**Application Development Architecture**

* Monolithic Application
* Microservice Application

**Monolithic Application**

* The entire project is built on the single code base.
* We will have only one repository for managing the source code of the application.
* We will have single artifact for the project.

**Advantages**

* Deployment of the application is easy.
* Scalability can be achieved in a simple way.

**Disadvantages**

* Build, deployment and starting of the application takes lot of time.
* Patching and upgrades impacts the application availability.
* If load increases on one module then also we have to scale the entire application.

**Microservice**

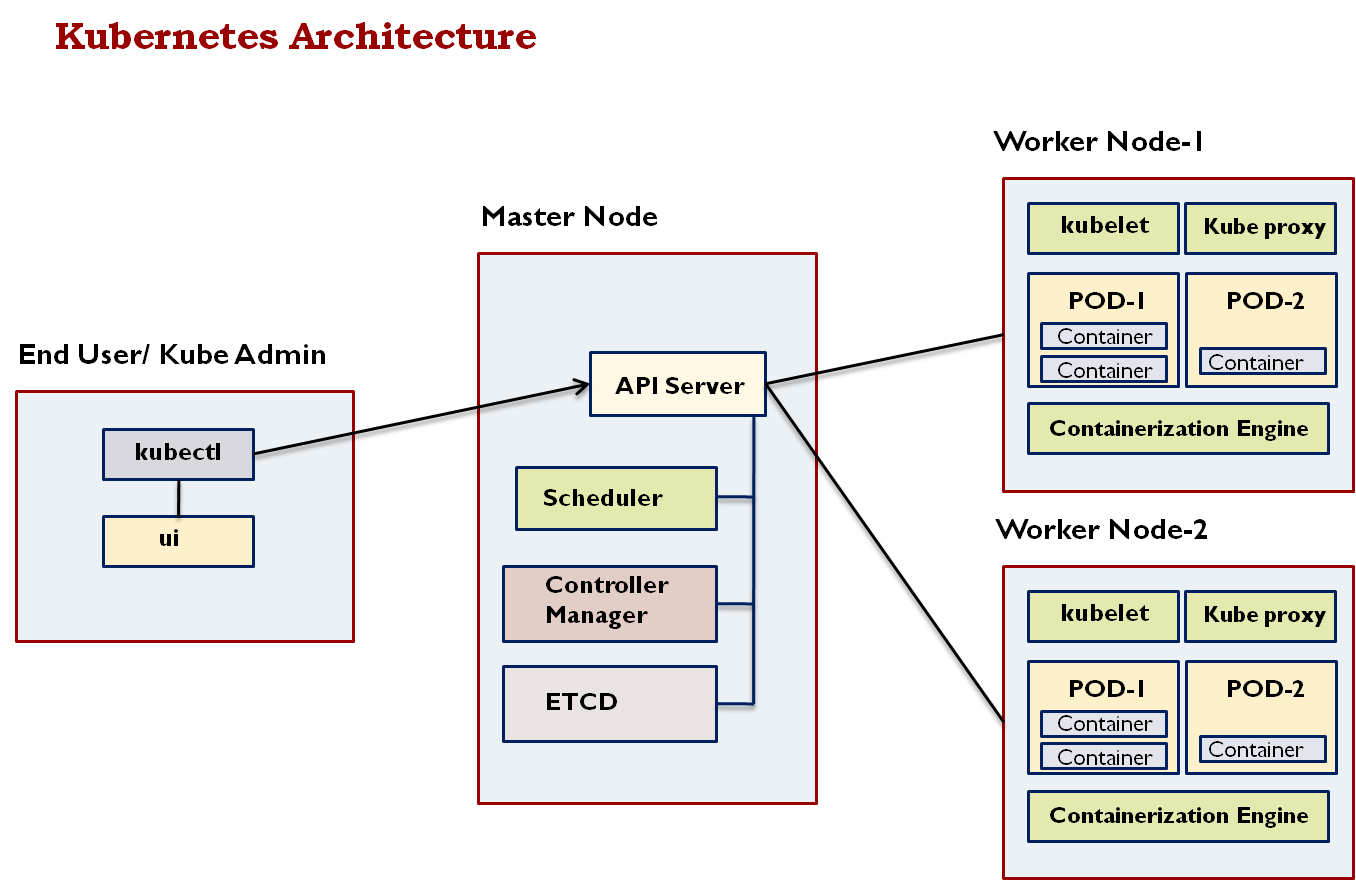
* In Microservice architecture the whole application is divided into small services.
* We will have multiple code base.
* We will have multiple repository.
* We will have multiple artifact for the project.
* Each service can be deploy independently.

**Challenges in Containerization Tool**

* Replication of components
* Auto-scaling
* Load balancing
* Rolling updates
* Logging across components
* Monitoring

**Kubernetes**

* Kubernetes is a container orchestration tool.



**End User**

* User will interact with master node using kubectl tool.
* User can also use UI interface like Kubernetes dashboard, GKE console to interact with master node.

**API Server**

* API server acts as an interface between kubectl and Kubernetes components like pods, replication controller, replication sets etc.

**ETCD**

* ETCD stores all the cluster data in key-value format.
* It stores following information:
  + How many nodes
  + How many containers
  + How many pods
  + Etc.

**Scheduler**

* Scheduler will schedule pods on the nodes with the help of kubelet.

**Controller Manager**

* Controller manager’s rune controllers in the background.
* Responsible for below activities
  + Managing Replication
  + Networking Configuration
  + End point

**Kubelet**

* Kubelet ensure that containers are running healthy in the POD.
* Kubelet will interact with containerization engine to create a container in the POD.

**Kube Proxy**

Kube proxy will maintain network rules on the PODS.