RAYA INFORMATION TECHNOLOGY

Solution Design Document

Verify Account Positions Process

Revision History

Rev. #	Date	Section/Page#	Revision Summary	Author
1	08/09/2025	All	Created first version	Amr Ayman
2				
3				
4				

Contents

I. PURPOSE

Outlines the major components of the Master Project (the overall output of the development, containing one or multiple projects that together cover the scope of the robotic process automation) taking into account all the business restrictions (scheduling, peaks, future increases in volume etc.).
The focus of the Solution Architect will be on:

- Robustness;
- Scalability;
- Efficiency;
- Replicability;
- Reusability of component

The information here is targeted primarily at the developers that will initially implement the solution and subsequently at the support developers in case of change requests.

The automation solution will utilize the RE framework with Dispatcher Performer Model

II. AUTOMATED PROCESS DETAILS

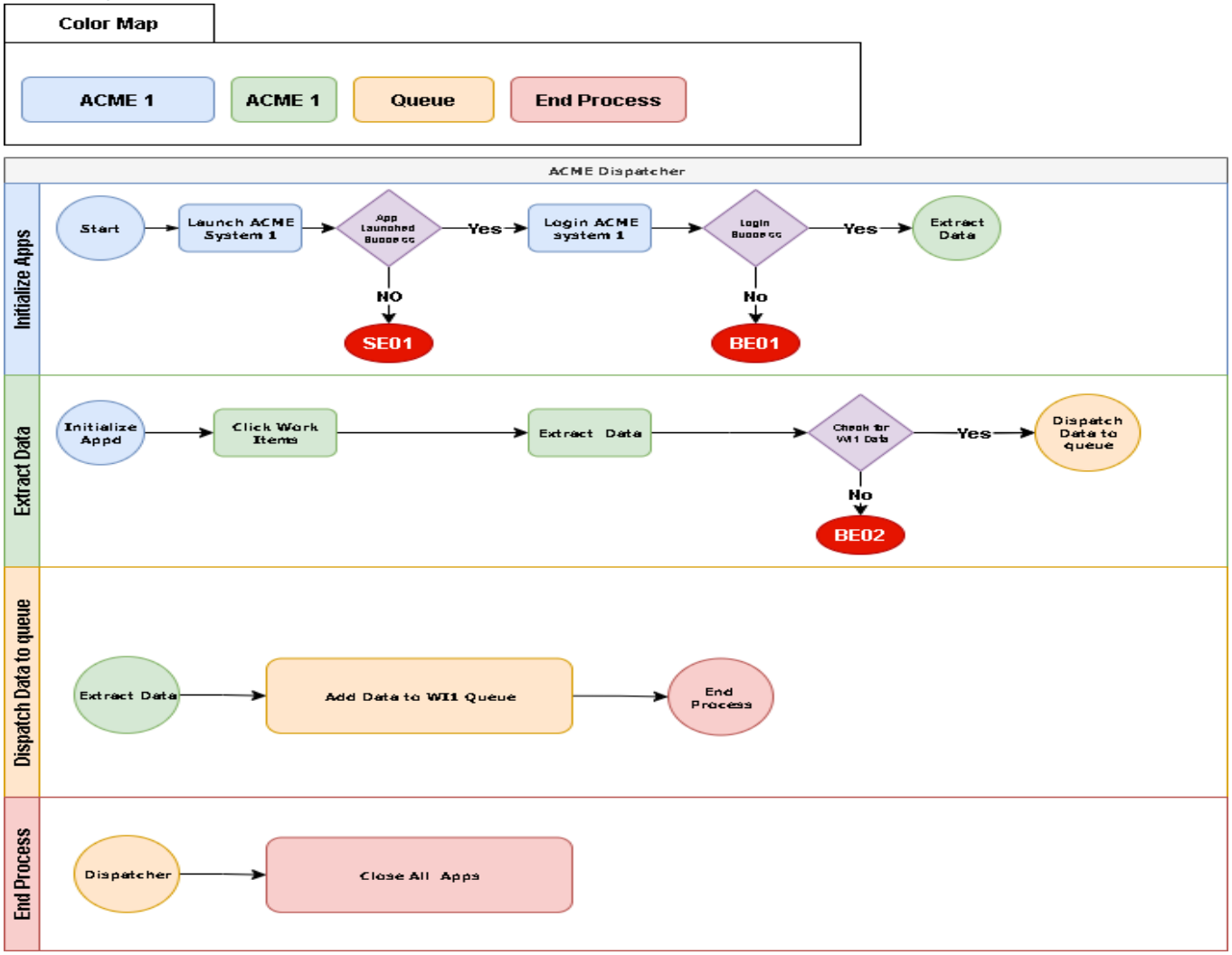
Details filled in need to reflect the actual information for the Master Project released for production. The following table will be populated:

Item	Description
Master Project Name	Verify account Positions
Robot Type	Unattended
Orchestrator used?	Yes
Scalable	Yes
UiPath version used	2025.10

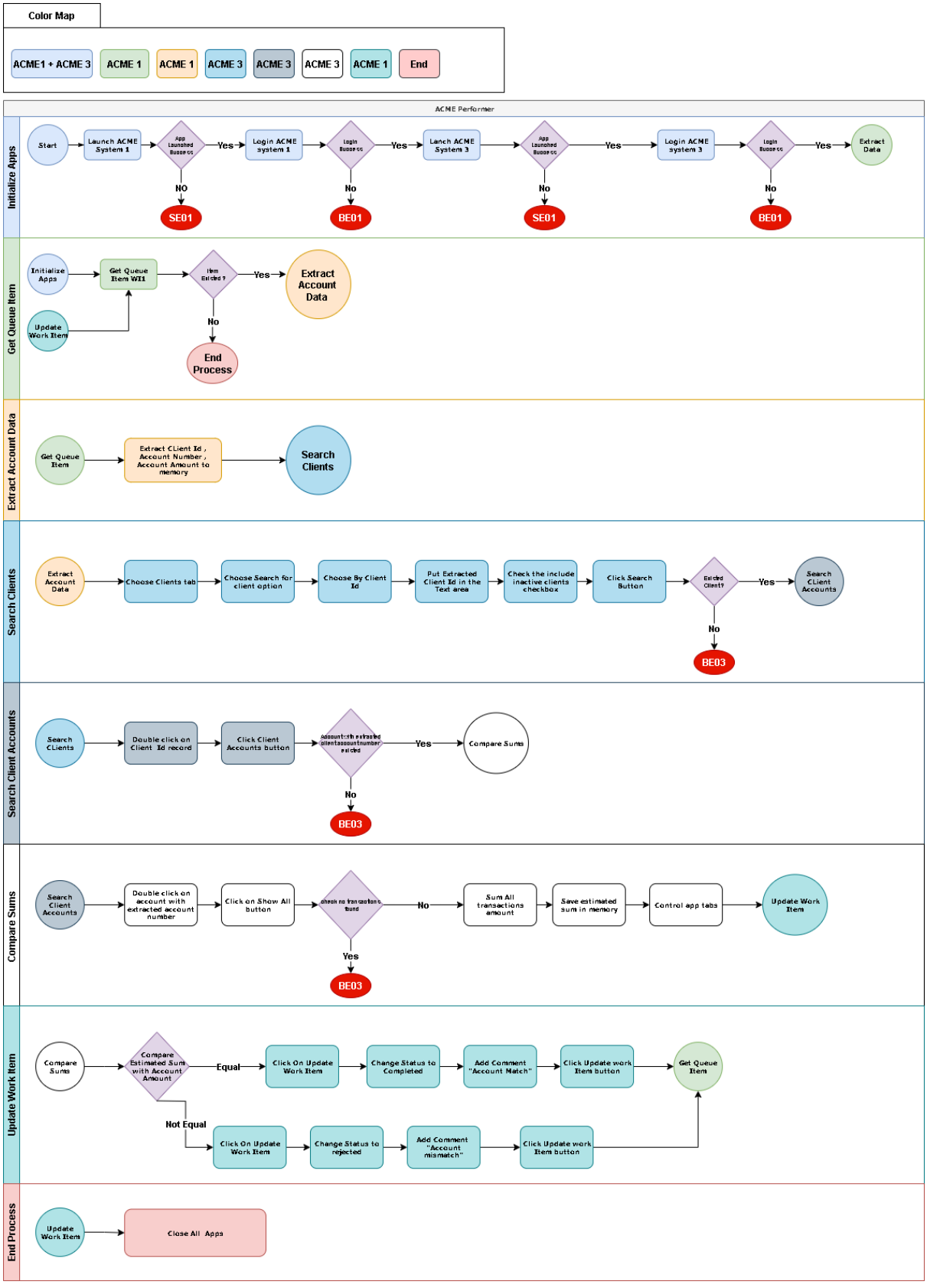
III. RUNTIME GUIDE

1. Architectural structure of the Master Project

Dispatcher



Performer


Color Map
ACME1 + ACME 3
ACME 1
ACME 1
ACME 3
ACME 3
ACME 3
ACME 1
End
ACME Performer
Initialize Apps
Start
Launch ACME System 1
App Launched Successfully
Yes
Login ACME system 1
Login Successful
Yes
Lanch ACME System 3
App Launched Successfully
Yes
Login ACME system 3
Login Successful
Yes
Extract Data
NO
SE01
No
BE01
No
SE01
No
BE01
Get Queue Item
Initialize Apps
Get Queue Item WI1
Item Exicted ?
Yes
Extract Account Data
Update Work Item
No
End Process
Extract Account Data
Get Queue Item
Extract CLient Id , Account Number , Account Amount to memory
Search Clients
Search Clients
Extract Account Data
Choose Clients tab
Choose Search for client option
Choose By Client Id
Put Extracted Client Id in the Text area
Check the include inactive clients checkbox
Click Search Button
Exicted Client?
Yes
Search CLient Accounts
No
BE03
Search Client Accounts
Search Clients
Double click on Client Id record
Click Client Accounts button
Account:sth extracted clientaccountnumber exicted
Yes
Compare Sums
No
BE03
Compare Sums
Search Client Accounts
Double click on account with extracted account number
Click on Show All button
check no transactions found
No
Sum All transactions amount
Save estimated sum in memory
Control app tabs
Update Work Item
Yes
BE03
Update Work Item
Compare Sums
Compare Estimated Sum with Account Amount
Equal
Click On Update Work Item
Change Status to Completed
Add Comment "Account Match"
Click Update work Item button
Get Queue Item
Not Equal
Click On Update Work Item
Change Status to rejected
Add Comment "Account mismatch"
Click Update work Item button
End Process
Update Work Item
Close All Apps

1. Master Project Runtime Details

Outlines the details of the automated process by filling in the table below.

ITEM NAME	DESCRIPTION
Production environment details	Edge Browser Desktop App, ACME System3
Prerequisites to run	Edge with uipath ext, acme system3
Input Data	Account Number, Account Amount
Expected output	Account Position Match ?
How to start the automated process	Daily, Monday to Friday, 9 am – 6 pm
Reporting (queues reporting, Kibana or another platform)	*Email for exceptions*
How is Orchestrator used?	*Orchestrator used for scheduling, assets, queue*
Password policies (mention any specific compliance requests)	*n/a*
Stored credentials (Never use hardcoded credentials in the workflow!)	*Acme credentials , sender mail credentials*
Schedule Details	daily
Multiple Resolutions Supported? (in case of image automation / Citrix and VDI)	n/a
Recommended Resolution	n/a

2. Project(s) workflows

1. Performer

Workflow Name	Description
ACMEDesktop_InitializeApp.xaml	***Initializes the ACME desktop application***
ACMEDesktop_Login.xaml	***Logs into the ACME desktop application***
ACMEDesktop_NavigateToAccountTransactions.xaml	***Navigates to account transactions in ACME desktop***
ACMEDesktop_NavigateToClientAccounts.xaml	***Navigates to client accounts in ACME desktop***
ACMEDesktop_NavigateToClientDetails.xaml	***Navigates to client details in ACME desktop***
ACMEDesktop_NavigateToClientSearch.xaml	***Navigates to client search in ACME desktop***
ACMEDesktop_SumAccountTransactions.xaml	***Sums account transactions in ACME desktop***
ACMEDesktop_ControlTabs.xaml	***Controls tab navigation in ACME desktop***
ACMEWeb_ExtractWorkItemData.xaml	***Extracts work item data from ACME web***
ACMEWeb_InitializeApp.xaml	***Initializes the ACME web application***
ACMEWeb_Login.xaml	***Logs into the ACME web application***
ACMEWeb_NavigateToUpdateWorkItem.xaml	***Navigates to update work item in ACME web***
ACMEWeb_NavigateWorkItems.xaml	***Navigates to work items page in ACME web***
ACMEWeb_UpdateWorkItem.xaml	***Updates work item status in ACME web***
Common_CheckAppCrash.xaml	***Checks if an application has crashed***
Common_CloseApp.xaml	***Closes a specified application***
Control_CheckAccountPosition.xaml	***Checks account position in performer***
Email_SendExceptionsMail.xaml	***Sends exception notification emails***

2. Dispatcher

Workflow Name	Description
ACMEWeb_ExtractWorkItems.xaml	***Extracts work items from ACME web application***
ACMEWeb_InitializeApp.xaml	***Initializes the ACME web application***
ACMEWeb_Login.xaml	***Logs into the ACME web application***
ACMEWeb_NavigateWorkItems.xaml	***Navigates to the work items page in ACME web***
Common_CheckAppCrash.xaml	***Checks if an application has crashed***
Common_CloseApp.xaml	***Closes a specified application***
Email_SendExceptionMail.xaml	***Sends exception notification emails***
Dispatcher_DispatchWorkItems.xaml	***Main dispatcher workflow to upload items to Orchestrator queue***

3. Packages

Include the list of packages and high-level description for each of them, to explain their purpose

Package Name	Description
UiPath.System.Activities	
UiPath.Testing.Activities	
UiPath.UIAutomation.Activities	
UiPath.Excell.Activities	

4. REPORTING

Email for exceptions

5. Testing

Testing is an essential part of any development process. Its main purpose is to ensure the automated process is functioning in the way it is supposed to. By beginning the testing process during development, any issues, bugs, and errors can be identified and potentially corrected at the earliest possible opportunity.

Each level of testing is responsible for different parts of the automated solution, reassuring high quality of each of the elements delivered. Each of the stages must be completed before the solution is released to the production environment.

The stages of testing are:

Unit Testing - Focuses on the individual components and workflows created specifically for the process in scope, assessing whether they function as according to the original design.

Responsibility: Development Team

System Integration Testing – Ensuring all required functionalities, both business and technical, are implemented, as well as ensure unit testing has been completed successfully, development Best Practices have been applied and code review has been completed by a Solution Architect. There will usually be a round of fixes to be implemented before the automation can proceed further.

Responsibility: Dedicated tester / Solution Architect

User Acceptance Testing – Process team take responsibility of this phase of testing to ensure they are happy with the results of the automated business process. Sign-off of this phase is business led and mandatory in order to proceed further. Generally, there will be iterative fixes to be implemented that have been discovered during the user testing of the automation.

Responsibility: Dedicated tester & Business Unit

Pre-Production Testing – Validates the readiness of the product prior to Production deployment.

Responsibility: Dedicated tester, Development Team & Solution Architect

Dry-Run –Monitor the initial cases run by the process to account for any environmental differences between Pre-Production and Production.

Responsibility: RPA Business Analyst & Business Unit

6. Security

Security is a vital aspect of digital processes as it provides an extra layer of protection from the public. There are different ways in which security is implemented when automating an existing process, some of which are discussed in this section.

Sensitive data such as passwords and client/customer information (like information stored in databases) must never be unnecessarily stored outside of its origin application when handled by the RPA solution.

Similarly, user and application credentials used by the RPA solution must also never be unnecessarily stored or shared with any unauthorized persons. In addition to that, if the robot is dealing with password reset policies that are internal to the client, those policies must be followed to ensure security is maintained.

API keys are often used within RPA solutions to allow access to certain applications, and these are ideally stored in Orchestrator credential assets and only accessed when needed. As with credentials, they should never be shared or stored elsewhere.