**STQA Mini Project No. 1**

**1.1 Title**

Mini-Project 1: Create a small application by selecting relevant system environment/ platform and programming languages. Narrate concise Test Plan consisting features to be tested and bug taxonomy. Prepare Test Cases inclusive of Test Procedures for identified Test Scenarios. Perform selective Black-box and White-box testing covering Unit and Integration test by using suitable Testing tools. Prepare Test Reports based on Test Pass/Fail Criteria and judge the acceptance of application developed.

**1.2 Problem Definition:**

Perform Desktop Application testing using Automation Tool like QTP.

**1.3 Prerequisite:**

Knowledge of Core Java, Basic Concepts of Unit Testing, Test Cases Writing using QTP tool.

**1.4 Software Requirements:**

HP UFT IDE tool.

**1.5 Hardware Requirement:**

PIV, 2GB RAM, 500 GB HDD, Lenovo A13-4089Model.

**1.6 Learning Objectives:**

We are going to learn how to Prepare Test Cases inclusive of Test Procedures for identified Test Scenarios.Perform selective Black-box and White-box testing covering Unit and Integration test by using suitable Testing tools. Also Prepare Test Reports based on Test Pass/Fail Criteria

**1.7 Outcomes:**

You are able to understand Unit and Integration testing with Tool with Test Report.

**1.8 Theory Concepts:**

**1.8.1 What is Unit Testing?**

Unit Testing of software applications is done during the development (coding) of an application.

The objective of Unit Testing is to isolate a section of code and verify its correctness. In procedural programming a unit may be an individual function or procedure

The goal of Unit Testing is to isolate each part of the program and show that the individual parts are correct. Unit Testing is usually performed by the developer.

**1.8.2 Unit Testing Tools**

There are several automated tools available to assist with unit testing. We will provide a few examples below:

1. [Jtest:](https://prsft.co/2n7GdAM) ParasoftJtest is an IDE plugin that leverages open-source frameworks (Junit, Mockito,PowerMock, and Spring) with guided and easy one-click actions for creating, scaling, and maintaining unit tests. By automating these time-consuming aspects of unit testing, it frees the

developer to focus on business logic and create more meaningful test suites.

1. [Junit:](https://www.guru99.com/junit-tutorial.html) Junit is a free to use testing tool used for Java programming language. It provides assertionsto identify test method. This tool test data first and then inserted in the piece of code.
2. [NUnit:](http://nunit.org/) NUnit is widely used unit-testing framework use for all .net languages. It is open sourcetool which allows writing scripts manually. It supports data-driven tests which can run in parallel.
3. [JMockit:](http://jmockit.github.io/index.html) JMockit is open source Unit testing tool. It is code coverage tool with line and pathmetrics. It allows mocking API with recording and verification syntax. This tool offers Line coverage, Path Coverage, and Data Coverage.
4. [EMMA:](http://emma.sourceforge.net/) EMMA is an open-source toolkit for analyzing and reporting code written in Javalanguage. Emma support coverage types like method, line, basic block. It is Java-based so it is without external library dependencies and can access to the source code.
5. [QTP:](https://phpunit.de/) QTP stands for Quick Test Professional, a product of Hewlett Packard (HP). This toolhelps testers to perform an automated functional testing seamlessly, without monitoring, once script development is complete. HP QTP uses Visual Basic Scripting (VBScript) for automating the applications.

These are few of the available unit testing tools.

**1.8.3 Extreme Programming & Unit Testing**

Unit testing in Extreme Programming involves the extensive use of testing frameworks. A unit test framework is used in order to create automated unit tests. Unit testing frameworks are not unique to extreme programming, but they are essential to it. Below we look at some of what extreme programming brings to the world of unit testing:

* Tests are written before the code
* Rely heavily on testing frameworks
* All classes in the applications are tested
* Quick and easy integration is made possible

**1.8.4 Bug taxonomy**

Bug taxonomies help in providing fast and effective feedback so that they can easily identify possible reasons for failure of the software. Using bug taxonomy, a large number of potential bugs can be grouped into few categories.

Whenever a new bug is reported, using bug taxonomy, a tester can easily analyse and put that bug into any of these categories.

At the end of testing, Testers can understand the type of categories of bugs that frequently occurred and thereby in successive rounds of testing he can focus on writing more test cases that would help to detect such bugs. In addition, test leaders can guide their testers to focus on such frequently occurring b

The summary of the Bug Taxonomy is given below.

* Requirements, Features, and Functionality Bugs
* Structural Bugs
* Data Bugs
* Coding Bugs
* Interface, Integration, and System Bugs
* Test and Test Design Bugs
* Testing and Design Style

**1.8.5 What is Integration Testing?**

In integration Testing, individual software modules are integrated logically and tested as a group. A typical software project consists of multiple software modules, coded by different programmers. integration Testing focuses on checking data communication amongst these modules. Hence it is also termed as 'I & T' (Integration and Testing), 'String Testing' and sometimes 'Thread Testing

**Integration Test Case:**

Integration [Test Case](https://www.guru99.com/test-case.html)differs from other test cases in the sense it **focuses mainly on the interfaces & flowof data/information between the modules**. Here priority is to be given for the **integrating links** ratherthan the unit functions which are already tested.

Sample Integration Test Cases for the following scenario: Application has 3 modules say 'Login Page', 'Mail box' and 'Delete mails' and each of them are integrated logically.

Here do not concentrate much on the Login Page testing as it's already been done in [Unit Testing.](https://www.guru99.com/unit-testing-guide.html) But check how it's linked to the Mail Box Page.

Similarly Mail Box: Check its integration to the Delete Mails Module.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test** | **Test Case Objective** | **Test Case Description** |  | **Expected Result** | |  |
| **Case ID** |  |  |
|  |  |  |  |  |  |
|  |  |  | |  |  |  |
| **1** | Check the interface link between the | Enter login credentials and click | | To be | directed | to the |
| Login and Mailbox module | on the Login button |  | Mail Box | |  |
|  |  |  |
|  |  |  |  |  |  |  |
|  | Check the interface link between the | From Mail box select the | an | Selected | email | should |
| **2** |  | in | the |
| Mailbox and Delete Mails Module | email and click delete button |  | appear |
|  |  | Deleted/Trash folder | | |
|  |  |  |  |
|  |  |  |  |  |  |  |

**1.8.6 Desktop Application Testing by Using Junit Tool**

**What is QTP?**

HP QTP is an automated functional[Testing](https://www.guru99.com/software-testing.html)tool that helps testers to execute automated tests in order to identify any errors, defects or gaps in contrary to the expected results of the application under test. It was designed by Mercury Interactive and later on acquired by HP. Full form of QTP is Quick Test Professional and full form of UFT is Unified Functional Testing.

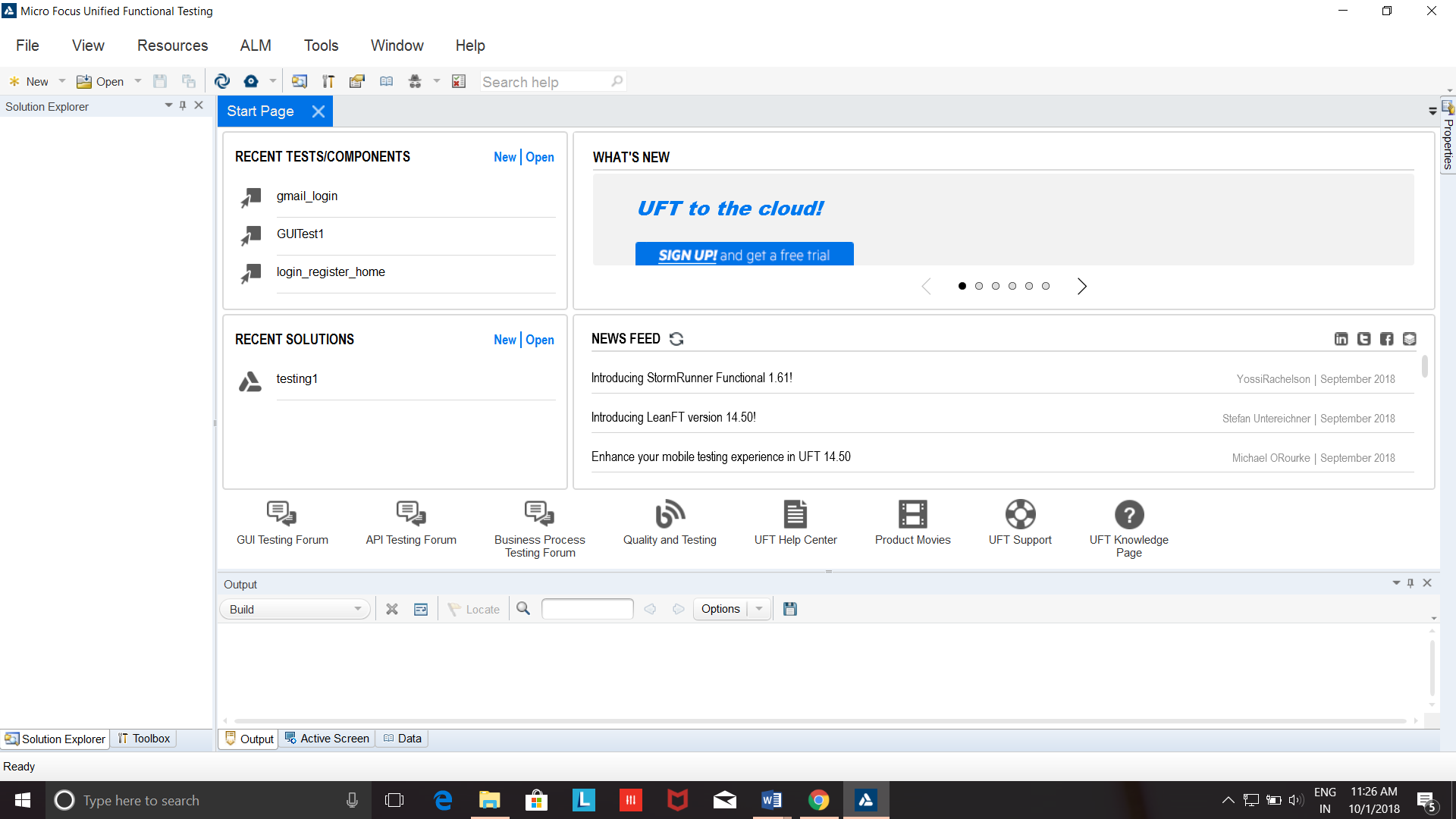
**Advantages of QTP Automation**

* It supports record and playback
* It uses an active screen to record scripts and helps tester in referring the screen object properties
* It has excellent object identification process or mechanism
* It supports different add-ins like Oracle, Java, SAP, NET, Web Forms, People soft, etc..
* It allows you to enhance the existing tests even without the AUT through an active screen
* It supports popular automation frameworks- keyword driven testing approach, modular testing approach, data-driven testing approach, etc..
* It comes with an inbuilt IDE
* It can be integrated with Test management tools like Quality Center, Test Director, and Wind runner
* Different types of suites like Smoke, Regression, Sanity can be easily maintained
* It supports XML
* Test reporting is possible through QTP for analysis purpose
* Easy to maintain

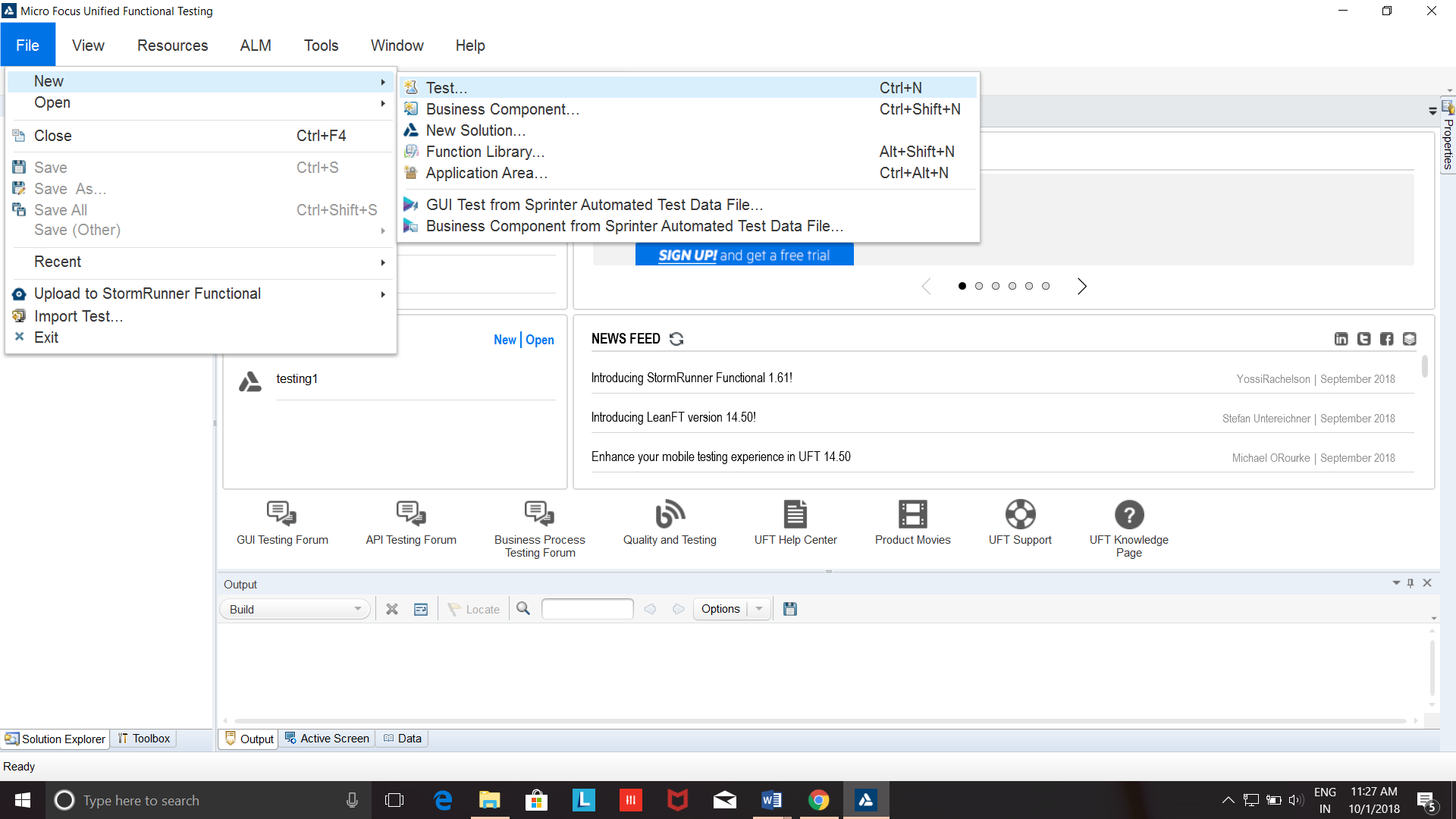
**1.8.7 How to Create Simple QTP Test.**

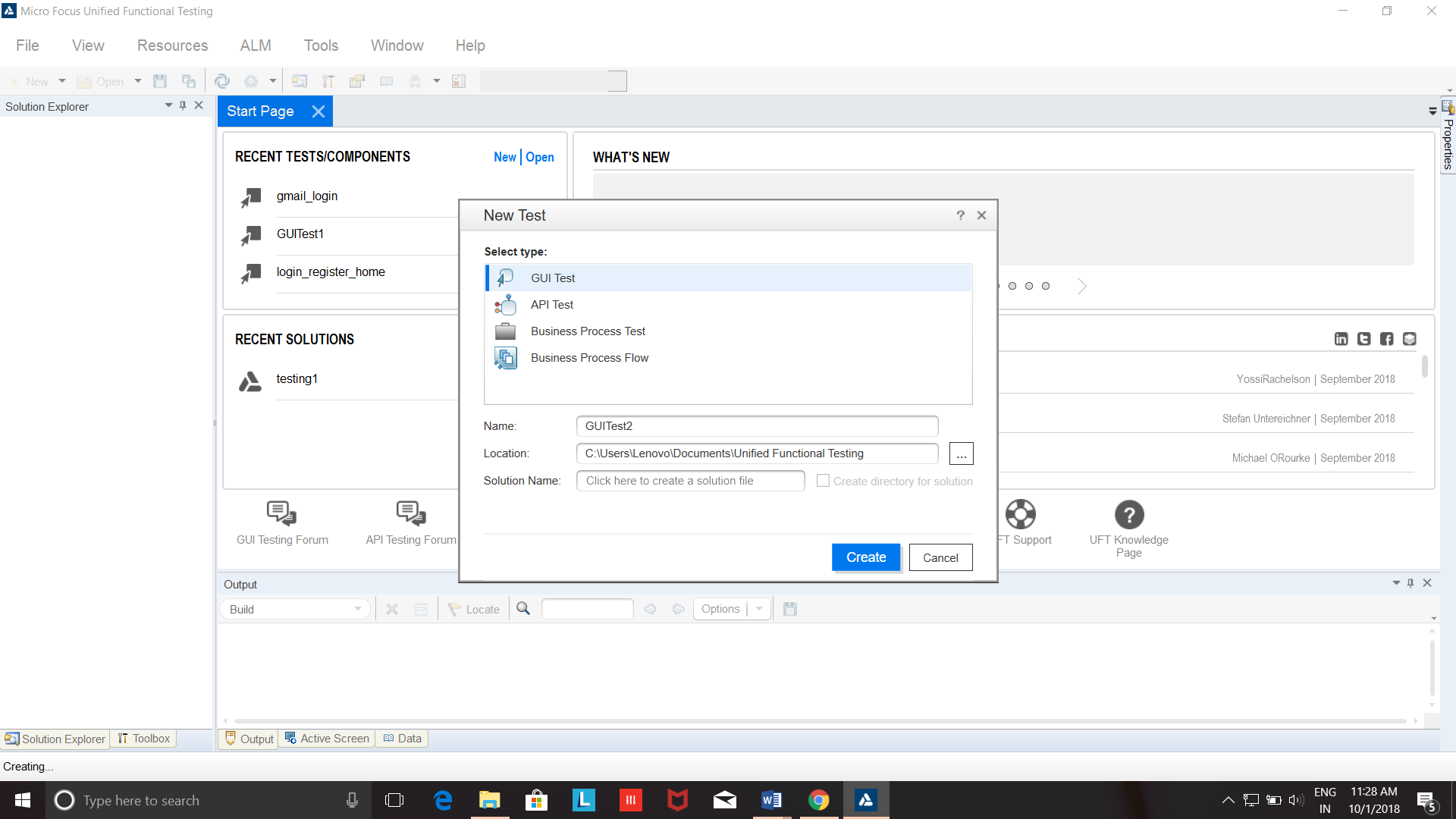
1. **Download QTP IDE tool from Micro Focus website.**

**2. Open QTP IDE**

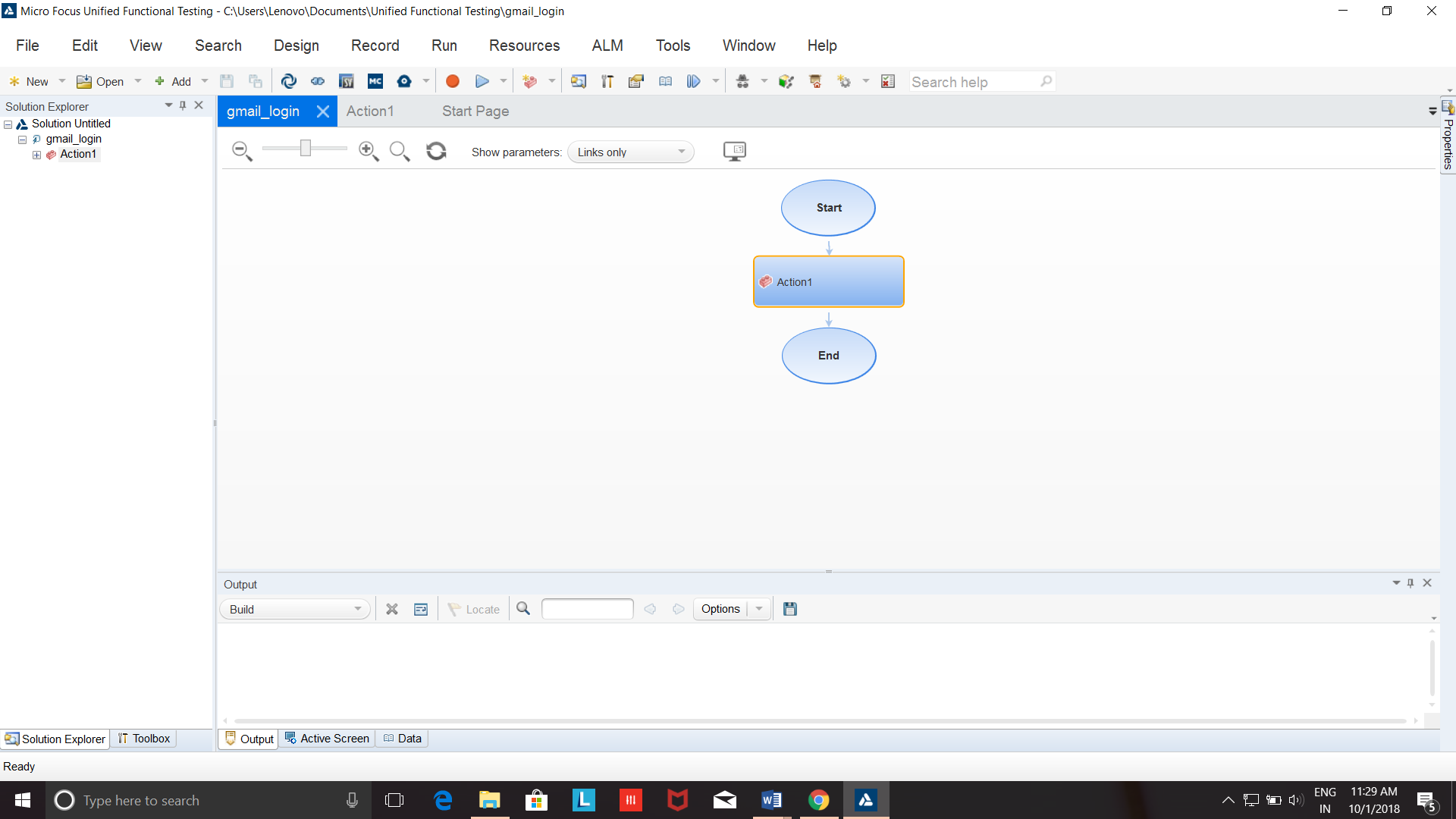


**3**. **Go to file->Create new test case.**

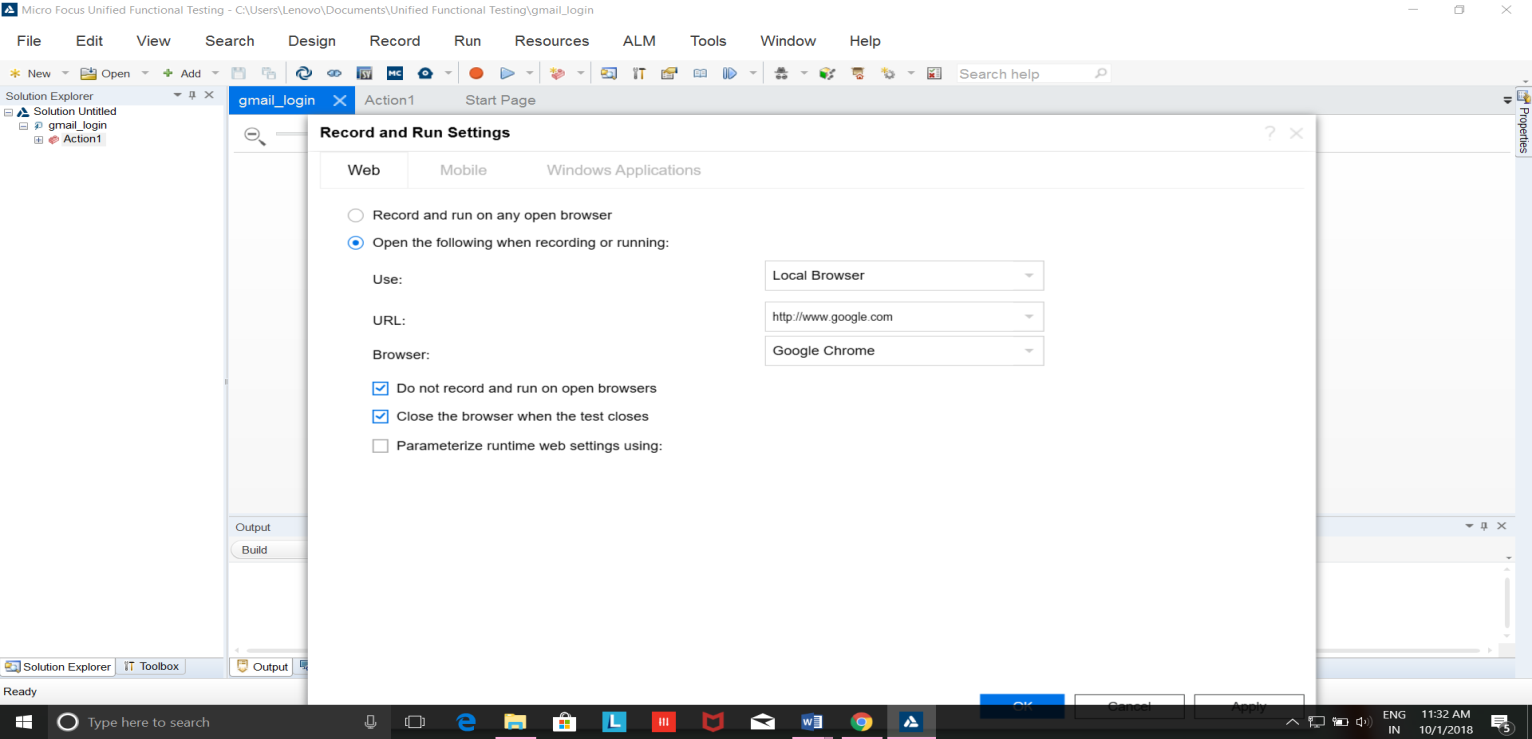




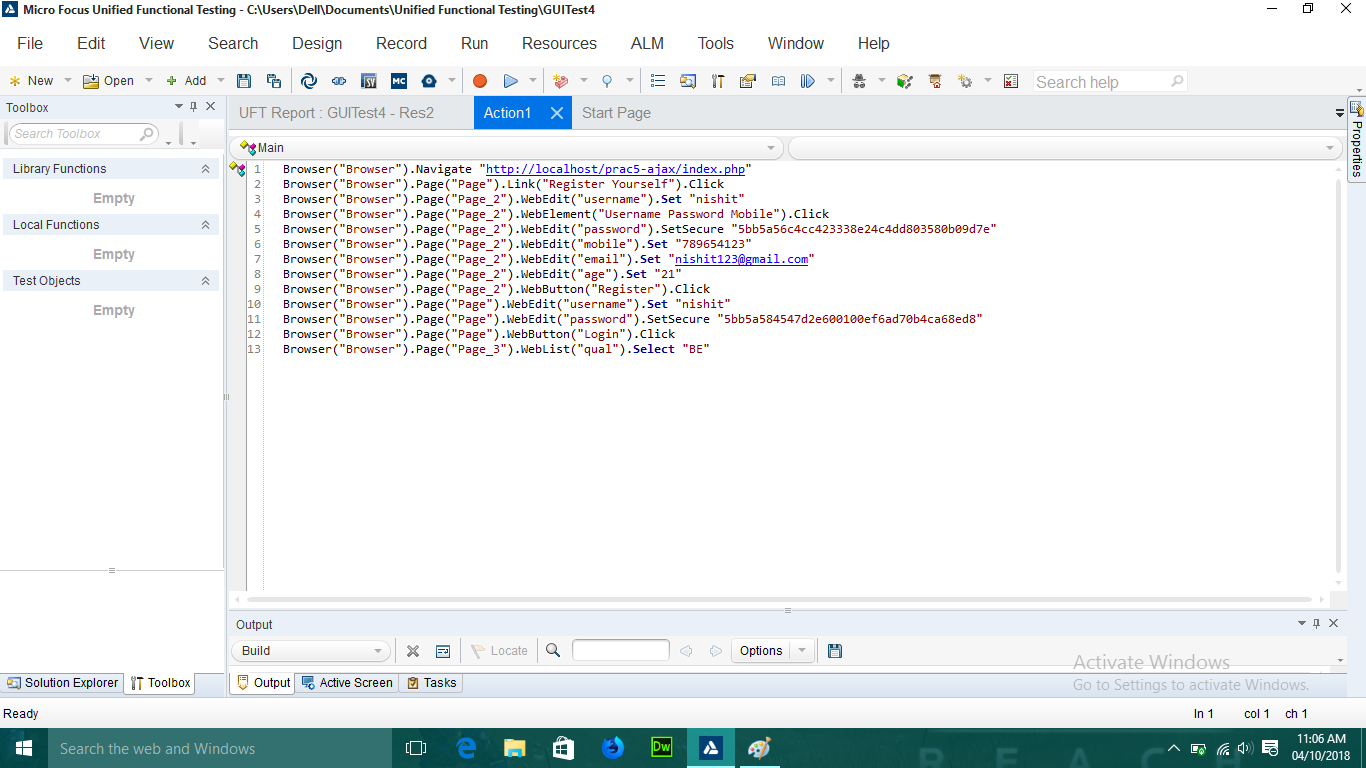
**4. Now a new test case will appear with action and start page.**



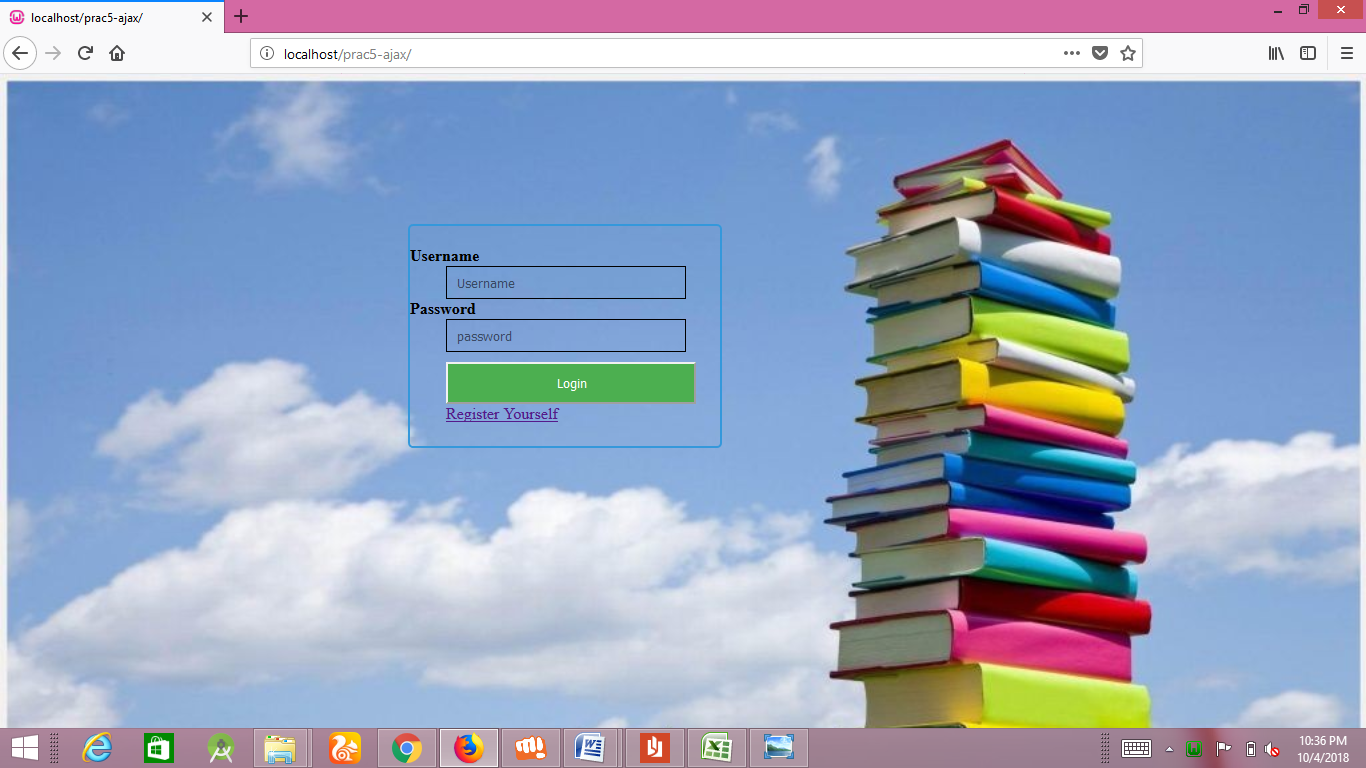
**5. Go to record and run a web link**

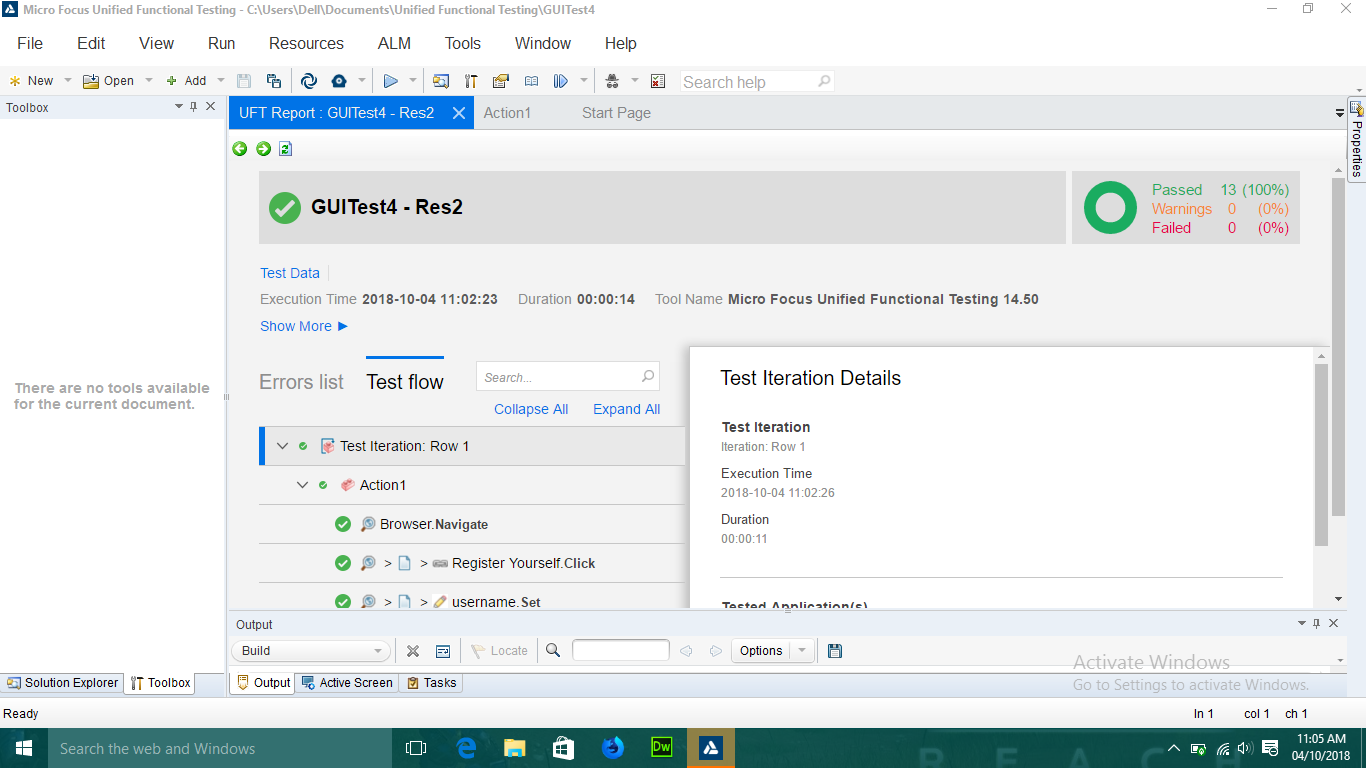


**6. Run the test cases for Website.**

****

**7. Passed test cases of wesite.**

****

****