

AppPerfect

BitBucket Bot - AppPerfect Corporation-ReadMe

Version 1.0

April 06, 2020

Table of Contents

1. Introduction	3
Overview.....	3
Common Use cases	5
2. Requirements & Prerequisites	6
System Requirements	6
Prerequisites.....	6
Security Measures.....	6
3. Getting Started	7
Skill Matrix	7
Installation Hierarchy	7
Quick Start.....	9
3.1.1 Setup	9
3.1.2 Inputting your Access Code.....	10
3.1.3 Configuration	10
4. Logs	34
5. Troubleshooting & Support	36
Support.....	36
FAQs	36
Appendix A: Record of Changes	37
Appendix B: Acronyms	38
Appendix C: References	39

1. Introduction

This document contains all essential information for the user to make full use of the Bot or Digital worker. This manual includes a description of the functions, capabilities and step-by-step procedures for setup & configuration of the Bot.

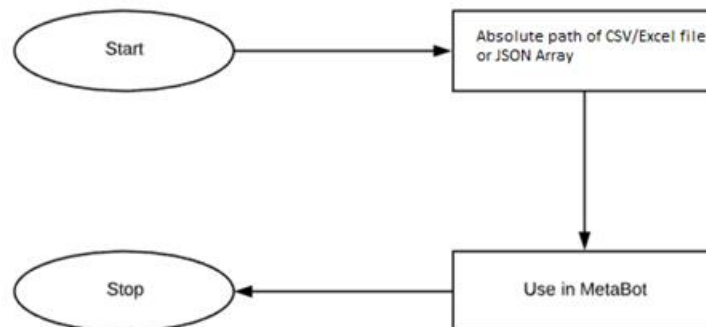
Overview

This bot performs various BitBucket operations like Create and Get operations for various entities like Forks, Issues, Pull Requests, Repositories and Snippets, Delete Repositories, Search Code, Get Milestones in Repository, Update Repositories based on input. In Output, JSON is returned in string format and can be stored in [CSV](#) File.

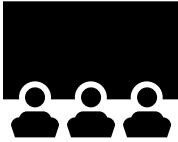
Detailed steps are as follows:

- 1) Reads data from an Input file or JSON Array.
For E.g. CreateFork.csv/ CreateFork.xls
- 2) Generate the result in an Output [CSV](#) file.
For E.g. CreateForkOutput.csv, GetPublicRepostoriesOutput.csv, etc.

- Visual Flow –



Common Use cases



A common use case, as shown above, is to create a fork in BitBucket.

The fields in the [CSV](#) template can be changed to any data value that needs to be added to the list via MetaBot. Examples of other use cases include:

- Bulk Create Fork
- Bulk Create Issue
- Bulk Create Pull Request
- Bulk Create Repository
- Bulk Create Snippet
- Bulk Delete Repository
- Bulk Update Repository
- Create Fork
- Create Issue
- Create Pull Request
- Create Repository
- Create Snippet
- Delete Repository
- Get Commit
- Get Deployments
- Get Environments
- Get Forks
- Get Hook Events
- Get Issues
- Get Pipelines in Repository
- Get Public Repositories
- Get Pull Request
- Get Pull Requests of Repository
- Get Snippets
- Get User Repositories
- Get User Repository Permissions
- Get User Teams Permissions
- List Milestones of Repository
- Search Code
- Update Repository

2. Requirements & Prerequisites

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.

Reference below for [Enterprise Client & Control Room system requirements](#).

Prerequisites



- ✓ **Software's needed -**
- ✓ [AA](#) Enterprise Client 11.3.3
- ✓ [AA](#) Enterprise Control Room 11.3.3
- **Accounts/License needed –**
- ✓ BitBucket Account
- ✓ [AA](#) Enterprise License

Security Measures

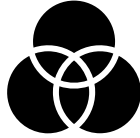


There are some security best practice recommendations that you may follow with your bot.

- It is not recommended to provide admin access to the Windows User Account executing the Bots, to avoid unintended data changes with the [CSV](#) Template.

3. Getting Started

Skill Matrix



The functionality of the Bot has been divided into a set of skills.

Below is an overview of how the task bots and metabots map to these skills:

Skill	Task Files	MetaBot Files
Create, Update, Delete Repositories	BitBucketOperations.atmx	BitBucketMetaBot.mbot

Installation Hierarchy



Once the bot is downloaded and installed, the installer creates the files in the folder structure as shown below.

Folder Structure on your Machine

Folder Structure	Description
<p><AA Application Path></p> <ul style="list-style-type: none"> ❖ My Tasks <ul style="list-style-type: none"> ➤ Bot Store <ul style="list-style-type: none"> ▪ BitBucketBot-AppPerfectCorporation <ul style="list-style-type: none"> ➤ Error Folder Logs: <ul style="list-style-type: none"> ○ Error logs Month-Day-Year Hour Min Sec.txt Snapshots: <ul style="list-style-type: none"> ○ Error Snap Month-Day-Year.png ➤ Input Folder <ul style="list-style-type: none"> ○ Input CSV / Excel files required by different logics should be kept in this folder. By default, we provide empty templates for all input files in Input Folder. ➤ Output Folder <ul style="list-style-type: none"> ○ Output CSV files generated by different operations are stored in the Output folder. ➤ My Tasks <ul style="list-style-type: none"> ○ BitBucket.atmx ❖ My MetaBots <ul style="list-style-type: none"> ➤ BitBucketMetaBot.mbot ➤ JsonUtil.mbot 	<p><AA Application Path> is the location where AA files are stored on your machine</p> <p>1. My Tasks</p> <p>My Task Folder is the default directory where Bot Files are saved.</p> <p>1.1 Bot Store</p> <p>Bot Store Folder contains the Bot Name Folder that the installer creates while installation of the Bot.</p> <p>1.1.1 BitBucketBot-AppPerfectCorporation</p> <p>This folder is created by the installer and contains files and folders that are needed for the bot execution.</p> <ul style="list-style-type: none"> ➤ 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 the execution of the use case are saved. ➤ Output folder is where the output files that the bot needs for the execution of the use case are saved. ➤ My Tasks folder contains all the developed Platform Source Code <p>2. My MetaBots folder contains the developed Metabot needed for the bot execution.</p>

Quick Start

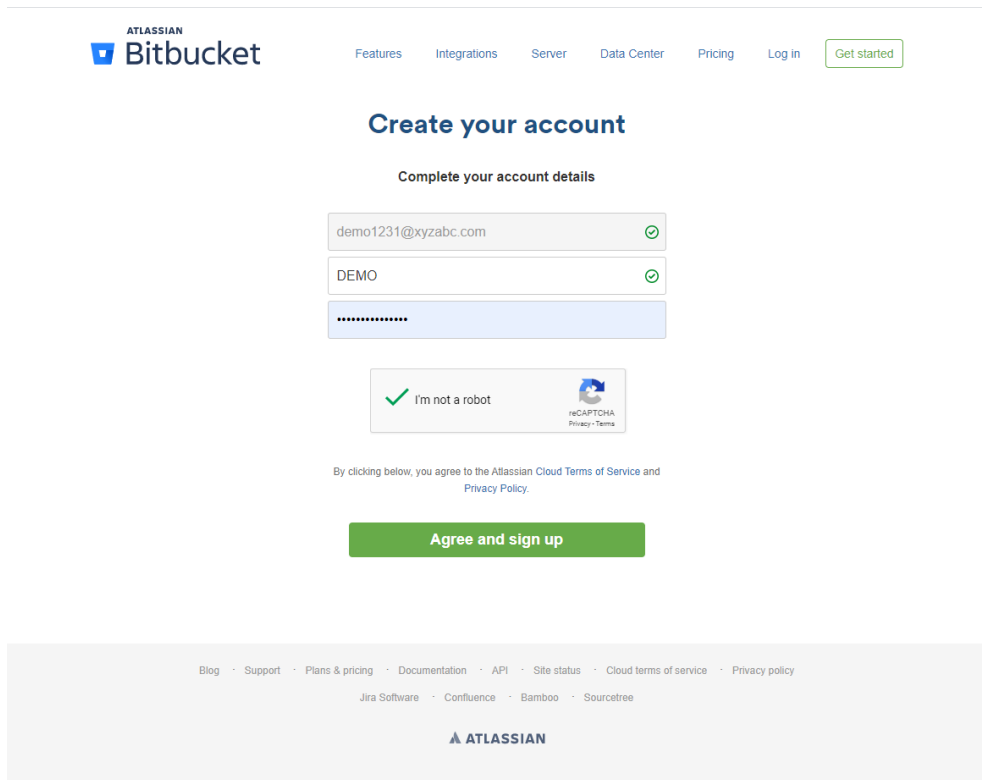
3.1.1 Setup



STEP 1 (Setting up a BitBucket Account):

Create your new BitBucket account [here](#).

The below page will appear once the above link is clicked –



The screenshot shows the BitBucket 'Create your account' page. At the top, there's the Atlassian logo and navigation links: Features, Integrations, Server, Data Center, Pricing, Log in, and a 'Get started' button. The main heading is 'Create your account'. Below it, the text says 'Complete your account details'. There are three input fields: an email field with 'demo1231@xyzabc.com', a username field with 'DEMO', and a password field with masked characters. Each field has a green checkmark icon to its right. Below the fields is a CAPTCHA box with a green checkmark and the text 'I'm not a robot', and a 'reCAPTCHA Privacy - Terms' link. Underneath, there's a line of text: 'By clicking below, you agree to the Atlassian Cloud Terms of Service and Privacy Policy.' At the bottom of the form is a green button labeled 'Agree and sign up'. The footer contains a list of links: Blog, Support, Plans & pricing, Documentation, API, Site status, Cloud terms of service, Privacy policy, Jira Software, Confluence, Bamboo, and Sourcetre. The Atlassian logo is centered at the very bottom.

- If you already have an account, click on **‘Log in’**.
- For new users, enter require details and click on **“Agree and sign up”**.

Step 2 (Generating Client Id and Client Secret):

Create a consumer and get ClientId and ClientSecret

- OAuth needs a key and secret, together these are known as an OAuth consumer.
- To create a consumer, do the following:
 - o From your avatar in the bottom left, click **Bitbucket settings**.
 - o From the left navigation, click **OAuth** and then, **Add Consumer** Button.
 - o Fill the required details and permissions. Required permissions that are required are as follows:
 - Account (Email, Read)
 - Repositories (Read, Write, Admin, Delete)
 - Pull Requests (Read, Write)
 - Issues (Read, Write)
 - Snippets (Read, Write)
 - Pipelines (Read)
 - o Click **Save**. The system generates a key and a secret for you.
 - o Toggle the consumer name to see the generated **Key** and **Secret** value for your consumer.

For more information, click [here](#).

3.1.2 Inputting your Access Code

Step by Step Guide for Customers: Inputting Your Access Code [click here](#)

3.1.3 Configuration



STEP 1 (Configuring Credential in Control Room's Credential Vault)

Sensitive information such as user credentials (Username and Password), Client ID, and Client Secret included in automation tasks, should be stored as credentials centrally in the Credential Vault.

These sensitive data items (stored in CR) can't be accessed locally and can be used by the bots in the execution of task bot.

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

- 'Locker Consumer Role' needs to be assigned to the [AA](#) user (bot runner) which ensures the user has full access to the locker.
- Locker needs to be created with name as "BitBucket"

- Credential needs to be created with name as “BitBucket_Credential”, where the Username, Password, ClientId, and ClientSecret are saved.

Locker Name	Credentials Name	Attribute Name	Value
BitBucket	BitBucket_Credential	Username	Email Address used to access your BitBucket Account
BitBucket	BitBucket_Credential	Password	Password of your BitBucket Account
BitBucket	BitBucket_Credential	ClientId	Add Client Id value which is generated using the above links
BitBucket	BitBucket_Credential	ClientSecret	Add Client Secret value which is generated using the above links

For more information on how to set up credentials in the Control Room, please visit this [link](#) on the Automation Anywhere Product Documentation portal.

STEP 2 (passing value to the variables using input file)

All the configuration files for different operations are provided at **AA Application Path > My Tasks > Bot Store > BitBucketBot-AppPerfectCorporation > Input Folder**.

Below is a table that summaries what the variable name is, what it does, and an example of the input:

SetAccessCode:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vAccessCode	Text	Yes	Provide Access code from your Thank you order email.	For E.g. AppPerfect Corporation 2020-01-01 - yvzasNkldUxrSdsadkyj n1d96JfCY9oZC4HGrsdQ=

Note: The logic Set_Access_Code should be the first logic to be executed before any other logics.

CreateFork:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository whose forks are created	Name of the repository where forks are created
vForkedRepoSlug	Text	Yes	Name of the fork repository	Name of the fork repository
vDescription	Text	No	Description of the fork repository	Description of the fork repository

CreateSnippet:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vTitle	Text	No	Name of the snippet	Name of the snippet
vFolderPath	Text	No	Path of the folder that consists of files to be uploaded	Path of the folder that consists of files to be uploaded

Create Issue:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository where the issue is to be created	Name of the repository where the issue is to be created
vTitle	Text	Yes	Title of the issue	Title of the issue
vDescription	Text	No	Description of the issue	Description of the issue
vKind	Text	No	Kind of the Issue	Kind of the Issue Possible values: bug, enhancement, proposal, task
vPriority	Text	No	Priority of the Issue	Priority of the Issue Possible values: trivial, minor, major, critical, blocker
vAssignee	Text	No	Assignee of the Issue	Assignee of the Issue

Note:

1. Ensure that repoSlug has issue tracker enabled before using this operation.

Steps to enable issue tracker in a repository:

- From a repository, click **Settings**.
- Select the **Issue tracker** from the left-hand tabs.
- Check the option for either a **Private issue tracker** or a **Public issue tracker**.
- The system saves your selection and the **Issue Tracker** item appears in the repository's menu bar.

CreatePullRequest:
INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository where pull request is to be created	Name of the repository where pull request is to be created
vTitle	Text	Yes	Title of the pull request	Title of the pull request
vSource	Text	Yes	Source Branch Name of the pull request	Source Branch Name of the pull request
vDestination	Text	No	Destination Branch Name of the pull request	Destination Branch Name of the pull request
vDescription	Text	No	Description of the pull request	Description of the pull request

CreateRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository to be created	Name of the repository to be created
vDescription	Text	No	Description of the repository	Description of the repository
vIsPrivate	Text	No	Whether the repository is private or not	Whether the repository is private or not Possible values: true, false

DeleteRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository to be deleted	Name of the repository to be deleted

UpdateRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository to be updated	Name of the repository to be updated
vUpdatedRepoSlug	Text	No	Updated name of the repository to be updated	Updated name of the repository to be updated
vDescription	Text	No	Description of the repository	Description of the repository
visPrivate	Text	No	Whether the repository is private or not	Whether the repository is private or not Possible values: true, false

BulkCreateFork:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vInput	Text	Yes	Input that is provided to the bot for the execution.	<p>This can be either an absolute path of CSV/Excel file or JSON Array.</p> <p>Case 1: Input is a CSV/Excel file Provide the complete path of the input CSV file. For E.g. -C:\Users\Administrator\Desktop\BulkCreateFork.csv.</p> <p>CSV/Excel file headers must be: 1.) workspaceId - Workspace id of the Bitbucket Account 2.) repoSlug - Name of the repository to be forked 3.) forkedRepoSlug - Name of the forked repository 4.) description - Description of the forked repository</p> <p>Case 2: Input is a JSON Array For E.g. – [{"workspaceId":"demo","repoSlug":"demo repo","forkedRepoSlug":"forked repo","description":"repo description"}]</p> <p>Note: The header - description is optional.</p>
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	<p>Provide the complete path of the CSV output file in which you want to save the response of the operation.</p> <p>For E.g. - C:\Users\Administrator\Desktop\BulkCreateForkOutput.csv</p>

BulkCreateIssue:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vInput	Text	Yes	Input that is provided to the bot for the execution.	<p>This can be either an absolute path of CSV/Excel file or JSON Array.</p> <p>Case 1: Input is a CSV/Excel file Provide the complete path of the input CSV file. For E.g. -C:\Users\Administrator\Desktop\BulkCreateIssue.csv. CSV/Excel file headers must be:</p> <ol style="list-style-type: none"> 1.) workspaceId - Workspace id of the Bitbucket Account 2.) repoSlug - Name of the repository 3.) title - Name of the issue 4.) description - Description of the issue 5.) kind - Kind of the issue 6.) priority - Priority of the issue 7.) assignee - Assignee of the issue <p>Possible values of kind: bug, enhancement, proposal, task Possible values of priority: trivial, minor, major, critical, blocker</p> <p>Case 2: Input is a JSON Array For E.g. – [{"workspaceId":"demo","repoSlug":"demo repo","title":"issue name","description":"issue description","kind":"task","priority":"major","assignee":"prikanksha_trial"}]</p> <p>Note: The headers – description, kind, priority and assignee are optional.</p>
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	<p>Provide the complete path of the CSV output file in which you want to save the response of the operation.</p> <p>For E.g. - C:\Users\Administrator\Desktop\BulkCreateIssueOutput.csv</p>

Note:

1. Ensure that repoSlug has issue tracker enabled before using this operation.

Steps to enable issue tracker in a repository:

- From a repository, click **Settings**.
- Select the **Issue tracker** from the left-hand tabs.
- Check the option for either a **Private issue tracker** or a **Public issue tracker**.
- The system saves your selection and the **Issue Tracker** item appears in the repository's menu bar.

BulkCreatePullRequest:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vInput	Text	Yes	Input that is provided to the bot for the execution.	<p>This can be either an absolute path of CSV/Excel file or JSON Array.</p> <p>Case 1: Input is a CSV/Excel file Provide the complete path of the input CSV file. For E.g. - C:\Users\Administrator\Desktop\BulkCreatePullRequest.csv.</p> <p>CSV/Excel file headers must be:</p> <ol style="list-style-type: none"> 1.) workspaceId - Workspace id of the Bitbucket Account 2.) repoSlug - Name of the repository 3.) title - Name of the pull request 4.) source - Source branch name of the pull request 5.) destination - Destination branch name of the pull request 6.) description - Description of the pull request <p>Case 2: Input is a JSON Array For E.g. – [{"workspaceId":"demo","repoSlug":"demo repo","title":"My Title", "source":"my-feature-branch","destination":"staging", "description":"description of pull request"}]</p> <p>Note: The headers – destination, and description are optional.</p>
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	<p>Provide the complete path of the CSV output file in which you want to save the response of the operation.</p> <p>For E.g. - C:\Users\Administrator\Desktop\BulkCreatePullRequestOutput.csv</p>

BulkCreateRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vInput	Text	Yes	Input that is provided to the bot for the execution.	<p>This can be either an absolute path of CSV/Excel file or JSON Array.</p> <p>Case 1: Input is a CSV/Excel file Provide the complete path of the input CSV file. For E.g. - C:\Users\Administrator\Desktop\BulkCreateRepository.csv.</p> <p>CSV/Excel file headers must be: 1.) workspaceId - Workspace id of the Bitbucket Account 2.) repoSlug - Name of the repository to be created 3.) description - Description of the repository 4.) is_private - Whether repository is private or not Possible values of is_private: true, false.</p> <p>Case 2: Input is a JSON Array For E.g. – [{"workspaceId":"demo","repoSlug":"demo repo","description":"repo description","is_private":"true"}]</p> <p>Note: The headers – description and is_private are optional.</p>
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	<p>Provide the complete path of the CSV output file in which you want to save the response of the operation.</p> <p>For E.g. - C:\Users\Administrator\Desktop\BulkCreateRepositoryOutput.csv</p>

BulkCreateSnippet:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vInput	Text	Yes	Input that is provided to the bot for the execution.	<p>This can be either an absolute path of CSV/Excel file or JSON Array.</p> <p>Case 1: Input is a CSV/Excel file Provide the complete path of the input CSV file. For E.g. - C:\Users\Administrator\Desktop\BulkCreateSnippet.csv.</p> <p>CSV/Excel file headers must be: 1.) workspaceId - Workspace id of the Bitbucket Account 2.) title - Name of the snippet to be created 3.) folderPath - Path of the folder which has all the files that need to be uploaded</p> <p>Case 2: Input is a JSON Array For E.g. – [{"workspaceId":"demo","title":"demo snippet","folderPath":"C:\Users\Administrator\Desktop\abc"}]</p> <p>Note: The headers – title and folderPath are optional.</p>
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	<p>Provide the complete path of the CSV output file in which you want to save the response of the operation.</p> <p>For E.g. - C:\Users\Administrator\Desktop\BulkCreateSnippetOutput.csv</p>

BulkDeleteRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vInput	Text	Yes	Input that is provided to the bot for the execution.	<p>This can be either an absolute path of CSV/Excel file or JSON Array.</p> <p>Case 1: Input is a CSV/Excel file Provide the complete path of the input CSV file. For E.g. - C:\Users\Administrator\Desktop\BulkDeleteRepository.csv.</p> <p>CSV/Excel file headers must be: 1.) workspaceId - Workspace id of the Bitbucket Account 2.) repoSlug - Name of the repository to be deleted</p> <p>Case 2: Input is a JSON Array For E.g. – [{"workspaceId":"demo","repoSlug":"demo repo"}]</p>
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	<p>Provide the complete path of the CSV output file in which you want to save the response of the operation.</p> <p>For E.g. - C:\Users\Administrator\Desktop\BulkDeleteRepositoryOutput.csv</p>

GetCommit:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository	Name of the repository
vCommitSHA	Text	Yes	The SHA1 of the commit	The SHA1 of the commit
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetCommitsOutput.csv

GetDeployments:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository in which deployments are present	Name of the repository in which deployments are present
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetDeploymentsOutput.csv

GetEnvironments:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository in which environments are present	Name of the repository in which environments are present
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetEnvironmentsOutput.csv

GetForks:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository whose forks are created	Name of the repository where forks are created
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetForksOutput.csv

GetHookEvents:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetHookEventsOutput.csv

GetIssuesInRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository in which issues are present	Name of the repository in which issues are present
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetIssuesOfRepositoryOutput.csv

GetPipelinesInRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository in which pipelines are present	Name of the repository in which pipelines are present
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetPipelinesInRepositoryOutput.csv

GetPublicRepositories:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vReposCount	Text	No	Number of repositories to be retrieved	By default, the value is 10.
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetPublicRepositoriesOutput.csv

GetPullRequestsInRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository in which pull requests are present	Name of the repository in which pull requests are present
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetPullRequestsOfRepository Output.csv

GetPullRequests:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vSelectedUser	Text	Yes	Username of the pull request Author	Username of the pull request Author
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetPullRequestsOutput.csv

GetSnippets:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetSnippetsOutput.csv

GetUserRepositories:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetUserRepositoriesOutput.csv

GetUserRepositoriesPermissions:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetUserRepositoriesPermissionsOutput.csv

GetUserTeamsPermissions:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetUserTeamsPermissionsOutput.csv

GetMilestonesInRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vRepoSlug	Text	Yes	Name of the repository in which milestones are present	Name of the repository in which milestones are present
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\GetMilestonesOfRepositoryOutput.csv

SearchCode:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vWorkspaceId	Text	Yes	Workspace id of the Bitbucket Account	Workspace id of the Bitbucket Account
vSearchQuery	Text	Yes	Query to search in code	Query to search in code
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	Provide the complete path of the CSV output file in which you want to save the response of the operation. For E.g. - C:\Users\Administrator\Desktop\SearchCodeOutput.csv

BulkUpdateRepository:

INPUT VARIABLES: NEEDS TO BE CONFIGURED BY THE USERS FOR INPUT PARAMETER

Variable Name	Type	Mandatory (Yes/No)	Purpose	Additional Info
vInput	Text	Yes	Input that is provided to the bot for the execution.	<p>This can be either an absolute path of CSV/Excel file or JSON Array.</p> <p>Case 1: Input is a CSV/Excel file Provide the complete path of the input CSV file. For E.g. - C:\Users\Administrator\Desktop\BulkUpdateRepository.csv.</p> <p>CSV/Excel file headers must be:</p> <ol style="list-style-type: none"> 1.) workspaceId - Workspace id of the Bitbucket Account 2.) repoSlug - Name of the repository 3.) updatedRepoSlug - Update name of the repository 4.) description - Description of the repository 5.) is_private - Whether repository is private or not <p>Possible values of is_private: true, false.</p> <p>Case 2: Input is a JSON Array For E.g. – [{"workspaceId":"demo","repoSlug":"demo repo","updatedRepoSlug":"repo new","description":"repo description","is_private":"true"}]</p> <p>Note: The headers - updatedRepoSlug, description and is_private are optional.</p>
vOutputFilePath	Text	No	Absolute file path where the output of the logic is to be saved.	<p>Provide the complete path of the CSV output file in which you want to save the response of the operation.</p> <p>For E.g. - C:\Users\Administrator\Desktop\BulkUpdateRepositoryOutput.csv</p>

JsonUtil.mbot –

This metabot provides the following JSON functionalities-

Functions	Parameter Values
<p>GetJsonKeyValue-</p> <p>Returns value of specified key in the given JSON structure.</p> <p>Note: In case of JSONArray provide input for vKey as [index].key to get value of a key from a particular index.</p>	<p>vJson: Provide valid JSON Array/Object structure.</p> <p>For E.g.</p> <pre>{ "user": "user1", "domain": "domain1.com" }</pre> <p>vKey: Provide valid key present in vJson.</p> <p>For E.g. Considering above vJson example, the value of vKey can be user</p>
<p>JsonGetType-</p> <p>Returns type of JSON structure if valid JSON structure is provided, throws Exception otherwise.</p>	<p>vJson: Provide valid JSON Array/Object structure.</p> <p>For E.g.</p> <pre>{ "user": "user1", "domain": "domain1.com" }</pre>
<p>JsonStringify-</p> <p>Returns String Response of valid JSON Structure,throws Exception otherwise.</p>	<p>vJson: Provide valid JSON Array/Object structure.</p> <p>For E.g.</p> <pre>{ "user": "user1", "domain": "domain1.com" }</pre>
<p>GetJsonArraySize-</p> <p>Returns size of provided JSON Array.</p>	<p>vJson: Provide valid JSON Array structure.</p> <p>For E.g.</p>

	<pre>[{ "user": "user1" }, { "user": "user2" }]</pre>
<p>CsvToJsonArray- Converts CSV string to JSON Array Response.</p>	<p>vCsv- Provide a comma-separated string to be converted into JSON Array String. For E.g. source1, source2, source3</p>

4. Logs



In case of Errors, Error Logs & Screenshots are generated within Error Folder (Highlighted below in yellow)

- My Tasks
 - Bot Store
 - Bot Name
 - Error Folder
 - Logs (Folder)
 - Error Logs Month-Day-Year.txt
 - Snapshots (Folder)
 - Error Snap Month-Day-Year HourMinSec.png

Error Logs will contain the below information -

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



Example:

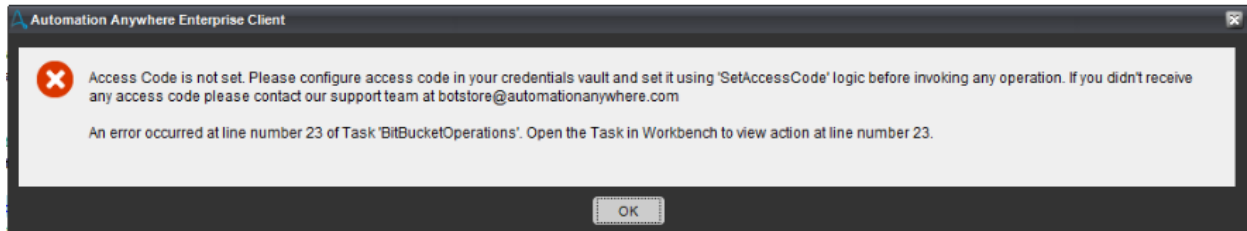
```
Error Logs 3-20-2020.txt - Notepad
File Edit Format View Help
(20-03-2020 13:15:23)
=====
Task Name C:\Users\Administrator\Documents\Automation Anywhere Files\My Tasks\Bot Store\BitBucketBot-AppPerfectCorporation\My Tasks\BitBucket
Error Line Number: "23"
Error Description: Access Code is not set. Please configure access code in your credentials vault and set it using 'SetAccessCode' logic before
=====
```

Note: There are no Custom Logs generated for this Bot.

Example of Screenshot Generated Error File:

Automation Anywhere Files > Automation Anywhere > My Tasks > Bot Store > BitBucketBot-AppPerfectCorporation > Error Folder > Snapshots

Name	Date	Type	Size	Tags
 Error Snap 3-18-202...	18-03-2020 13:27	PNG File	278 KB	
 Error Snap 3-20-202...	20-03-2020 13:15	PNG File	263 KB	



5. Troubleshooting & Support



Support



Please visit our [Support Portal](#) for any assistance on Bot functionality or Feature.



Automation Anywhere provides a [Product Documentation portal](#) that 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 Advanced Bots - details on MetaBots and the approach to integrating code into them
- Build Digital Workers - high-level architecture

FAQs



Q: The 'Create Locker' button is 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 (DD/MM/YYYY)	Author	Notes
1	1.0	06/04/2020	AppPerfectCorporation	Version 1 Release

Appendix B: Acronyms

No.	Acronym	Description
1	AA	Automation Anywhere
2	CSV	Comma Separated File
3	DW	Digital Worker
4	CR	Credential Vault
5	CRM	Customer Relationship Management
6	API	Application Programming Interface
7	PC	Personal Computer
8	RAM	Random Access Memory
9	OS	Operating System

Appendix C: References

No.	Topic	Reference Link
1	System Requirements - Client	Click here
2	System Requirements – Control Room	Click here
3	How to create credential & Locker?	Click here
4	Credential Overview	Click here
5	Apeople Community Site	Click here
6	Product Documentation portal	Click here