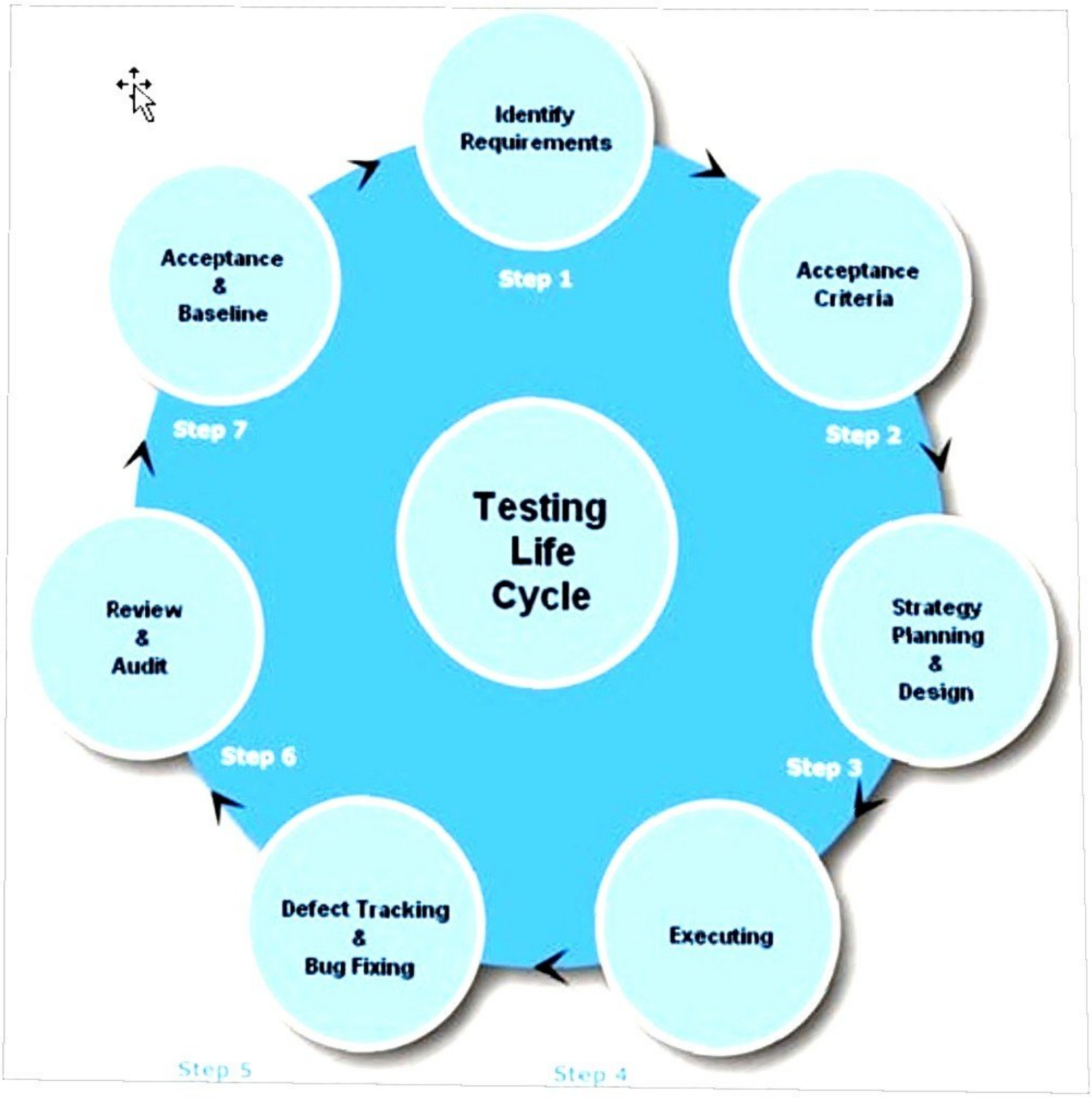
SOFTWARE
TESTING

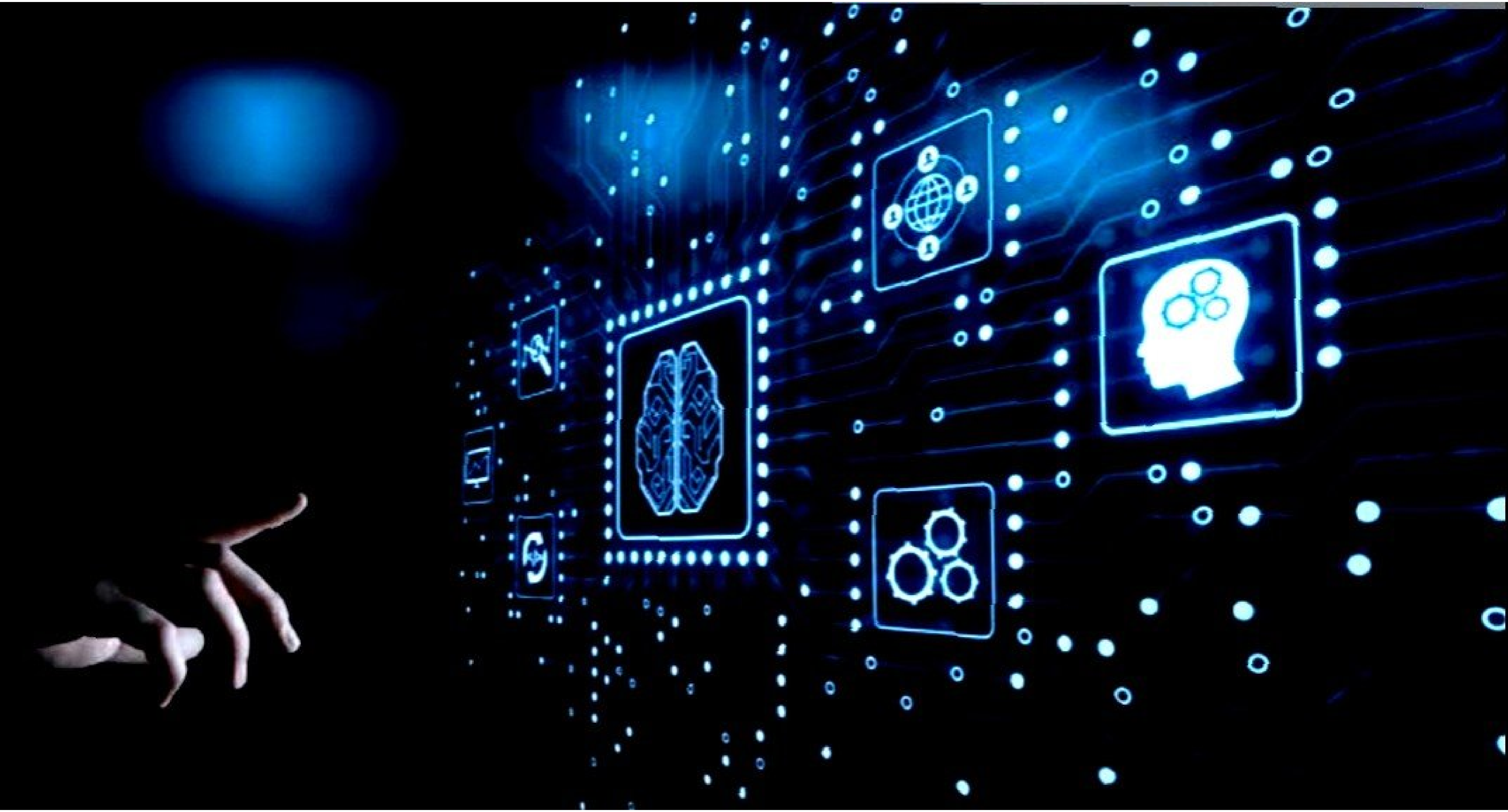
Test cases should be prepared after a clear understanding of the functional and technical aspect of the software. The tester must possess a keen observational capacity and complete knowledge about the software.



Moreover, cost plays an effective role here. Customers prefer to accept software with maximum quality at a minimal cost. When we go for manual testing, the process is more tedious and time-consuming as it is all done manually by a tester.

7 SOFTWARE TESTING PRINCIPLES

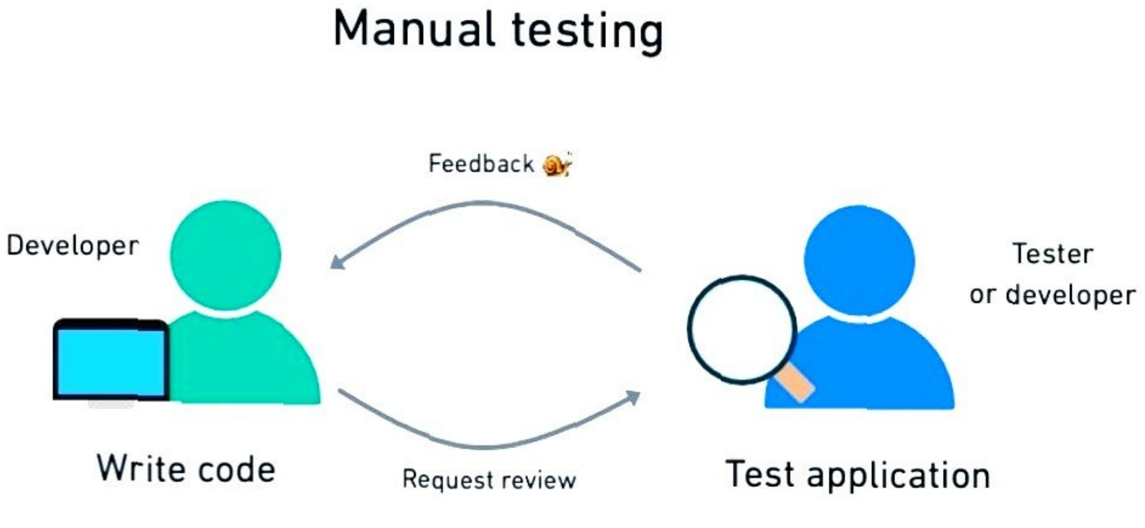
1. Testing shows presence of defects
2. Exhaustive Testing is not possible
3. Early Testing
4. Defect Clustering
5. Pesticide Paradox
6. Testing is context-dependent
7. Absence of errors fallacy

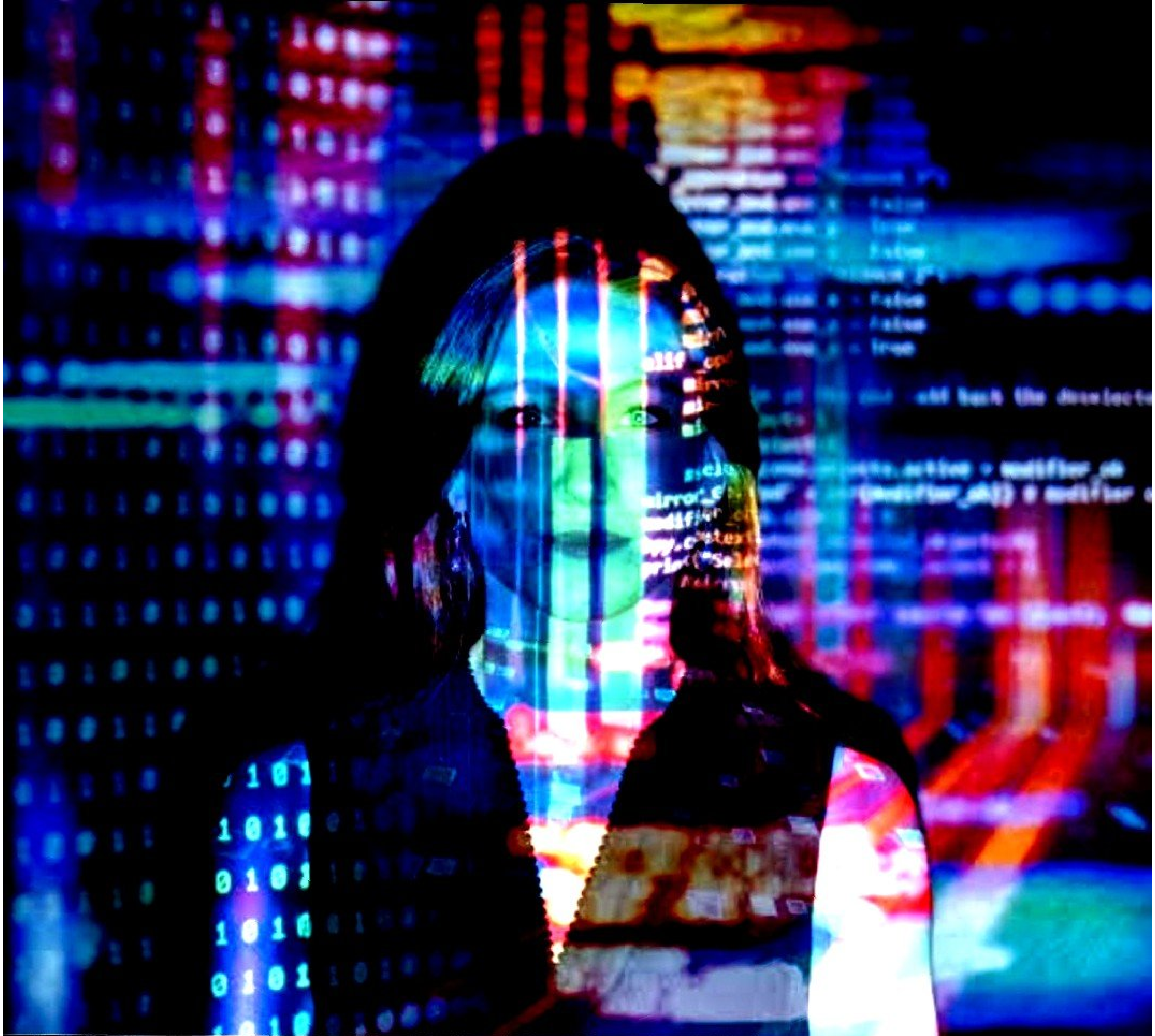


The word ‘test’ originally referred to “a small vessel used in assaying precious metals”. This meant that testing was a method of ascertaining the quality of gold or silver. It was also used in the process of refining valuable alloys, such as tin.

Later, the term was adopted in other fields, and these days it’s common to find it in contexts such as education, medicine, or software development. Its essence, however, has not changed: testing is used to refine end value.

We use tests in software development to ensure that code works as expected. Tests can be manual or automated. Manual testing is similar to automobile manufacturers crashing cars to verify that they’re safe for the road. It works, but it’s far too expensive to do frequently, so it is typically done at the end of the production cycle. The trouble with this method is that problems found at this stage can delay a product’s launch by months.

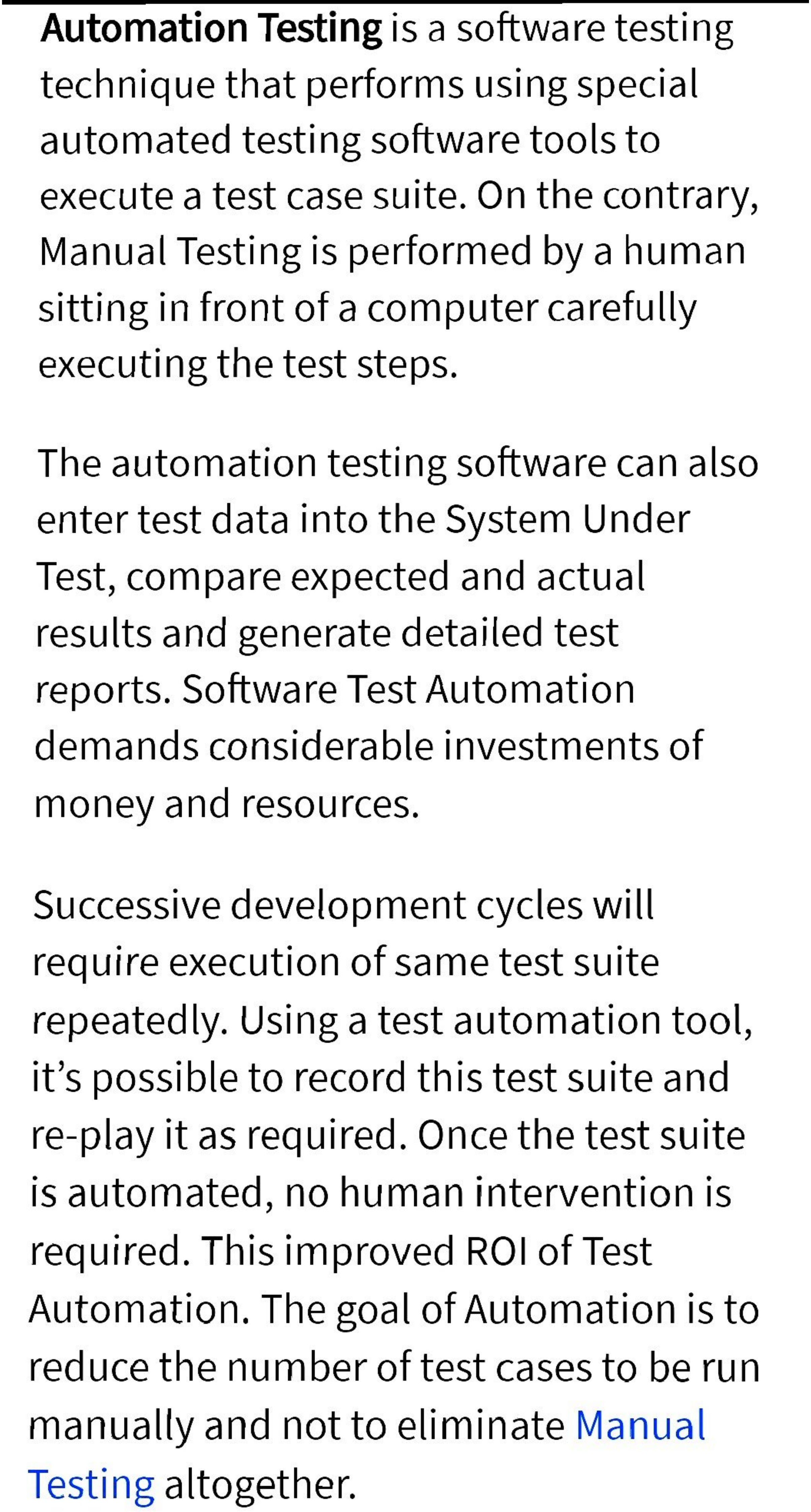




= False
= True
= False
= False
= False
= True
add back the deselecte
elif
.active = modifier_ob
modifier_ob

Advantages of Automation Testing

- **Simplifies Test Case Execution:** Automation testing can be left virtually unattended and thus it gives an opportunity to monitor the results at the end of the process. . Thus, simplifying the overall test execution and increasing the efficiency of the application.
- **Improves Reliability of Tests:** Automation testing ensures that there is equal focus on all the areas of the testing, thus ensuring the best quality end product.
- **Increases amount of test coverage:** Using automation testing, more test cases can be created and executed for the application under test. Thus, resulting in higher test coverage and the detection of more bugs. This allows for the testing of more complex applications and more features can be tested.
- **Minimizing Human Interaction:** In automation testing, everything is automated from test case creation to execution thus there are no changes for human error due to neglect. This reduces the necessity for fixing glitches in the post-release phase.
- **Saves Time and Money:** The initial investment for automation testing is on the higher side but it is cost-efficient and time-efficient in long run.



Automation Testing is a software testing technique that performs using special automated testing software tools to execute a test case suite. On the contrary, Manual Testing is performed by a human sitting in front of a computer carefully executing the test steps.

The automation testing software can also enter test data into the System Under Test, compare expected and actual results and generate detailed test reports. Software Test Automation demands considerable investments of money and resources.

Successive development cycles will require execution of same test suite repeatedly. Using a test automation tool, it’s possible to record this test suite and re-play it as required. Once the test suite is automated, no human intervention is required. This improved ROI of Test Automation. The goal of Automation is to reduce the number of test cases to be run manually and not to eliminate Manual Testing altogether.

Automation Testing Process

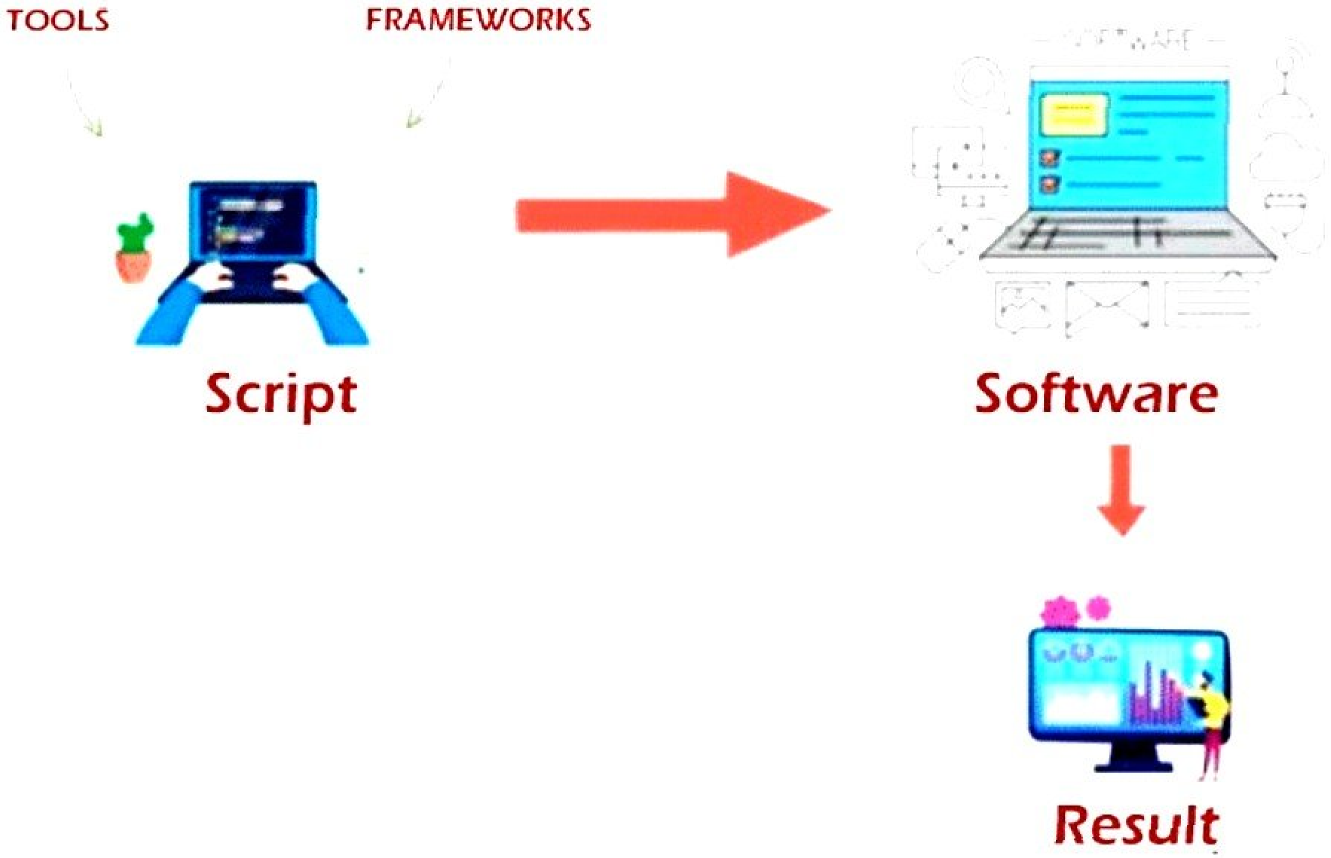
1. **Test Tool Selection:** There will be some criteria for the Selection of the tool. The majority of the criteria include: Do we have skilled resources to allocate for automation tasks, Budget constraints, and Do the tool satisfies our needs?
2. **Define Scope of Automation:** This includes a few basic points such as the Framework should support Automation Scripts, Less Maintenance must be there, High Return on Investment, Not many complex Test Cases
3. **Planning, Design, and Development:** For this, we need to Install particular frameworks or libraries, and start designing and developing the test cases such as NUnit, JUnit, QUnit, or required Software Automation Tools
4. **Test Execution:** Final Execution of test cases will take place in this phase and it depends on Language to Language for .NET, we'll be using NUnit, for Java, we'll be using JUnit, for JavaScript, we'll be using QUnit or Jasmine, etc.
5. **Maintenance:** Creation of Reports generated after Tests and that should be documented so as to refer to that in the future for the next iterations.

Criteria to Select Automation Tool

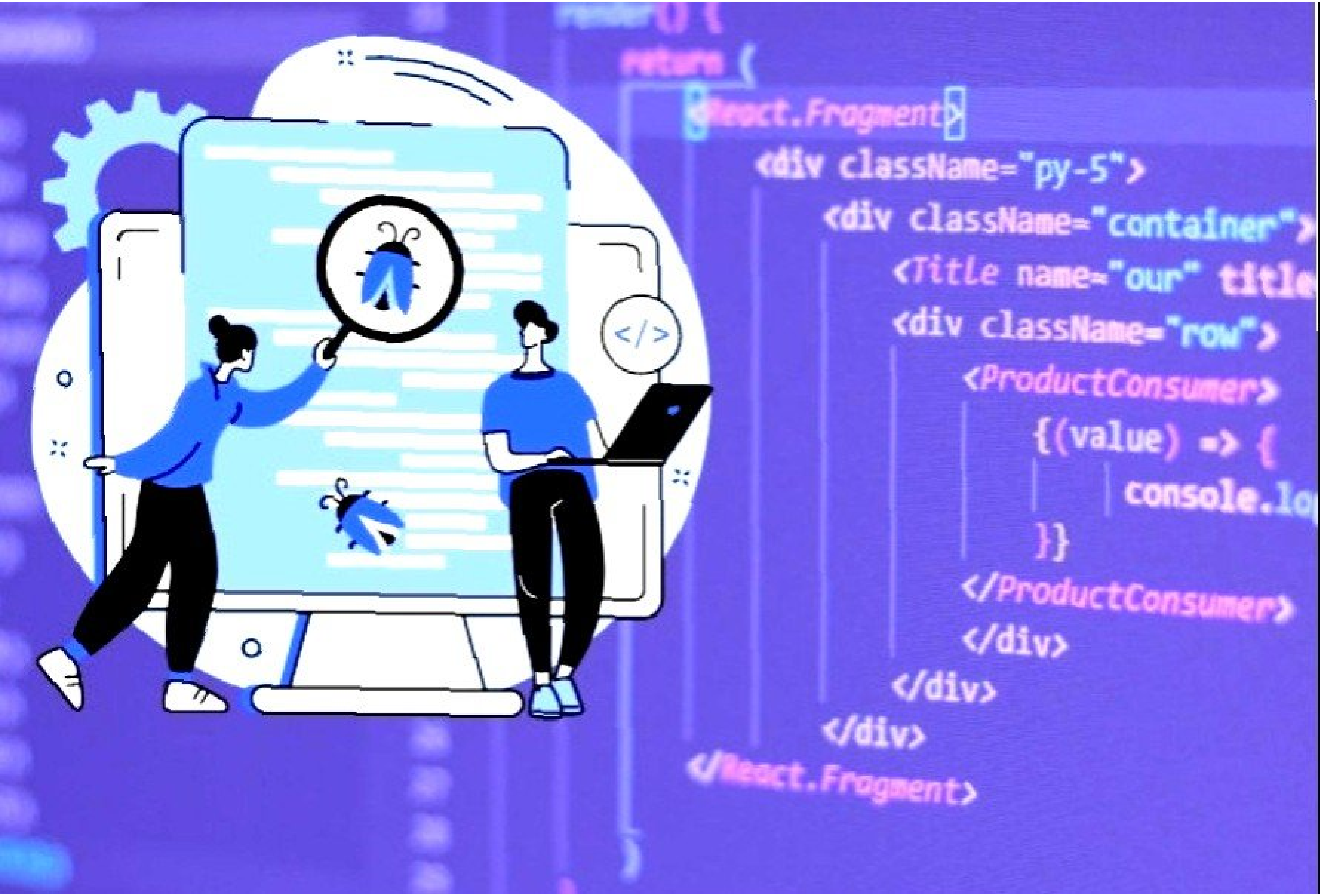
Automation Testing

- Fast
- Reliable
- Reusable
- Improves Accuracy
- Saves time and money
- Reduces Human-generated error
- Supports the execution of repeated test cases

WHAT IS AUTOMATION TESTING ?

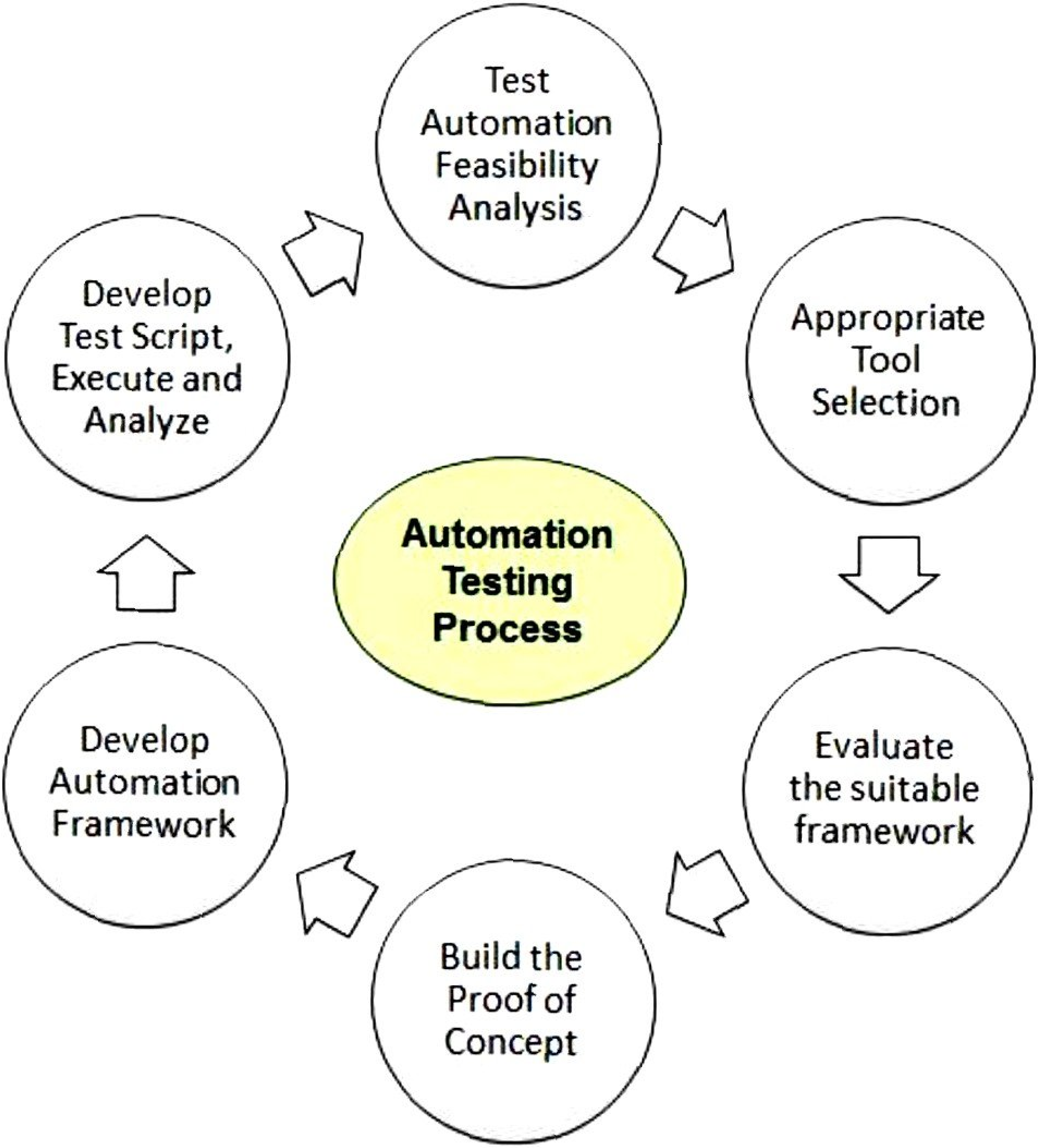


Manually when testers write the test cases and execute them repeatedly, known as manual testing, which is time-consuming and the test results are not sure. So to recover these drawbacks, automation testing came into existence. With automation testing, developers or testers keep everything in their hands, boosting the product's productivity.



```
render() {
  return (
    <React.Fragment>
      <div className="py-5">
        <div className="container">
          <Title name="our" title
          <div className="row">
            <ProductConsumer>
              {(value) => {
                console.lo
              }}
            </ProductConsumer>
          </div>
        </div>
      </div>
    </React.Fragment>
```

For any automated tool implementation, the following are the phases/stages of it. Each one of the stages corresponds to a particular activity and each phase has a definite outcome.

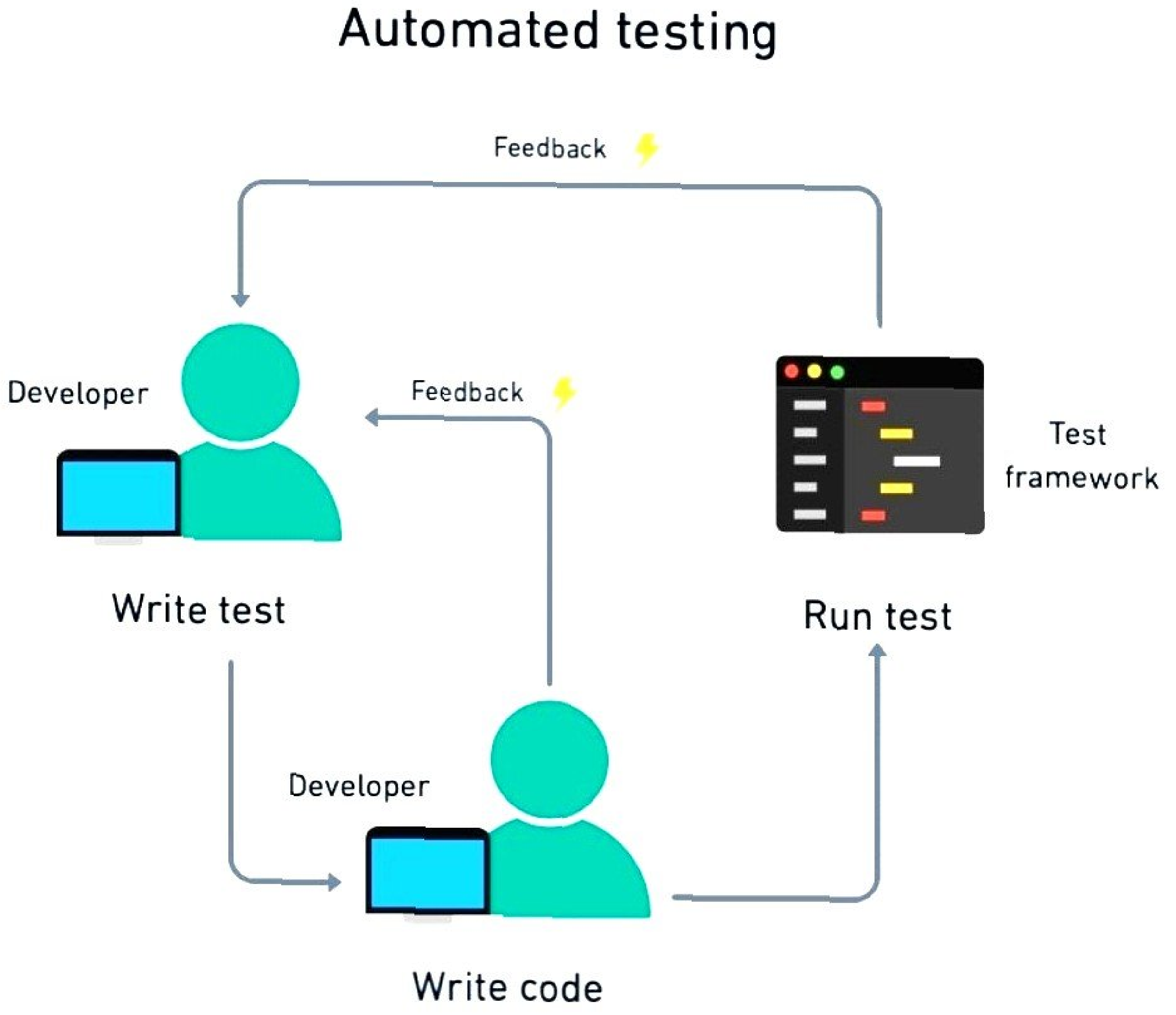

Test Automation Feasibility Analysis
Appropriate Tool Selection
Evaluate the suitable framework
Build the Proof of Concept
Develop Automation Framework
Develop Test Script, Execute and Analyze
Automation Testing Process

Makes Up for an Automation Testing Directory, One of Its Kind!

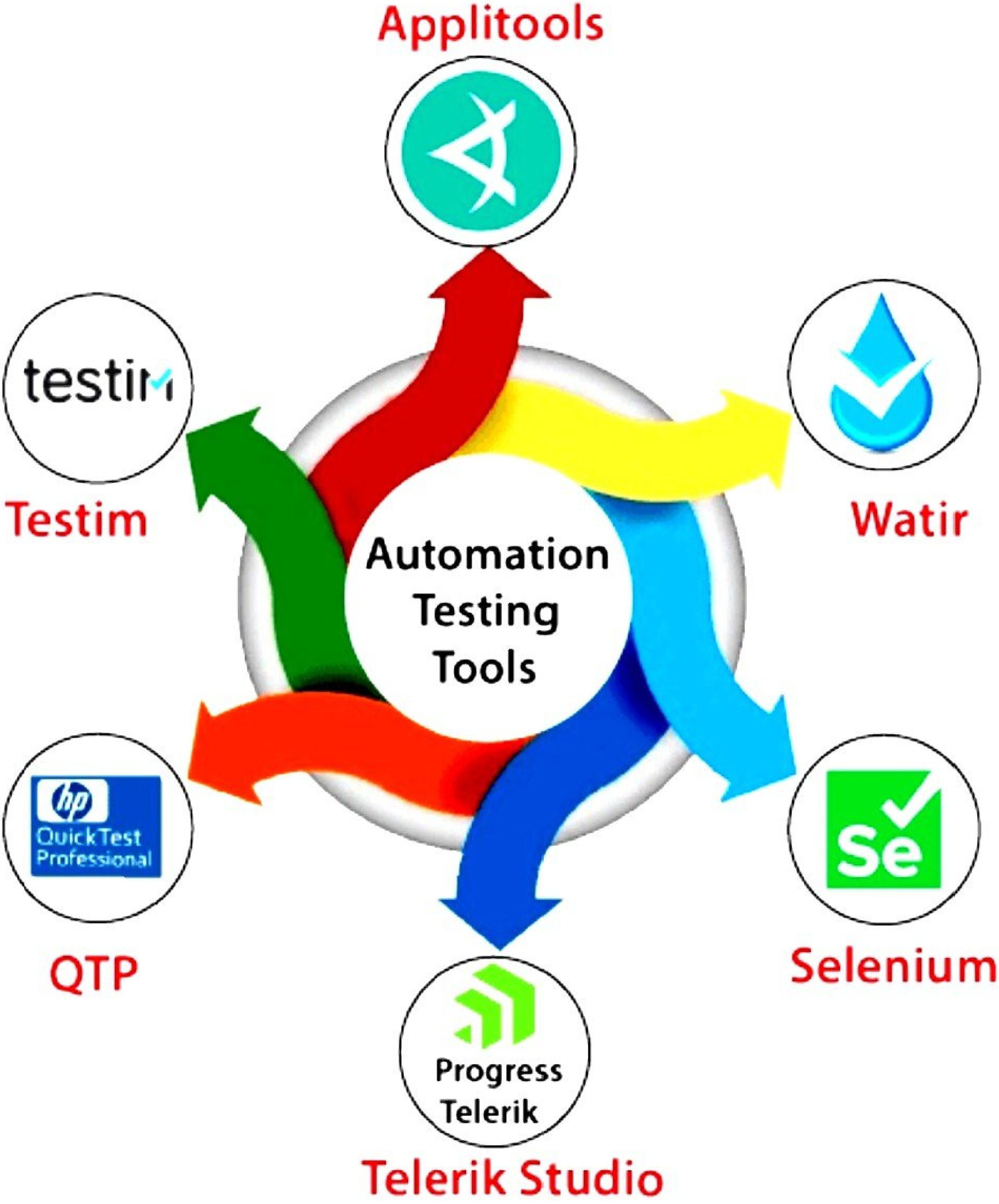
Constant transformations in automation are advancing at a lightning speed in our day-to-day lives. From voice tech to self-driving cars, technology is accelerating automation to make lives on earth simpler.

Today, robots are developed that can do better tasks than humans. But, the quality of automation applied for any activity should be tested before it is put to any use. It should be a parallel priority with the pace at which automation is progressing.

Automated software testing has an entirely different cost structure. There's an initial inversion plus periodical maintenance, but once tes automation is in place, we can run our tests as often as we need—for pennies.



With test automation, developers get continuous feedback, allowing them to spot problems very early in the production cycle. Quick iteration results in improved design, better quality, and safer launches.


Applitools
testim
Testim
Watir
Automation Testing Tools
QTP
hp QuickTest Professional
Se
Selenium
Progress Telerik
Telerik Studio