AUTOMATION OF QA ACTIVITIES

1. Introduction

1.1 Purpose of this Document

This document outlines the functional and non-functional requirements for the QA Automation Module of the ERP Tool. The purpose of this document is to serve as a single reference point for all stakeholders including developers, testers, designers, QA engineers, and project managers by clearly defining the scope, features, constraints, and operational flow of the system. This module focuses on automating critical QA activities such as plan document handling with version tracking, inspection requisition workflows, collaborative review and feedback mechanisms, task assignments, and report archival. It also includes real-time dashboards and secure, role-based access control. This document ensures that the development team and end users share a common understanding of the system's objectives, expected behavior, and compliance requirements.

1.2 Scope of the Document

This document defines the scope, requirements, and structure of the QA Automation Module, which aims to replace the existing manual and paper-based quality assurance processes with a secure, digitized solution integrated into the organization's ERP system. The module automates plan document uploads, reviews, and alphabet-based version tracking; manages the end-to-end lifecycle of inspection requisition forms with internal and external test categorization; and supports structured, comment-based collaboration between QA and design teams. It also centralizes the storage and archival of inspection reports and test results, provides real-time dashboards for monitoring requisition statuses and document history, and ensures secure, role-based access control and audit logging throughout the workflow.

The scope includes:

- Upload, review, and track versions of Plan Documents using alphabet-based revision control.

- Submit, approve, and track Inspection Requisition Forms, including internal/external test categorization.

- Collaborate through comment-based feedback for document reviews and QA responses, with structured observations and justification workflows.

- Assign inspection tasks and update their status throughout the lifecycle.
- Upload and archive inspection reports and test results for secure storage and reference.
- Track requisition statuses, review comments, and document history through interactive dashboards, including turnaround metrics.
- Enforce role-based access control with secure operations and detailed activity logs.

1.3 Overview

The QA Automation Module is a secure, web-based solution integrated into the organization's ERP system, designed to digitize and streamline the entire QA lifecycle. It replaces manual processes like Excel tracking and paper-based requisitions with a centralized workflow for uploading and reviewing Plan Documents, managing inspection requisitions, and archiving test reports.

The system supports version-controlled document handling, structured comment-based collaboration between QA and design teams, and categorization of internal and external tests. Dashboards provide real-time visibility into requisition statuses, document history, and turnaround metrics. By automating these processes, the module improves efficiency, ensures traceability and enhances collaboration.

2. General Description

The QA Automation Module is a key component of the ERP system designed to digitalize and streamline the quality assurance workflow. It replaces manual processes such as Excel-based tracking and paper records with a secure, centralized platform that ensures traceability, accuracy, and efficiency.

2.1 Product Functions

The QA Automation Module provides the following core functionalities:

- Upload and review of Plan Documents and detailed Inspection Requisition Forms.
- Version control with change history tracking and timestamped revisions.
- Structured, comment-based collaboration between QA and design/development teams.
- Inspection task management, including scheduling of internal tests and assignment to responsible teams.

- Requisition review workflows with accept/reject actions, reviewer comments, and QA admin authentication.
- Capture of key requisition details such as project name, part number, test type, test venue, date, and reference documents.
- Handling of external tests with support for uploading external request letters.
- Exportable and printable test documentation, verified and approved by the QA admin.
- Real-time dashboards for the design team to monitor requisition statuses, QA comments, and document submission history.
- Real-time dashboards for QA team members to track their assigned tasks and progress.
- Performance dashboards for QA managers and admins to monitor team workload, task status, and review efficiency.
- Secure, role-based access control for all stakeholders, ensuring proper data visibility and authorization.

2.2 User Characteristics

Users include:

Design/Development Team

- Submit Plan Documents and Inspection Requisition Forms.
- View dashboards to track submission statuses and QA feedback.
- Respond to review comments with accept/reject actions and justification.

QA Team / Reviewers

- Review submitted documents and provide structured comments.
- Classify observations as Minor or Major.
- Handle inspection tasks and update inspection status.
- View personalized dashboards for task tracking.

QA Manager

- Add and manage internal test details.
- Monitor the status of inspections and document reviews.
- Access and analyze team performance data, activity logs, and reports.

Admin

- Approve or reject Inspection Requisition Forms and plan submissions.
- Add internal test details and assign QA team members.
- Monitor overall QA workflows and process status.
- Access comprehensive data logs, audit trails, and reporting features.

The system is designed for ease of use with minimal training, featuring a clean and intuitive interface for all roles.

2.3 Features & Benefits

- Digitizes QA workflows to eliminate manual tracking and enable faster, more accurate processing of documents and inspections.
- Tracks the complete lifecycle of requisitions, plan document revisions, inspections, and QA responses with full version history and turnaround metrics.
- Enables structured communication between QA and design teams through contextual, comment-based collaboration with actionable feedback.
- Supports both internal and external testing processes, including scheduling, task assignment, and documentation with necessary attachments and approvals.
- Centralizes storage of QA artifacts, including test reports, requisitions, and plan documents, with export and print capabilities for audit or reporting needs.
- Provides real-time dashboards for design teams, QA members, managers, and admins to monitor statuses, team performance, and document activity.
- Enforces secure access control and ensures auditability through role-based permissions and detailed activity logs.

2.4 Importance

This module plays a critical role in modernizing quality assurance practices within the organization. It enhances coordination between design and QA teams, improves traceability, and significantly reduces turnaround times. By automating inspection workflows, requisition handling, and document collaboration, the system ensures data integrity and process transparency. Ultimately, it supports the delivery of products with greater efficiency while minimizing delays associated with manual tracking and fragmented communication.

3. Functional Requirements

This section outlines the core functions of the QA Automation Module, describing the expected behavior of the system, and how each functional unit contributes to the automation of the QA process. Each requirement includes the data involved, its source, and the expected system response.

3.1 Document Upload and Review

3.1.1 Document Upload

Function:
Allow design/development teams to upload Plan Documents into the system with version tracking.

Inputs:

- PDF or DOCX files from authenticated users
- Project metadata (project name, LRU name, Doc num, Doc name, Version & Revision, date)

Processing:

- Files are validated, assigned versions, and stored with timestamps
- System automatically logs changes when updated versions are submitted

Outputs:

- Versioned document entries visible to authorized users
- Notification to QA team for review

Validations:

- File format must be .pdf or .docx
- Mandatory fields (e.g., project name, LRU name, Doc num, Doc name, Version & Revision, date) must be filled

3.1.2 Document Review and Feedback

Function:
Enable QA team to review submitted documents and provide approval, rejection, and comments.

Inputs:

- Reference ID of document
- Project Metadata (e.g., project name, LRU name, Doc num, Doc name, Version & Revision, date)

Processing:

- System logs reviewer details, status, timestamp
- Stores feedback as version-specific comments

Outputs:

- Comments and status (Approved / Rejected / Needs Revision) with pop up comments and final consolidated observation document.
- Status updates to the design team and QA team dashboard
- Comment log for historical tracking

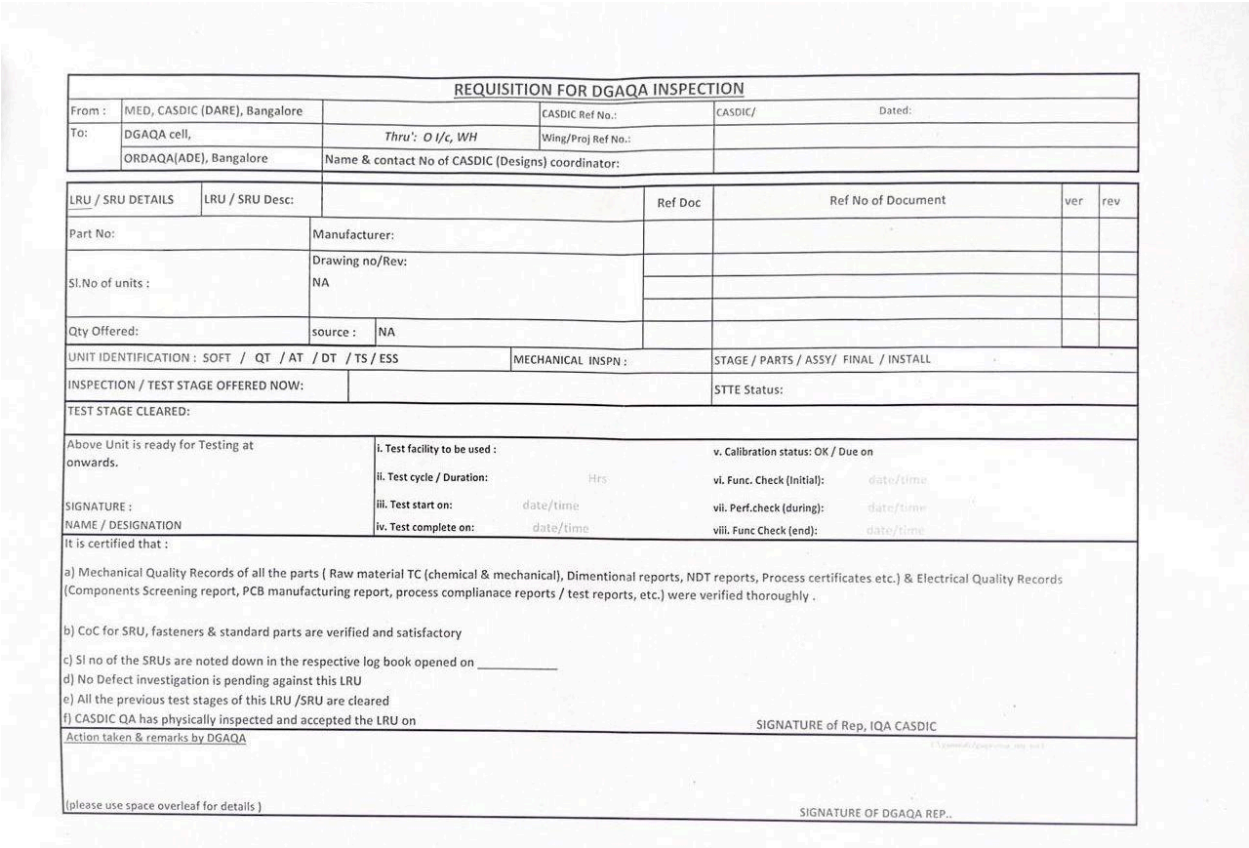
3.2 Test Requisition and Inspection

3.2.1 Test Requisition Submission and Tracking

Function:
Allow design teams to submit inspection requests post-document approval.

Inputs:

- Requisition form data (Reference no., Project no., LRU Name,part no., SL no.,Quantity,Reference documents, type of test,test stage offered, cleared test stage, venue, schedule(start and end),date, duration, attached references, authentication)



REQUISITION FOR DGAQA INSPECTION

From :	MED, CASDIC (DARE), Bangalore		CASDIC Ref No.:	CASDIC/ Dated:
To:	DGAQA cell,	*Thru': O I/c, WH*	Wing/Proj Ref No.:	
	ORDAQA(ADE), Bangalore	Name & contact No of CASDIC (Designs) coordinator:		

LRU / SRU DETAILS	LRU / SRU Desc:			Ref Doc	Ref No of Document	ver	rev
Part No:		Manufacturer:					
Sl.No of units :		Drawing no/Rev: NA					
Qty Offered:		source :	NA				

UNIT IDENTIFICATION : SOFT / QT / AT / DT / TS / ESS	MECHANICAL INSPN :	STAGE / PARTS / ASSY/ FINAL / INSTALL
INSPECTION / TEST STAGE OFFERED NOW:		STTE Status:
TEST STAGE CLEARED:		

Above Unit is ready for Testing at onwards.	i. Test facility to be used :	v. Calibration status: OK / Due on
	ii. Test cycle / Duration: Hrs	vi. Func. Check (Initial): date/time
SIGNATURE :	iii. Test start on: date/time	vii. Perf.check (during): date/time
NAME / DESIGNATION	iv. Test complete on: date/time	viii. Func Check (end): date/time

It is certified that :

a) Mechanical Quality Records of all the parts (Raw material TC (chemical & mechanical), Dimentional reports, NDT reports, Process certificates etc.) & Electrical Quality Records (Components Screening report, PCB manufacturing report, process complianace reports / test reports, etc.) were verified thoroughly .

b) CoC for SRU, fasteners & standard parts are verified and satisfactory

c) Sl no of the SRUs are noted down in the respective log book opened on _____

d) No Defect investigation is pending against this LRU

e) All the previous test stages of this LRU /SRU are cleared

f) CASDIC QA has physically inspected and accepted the LRU on SIGNATURE of Rep, IQA CASDIC

Action taken & remarks by DGAQA

(please use space overleaf for details) SIGNATURE OF DGAQA REP..

Processing:

- System checks for Plan Document approval status before submission
- Accepts or rejects based on form completeness and previous test status.
- QA admin reviews and adds internal test plans or rejects with reasons

Outputs:

- Requisition request status: Pending / Approved / Rejected along with comment
- Notification to design team and QA team on status update.
- Trackable status for design team

Valid Range:

- Test types : SOFT, QT, AT, DT, FT, TS, ESS, LQT
- Schedule date: Must be ≥ current date

3.2.2 Task Assignment and Inspection Management

Function:
Enable QA head to assign inspection tasks and QA team to update statuses, and upload results.

Inputs:

- Test ID (linked to requisition)
- Status update (Not Started / Initiated / Completed)
- Inspection result files of previous test stages (PDF,docx)

Processing:

- Task assigned to specific tester(s)
- Progress tracked with time stamps

Outputs:

- Inspection dashboard update
- Final inspection report
- Result files linked to the original requisition

3.2.3 Comment-Based Communication

Function:
Allow QA and design teams to exchange feedback or clarification via contextual commenting.

Inputs:

- Comment text
- Associated document or task ID

Processing:

- Stored with metadata (QA Reviewer ID, Page Number, Section Number, QA Comment, and Classification (Minor/Major) on the QA interface. On the Design interface, additional fields are captured such as: Justification Comment, Accept/Reject Status, Updated Page Number, and Updated Section Number)
- Viewable only by users with permission for that document/task

Outputs:

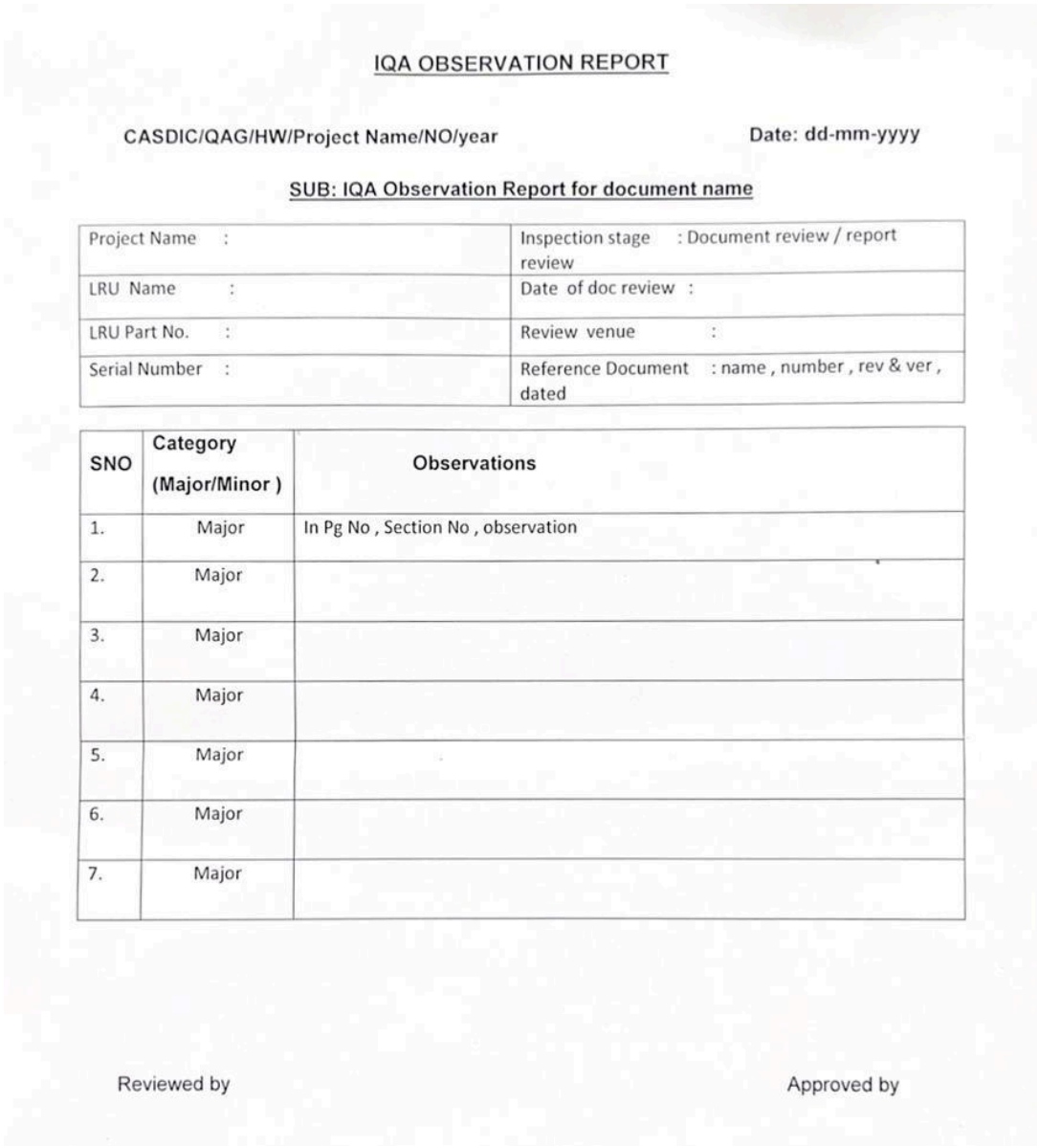
- Real-time update in document/task thread
- Notification alert to relevant user(s)

3.2.4 Report Generation and Archival

Function:
Store and retrieve structured test reports and inspection outcomes.

Inputs:

- Final test report files
- Metadata (Project Name, LRU no.,No. Of observation, Year, Date)



IQA OBSERVATION REPORT

CASDIC/QAG/HW/Project Name/NO/year Date: dd-mm-yyyy

SUB: IQA Observation Report for document name

Project Name :	Inspection stage : Document review / report review
LRU Name :	Date of doc review :
LRU Part No. :	Review venue :
Serial Number :	Reference Document : name , number , rev & ver , dated

SNO	Category (Major/Minor)	Observations
1.	Major	In Pg No , Section No , observation
2.	Major	
3.	Major	
4.	Major	
5.	Major	
6.	Major	
7.	Major	

Reviewed by Approved by

Processing:

- Files validated and stored
- Tagged with searchable parameters

Outputs:

- Downloadable/exportable reports (PDF, Docx)

3.7 Role-Based Access Control (RBAC)

Function:
Restrict operations based on user roles (Design, QA Reviewer,QA Manager, Admin)

Inputs:

- User login credentials

Processing:

- Authentication and role mapping
- Permission enforcement for each operation

Outputs:

- Access granted or denied based on role and activity type
- Audit logs updated

3.8 Audit Trail and Activity Logs

Function:
Maintain logs for all major actions (upload, review, inspection, comments)

Inputs:

- User actions

Processing:

- Logged with timestamps, user identity, and action description

Outputs:

- Viewable and queryable logs for Admin and QA Manager to monitor all system activities.
- Exportable for audit purposes

4. Interface Requirements

The QA Automation Module interfaces with existing systems and applications to ensure smooth data flow, user interaction, and operational continuity. The interface requirements are outlined as follows:

4.1 User Interfaces

- Web-based, role-specific dashboards for Design Teams, QA Reviewers, QA Managers, and Admins.
- Interactive forms for uploading documents, submitting requisitions, managing inspections, and responding to review comments.
- Comment and response panels supporting inline annotations, accept/reject actions, and justification inputs.
- Real-time status indicators and notifications for tracking task progress and approvals.

4.2 Software Interfaces

- Integration with existing systems for authentication, metadata exchange, and user-role mapping.
- Database connectivity for managing documents, inspection records, comments, version history, and activity logs.
- Export/Print support to generate downloadable test documentation and consolidated reports in standard formats (e.g., PDF, CSV).
- Logging and query interface for admin and QA manager access to filtered activity logs based on user, action, or time.

4.3 Data Exchange

- Data communication between modules for updating requisition statuses, syncing review comments, and rendering dashboard metrics.
- Support for uploading and downloading documents such as plan files, inspection reports, and external test letters.

5. Performance Requirements

This section outlines the performance characteristics expected of the QA Automation Module under specified conditions. It includes both static and dynamic performance requirements to ensure reliability, responsiveness, and scalability of the system.

- The system shall load dashboards and data-driven views (e.g., requisition statuses, review comments) under normal network conditions.
- File uploads (plan documents, reports) up to 20 MB shall be processed and stored.
- The system shall support simultaneous access without performance degradation.
- All user-triggered operations such as form submissions, status updates, and comment actions shall be acknowledged.
- Document version tracking and comment logs retrieval shall complete in under 3 seconds for up to 20 historical versions.
- Activity logs and audit trails shall be retained for a minimum of 12 months without impacting retrieval performance.

6. Design Constraints

This section outlines the limitations and restrictions that must be considered during the design and development of the QA Automation Module. These constraints arise from environmental, technical, operational, and policy-related factors, and must be adhered to by the design and development teams.

6.1 Technology and Platform Constraints

- The system must be developed as a web-based application and be compatible with modern browsers (e.g., Chrome, Edge, Firefox).
- The module must integrate seamlessly with the organization’s existing systems.

6.2 Resource and Infrastructure Constraints

- The application must operate within the existing hosting infrastructure, with limits on memory, storage, and CPU defined by the organization’s IT policies.

- Use of external third-party services is restricted and must be approved based on compliance and data security policies.

6.3 Security and Access Control

- The system must enforce role-based access control (RBAC) for all operations and restrict unauthorized access to sensitive QA data.
- All data transmissions must occur over secure HTTPS protocols.
- User authentication must comply with the organization's standard Single Sign-On (SSO) or identity management system, if applicable.

6.4 Standards and Compliance

- The design must adhere to internal QA process standards and document formatting requirements.
- Auditability must be built into all actions affecting documents, comments, requisitions, and inspections.
- Regulatory requirements regarding quality documentation and traceability must be supported where applicable.

6.5 Reliability and Availability

- The system must be designed for high availability and fault tolerance, with minimal downtime and robust error handling.
- Failures in one module (e.g., document upload) must not impact the overall availability of the system.

7. Non-Functional Attributes

This section outlines the key non-functional attributes required to ensure the QA Automation Module performs reliably, securely, and efficiently in a production environment.

7.1 Security

- Role-based access control (RBAC) must be enforced across all modules.

- All data exchanges must occur over secure protocols (e.g., HTTPS).
- User actions must be authenticated and authorized according to organizational policies.
- Sensitive data such as comments, inspection results, and document revisions must be protected from unauthorized access. Also passwords will be hashed and stored.

7.2 Reliability

- The system must operate with a minimum uptime of 99% to ensure consistent availability.
- Built-in fail-safes should prevent data loss during incomplete transactions or system interruptions.
- All critical operations should be logged and recoverable in the event of unexpected failures.

7.3 Scalability

- The system must support an increasing number of users, document submissions, and concurrent sessions without performance degradation.
- The architecture should allow for horizontal or vertical scaling based on workload demand.

7.4 Data Integrity

- All user actions, including document submissions, comments, and approvals, must be accurately stored and preserved without tampering.
- Version control mechanisms must ensure that historical data remains accessible and unaltered.

7.5 Reusability

- Modular code structure and standardized APIs should support reuse of components across other modules or future enhancements.

7.6 Compatibility

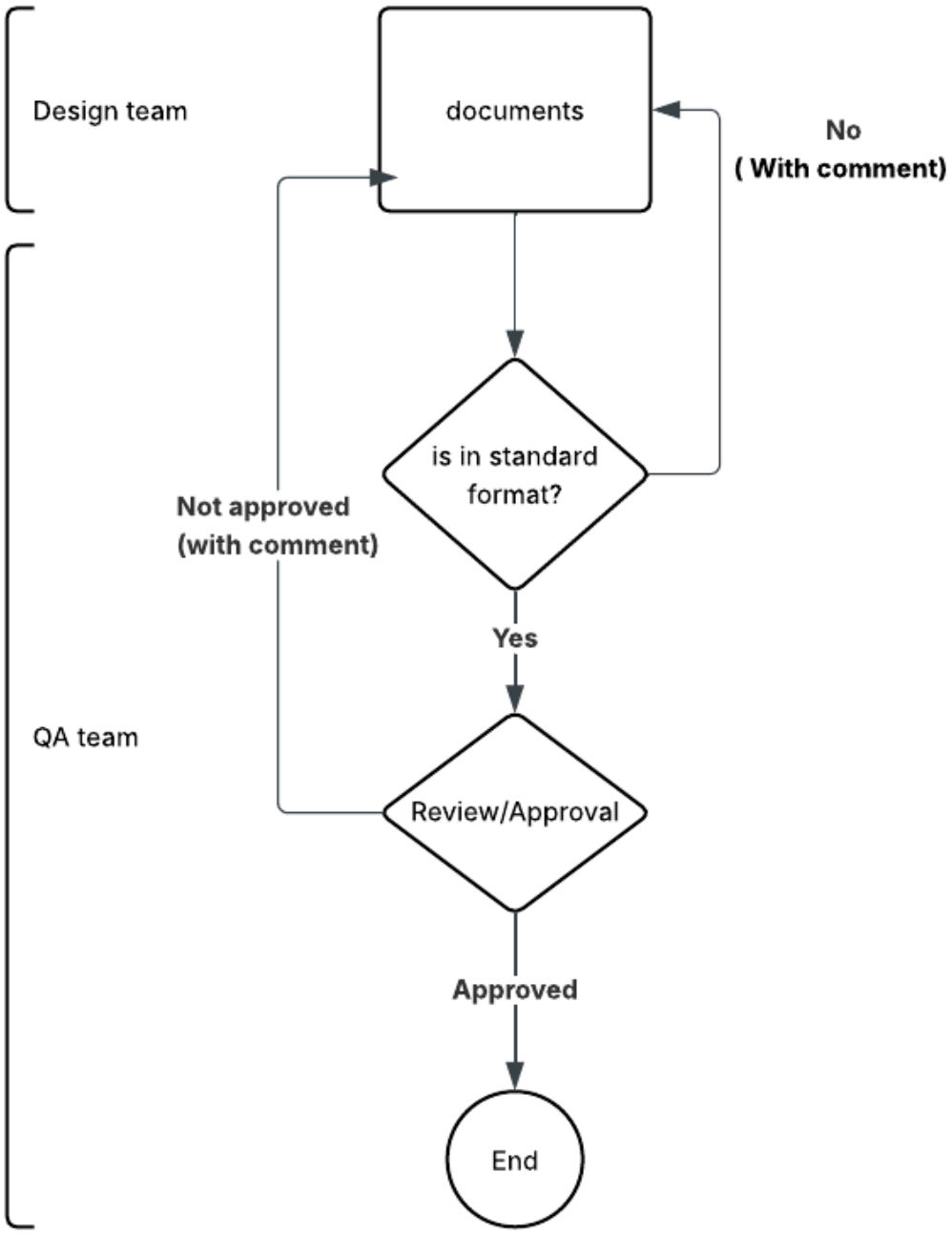
- The system must integrate seamlessly with existing organizational applications for user management, reporting, and data storage.

- Outputs such as reports and exported files must be compatible with common tools (e.g., PDF readers, spreadsheets).

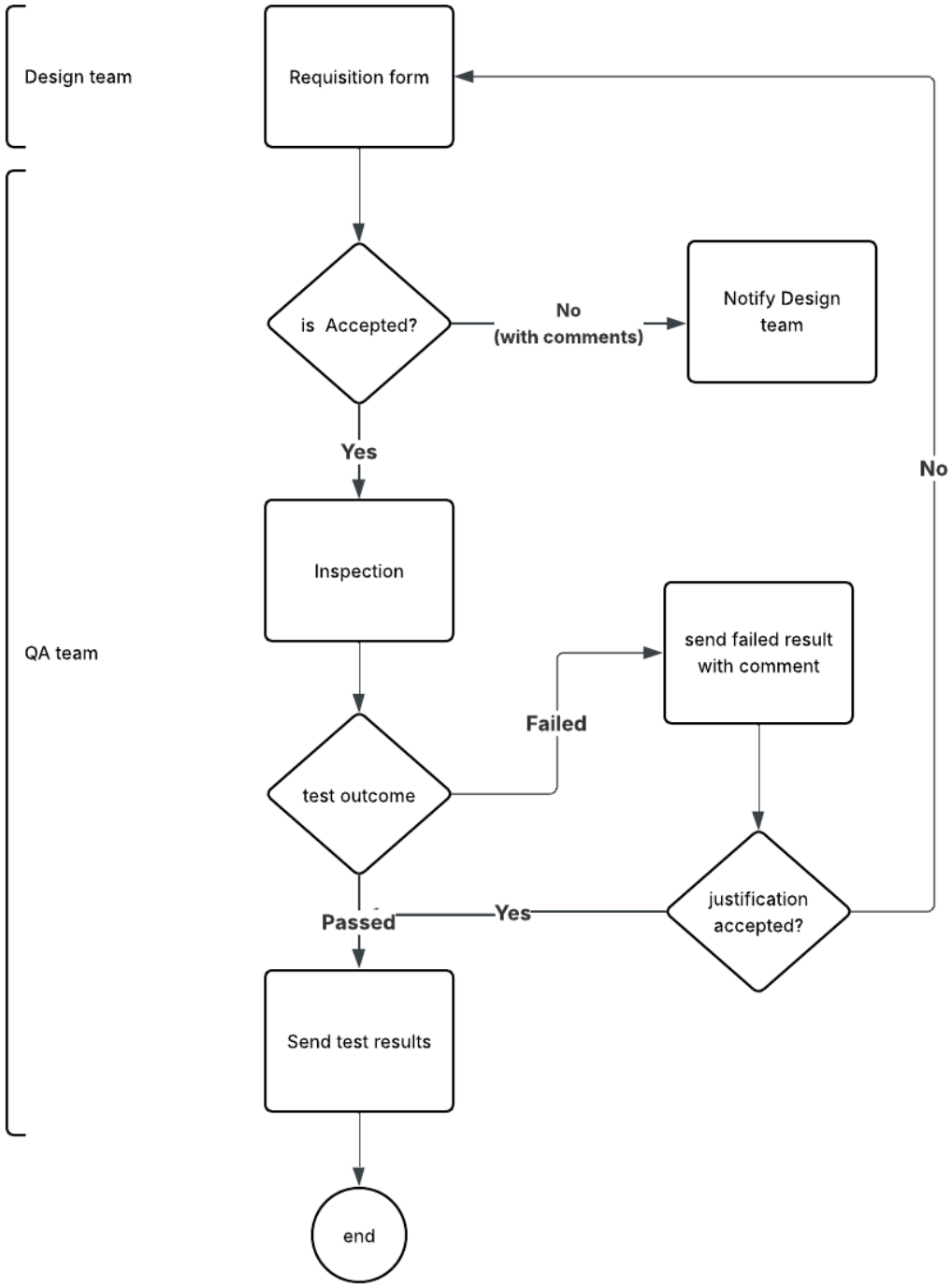
8. Conclusion

This SRS provides a clear and structured foundation for developing the QA Automation Module. It ensures alignment among stakeholders, defines both functional and non-functional requirements, and supports effective development, testing, and project management. By following this specification, the project aims to deliver a reliable, efficient, and secure solution that modernizes QA processes.

System Workflow – Flowchart (Documents)



System Workflow – Flowchart (Requisition and Inspection)



Sequence Diagram – Document Review & Inspection