- Reducir procesos para nuevas solicitudes de créditos bancarios e hipotecarios.

Con base en lo investigado en el punto 1, determina qué tipo de plataforma de servicios les conviene contratar y cuáles serían las ventajas de hacerlo. Favor de ampliar tu respuesta.



¿Cuál servicio en la nube consideras más apropiado para pymes y por qué?

¿Cuáles deberían ser los parámetros que nos permitan tomar una decisión de adopción de una plataforma y bajo qué esquema?, ya sea físico, virtual o híbrido.

El ejemplo de implementación implica no solamente llevar a cabo las instrucciones y ejecución de líneas de código de manera organizada y estructurada, sino contar con herramientas de apoyo y configuraciones previas para trabajar en un ambiente seguro y amigable.

Es importante recordar que cuanto más podamos conocer acerca de las características y ventajas que ofrece un servicio, más podremos explotar dichos recursos junto con nuestro equipo de trabajo y así estar alineados en la misma estrategia de operación para alcanzar las metas establecidas.



Bibliografía



S&M Services. (2020). *¿Qué es OpenShift?* Recuperado de <https://sm-services.es/que-es-openshift/>

Amazon EKS. (2022). *Implementar una aplicación de ejemplo*. Recuperado de https://docs.aws.amazon.com/es_es/eks/latest/userguide/sample-deployment.html