# .NET TCT Tester User Guide

Version 1.0, for Tizen 6.0

# Contents

# 1 Introduction

This document provides comprehensive information about .NET TCT Test Set, including the following: Overview, Installation, Usage and Test Environment setup etc.

## 1.1 What is .NET TCT?

TCT is short for the Tizen Compliance Tests, which validates platform compatibility for Tizen. .NET TCT consists of .NET TCT Manager (UI tool), .NET TCT Shell (console tool), Testkit-lite (backend test runner), and .NET TCT Behavior Test Tool (device behavior checker).

## 1.2 How does .NET TCT Work?

.NET TCT has three main components:

**a.** **.NET TCT Manager** is a java GUI tool that runs on the host machine, allow users to create a test execution plan, trigger the test execution, and view the test report. By supporting both automated and manual .NET API testing, .NET TCT Manager makes it much easier for users to conduct TCT tests and enter hardware capability information.

**b.** **.NET TCT Shell** is a lightweight console tool that runs on the host machine, allowing users to debug single failed case, or trigger TCT testing with an existing test plan by specifying a test suite list or test case ID. Test suites are executed on target devices under the management of Testkit-lite.

**c.** **Testkit-lite** is a back-end test runner that communicates with Testkit-stub through the Smart Development Bridge (SDB).

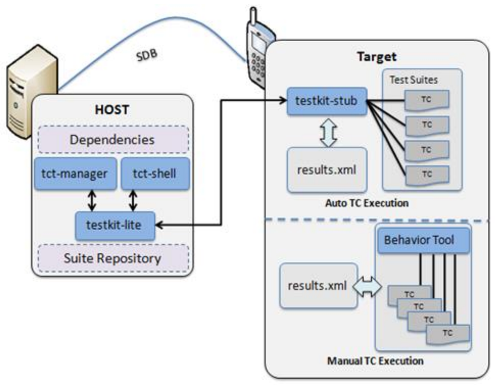


Figure 1-1: .NET TCT Workflow

# 2 Test Environment Setup

## 2.1 Symbols and Abbreviations

|  |  |
|---|---|
| TC | - Test Case |
| TCT | - Tizen Compliance Test |
| SDB | - Smart Development Bridge |
| <name> | - Mandatory argument |
| [name] | - Optional argument |
| $ (in shell command) | - Indicates the beginning of a command |
| # (in shell command) | – In long commands, the backslash character ensures that newline character is ignored (if you join consecutive lines, please remove unnecessary backslashes) |

## 2.2 Hardware Requirements

Make sure these items in place are ready before starting:

a. PC or Laptop that will work as host on which TCT-Manager will be installed

b. Tizen device that will work as target on which TCs will be executed

c. USB Cable for connecting device to host

## 2.3 Software Requirements

a. Install 32 or 64 bit Ubuntu OS.

b. Install JDK 1.6 or newer version on Linux PC.

c. Install Tizen 5.0 SDK on Linux PC for SDB connection.

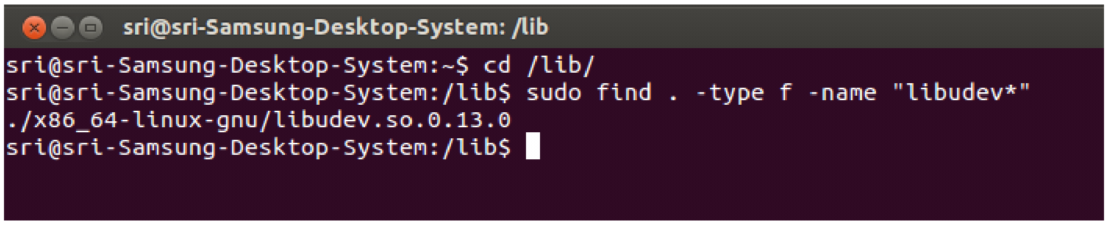
d. These packages should be installed before installing TCT-Manager

```
$ sudo apt-get install rpm2cpio
$ sudo apt-get install tree
$ sudo apt-get install python-pip
$ sudo pip install requests
$ sudo apt-get install python-dbus
$ sudo apt-get install python-support

$ sudo apt-get install python-requests

$ sudo apt-get install python-setuptools
```

e. libudev1 or libudev-dev package should be installed for SDB.

f. First find the library ‘libudev’ installation location using command:~$ cd /lib/

```
$ find . –type f – name "libudev*"
```



```
sri@sri-Samsung-Desktop-System: /lib
sri@sri-Samsung-Desktop-System:~$ cd /lib/
sri@sri-Samsung-Desktop-System:/lib$ sudo find . -type f -name "libudev*"
./x86_64-linux-gnu/libudev.so.0.13.0
sri@sri-Samsung-Desktop-System:/lib$
```

Figure 2-1: Getting Location of libudev

If the package is not properly linked, use the following command:

```
$ sudo ln –s /lib/<installation-folder>/libudev.so.<version> /lib/<installation-
folder>/libudev.so.0

e.g. $ sudo ln –s /lib/i386-linux-gnu/libudev.so.0.13.0 /lib/i386-linux-gnu/libudev.so.0
```

## 2.4 Getting TCT-binary and TCT-manager

### 2.4.1 Download TCT Binary

a. Download .NET TCT release from http://download.tizen.org/tct/4.0/Csharp_TCT/csharp-tct_4.0_xxx.tar.gz to your host machine.

b. Reboot your device to make sure the device environment is clean.

c. To burn the new Tizen image to the target device (refer to https://source.tizen.org/documentation/reference/flash-device) and make sure the host machine is well connected to the target device through USB.

d. The device need enable ‘USB debugging’ in setting. If you already have SDB installed on your host machine, you can check the device connected with sdb command:

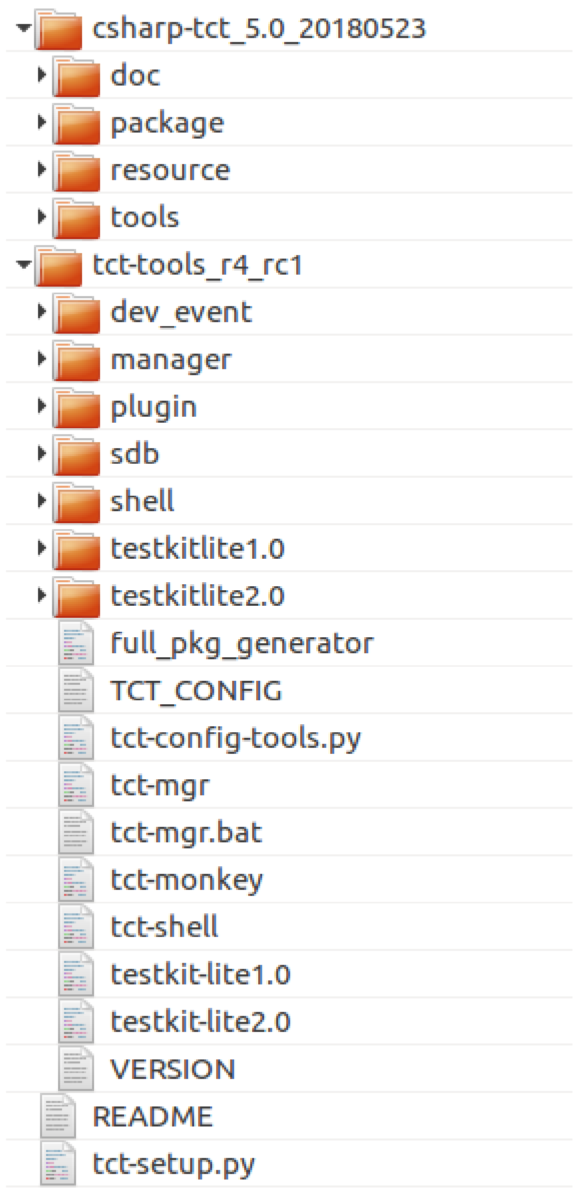
```
$ sdb devices
```

The key packages of .NET TCT on Host: .NET TCT Manager, .NET TCT Shell, Testkit-lite.

The key packages of .NET TCT on Device: Testkit-stub.

## 2.4.2 Folder Structure

Un-compress .NET TCT tar ball to local path on Ubuntu Host. You will find folder structure like below:



```
csharp-tct_5.0_20180523
    doc
    package
    resource
    tools
tct-tools_r4_rc1
    dev_event
    manager
    plugin
    sdb
    shell
    testkitlite1.0
    testkitlite2.0
    full_pkg_generator
    TCT_CONFIG
    tct-config-tools.py
    tct-mgr
    tct-mgr.bat
    tct-monkey
    tct-shell
    testkit-lite1.0
    testkit-lite2.0
    VERSION
README
tct-setup.py
```

**Figure 2-2: .Net TCT-Manager folder structure**

The following table describes the folders contents.

| **Folder** | **Description** |
|---|---|
| doc | Packaging & User Guid |
| package | All the packages to be tested in device |
| resource | Different kind of helper applications and utilities required for device to execute .NET TCT |
| tools | Installation scripts to install host |
| tct-tools | Contains tct tools and testkitlite etc. |
| | |

**Table 1-1: TCT-Manager folders**

## 2.5 Installation .NET TCT Test Tool

Un-compress .NET TCT tar ball to local path on Ubuntu Host

You can get help information of the config script firstly.

```
$ cd /path/to/<TCT_pkg_folder>/<TCT_pkg_folder>/tools
$ python ./tct-config-host.py -h
$ python ./tct-config-device.py -h
```

a. Deployment on Host

```
$ cd /path/to/<TCT_pkg_folder>
$ sudo python ./tct-setup.py
```

b. Deployment on Device

```
$ cd /path/to/<TCT_pkg_folder>/<TCT_pkg_folder>/tools
$ sudo python ./tct-config-device.py
```

# 3 How to Execute TCT

## 3.1 Using .NET TCT Manager

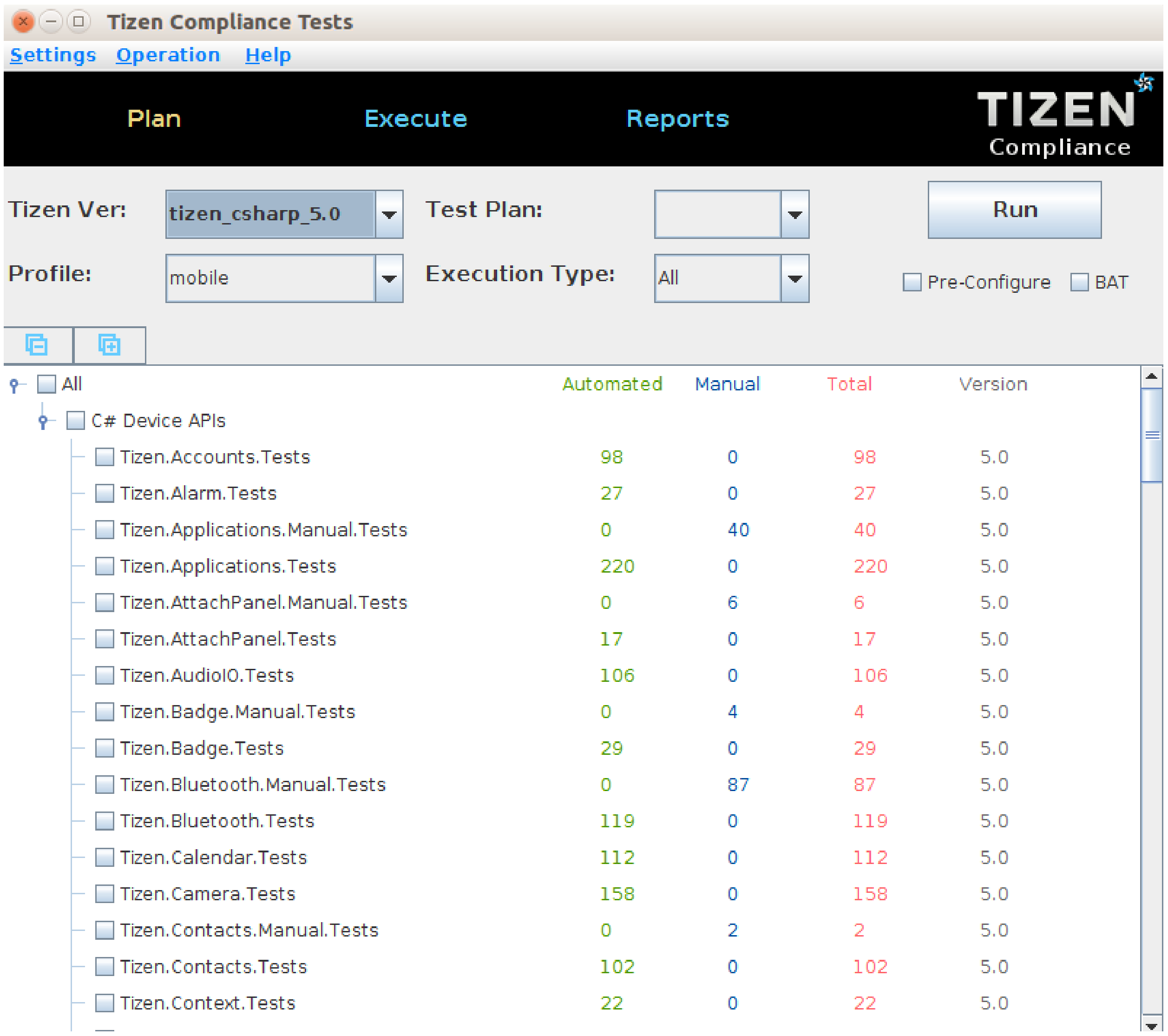
### 3.1.1 Launching .NET TCT Manager on Host Machine for Automated TC

Launch the .NET TCT Manager by shell command:

**On Ubuntu Host:**

```
$ tct-mgr
```

When the below screen is shown, .NET TCT Manager is launched successfully as shown in Figure 3-1-1.



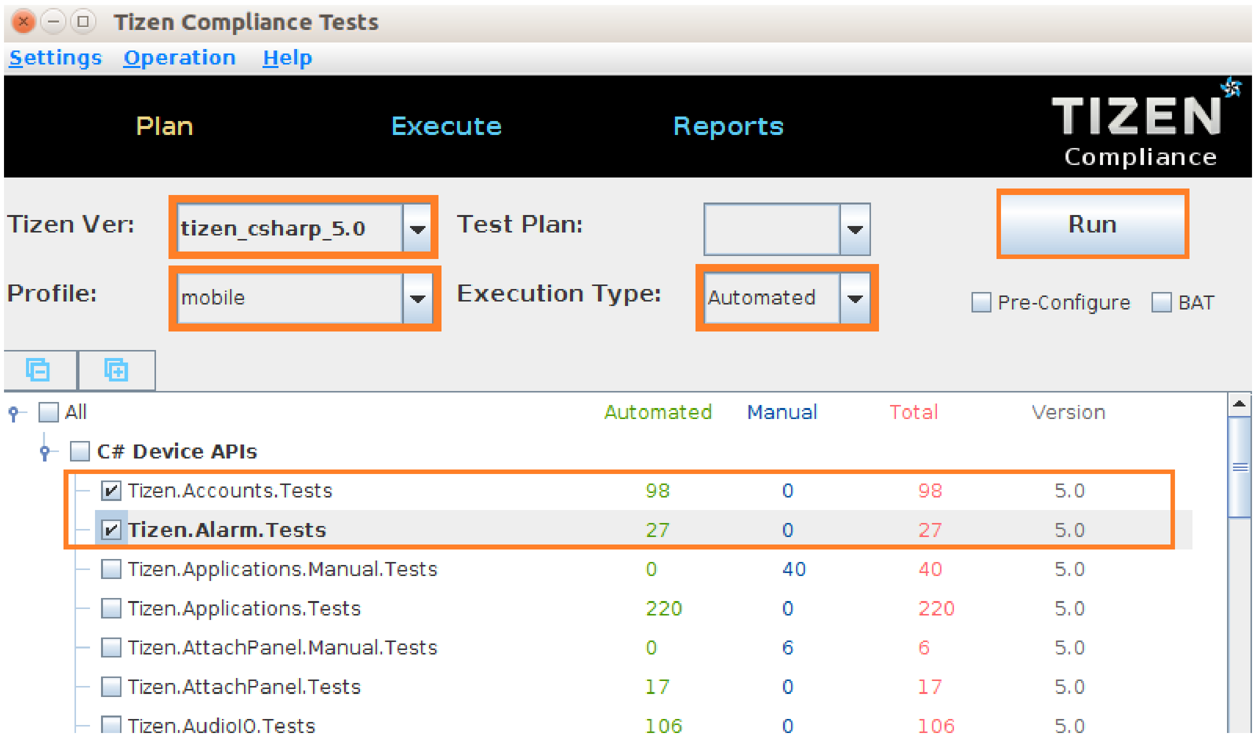
Tizen Ver: tizen_csharp_5.0
Test Plan:
Profile: mobile
Execution Type: All

| | Automated | Manual | Total | Version |
|---|---|---|---|---|
| All | | | | |
| C# Device APIs | | | | |
| Tizen.Accounts.Tests | 98 | 0 | 98 | 5.0 |
| Tizen.Alarm.Tests | 27 | 0 | 27 | 5.0 |
| Tizen.Applications.Manual.Tests | 0 | 40 | 40 | 5.0 |
| Tizen.Applications.Tests | 220 | 0 | 220 | 5.0 |
| Tizen.AttachPanel.Manual.Tests | 0 | 6 | 6 | 5.0 |
| Tizen.AttachPanel.Tests | 17 | 0 | 17 | 5.0 |
| Tizen.AudioIO.Tests | 106 | 0 | 106 | 5.0 |
| Tizen.Badge.Manual.Tests | 0 | 4 | 4 | 5.0 |
| Tizen.Badge.Tests | 29 | 0 | 29 | 5.0 |
| Tizen.Bluetooth.Manual.Tests | 0 | 87 | 87 | 5.0 |
| Tizen.Bluetooth.Tests | 119 | 0 | 119 | 5.0 |
| Tizen.Calendar.Tests | 112 | 0 | 112 | 5.0 |
| Tizen.Camera.Tests | 158 | 0 | 158 | 5.0 |
| Tizen.Contacts.Manual.Tests | 0 | 2 | 2 | 5.0 |
| Tizen.Contacts.Tests | 102 | 0 | 102 | 5.0 |
| Tizen.Context.Tests | 22 | 0 | 22 | 5.0 |

Figure 3-1-1: .NET TCT Manager Plan Page

## 3.1.2 Creating and Executing a Test Plan

On the Plan page, select a test profile to switch the test plan set supported for different profiles (mobile, tv, wearable). Then select an execution type of testcases (automated, manual). This time we select automated execution type. Select some packages or all packages for auto test plan execution. Refer figure 3-1-2a



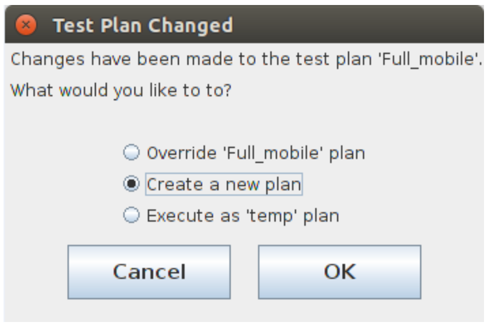
Tizen Ver: tizen_csharp_5.0
Test Plan:
Profile: mobile
Execution Type: Automated

| | Automated | Manual | Total | Version |
|---|---|---|---|---|
| All | | | | |
| C# Device APIs | | | | |
| ☑ Tizen.Accounts.Tests | 98 | 0 | 98 | 5.0 |
| ☑ Tizen.Alarm.Tests | 27 | 0 | 27 | 5.0 |
| ☐ Tizen.Applications.Manual.Tests | 0 | 40 | 40 | 5.0 |
| ☐ Tizen.Applications.Tests | 220 | 0 | 220 | 5.0 |
| ☐ Tizen.AttachPanel.Manual.Tests | 0 | 6 | 6 | 5.0 |
| ☐ Tizen.AttachPanel.Tests | 17 | 0 | 17 | 5.0 |
| ☐ Tizen.AudioIO.Tests | 106 | 0 | 106 | 5.0 |

Figure 3-1-2a: Selecting profile, execution type and packages

Now, Click on Run button in Figure 3-1-2b to create a test plan

A dialog will pop-up to guide to save a new plan as shown in Figure 3-1-2b, configure the device, and start to run the test.

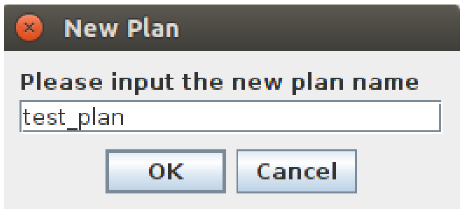


Changes have been made to the test plan 'Full_mobile'.
What would you like to to?

○ Override 'Full_mobile' plan
● Create a new plan
○ Execute as 'temp' plan

Figure 3-1-2b: Prompt for Creating Test Plan

Select the item “Create a new plan” and click **OK** button to save a new test plan as shown in Figure 3-1-2c

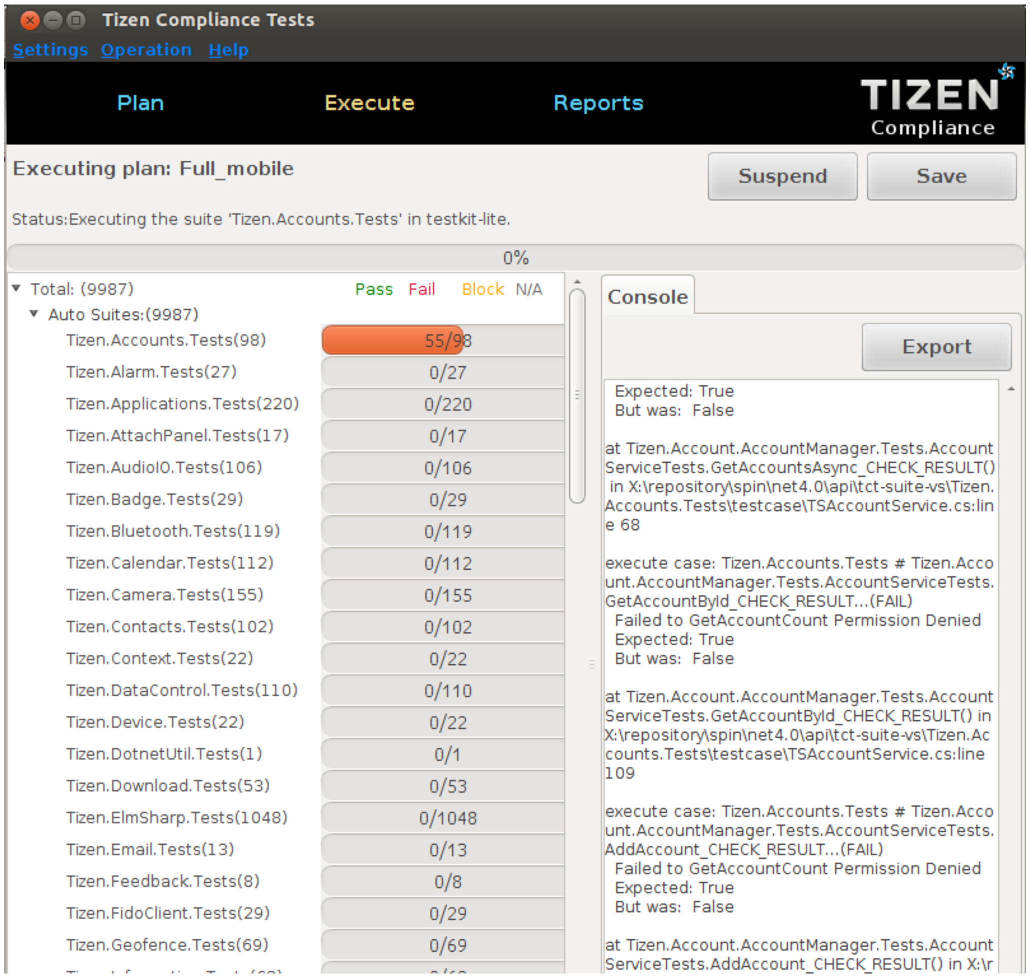


Please input the new plan name

test_plan

Figure 3-1-2c: Create a New Test Plan

After entering the new plan name, click **OK** button. The Plan UI will switch to Execution Page, and run selected test plan as shown in Figure 3-1-2d for Auto Test Plan. If pre-configurations are required to execute the testcases, then a pre-configuration popup will appear as Figure 3-1-3. Follow the precondition and click Continue to run the test plan.



Executing plan: Full_mobile

Status:Executing the suite 'Tizen.Accounts.Tests' in testkit-lite.

0%

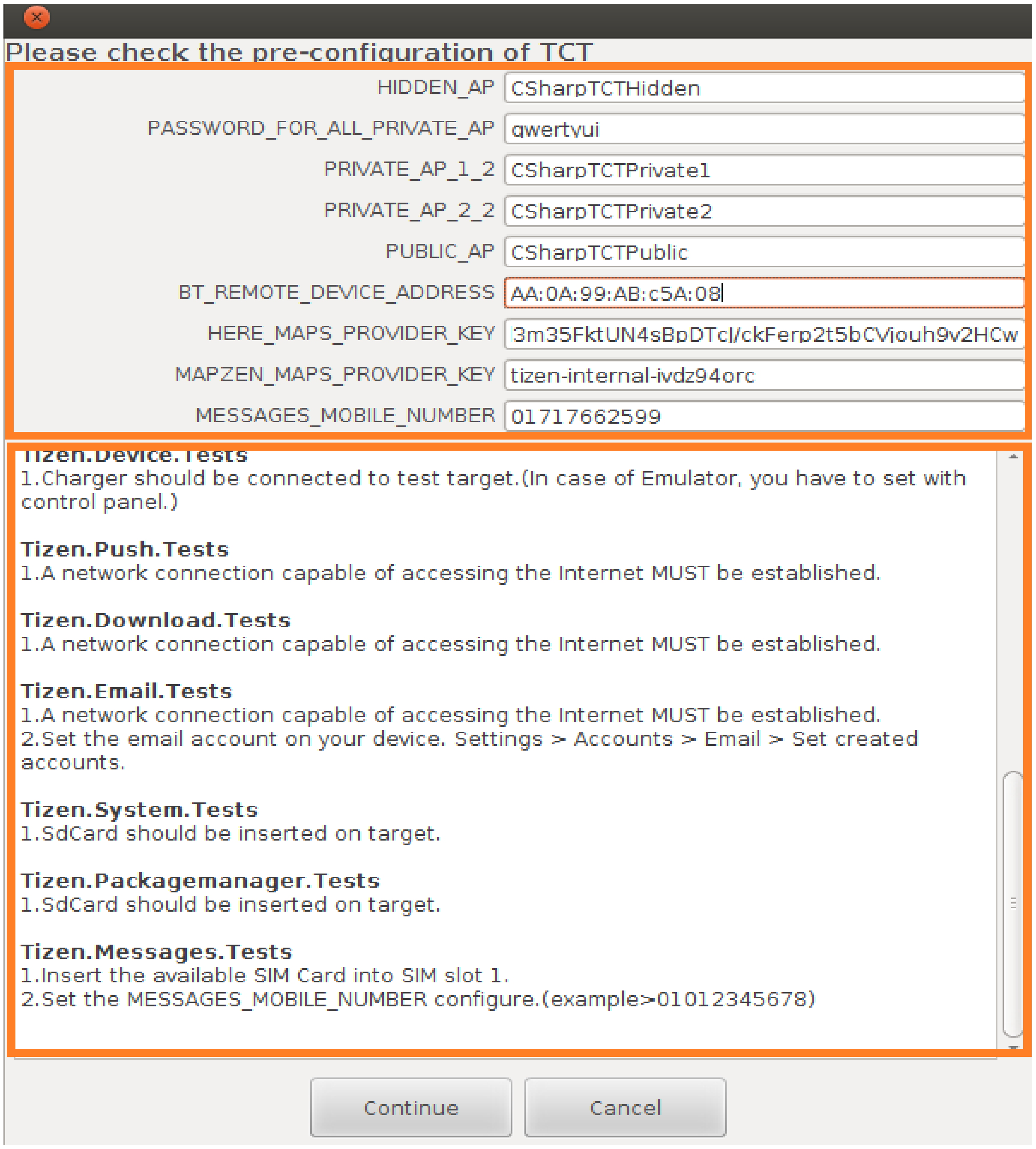
| Total: (9987) | Pass Fail Block N/A |
|---|---|
| Auto Suites:(9987) | |
| Tizen.Accounts.Tests(98) | 55/98 |
| Tizen.Alarm.Tests(27) | 0/27 |
| Tizen.Applications.Tests(220) | 0/220 |
| Tizen.AttachPanel.Tests(17) | 0/17 |
| Tizen.AudioIO.Tests(106) | 0/106 |
| Tizen.Badge.Tests(29) | 0/29 |
| Tizen.Bluetooth.Tests(119) | 0/119 |
| Tizen.Calendar.Tests(112) | 0/112 |
| Tizen.Camera.Tests(155) | 0/155 |
| Tizen.Contacts.Tests(102) | 0/102 |
| Tizen.Context.Tests(22) | 0/22 |
| Tizen.DataControl.Tests(110) | 0/110 |
| Tizen.Device.Tests(22) | 0/22 |
| Tizen.DotnetUtil.Tests(1) | 0/1 |
| Tizen.Download.Tests(53) | 0/53 |
| Tizen.ElmSharp.Tests(1048) | 0/1048 |
| Tizen.Email.Tests(13) | 0/13 |
| Tizen.Feedback.Tests(8) | 0/8 |
| Tizen.FidoClient.Tests(29) | 0/29 |
| Tizen.Geofence.Tests(69) | 0/69 |

```
 Expected: True
 But was:  False

at Tizen.Account.AccountManager.Tests.Account
ServiceTests.GetAccountsAsync_CHECK_RESULT()
 in X:\repository\spin\net4.0\api\tct-suite-vs\Tizen.
Accounts.Tests\testcase\TSAccountService.cs:lin
e 68

execute case: Tizen.Accounts.Tests # Tizen.Acco
unt.AccountManager.Tests.AccountServiceTests.
GetAccountById_CHECK_RESULT...(FAIL)
 Failed to GetAccountCount Permission Denied
 Expected: True
 But was:  False

at Tizen.Account.AccountManager.Tests.Account
ServiceTests.GetAccountById_CHECK_RESULT() in
X:\repository\spin\net4.0\api\tct-suite-vs\Tizen.Ac
counts.Tests\testcase\TSAccountService.cs:line
109

execute case: Tizen.Accounts.Tests # Tizen.Acco
unt.AccountManager.Tests.AccountServiceTests.
AddAccount_CHECK_RESULT...(FAIL)
 Failed to GetAccountCount Permission Denied
 Expected: True
 But was:  False

at Tizen.Account.AccountManager.Tests.Account
ServiceTests.AddAccount_CHECK_RESULT() in X:\r
```

Figure 3-1-2d: Execute a New Auto Test Plan

## 3.1.3 Configuring Test Environment

According to pre-configuration steps in the pop-up dialog, set the test environment before start testing.



Please check the pre-configuration of TCT

| | |
|---|---|
| HIDDEN_AP | CSharpTCTHidden |
| PASSWORD_FOR_ALL_PRIVATE_AP | qwertyui |
| PRIVATE_AP_1_2 | CSharpTCTPrivate1 |
| PRIVATE_AP_2_2 | CSharpTCTPrivate2 |
| PUBLIC_AP | CSharpTCTPublic |
| BT_REMOTE_DEVICE_ADDRESS | AA:0A:99:AB:c5A:08 |
| HERE_MAPS_PROVIDER_KEY | 3m35FktUN4sBpDTcl/ckFerp2t5bCVjouh9v2HCw |
| MAPZEN_MAPS_PROVIDER_KEY | tizen-internal-ivdz94orc |
| MESSAGES_MOBILE_NUMBER | 01717662599 |

**Tizen.Device.Tests**
1.Charger should be connected to test target.(In case of Emulator, you have to set with control panel.)

**Tizen.Push.Tests**
1.A network connection capable of accessing the Internet MUST be established.

**Tizen.Download.Tests**
1.A network connection capable of accessing the Internet MUST be established.

**Tizen.Email.Tests**
1.A network connection capable of accessing the Internet MUST be established.
2.Set the email account on your device. Settings > Accounts > Email > Set created accounts.

**Tizen.System.Tests**
1.SdCard should be inserted on target.

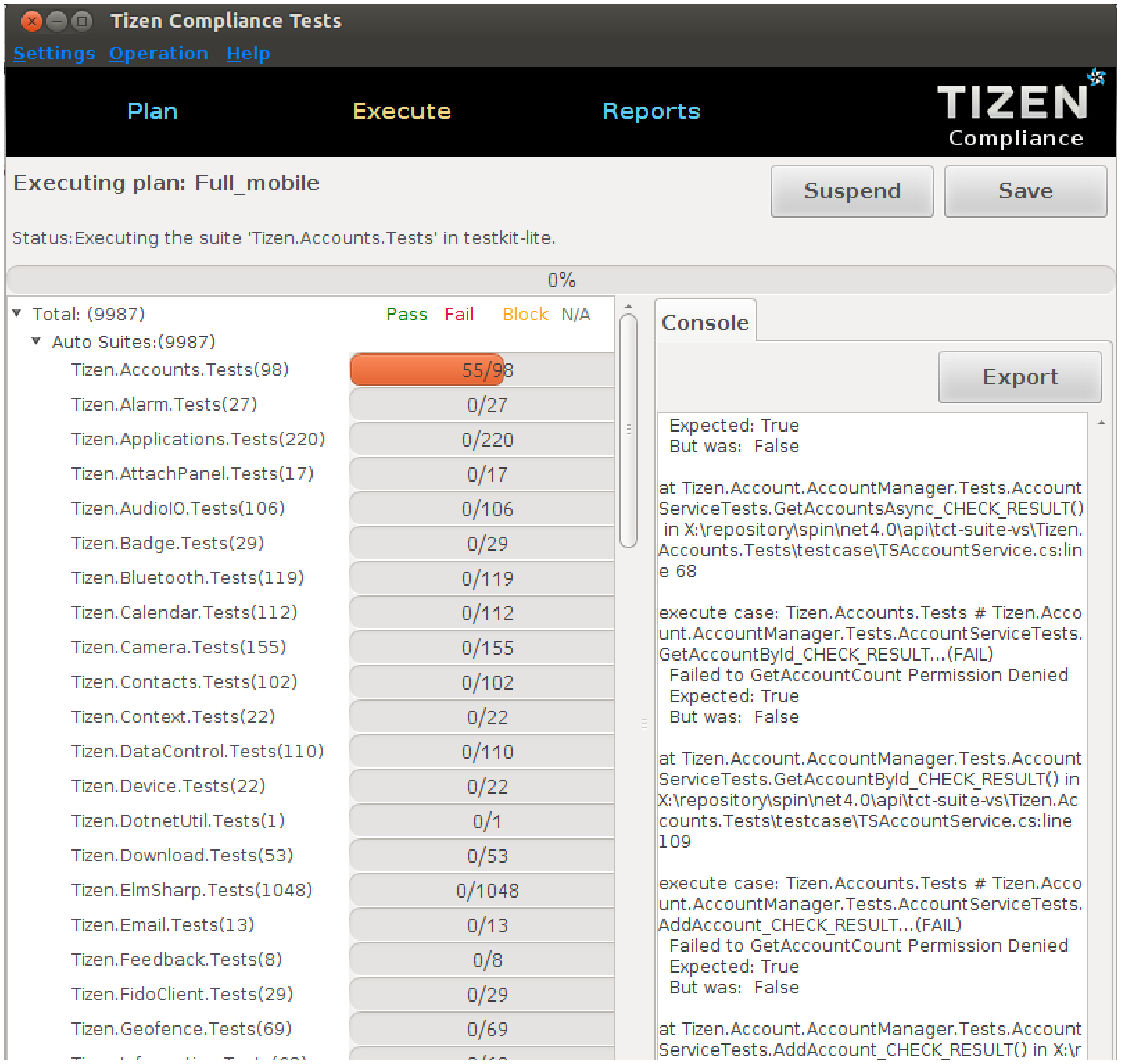
**Tizen.Packagemanager.Tests**
1.SdCard should be inserted on target.

**Tizen.Messages.Tests**
1.Insert the available SIM Card into SIM slot 1.
2.Set the MESSAGES_MOBILE_NUMBER configure.(example>01012345678)

Figure 3-1-3: Pre-configuration Popup

### 3.1.4 Monitoring Test Execution

After clicking **Continue** button in the configuration dialog, .NET TCT Manager will go to Execute page. The test status and log information will appear there.



Executing plan: Full_mobile

Status:Executing the suite 'Tizen.Accounts.Tests' in testkit-lite.

0%

| | |
|---|---|
| Total: (9987) | |
| Auto Suites:(9987) | |
| Tizen.Accounts.Tests(98) | 55/98 |
| Tizen.Alarm.Tests(27) | 0/27 |
| Tizen.Applications.Tests(220) | 0/220 |
| Tizen.AttachPanel.Tests(17) | 0/17 |
| Tizen.AudioIO.Tests(106) | 0/106 |
| Tizen.Badge.Tests(29) | 0/29 |
| Tizen.Bluetooth.Tests(119) | 0/119 |
| Tizen.Calendar.Tests(112) | 0/112 |
| Tizen.Camera.Tests(155) | 0/155 |
| Tizen.Contacts.Tests(102) | 0/102 |
| Tizen.Context.Tests(22) | 0/22 |
| Tizen.DataControl.Tests(110) | 0/110 |
| Tizen.Device.Tests(22) | 0/22 |
| Tizen.DotnetUtil.Tests(1) | 0/1 |
| Tizen.Download.Tests(53) | 0/53 |
| Tizen.ElmSharp.Tests(1048) | 0/1048 |
| Tizen.Email.Tests(13) | 0/13 |
| Tizen.Feedback.Tests(8) | 0/8 |
| Tizen.FidoClient.Tests(29) | 0/29 |
| Tizen.Geofence.Tests(69) | 0/69 |

```
 Expected: True
 But was:  False

at Tizen.Account.AccountManager.Tests.Account
ServiceTests.GetAccountsAsync_CHECK_RESULT()
 in X:\repository\spin\net4.0\api\tct-suite-vs\Tizen.
Accounts.Tests\testcase\TSAccountService.cs:lin
e 68

execute case: Tizen.Accounts.Tests # Tizen.Acco
unt.AccountManager.Tests.AccountServiceTests.
GetAccountById_CHECK_RESULT...(FAIL)
 Failed to GetAccountCount Permission Denied
 Expected: True
 But was:  False

at Tizen.Account.AccountManager.Tests.Account
ServiceTests.GetAccountById_CHECK_RESULT() in
X:\repository\spin\net4.0\api\tct-suite-vs\Tizen.Ac
counts.Tests\testcase\TSAccountService.cs:line
109

execute case: Tizen.Accounts.Tests # Tizen.Acco
unt.AccountManager.Tests.AccountServiceTests.
AddAccount_CHECK_RESULT...(FAIL)
 Failed to GetAccountCount Permission Denied
 Expected: True
 But was:  False

at Tizen.Account.AccountManager.Tests.Account
ServiceTests.AddAccount_CHECK_RESULT() in X:\r
```

Figure 3-1-4: .NET TCT Manager Execution UI Page

Click the **Suspend** button in *Execute UI* to stop executing the test plan. After stopping a test plan, click icon of *Reports UI* to resume test.

### 3.1.5 Retrieving Test Result

The test summary and details of the current test appear when the test is done. When select the Reports page, the history testing reports will show up.

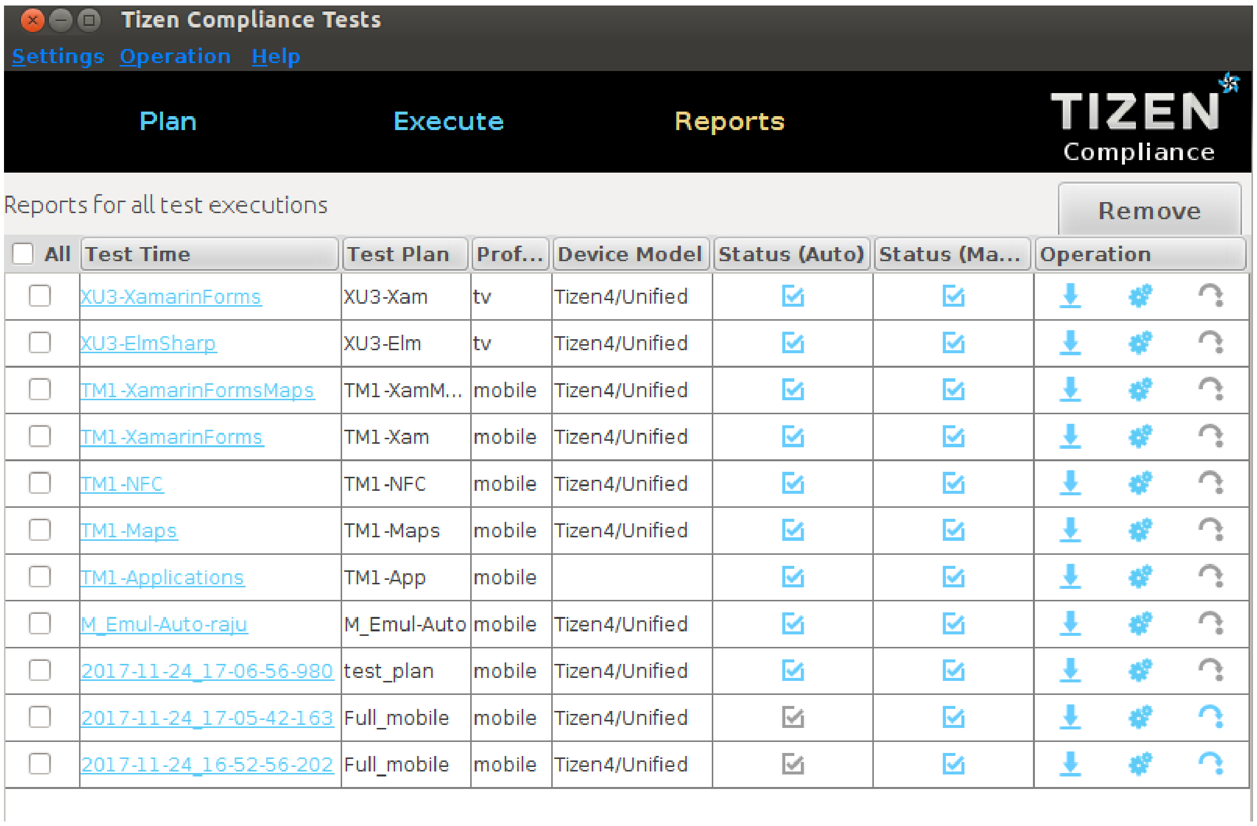
To view all reports list in reports UI, click one item to view summary information. Click functional icons can easily export results or re-execute known failures for any test plan. As shown in Figure 3-1-5a

**a.** Click the icon of ⬇ to download consolidated reports.

**b.** Click the icon of ⚙ will rerun non-passed test cases.

**c.** Click icon of ↻ to resume the test plan if the status of test plan is stopped.

**d.** Select one or multiple test reports in reports UI, then click **Remove** button, the selected items will be removed after user confirming.

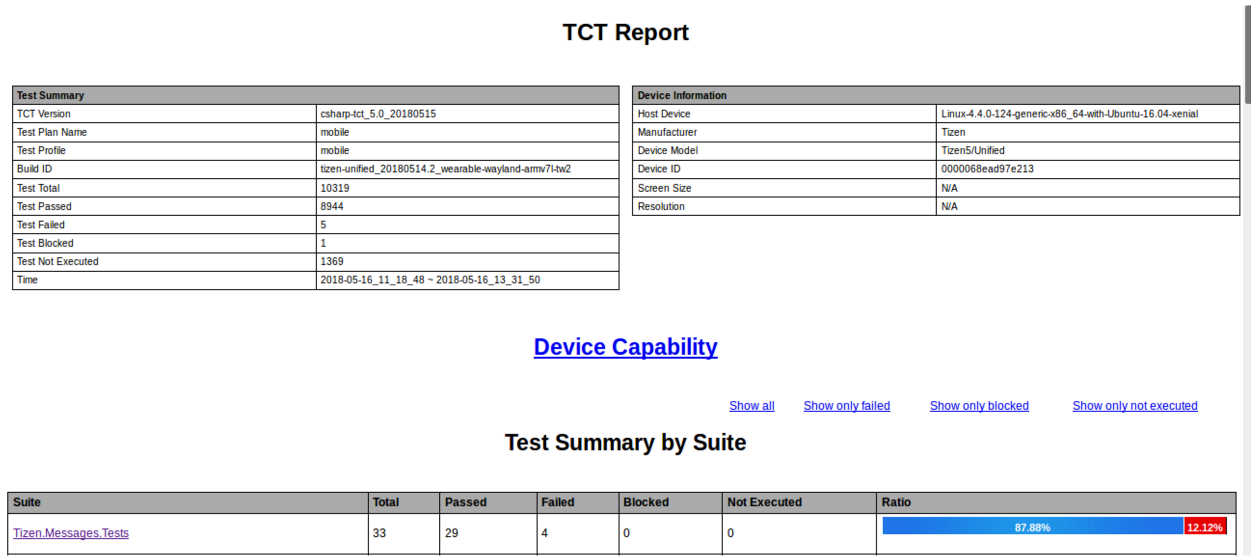


Reports for all test executions

| All | Test Time | Test Plan | Prof... | Device Model | Status (Auto) | Status (Ma... | Operation |
|---|---|---|---|---|---|---|---|
| ☐ | XU3-XamarinForms | XU3-Xam | tv | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | XU3-ElmSharp | XU3-Elm | tv | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | TM1-XamarinFormsMaps | TM1-XamM... | mobile | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | TM1-XamarinForms | TM1-Xam | mobile | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | TM1-NFC | TM1-NFC | mobile | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | TM1-Maps | TM1-Maps | mobile | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | TM1-Applications | TM1-App | mobile |  | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | M_Emul-Auto-raju | M_Emul-Auto | mobile | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | 2017-11-24_17-06-56-980 | test_plan | mobile | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | 2017-11-24_17-05-42-163 | Full_mobile | mobile | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |
| ☐ | 2017-11-24_16-52-56-202 | Full_mobile | mobile | Tizen4/Unified | ✓ | ✓ | ⬇ ⚙ ↻ |

Figure 3-1-5a: Report List

To view summary of the report, click on the **Test Time** entry, as shown in Figure 3-1-5b



TCT Report

| Test Summary | |
|---|---|
| TCT Version | csharp-tct_5.0_20180515 |
| Test Plan Name | mobile |
| Test Profile | mobile |
| Build ID | tizen-unified_20180514.2_wearable-wayland-armv7l-tw2 |
| Test Total | 10319 |
| Test Passed | 8944 |
| Test Failed | 5 |
| Test Blocked | 1 |
| Test Not Executed | 1369 |
| Time | 2018-05-16_11_18_48 ~ 2018-05-16_13_31_50 |

| Device Information | |
|---|---|
| Host Device | Linux-4.4.0-124-generic-x86_64-with-Ubuntu-16.04-xenial |
| Manufacturer | Tizen |
| Device Model | Tizen5/Unified |
| Device ID | 0000068ead97e213 |
| Screen Size | N/A |
| Resolution | N/A |

Device Capability

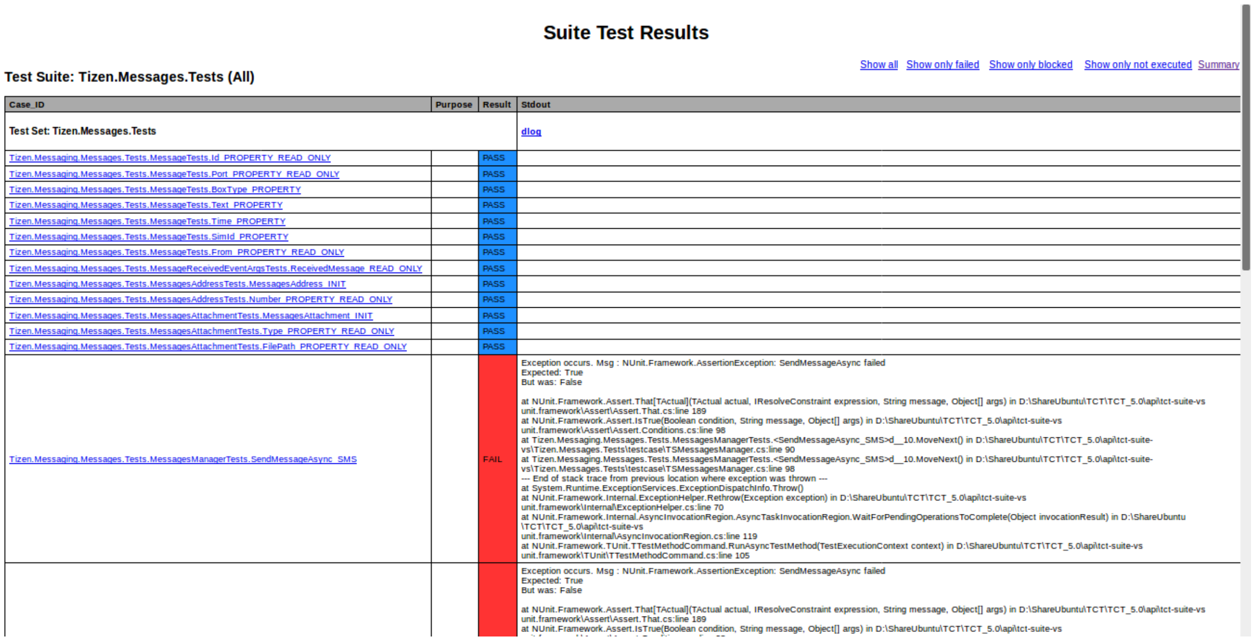
Show all  Show only failed  Show only blocked  Show only not executed

Test Summary by Suite

| Suite | Total | Passed | Failed | Blocked | Not Executed | Ratio |
|---|---|---|---|---|---|---|
| Tizen.Messages.Tests | 33 | 29 | 4 | 0 | 0 | 87.88% 12.12% |

Figure 3-1-5b: Test Summary by Suite

Click on the **suite** name in the Test Summary by Suite table to see details as shown in Figure 3-1-5c



Suite Test Results

Show all  Show only failed  Show only blocked  Show only not executed  Summary

Test Suite: Tizen.Messages.Tests (All)

| Case_ID | Purpose | Result | Stdout |
|---|---|---|---|
| Test Set: Tizen.Messages.Tests | | | dlog |
| Tizen.Messaging.Messages.Tests.MessageTests.Id_PROPERTY_READ_ONLY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessageTests.Port_PROPERTY_READ_ONLY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessageTests.BoxType_PROPERTY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessageTests.Text_PROPERTY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessageTests.Time_PROPERTY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessageTests.SimId_PROPERTY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessageTests.From_PROPERTY_READ_ONLY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessageReceivedEventArgsTests.ReceivedMessage_READ_ONLY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessagesAddressTests.MessagesAddress_INIT | | PASS | |
| Tizen.Messaging.Messages.Tests.MessagesAddressTests.Number_PROPERTY_READ_ONLY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessagesAttachmentTests.MessagesAttachment_INIT | | PASS | |
| Tizen.Messaging.Messages.Tests.MessagesAttachmentTests.Type_PROPERTY_READ_ONLY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessagesAttachmentTests.FilePath_PROPERTY_READ_ONLY | | PASS | |
| Tizen.Messaging.Messages.Tests.MessagesManagerTests.SendMessageAsync_SMS | | FAIL | Exception occurs. Msg : NUnit.Framework.AssertionException: SendMessageAsync failed<br>Expected: True<br>But was: False<br><br>at NUnit.Framework.Assert.That[TActual](TActual actual, IResolveConstraint expression, String message, Object[] args) in D:\ShareUbuntu\TCT\TCT_5.0\api\tct-suite-vs unit.framework\Assert\Assert.That.cs:line 189<br>at NUnit.Framework.Assert.IsTrue(Boolean condition, String message, Object[] args) in D:\ShareUbuntu\TCT\TCT_5.0\api\tct-suite-vs unit.framework\Assert\Assert.Conditions.cs:line 98<br>at Tizen.Messaging.Messages.Tests.MessagesManagerTests.<SendMessageAsync_SMS>d__10.MoveNext() in D:\ShareUbuntu\TCT\TCT_5.0\api\tct-suite-vs\Tizen.Messages.Tests\testcase\TSMessagesManager.cs:line 90<br>at Tizen.Messaging.Messages.Tests.MessagesManagerTests.<SendMessageAsync_SMS>d__10.MoveNext() in D:\ShareUbuntu\TCT\TCT_5.0\api\tct-suite-vs\Tizen.Messages.Tests\testcase\TSMessagesManager.cs:line 98<br>--- End of stack trace from previous location where exception was thrown ---<br>at System.Runtime.ExceptionServices.ExceptionDispatchInfo.Throw()<br>at NUnit.Framework.Internal.ExceptionHelper.Rethrow(Exception exception) in D:\ShareUbuntu\TCT\TCT_5.0\api\tct-suite-vs unit.framework\Internal\ExceptionHelper.cs:line 70<br>at NUnit.Framework.Internal.AsyncInvocationRegion.AsyncTaskInvocationRegion.WaitForPendingOperationsToComplete(Object invocationResult) in D:\ShareUbuntu \TCT\TCT_5.0\api\tct-suite-vs<br>unit.framework\Internal\AsyncInvocationRegion.cs:line 119<br>at NUnit.Framework.TUnit.TTestMethodCommand.RunAsyncTestMethod(TestExecutionContext context) in D:\ShareUbuntu\TCT\TCT_5.0\api\tct-suite-vs unit.framework\TUnit\TTestMethodCommand.cs:line 105 |
| | | | Exception occurs. Msg : NUnit.Framework.AssertionException: SendMessageAsync failed<br>Expected: True<br>But was: False<br><br>at NUnit.Framework.Assert.That[TActual](TActual actual, IResolveConstraint expression, String message, Object[] args) in D:\ShareUbuntu\TCT\TCT_5.0\api\tct-suite-vs unit.framework\Assert\Assert.That.cs:line 189<br>at NUnit.Framework.Assert.IsTrue(Boolean condition, String message, Object[] args) in D:\ShareUbuntu\TCT\TCT_5.0\api\tct-suite-vs |

Figure 3-1-5c: Show Detail Test Results

Choose one of the following to customize the report view, referring to Figure 6-5

**e.** **Show all**: show all the results

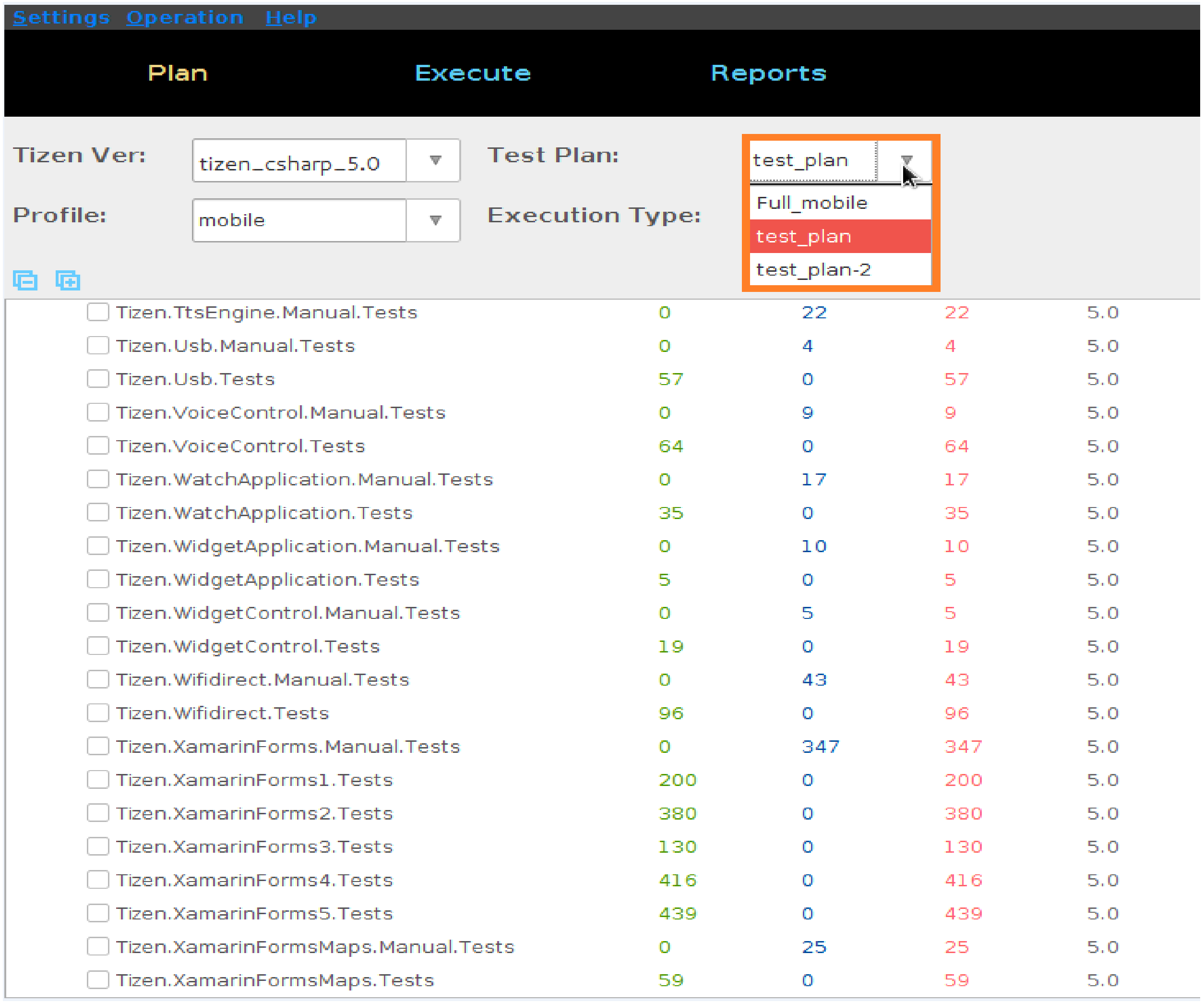
**f.** **Show only failed**: show cases that failed

**g.** **Show only blocked**: show cases that have blocked results

**h.** **Show only not executed**: show cases that have non-applicable(N/A) results

### 3.1.6 Re-Executing an existing test plan

Select an existing test plan (the packages in the plan will be selected) as Figure 3-1-6 and click Run button to re-execute the failed testcases.



Tizen Ver: tizen_csharp_5.0

Profile: mobile

Test Plan: test_plan

Execution Type:

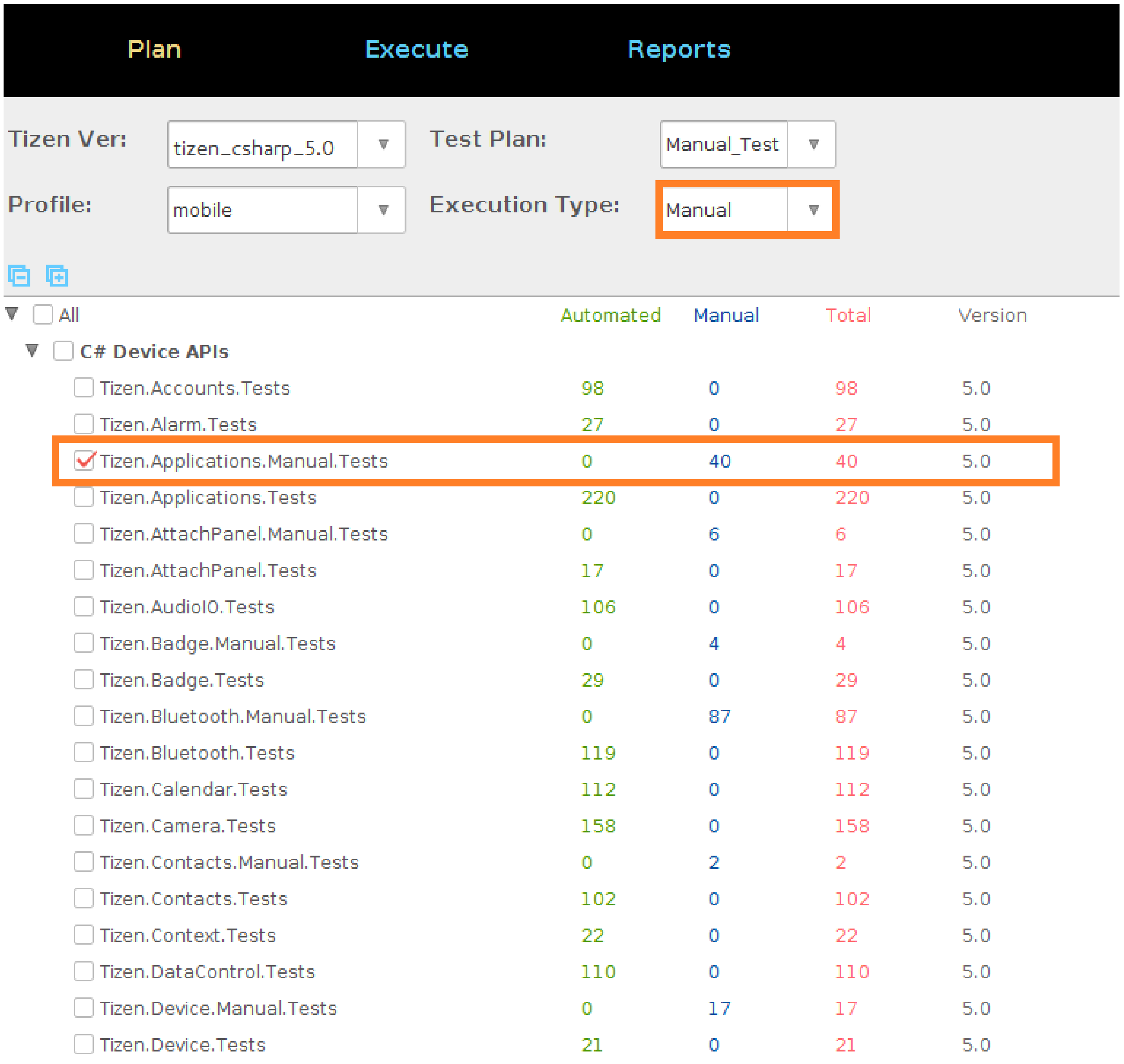
Full_mobile
test_plan
test_plan-2

| | | | | |
|---|---|---|---|---|
| Tizen.TtsEngine.Manual.Tests | 0 | 22 | 22 | 5.0 |
| Tizen.Usb.Manual.Tests | 0 | 4 | 4 | 5.0 |
| Tizen.Usb.Tests | 57 | 0 | 57 | 5.0 |
| Tizen.VoiceControl.Manual.Tests | 0 | 9 | 9 | 5.0 |
| Tizen.VoiceControl.Tests | 64 | 0 | 64 | 5.0 |
| Tizen.WatchApplication.Manual.Tests | 0 | 17 | 17 | 5.0 |
| Tizen.WatchApplication.Tests | 35 | 0 | 35 | 5.0 |
| Tizen.WidgetApplication.Manual.Tests | 0 | 10 | 10 | 5.0 |
| Tizen.WidgetApplication.Tests | 5 | 0 | 5 | 5.0 |
| Tizen.WidgetControl.Manual.Tests | 0 | 5 | 5 | 5.0 |
| Tizen.WidgetControl.Tests | 19 | 0 | 19 | 5.0 |
| Tizen.Wifidirect.Manual.Tests | 0 | 43 | 43 | 5.0 |
| Tizen.Wifidirect.Tests | 96 | 0 | 96 | 5.0 |
| Tizen.XamarinForms.Manual.Tests | 0 | 347 | 347 | 5.0 |
| Tizen.XamarinForms1.Tests | 200 | 0 | 200 | 5.0 |
| Tizen.XamarinForms2.Tests | 380 | 0 | 380 | 5.0 |
| Tizen.XamarinForms3.Tests | 130 | 0 | 130 | 5.0 |
| Tizen.XamarinForms4.Tests | 416 | 0 | 416 | 5.0 |
| Tizen.XamarinForms5.Tests | 439 | 0 | 439 | 5.0 |
| Tizen.XamarinFormsMaps.Manual.Tests | 0 | 25 | 25 | 5.0 |
| Tizen.XamarinFormsMaps.Tests | 59 | 0 | 59 | 5.0 |

Figure 3-1-6: Select an existing test plan

## 3.2 Launching .NET TCT Manager on Host Machine for Manual TC

Launch the .NET TCT Manager by shell command like automatic testcases. This time select execution type as Manual and select a package for manual test plan execution. Refer figure 3-2-1. The follow the pre-configuration popup(if needed) and click continue to execute manual tc.



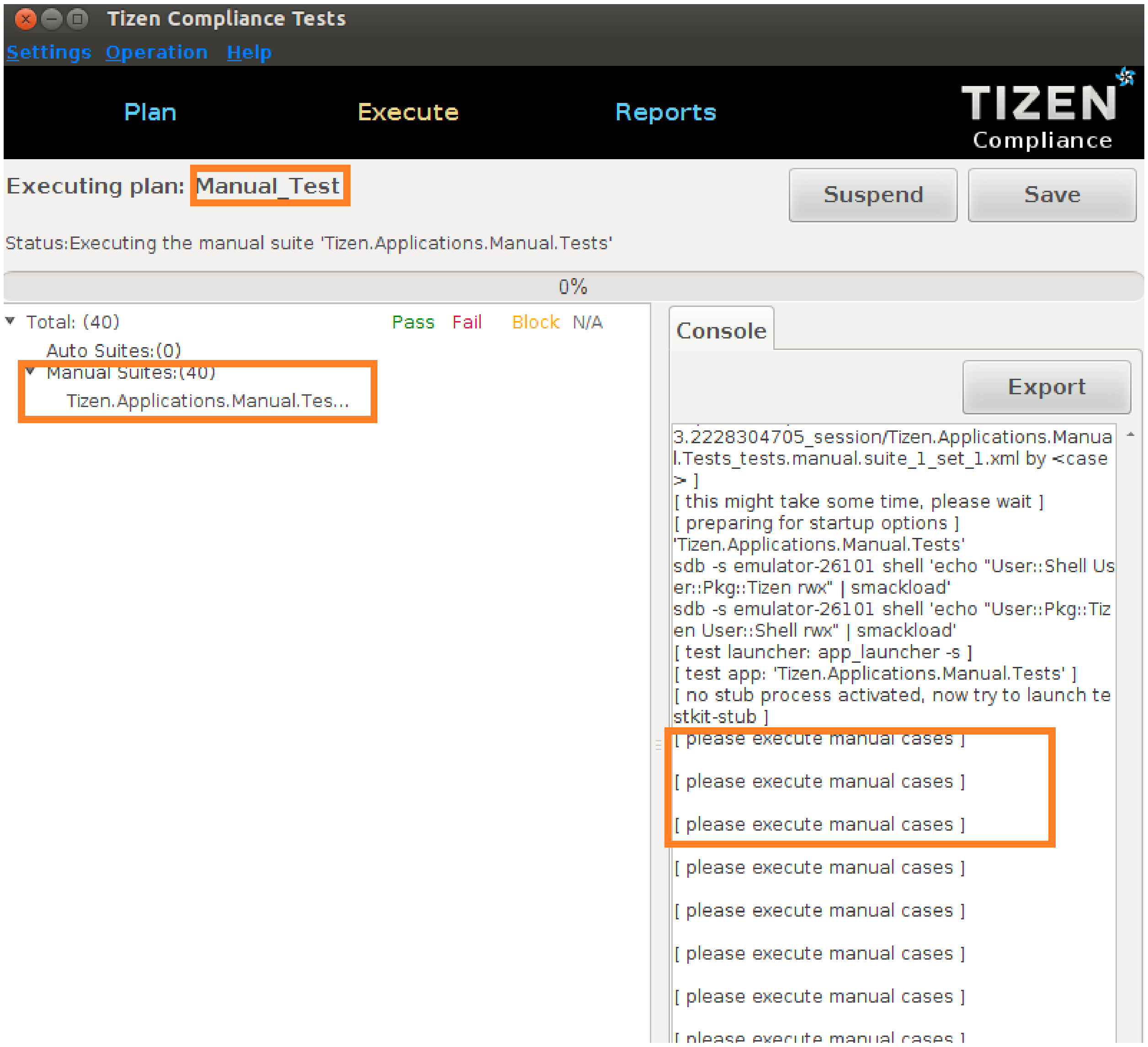
Tizen Ver: tizen_csharp_5.0
Test Plan: Manual_Test
Profile: mobile
Execution Type: Manual

| All | Automated | Manual | Total | Version |
|---|---|---|---|---|
| **C# Device APIs** | | | | |
| ☐ Tizen.Accounts.Tests | 98 | 0 | 98 | 5.0 |
| ☐ Tizen.Alarm.Tests | 27 | 0 | 27 | 5.0 |
| ✓ Tizen.Applications.Manual.Tests | 0 | 40 | 40 | 5.0 |
| ☐ Tizen.Applications.Tests | 220 | 0 | 220 | 5.0 |
| ☐ Tizen.AttachPanel.Manual.Tests | 0 | 6 | 6 | 5.0 |
| ☐ Tizen.AttachPanel.Tests | 17 | 0 | 17 | 5.0 |
| ☐ Tizen.AudioIO.Tests | 106 | 0 | 106 | 5.0 |
| ☐ Tizen.Badge.Manual.Tests | 0 | 4 | 4 | 5.0 |
| ☐ Tizen.Badge.Tests | 29 | 0 | 29 | 5.0 |
| ☐ Tizen.Bluetooth.Manual.Tests | 0 | 87 | 87 | 5.0 |
| ☐ Tizen.Bluetooth.Tests | 119 | 0 | 119 | 5.0 |
| ☐ Tizen.Calendar.Tests | 112 | 0 | 112 | 5.0 |
| ☐ Tizen.Camera.Tests | 158 | 0 | 158 | 5.0 |
| ☐ Tizen.Contacts.Manual.Tests | 0 | 2 | 2 | 5.0 |
| ☐ Tizen.Contacts.Tests | 102 | 0 | 102 | 5.0 |
| ☐ Tizen.Context.Tests | 22 | 0 | 22 | 5.0 |
| ☐ Tizen.DataControl.Tests | 110 | 0 | 110 | 5.0 |
| ☐ Tizen.Device.Manual.Tests | 0 | 17 | 17 | 5.0 |
| ☐ Tizen.Device.Tests | 21 | 0 | 21 | 5.0 |

Figure 3-2-1: Creating a manual test plan

### 3.2.1 Executing Manual Testcase

After clicking **Continue** button in the configuration dialog, .NET TCT Manager will go to Execute page. The test status and log information will appear there like Figure 3-2-2. And a test page will appear in the target device like Figure 3-2-3.



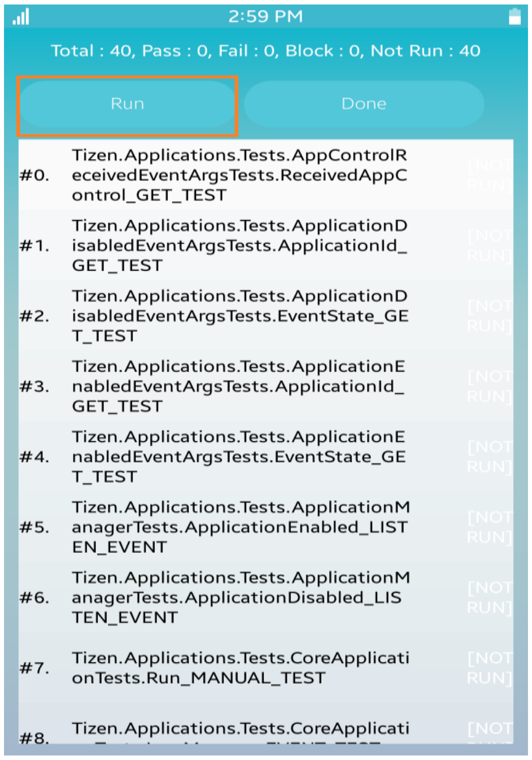
Status:Executing the manual suite 'Tizen.Applications.Manual.Tests'

```
3.2228304705_session/Tizen.Applications.Manua
l.Tests_tests.manual.suite_1_set_1.xml by <case
> ]
[ this might take some time, please wait ]
[ preparing for startup options ]
'Tizen.Applications.Manual.Tests'
sdb -s emulator-26101 shell 'echo "User::Shell Us
er::Pkg::Tizen rwx" | smackload'
sdb -s emulator-26101 shell 'echo "User::Pkg::Tiz
en User::Shell rwx" | smackload'
[ test launcher: app_launcher -s ]
[ test app: 'Tizen.Applications.Manual.Tests' ]
[ no stub process activated, now try to launch te
stkit-stub ]
[ please execute manual cases ]

[ please execute manual cases ]

[ please execute manual cases ]

[ please execute manual cases ]

[ please execute manual cases ]

[ please execute manual cases ]

[ please execute manual cases ]

[ please execute manual cases ]
```

Figure 3-2-2: .NET TCT Manager Execution UI Page for Manual TC

### 3.2.2 Running Manual Testcases on Target Device

A home test page will appear in the target device like Figure 3-2-3 which includes all testcases for that .NET test application. Click Run Button to start manual tc execution. A new page will appear for a specific testcase as Figure 3-2-4.

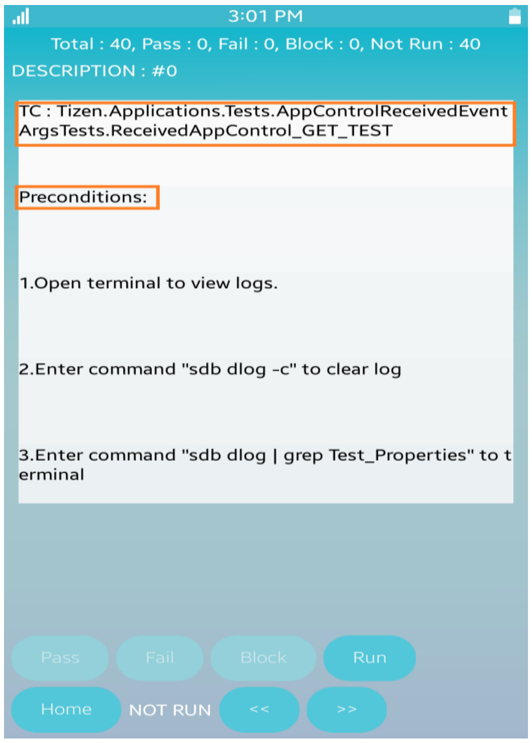


Total : 40, Pass : 0, Fail : 0, Block : 0, Not Run : 40

#0. Tizen.Applications.Tests.AppControlReceivedEventArgsTests.ReceivedAppControl_GET_TEST [NOT RUN]
#1. Tizen.Applications.Tests.ApplicationDisabledEventArgsTests.ApplicationId_GET_TEST [NOT RUN]
#2. Tizen.Applications.Tests.ApplicationDisabledEventArgsTests.EventState_GET_TEST [NOT RUN]
#3. Tizen.Applications.Tests.ApplicationEnabledEventArgsTests.ApplicationId_GET_TEST [NOT RUN]
#4. Tizen.Applications.Tests.ApplicationEnabledEventArgsTests.EventState_GET_TEST [NOT RUN]
#5. Tizen.Applications.Tests.ApplicationManagerTests.ApplicationEnabled_LISTEN_EVENT [NOT RUN]
#6. Tizen.Applications.Tests.ApplicationManagerTests.ApplicationDisabled_LISTEN_EVENT [NOT RUN]
#7. Tizen.Applications.Tests.CoreApplicationTests.Run_MANUAL_TEST [NOT RUN]
#8. Tizen.Applications.Tests.CoreApplicati [NOT

Figure 3-2-3: Manual Testcases Home Page in Target Device

In the test page there are testcase name, preconditions, steps, postconditions,
**Preconditions**: Follow the described procedure before running that testcase.
**Steps**: Follow the described procedure after clicking **Run** button.
**Run**: To execute the testcase click the Run button.
**Postconditions**: Follow the described process after completing **Steps** section.
**Pass**: Testcase will **Pass** automatically if expected results has come after following the procedures of Preconditions and Steps. Or click **Pass** button if expected results has come after following the procedures of Preconditions and Steps.
**Fail**: Testcase will **Fail** automatically if expected results has not come after following the procedures of Preconditions and Steps. Or click **Fail** button if expected results has not come after following the procedures of Preconditions and Steps.
**Not Run**: For some hardware limitations keep the testcase as not run.
**Next**: Click the next button to execute the next testcase.
**Previous**: Click the previous button to execute the previous testcase.
**Home**: Click the Home button to see the testcases home page.


3:01 PM

Total : 40, Pass : 0, Fail : 0, Block : 0, Not Run : 40

DESCRIPTION : #0

TC : Tizen.Applications.Tests.AppControlReceivedEventArgsTests.ReceivedAppControl_GET_TEST

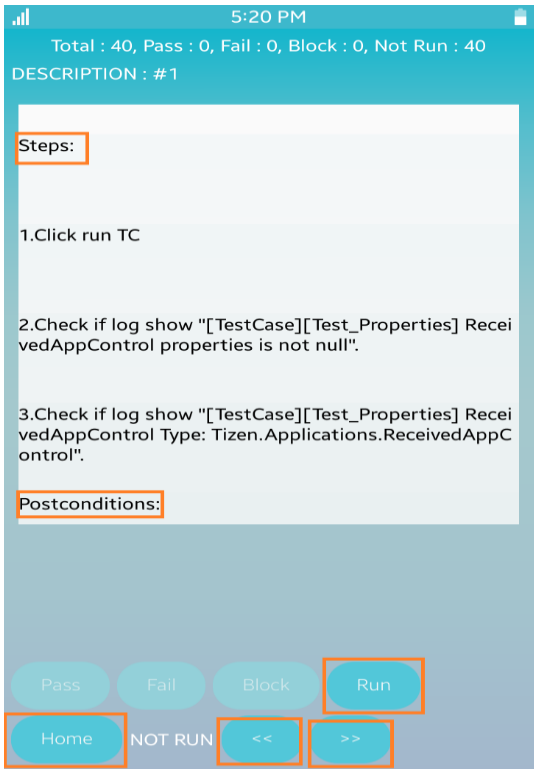
Preconditions:

1.Open terminal to view logs.

2.Enter command "sdb dlog -c" to clear log

3.Enter command "sdb dlog | grep Test_Properties" to terminal

Figure 3-2-4a: Manual Testcases Home Page in Target Device with TC name & Preconditions


5:20 PM

Total : 40, Pass : 0, Fail : 0, Block : 0, Not Run : 40

DESCRIPTION : #1

Steps:

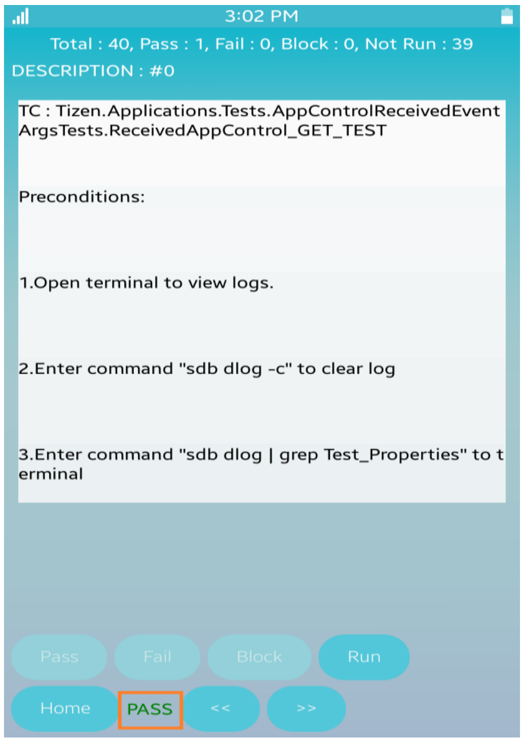
1.Click run TC

2.Check if log show "[TestCase][Test_Properties] ReceivedAppControl properties is not null".

3.Check if log show "[TestCase][Test_Properties] ReceivedAppControl Type: Tizen.Applications.ReceivedAppControl".

Postconditions:

Figure 3-1-4b: Manual Testcases Home Page in Target Device with Steps & Postconditions



3:02 PM

Total : 40, Pass : 1, Fail : 0, Block : 0, Not Run : 39

DESCRIPTION : #0

TC : Tizen.Applications.Tests.AppControlReceivedEventArgsTests.ReceivedAppControl_GET_TEST

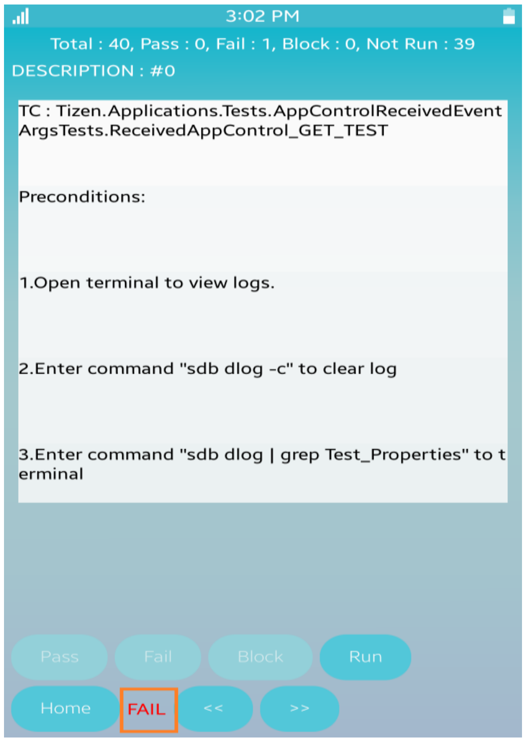
Preconditions:

1.Open terminal to view logs.

2.Enter command "sdb dlog -c" to clear log

3.Enter command "sdb dlog | grep Test_Properties" to terminal

Figure 3-1-4c: Manual Testcases Home Page in Target Device with test result Pass.



3:02 PM

Total : 40, Pass : 0, Fail : 1, Block : 0, Not Run : 39

DESCRIPTION : #0

TC : Tizen.Applications.Tests.AppControlReceivedEventArgsTests.ReceivedAppControl_GET_TEST

Preconditions:

1.Open terminal to view logs.

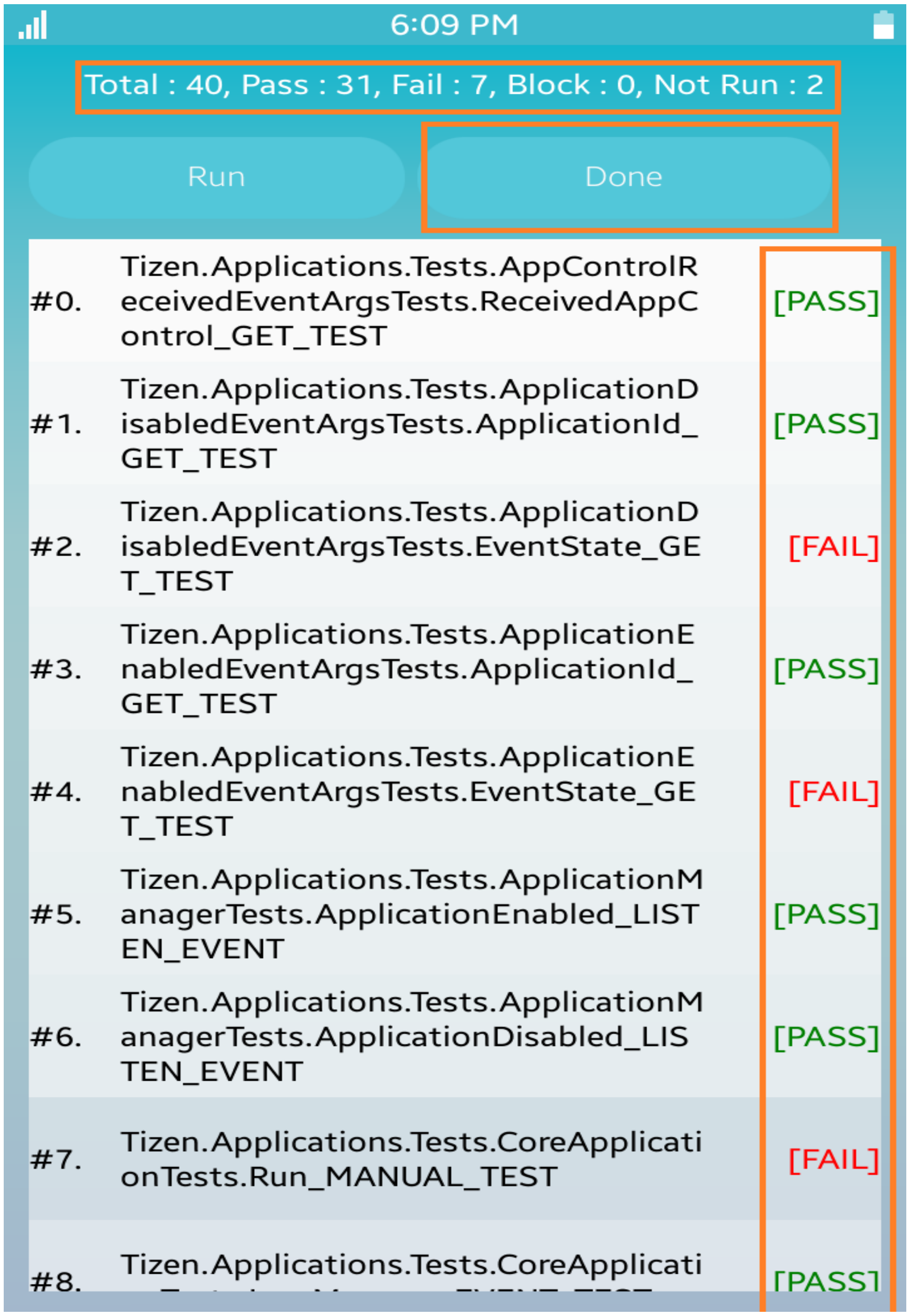
2.Enter command "sdb dlog -c" to clear log

3.Enter command "sdb dlog | grep Test_Properties" to terminal

Figure 3-1-4c: Manual Testcases Home Page in Target Device with test result Fail.

### 3.2.3 Results Generation for Manual Testcases

After completing the executions of all manual testcases click the **Done** button as Figure 3-2-5.



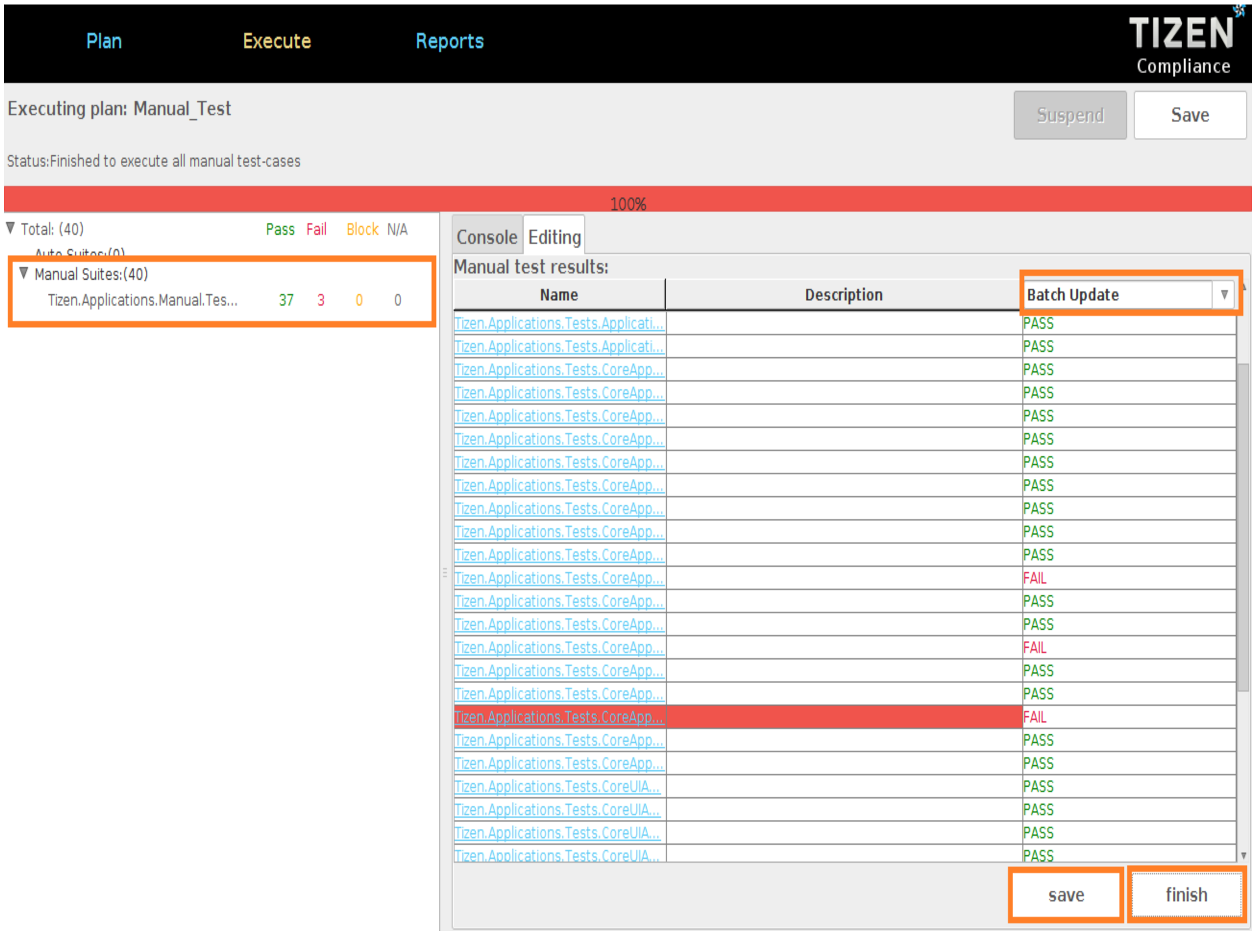
6:09 PM

Total : 40, Pass : 31, Fail : 7, Block : 0, Not Run : 2

```
#0. Tizen.Applications.Tests.AppControlReceivedEventArgsTests.ReceivedAppControl_GET_TEST [PASS]
#1. Tizen.Applications.Tests.ApplicationDisabledEventArgsTests.ApplicationId_GET_TEST [PASS]
#2. Tizen.Applications.Tests.ApplicationDisabledEventArgsTests.EventState_GET_TEST [FAIL]
#3. Tizen.Applications.Tests.ApplicationEnabledEventArgsTests.ApplicationId_GET_TEST [PASS]
#4. Tizen.Applications.Tests.ApplicationEnabledEventArgsTests.EventState_GET_TEST [FAIL]
#5. Tizen.Applications.Tests.ApplicationManagerTests.ApplicationEnabled_LISTEN_EVENT [PASS]
#6. Tizen.Applications.Tests.ApplicationManagerTests.ApplicationDisabled_LISTEN_EVENT [PASS]
#7. Tizen.Applications.Tests.CoreApplicationTests.Run_MANUAL_TEST [FAIL]
#8. Tizen.Applications.Tests.CoreApplicati [PASS]
```

Figure 3-2-5: Creating a manual test plan

After clicking the **Done** button in target device .NET TCT Manager will go to Reports page. The test status and log information will appear there like Figure 3-2-6.



Executing plan: Manual_Test

Status:Finished to execute all manual test-cases

| | Pass | Fail | Block | N/A |
|---|---|---|---|---|
| Total: (40) | | | | |
| Auto Suites:(0) | | | | |
| Manual Suites:(40) | | | | |
| Tizen.Applications.Manual.Tes... | 37 | 3 | 0 | 0 |

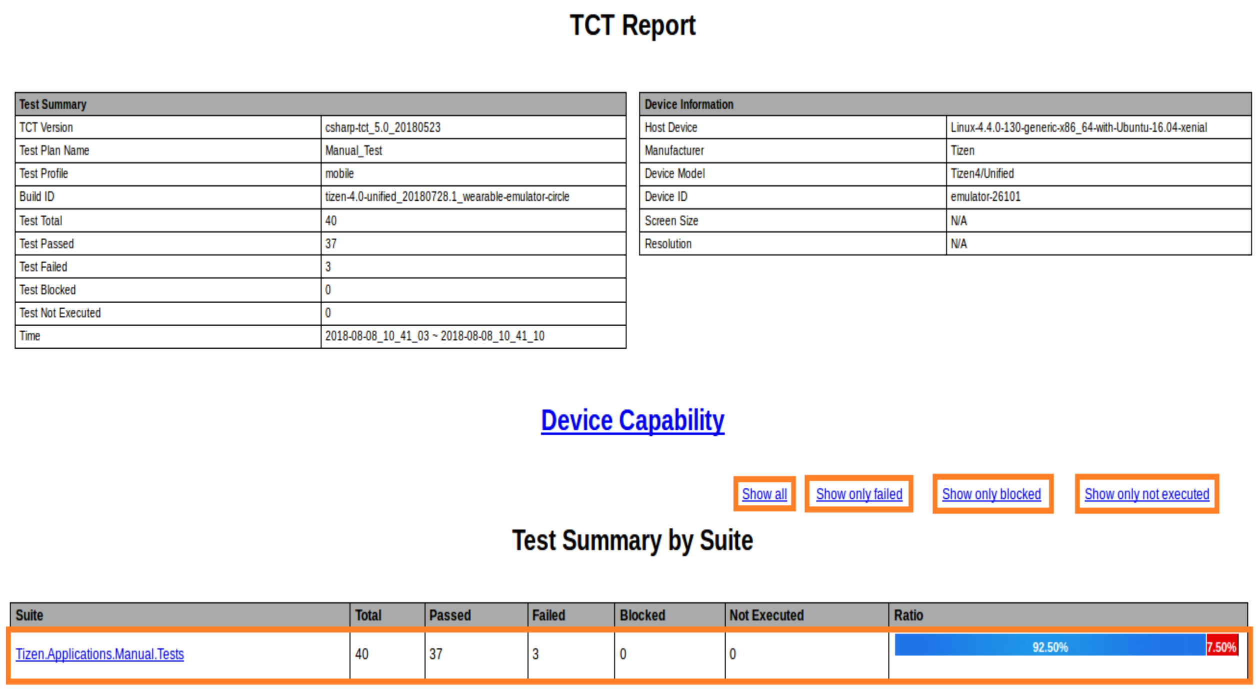
Manual test results:

| Name | Description | Batch Update |
|---|---|---|
| Tizen.Applications.Tests.Applicati... | | PASS |
| Tizen.Applications.Tests.Applicati... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | FAIL |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | FAIL |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | FAIL |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreApp... | | PASS |
| Tizen.Applications.Tests.CoreUIA... | | PASS |
| Tizen.Applications.Tests.CoreUIA... | | PASS |
| Tizen.Applications.Tests.CoreUIA... | | PASS |
| Tizen.Applications.Tests.CoreUIA... | | PASS |

Figure 3-2-6: Reports Page of .NET Manual TC

In this reports page click the **Batch Update** drop-down to change the results of all testcases or to change the result of a specific tc click the Batch Update field of that tc and change the result.

At last click the **Save** and **Finish** button to generate the reports for the tizen test application. A results file will be generated for .NET manual testcases like Figure 3-2-7 same as .NET automated testcases.



TCT Report

| Test Summary | |
|---|---|
| TCT Version | csharp-tct_5.0_20180523 |
| Test Plan Name | Manual_Test |
| Test Profile | mobile |
| Build ID | tizen-4.0-unified_20180728.1_wearable-emulator-circle |
| Test Total | 40 |
| Test Passed | 37 |
| Test Failed | 3 |
| Test Blocked | 0 |
| Test Not Executed | 0 |
| Time | 2018-08-08_10_41_03 ~ 2018-08-08_10_41_10 |

| Device Information | |
|---|---|
| Host Device | Linux-4.4.0-130-generic-x86_64-with-Ubuntu-16.04-xenial |
| Manufacturer | Tizen |
| Device Model | Tizen4/Unified |
| Device ID | emulator-26101 |
| Screen Size | N/A |
| Resolution | N/A |

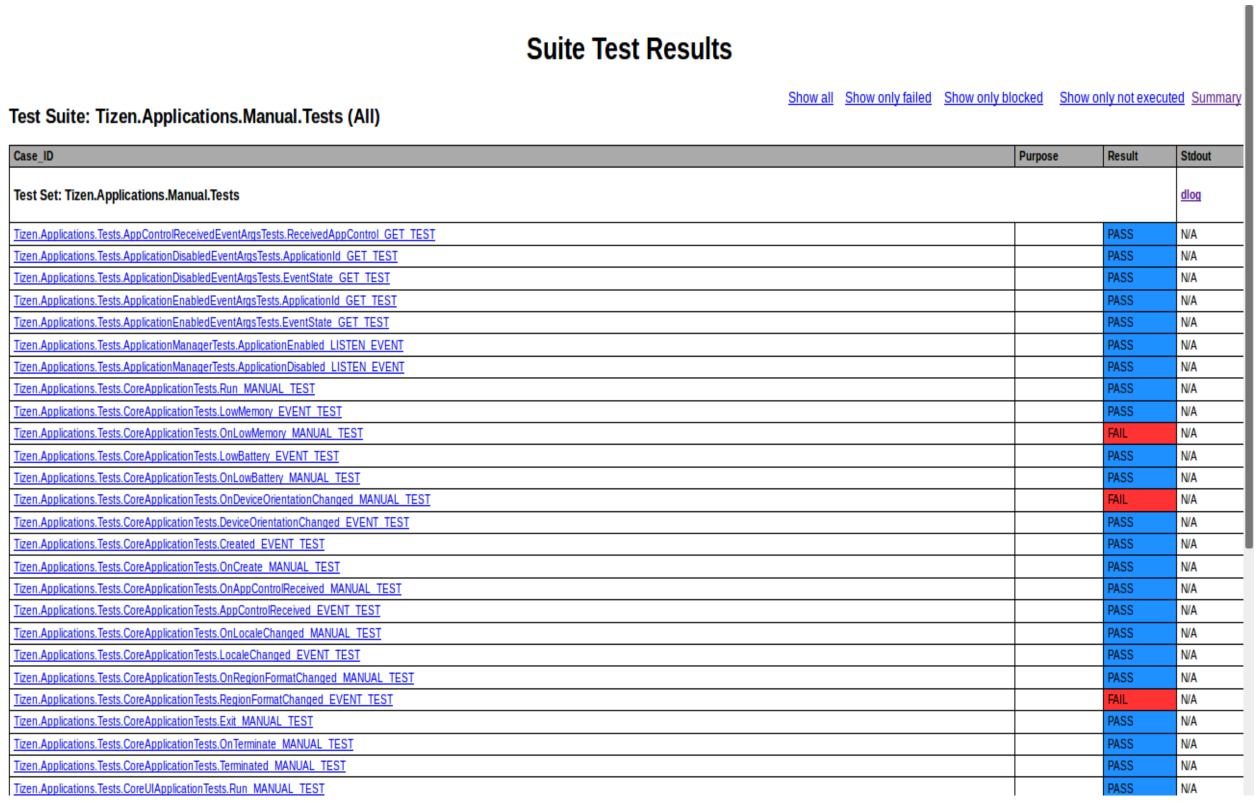
Device Capability

Show all　Show only failed　Show only blocked　Show only not executed

Test Summary by Suite

| Suite | Total | Passed | Failed | Blocked | Not Executed | Ratio |
|---|---|---|---|---|---|---|
| Tizen.Applications.Manual.Tests | 40 | 37 | 3 | 0 | 0 | 92.50% 7.50% |

Figure 3-2-7a: Test Summary by Suite



Suite Test Results

Show all　Show only failed　Show only blocked　Show only not executed　Summary

Test Suite: Tizen.Applications.Manual.Tests (All)

| Case_ID | Purpose | Result | Stdout |
|---|---|---|---|
| Test Set: Tizen.Applications.Manual.Tests | | | dlog |
| Tizen.Applications.Tests.AppControlReceivedEventArgsTests.ReceivedAppControl_GET_TEST | | PASS | N/A |
| Tizen.Applications.Tests.ApplicationDisabledEventArgsTests.ApplicationId_GET_TEST | | PASS | N/A |
| Tizen.Applications.Tests.ApplicationDisabledEventArgsTests.EventState_GET_TEST | | PASS | N/A |
| Tizen.Applications.Tests.ApplicationEnabledEventArgsTests.ApplicationId_GET_TEST | | PASS | N/A |
| Tizen.Applications.Tests.ApplicationEnabledEventArgsTests.EventState_GET_TEST | | PASS | N/A |
| Tizen.Applications.Tests.ApplicationManagerTests.ApplicationEnabled_LISTEN_EVENT | | PASS | N/A |
| Tizen.Applications.Tests.ApplicationManagerTests.ApplicationDisabled_LISTEN_EVENT | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.Run_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.LowMemory_EVENT_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.OnLowMemory_MANUAL_TEST | | FAIL | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.LowBattery_EVENT_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.OnLowBattery_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.OnDeviceOrientationChanged_MANUAL_TEST | | FAIL | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.DeviceOrientationChanged_EVENT_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.Created_EVENT_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.OnCreate_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.OnAppControlReceived_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.AppControlReceived_EVENT_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.OnLocaleChanged_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.LocaleChanged_EVENT_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.OnRegionFormatChanged_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.RegionFormatChanged_EVENT_TEST | | FAIL | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.Exit_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.OnTerminate_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreApplicationTests.Terminated_MANUAL_TEST | | PASS | N/A |
| Tizen.Applications.Tests.CoreUIApplicationTests.Run_MANUAL_TEST | | PASS | N/A |

Figure 3-2-7b: Test Summary by Suite

## 3.2.4 Purposes and preconditions for each manual suites

### 3.2.4.1 Tizen.Applications.EventManager.Manual.Tests

The testing suite will catch some events. Thus, you need to prepare an ear-jack to use for tcs in the testing suite.

### 3.2.4.2 Tizen.Applications.Manual.Tests

Each and every testcase of this suite will display a specific log in the terminal for some actions like change memory, battery level, language change, app installation/uninstallation, orientation change, event occur and so on.

**Note**: The date and time of the target device must be correct.

### 3.2.4.3 Tizen.AttachPanel.Manual.Tests

To test this suite there must be available some pictures in the Gallery which are captured by the target device's camera.

### 3.2.4.4 Tizen.Badge.Manual.Tests

If the target device does not support Badge feature this suite should be passed after tc run. The purpose of Badge app is to display a label with a number above 'CertBadgeTest' application icon.

### 3.2.4.5 Tizen.Bluetooth.Manual.Tests

To test this suite there must be available two BT enable target devices. In pre-configurations set Remote HID and BT address from Settings > About Device > Status.

**Note**: For paring related testcases if the expected popup does not appear after clicking the Run button then immediately go to the 'Settings' page of the target device. After this the popup may appear.

### 3.2.4.6 Tizen.ComponentBased.Default.Manual.Tests

Need to run command line "sdb log –c" to clean logs before TC's executed. TCs of testing suite will check by logs

#### 3.2.4.7 Tizen.ComponentBase.Manual.Tests

Similar testing suite “Tizen.Component.Based.Default”, you need to run command line “sdb log –c” to clean logs before TC’s executed. TCs of testing suite will check by logs

#### 3.2.4.8 Tizen.Connection.Manual.Tests

Connection need to support inthernet cable, it should be tested in real devices insteal emulator because TCs need some ethernet port.

#### 3.2.4.9 Tizen.Contacts.Manual.Tests

Contact need to support in the testing device. If not supported this feature tc will pass automatically. This module tcs mainly based on change language (English (United States) to Korean and Korean to English (United States) from settings.

#### 3.2.4.10 Tizen.Device.Manual.Tests

Most of the testcases of this suite are battery level, charger connection, LED display, brightness status related. To change the battery level in target OTG cable can be used to drain the battery level.

#### 3.2.4.11 Tizen.DevicePolicyManager.Manual.Tests

DevicePolicyManager include TCs, which will check policies of devices. We need run “sdb dlog –c “ to clean logs before TCs are executable. All TCs of testing suite need to check result in log console

#### 3.2.4.12 Tizen.Download.Manual.Tests

Internet connection is needed for this suite to download some specific files over the net.

#### 3.2.4.13 Tizen.ElmSharp.Manual.Tests

ElmSharp is a simple C# wrapper of native elementary. In this suite there are lots of widget to test like Calendar, DateTimeSelector, Entry, Hover effect, Box, Genlist, grid and various effect selectors.

#### 3.2.4.14 Tizen.ElmSharpWearable.Manual.Tests

This suite only test on Wearable devices

#### 3.2.4.15 Tizen.Information.Manual.Tests

To verify this suite a SIM must be inserted into the target device with internet connection. Bluetooth, tethering, Gps, TvOut feature should be supported in the target device. Other tcs are audiojack, charging, rotation related.

**Note**: To change GpsStatus use below command.

```
$ sdb root on
$ sdb shell
# location_test > select case 1
```

#### 3.2.4.16 Tizen.Inputmethod.Manual.Tests

All testcases in suite should be show some logs corresponding.

#### 3.2.4.17 Tizen.Location.Manual.Tests

To verify this suite Location Service feature should be supported in the target device. Location should be enabled in target device.

#### 3.2.4.18 Tizen.Log.Manual.Tests

The date and time of the target device must be correct. For successful execution of this app specific log will be displayed in the terminal.

#### 3.2.4.19 Tizen.Maps.Manual.Tests

A network connection capable of accessing the internet must be established. In preconfig page set Maps Provider Key. Maps must be loaded and displayed properly. Then perform various gesture and event test on the maps like tap, click, double click, zoom, two finger zoom, two finger rotations, two finger click, press, long press etc.

#### 3.2.4.20 Tizen.MediaKey.Manual.Tests

A headset(mobile) or remote controller(tv, xu3) with play, pause, next, previous, stop and other media key is needed to verify this suite.

#### 3.2.4.21 Tizen.MediaVision.Manual.Tests

Media Vision app provides functionality for barcodes detection and generation. Just verify whether barcode image displayed properly or not.

#### 3.2.4.22 Tizen.Mediacontent.Manual.Tests

Just verify whether a thumbnail image displayed properly or not.

#### 3.2.4.23 Tizen.Multimedia.Manual.Tests

This suite verifies the functionalities of camera, face detection, recorder, video and audio player and manager, radio and so on. Earphone should be connected. If earphone does not work use BT headphone. To test screen mirroring related test Wifi-Direct should be available. It may require kill and start 'scm_testfile.tpk' app on verifying every screen-mirroring related testcase.

#### 3.2.4.24 Tizen.NUI.Manual.Tests

NUI (Natural User Interface) provides keyboard, touch, and mouse handling. To test connect a Bluetooth keyboard with the target device. Do not touch on the screen of the target device and use keyboard/BT keyboard to test keyboard related test.

#### 3.2.4.25 Tizen.NUI.Components.Manual.Tests

This suite provides keyboard, touch and mouse handling.

#### 3.2.4.26 Tizen.NUI.Wearable.Manual.Tests

#### 3.2.4.27 Tizen.Nfc.Manual.Tests

Testing device need to be NFC supported. Turn on NFC on testing device. Make this TC application ready for HCE through NFC settings. Prepare to send Apdu via NFC reader. (NFC Reader is a PC-linked contactless smart card reader/writer developed based on 13.56 MHz Contactless (RFID) Technology).

#### 3.2.4.28 Tizen.Notifications.Manual.Tests

Testing device should have notification capabilities. Please follow instruction written on the each testcase to execute test on this module.

#### 3.2.4.29 Tizen.Nsd.Manual.Tests

Two testing devices must be connected to same AP (access point-Wifi Hotspot). This nsd module needs to install on these two testing devices. Please follow instruction written on the each testcase to execute test on this module.

#### 3.2.4.30 Tizen.Packagemanager.Manual.Tests

Device must have SD card inserted. Please follow instruction written on the each testcase to execute test on this module.

#### 3.2.4.31 Tizen.PrivacyPrivilegeManager.Manual.Tests

To change privacy settings go to "Settings -> Privacy and security -> Privacy setting" To change account setting go to "Account" and uncheck Tizen.PrivacyPrivilegeManager" and to change call setting go to "Call" and uncheck Tizen. PrivacyPrivilegeManager".

#### 3.2.4.32 Tizen.Sensor.Manual.Tests

Sensor manual mostly include sensor (Accelerometer, FaceDownGesterDetector, GravitySensor, Gyroscope etc.) related TCs. Mostly the goal is to invoke the sensor event. In case of emulator, to invoke sensor event go to control panel and under expected sensor option change value as expected. If the functionality works as expected, then TC will Pass Automatically. Otherwise, press Fail.

#### 3.2.4.33 Tizen.Stt.Manual.Tests

In Stt manual there are some TCs related to changing "Language and Input" to "Stt", to do this you need to simply go to "Settings" and change the option. For TCs related to Speech Recording make sure the device is connected to an active internet connection (In case of emulator Desktop should be connected to an active internet connection). Otherwise speech will not be recognized. For better result these test should be done in a firewall free internet environment. Otherwise speech may not be recognized. To test on TV 'Samsung One Remote Control' is needed when input tester voice.

#### 3.2.4.34 Tizen.SttEngine.Manual.Tests

Need to ensure Wifi or Data Network is present. Prior to running a TC go to "Settings -> Language and Input -> Speech-to-text(STT)". Under "Preferred Engine" section select "STT Engine".

#### 3.2.4.35 Tizen.System.Manual.Tests

System manual checks if the SD card or USB removed event is invoked. So make sure a SD card or USB is inserted (In case of emulator SD card can be inserted from control panel). If the functionality works as expected, then TC will Pass Automatically. Otherwise, press Fail.

#### 3.2.4.36 Tizen.Telephoney.Manual.Tests

Telephony manual checks various situations about making or receiving a call. To check it on device you need to insert a SIM card to make calls (In case of emulator Call can be made from control panel under Telephony section). If the functionality works as expected, then TC will Pass Automatically. Otherwise, press Fail.

#### 3.2.4.37 Tizen.Tts.Manual.Tests

Some TCs require changing Tts language. To change Tts language go to “Settings -> Language and Input -> Text-to-speech (TTS)” and change the Language Under “Language”. If the TC requires changing Tts Engine go to “Settings -> Language and Input -> Text-to-speech (TTS)” and under “Preferred engine” section select “TTS engine” available version. Target device should be connected to internet. Need to connect earphone/headphone/BT headphone.

#### 3.2.4.38 Tizen.TtsEngine.Manual.Tests

Need to ensure Wifi or Data Network is present. Prior to running a TC go to Settings -> Language and Input -> Text-to-speech (TTS). Under “Preferred Engine” section select “TTS Engine” available version.

#### 3.2.4.39 Tizen.Usb.Manual.Tests

To test this suite need to connect a USB device with target device through OTG cable (Micro USB to USB). Connect the micro USB end with target device and connect the USB end with a pendrive.

#### 3.2.4.40 Tizen.VoiceControl.Manual.Tests

In VoiceControl manual there are some TCs related to changing “Language and Input” to “Voice Control”, to do this you need to simply go to “Settings” and change the option. In some TCs you need to execute “killall” command, for this make sure you are in Root Mode by executing “sdb shell” and then “su” command. In TCs where you need to Launch Voice Control panel, you can run the panel from Notification Panel. Target device should be connected to internet. To test on TV ‘Samsung One Remote Control’ is needed when input tester voice.

#### 3.2.4.41 Tizen.WatchApplication.Manual.Tests

This module can only test in wearables and wearable emulators. Watch Application test mostly include viewing logs. Only watchface named “watch” need to select during the test.

#### 3.2.4.42 Tizen.Webview.Manual.Tests

#### 3.2.4.43 Tizen.WidgetApplication.Manual.Tests

Widget Application test mostly include viewing logs. To do that you need execute “dlog” command in terminal. You can view installed app from SDB shell using “app_launcher –l”. Also can launch app using “ app_launcher –s <pkg id>”. To kill a app use “app_launcher –k <pkg id>”. <PKG ID> can be viewed from the list when “app_launcher –l” command id executed.

#### 3.2.4.44 Tizen.WidgetControl.Manual.Tests

Widget Control test mostly include viewing logs. To do that you need execute “dlog” commands in terminal. You can view installed app from SDB shell using “app_launcher –l”. Also can launch app using “ app_launcher –s <pkg id>”. To kill app use “app_launcher –k <pkg id>”. <PKG ID> can be viewed from the list when “app_launcher –l” command id executed.

#### 3.2.4.45 Tizen.Wifi.Manual.Tests

A WiFi AP (Access point) with WPS PBS and WPS PIN support is required to test Network manual package. You need to input the Pin number and WPS PBS AP & WPS PIN AP in the host pc during installation time on the test device. Before test please check that you device is already connected with AP. If it is already connected then please make forget it.

#### 3.2.4.46 Tizen.Wifidirect.Manual.Tests

You need to input the Pin number and WPS PBS AP & WPS PIN AP in the host pc during this package installation time on the test device. Need another device with wi-fi direct supported. Need to set the device name as WifidirectCsapiTest. In a particular tc named Tizen.Network.WiFiDirect.Tests.ConnectionStatusChangedEventArgsTests.ConnectionState_PROPERTY_GET_ENUM_DISASSOCIATE need two other wi-fi direct enable device named as WifidirectCsapiTest1 & WifidirectCsapiTest2. Please follow instruction written on the each testcase to execute test on this module.

# A Appendix

**a. Known Issues**

**Symptom:** All automated testing fails and an error message says: "fail to connect with test service."

**Solution:** After changing to use another target device, need to rerun tct-config-device.sh on host side to set up the test environment of the new target device.

**Symptom:** When rerunning failed cases, the UIFW package will not be tested even there is failed cases in UIFW.

**Solution:** Rerunning UIFW failed case is not supported yet.

**b. Troubleshooting**

**Q:** On target device, power consumption is faster than power charging through a USB cable. What should I do to make sure the full TCT test can be executed on my device?

**A:** Use target device with a power supply

**Q:** Some .NET test packages failed to be installed on the target device. What should I do?

**A:** It might be because the certification for these packages did not pass. Set the target device's time and date to the current date to avoid this issue.