OPERATIONS
COE COMITEE
DIVISION

QA Processes CoEs

LEANTECH

Revision History

Revision ID	Description / Change	Name	Date
1	Document created	Luis Lozano	10/13/2023

Purpose

This document serves to outline the fundamental purpose and objectives of our Quality Assurance (QA) processes within our organization Here. Quality Assurance is a critical component of our commitment to delivering exceptional products and services to our customers, ensuring consistency, reliability, and adherence to industry standards. The primary purpose of this document is to provide clarity on our QA processes.

Disclaimer

The information presented in this document is intended for internal use only and is subject to change as our QA processes evolve. Specific projects or departments may have unique QA requirements, which will be addressed separately.

Annexes

Annex ID	Document Description	Annex URL
1	QA Workflow	Link

QA Material.

General Information

- All QA members should store their deliverables and reports in this location. Here
- All QA members should have enough time to perform regression testing, if not please contact QA CoE here

RASCI Matrix

A RASCI matrix is a simple tool used to clarify and document the roles and responsibilities of individuals or groups involved in a project or process. It assigns one of five roles to each party: Responsible (R), Accountable (A), Support (S), Consulted (C), or Informed (I), making it clear who does the work, who makes decisions, who provides help, who offers input, and who receives updates. This helps improve accountability, communication, and collaboration within a team or organization.

Responsible (R): The person or team responsible for executing the task or activity. They are the individuals who perform the work and ensure it is completed as required.

Accountable (A): The person who is ultimately answerable for the task or activity's success or failure. They make the final decision, give approval, and are responsible for the overall outcome. There should only be one "A" for each task.

Support (S): Those who provide assistance, resources, or information to the responsible party to help them complete the task. Support roles collaborate with the responsible person but don't have the ultimate decision-making authority.

Consulted (C): Individuals or groups who need to be consulted before decisions or actions are taken but don't have the direct responsibility for execution. Their input and expertise are important, and their opinions are considered before a final decision is made.

Informed (I): People who need to be informed about the task's progress or outcome but don't directly participate in the task's execution or decision-making. They are kept in the loop to stay aware of developments.

Phase	Activity	PM	Development	BA	QA
Analysis	Reviewing the software requirements document (SRD) and other related documents	A	I	C	R
	Identifying any ambiguities or inconsistencies in the requirements	A	I	C	R
	Read SOW if available	A	I	R	I
Planning	Creating test plan	I	I	C	R
Test case development	Test case Design	I	I	C	R
Set Up Environment	Set up of the QA Environment	I	R	I	I
Execution	Testing execution	I	I	I	R
	Loggin defects	A	I	C	R
	Sprint Report	A	I	I	R
Closure	Go / No Go meeting	A	A	A	R
	Sign Off	A	I	I	R

Phases & Activities

Analysis Phase

Analysis: In this phase the software quality assurance team understands the requirements like what is to be tested. If anything is missing or not understandable then the software quality assurance team meets with the stakeholders to better understand the detailed knowledge of requirements.

Activities:

- ➢ Reviewing the software requirements document (SRD) and other related documents
- ➢ Identifying any ambiguities, inconsistencies or missing requirement
- ➢ Read the SOW document

Inputs:

- ➢ Epics
- ➢ User stories with the Acceptance criterias
- ➢ Designs

Deliverables:

- ➢ N/A

Planning Phase

Planning: in this phase is to determine the test plan strategy, the scope of testing, risks, tests deliverables, for the project. Moreover, the resources, test environment and test limitations are also determined.

Activities:

- ➢ Defining the testing objectives and scope
- ➢ Developing a test strategy: selecting the testing methods and techniques that will be used
- ➢ Defining the testing environment and resources needed (QA team and tools)
- ➢ Defining the test deliverables
- ➢ Defining Risks and define a mitigation plan
- ➢ Reviewing and approving the test plan (reviewers QA Manager, PM)

Inputs:

- ➢ Epics
- ➢ Designs
- ➢ User stories or Acceptance criteria refined and approved by the product owner and team

Deliverables:

- ➔ Test Plan

Design

Test cases development: In this phase the testing team notes down the detailed test cases. The testing team also prepares the required test data for the testing. When the test cases are prepared then they are reviewed by the quality assurance team.

Activities:

- Writing test cases that are clear, concise, and easy to understand
- Creating test data and test scenarios that will be used in the test cases
- Reviewing and validating the test cases (with other QA members or DEV,BA)

Inputs:

- Epics
- Designs
- User stories or Acceptance criteria refined and approved by the product owner and team

Deliverables:

- Test cases

Recommendations :

QASE.IO (free version available)
TEST RAIL
Recommendations

Execution Phase

Execution in this phase testing team starts executing test cases based on prepared test cases in the earlier step.

Activities :

- Test execution: The test cases and scripts created in the test design stage are run against the software application to identify any defects or issues.
- Defect logging: Any defects or issues that are found during test execution are logged in a defect tracking system, along with details such as the severity, priority, and description of the issue.
- Test data preparation: Test data is prepared and loaded into the system for test execution.
- Evidence Gathering : The test evidence is collected.
- Defect retesting: Any defects that are identified during test execution are retested to ensure that they have been fixed correctly.
- Test Reporting: Test results are documented and reported to the relevant stakeholders.

Inputs:

- ➢ Test cases approved by QA Lead
- ➢ Access to the Database
- ➢ Access to the API
- ➢ APK if the application is for Android / IOS

Deliverables:

- → Test cases updated with results

- Test Evidence should be in the google project drive folder.
- End of each Sprint - Sprint Test Report (in the tool you use, such as Jira, Azure, Monday, ClickUp, etc.)
    - Executive Summary: A brief summary of the test status. Indicate whether the sprint's test objectives were met. Highlight key points, such as critical issues or notable achievements.
        - Test Coverage: Description of test coverage (functional, non-functional, regression, etc.).
        - Percentage of test cases executed. Percentage of successful test cases.
        - Areas or functionalities that couldn't be tested and reasons.
    - Defects and Issues:
        - Total number of defects found.
        - Classification of defects by severity (critical, major, minor).
        - Current status of defects (open, in progress, closed).
        - Summary of the most critical or urgent defects.
    - Execution Statistics:
        - Sprint duration.
        - Number of planned test cases vs. executed test cases.
        - Average defect resolution time.
        - Test execution time by type (automated vs. manual).

Recommendations.

- Bug detailed report (just informative document on how detailed a bug should be)

Closure Phase (for release / project ends)

Closure : in this phase all testing-related activities are completed and documented. The main objective of the test closure stage is to ensure that all testing-related activities have been completed and that the software is ready for release.

Activities:

- Defect tracking: All defects that were identified during testing are tracked and managed until they are resolved.
- Sign-Off : It is the process by which formal approval or consent is given to indicate that the tests meet certain criteria and that the software has successfully passed the testing process.
- Go/No Go meeting : is a decision-making meeting where its determine wheter it should proceed for a release ("Go") or be halted ("No Go"), QA team explain the risks if any for the release.

Inputs:

- Bugs report
- Test cases execution report

Deliverables:

- Certification letter
    - Concurrencia de usuarios que soporta la plataforma
    - Funcionalidades acordadas (acorde al scope), que se aprobo (pasaron y cuales no)
    - Temas de seguridad
    - Temas responsivos (mobile o no)
    - Si hay algun tema de seguridad

- Resultado de las pruebas de performance
- Si hubo herrores cual es la magnitud de los errores y las posibles soluciones - como impacta eso en las funciones, queda el sistema trabajando?

- *Deployment inform - Pending to check the value of this deliverable*
- *Production Report KPIS*
- *Any other deliverbale?*

QA Policies

1. The QA workflow must be defined at the beginning of the project and must be executed as such in every sprint or iteration of the project.
2. The Bug/Issue workflow must be defined at the beginning of the project and must be executed as such every sprint or iteration of the project.
3. Quality effort and/or time must be included in the estimate of tickets to be worked in each sprint or iteration.
4. The test plan should be created and/or updated at the beginning of the project or when necessary due to changes in project objectives or quality.
    a. Parts of the Test Plan
        i. Objective
5. Scope
6. Test Methodology
7. Approach
8. Assumption
9. Risk
10. Mitigation plan
11. Role and responsibility
12. Defect tracking
13. Test Environment
14. Test Automation
15. Effort Estimation
16. Test Deliverable
17. The test plan must be reviewed by other QA team members or the QA Lead/Manager
18. Should have Manual tests and Automated Tests for regression Testing

19. Every execution testing should have a certification letter or report about the QA process in the sprint or Iteration
20. User stories / tickets tested
21. Total test cases / Scenarios executed
22. Total Bugs
23. Total Bugs fixed
24. Growth of knowledge in the latest technologies
25. Must have three environments (Dev, STG, UAT) if the project is ready to production should have PROD environment too

Good practices

1. Setting Testing Goals and Requirements
2. Establishing a Well-Designed Workflow
3. Performing Both Manual and Automated Tests
4. Use integration and unit testing
5. Conducting Risk-Based Testing
6. Implementing Regression Testing
7. Create A List Of Negative Scenarios
8. Test Across Multiple Browsers and Devices
9. Cooperating with a Development Team
10. Incorporating agile methodologies into software testing

11. QA testing & training for quality engineers

12. Choose a QA management tool

13. Using design patterns in the automation project

   a. Factory Method
   b. Abstract Factory
   c. Singleton
   d. Composite
   e. POM

14. Using Test framework

15. Use continuous integration and continuous delivery (CI/CD)

16. Continuously Improve the QA Process

   a. Manage and Maintain Documentation
   b. Test plan
   c. Certification Letter
   d. User Guide
   e. Test cases / Scenarios
   f. Technical documentation

17. Implement Exploratory and Ad-hoc Testing

Analysis	Action plan	Project
The test plan does not have a good structure	Use the same test plan structure for every project	
The project does not have defined the QA workflow	Propose the QA workflow to the project	

The project does not have the Bug workflow	Propose the Bug workflow to the project	
The project has a different methodology for working	Know about how is the current workflows	Mansueto Ventures
The automated testing is nor implemented	Research why the project has not implemented automated testing	
The project does not have performance testing	Research why the project has not implemented Performance testing	
The project does not have API testing	Research why the project has not implemented API testing	
The QA team does not create Database test cases	Verify if those queries are executed such as precondition	