How to conduct a good automation tool research

Prashant Xavier Lopes

Abstract— This paper showcase what all areas should be encountered while doing an Automation tool research. Also, some examples are there for each section of report, so that reader can take a basic idea of look & feel of the content that could be present in the sections.

Index Terms— Automation, Automation research, Automation tool, Automation tool research, Best Automation tool, Comparison between different automation tool, Conclude which automation tool, Research on automation tool, Shortlisting automation tool, Which automation tool is better

1 BACKGROUND

In software industry, testing a software is as important as developing it. An_I0127138_How to conduct a good Automation tool researchd as new features are added to the product, it is equally important that our old features are working as expected.

This testing of existing features (or doing regression testing) consumes huge amount of cost, time & resources. So the best way to save all these is to have a good automation tool in place.

Choosing a "good automation tool" opens a new challenge, as market is full of variety of tools, both free & paid; and making a right choice of tool is essential as it will impact the productivity and deliveries of the project.

We can say that we have to find a perfect needle suited for our purposes from the haystack.

In this paper, I will try to showcase how or what all things should be present in a report that will help you and other stakeholders' shortlist a good tool.

2 INTRODUCTION

While performing such researches, we should be open to evaluate all aspects of different tools and should keep couple of points in back of our head –

- We will not get everything we want in a single tool.
- Cost is a major factor, but not everything.
- The time & resources skills those are available to execute the project.

The next step is to shortlist at-least 10-12 tools. This is a bit complex thing and requires both effort & time. You have to read through several articles, visit various websites and go through many reviews to accumulate a general frame of what all tools are available in the market that suits your need.

Once the tools are shortlisted, your research span for new tool is decreased as now you will only be evaluating/comparing these shortlisted tools.

I would suggest that we should have all variety of tools, say tools having high cost to free, tools supporting many technologies to few and so on.

3 RESEARCH REPORT

Starting with the research report, I will try to highlight different sections a report should have with some examples.

3.1 TOOLS & SPECIFICATIONS

This section should show basic specifications of tools. The specifications could be categorized as –

a) Available since – The time since when tool is in the market. Older could be more reliable.
b) Platforms supported – On what all platforms, it can be operated (Windows, Linux...)
c) Pricing – Showing price pattern for the tool (Free, Seat, Concurrent…..)
d) Application under test – What all applications it can test (Web, mobile…)
e) Scripting languages – Supported languages (Java, C#, Perl…)
f) Programing Skills – Level of skills needed for authoring & debugging the scripts
g) Ease of installation
h) Integrated tools – What all tools it can be integrated with (JIRA, TFS…)
i) Availability of customer support & trainings – Through community, dedicated support...
j) Scripts based on – Scripting flow (Test case/ steps based, Flow based….)
k) User base – Some expected count; can help in guessing how easily the resources knowing the tool are available in the market.

For example –

Product	*ABC*	*DEF*

Available since	*2004*	*1998*
Platforms supported	*Windows* *Linux* *OS X*	*Windows*
Pricing	*Free*	*$ XXXXX/year (Seat licence) + $XXXXX/ year (support)* *$ XXXXX/year (Concurrent license) + $XXXXX/year (support)*
Application under test	*Web apps*	*Web (UI & APIs), Mobile, Desktop, packaged apps*
Scripting languages	*Java, C#, Perl, Python, JavaScript, Ruby, PHP*	*VB script, Ruby, PHP*
Programming skills	*Advanced skills needed to integrate various tools*	*Not required. Recommended for advanced test scripts*
Ease of installation and use	*Advanced skills to install and use*	*Easy to install and use*
Integrated tools	*Using Kovair Omnibus - HP ALM / HP QC, Bugzilla, JIRA, and TFS*	*HP ALM / HP QC,Other ALM tools, Jenkins, JIRA, Bamboo and TFS*
Costumer support/Trainings	*Community*	*24 X 7 Support by MF additionally, partner ecosystem for paid support*
Script based on	*Test case/steps based*	*Test case/steps based*
User base	*100000+*	*45,000 (15000 only in India)*

3.2 SCORECARD

Creating a weightage scorecard is essential step and many companies perform this while shortlisting any vendor from the bunch of available ones.

I would suggest for developing a better understanding; we take demos of the shortlisted tools from the sales peoples before creating a scorecard.

In scorecards, we give weightage to each requirement with respect to tool and compare the scores.

Remember, that weightage for the requirements should sum 100 and points against tool should be given from 1 to 10 (1 lowest to 10 highest).

The total score for any tool is calculated as R1 weightage* Tool Points + R2 weightage *Tool Points…

*For below example, score is calculated as 24*5 + 10*7 + 8*5 +…….=617*

For example –

Requirement	*Weightage*	*ABC*	*DEF*
Compatibility with application under test (object recognition, Operation recognition & Object management)	*24*	*5*	*8*
Platforms supported and tested (Web, API & Mobile)	*10*	*7*	*8*
Ease of use (record & playback/ page scan, keyword/module driven testing, data driven testing)	*8*	*5*	*8*
Popularity (widespread use)	*4*	*10*	*10*
Ease of maintenance	*8*	*6*	*9*
Documentation & training	*5*	*5*	*8*
Support	*8*	*4*	*8*
License cost (low is better)	*15*	*10*	*1*
System requirements	*2*	*8*	*8*
Integration with other tools (bug tracking & test management or others)	*5*	*5*	*8*

Test management & Reporting	*6*	*6*	*8*
Other test supported - Load, performance, DB	*5*	*3*	*7*
	100	*617*	*706*

The scores for different tools can also be represented as graphs which can help in comparing the tools.

3.3 Cost Analysis

This section should show tool cost based on resources, licenses and training needs.

For example -

# Resources	*1*	*6*
Cost	*1400*	*8400*

Training time (in days)	*5*

# Tool licenses	*7*
# Parallel execution licenses	*8*

	Per year					
Tool	*Tool Cost*	*Parallel execution*	*Support Cost*	*Training Cost*	*Cost/month*	*Total cost/year*
ABC	*0*	*0*	*0*	*0*	*8400*	*100800*
DEF	*14000*	*0*	*5000*	*0*	*153400*	*184080*
GHI	*10000*	*0*	*0*	*0*	*18400*	*220800*
KLM	*70000*	*16000*	*0*	*4000*	*98400*	*118080*

***Here,*

*Resource cost = Cost of single resource per year * No. of resources needed*

*Training cost = Cost of training per day * No. of days training required*

*Tool cost = Cost of single license * No. of licenses*

*Parallel execution cost = Cost of single parallel execution license * No. of licenses*

Cost per month = Tool cost + Parallel execution + Support cost + Training Cost + Resource cost

This too can be shown in graphical format.

3.4 Time Estimation

Estimating time of completion for automation activity will help the stakeholders determine whether they want to invest in a more productive but high paid tool or a low paid/ free tool with moderate/low productivity.

The best way I would suggest is to evaluate the time based of test cases and resources availability.

Three point estimation technique would be best to get a respectable figure.

For example –

A product is having 4236 test cases available for automation.

	Authoring rate/day by single resource
Authoring/ Scripting rate when everything is perfect (A)	*8*
Authoring/ Scripting rate when everything is not perfect (C)	*3*
Authoring/ Scripting rate when things are ok but not perfect (M)	*5*

Estimate in days(E)=(A+4M+C)/6	*SD= (C-A)/6*
888.38	*147.08*

**You can also depict this further by confidence level (please read Three-point estimation technique in detail for this.)

Finally, divide it with working days to convert it in months and then with number of resources to estimate the complete time in months.

3.5 Reviews & Ratings

Search a reliable website which share reviews and ratings for the selected tools. This will help you and other stakeholders know the feedback from the users who already used those tools.

For example –

Tool	*Link (Reviews are from XYZ)*	*Ratings*
ABC	*https://www.xyz.com/reviews/ABC*	*4.2*
DEF	*https://www.xyz.com/reviews/DEF*	*4.1*

3.6 Pros & Cons

This tab should show the pros and cons of all the tools.

For example -

Tool	*Pros*	*Cons*
ABC	*1. Low cost* *2. Easy to use as compared to other tools* *3. Business Process Testing supported*	*1. Does not support MAC & Desktop applications* *2. Recording feature for mobile application not available*
DEF	*1. Free tool* *2. Created a mixture of selenium + UFT* *3. Great forum support*	*1. Very new in market and needs may improvements* *2. Dedicated customer support team still being developed*

3.7 Licenses

Knowing the count of licenses needed is very essential as this will directly impact the cost of any tool and we all know that cost is always a major factor in any project.

I am taking here an example of team of 6 members situated in three different locations. Addition to the team licenses, 1 VM license for centralized testing server and 8 parallel execution licenses for cross-browser executions.

Management will always be willing to know how in minimum amount of licenses we can get maximum productivity.

It is important to have a basic idea of different license types as you might need less seat licenses which cost less but more concurrent licenses which cost more but same license can be used by different resources at different times.

Determining this totally depends upon the project strategy and resources availability.

For example -

# Con-current licenses	6
# Seat Licenses	1
# Parallel execution licenses	8

	Gurgaon, India	XXX Eye St. – DC	GTH Penn ave - DC
Automation team	4	1	1

Time-zone	# Licens-es used	Left over	VM Li-cense	Buffer licenses
Night – EST	4	2	1	3
Overlap	6	0	1	1
Morning EST	2	4	1	5

3.8 Suggested Approaches

As any business owner or sponsor would like his managers to come to them with different ideas & solutions rather than problems.

You too cannot just showcase the report; you have to suggest possible approaches. This could be time effective, cost effective, benefits effective etc... ;but should be suggested, so that we can buy-in the votes from stakeholders on what we are thinking & planning rather than implementing on what we are being directed to.

For example –

S.No.	Suggested approach	Rank
1	Saving tool cost, we can go with ABC, keeping in mind that it is a free tool and have a risk that it can be shut down any day.	4
2	Instead of Selenium, we can go for DEF. It is free and a small cost is associated for business support. Risk involve here is that it Is very new in market.	3
3	We can choose GHI having small license cost but having dedicated customer support. Risk involve here is that it Is very new in market and many features are still under development.	2
4	We go for KLM, commercial tool, not much costly and having dedicated customer support. It can run Selenium scripts too.	1

4 Conclusion

The Automation tool is an important tool as it will directly impact the project deliveries. Also, we cannot keep switching tools as re-scripting already created scripts will also be a loss of money, time & resources.

So we should evaluate any tool from all perspectives like cost, specifications etc... and should compare it with the availability of time and resources to make the final decisions.

References

[1] Specification table – Took idea from an article over internet.
[2] Scorecard table – Watched over YouTube video.