|  |  |
| --- | --- |
| D:\PAL\Logo_4_21_15\Logo_4_21_15\Primary Logo\png_files\EPAM_LOGO_Full_Color_RGB.png | L&D Automated Testing Mentoring Program |

Introduction to test automation

## home task

As a basis for this home task, take SUT (system under test) of your current project.

1. Think about things below:

* Is it necessary to set up test automation processes for this SUT? Why?
* What should/could be automated for this SUT? Why? How?

Write a short essay answering the questions above.

1. Define and collect information and criteria for calculating ROI (Return of Investments) value for this SUT. Provide rationale of your calculation with description on each step.

As a formula for ROI you may use simplified one (that is based on man-hours approach) as:

,

CM – cost of manual testing (man-hours)

I – investments into automation (man-hours)

,

FW – time spent for implementing framework;

S – time spent for creating TA scenarios;

E – time spent for tests execution (human job);

R – time spent for results analyzing.

**Sample of calculation:**

Project X has stable SUT with long-term perspective of supporting as 5 years. Average time required for manual testing is 20 man-hours per week.

Cost of manual testing: 20 man-hour per week \* 52 weeks \* 5 years = 5200 man-hours

TAF Implementation: 80 man-hours

Time spent for creating automated scenarios: 40 man-hours per week during 6 months: 1120 man-hours

Automated test execution & result analyzis: 8 man-hours per week \* 5 years \* 52 = 2080 man-hours

ROI = (5200-(80+1120+2080))/(80+1120+2080)= 58.5%

## BONUS TASK:

1. Use **ROI calculator from additional materials provided “**AutoTest-ROI-Calculator.xls” for detailed ROI calculation.
2. Create detailed automation plan similar to EPAM test plan template.

## acceptance criteria

1. Answer is given for a question: “Why/why not” it’s necessary to set up automation processes for the project?
2. Answers are given for questions: What should/could be automated? Why? How?
3. ROI is calculated.
4. There should be descriptions provided for each step of ROI calculation.
5. There should be clear conclusion about the results of ROI calculation.
6. Bonus task: detailed plan is done according to the template
7. Bonus task: screenshots of detailed calculations in **Online ROI calculator** view are provided to a mentor.

## SOLUTION

System under test is mobile application that helps users to digitally shape their social life. They can use it to discover restaurants, bars and cafés as well as access additional information such as menus. They can also access an overview of their past visits. For individual locations, the app can perform other functions. For example, pay receipts, reservations can be booked. Additionally, users can store vaccination records and tests.  
  
 SUT uses Java, Swift and JS.  
  
 Types of testing need to be performed on the SUT: functional, compatibility, localization, exploratory, regression, mobile, GUI, usability.

1. Why/why not it’s necessary to set up automation processes for the project? - Automating the testing process for the SUT is highly recommended due to the complexity of the application and the number of different types of testing that need to be performed. Automated tests can be run repeatedly and consistently, increasing efficiency and reducing the time and costs associated with manual testing. Automated testing can ensure that the SUT is thoroughly tested across multiple devices and platforms and languages, while reducing the risk of introducing new issues or defects.
2. What should/could be automated? Why? How? – To automate tests for this SUT, we could use a combination of tools and frameworks such as Appium, Selenium WebDriver, TestNG, and Cucumber. Appium is a popular tool for automating mobile application tests, while Selenium WebDriver can be used to automate browser-based tests. TestNG and Cucumber can be used for automating functional and regression tests.  
     
   Regarding what should/could be automated for this SUT, we could start with automating smoke tests such as login, registration, and payment processing. We could also automate compatibility tests to ensure that the application works on different mobile devices and platforms. Localization testing could be automated to check that the application works correctly in different languages and regions.  
     
   Exploratory testing and GUI testing will require manual testing to ensure that the user interface is user-friendly and intuitive. However, we could automate regression tests to ensure that changes or updates to the SUT do not introduce any new defects or issues. By automating tests, we can increase test coverage, reduce testing time and costs, and identify defects earlier in the development cycle.
3. Project has unstable SUT with long-term perspective of development and supporting as at least 5 years. Average time required for manual testing is 25 man-hours per week.  
     
   Cost of manual testing: 25 man-hour per week \* 52 weeks \* 5 years = 6500 man-hours.  
     
   TAF Implementation: 40 man-hours.  
     
   Time spent for creating automated scenarios: 40 man-hours per week during 7 months: 40 \* 30 = 1200 man-hours.  
     
   Automated test execution & result analysis: 8 man-hours per week \* 5 years \* 52 = 2080 man-hours.  
     
   ROI = (6500-(40+1200+2080))/( 40+1200+2080) = (6500 – 3320) / 3320 = 95,7%
4. An ROI value of 95.7% for setting up test automation processes means that the benefits of implementing test automation are almost double the cost. This indicates that implementing test automation is likely to be financially beneficial for the project in the long run, and the benefits outweigh the costs. Therefore, implementing test automation processes is a good investment for the project.