Middleware Components – Execution Procedure

Pre-check:

Step 1: Ensure No Conflicting Applications Are Running

- Check if any related application processes are running in the background:
  $ top (or) $ ps -ef | grep <app_name>
- If any such processes are found, terminate them to avoid conflicts:
  $ kill -9 <PID>

Step 2: Check Host IP Address

- Find your active network interface IP:
  $ ifconfig

Step 3: Update IP in Configuration Files

- Modify the noted IP address in the following files:
  $ AppMain/ip_config.ini
  $ HMIC/ip_config

Step 4: Launch Middleware Containers

- Navigate to the middleware directory and start the script:
  $ cd mw_docker
  $ chmod 777 run_middleware.sh
  $ ./run_middleware.sh

Note: *If the docker containers are already Up and running, no need to run above script file*

Step 5: Verify Docker Container Status

- Check if the containers are up:
  $ docker ps –a
- If containers are not running, manage them using:
  $ docker start <container_name_or_id>
  $ docker stop <container_name_or_id>
  $ docker restart <container_name_or_id>

Example:

$ docker start velocitas-container

Startup Order Preference:

1. kuksa-databroker
2. mock-provider
3. velocitas-container
4. vpredictor

Note: *The startup order is important because the mock-provider publishes VSS signals to the kuksa-databroker, which must be running first, and then velocitas-container consumes those signals. The vpredictor should also start after the data broker is available to ensure it receives the required inputs.*

Step 6: Run the Applications

Once the middleware containers are running, start the application services as required.