Planning a Software Test Automation Framework for Windows Application utilizing Coded UI in Visual Studio Tool and Page Object Design

Deepak G R[1], Vinayprasad MS[2]

[1,2]*Department of Electronics and Communiation, Jss science and technology university, SJCE, Mysuru*

Abstract- **In the current Digital data age, Software enterprises, government association and other IT ventures has the need of business over the world has concentrated on quality and unwavering quality conveyed programming through process and innovation. Nature of programming is given with more noteworthy consideration, which is bringing about decrease of programming mistake keeping in mind the end goal to make quality programming. In programming building pattern, programming testing assumes a noteworthy part. Programming testing isn't an independent procedure; it is arrangement of ventures for discovering surrenders that is fluctuation between expected outcome and genuine outcome. This work contributes the examination which incorporates hypothetical viewpoints about different programming testing procedures, a dialog of various testing instruments and its working by taking down to earth case. Through this paper, we examine that how the experiments are arranged and applying in programming application through manual and robotization testing.**

Index Terms- Test automation, Coded UI, Test automation framework, Page object model.

1. INTRODUCTION

Testing is characterized as a procedure of assessment that either the particular framework meets its initially indicated prerequisites or not. Along these lines, this action brings about a contrast amongst real and expected outcome. Programming testing alludes to discovering bugs, blunders or missing prerequisites in the created framework or programming. Testing is broadly utilized as a part of industry for quality affirmation, this is an examination that furnishes the partners with the correct information about the nature of the item [1]. Programming testing is an unavoidable piece of the product improvement lifecycle, and keeping in accordance with its criticality in the pre-and post-advancement process. Programming testing should be possible by two different ways, one is manual and other is mechanization.

In manual testing, test cases are executed physically with no help from apparatuses or contents. To ensure fulfilment of testing, the analyser frequently takes after a composed test arrange for that leads them through an arrangement of essential experiments. An exceptionally difficult section of programming testing is programming trying robotization. This falls between programming testing and programming advancement, utilizing both programming ideas and in addition testing. Numerous associations have made the change from manual to computerized programming testing. In Automated programming testing experiments are executed with the help of instruments, contents and programming. Tests are robotized to accelerate execution cycles, give provoke input, free analysers from dreary errands and decrease human exertion.

2. MANUAL TESTING

Manual testing is the most settled and most thorough sort of programming testing. In this, product analysers physically execute test cases without utilizing any computerization devices. It requires an analyser to perform manual test activities on the product application without the assistance of Test mechanization. Manual testing is the most crude of all testing writes and helps discover deserts in the product framework.

3. AUTOMATION TESTING

Each Organization has one of a kind purpose behind mechanizing programming quality exercises, however a few reasons are normal crosswise over businesses. Mechanized testing devices are capacity of executing tests, portrays results and gauge comes about with prior trials. Tests finished with these apparatuses can be kept running on and on whenever. The strategy is being utilized to actualize robotization is known as a test computerization structure. A few structures have been produced throughout the years by business merchants and testing associations. Progressive advancement cycles will require execution of same test suite rehash commonly. Utilizing a test robotization apparatus it's conceivable to record this test suite and replay it as required. Robotization testing is imperative on the grounds that

• It enhances proficiency of testing.

• Reducing testing costs.

•Replicating testing crosswise over various stages.

• To give steady and exact outcomes.
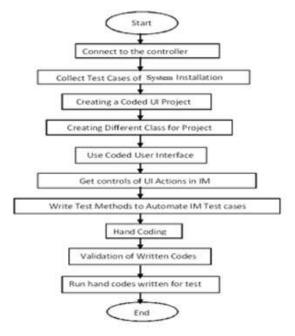
4. METHODOLOGY USED TO RUN AUTOMATION TEST CASES

Fig1:Steps used to execute automation test cases.

Fig 1 clarifies about the hand coding on computerization test cases PC is associated with a controller and gather the experiments of Related to the establishment supervisor, make a coded UI venture and particular class at that point utilize the coded UI to interface all the task and get controls on

the UI activities in the establishment director. In the test strategy hand coded is done then it will be confirmed and validated. Run all the hand coded test cases to get the particular outcome.

5. AUTOMATION TOOLS

There are different apparatuses that assistance programming groups fabricate and execute mechanized tests. Numerous product engineers are currently utilizing unit tests as a major aspect of their improvement endeavours to check basic parts of their undertakings, for example, libraries, models and methods. Choosing a computerized programming testing device is an imperative procedure. For the most part a decent testing instrument should:

- It should test all capacities in application programming.
- It ought to have great investigating offices.
- It ought to have clear help record and a client manual.

6. VISUAL STUDIO TEST PROFESSIONAL

This is the most comprehensive testing solution for all Microsoft platforms, including desktops, phones, tablets, servers, and also the cloud. With MSDN subscription and also access all the other Microsoft products and services, which can further help the design, develop, and to test applications on multiple platforms.

7. CODED UI

Mechanized tests that drive our application through its client interface (UI) are known as coded UI tests. These tests incorporate practical testing of the UI controls. They confirm that the entire application, including its UI, is working accurately. Coded UI was made accessible as a major aspect of the visual studio 2010 refresh. Making coded UI test is simple. Coded UI has a component called Test Builder, which keeps running out of sight. The Coded UI Test Builder records client activities and creates codes. After the test is made, it is conceivable to alter the codes. Scarcely any reasons why Coded UI instrument is a favoured decision for programming analysers are programming analysers and engineers can work utilizing a similar device or dialect which

empowers them to team up viably, the component ID system is a ground-breaking highlight Coded UI, Coded UI permits building up a broad test suite and performing tests in nearby conditions and utilizing Coded UI with layered structure mechanization groups can create refined tests.
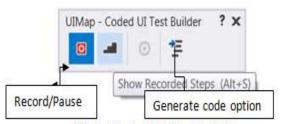

Fig2: Coded UI Test builder

Making a Coded UI test produces a UIMap question that is particular to the test and speaks to the windows, controls, parameters and affirmations that are in the UI or made amid test recording. A Coded UI test class is distinguished by a class. Each coded UI test is a test technique in a coded UI test class. It is conceivable to add different test techniques to each coded UI test class and recognize each coded UI test strategy by utilizing Test Method Attribute. The test technique can utilize an Assert explanation to look at the genuine estimation of the property to a normal esteem. The aftereffect of this examination decides the result of the test. Each time client run a Coded UI test, it is conceivable to examine the test outcomes and if the test falls flat, client can see or store the points of interest of statement.
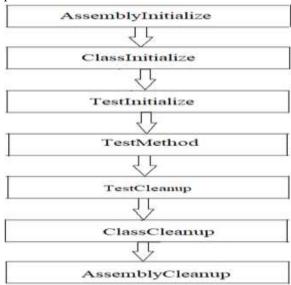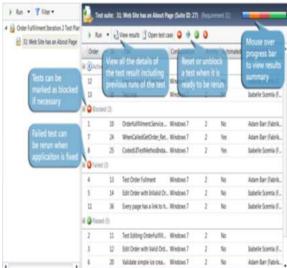

Fig3:Attributes used in test execution in Coded UI Framework

Working:
Step1:

Screenshot1:Tracking test Suite progress

In Microsoft Test Manager, To keep tabs on development for the test suites in the present test design promptly after to run test cases. The tests that have passed and fizzled.this can stamp tests as blocked or reset tests to dynamic when they are prepared to run them once more. This can track the outcome after execution as appeared in screenshot Tracking test suite advance.

Step2:

Screenshot 2: Microsoft Test manager Result sheet likewise screen the advance of this test design by utilizing the test design comes about component in Microsoft Test Manager as appeared in the fig 1 Microsoft Test director Result sheet . The test design comes about incorporate outlines and numerical insights on the tests in the test design. The insights incorporate the tests that are as of now passed, fizzled, blocked, uncertain, cautioning, and dynamic. Furthermore, the test design comes about incorporate

diagrams that demonstrate the disappointment writes and determination information.

The test design results can be separated to indicate the test suites and test setups that need to be incorporated. For instance, it may just need to see the test comes about for particular test suites in the test arrange for that group is presently dynamic in. Furthermore, it can channel the test designs to just view the test outcomes set to Windows 7. Of course, the majority of the test suites and test designs that are in the test design are incorporated into the test design comes about.

8.CONCLUSION

Most regular motivation to utilize mechanization in the testing procedure is to execute an arrangement of tests substantially quicker and abstaining from rehashing manual testing. Likewise, it makes it simpler to send visit manufactures and aides in increment the trust in the created application. To accomplish this one needs to pick a test computerization structure to address the organization's issues. A large portion of the organizations requires a test mechanization system that must be adaptable and reusable over different applications that expects almost no programming abilities from the interior test group. To address this issues Page Object configuration is the approach that demonstrated to have minor weaknesses when it came to make new experiment and keep up the current ones because of the next to zero programming abilities in the test group.

Utilizing Coded UI highlight in Visual Studio apparatus and Page Object approach, structure can be created effectively as indicated by the required application that should be robotized. Since Visual Studio test proficient mechanization apparatus is most appropriate for windows application, one can create test robotization structure for all windows application utilizing this instrument and Page Object approach. These apparatuses and method has been tried and effectively actualized to a portion of the windows applications.

REFERENCES

[1] P.Nagarani,R.VenkataRamanaChary "A tool based approach for Automation of gui applications" IEEE-20180.

[2] Niranjana S, Balamurugan"automation testing using coded ui test" A International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-2, Issue-4,April 2016.

[3] Antonia Bertolino" Software Testing Research: Achievements, Challenges, Dreams" IEEE computer society.

[4] Shruti.N. Pardeshi "Study of Testing Strategies and availableTools" International Journal of Scientific and Research Publications, Volume 3, Issue 3, March 2013.

[5] Pradeep P PQuest Global "a new generation software test automation framework – CIVIM.

[6] Hitesh Tahbildar and BichitraKalita "Automated software test data Generation: direction of research" International Journal of Computer Science & Engineering Survey (IJCSES) Vol.2, No.1, Feb 2011.

[7] Muhammad AbidJamil, Muhammad Arif "Software Testing Techniques: A Literature Review ",6th International Conference on Information and Communication Technology.