

IT Policy: Robotic Process Automation (RPA) Governance and Standards Guide

1. Purpose

This document outlines the framework, structure, goals, and standards for Robotic Process Automation (RPA) at our company. It serves as a guide to ensure consistent implementation and compliance across all RPA initiatives.

2. Scope

This guide introduces new users to RPA and provides step-by-step instructions to ensure uniformity in process implementation across the organization.

3. Introduction

What is RPA?

Robotic Process Automation (RPA) uses bots to automate routine tasks such as data entry, invoice processing, and customer service. These bots mimic human interactions with software and streamline workflows, improving efficiency and reducing errors. Unlike traditional automation tools, RPA does not require complex coding, making it cost-effective and user-friendly.

4. Roles and Responsibilities

RPA Developer

- Design, develop, and test automation scripts.
- Troubleshoot issues and optimize performance.

Business Analyst

- Identify processes for automation and document requirements.
- Plan workflows and analyze benefits.

IT Administrator

- Maintain infrastructure and ensure compatibility with IT systems.
- Manage user access, security, and upgrades.

QA Tester

- Develop test plans and validate bot functionality.

Governance Team

- Define governance metrics and ensure adherence to standards.

Bot Owner

- Maintain bots and report issues.

Center of Excellence (CoE)

- Oversee RPA execution and monitor performance in DataBricks and Power BI.

5. Naming Conventions

Best Practices:

1. Ensure names describe the bot's purpose.
2. Use consistent prefixes (e.g., IT, FIN).
3. Include bot type (e.g., attended, unattended).
4. Track all names in an inventory log.

Example Naming Structure: <Department>_<Process>_<BotType>_<ID>

- Example: IT_UserProvisioning_Unattended_137

6. Training New Employees and RPA Developers

1. Learn Tools:
 - Become familiar with UiPath Studio, Orchestrator, and Robot.
 - Create a simple "Hello World" bot for practice.
2. Understand Frameworks:
 - Study PDD and SDD templates.
 - Implement reusable components in sample projects.

7. Framework Development Action Plan

Phases of Framework Development:

1. Foundation (1 Month)
 - Gather requirements and identify stakeholders.
 - Draft governance policies and initial standards.
2. Development (2 Months)
 - Create templates for PDD, SDD, and test scripts.
 - Build reusable workflows and integrate DataBricks.
3. Implementation (1 Month)
 - Pilot the framework and train stakeholders.
 - Document and refine based on feedback.
4. Deployment and Scale-Up
 - Roll out organization-wide and establish feedback loops.

8. Guidelines for Bot Development

1. Plan and Gather Requirements

- Map workflows and define objectives.
 - Set measurable KPIs like time saved or errors reduced.
2. Design Workflows
 - Ensure modularity, error handling, and security.
 - Use UiPath Orchestrator for credential management.
 3. Integrate Systems
 - Build reusable components for DataBricks operations.
 - Secure role-based access to the data lake.
 4. Test and Validate
 - Conduct unit, integration, and user acceptance testing (UAT).
 - Validate with diverse data scenarios, including edge cases.
 5. Deploy and Maintain
 - Deploy bots in stages and monitor performance.
 - Use UiPath Orchestrator and Power BI for advanced analytics.

9. Monitoring and Analytics

Dashboards in Power BI:

Track metrics such as:

- Execution success rate
- Processing time
- Errors by category

Steps to Set Up Monitoring in Power BI:

1. Prepare data in DataBricks with fields like bot name, status, and errors.
2. Use the Azure DataBricks connector for Power BI integration.
3. Design dashboards with filters for date, bot name, and status.

10. Revision History

Revision #	Date	Revision History	Revised By