1. Is it necessary to set up test automation processes for this SUT? Why?

AT processes are already set up on my current project. If we can imagine that it is not, I will answer that yes.

Testers do not have access to the backend, so testing is only about UI. Most test cases can be automated.

We have a release every week, and every 4 months big release on production. The regression suite is executed once per week, the smoke suit can be run up to 5 times per week. So, thanks to the automated regression manual QA do not have to run it manually at all. It saves a lot of time for mental work and AD hoc testing new features.

1. What should/could be automated for this SUT? Why? How?

As I said most test cases are already automated on my project. My project is about keeping and searching for documents. So, the process of opening, searching, filtering during the search is automated. Also checking the structure of documents, some metadata information, login process, checking permissions are also automated.

Also, document artifacts as title, links, structure, pagination can be automated too.

But there are elements, that cannot be automated. For instance: checking of saved on a computer document. We can check that document is downloaded, but not checking itself.

Also, there are some problems with the possibility to automate elements that have changing structure in DOM.

1. Calculation of ROI

As a base I took real quantity of test cases that already automated. It is 666. This number of TC I put in manual part too.

|  |  |
| --- | --- |
| Parameter | Value |
| Manual |  |
| TС | 666 |
| creation 1 TC | 0.25 h |
| run 1 TС | 0.25 h |
| Automated |  |
| framework | 30 h |
| creation 1 scenario | 18 h |
| execution and analysis | 9 h |
| Project |  |
| weeks | 52 |
| years | 10 |

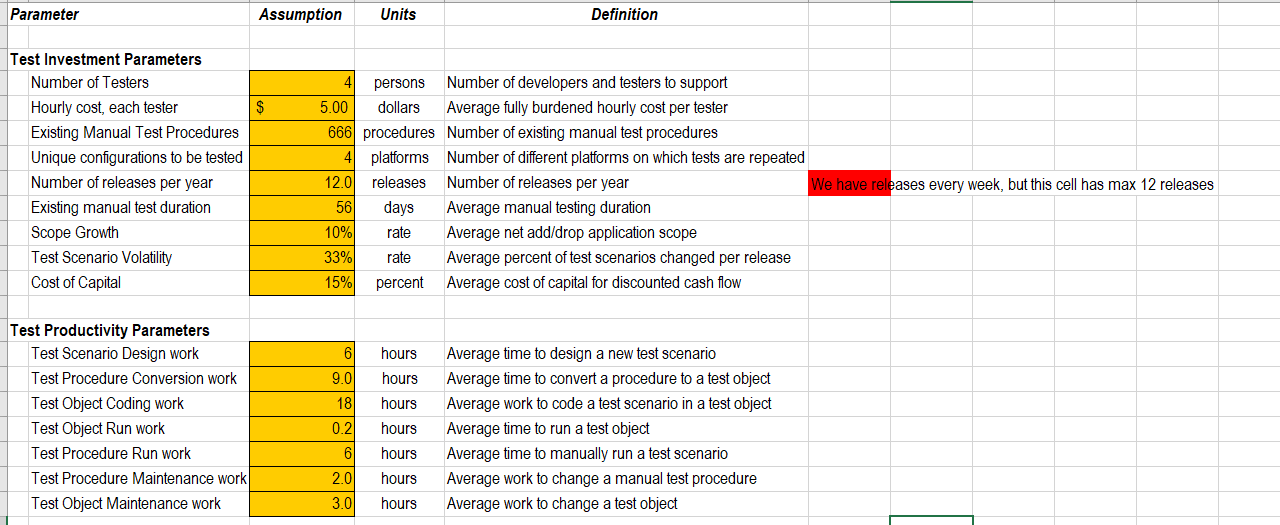
|  |  |
| --- | --- |
| |  | | --- | | CM = 666\*0.25 + 666\*0.25\*52\*10 = 86 746 h  I = 30 + 18\*666 + 9\*52\*10 = 16 698 h  ROI = (86746 – 16698) / 16698 \* 100% = 420% | |

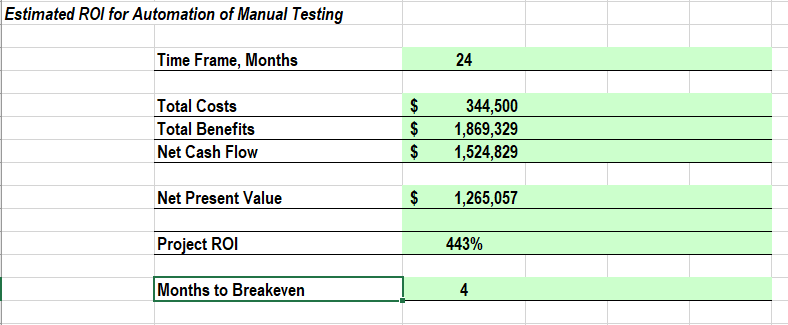
So, I can say that automation on our project is for good and investment will return soon.

But there are some important questions that are not considered in the formula:

1. time for maintenance/update of TA scenarios
2. quantity of releases
3. regression is increasing every sprint
4. manual part is not divided into items, so we really did not know what exactly this part contains
5. number of test cases that do not need to be automated
6. there is no information about the time that the TA scenario needed to pass
7. time for analyzing depends on how many scenarios failed. So, it could not be one number, because one week all passed, the second one a half is failed.
8. scheduled time for refactoring scenarios is not considered. For instance: the situation when the home page is redesigned and rewritten to another language, so all scenarios that connected to the home page will be failed. And it is not a bug.
9. the number of manual QA and TA, their seniority level. Time for coding scenarios depends on experience and qualification.
10. automated regression is bigger than manual one, because most of the times manual QA do not have enough time for full regression, so he has to decrease number of test cases.

To conclude I would like to say that it will be wrong to take just number of ROI from previous formula in consideration about should TA processes will be on a project or not. For these cases I suppose will be better to use more complicated formula or calculator. One of them we have in extras.

1. AutoTest-ROI-Calculator.xls



1. Test strategy

Let us imagine dev team with TA inside, sprint 2 weeks. Testing process will be the following:

|  |  |  |  |
| --- | --- | --- | --- |
| Sprint - week | Dev | QA | TA |
| 1-1 | Development (till the end of the week) | TC creation  Testing | Scenarios writing  Scenarios coding |
| 1-2 | Bug fixing | Testing  TC creation | Scenarios coding  Run + analysis |
| 2-1 | Development  … | TC creation  … | Scenarios writing  ... |

Black box manual testing will be performed. For every feature different branch will be created. After testing all branches will be merged into one release.

Smoke suit will be run after every feature branch is ready.

Regression suit will be run on every release branch (ones per week).

**Test methods:**

Manual testing – new feature testing, integration testing, requirement testing, GUI, Acceptance, Compatibility, AD Hoc

Automated testing – smoke, regression

**Test types:**

Testing requirements, new feature testing, Acceptance testing, GUI testing, Compatibility testing, AdHoc testing, Regression testing, Integration testing

**Test levels:**

Smoke, Critical path, Extended

**Bug tracking:**

Bug can be created by manual and TA QA in Jira.

Bug severity: Blocker, Critical, Major, Trivial