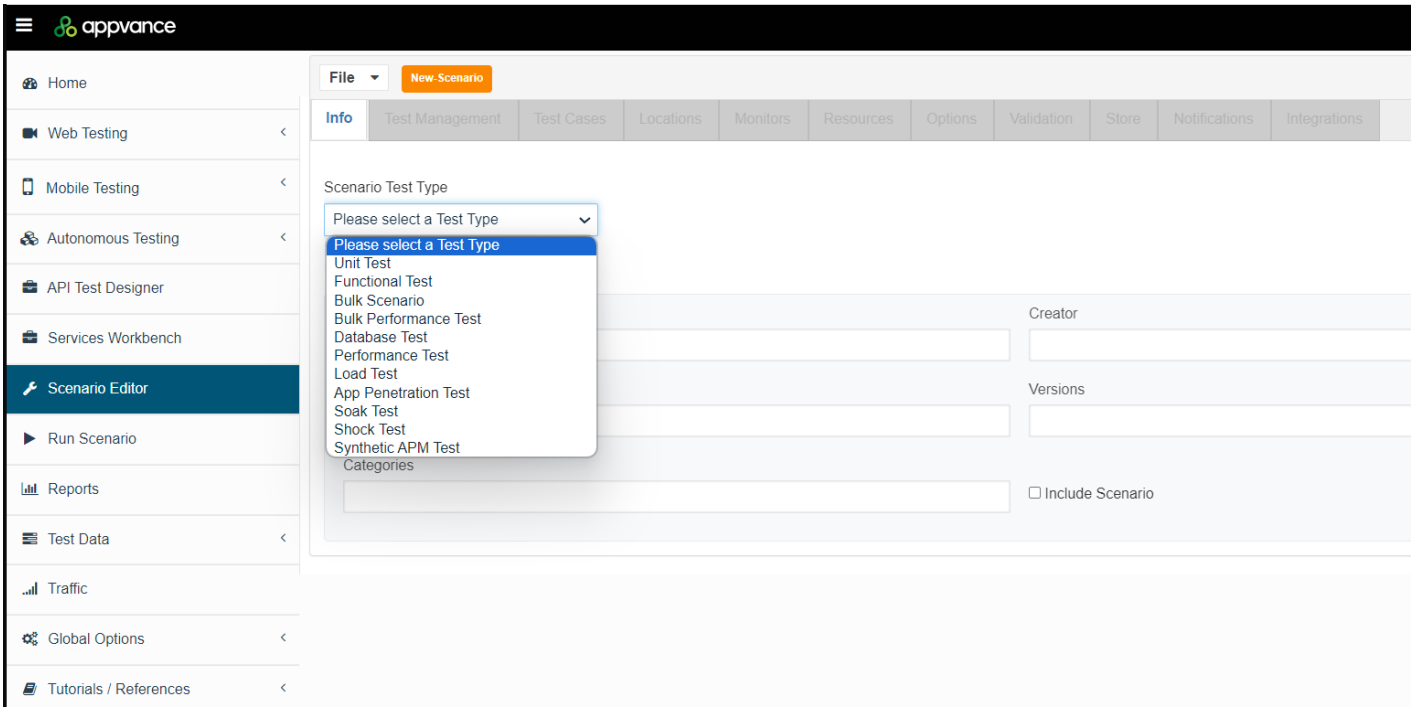
Types of Tests

Types of Tests



The following scenario test types are available:

- Unit Test
- Functional Test
- Bulk Scenario
- Bulk Performance
- Database Test
- Performance Test
- Load Test
- App Penetration Test
- Soak Test
- Shock Test
- Synthetic APM Test

Unit Test

Unit tests are used to test small, discreet parts of an application. Utilize your favorite domain-specific language (Selenium, C#, Java) at the API level or UX level.

Below is the list of Script types that a user can select in the test case panel when working with Unit Testing.

- Appium TestNG
- Command Line
- Command Line Resource
- Java
- JRuby
- Service Suite (Services Workbench)
- TestNG
- API Test

Functional Testing

Also known as a regression test or smoke test, is used to ensure that the existing functionality of an application works as new features are added. Run all your functional tests in series or parallel. Record tests in minutes and parameterizes them without any coding needed.

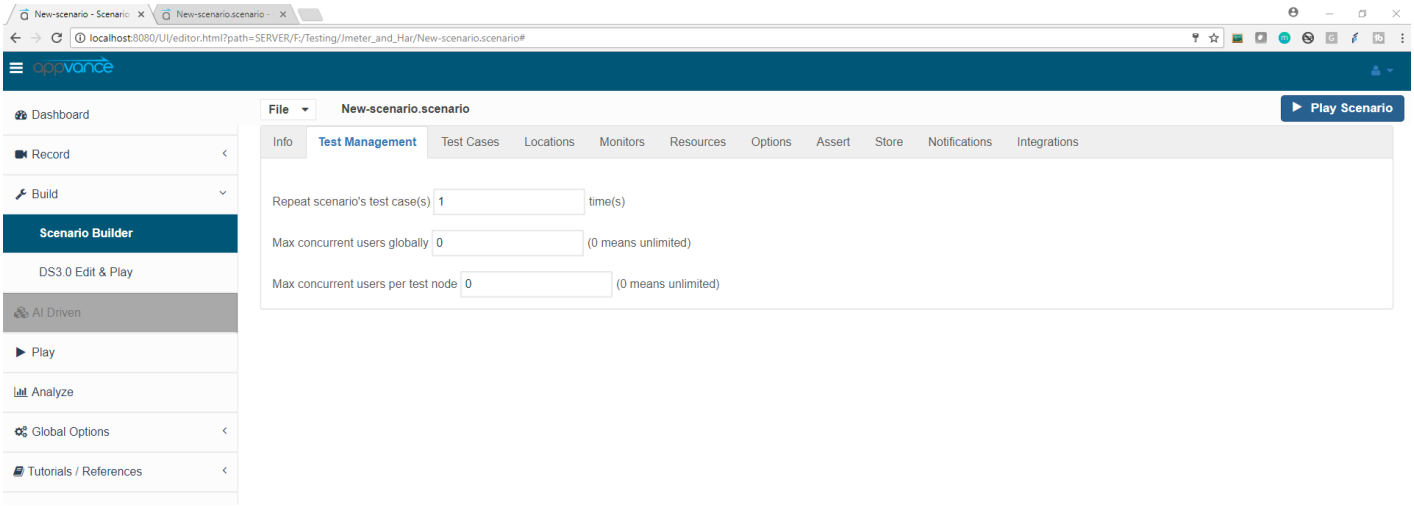
Below is the list of Script types that a user can select in the test case panel when working with Functional Testing.

- Appium-testNG
- Test Designer
- Mobile Designer
- Appvance Java Technology
- Command Line
- Command Line Resource
- Groovy
- HAR
- Java
- JMeter
- JRuby
- Selenium
- Service Suite (Service Workbench)
- SoapUI
- SQL
- TestNG
- API Test

SoapUI support was removed in release 5.2.0 of AIQ.

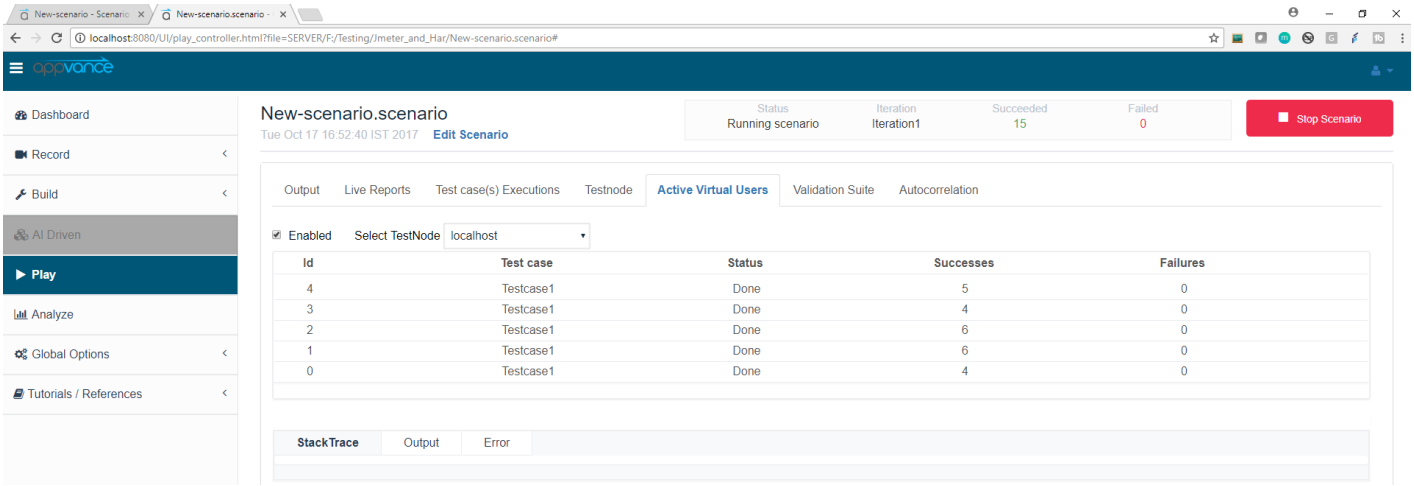
For a **Functional Test**, the user can choose how many times to **Repeat this test-case** option from the **Test Management** Panel

Providing a value in "Repeat this use case", the test case that is defined will run the specified number of times sequentially.



With functional testing, users also have options to control how many test cases to run concurrently with the options "Max Concurrent users globally" and "Max concurrent users per test node".

Assume a scenario is created with 3 test cases (Appvance Designer scripts) and "Max concurrent users per test node" is set to 2, After saving and clicking Play scenario, While running this scenario navigate to the Active Virtual Users tab on the controller window and select Enabled check-box, user can see two users executing and 1 user in waiting for status.



Also, users can set a maximum number of concurrent executions considering all test nodes using the "Max concurrent users globally" option.

NOTE: Normally all test cases in a functional test will run concurrently.

Database Test

Database tests enable you to run SQL scripts to validate your DB with every build.

Please refer to this chapter for more information: Database Testing in Appvance IQ.

Load and Performance Test

Performance Tests: Simulate millions of users with real use cases from beginning to end (from the UX to the back end). Identify performance issues including client-side code. Runs using use cases already created for functional tests.

Load Tests: Simulate users, usually at the API level, over varying and increasing loads. Ability to use existing frameworks like SoapUI, JMeter, etc.

Load and Performance tests operate the test cases in one or more virtual users as many times as possible in a given amount of test time or with Different Completion Criteria available. By running a load test at several levels of virtual users, AIQ identifies the Scalability Index of your application's scalability.

Choose one or more virtual user levels to operate the test. Click Add virtual user level to add a level and Define an Iteration.

Below are some of the bookkeeping activities for your test scenario, none of these fields are required fields.

Author Name - The name of the author of the test scenario details can be saved, this detail is also shown in the CSV iteration summary report.

Links - Any link details can be mentioned here.

Versions - Can be used for version control.

Categories - This is a project category, once entered shall be shown on Analyze and Play pages

Include Scenario - Include Scenario is a useful feature. You can have one in include per each possible database you want to use, or per each set of predefined locations or cloud configurations.

You can have a predefined set of scenario files that typically have only details of Locations and Logs.

When you want to run tests in so and so locations and so and so Database details, all those can be created as individual files and can be included here and those scenarios shall be run in those mentioned locations and logs will be stored in the mentioned databases.

Example Include Scenario file below:

```
<testscenario xmlns="www.pushtotest.com/tm5"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="www.pushtotest.com/XSD/testscenario.xsd" version="3">
<testnodes>
<node name="cloudhost" location="http://localhost:8080/TMServer/ws/TestNode"/>
</testnodes>
<logs>
<log type="database">
<opt name="hibernate.connection.username" value="appvance"/>
<opt name="hibernate.connection.password" value="appvance"/>
<opt name="hibernate.connection.url"
value="jdbc:oracle:thin:@smoketestutp30oracle.cjzzvczxudhc.us-east-
1.rds.amazonaws.com:1521:ORCL"/>
<opt name="hibernate.default_schema" value="utp"/>
<opt name="hibernate.connection.driver_class" value="oracle.jdbc.OracleDriver"/>
</log>
</logs>
</testscenario>
```

Revision #2
Created 2026-07-31 14:12:51 UTC by Admin
Updated 2026-07-31 14:15:35 UTC by Admin