**Functional Test Report for Inji certify v0.11.0**

**09-04-2025**

# 

[Testing Scope 3](#_Toc195603918)

[Test Approach 3](#_Toc195603919)

[Verified configuration 4](#_Toc195603920)

[Feature Health 4](#_Toc195603921)

[Test execution statistics 5](#_Toc195603922)

[Functional test results 5](#_Toc195603923)

[Detailed Test Metrics 6](#_Toc195603924)

[Tested with Components 7](#_Toc195603925)

# 

# Testing Scope

The scope of testing is to verify fitment to the specification from the perspective of

* Functionality
* Deployability
* Configurability
* Customizability

Verification is performed not only from the end user perspective but also from the System Integrator (SI) point of view. Hence Configurability and Extensibility of the software is also assessed. This ensures readiness of software for use in multiple countries. Since MOSIP is an “API First” product platform.

Testing scope has been focused on the following features:

* Inji certify Docker compose testing (Removal of artifactory dependency – Data provider CSV plugin and mdoc plugin) which covered Farmer Use case
* Esignet compatibility with 1.5.1 version and 1.4.1 version
* Insurance, Mock (Issuance and Data provider (Postgres plugin)), MosipId which covered Land registry, school use case also
* Integration with Inji Web and Inji verify
* Keycloak integration with Injicertify
* Ed25519 signing of VC request – Scope from automation
* Support of ECC k1 verification signature

# Test Approach

Persona based approach has been adopted to perform the IV&V, by simulating test scenarios that resemble real-time implementation.

A Persona is a fictional character/user profile created to represent a user type that might use a product/or a service in a similar way. Persona based testing is a software testing technique that puts software testers in the customer's shoes, assesses their needs from the software and thereby determines use cases/scenarios that the customers will execute. The persona needs may be addressed through any of the following.

* Functionality
* Deployability
* Configurability
* Customizability

The verification methods may differ based on how the need was addressed.

# Verified configuration

Verification is performed on configurations as mentioned below

* Default configuration
  + English

# Feature Health

A graph with blue and orange lines

AI-generated content may be incorrect.

**Note:**

1. For keycloak integration – keycloak is pointing to dev2 env
2. For Esignet compatibility

* eSignet 1.5.1 from dev-int-inji env
* eSignet 1.4.1 from released env

# Test execution statistics

## Functional test results

Below are the test metrics by performing functional testing. The process followed was black box testing which based its test cases on the specifications of the software component under test. Functional test was performed in combination of individual module testing as well as integration testing. Test data were prepared in line with the user stories. Expected results were monitored by examining the user interface. The coverage includes GUI testing, System testing, and End-To-End flows across multiple configurations. The testing cycle included simulation of multiple identity schema and respective UI schema configurations.

|  |  |  |  |
| --- | --- | --- | --- |
| **Total** | **Passed** | **Failed** | **NA** |
| 643 | 612 | 28 | 2 |
| Test Rate: 99%, With Pass Rate: 95% and Fail Rate: 4.35% | | | |

Automation Statistics

* Sunbird use case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Total** | **Passed** | **Failed** | **Ignored** | **Known issues** |
| 225 | 61 | 0 | 155 | 9 |
| Test Rate: 100%, With Pass Rate: 96% and Fail Rate: 4% | | | | |

* Mock Use case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Total** | **Passed** | **Failed** | **Ignored** | **Known issues** |
| 225 | 47 | 0 | 169 | 9 |
| Test Rate: 100%, With Pass Rate: 96% and Fail Rate: 4% | | | | |

* Mock (Data provider) Use case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Total** | **Passed** | **Failed** | **Ignored** | **Known Issues** |
| 225 | 79 | 0 | 137 | 9 |
| Test Rate: 100%, With Pass Rate: 96% and Fail Rate: 4% | | | | |

Note- Ignored scenarios are Not related to particular use case and 9 scenarios are known issues can be tracked from **INJICERT-681**, **Mosip id** use case is being ignored from Automation for the current release.

## Detailed Test Metrics

Below are the detailed test metrics by performing manual/automation testing. The project metrics are derived from Defect density, Test coverage, Test execution coverage, test tracking and efficiency.

The various metrics that assist in test tracking and efficiency are as follows:

* Passed Test Cases Coverage: It measures the percentage of passed test cases. (Number of tests passed / Total number of tests executed) x 100
* Failed Test Case Coverage: It measures the percentage of all the failed test cases. (Number of failed tests / Total number of test cases executed) x 100

# Tested with Components

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Module/Repo** | **Image** | **POM version** | **Dependent artifactID** | **Comments** |
| Inji-certify-mosipid | mosipqa/inji-certify-with-plugins:0.11.x |  | Digital-credential-plugin - 0.4.0 |  |
| Inji-certify-mock | mosipqa/inji-certify-with-plugins:0.11.x |  | Digital-credential-plugin - 0.4.0 |  |
| Inji-certify-Insurance | mosipqa/inji-certify-with-plugins:0.11.x |  | Digital-credential-plugin - 0.4.0 |  |
| Inji-certify- landregistry | mosipqa/inji-certify-with-plugins:0.11.x |  | Digital-credential-plugin - 0.4.0 |  |
| Inji-certify- academic | mosipqa/inji-certify-with-plugins:0.11.x |  | Digital-credential-plugin - 0.4.0 |  |
| Mdoc-mdl | mosipqa/inji-certify-with-plugins:0.11.x |  | Digital-credential-plugin - 0.4.0 |  |
| Inji-config | Releasing from release-0.8.x branch |  | Digital-credential-plugin - 0.4.0 | <https://github.com/mosip/inji-config/tree/release-0.8.x> |
| Keymanager |  | 1.3.0-beta.2 |  | Will be released as 1.3.0-beta.2 |
| eSignet | eSignet-1.4.1  eSignet-1.5.1 |  |  | 1.5.1 eSignet from dev2 env  1.4.1 eSignet from released env |