

AUTOTEST INNOVATIVE SOLUTIONS LLP

# AutoRun- Mobile Test Automation

---

## User Manual

**ATIS**

**January 17**

*This documentation is user manual for using AutoRun automatic play back utility and proprietary intellectual property of AutoTest. Any unauthorized use, reproduction, preparation of derivative works, performance, or display of this document, or software represented by this document, without the express written permission of AutoTest is strictly prohibited.*

## Table of Contents

---

|       |   |    |
|-------|---|----|
| 1     | General Info .....                                | 2  |
| 1.1   | Introduction.....                                 | 2  |
| 1.2   | Project Overview - AutoRun.....                   | 2  |
| 1.2.1 | Improved resource productivity .....              | 2  |
| 2     | AUTORUN DETAILS.....                              | 4  |
| 2.    | Process flow for Autotrun.....                    | 4  |
| 2.1   | Setting Tab (+ Automated Script Generation) ..... | 4  |
| 2.2   | Execution Tab .....                               | 10 |

## 1 General Info

---

### 1.1 INTRODUCTION

The purpose of this document is to explain the features and functions of the AutoRun (Mobile Test Automation).

### 1.2 PROJECT OVERVIEW - AUTORUN

AutoRun is developed for Automating Regression Testing for any Mobile app under Testing. Objective is to have a ready to deploy, high on usability software infrastructure to facilitate automation of regression testing irrespective of business, design and technology of the application. Product can be used for modular as well as end to end scenario Testing. Business benefits include:

- Drastically reduce the time to script due to automated script generation feature due to ease of use
- Significant reduction in training time and cost to onboard any team member (non-technical)
- Very Low maintenance efforts, time and cost
- Automated Test Data Structure creation
- 3 Level of Reporting for detailed analysis
- Auditable Screenshots with execution results

#### 1.2.1 Improved resource productivity

Our automation products save on manual efforts required for repeated execution of test cases under multiple cycles and regression testing due to change in application. Automation is intended to save on manual efforts, improve accuracy, enhance test coverage and increase efficiency.

There are a few checkpoints to keep in mind for effective use of the product:-

- Install the pre-requisite software before using the product
- Provide adb and appium location information properly
- Provide location for storing screenshots

## AUTORUN

- Define the scenario / test cases with verification points
- Do not interfere manually while running the scripts.

This product is being developed with a perspective to help testing teams to automate easily and focus more on coverage and execution analysis then struggling for technical skills for scripting and change management.

## 2 AUTORUN DETAILS

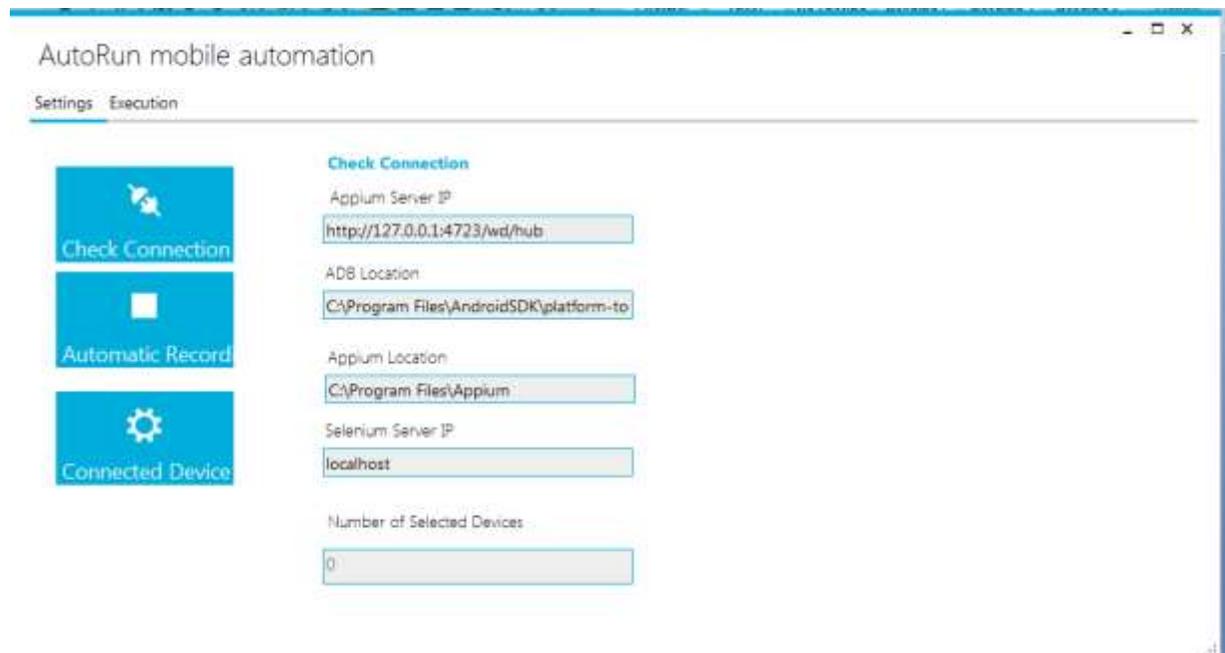
### 2. PROCESS FLOW FOR AUTOTRUN

There are 2 Tabs when AutoRun is launched:-

- Settings Tab
- Execution Tab

#### 2.1 SETTING TAB (+ AUTOMATED SCRIPT GENERATION)

The Settings tab will be shown first time when AutoRun launch. This tab includes the automatic script generation:-



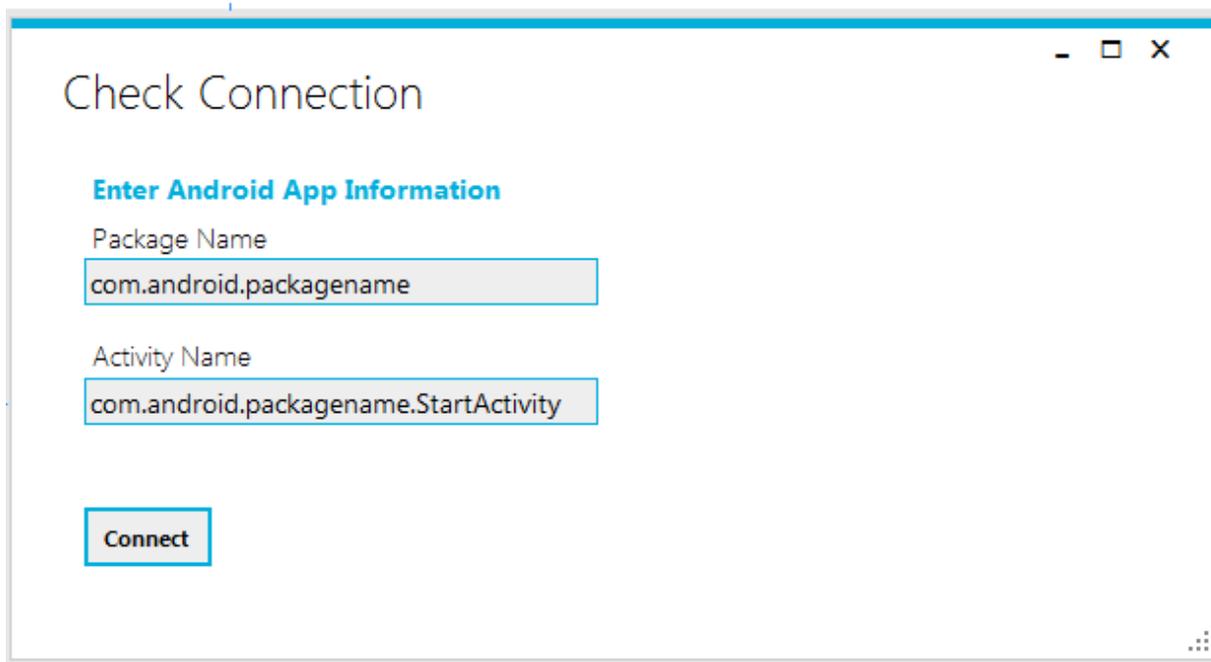
1. **Fill the following information :**

- Provide Appium Server address.
- Enter ADB Location.
- Enter Appium Location
- Selenium Server IP (optional; only required for multiple mobile device)

## 2. Check Connection

Pre-requisite: Start Appium server.

- Enter package name and activity name.
- Click connect button



## 3. Automatic Recording (Automated Script Generation):

The automatic recording page generates scripts automatically in excel file. These scripts will be uploaded in AutoRun to execute the scenario.

**Pre-requisite:** ADB location and select attached mobile device (in the case of multiple mobile devices).

**Below are steps to create script automatically:**

Step 1: Press the take screenshot button as shown in figure.

Step 2: Then click the element which you want to record in the flow.

Step 3: Add description, expected result (optional) and enter the input value in case of input field in sendkey row in element details.

Step 4: Press record element button to record element.

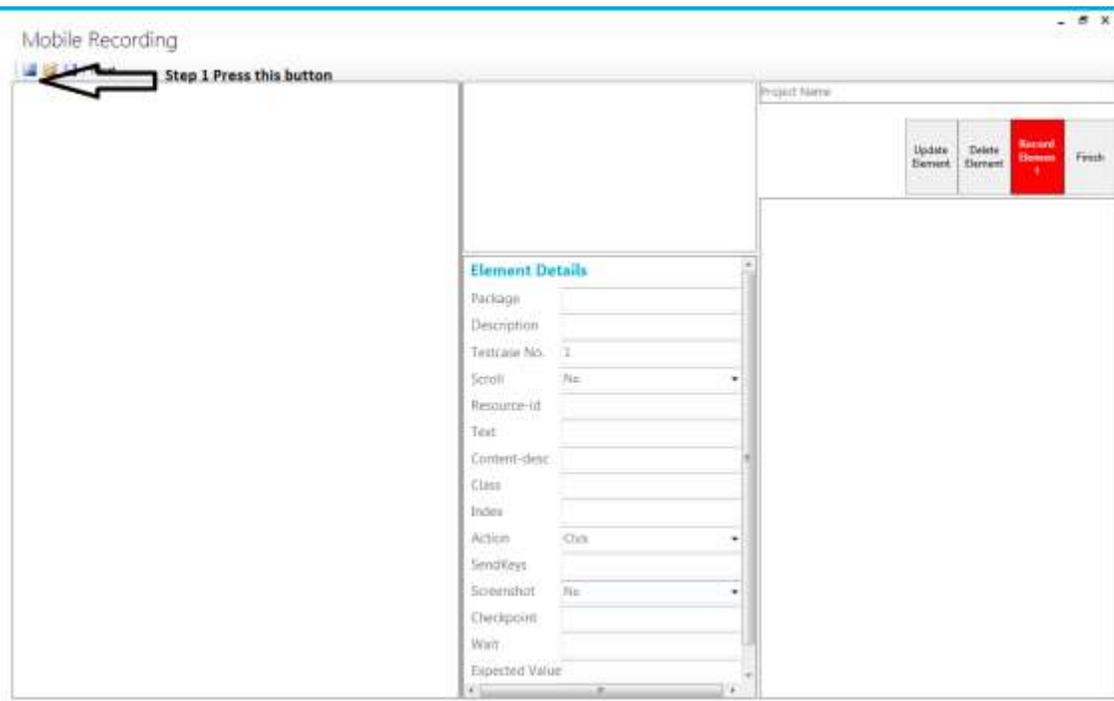
## AUTORUN

Step 5: Repeat the process till the flow ends

Step 6: Press the Finish button to generate the excel script

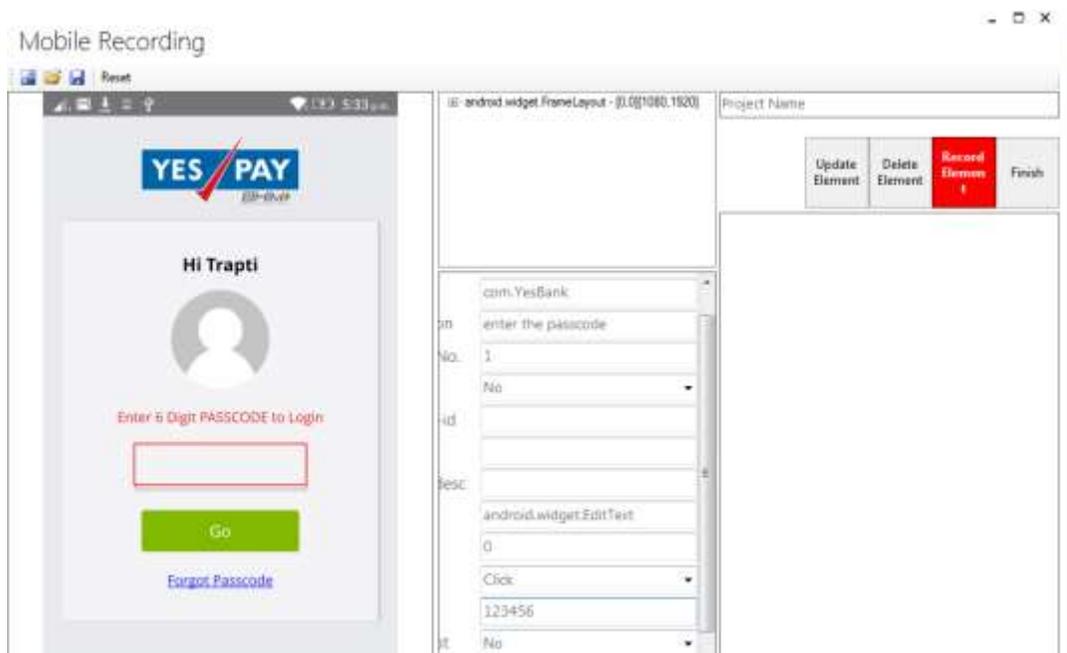
### Explaining each step with screenshots

Step 1 - Press the Take screenshot button (screen on the attached mobile will be shown)

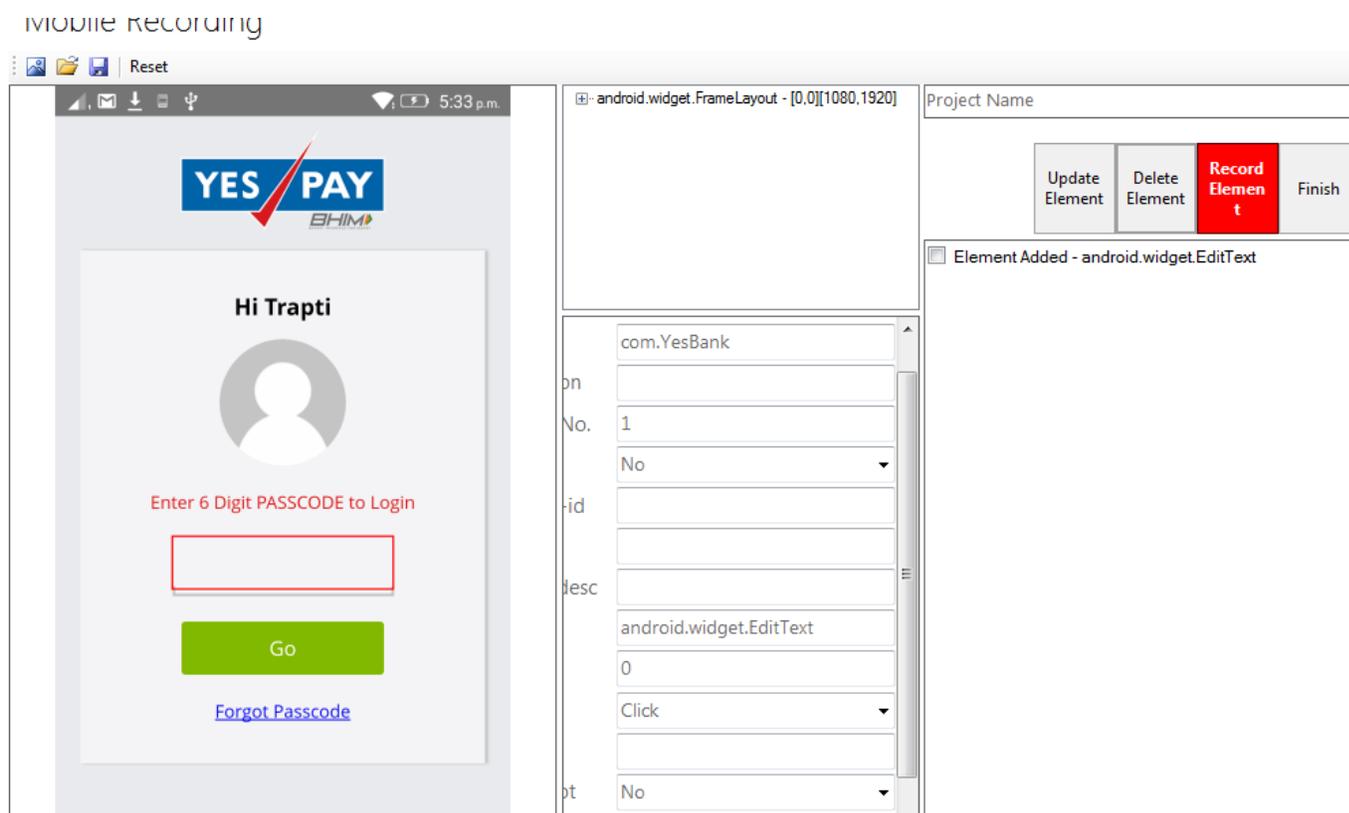


AUTORUN

Step 2: Then click the element which you want to record in the flow.

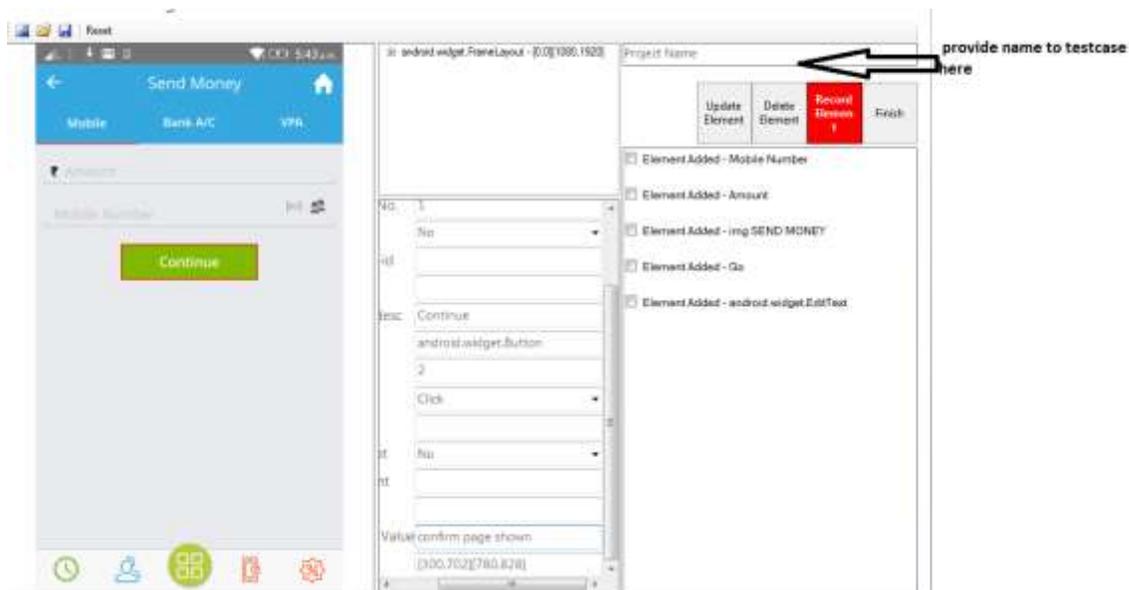


Step 3: Add description, expected result (optional) and enter the input value in case of input field in sendkey row in element details.



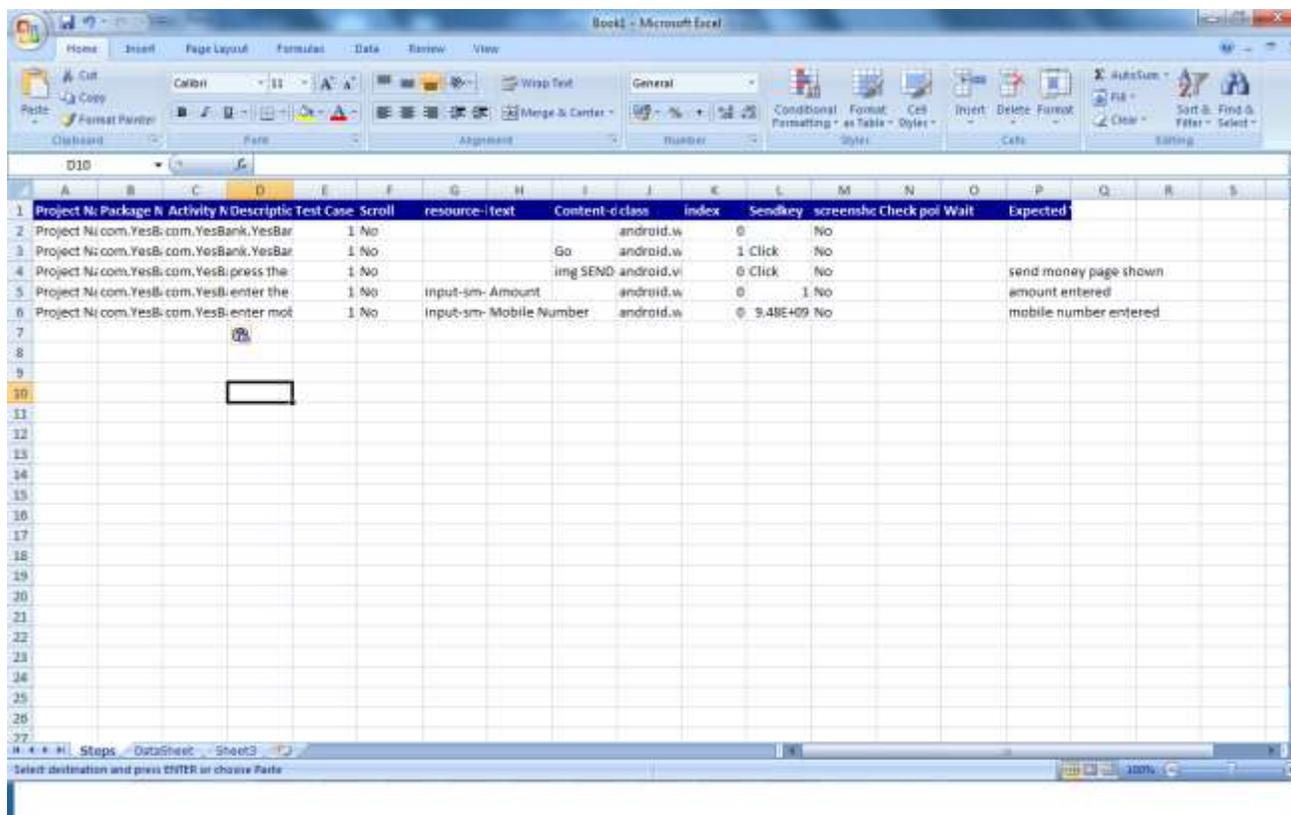
AUTORUN

Step 4: Press record element button to record element.



Step 5: Repeat the process till the flow ends

Step 6: Press the Finish button to generate the excel Script automatically with DataSheet headers



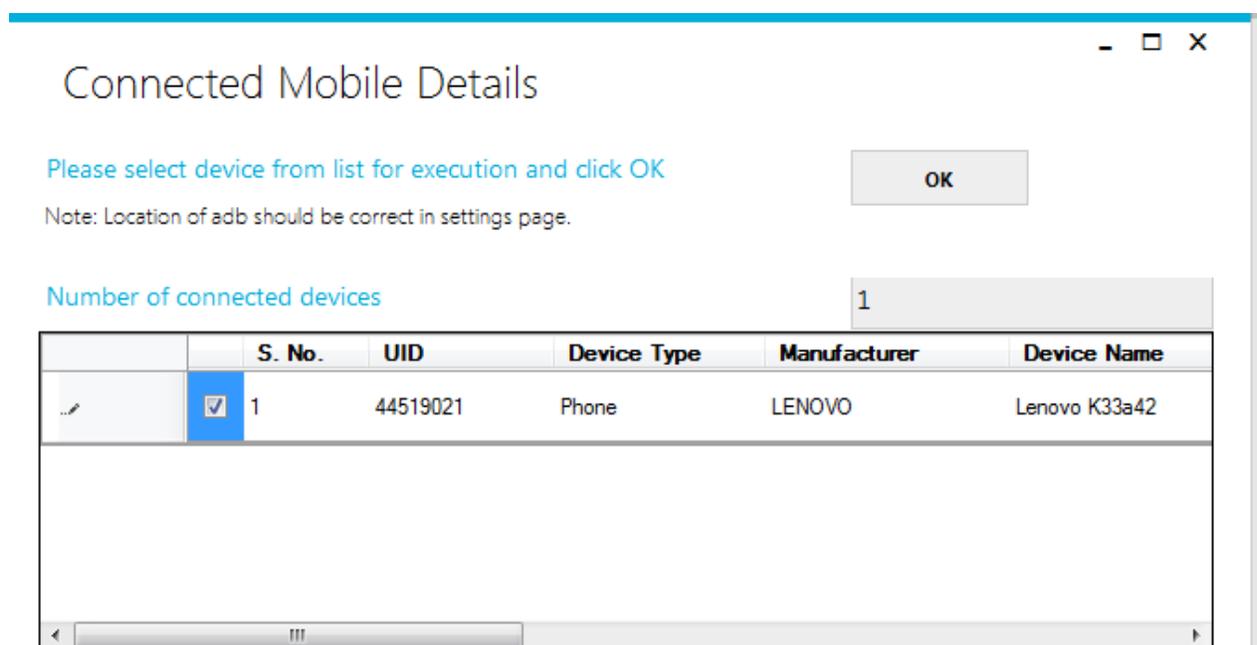
## AUTORUN

User can use other features like

- Save screenshot for future use while automating
- While scripting user can delete, update any selected element.
- open already saved screenshot and reset all recording element.

### 4. Connected Mobiles:

After click the connected mobile autorun shown the list of connected device and user can select the number of device to run the script



This step is mandatory to run the scripts on autorun.

## AUTORUN

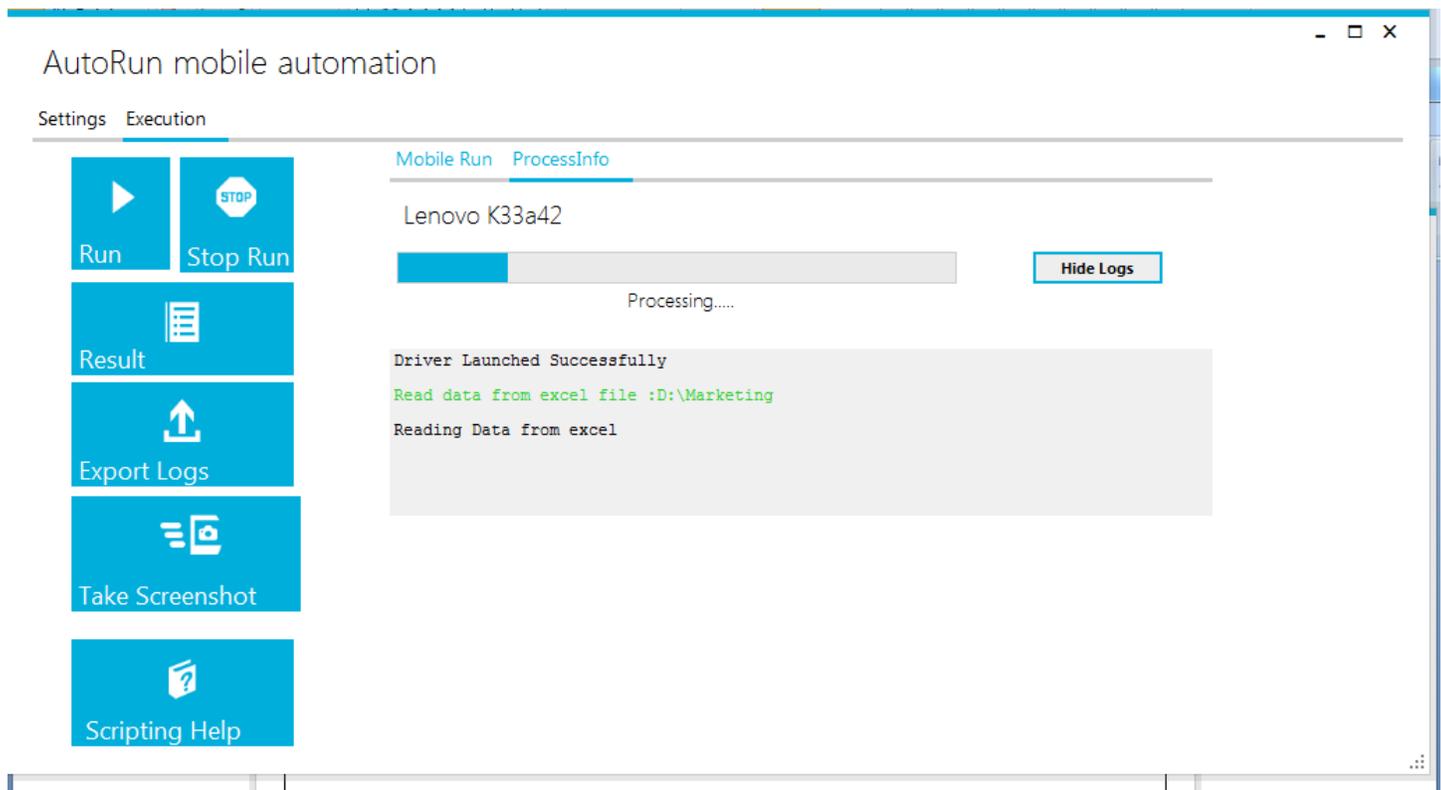
## 2.2 EXECUTION TAB

**1. Mobile Run (Execution with multiple datasets)**

Pre requisite: Select mobile device Connected mobile page and automatically generated script in excel.

- After selecting mobile device, go to execution tab
- Upload excel sheet.
- Browse to Screenshots Location.
- Click Run button.

\*Process log shows the Progress status of the script during execution.

**2. Stop Run**

User can stop the execution by clicking stop run button.

AUTORUN

3. **Result**

User can see the results by clicking result button. Level 1 is here. (there is 3 level reporting)

-
□
×

### DeviceTestcaseResultSummary

Exit

| #                    | Device Name   | Execution on Device | Checkpoint Result |
|----------------------|---------------|---------------------|-------------------|
| 1                    | Lenovo K33a42 | Pass                | Fail              |
| <b>Total Devices</b> |               | <b>1</b>            | <b>1</b>          |
| <b>Passed</b>        |               | <b>1</b>            | <b>0</b>          |
| <b>Failed</b>        |               | <b>0</b>            | <b>1</b>          |
| <b>Unavailable</b>   |               | <b>0</b>            | <b>0</b>          |

4. **Export Log**

User can export the execution log using export log button.

-
□
×

### AutoRun mobile automation

Settings
Execution

▶  
Run

STOP  
Stop Run

☰  
Result

↑  
Export Logs

📷  
Take Screenshot

📖  
Scripting Help

Mobile Run
ProcessInfo

Lenovo K33a42

Hide Logs

complete with Success

×

Export log to D:\Marketing presentation\_AutoTest\Citrus\POC\_scenarios\Scan and pay\citrus\_scan and\_pay\_old\_logs Successfully

OK

AUTORUN

**5. Manual Screenshot**

User can take screenshot manually using manual screenshot feature.

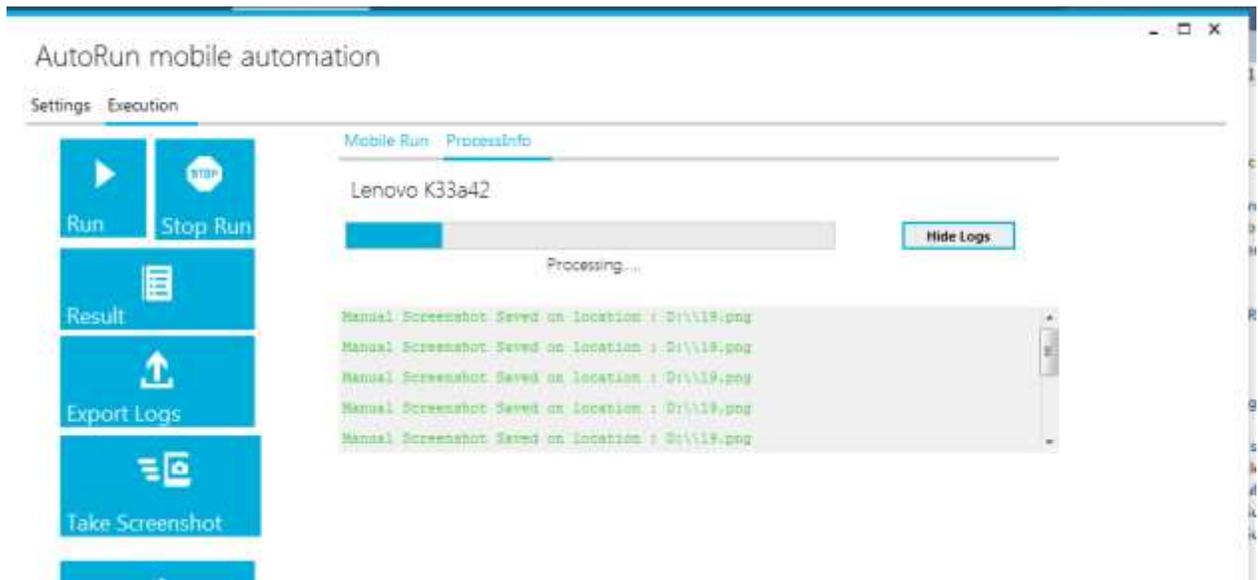
**Below are steps to take screenshot of mobile device and saved automatically on user define location:**

Step 1: Connect the mobile device to the machine.

Step 2: Then enter the ADB location of the machine

Step 3: Browse the location where user want to save the screenshot.

Step 2: Then click the manual screenshot button to automatically save the screenshot to pre specified location.



**Scripting Help**

# AUTORUN

User can take help from scripting help function. Scripting help shows more than 20 in-built function and logics. User can take help of functions in creating Script logics.

AutoRun Scripting Help Page

| Sr No. | Functionality  | Description   | Column F | Column L | column M | column N      | column O | Comment |
|--------|--|---|----------|----------|----------|---------------|----------|---------|
| 1      | Click any element                                    | suppose you want to click Login button so write 'Click' in column L   |          | Click    |          |               |          |         |
| 2      | Input any value in element                           | suppose you want to enter user name 'abc' so write 'abc' in column L  |          | abc      |          |               |          |         |
| 3      | Click element basis of text/content and ignore index | suppose you want to click drop down value on text basis and ignore index then write 'partial' in column N   |          | Click    |          | partial       |          |         |
| 4      | Click element before waiting some seconds            | suppose you need 4 seconds wait for loading an element then write '4' in column O   |          | Click    |          |               | 4        |         |
| 5      | Check value with text/content                        | suppose you want to Check particular element text/content so here steps - 1 write that text or content in appropriate column (column # for text and I for content) 2 write 'Check' in column L 3 write text/content in column N |          | Check    |          | text          |          |         |
| 6      | Capture sting value                                  | suppose you want to Capture particular element text/content so here steps - 1 write 'Capture' in column L 2 write text/content string 'I' in column N and index number which will used in Check                                 |          | Capture  |          | text.string.1 |          |         |
| 7      | Check value with previously capture text value       | suppose you want to Check particular element text/content previously capture value in I here steps - 1 write 'Check' in column L 2 write text.string.1 in column N  |          | Check    |          | text.string.1 |          |         |
| 8      | Capture number value                                 | suppose you want to Capture particular element numeric value so here steps - 1 write 'Capture' in column L 2 write text/content int' in column N and index number which will used in Check                                      |          | Capture  |          | text.int.1    |          |         |