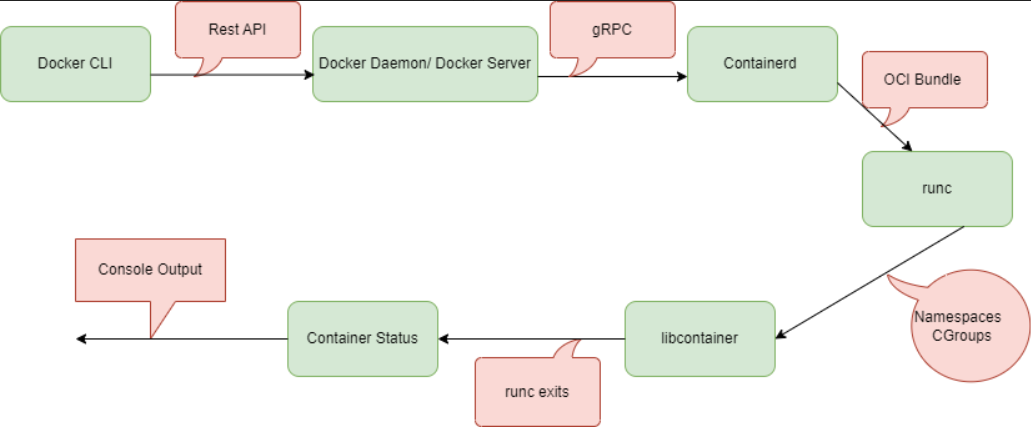
**DOCKER CONTAINER CREATION WORKFLOW**



* Docker CLI is used for executing commands like docker create, docker run. When we pass these commands in the CLI, they will pass the information to the docker daemon.
* Rest API calls act as a medium to send the information from docker CLI to docker daemon.Here the unix socket will use the rest api calls to transfer the information from the docker client and docker server.
* Docker server or Docker daemon calls the Containerd which is preinstalled on the machine along with docker to create containers.
* Docker daemon uses gRPC call (CRUD style of API) to transfer the information to Containerd.
* Containerd will first check whether the image is available in the local system or specified registry (docker hub, by default) if it is not available, then it will throw an error.
* Containerd will create the OCI bundle from docker image. Docker will create the image and that image information is given to the Containerd as an OCI bundle.
* Runc will create the container using that OCI bundle with the specified image. Runc interacts with the OS kernel to gather the requirements needed to create a container.
* Runc will call libcontainer to do those particular actions (like Kernel Namespaces and CGroups).
* By using Namespaces and CGroups containers will be created in the backend. Container will be created as a child process in the docker main process.
* Now Runc will exit from the task once the container starts, Finally this process will end when the container is running in the background.