Testing Overview

EdgeX testing is focused on ensuring that the functional aspects of EdgeX work correctly. This level of testing includes a number of automated test suites for unit, integration, black-box, API testing and validation. Also, EdgeX has been tested on TAF, which stands for Test Automation Framework. To monitor the stability and efficiency of EdgeX, a Jenkins pipeline has been built to trigger jobs automatically. The testing processes are listed below.

PR merge validation

When developers make changes and open PRs on EdgeX, they have to go through a different validation process depending on the type of repository.

- Go modules
  - Unit test

> ### Where is go module?
>
> https://github.com/edgexfoundry
>
> Type "go-mod" in the search field. All repositories whose titles contain "go-mod" are go-modules.
> Once developers open PRs in a go-mod repository, a unit test is triggered to check if it is available.

- Service
  - Unit test
  - Image build validation

> ### Where is service?
>
> https://github.com/edgexfoundry/edgex-go
>
> This repository contains the Go implementation of EdgeX Foundry microservices, such as core-data, core-metadata, support-notifications, etc.
> Once developers open PRs in the edgex-go, it will trigger unit test to check if it's fit to use and build dev images on nexus3.edgexfoundry.org to validate its usage.

- edgex-compose
  - smoke test by TAF

> ### Where is edgex-compose?
>
> https://github.com/edgexfoundry/edgex-compose
>
> Smoke-test will be introduced later.
> Once developers open PRs in the edgex-compose, a smoke-test will be triggered to check its availability.

Functional-test

- Purpose: Test each feature of EdgeX by providing the appropriate input and validating the output against the requirements.
- Jenkins schedule: **Daily** run on edgex-taf-pipeline

> ### How to run functional-test on local?
>
> https://github.com/edgexfoundry/edgex-taf/blob/main/docs/run-tests-on-local.md

Integration-test

- Purpose: Check data communication among different software modules.
- Jenkins schedule: **Daily** run on edgex-taf-pipelines

> ### How to run integration-test on local?
>
> https://github.com/edgexfoundry/edgex-taf/blob/main/docs/run-tests-on-local.md

Performance-test

- Purpose: Monitor memory usage/ startup time/ CPU usage/ response time/ event exported time of EdgeX
- Jenkins schedule: **Weekly** run on edgex-taf-pipelines
- Note: When a new version of EdgeX is released, we will run performance-test manually on x86_64 and arm64 machines to update the reports in here.

ⓘ **How to run performance-test on local?**

---

https://github.com/edgexfoundry/edgex-taf/blob/main/docs/run-performance-metrics-collection-on-local.md

---

Smoke-test

- Purpose: Developers will select tests that validate key EdgeX features to run smoke-test.
- Jenkins schedule: Only when **edgex-compose PR is opened** will smoke-test start running.

ⓘ **How to run smoke-test on local?**

First, where is smoke-test?

Example**:** https://github.com/edgexfoundry/edgex-taf/blob/f7c5ac7bf88c2cca21e5cb485ee655b14a2023d7/TAF/testScenarios/functionalTest/V2-API/core-metadata/deviceprofile/GET-Positive.robot#L15
In this example, only does **ProfileGET001 - Query all device profiles** have the **SmokeTest** tag so it will run on this testcase only.

Second, how to run smoke-test on local?

Example**:** https://github.com/edgexfoundry/edgex-taf-pipelines/blob/5cb7fa71727f8b6c6470626ed0394a449b223b2c/runSmokeTestScripts.groovy#L32-L51
When running tests locally with docker command, remember to specify **--include SmokeTest** to point to those testcases with the **SmokeTest** tag.

Reference

- Jenkins-edgexfoundry
- GitHub - edgexfoundry/edgex-taf
- GitHub - edgexfoundry/edgex-taf-pipelines