Gamanza technical assignment

1. What bug reporting tools have you used in the past and what do you recommend the most and why?

I have used Jira for my faculty projects always as it proved to be the most simple and yet most effective tool with a fairly easy framework. I have also used Gitlab, Github and Clickup, however I quickly returned to Jira. Therefore, I always recommend it to someone that asks me this question as I find it best suitable for myself. I am a simple man and Jira provides me with all I need from a bug reporting tool.

1. Are you familiar with test scenarios? Have you ever used any tool for that? How would you introduce writing test scenarios for our apps?

Yes, I am familiar with test scenarios. Since Testrail requires a payment - I've used Google Sheets for writing test scenarios so far. This way, it was easy to share it with my colleagues that were working with me on the same project (faculty project). It was easy to write the test cases with steps in separate sheets. In addition, Google sheets support photos and short Gif videos, and adding them for some cases was easy. As for introducing test scenarios, I would definitely use Testrail as I know that most firms today use it and there is a good amount of documentation to get the team started on it.

1. How important is writing test scenarios for you? Why?

Very important. Without having test scenarios we are uncertain whether we have tested the full functionality of our app. We need them written down in order to stay in track of all use cases possible.

1. Assume you find a problem while testing the application, what would be your recommended approach to get that problem resolved?

First, I recommend trying to reproduce this bug or if it is a problem try again to see what is causing it. Once that is done, I recommend writing a detailed report to explain what the bug/problem is and where it occurred and in what environment to reproduce it. Specify the priority and contact the developer that will be responsible for fixing it.

1. You find and report a problem within the application, but the developer doesn’t believe this to be a problem and resolve the task as won't fix? How would you deal with this scenario?

I would contact my superior and consult with the disagreement or if possible, I would contact the client since he knows best what he wants.

1. It is late in the evening. You are responsible for the GO/NO GO decision. The workflow is very strict and there are numerous tasks to be done:

• Verifying bugs found in the very last stage of testing

• One round of regression testing, etc.

• Conclude the testing

• Filling out the report,

• Check everything is ready for release

You just discovered that some part of the app doesn't feel right - UI/UX it's not catchy enough. What do you do? Why? You know if following the procedure the app won’t be released on time, but timing is everything on this release. What do you do? Why?

-I would check the bugs with highest priority/severity

-I would only do regression testing for the most fragile part of the app

-I would then check if everything is ready to be merged properly

The reason is due to restricted time available and there is a hurry in the decision. Other things are just time wasting in this dire situation and can be done afterwards.

1. Assume you are the QA lead for a new major product with a set project deadline. Briefly describe how you would go about the following:

• Implementing QA process into the project

• Ensuring product successfully passes QA

• Making sure the project deadline is met.

* + I would start with getting a clear idea with my team on what the process should look like. Go over the requirements in great detail so we are in track with what the functionalities and non-functionalities will be. As soon as the software requirements are grasped, I would then start planning the test cases with my team-Then we would continue with designing the test cases. We would then execute them and report bugs and defects found along the way. After running the tests, we test them again to see if we missed anything and do regression testing after the fixes. Once the developers confirm the project is due for release, my team and I would perform release testing
  + I would keep in contact with the developers, keep consulting the requirements document, and make regular checkups on the team to follow their progress in the QA process. Here the communication is extremely important between all parties.
  + I would introduce automated testing into the project to save unnecessary time wasted on low level manual testing. Scrum sprints are a great way to follow the team's progress while also giving the team a greater picture on time organization.

1. Assume that only one area of the application has been changed, how much time do you think should be spent testing each segment of the app (new functionality, core functionality, all functionality, etc.)? Give an estimate in % for each segment and explain your decision.

Core functionality should be tested always when introducing new features as problems might occur, therefore it should take 30% time of the testing. Usually, we have prepared test cases and test scenarios for this so it does not take much time.

New/Updated functionality should be tested 50% of our time as it is the part we are trying to introduce and should pass all tests possible. Therefore, it is deemed important to spend most time on it.

All functionality of the application should be tested once we have finished with the core and new functionality therefore, it should be tested with the remaining 20% of the time needed for the whole process.

1. Are you familiar with any test automation tool? Have you ever used automated testing in apps? How would you introduce automated testing to our apps?

I am familiar with Cypress and Selenium. I have never used automated testing before since my projects in faculty did not require it. However, I have done research and watched YouTube tutorials on both of these automation tools to get familiar with them since I will need to use them one day. I have to mention also that I am currently finishing a course on Software testing and am finishing week 1, and the automation is planned for week 4, therefore I am definitely going to get a grasp on it. I would introduce automated testing through Cypress as the framework is much simpler than Selenium.