Pod in Detail:-

- An abstraction layer that represents the group of containers that are deployed together on the same Host/Node.

- It is used to manage the applications in the Kubernetes.

- A wrap layer for the deployed applications.

- In one Pod multiple containers can also been ran.

Life Cycle of Pod:-

- pod.yml - description of the pod details

- API server - will gets the pod.yml through the client [kubectl]

- Status of POD

1) Pending - After issuing the command to create pod, K8S will accept it & starts working with Scheduler allocate the resource.

2) Running - As per the Scheduler suggestion, the node will be allocated and container starts to creates by pulling the image from DockerHub.

3) Succeeded - Successfuly container getting terminated.

4) Failed - Due to some reason, Application getting crashed and not started.

5) CrashLoopBackOff - Its a Retry status to restart the containers after a failure.

Networking in POD:-

Application to Application communication inside K8S can be by following methods,

- Container to Container communication --> Network Namespace

- Container from one Pod to Another Pod from same Node --> Network Bridge using Ethernet

- Container from One Node to Another Node --> Network Routing Table