### Test Automated Deployment

#### **4.2.4.1 Objective and Scope**

* **Objective**: Automate deployment and testing on automotive clusters, RTOS, and HMI.
* **Scope**: This automated deployment and testing setup is intended for the automotive cluster and RTOS model, specifically for the regression test suite. Testing includes.
  + **Functional Testing**: Conducted through screen comparison and widget validation.
  + **Communication Protocols**: CAN and LIN protocols are used to send signals via the GM Vehicle Simulator.

#### **4.2.4.2 Environment Setup**

* Refer section 5

#### **4.2.4.3 Jenkins Configurations**

* **Node Creation**: Start by creating a node for the systems.
* **Job Creation**: Set up a job using a freestyle pipeline in Jenkins.
* **Job Configuration**: Configure the job and schedule it is using the task scheduler.
* Scripting:
  + Use Groovy scripts in Jenkins to define and automate tasks.
  + The scripts will leverage the task scheduler to execute multiple stages in the process.

#### **4.2.4.4 Build & Deployment Process**

* A Jenkins job will trigger a task scheduler that manages all related stages
  + Stage One
* Pull the build from the repository.
* Launch the VIP application.
* Rename required files (e.g., Contains.txt).
* Complete the VIP process, concluding Stage One.
* Stage Two
  + Execute QFil tasks.
  + Upon completion, proceed to Stage Three.
* Stage Three
  + Perform tasks using Tera Term.
  + After Tera Term tasks are complete, move to Stage Four.
* Stage Four
* Launch automation scripts to execute test scenarios.
* Once all scenarios are completed, Jenkins will compile and display results on the current job's dashboard.
* **Additional Features:**
  + **Status Monitoring with Light Bulb Indicators**
  + **Red**: Indicates a stage failure.
  + **Amber**: Shows a stage in progress.
  + **Green**: Confirms all stages have passed successfully.