**Container Registry System**

**Abstract:**

The title itself is self-explanatory in this use case we are creating a container system which accepts the client request to allocate a node with sufficient resources available to run the selected docker image as requested by the user. The container system up looks for a node within its cluster and selects one to install the docker image based on the usage of the resources like RAM usage, CPU process time the registry system calculates the price based on its price strategy and notifies the client.

**Detail Work Flow:**

The client logs in to the registry system with his credentials provided at the time of registration. Once the registration is successful the client gets redirected to the form which provides the details of the available docker images. Based on the docker image selected and the further resources needed like the CPU and RAM necessary.

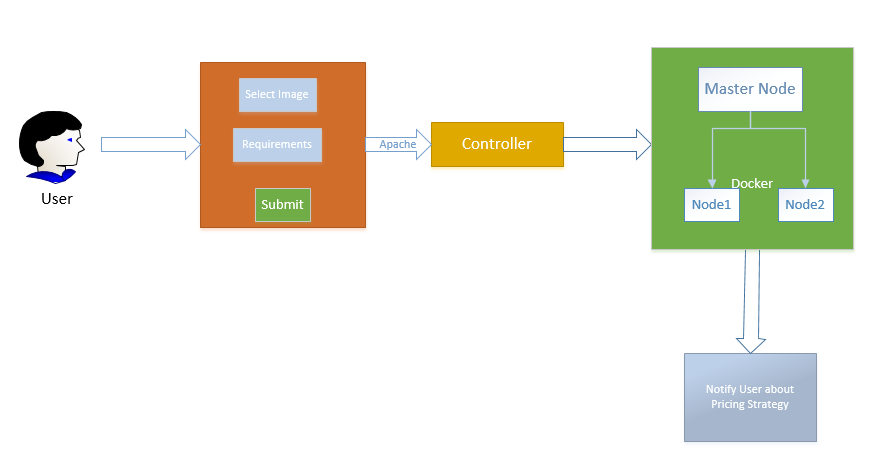
Based on the details furnished by the client the registry system selects a node which satisfies the client request from the cluster.

One the node selection is done the registry system makes sure that the node is active and healthy to install the appropriate docker image as selected by the client gets installed from the docker hub. After successful installation the registry system calculates the RAM usage and CPU process time with hold by the docker image to execute.

Based on these details the registry calculates the amount to be charged for the client for the resources usage.

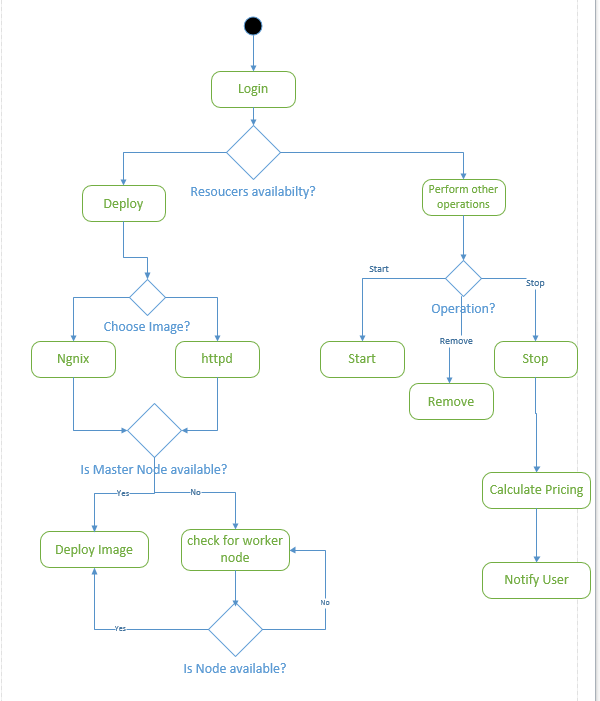
If the client desires to delete the service that is running he can request for “remove service” then the registry system deletes the service that is running on the node.

**System Architecture:**

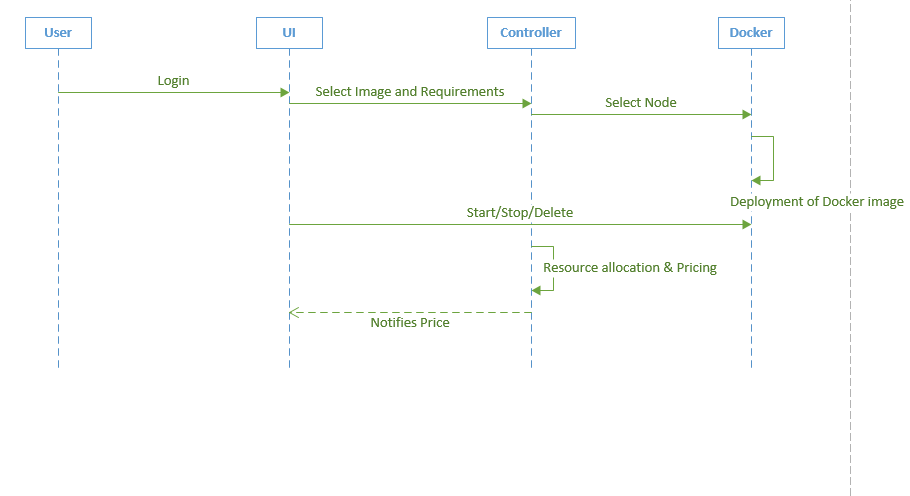


**Fig1: The figure shows the over-all flow of the container registry system**

**Activity Diagram:**



**Sequence Diagram**:



**Fig3: The figure shows the interaction among the different components involved in the system**