

## **Auto**SQUARE - "Automating the Automation"

Ever wondered why automation is part product/software testing process? Speed,

Efficiency, Accuracy, Cross-Browser testing or Coverage! Right?

But what about the challenges, complexity and learning curve that hits when it comes to automation?

And even after that are you sure your automation suite is complete, consistent and reliable?

Are you confident about the automation logics written? Are you ready to give your valuable time to scratch your mind in writing redundant steps again??

Gone are the early and blooming days of automation testing and what's the new cool now is "Automating the Automation".

And here I am with the exact required solution I believe the industry was hoping for – "**Auto**SQUARE", a software which will speed-up your automation scripting.

Firstly, let's get ourselves aware of what are the challenges in Automation testing:

- As the people belonging to the Selenium world might know what POM is, and how it's needed to create POM (Page Object Model) manually or by Selenium IDE. But Selenium IDE and many other recording tools providing export only create POM not factory methods for POM. This task is error-prone and time-consuming. For those who don't know POM, just imagine that in UI automation testing against each UI page in an application there is the corresponding page class that contains information of the fields of the page. [Selenium Ide is deprecated, and newest Firefox no longer support ide]
- Most of the time this custom-made architecture does not comply with the standard and therefore making resulting test suite obviously not reliable.
- Then comes the most "maneuvering" task of creating reports, and I bet this task does require minimal expertise to generate efficient reports.
- Internationalization!! Oh right, have you thought of what it would be like, to build your test suite that support's test case data in the different language's for internationalization testing. It's painful and time-consuming.
- Then you meet the challenge of parallel and cross-browser testing. The features and support of something may be present in browser x but may not be present in browser y and the head storming is not worth your time.
- Dockerization, the CI and CD support.
- Grid environment with parallel testing

Now let's talk about the solution, how exactly AutoSQUARE would be solving these problems.

**AutoSQUARE** is made up of different modules which provide you with the best quality code; let's explore them one by one.

### **AutoSQUARE POM Creator**

This is a custom build POM creator which has the capability to generate POM for recorded events with factory methods. Yes, it will create factory methods too, which will reduce your development time by 10% – 15%.

### **AutoSQUARE Data Factory**

Data Factory module deals with creating test data for you. Yes, it can create test data as well and that too in six languages (English, German, Spanish, French, Japanese, and Russian). It creates meaningful test data for regression testing.

### **AutoSQUARE Core**

This module serves the actual purpose; it brings all the services as a bundle and serves your needs end to end. It will convert your recorded test case into actual selenium code, it completes the runnable maven selenium project with TestNG, Extent Report, etc.

Let's talk about how we typically create a selenium script

- We study the page and create POM (sometimes Selenium IDE might help [here](#))
- Create factory methods for POM
- Create a new maven project and add selenium dependencies and earlier created POM classes to it.
- Create an Excel sheet which contains test data, write Data-Providers and plug data provider to test cases.
- Create Reports

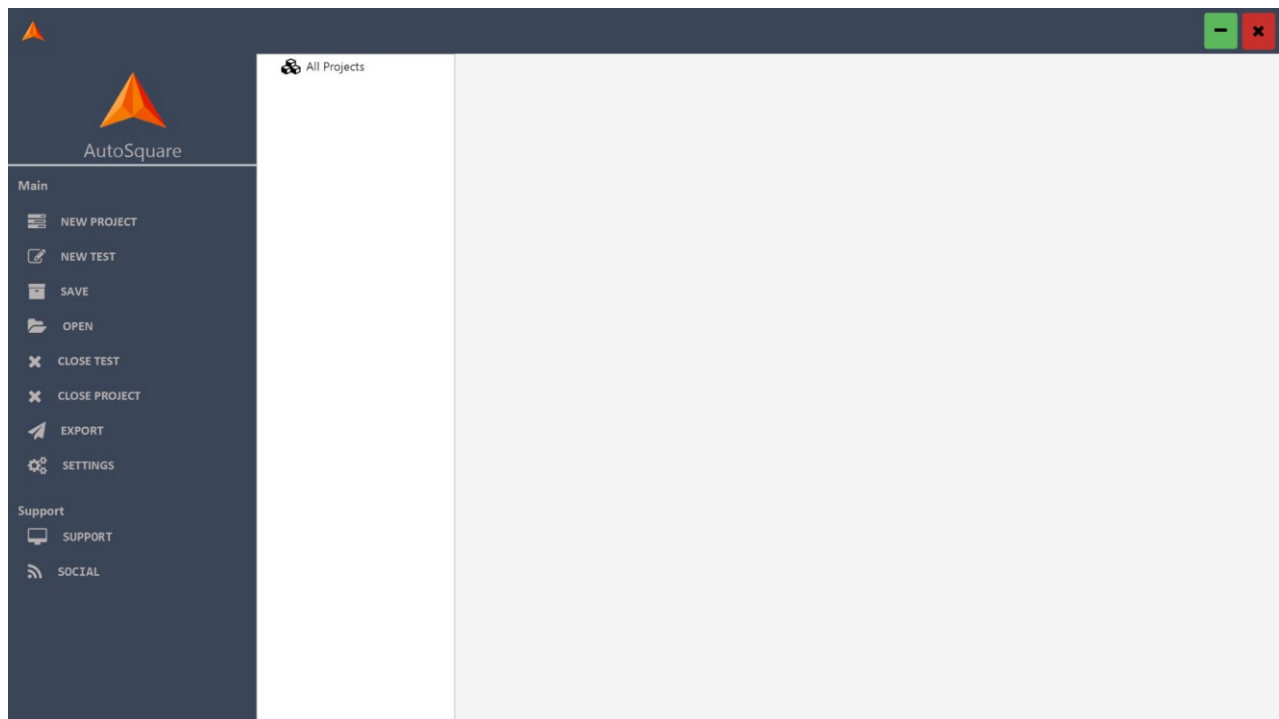
But what if I tell you, "AutoSQUARE" will do all these steps for you in a minute? Yes, that's true.

AutoSQUARE will do all these steps for you in a minute!

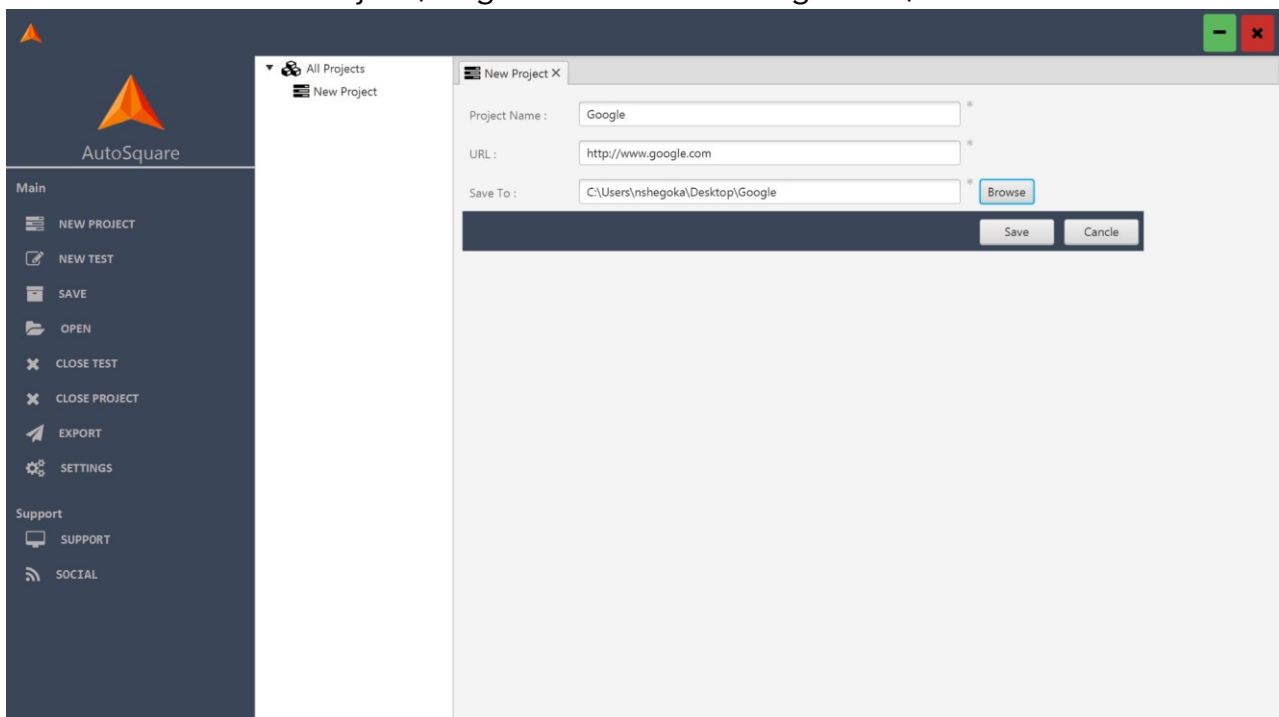
I would be happy to take you on an interesting ride of AutoSQUARE.

Here is how it is done.

1. Open Auto**SQUARE** (Ex: Google [we will create an automation script for google.com fordemo])



2. Click on New Project (Image Search Test on Google.com)



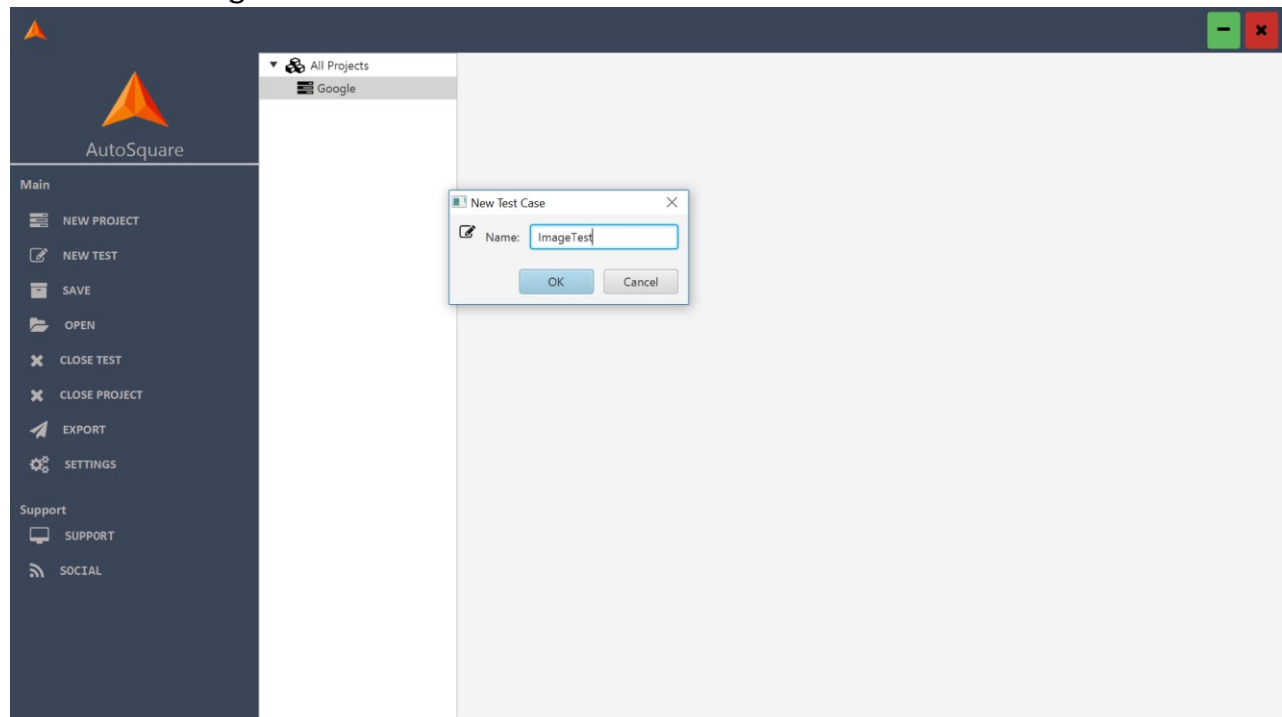
Provide new project details as below:

Project Name: Google

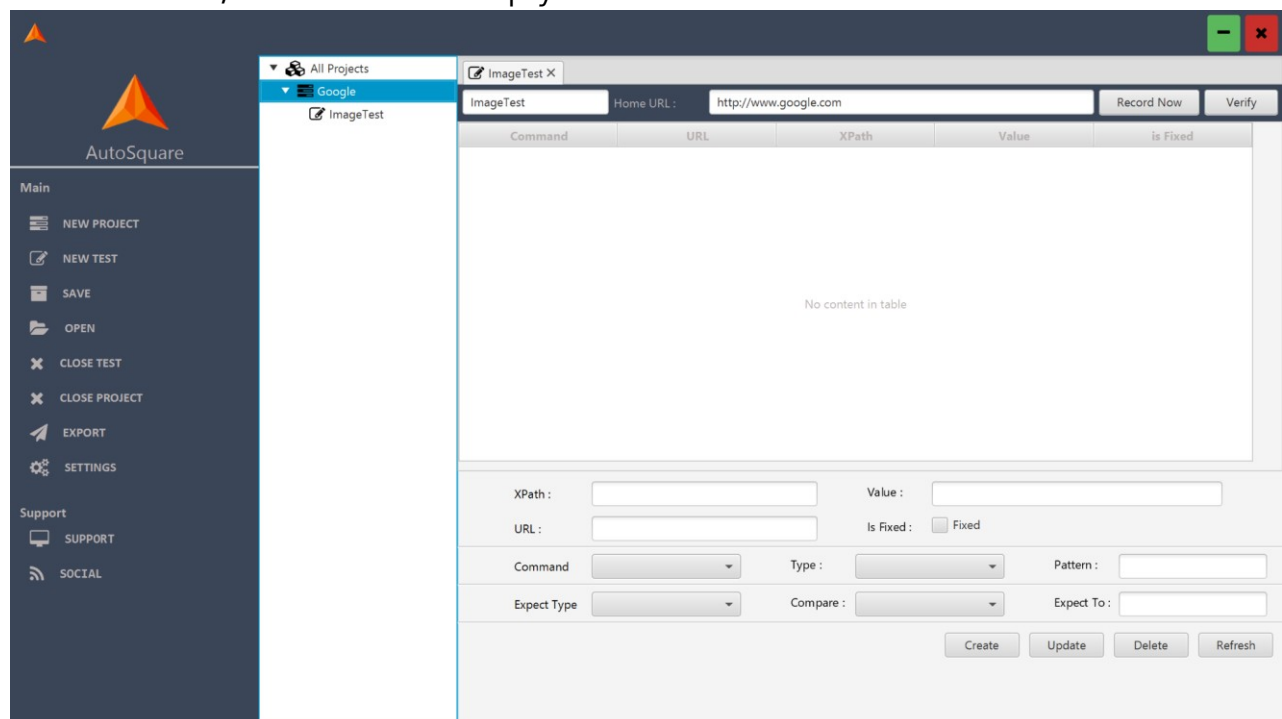
URL: <http://www.google.com>

Save To: Location to save this project

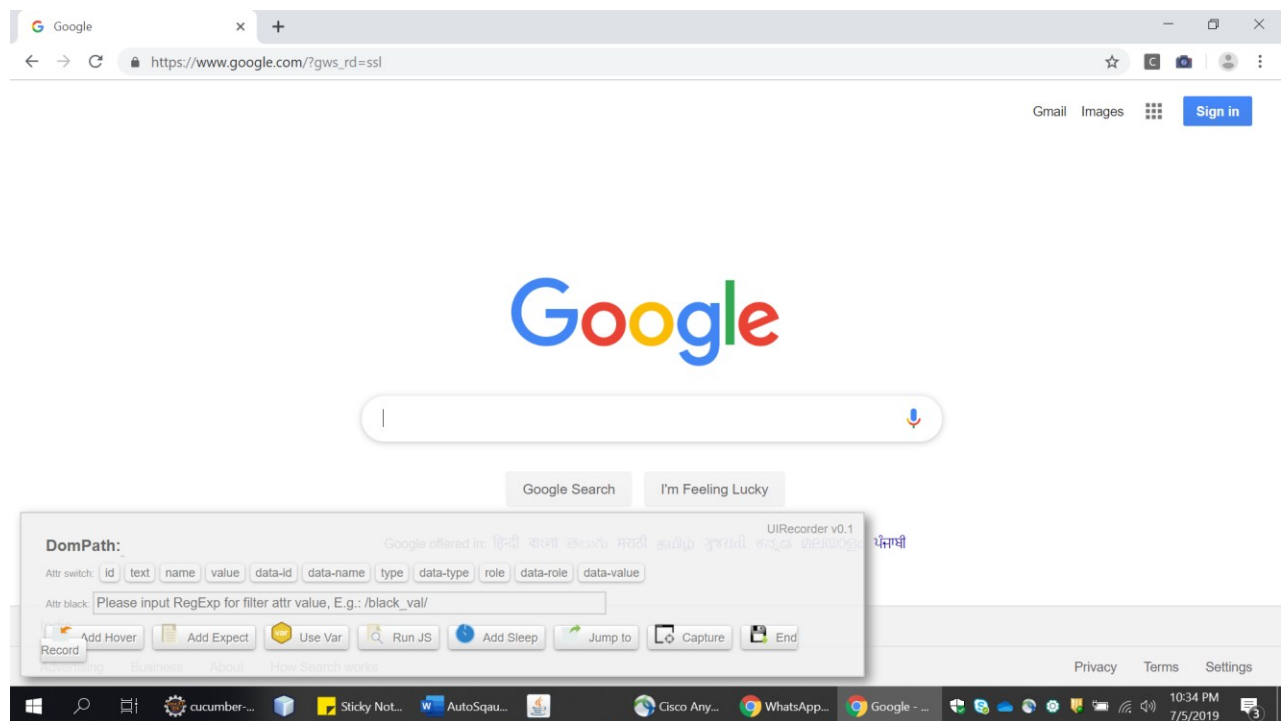
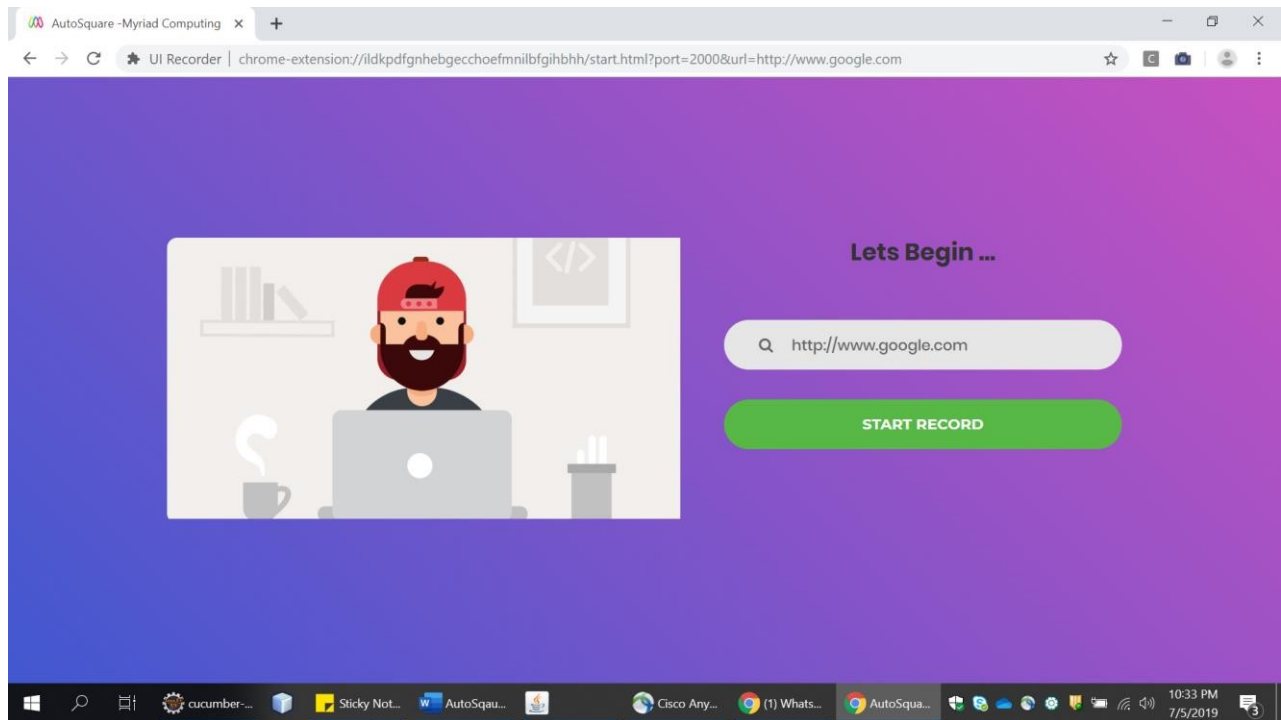
3. Now Click on New Test to create an actual test case and Enter the value as "ImageTest"



On click of OK; it will create an empty test case as below:



Click on "Record Now" to start recording a test case



Steve Jobs - Google Search

https://www.google.com/search?q=Steve+Jobs&source=lnms&tbn=isch&sa=X&ved=0ahUKEwjxOm4oZ7jAhWt7XMBHWqAAAkQ\_AUIECgB&biw=12...

Google

Steve Jobs

Sign in

SafeSearch

Q All Images Books News Videos More Settings Tools

wallpaper apple quote iphone death cartoon biography next pixar

Steve Jobs: Walter Isaacson ... Steve Jobs - Wikipedia Steve Jobs - Starting Apple, Family ... Steve Jobs - Starting Apple, Family ... Steve Jobs: Walter Isaacson ... amazon.com

DomPath: //span[text()='SafeSearch']

Attr switch: id text name value data-id data-name type data-type role data-role data-value

Attr black: Please input RegExp for filter attr value, E.g.: /black\_val/

Record Add Hover Add Expect Use Var Run JS Add Sleep Jump to Capture End

UIRecorder v0.1

10:35 PM 7/5/2019

Steve Jobs - Google Search

https://www.google.com/search?q=Steve+Jobs&source=lnms&tbn=isch&sa=X&ved=0ahUKEwjxOm4oZ7jAhWt7XMBHWqAAAkQ\_AUIECgB&biw=12...

Google

Steve Jobs

Sign in

SafeSearch

Q All Images Books News Videos More Settings Tools

wallpaper apple quote biography next pixar

Steve Jobs: Walter Isaacson ... Steve Jobs - Wikipedia

Steve Jobs: Walter Isaacson ... amazon.com

DomPath: //span[text()='SafeSearch']

Attr switch: id text name value data-id data-name type data-type role data-role data-value

Attr black: Please input RegExp for filter attr value, E.g.: /black\_val/

Record Add Hover Add Expect Use Var Run JS Add Sleep Jump to Capture End

UIRecorder v0.1

10:35 PM 7/5/2019

**Add Expect**

Delay Time: 300 ms

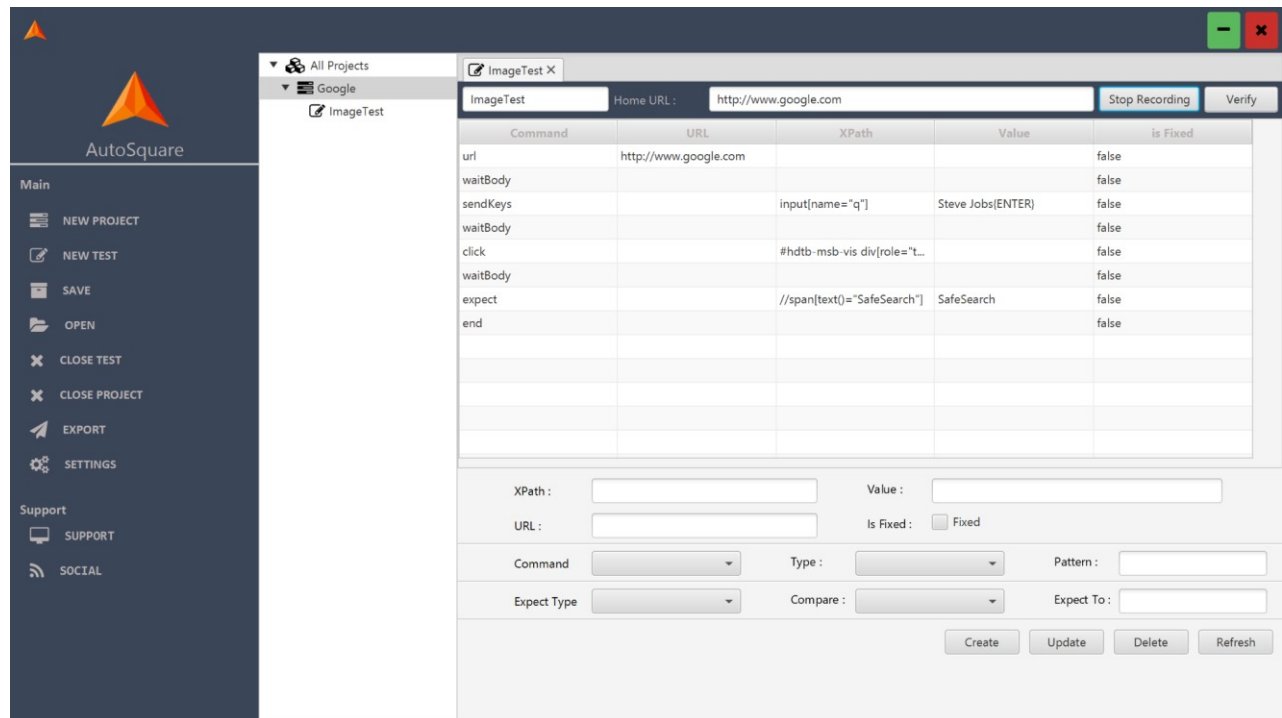
Expect Type: text

Expect DOM: //span[text()='SafeSearch']

Expect compare: equal

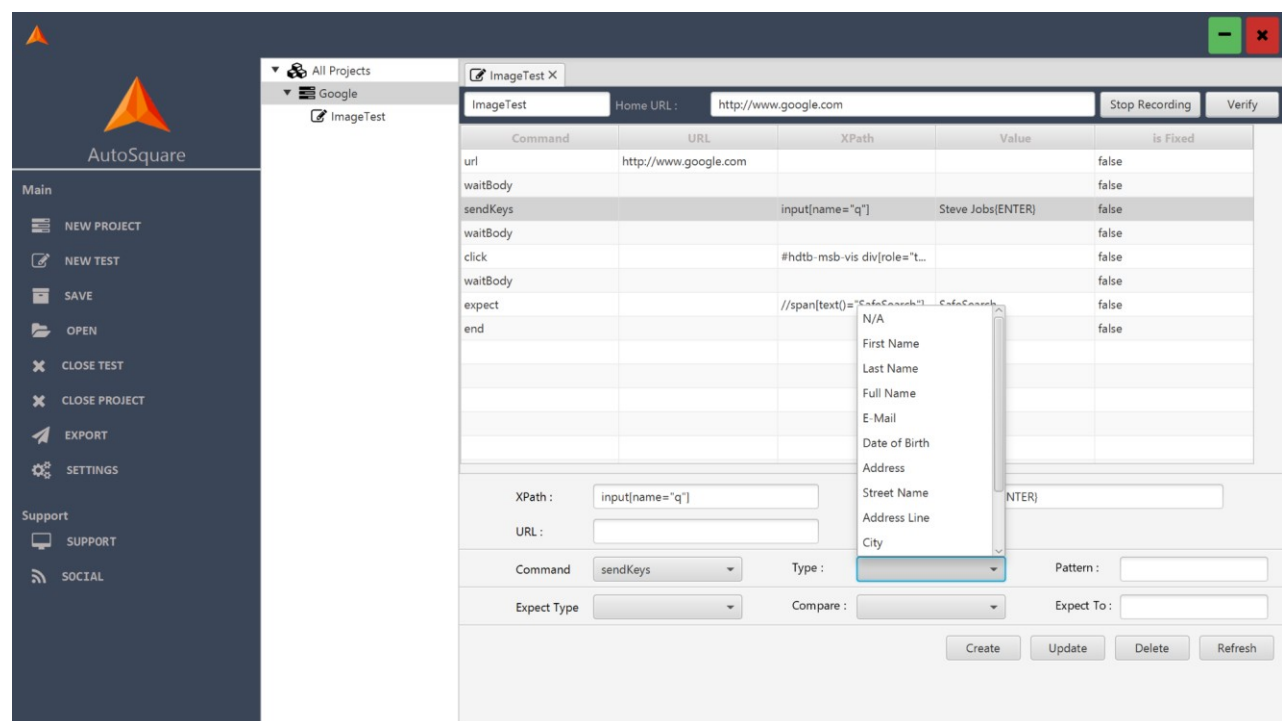
Expect to: SafeSearch

OK Cancel



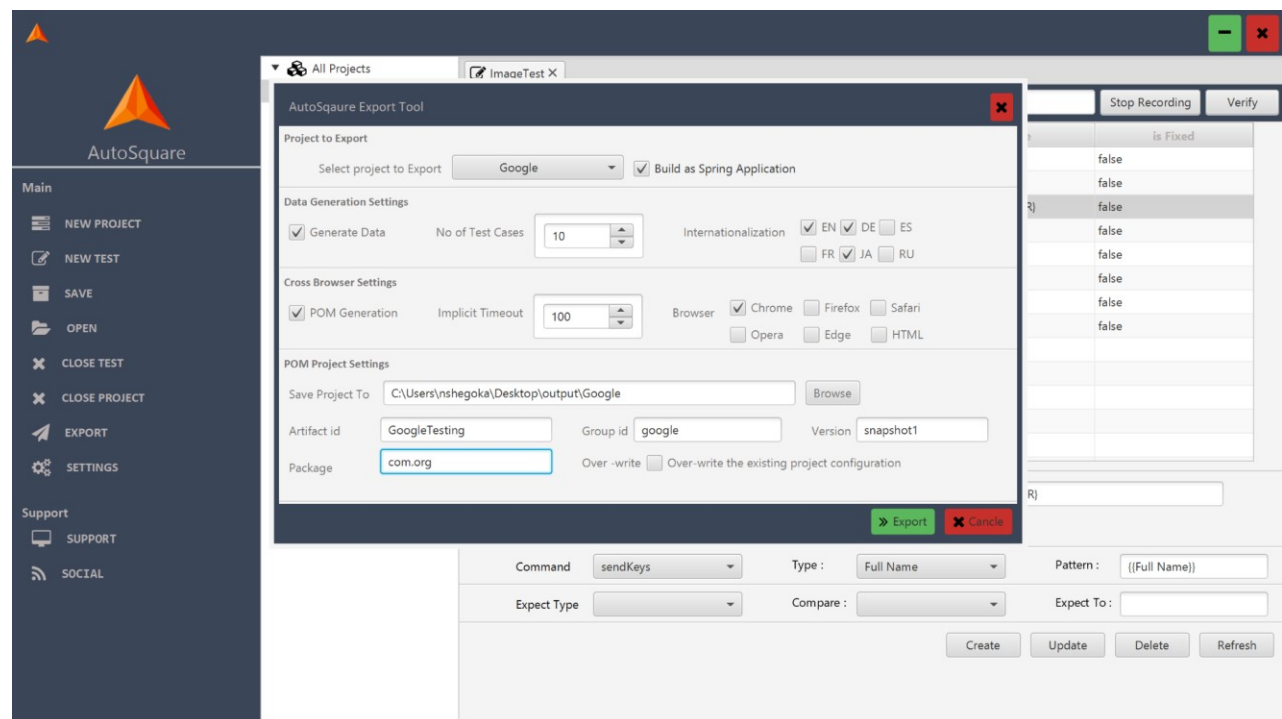
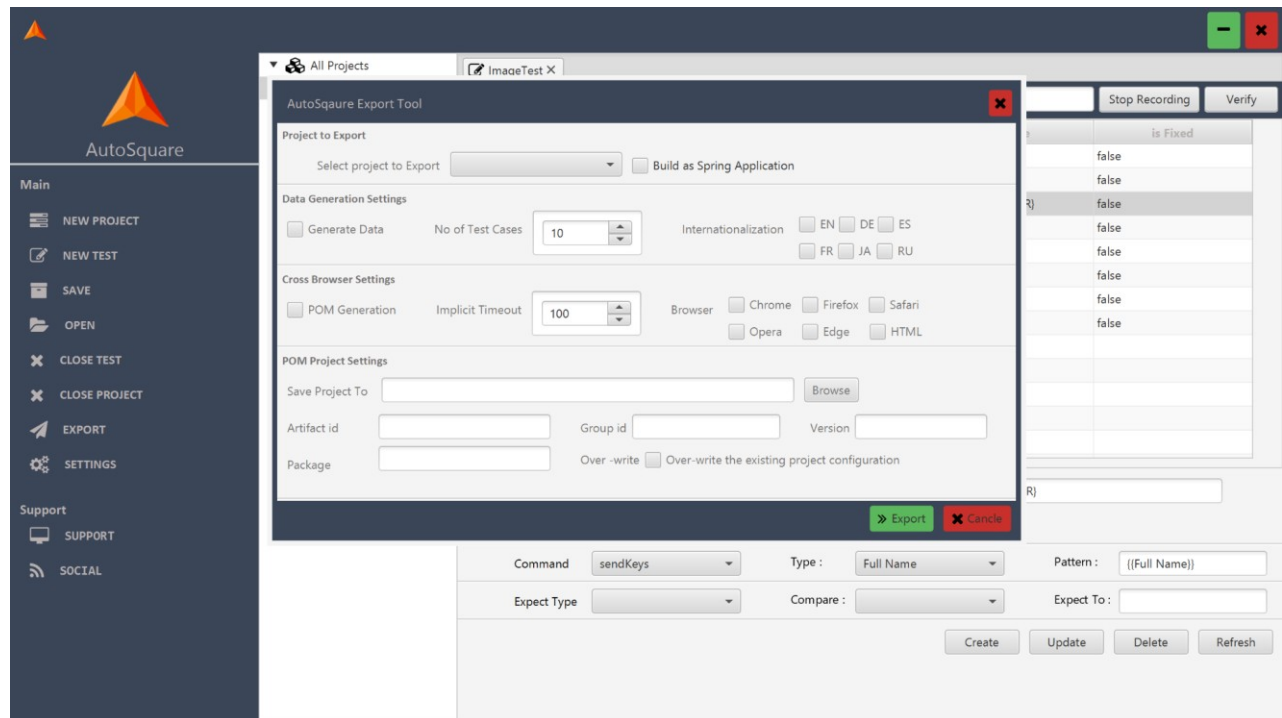
Once you are done with recording test case click on the STOP button.

- Select the field you want to edit; like provide the type of the field which will be used to create meaningful test case data for you.



Now you are ready to convert your testcase into actual Selenium script. Select the project and click Export to Selenium option





Just select the configuration you want for your project and you are good to go!

#### Test Data Generation:

1. Generate (Checkbox) will ensure creating test cases for you.
2. The number of test cases defines how many test cases it should create.
3. Internationalization defines test case should be in which language; EN is the default.

#### Cross-Browser Settings:

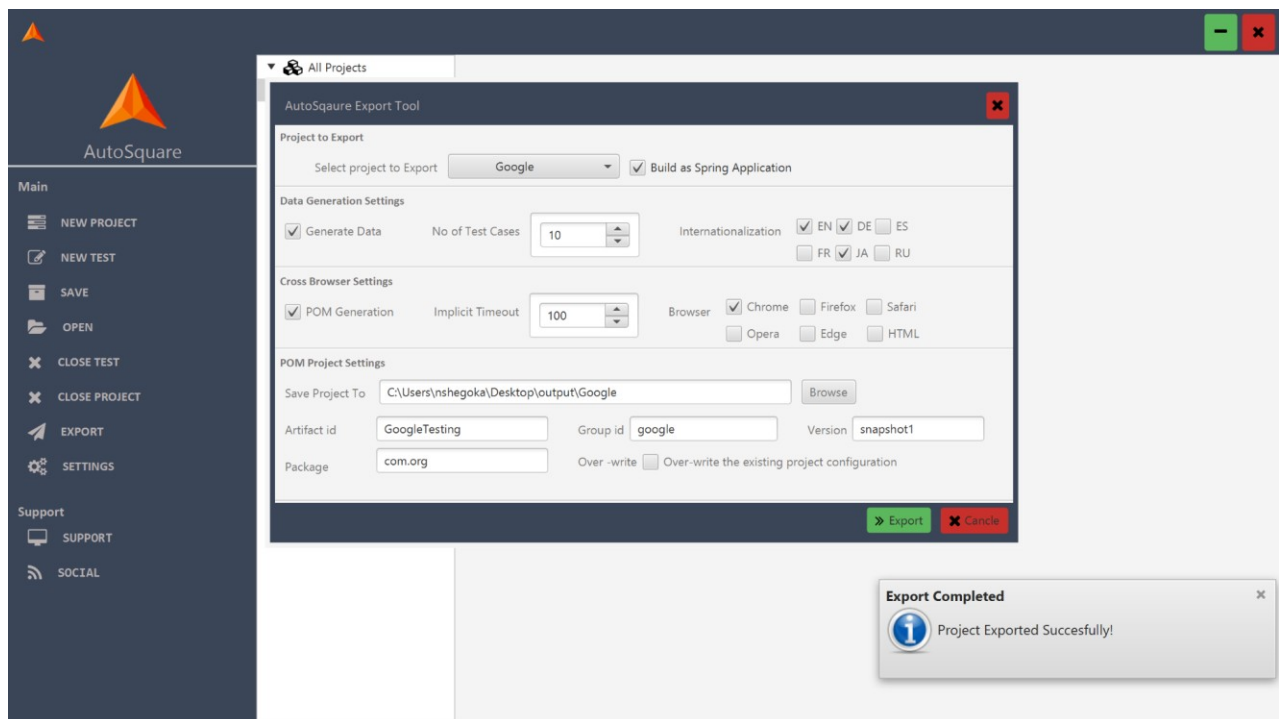


1. Cross Browsers checkboxes will ensure to write the configuration for you which will run test cases in multiple browsers.
2. Implicit Timeout: it will set implicit timeout property in configuration.

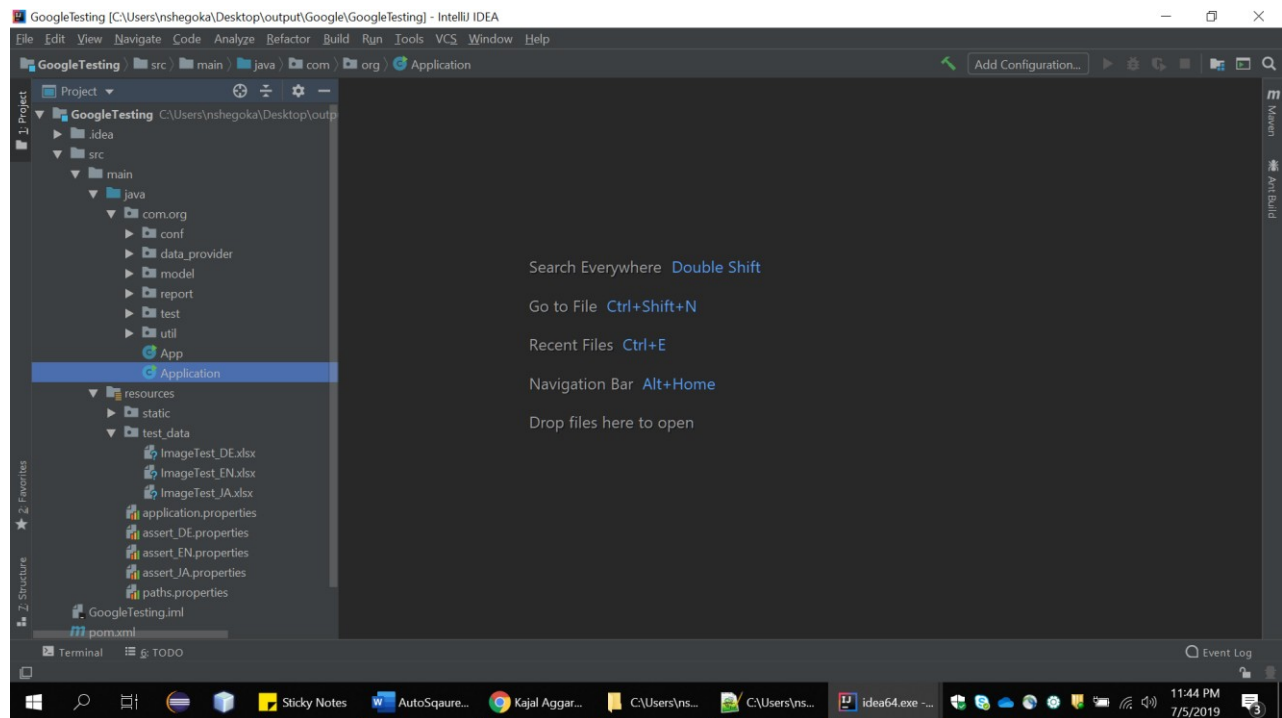
### Project Export Settings

1. Save the project to: it will let you browse where to store the generated project.
2. Artifact ID, Group ID, and Version: as AutoSQUARE generate maven project we need to provide maven details
3. Package: we need to provide the package path where POM classes will be stored.
4. Overwrite: in the case already project exists on given location it will overwrite the existing project.

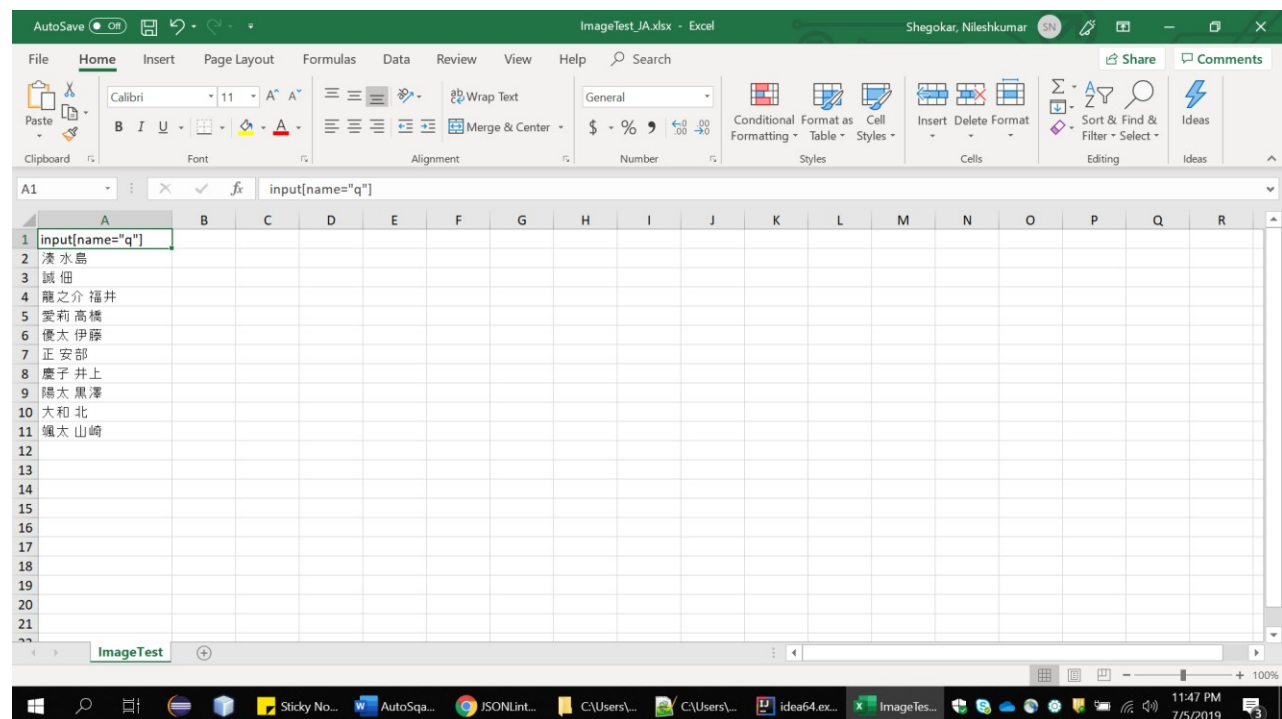
Click on EXPORT; once export completed it will show notification

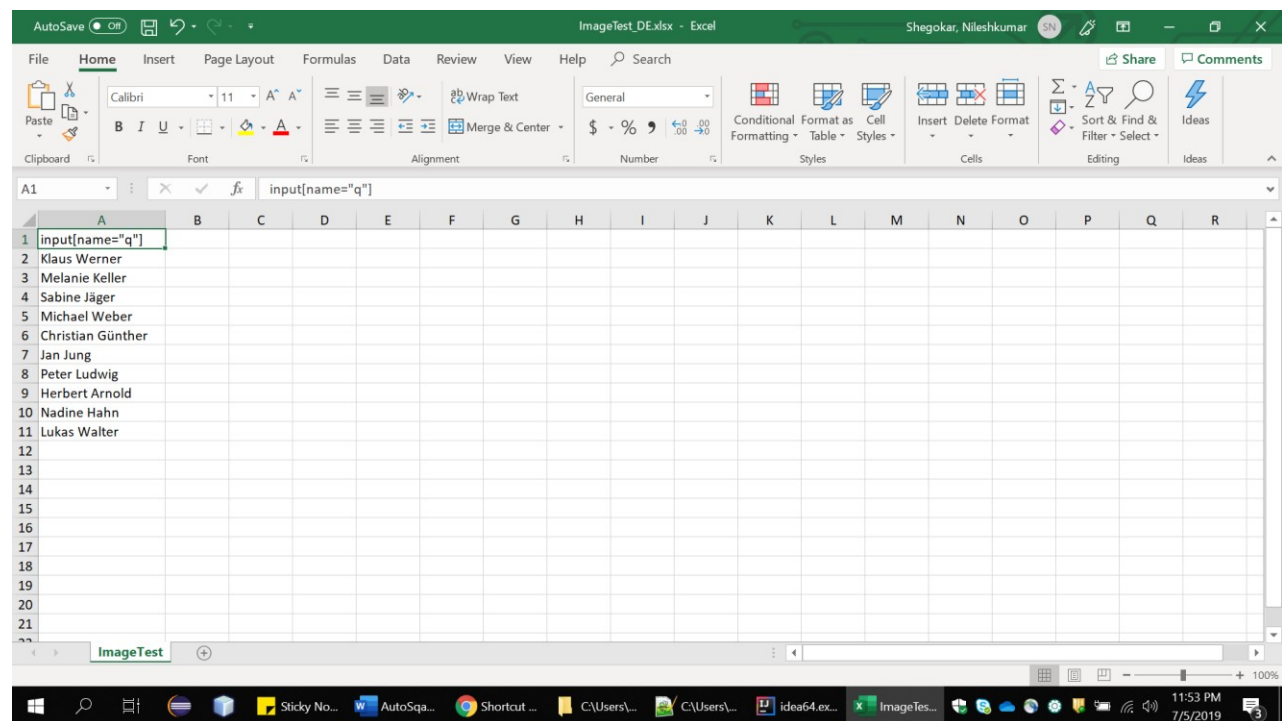
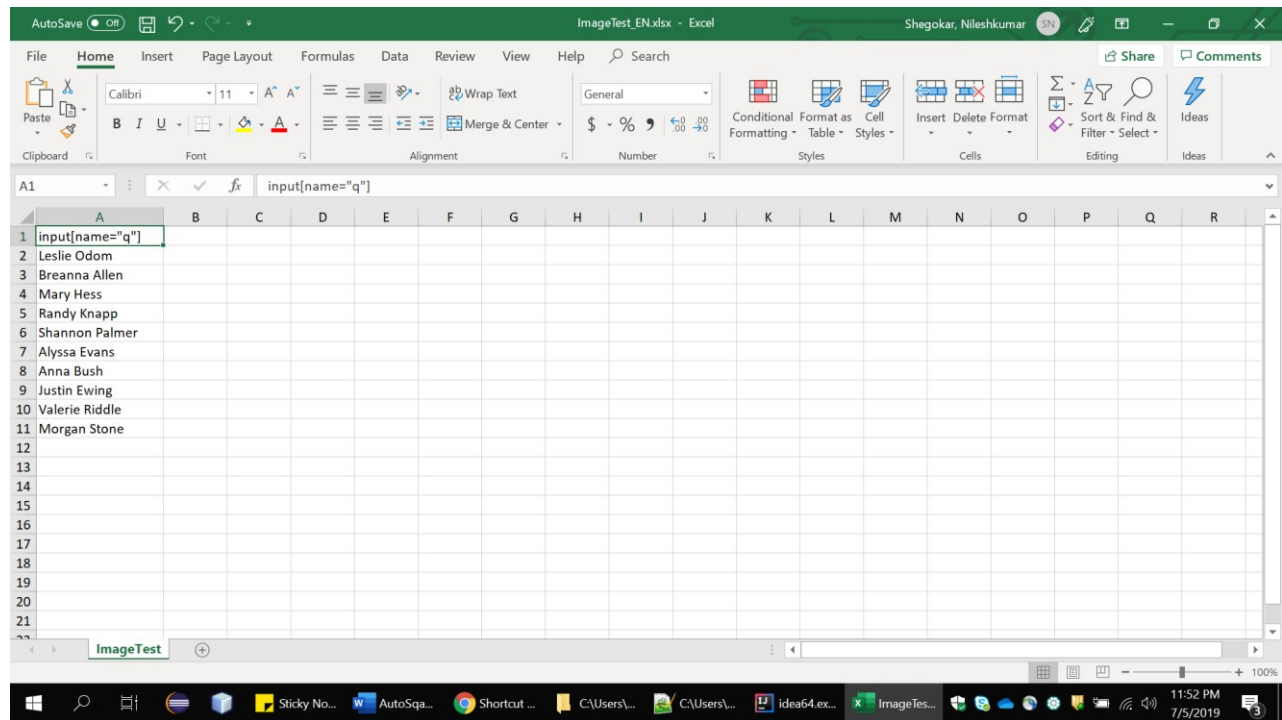


Import this to any of your favorite IDE as a maven project. Let us see what exactly AutoSQUARE has created for you:



Let's check what we have inside excel files as it must have our test data; it has randomly generated the testing data (in the different languages, as requested)





Let's run the code now:

Starting ChromeDriver 2.38.551601  
(edb21f07fc70e9027c746edd3201443e011a61ed) onport 34557 Only local  
connections are allowed.  
Jul 16, 2018 4:18:53 PM  
org.openqa.selenium.remote.ProtocolHandshake createSession INFO:  
Detected dialect: OSS



## Default Suite

Total tests run: 10, Failures: 0, Skips: 0

The screenshot displays the AutoSquare Report Tool interface, which is a web application running on a local host. The interface features a dark blue sidebar on the left with a 'Feature' label. The main content area is light gray and contains several colored buttons: a blue 'Start Test Suite' button with a rocket icon, a green '0 Tests Passed' button with a checkmark icon, a red '0 Tests Failed' button with an 'X' icon, and a yellow '0 Tests Skipped' button with a question mark icon. Below these buttons are two white boxes labeled 'Environment' and 'Host', each with a red circular icon. The interface is viewed through a web browser window with the address bar showing 'localhost:8080/index.html#/feature'. Below the browser window, a Google search page is visible, showing the Google logo, a search bar, and buttons for 'Google Search' and 'I'm Feeling Lucky'. The page also includes a footer with links for 'Advertising', 'Business', 'About', 'How Search works', 'Privacy', 'Terms', and 'Settings'. The system tray at the bottom of the screen shows the date and time as 12:21 AM on 7/6/2019.

AutoSquare Report Tool

localhost:8080/index.html#/feature

Feature

Start Test Suite  
Build Suite

0 Tests Passed

0 Tests Failed

0 Tests Skipped

Environment

Host

Google

https://www.google.com/?gws\_rd=ssl

Chrome is being controlled by automated test software.

Gmail Images Sign in

Google

Google Search I'm Feeling Lucky

Google offered in: हिन्दी বাংলা తెలుగు मराठी தமிழ் ગુજરાતી ಕನ್ನಡ മലയാളം ਪੰਜਾਬੀ

India

Advertising Business About How Search works Privacy Terms Settings

12:21 AM 7/6/2019

AutoSquare Report Tool

localhost:8080/index.html#/feature

Feature

Start Test Suite  
Build Suite

1  
Tests Passed

9  
Tests Failed

0  
Tests Skipped

Pass com.org.test.ImageTest : test Leslie Odom

Test Name : test  
Test Start Time : 1562352688892  
Test End Time : 1562352775265  
Invocation count : 1  
Class : com.org.test.ImageTest  
Arguments : Leslie Odom  
Status : 1  
Success Percentage : 100%

Fail com.org.test.ImageTest : test Breanna Allen

Test Name : test  
Test Start Time : 1562352775292  
Test End Time : 1562352896576  
Invocation count : 2  
Class : com.org.test.ImageTest  
Arguments : Breanna Allen  
Status : 2  
Success Percentage : 100%

Environment  
Windows

Host  
172.20.10.2

localhost:8080/index.html#collapseOne

Sticky Notes

AutoSquar...

Google Chr...

C:\Users\ins...

C:\Users\ins...

idea64.exe ...

12:27 AM  
7/6/2019