**Jorge Edison Lascano**

**Dockers y Kubernetes**

**TAREA Unidad 2**

Explique cada uno de los componentes del ciclo de vida de un contenedor en Kubernetes, cada componente deberá ser explicado en menos de 5 líneas.

Componentes de un contenedor en Kubernetes

* Pods
* Nodos
* Control plane
* Scheduler
* Deployments
* Service
* ConfigMap
* Secret
* Namespace
* Pods
  + Un pod es un conjunto de contenedores, que comparten almacenamiento, red, y especificaciones.
  + Los pods se crean, se calendarizan, se ejecutan, se realiza chequeos de salud, y se terminan.
* Nodos
  + Son las Máquinas virtuales o las computadoras físicas donde se despliegan los pods. Los nodos se agrupan en clústeres.
  + Los nodos se registran, se monitorean, se administra su ciclo de vida y se manejan errores.
* Control Plane
  + El control plane se compone de API que permite la administración de nodos, Scheduler, Contoller Manager que monitorea el estado de los clústeres y hace ajustes tales como el escalamiento o las fallas de los nodos y etcd que es un almacenan de pares key-value distribuidas que contienen los datos de configuración y el estado del clúster, permiten asegurar la persistencia.
* Scheduler
  + Se encarga de crear nuevos pods o asignar recursos a los pods en ejecución maximizando la eficiencia del uso de recursos de los nodos.
* Deployment
  + Define la administración, actualización y escalado de pods. Se implementan nuevas versiones de aplicaciones y en caso de fallos se recupera a estados correctos. Se realizan deployments on-premise en master nodes y worker nodes, en la nube en worker nodes.
  + El deployment se define, se crea, se actualizan con rollbacks, se escalan y se hacen roll-back en caso de fallas.
* Service
  + Proporciona direcciones IP y un nombre de DNS para acceder a los pods. Permite la comunicación entre pods y también un correcto load balancing. En general, abstrae el acceso a un conjunto de pods.
  + Los servicios se crean, se descubren, y se actualizan.
* ConfigMap
  + Permite al almacenar configuraciones y acceso a la información sin necesidad de reconstruir imágenes.
  + El configmap se crea, se usa y se actualiza
* Secret
  + Permite el manejo de datos sensibles. Los Secrets están encriptados, están en el deployment y pueden ser inyectados como variables de ambiente. Maneja passwords y API keys.
  + Los secrets se crean, se usan y se actualizan.
* Namespace
  + Proporciona un contexto lógico para organizar los recursos de un clúster. Permite administrar políticas de acceso y recursos.
  + Los namespaces se crean y se administran