ASSIGNMENT-4

Q.1 Describe various testing tools & their key features.

The exponential and unparalleled change in technology affects the way in which the organizations develop, validate, deliver, and operate the software. So, it's necessary that organizations must consistently find new ways to deliver high-quality software. Software Testing provides development teams with ways and tools to determine the quality of their software.
Now, there are tons of software testing tools available these days. It's really hard to keep track of 100+ software testing tools out there. Some of the well-known software testing tools in various categories, are as mentioned below:

- Automation Testing Tools
- Agile Testing Tools
- Mobile Testing Tools
- Load Testing Tools
- Cross – Browser Testing Tools
- Test Management Tools

Automation Testing Tools

Automation Testing Testing these days is a must for most software projects to ensure automatic verification of key functionalities. Also help teams efficiently run a large number of tests in a short period of time. Listed below are a few tools that help software teams build and execute automated tests:

1. Selenium

Selenium is a popular testing framework to perform web application testing across various browsers and platforms like Windows, Mac, and Linux. With selenium, you can come up with very powerful, browser-centered automation

testing scripts which are scalable across different environments. It is compatible with several programming languages & automation testing frameworks.

2. Watir

Watir, pronounced as water, is an open source testing tool made up of Ruby libraries to automate web application testing. Loaded with Ruby libraries, it also supports applications scripted in other languages. You can link it with databases, export XML files, read files, spreadsheets, and synchronize code as reusable libraries. It is a very light-weight open source tool.

3. Ranorex

Ranorex is flexible, all in one, GUI testing tool using which you can execute automated tests flawlessly throughout all environments and devices. When compared to other GUI testing tools, Ranorex offers super smart object recognition feature that automatically detects any change in the user interface and keeps the test going. Other features include reusable code modules, early bug finding, and integration with other tools.

4. HPE Unified Functional Testing (UFT)

HPE Unified Functional Testing (UFT) software, formerly known as HP QuickTest Professional(QTP) is an automated functional GUI testing tool which allows the automation of user actions on a client based computer application. It offers features like object recognition, error handling mechanism, and automated documentation. It also uses a scripting language to manipulate the objects and controls of the application under test.

5. Tricentis Tosca

Tricentis Tosca is a very popular software testing tool that is used to automate end-to-end testing for software applications. This tool offers a single repository for all functional test artifacts, including requirements, user stories, test data,

virtualization assets. Tosca comes with capabilities like test data provisioning, service virtualization network, tests mobile apps, and risk coverage.

Agile Testing Tools

Companies are adopting agile software development methodologies such as Scrum, Extreme Programming (XP) for their projects. Agile testing comes with many challenges, requires experimenting and trying new ideas. Listed below are some agile testing tools that benefit testers and developers who work on agile projects:

1. JIRA

JIRA is a popular agile testing as well as a project management tool developed by Atlassian, a software company that develops products for project managers, software developers, etc. This tool can be used for tracking defects, planning, creating reports, and managing all agile software development projects. It supports an agile methodology like Scrum, Kanban.

2. SoapUI

It is an agile testing tool and is the most advanced REST and Service Oriented Architecture developed by SmartBear. SoapUI is basically used for functional testing of web services which includes, web service development, invoking of web services, etc. It is a free and an open source tool which allows you to create and execute functional, regression and load tests. Allows you to create test cases using drag and drop interface.

3. Selenium WebDriver

Selenium WebDriver is another popular agile automation tool which is widely used across the software industry. It is used only to automate browser-based application and it does not support desktop based applications. Users can write

automation scripts in multiple programming languages like JAVA, C#, Python, Ruby, PHP.

Mobile Testing Tools

Mobile applications have become more and more important for businesses. So, testing teams need to adapt and get ready to verify and evaluate mobile apps as part of their projects. There are various tools and online resources to help testers build tests for your mobile devices, record and run automated UI and unit tests for mobile apps and code libraries. Listed below are few such tools:

1. eggPlant

eggPlant is a licensed tool built by TestPlant and is primarily aimed for application testing and GUI testing in mobile devices. It is the combination of eggPlant Functional and eggOn, the mobile agent that enables to fully test mobile apps on iOS, Android, Windows Phone, BlackBerry, and any other mobile devices. Rather than the object-based approach like in most of the test automation tools, eggPlant works on image-based approach.

2. SeeTest

SeeTest is a mobile application test tool developed by Experitest Ltd, for iOS, Android, Blackberry and Windows Phone. It offers visual testing, functionality testing, and is used for both emulators as well as real devices. It leverages self-learning algorithms and a modular self-enhancing image recognition technology. Some other mobile-related tools available include SeeTest Cloud, SeeTest Network Virtualization, Mobile Add-on for UFT.

3. Silk Test

Silk Test is a licensed product of Microfocus which offers functionality and regression testing. You can use this tool for regression, cross-platform, and

localization testing of all mobile application types, like mobile web, native, and hybrid applications. It has cross-browser support and enables efficient, speedy and high-quality automation testing. Silk Test also makes it easy to integrate functional testing in the CI/CD pipeline.

Load Testing Tools

The use for websites, web applications, and APIs has become more and more critical in recent times, it's important to design and build them efficiently so that they can handle a huge number of requests. So, to actually test and verify the performance of services under load, developers can perform load and stress tests. Listed below are some tools that load testers use:

1. Apache JMeter

It is an open source Java desktop application designed for load testing. Its architecture is actually centered around plugins with the help of which JMeter provides a lot of out of box features. It supports different types of applications, servers and protocols like Web, SOAP, FTP, TCP, LDAP, SOAP, MOM, Mail Protocols, shell scripts, Java objects. Other features of JMeter include powerful Test IDE, dynamic reporting, command line mode, portability, and multithreading.

2. Tsung

Tsung is a popular open-source multi-protocol distributed load testing tool. The main purpose of this tool is to help users test the scalability and performance of IP based client/server applications. It can also be used to perform load and stress testing on servers. It can be used to check the load on HTTP, WebDAV, SOAP, PostgreSQL, MySQL, LDAP, MQTT, and Jabber/XMPP servers.

3. WAPT

WAPT is a load and stress testing tool that allows you to build, generate and monitor load tests via a graphical user interface. It provides an easy and cost-effective way to specifically test business applications websites, mobile websites, web portals, etc. It is based on AJAX and RIA technology. WAPT works on secure HTTPS websites, dynamic content and RIA applications under a data-driven mode.

Cross – Browser Testing Tools

The process of testing a website or application in multiple browsers and making sure that it works consistently and as in intended without any dependencies, or compromise in quality is called Cross Browser Testing. There are a lot of cross-browser testing tools available in the software testing market.

1. LambdaTest

It is a popular and very easy to use cross browser testing tool that allows you to run live testing on over 2000 modern day mobile and desktop browsers and devices in a matter of seconds. LambdaTest tool has really clean UI and offers a lot of impressive features like automated screenshot testing, local testing functionality and integration with most popular bug tracking and project management tools like Jira, Asana, Trello, etc.

2. BrowserStack

BrowserStack is a cloud-based cross browser testing tool that doesn't require any software to be downloaded or installed, which allows for testing for over a wide range of browsers and real devices of various sizes and resolutions. It is one of the oldest players in this space with a huge market share. One of the key features is that in addition to simulators and emulators, BrowserStack offers interaction with real browsers on remote machines

3. Sause Labs

Being one of the leaders in the cross-browser market, Sauce Labs lets you test your website's compatibility with various browsers, devices, and operating systems. It offers both automated and manual testing of web and mobile applications. What separates it from the other cross-browser testing tools is the very clean UI it offers for automated mobile testing.

Test Management Tools

Software development teams can benefit by test case management tools. They can use web-based management tools to manage their projects, testing resources, record test results and generate reports to help optimize all testing activities. There are various test management tools available for different needs and most popular ones are listed below:

1. ZephyrTestRail

Zephyr is a test management software that provides companies with the ability to execute manual and automated tests. It offers integration with other popular testing platforms like JIRA, Confluence, Jenkins, Bamboo. With Zephyr, you can get the flexibility, visibility, and insights you need to release better software faster. Mostly there are two versions of this product.

2. QMetry

QMetry is the most comprehensive test management tool designed for agile testing and DevOps teams to build, manage, and deploy quality software faster with confidence. With this tool test management, automation, and predictive analytics are covered within a single platform. It has integrations with platforms such as JIRA, Selenium, Appium, Bamboo, Jenkins, HP ALM, Rally, etc.

3. TestRail

   TestRail is a modern day test management tool that is available both for your own server as well as cloud edition. It is the only tool that integrates with full JIRA add-on integration, including JIRA Cloud. TestRail is a helpful tool to manage, track and coordinate testing efforts. This tool helps you manage, organize and track all testing efforts at the same place. It tests report & metrics automation.

Q.2 Compare Open source Testing tools with Commercial Testing tools

Open Source Testing Tools

Some of the most popular open source load testing tools include JMeter, The Grinder, Gatling, Webload, Tsung, and more.

Pros:

- No initial discussions about buying software
- No need to worry about licenses. It’s free to use
- The code is open and so we can extend based on our needs
- More reachable because it's free
- Managed by open source community
- Attracting more customers by giving competitive pricing

Cons:

- Security
- Less resource availability in the market
- Time is taken to develop
- An employee with that tool knowledge leaving the company
- Not all the features available
- More time required to get code issues fixed
- Little to no support or training

- Free tools but your time is expensive and may also affect project deadlines
- Limited UI support for script development and feedback while running tests
- Overly focused on web based protocol
- Memory and CPU issues when running large loads
- Large scale tests may be difficult

Commercial Testing Tools

There are lots of commercial load testing tools on the market including HP Loadrunner, Silkperformer, Neoload, and SOASTA CloudTest. Some of our favorite tools are cloud based and offer pay-as-you-use pricing models.There are lots of commercial load testing tools on the market including HP Loadrunner, Silkperformer, Neoload, and SOASTA CloudTest. Some of our favorite tools are cloud based and offer pay-as-you-use pricing models.

Pros:

- Business focused
- Full Featured and target based that will have all features as a package
- Mostly all in one solution (Includes Test Development, Management, Tracking and so on.)
- On Time / Quick Support
- Secured and reliable
- Avail training services or step-by-step documentation
- Quick fixes
- Constantly updated as technologies evolve helping you concentrate on your application and not your tool
- Relatively easy to use scripting interface
- Readily available support teams, which can help you, fix unforeseen problems
- Supports a wide range of protocols with support for proprietary software
- Supports strong real-time monitoring helping identify server side bottlenecks
- Excellent integrated analysis tools that can dynamically generate graphs
- A large number of customized graphs can be generated
- Advanced reporting with the ability to generate different formats of report
- Easy availability of experienced tool users

Cons:

- Expensive
- Some companies even charge separately even for support
- Limited platforms
- Cannot customize
- Increased dependency
- Less reachable customers by giving competitive pricing

Pros and Cons of Commercial and Open source software:

Cost — As stated earlier the commercial software costs more and is ideally a big hurdle in selecting and investing funds over the open source software because, open source software is available freely; you hardly have to pay anything for it.

Security — It is again a big question. Commercial software is mostly developed with certain intension and under consideration of specific standards hence security constraints are taken care of in a better way. This may not be the case with Open source software — you may have to compromise over security (up to some extent)!

Reliability and Ownership — A software developed by a commercial software/ solution provider will always be highly reliable as it is produced with certain considerations and parameters. The vendor developing the software is solely responsible for all the failures (if at all) and also responsible for its long term (may be life time) support and maintenance till software is literally phased out. In Open source software reliability may not be confirmed, as everyone's requirements can't be fulfilled neither there is anyone responsible to own the failures.

Customization — Until few years customization in commercial software was not possible because, the software were distributed in packaged form (to be used as it is). But, now customization options are available for few commercial software. Ideally, open source software provides wide scope of customization, which is another great advantage that, one can mould the software as required.

Availability — This may be the case in both, what you need may not be available or cannot be made available either in commercial software or open

source software. Thus, meeting exact requirements may be difficult. But, both the options up to certain level provide availability of making the software as required, especially the open source.

Q.3 What is Selenium? Introduction to Selenium Automation Testing

Selenium is a free (open source) automated testing suite for web applications across different browsers and platforms. It is quite similar to HP Quick Test Pro (QTP now UFT) only that Selenium focuses on automating web-based applications. Testing done using Selenium tool is usually referred as Selenium Testing.

- There are many experts available because it is in the market for more than a decade and many companies are using it.
- It is highly secure.
- It is maintained by a reliable team (Google and some other companies are also involved).
- Strong support is available through many ways (Blogs, Selenium GitHub Issue section, Online Forums, StackOverflow site, Expert Freelancers).
- It is not having all features like IDE to develop tests, Framework, Reporting…etc.
- There are Open Source IDE's, Frameworks and Reporting Utilities available in the market which can be integrated to Selenium.
- It takes more time to develop tests when compared with some commercial tools like UFT.
- But it gives parallel and distributed test execution features which are not there in many commercial tools.

Key Features of Selenium

Some of the key features that make Selenium most popular are:

- Selenium WebDriver has become a default for web testing and is now a W3C standard.
- Other commercial tools which are in competition are integrated and integrating with Selenium.
- Almost every commercial tool vendor providing cloud test execution feature for Selenium tests.

- Vendors like Micro Focus, Ranorex, BrowserStack, Sauce Labs, Experitest are sponsors for Selenium.
- The marketing strategy is changed and commercial vendors are now providing services to opensource tools

Selenium is not just a single tool but a suite of software's, each catering to different testing needs of an organization. **It has four components.**

- Selenium Integrated Development Environment (IDE)
- Selenium Remote Control (RC)
- WebDriver
- Selenium Grid

Since Selenium is a collection of different tools, it had different developers as well. Below are the key persons who made notable contributions to the Selenium Project

Primarily, Selenium was **created by Jason Huggins in 2004**. An engineer at ThoughtWorks, he was working on a web application that required frequent testing. Having realized that the repetitious Manual Testing of their application was becoming more and more inefficient, he created a JavaScript program that would automatically control the browser's actions. He named this program as the "**JavaScriptTestRunner**."

Seeing potential in this idea to help automate other web applications, he made JavaScriptRunner open-source which was later re-named as **Selenium Core**.

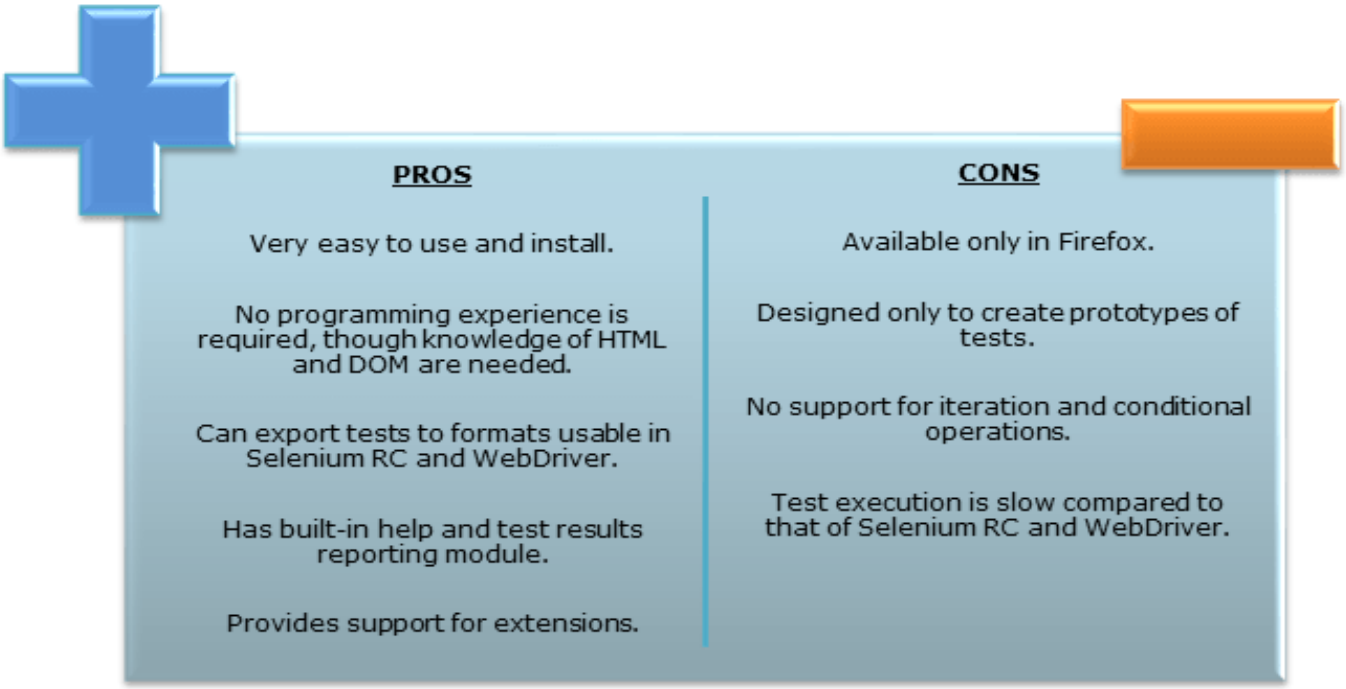
The Same Origin Policy Issue

Same Origin policy prohibits JavaScript code from accessing elements from a domain that is different from where it was launched. Example, the HTML code in www.google.com uses a JavaScript program "randomScript.js". The same origin policy

will only allow randomScript.js to access pages within google.com such as google.com/mail, google.com/login, or google.com/signup.

Selenium IDE

Selenium Integrated Development Environment (IDE) is the **simplest framework** in the Selenium suite and is **the easiest one to learn**. It is a **Firefox plugin** that you can install as easily as you can with other plugins. However, because of its simplicity, Selenium IDE should only be used as a **prototyping tool**. If you want to create more advanced test cases, you will need to use either Selenium RC or WebDriver.

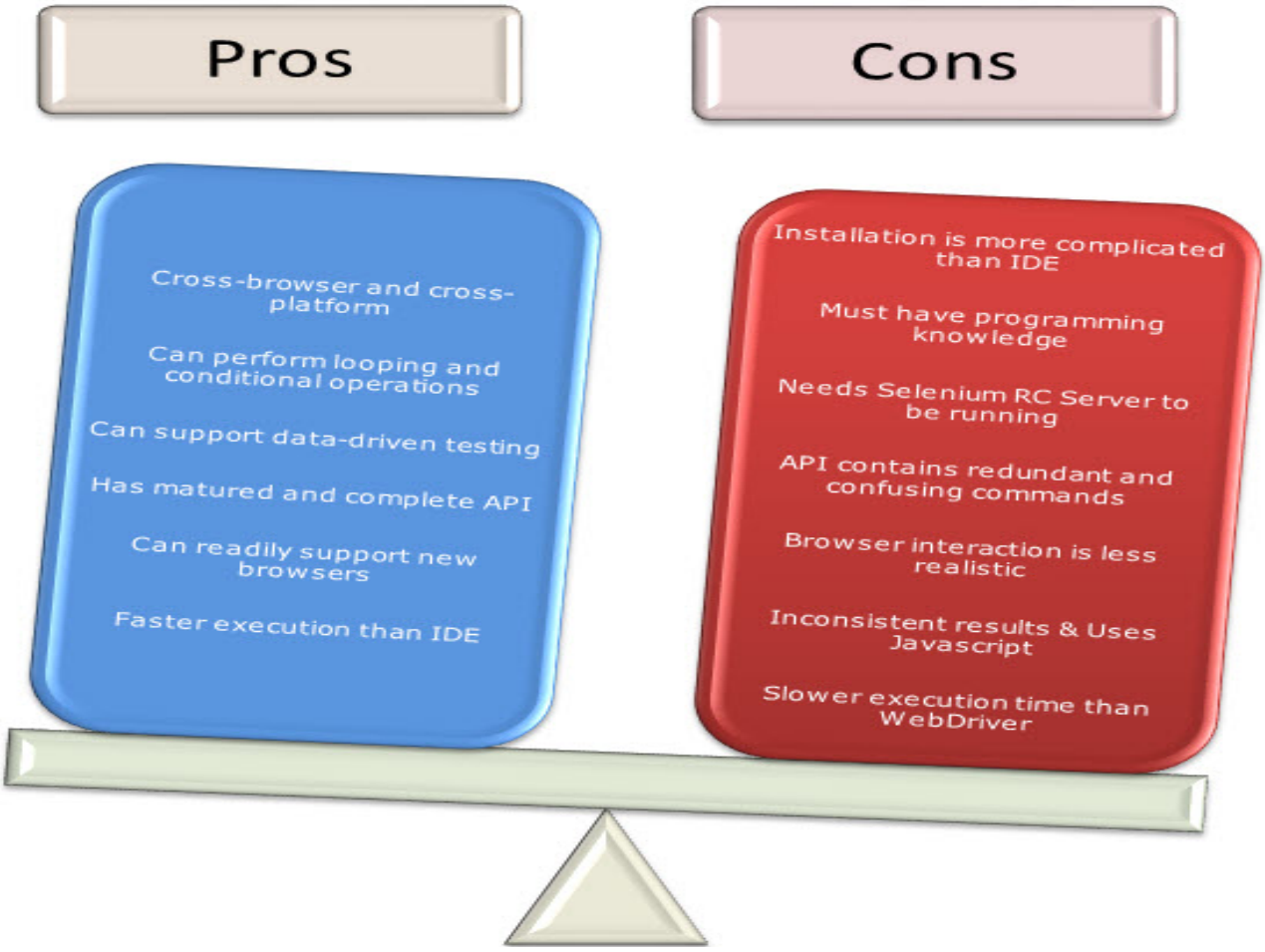


Selenium Remote Control (Selenium RC)

Selenium RC was the **flagship testing framework** of the whole Selenium project for a long time. This is the first automated web testing tool that **allowed**

users to use a programming language they prefer. As of version 2.25.0, RC can support the following programming languages:

- Java
- C#
- PHP
- Python
- Perl
- Ruby



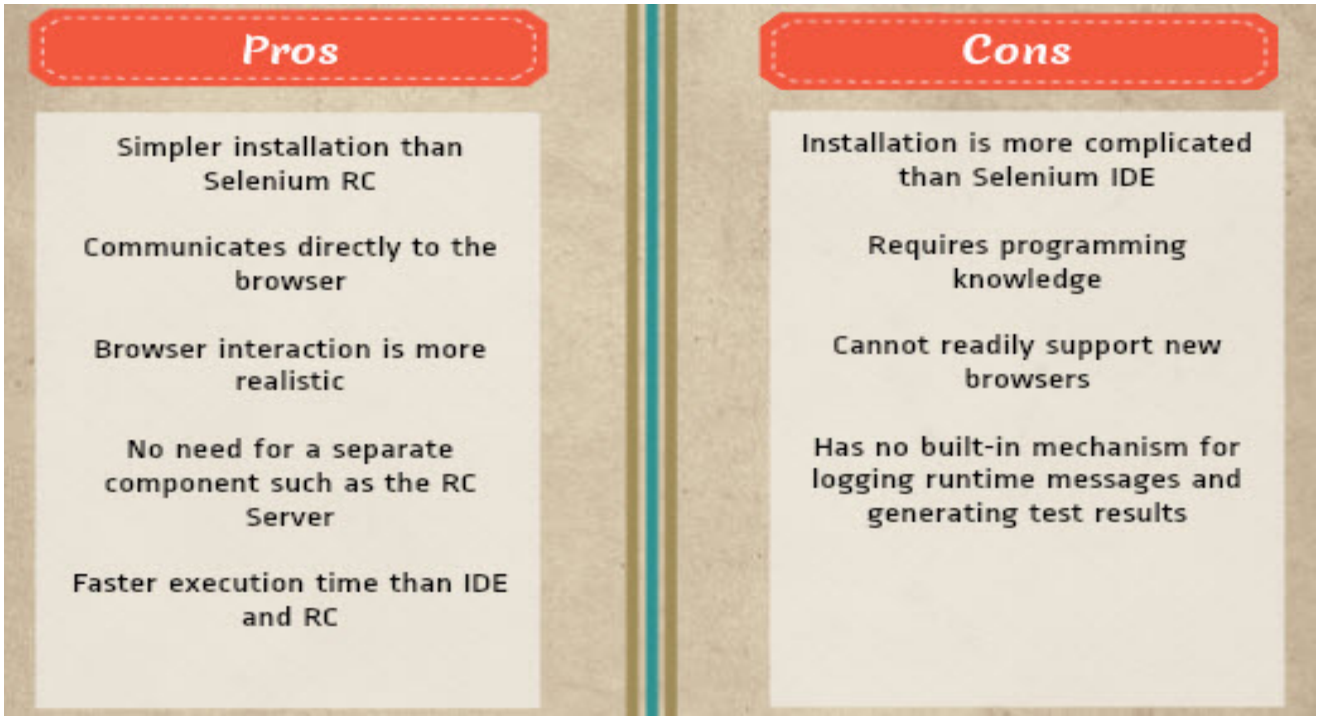
Selenium WebDriver

The WebDriver proves itself to be **better than both Selenium IDE and Selenium RC** in many aspects. It implements a more modern and stable

approach in automating the browser's actions. WebDriver, unlike Selenium RC, does not rely on JavaScript for Automation. **It controls the browser by directly communicating with it.**

The supported languages are the same as those in Selenium RC.

- Java
- C#
- PHP
- Python
- Perl
- Ruby



Selenium Grid

Selenium Grid is a tool **used together with Selenium RC to run parallel tests** across different machines and different browsers all at the same time. Parallel execution means running multiple tests at once.

Features:

- Enables **simultaneous running of tests** in **multiple browsers and environments.**
- **Saves time** enormously.
- Utilizes the **hub-and-nodes** concept. The hub acts as a central source of Selenium commands to each node connected to it.