



[Add Comment to Jira Ticket](#)

Readme

Version 2.0

01/07/2020

Table of Contents

1. Introduction.....	3
1.1 Overview.....	3
1.2 Common Use cases.....	5
2. Requirements & Prerequisites.....	6
2.1 System Requirements.....	6
2.2 Prerequisites.....	6
2.3 Security Measures	6
2.4 Disclaimers	6
3. Getting Started.....	7
3.1 Skill Matrix	8
3.2 Installation Hierarchy	8
3.3 Quick Start	8
3.3.1 Setup	8
3.3.2 Configuration	8
4. Reports	12
5. Logs	13
6. Troubleshooting & Support	15
6.1 Support	15
6.2 FAQs.....	15
Appendix A: Record of Changes	16
Appendix B: Acronyms.....	17
Appendix C: References.....	18

1. Introduction

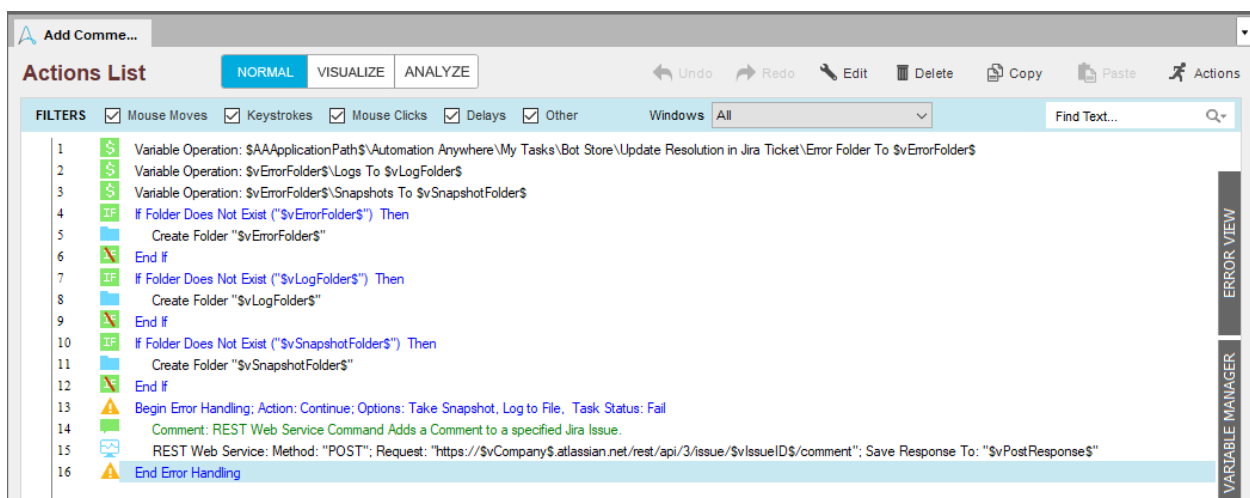
This document contains all essential information for the user to make full use of the Add Comment to Jira Ticket Bot. This manual includes a description of the functions and capabilities and step-by-step procedures for setup & configuration of the Bot.

1.1 Overview

This bot issues an API call to add a comment to a specified Jira issue. Detailed steps are as follows:

- Creates Error folder, Log folder, and Snapshot folder if they do not currently exist
- Issues POST API call to add comment to a Jira issue

Visual Walkthrough:



Input Example:

The screenshot shows the 'REST Web Service' application window. The title bar says 'REST Web Service'. The main header is 'REST Web Service' with a help icon. The 'URI' field contains 'https://\$vCompany\$.atlassian.net/rest/api/3/issue/\$vissueID\$/comment'. Below it is an example: 'e.g. http://www.domainname.com/resource/path/search?param1=value& param2=value2'. The 'Method' dropdown is set to 'POST'. The 'PARAMETER' tab is selected, showing a JSON body in the main area. The left sidebar has 'URL', 'Post', and 'Authorization' sections. The bottom has a 'Send Request' button and a status bar with 'Press F2 to insert variable.', 'Save', and 'Cancel' buttons.

REST Web Service

URI :

e.g. http://www.domainname.com/resource/path/search?param1=value& param2=value2

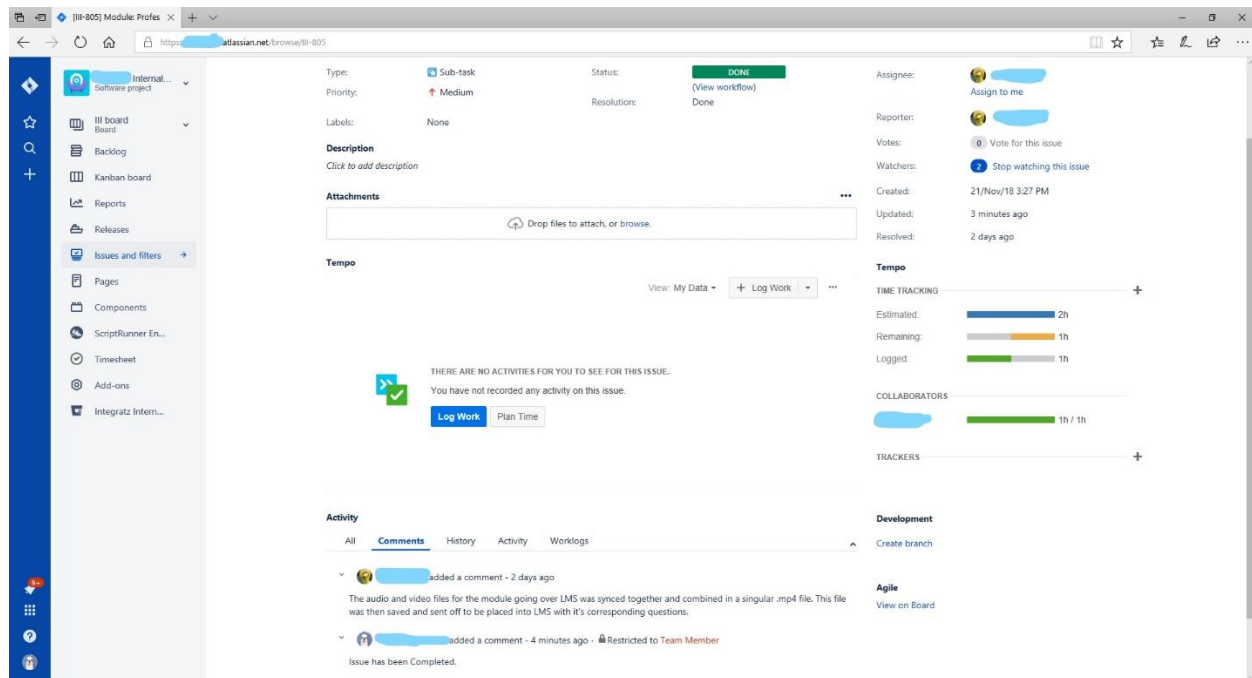
Method : POST

REQUEST	PARAMETER	RESPONSE
URL	<pre>{ "body": { "type": "doc", "version": 1, "content": [{ "type": "paragraph", "content": [{ "type": "text", "text": "\$vComment\$" }] }] }, "visibility": { "type": "role", "value": "Team Member" } }</pre>	
Post		
Authorization		

Send Request

Press F2 to insert variable. Save Cancel

Output Example:



1.2 Common Use cases

- Updating a ticket in Jira through a comment.
- Informing assignees and watchers of ticket status via comment.

2. Requirements & Prerequisites

2.1 System Requirements

For the PC or server where the bot needs to run:

- RAM: 8GB or higher
- PROCESSOR: Intel Core i5 or higher and equivalent for any other OS
- Hard Disk: Up to 2GB of overall free space in the AA Client installation drive.

2.2 Prerequisites

- Automation Anywhere v10.X.X - v11.X
- Atlassian Account
- Jira API Basic Authentication Token

2.3 Security Measures

There are some security best practice recommendations that you may follow with your bot.

- It is recommended to treat API tokens as securely as any other password.
- For security reasons it isn't possible to view the token after closing the creation dialog; if necessary, create a new token.

2.4 Disclaimers

- POST response 201:

Returned if any of the issue or sub-task creation requests were successful. A request may be unsuccessful when it:

- is missing required fields.
- contains invalid field values.
- contains fields that cannot be set for the issue type.
- is by a user who does not have the necessary permissions.
- is invalid for any other reason.

3. Getting Started

3.1 Skill Matrix

Skill	Task Files	MetaBot Files
Issue POST API Call to Add Comment to Jira Ticket	Add Comment in Jira Ticket.atmx	No MetaBots

3.2 Installation Hierarchy

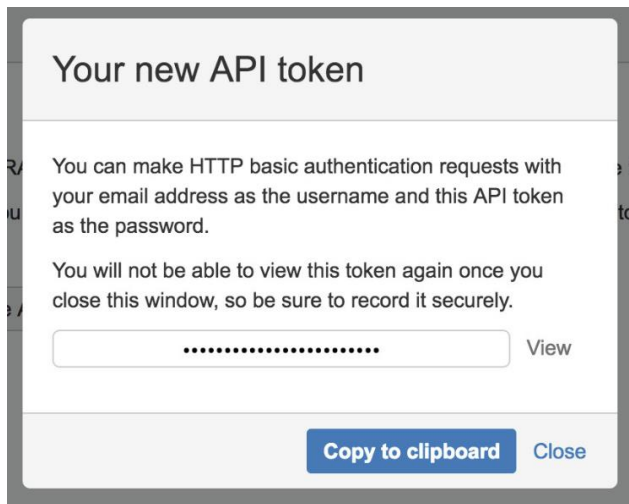
Folder Structure	Description
<AA Application Path> 1. My Tasks 1.1 Bot Store 1.1.1 Add Comment to Jira Ticket (Folder) ➤ Error Folder ○ Logs ▪ Error Logs Month-Day-Year.txt ○ Snapshots ▪ Error Snap Month-Day-Year.txt ➤ My Tasks ○ Add Comment to Jira Ticket.atmx ➤ Input Folder ○ N/A ➤ My Metabots ○ N/A	<AA Application Path> is the location where AA files are stored on your machine. 1. My Tasks My Task Folder is the default directory where Bot Files are saved. 1.1 Bot Store Bot Store Folder contains the Bot Name Folder which the installer creates while installation of the Bot. 1.1.1 Notify with Slack on Record Update This folder gets created by the installer and contains files and folders needed for the bot execution. • Error folder is where logs and snapshots of screens will be placed if something goes wrong with the bot during execution • Input Folder is where the input files that the bot needs for execution of the use case is saved • My Tasks folder contains all the Developed Platform Source Code 2. My MetaBots folder contains the Developed Metabots needed for the bot execution.

3.3 Quick Start

3.3.1 Setup

Create an API token from your Atlassian account:

1. Log in to <https://id.atlassian.com/manage/api-tokens>.
2. Click **Create API token**.
3. From the dialog that appears, enter a memorable and concise **Label** for your token and click **Create**.
4. Click **Copy to clipboard**, then paste the token to your script, or elsewhere to save:



Note:

- For security reasons it isn't possible to view the token after closing the creation dialog; if necessary, create a new token.
- You should store the token securely, just as for any password.

3.3.2 Configuration

Step 1: Configuring Credential in Control Room's Credential Vault

Passwords and other sensitive information such as user credentials, account number, and social security numbers included in automation tasks, should be stored as credentials centrally in the Credential Vault.

These sensitive data items become secured (stored in CR) and can't be accessed locally which is used by the bots when it executes in the machine.

Below are the step-by-step instructions for what needs to be done to set up the Token for Jira in the Credential Vault.

- 'Locker Consumer Role' needs to be assigned to the AA user (bot runner) which ensures the user have full access to the locker.

- Locker needs to be created with name as "Locker_Jira"
- Credential needs to be created with name as "Credential_Jira", where the Token needs to be saved.

Locker Name	Credentials Name	Attribute Name	Value
Locker_Jira	Credential_Jira	Token	Generated Token

* REST Web Service Command

- This is the command which contains the Jira Add Comment POST API (<https://developer.atlassian.com/cloud/jira/platform/rest/v3/#api-api-3-issue-post>) and thus will need to be configured to fit the needs of your specific Jira comment.

INPUT VARIABLES				
Variable Name	Type	Mandatory	Purpose	Example Input
URI (Within REST Web Service Command)	String	Input	- Connect to Jira to execute POST API call.	https://[COMPANY].atlassian.net/rest/api/3/issue/[ISSUEID]/comment
vCompany	String	Input	Connects to correct Jira URL	https://\$vCompany\$.atlassian.net/rest/api/3/issue/\$vIssueID\$/comment
vBasicJiraAuth	String	Input	Variable in which the API token is stored	N/A
vComment	String	Input	Adds the text you want to put in the comment.	"This issue is currently in progress"
vIssueID	String	Input	Includes the Jira Issue ID in which	ABC-12345

			you want to add the comment .	
--	--	--	-------------------------------	--

OUTPUT VARIABLES				
Variable Name	Type	Mandatory	Purpose	Example Output
vPostResponse	String	No	Variable in which the API Call POST response is stored	HTTP/ 201 Created

Step 2: Configuring REST WEB SERVICE Command

1. In the URI, Change "Your-Company.Atlassian.net" to your specific Jira URL.
2. Requires API Token from Jira; <https://id.atlassian.com/manage/api-tokens>)
 - a. Create the following string: "YourReporterEmail.gmail.com:YourReporterAPIToken"
 - b. Encode the string to base64 (<https://www.base64encode.org/>)
 - c. Paste encoded string into vBasicJiraAuth in Variable Manager or Credential Vault

Expected Result:

REST Web Service

URI :

e.g. <http://www.domainname.com/resource/path/search?param1=value& param2=value2>

Method :

REQUEST PARAMETER RESPONSE

Key	Value
Content-Type	application/json
Authorization	Basic \$vBasicJiraAuth\$
Accept	application/json

Add Delete

REST Web Service

REST Web Service

URI :
e.g. http://www.domainname.com/resource/path/search?param1=value& param2=value2

Method :

REQUEST

PARAMETER

RESPONSE

URL

Post

Authorization

```
{
  "body": {
    "type": "doc",
    "version": 1,
    "content": [
      {
        "type": "paragraph",
        "content": [
          {
            "type": "text",
            "text": "$vComment$"
          }
        ]
      }
    ]
  },
  "visibility": {
    "type": "role",
    "value": "Team Member"
  }
}
```

Send Request

Press F2 to insert variable.

Save

Cancel

4. Reports

There are no Bot Insight Reports generated for this Bot.

5. Logs

In case of Errors, Error Logs and Snapshots are generated within the Error Folder (highlighted below in yellow)

- My Tasks
 - Bot Store
 - Add Comment to Jira Ticket (Folder)
 - Error Folder
 - Logs
 - Error Logs Month-Day-Year.txt
 - Snapshots
 - Error Snap Month-Day-Year.txt

Error Logs will contain the below information:

- Task Name
- Error Line Number
- Error Description
- Timestamp

Example of a Log File:

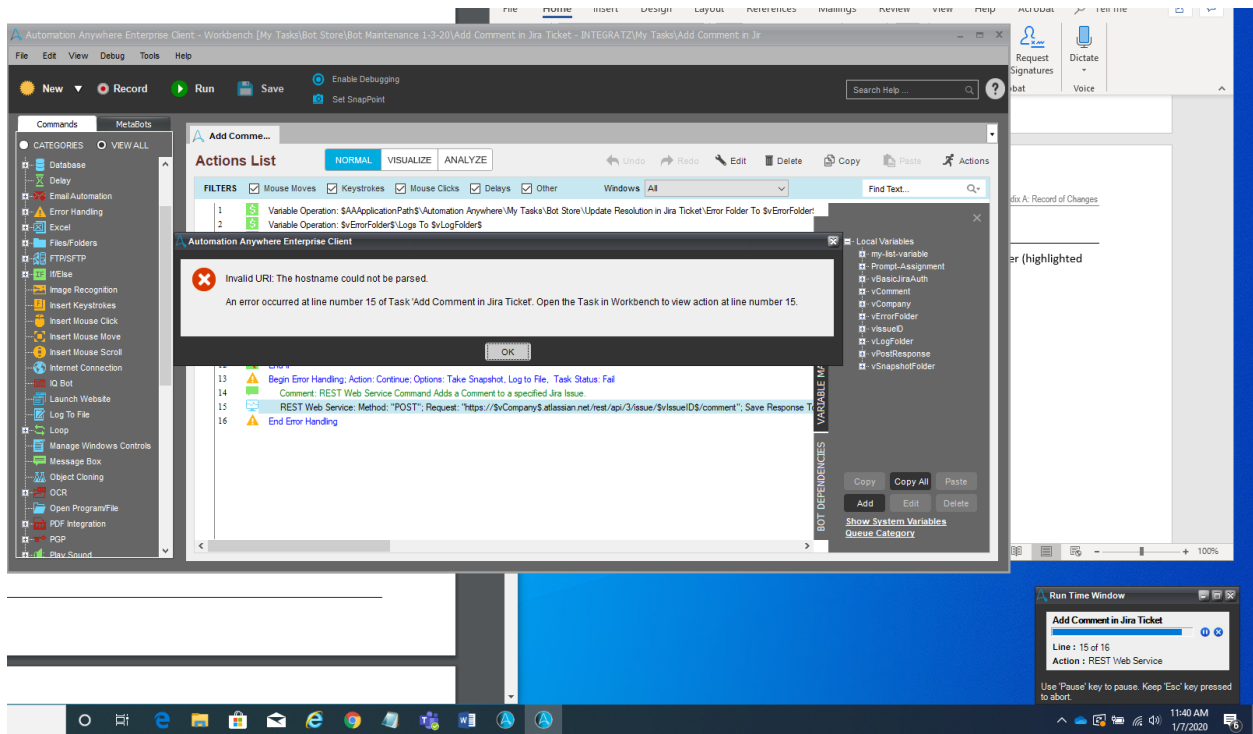
Name	Date modified	Type	Size
 Error Logs 1-7-2020	1/7/2020 11:40 AM	Text Document	1 KB

 Error Logs 1-7-2020 - Notepad

File Edit Format View Help

Error Line Number: "15" Error Description: Invalid URI: The hostname could not be parsed.

Example of Screenshot Generated Error File:



6. Troubleshooting & Support

6.1 Support

Company Support:



Integrätz

Email: automate.today@integratz.com

Questions on Bot Functionality or Feature can also be posted to our Community site Apeople

Automation Anywhere provides a Product Documentation portal which can be accessed for more information about AA's products and guidance on building bots and Digital Workers.

The "Build" section of the portal includes these sections:

- Getting Started - information on building bots recommended practices (including use of the Credential Vault)
- Build Digital Workers - high level architecture

6.2 FAQs

Q: The 'Create Locker' button not visible in the Credential Vault Page. What do I do?

A: Please check if the role 'Locker Consumer' is assigned to the user and you are using an Enterprise Client version.

Appendix A: Record of Changes

No.	Version Number	Date of Change	Author	Notes
1	2.0	1/7/2020	Michaiah Gartner	Version 2 Release

Appendix B: Acronyms

No.	Acronym	Description
1	AA	Automation Anywhere
2	API	Application Programming Interface
3	CR	Credential Vault
4	OS	Operating System
5	PC	Personal Computer
6	RAM	Random Access Memory

Appendix C: References

No.	Topic	Reference Link
1	Encoding an API string	<u>https://www.base64encode.org/</u>
2	Jira Comment POST API	<u>https://developer.atlassian.com/cloud/jira/platform/rest/v3/#api-api-3-issue-post</u>
3	Create Jira API Token	<u>https://id.atlassian.com/manage/api-tokens</u>