



Quick_Assist-ADO

Readme

Version 1.0.0

31/12/2025

Table of Contents

1. Introduction.....	3
1.1 Overview.....	3
1.2 Use Cases	3
2. Requirements & Prerequisites.....	4
2.1 System Requirements.....	4
2.2 Prerequisites	5
3. Getting Started	6
3.1 Quick Start	6
3.1.1 Setup	6
3.1.2 Configuration	6
4. Support & FAQs	11
4.1 Support	11
4.2 FAQs.....	11
Appendix A: Record of Changes	12
Appendix B: References.....	13



1. Introduction

This document contains all essential information for the user to make full use of this *Quick Assist -ADO*. It includes a description of the functions and capabilities and step-by-step procedures for setup & configuration of the *Quick Assist-ADO*.

1.1 Overview

Creating an IT Ticket management service which includes more protected and secure method than the existing applications like Jira, because here data is 100% secured and most organizations can easily shift to this tool because of existing infrastructure of Azure Devops. More user friendly and manageable tool than the traditional ones.

1.2 Use cases

The key use cases include:

- 1) Ensuring integrity and security of allocation of resources because Azure Devops only allows people in your organization to allocate the issue, it does not allow mails outside your organization.
- 2) Faster and in-built notification system on outlook, thus handling a lot of manual work load.
- 3) Efficient utilization of time of senior managers and technical heads.
- 4) Efficient utilization for organizations having existing infrastructure of Azure and time management for senior staff for allocation of resources.

2. Requirements & Prerequisites

2.1 System Requirements

[Enterprise \(Cloud deployed\) and Community Edition device requirements.](#)

Device	Processor (CPU)	RAM	Storage(free disk space)	Network
Single user machine	x86-64 CPU higher than or equal to 4 vCPU: Intel Core i3 (2.6 GHz) or higher, Intel Xeon, or AMD Ryzen/EPYC equivalent. Size to meet or exceed i3-class performance for your workload.	4 GB (Minimum) 8 GB (Recommended)	32 GB Add 100 through 150 KB per Automation Anywhere script Add 40 to 50 GB per long-term project (logs, screenshots, and so on).	5Mbps (Minimum) 20Mbps or higher (Recommended)
Multi-user machine	x86-64 CPU with higher than or equal to 4 vCPU baseline + 2 vCPU per additional concurrent user: Intel® Core, Intel Xeon, or AMD Ryzen/EPYC equivalent. Size based on concurrent bot activity and application mix..	4 GB base plus 4 GB per additional user	32 GB Add 100 through 150 KB per Automation Anywhere script Add 40 to 50 GB per long-term project (logs, screenshots, and so on).	5Mbps (Minimum) 20Mbps or higher (Recommended)

2.2 Prerequisites

Below are the required software version for performing the actions for Quick_Assists-ADO :

- 1) Azure Devops: Version Dev20.M265.1 (AzureDevOps_M265_20251210.1)
 - You also need to setup and create organization and project name on Azure Devops.
 - After this, you need to create Epics in Azure Devops Boards and write client's name in place of the Epics.
 - Also, add the clients name in Ticket Review Form -> Add Dropdown content for getting the client names.
- 2) Latest version of Automation Anywhere Bot Agent.
- 3) Outlook: Web Version.



3. Getting Started

3.1 Quick Start

3.1.1 Setup

Follow the below steps for creating a **personal access token** in Azure Devops:

- 1) Sign in to your Azure DevOps organization (https://dev.azure.com/{Your_Organization}).
- 2) Click on your user icon (profile picture) in the top-right corner, and then select User settings.
- 3) In the left navigation menu, under the Security section, select Personal access tokens.
- 4) Select + New Token.
- 5) In the creation window, fill in the following details:
 - a. Name: Enter a descriptive name for the token to help you remember its purpose.
 - b. Organization: Select the specific organization where you will use the token, or choose "All accessible organizations" if needed (subject to admin policies).
 - c. Expiration: Set an expiration date for security purposes. Microsoft recommends keeping lifespans short (e.g., a few days or weeks).
 - d. Scopes: Select only the minimum required permissions (scopes) for the task the token will perform (grant the principle of least privilege). Avoid using "Full access" unless absolutely necessary.
- 6) Click Create.
- 7) Copy the generated PAT immediately. For your security, it will not be displayed again once you close the window. Store it in a secure location, like Azure Key Vault.

3.1.2 Configuration and Use

For Packages:

Details of Packages used are provided in below table:

Package Name	Version	Vendor	Bot Agent Version	Control Room Version



Dictionary	3.15	Automation Anywhere	20.11 or above	8750 or above
Error handler	2.13.0-20241115-120032	Automation Anywhere	20.11 or above	8750 or above
If	3.8.0-20241115-115929	Automation Anywhere	20.11 or above	8750 or above
Json	1.9.1	Automation Anywhere	21.98 or above	10520 or above
Legacy automation	6.6.1	Automation Anywhere	21.200 or above	13279 or above
Message box	3.8.1	Automation Anywhere	20.11 or above	8750 or above
Number	3.10.0	Automation Anywhere	20.18 or above	9130 or above
Python script	2.20.0	Automation Anywhere	20.11 or above	8750 or above
REST Web Services	3.24.1	Automation Anywhere	20.11 or above	8750 or above
String	5.11.3	Automation Anywhere	20.18 or above	9130 or above

- *For Credential Vault –*

Need to store the Azure Devops PAT(Personal Access Token) for security reason.

<i>Locker Name</i>	<i>Credentials Name</i>	<i>Attribute Name</i>	<i>Value</i>
<i>Locker_ADO</i>	<i>Creds_ADO</i>	<i>ADO_PAT</i>	<i><Value of personal access token></i>

- For configuring the bot –

INPUT VARIABLES				
Variable Name	Type	Mandatory	Purpose	Example Input
<i>iStrApproval</i>	<i>String</i>	<i>Yes</i>	<i>Approval for performing the next action.</i>	<i>The input data contains either "Yes" or "No" stating the approval for performing further action.</i>
<i>iStrAssignedTo</i>	<i>String</i>	<i>Yes</i>	<i>MailID of the person assigned the ticket.</i>	<i>The input data contains the mailid of person in your organization for assigning and mailing him of assigned tasks on Azure Devops.</i>
<i>iStrClientName</i>	<i>String</i>	<i>Yes</i>	<i>Name of the Client</i>	<i>The input data contains the client's name.</i>
<i>iStrDescription</i>	<i>String</i>	<i>Yes</i>	<i>Description of issue occurring for that particular client</i>	<i>The input data contains the description of the issue occurring in the system or software of that specific client.</i>
<i>iStrPriority</i>	<i>String</i>	<i>Yes</i>	<i>Priority of the issue</i>	<i>The input data contains the priority of the issue which can be "Critical", "Normal", "Warning".</i>

VARIABLES				
Variable Name	Type	Mandatory	Purpose	Example
pDictResponse	Dictionary Subtype-String	Yes	Dictionary for storing the response from REST Web Service	<i>It contains the response from api call, will have various status code and their other response parameters.</i>
pDictParsed	Dictionary	Yes	Dictionary for storing the Work ID and other info. of Azure Devops Boards.	<i>It contains the Work ID and other important information from Azure Devops.</i>
pDictResponse2	Dictionary Subtype-String	Yes	Dictionary for storing the response from REST Web Service	<i>It contains the response from api call which links the issue with specific client, will have various status code and their other response parameters.</i>
pStrCred	String	Yes	Credentials of Azure PAT	<i>It contains the personal access token of Azure Devops which is encoded in base64 format.</i>
pStrErrorMessage	String	Yes	Error message occurred in bot execution	<i>It contains the information of specific error occurred while execution of bot.</i>
pNumErrorLineNumber	Number	Yes	Line number of Error	<i>It contains the line number of error occurred while execution of bot.</i>



			<i>occurred</i>	
<i>pStrID</i>	<i>String</i>	Yes	<i>Work ID of Azure Devops Boards</i>	<i>It stores the Work ID of Azure Devops for that particular ticket extracted from the response of api call while creation of ticket.</i>
<i>pStrURI</i>	<i>String</i>	Yes	<i>URI of your Azure Devops - organization</i>	<i>https://dev.azure.com/{org_name}/{project_name}/_apis/wit/workitems/\$Issue?api-version=7.1-preview.3</i>
<i>pStrJsonBody</i>	<i>String</i>	Yes	<i>Json body of response from dictionary</i>	<i>It stores the response from dictionary into json format and stores it in string format for better access and usability.</i>

The customers must be having a lot of clients with different issues occurring in their software or system, in order to manage these and also having Azure Devops subscription of your organization, you can easily use it as Ticket management tool and handle them while maintaining the notification system on outlook for automating the notification via mail and informing the allocated issue with description on Azure Devops, later they can manually change the status to “Done” on Azure Devops, thus provides efficient and systematic approach of managing this.

Inside Azure Devops, the customer needs to configure below details:

- 1) **Epics** : Here, epics represents the client names which your organization maybe having or you can also specify different processes.
- 2) **Issues**: This includes the ticket and issue description which will be allocated to specific person via his mailid with status as Critical, Normal, Warning, etc. .



4. Support & FAQs

4.1 Support

Free bots are not officially supported. You can get access to Community Support through the following channels:

- You can get access to community support, connecting with other Automation Anywhere customers and developers on the Community in the [Developers Forum](#).
- Automation Anywhere also provides a [Product Documentation portal](#) which can be accessed for more information about our products and guidance on [Automation 360](#).
- You can contact me on my personal mail ID – mayureshshinge01@gmail.com .

4.2 FAQs

Instructions: Add any Frequently Asked Questions related to your automation that you think would be helpful to the customer

For questions relating to the Control Room: See the [Automation 360 FAQs](#).

Appendix A: Record of Changes

Instructions: Provide information on the version number, the date of the version, the author/owner of the version, and a brief description of the reason for creating the revised version.

No.	Version Number	Date of Change	Author	Notes
1	1.0.0	31-12-2025	Mayuresh Shinge	-

Appendix B: References

No.	Topic	Reference Link
2	Guidance: Building basic Automation 360 bots	Click here
3	Guidance: Building Automation 360 packages	Click here
4	Community Developers Forum	Click here
5	Automation Anywhere University	Click here