|  |  |  |
| --- | --- | --- |
| |  | | --- | |  | |  |

***Scenario:***

***You have been hired as a QA Engineer for a startup company that is developing a new financial application. The company is in the early stages of growth, and the development team is still finding its footing. As the QA Engineering, you will be responsible for establishing quality assurance processes and collaborating with stakeholders to ensure the application meets their requirements.***

***Challenge:***

1. ***Leadership and Startup Mindset: Describe your approach to building a quality-focused culture within a startup environment.***

First understand the dynamics of startups since they have a fast pace and their resources are limited, quality processes must be implemented where the product is assured reliably and at the same time is flexible to be able to make changes quickly.

Among the strategies that must be taken into account are training at all levels on best practices, the standards used with resources such as tutorials, user manuals.

At the same time, establish a quality process that ensures clear quality standards from the creation of the processes, their documentation is clear and they can be tested.

Furthermore, in the development process there must be unit and integration tests where its quality is guaranteed. Since quality is not only the duty of the QA, but it begins from the very writing of the code by the developers, as well as the other actors immersed in the project, creating user stories and requirements in a reliable way so that can be developed in the same way. This can be achieved through periodic and retrospective reviews that allow the product and its processes to be evaluated.

Based on existing tools and processes that allow monitoring the product, as well as its flow through the implementation of agile practices such as SCRUM that allow organizing work in sprints or small projects that allow an iterative and incremental approach to development and in turn testing ensuring that it is continually being improved, and with the implementation of Continuous Integration and Deployment it is guaranteed that each change in the code is automatically tested, quickly identifying problems.

It should also be taken into account that there are communication tools such as JIRA, Slack, Trello, Confluence where everyone can have access so there is a defined and unified criterion that supports the follow-up and alignment meetings that must be held regularly. periodically.

1. ***Communication and Stakeholder Management: Describe your strategy for effectively communicating with technical and non-technical stakeholders.***

Communication is important between all members involved in the process since unified criteria must be ensured by all parties involved, whether these techniques are used or not.

My strategy is to maintain clear information channels, adapting language and information to the specific audience, thus encouraging collaboration.

Through the generation of periodic meetings by the different groups of interested parties, whether daily in nature, where the monitoring of the topics, their blockages, the daily progress and the priorities by development, QA and PO are exposed in case of using agile methodology. SCRUM, weekly in nature to review technical topics and other biweekly in nature where the topics to be carried out in the following cycles are presented, the summary of progress in the current sprint or work cycle, taking into account the behavior of the previous sprint either positively and/or negatively.

Keep in mind that there are tools to establish clear channels in which you have access, such as Slack (Quick Communication), JIRA (Project Management), Confluence (Documentation Centralizer).

To adapt the language to non-technical people, clear, simple language without technicality should be used, presenting summaries and/or reports showing project information at the level of achievements, risks, risk mitigation management, and perhaps with images or graphics that support the presentation and easy understanding of what is shown.

Clear and precise documentation must also be shown to technical-level people about the status of the processes, such as test coverage, reported defects, resolved defects, etc. As well as test cases, results of the tests executed must also be shown.

1. ***Quality Assurance Strategies: Discuss your approach to creating a comprehensive quality assurance plan for your application.***

In creating a comprehensive quality assurance plan for a new financial application, it is crucial to have solid strategies that can generate a reliable and strong product to apply the plan, we can take these steps:

* 1. **Analysis of requirements**

This stage involves holding meetings with the developers, designers, product manager and other stakeholders to understand the functional and non-functional requirements of the application. At this time, detailed documentation is created about the requirements such as technical specifications, use cases and clear acceptance criteria, in addition to a risk analysis that affects the quality of the product, as well as the security of the application considering that it is directed. to finances.

* 1. **QA processes**

At this point it is necessary to establish a framework for the QA processes that will be used where the tests to be executed are planned, as well as defects and results reporting. At this same level we will define the QA testing strategies to be used such as unit tests, integration tests, system tests, user acceptance tests (UAT) and regression tests.

Additionally, it will be established how test automation will be carried out, that is, how the test automation tools and frameworks will be defined that allow improving their efficiency and coverage, especially for repetitive tests and in which environments they will be used (QA, UAT).

* 1. **Test Planning**

At this stage, the creation of the test plan begins, which details the scope of the tests, the types of tests to be performed, the necessary resources, the dates and schedules of tests, and the expected deliverables. The design of test cases is also carried out, which includes functional and non-functional scenarios, including performance, security and usability tests, where the test data to be used is defined, taking into account that the security and confidentiality of the data must be maintained. These are sensitive data.

**D. Testing execution**

Here the tests that will be used are defined, such as the following:

* *Unit Tests:* Verify that they are implemented and executed effectively in collaboration with developers.
* *Integration Testing:* Ensuring that the different modules of the application work correctly when combined.
* *System and UAT Testing:* Execution of complete systems tests with end users ensuring product quality according to expectations *.*
* *Automation and CI/CD:* Detect and fix defects quickly when integrating automated tests into the integration pipeline (CI/CD)
  1. **Defect Management**

A defect management tool should be used where defects found in testing can be recorded, prioritized and monitored, with their respective root cause analysis to identify which process is causing this defect, its fix by development and its subsequent verification, which was effectively corrected, and no new ones were introduced.

* 1. **Report and Communication.**

Different reports and interactions must be created where the status of the tests is delivered to different members, these reports and interactions would be:

* *Test Results Report:* It is a detailed report of the test results including metrics, statistics and risk assessment.
* *Feedback Meeting:* Regular meeting where test results, defects found and strategies to improve quality are discussed with the development area and interested *parties*
* *QA Documentation:* It is maintaining updated documentation related to QA processes, test cases, results and lessons learned.
  1. **Continuous Improvement.**

In this stage, a review and adjustment is made to the QA processes on a continuous basis based on the feedback received, and continuous training and skill development is promoted within the QA team to stay updated with best practices and new technologies.

**Conclusion.**

The assurance plan aims to successfully ensure that the financial application is secure, functional and reliable before its release into production. Through a combination of testing strategy, clear and structured QA processes based on clear and effective communication from each of the team members and stakeholders.