

Configuring a GIT Repository

If you are going to use Visual Accessors, you must use a SFTP, FTP, or S3 repository. This is because Images are resource-consuming elements, at the disk, memory, and CPU level, so using an S3 repository will yield the best performance. The images will be saved with the script in your S3 repository which simplifies image management.

Appvance does not recommend using Git to store your images if you are using the Visual Accessor functionality. This is due to Git not being designed to handle large amounts of binary files efficiently.

Creating a new GIT Repository

1. Enter a name for the repository in the Name box. This is a locally referenced name, and must not match the actual repository name.

2. Click the Type box and select Git or SVN.
3. Enter the URL to your repository, for example,
<https://gitlab.com/CompanyABC/myRepository.git>
4. Enter your credentials in the Username and Password boxes
 - a. For Git SSH keys see <https://gitlab.com/help/ssh/README>
 - b. Click Clone Repository
5. The Status indicator displays a Cloning status until all assets are retrieved.
6. Optionally click Checkout Branch to select an alternate branch.
 - Note: As many users do not have write permissions to Master branches, selecting an alternate branch is recommended. Switching branches can take time to retrieve all

files. The Status indicator displays a Cloning status until all assets are retrieved.

7. Any number of repository connections can be created, and appear in the Select box on the Repositories tab.

Repository Updates

Repositories are updated in real-time and are available throughout the entire Appvance IQ system, including, but not limited to, Scenario Builder and Test Designer and other script and test assets.

Files generated or updated using Appvance IQ are automatically added, committed, and (in the case of Git, pushed) to their respective repository servers.

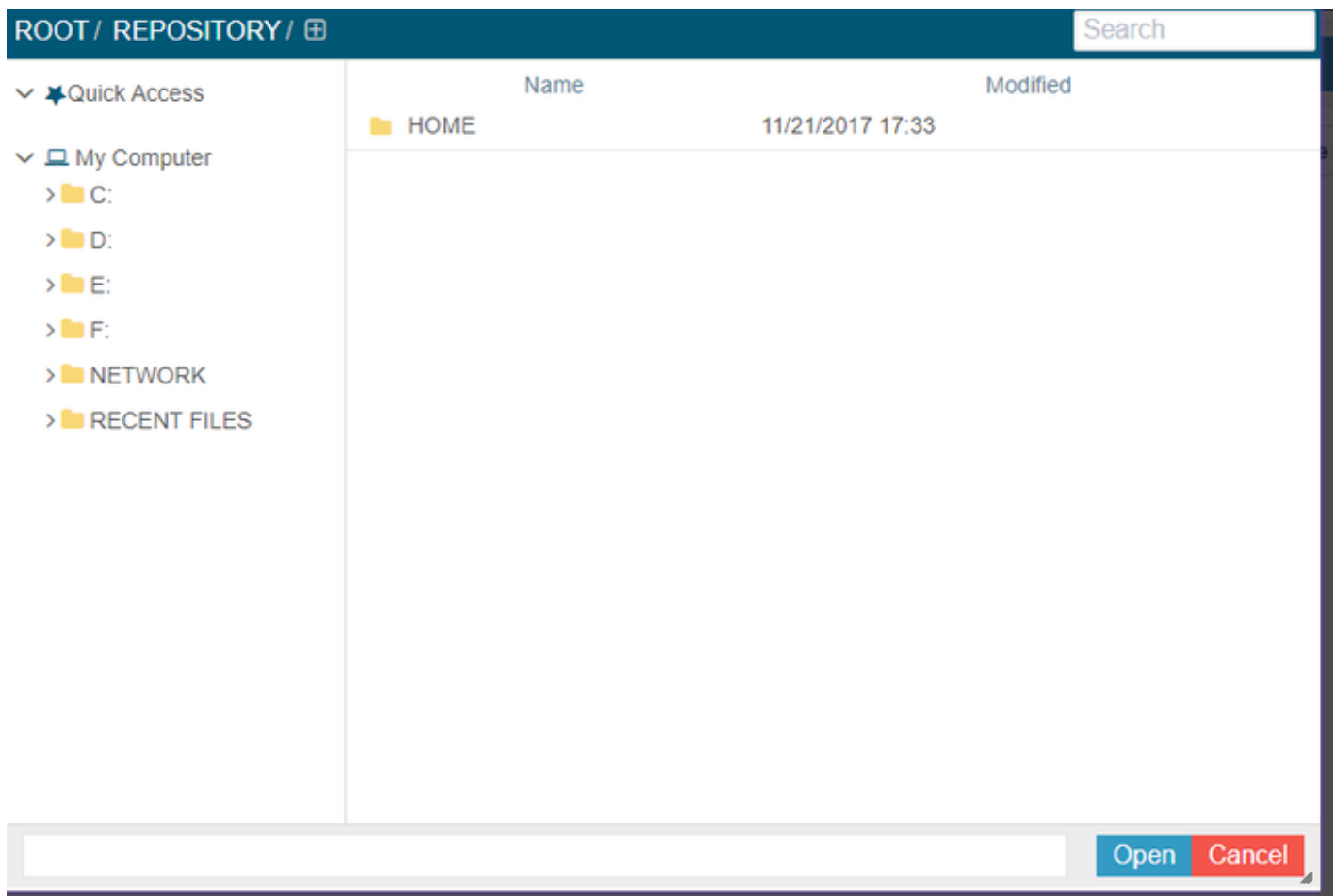
Repositories are pulled at a 5-minute interval. To force a pull sooner (retrieving any changes to the repository not already present within AIQ), select a repository configuration and click Pull selected repository. For example, if another user updates or adds files, or you add or update files from another location, you may wish to retrieve the latest changes before the auto-update interval.

Click Clear to remove the currently selected repository configuration, providing an empty form to add additional repository connections.

Click Delete to remove the repository connection. Note: This will not delete a repository or any of its contents from the repository database. Deleting a repository will only remove its availability from AIQ.

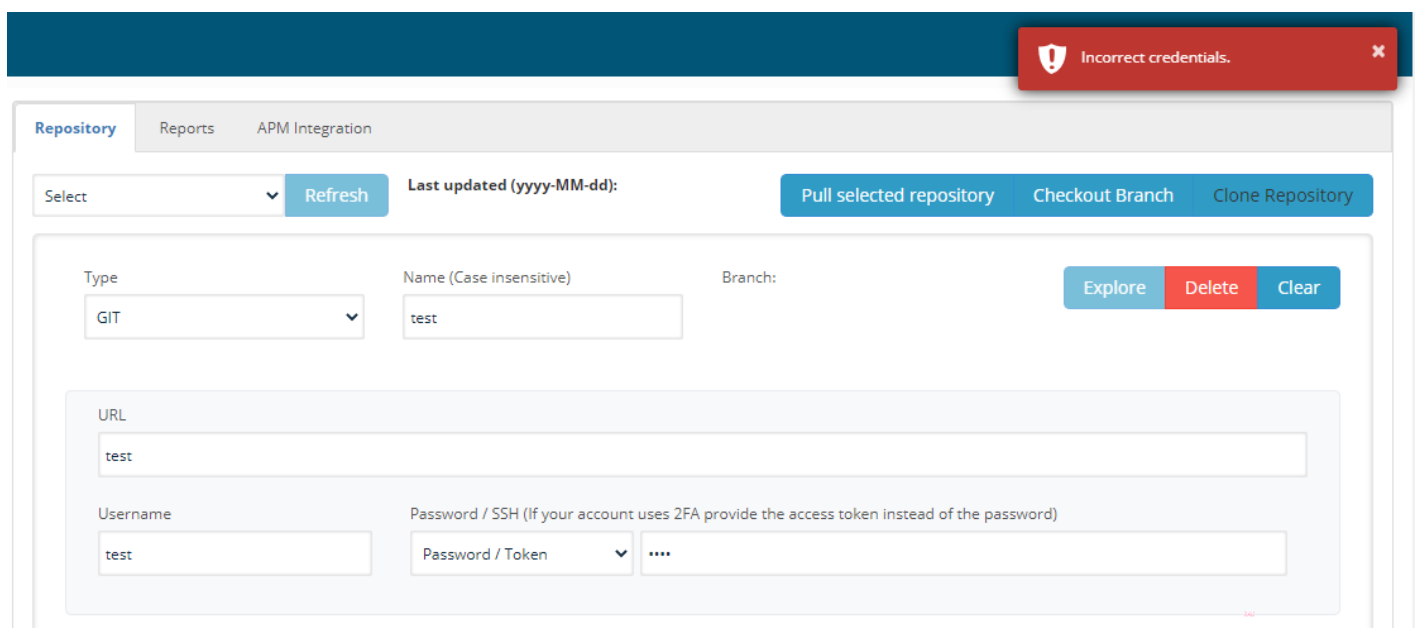
Accessing Repositories in AIQ

Connected repositories appear in the QuickStart list within the REPOSITORIES folder. Expand the Quick Start list to view any connected repositories.



Should a repository not contain the expected files, return to **Global Options > Preferences** and select the **Repositories** tab to confirm the connection status.

When a user provides invalid credentials, the user shall be notified before adding the repository to the list like below.



Saving to Repositories: When the same file is used by multiple users

The implementation of the GIT repository tries to avoid the possibility of a merge conflict. To achieve that, every time you SAVE a file, Appvance will FIRST PULL any changes from the remote branch, then save, then commit and finally push. This is every time someone saves a file.

This implies that, as a best practice, no one should be working on the same file at the same time (Just like how a windows file should not be shared among users for editing). If this happens, and both persons save their changes one after the other, the changes that will prevail are the changes of the person who saved last.

Implementing a merge conflict resolution method might bring more problems than fixes to this matter, as opens the possibility of, for example, breaking scripts or making a scenario XML file corrupt or at least invalid as the merge can produce not system congruent files. This kind of error caused by merged scenario files, for example, would be quite difficult to detect and fix, causing more problems.

How to add Git repository using SSH key

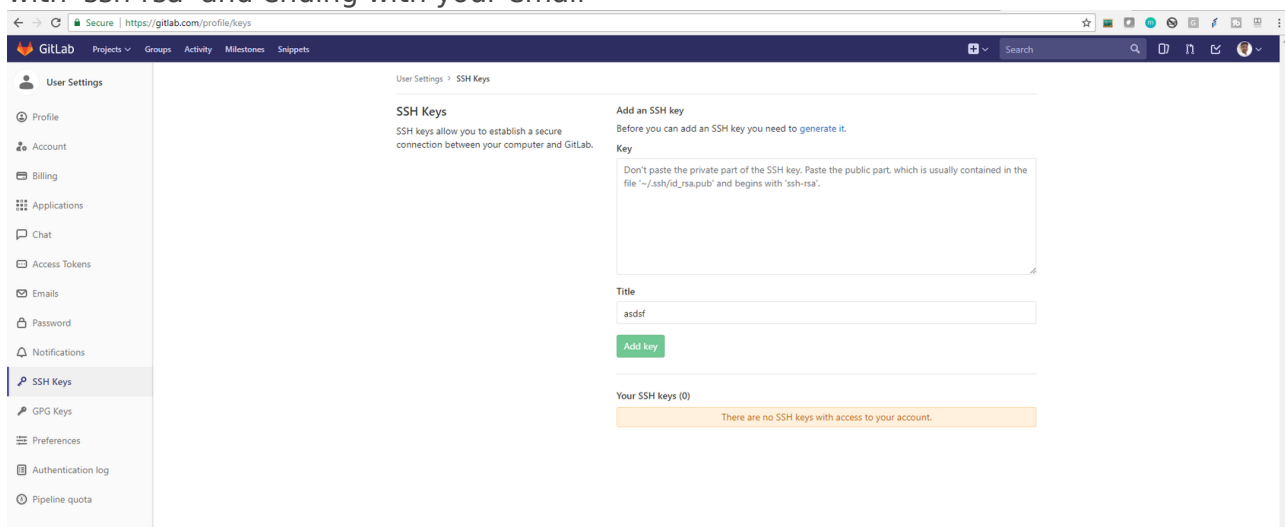
1. Generate SSH key:

Git Bash on Windows / GNU/Linux / macOS:

```
ssh-keygen -t RSA -C "your.email@example.com" -b 4096
```

For more details refer: [Gitlab ReadMe](#)

2. The next step is to add your public SSH key to GitLab. Navigate to the 'SSH Keys' tab in your 'Profile Settings'. Paste your key in the 'Key' section and give it a relevant 'Title'. If you manually copied your public SSH key make sure you copied the entire key starting with 'ssh-rsa' and ending with your email



3. Navigate to Global Options > preferences > Repository page and select GIT from the Type drop-down

4. Enter a name in the Name field. Fill the URL and user name with your account information on git
5. Select SSH key
6. Open the file 'id_rsa' (public key) on any of your favorite editors like notepad++. Copy and paste the entire content
7. Click Accept button
8. Select the above-created repository name from the select drop-down
9. The status will be shown as "WORKING", after fetching data from the repository

The screenshot shows the 'Repository' tab in the APM Integration section. At the top, there are tabs for 'Repository', 'Reports', and 'APM Integration'. Below the tabs, there is a 'Select' dropdown menu, a 'Refresh' button, and a 'Last updated (yyyy-MM-dd):' label. To the right, there are three buttons: 'Pull selected repository', 'Checkout Branch', and 'Clone Repository'. The main form area contains several fields: 'Type' (set to 'GIT'), 'Name (Case insensitive)' (set to 'testrepo'), and 'Branch:'. There are also buttons for 'Explore', 'Delete', and 'Clear'. Below these, there is a 'URL' field containing 'git@gitlab.com:naveen8827/dummytest.git'. Underneath, there are fields for 'Username' (set to 'test') and 'Password / SSH (If your account uses 2FA provide the access token instead of the password)'. The 'Password / SSH' field has a dropdown menu set to 'SSH Key' and a text area for 'Paste your private key here...'. There is also a 'Passphrase(Optional)' field. At the bottom, there is a 'Default Commit Message' field and a 'Status:' field showing 'WORKING'. A 'Stacktrace:' section is also visible at the bottom.

How to specify a branch with the git repository

1. Provide all the required details as mentioned in earlier sections with the GIT Repo URL and click Clone Repository

Select ▼ Refresh Last updated (yyyy-MM-dd): Pull selected repository Checkout Branch Clone Repository

Type: GIT ▼ Name (Case insensitive): testrepo Branch: Explore Delete Clear

URL:

Username: Password / SSH (If your account uses 2FA provide the access token instead of the password): Password / Token ▼

Proxy Options: Select an option ▼

2. You will notice the current branch: MASTER below the status, now you can click Checkout Branch on top

Checkout new branches ✕

Please select the brach to checkout:

Select ▼

- Select
- master
- patch-1
- patch-2
- patch-3

Checkout Branch Cancel

3. You can select the branch from this list and click Checkout Branch.
It will take a while to download and sync the files based on the new branch selected and you can notice that the branch would be changed

Repository Reports APM Integration

Dummy ▼ Refresh Last updated (yyyy-MM-dd): Pull selected repository Checkout Branch Clone Repository

Type: GIT ▼ Name (Case insensitive): Dummy Branch: **BRANCH3** Explore Delete Clear

URL:

Username: Password / SSH (If your account uses 2FA provide the access token instead of the password): Password / Token ▼

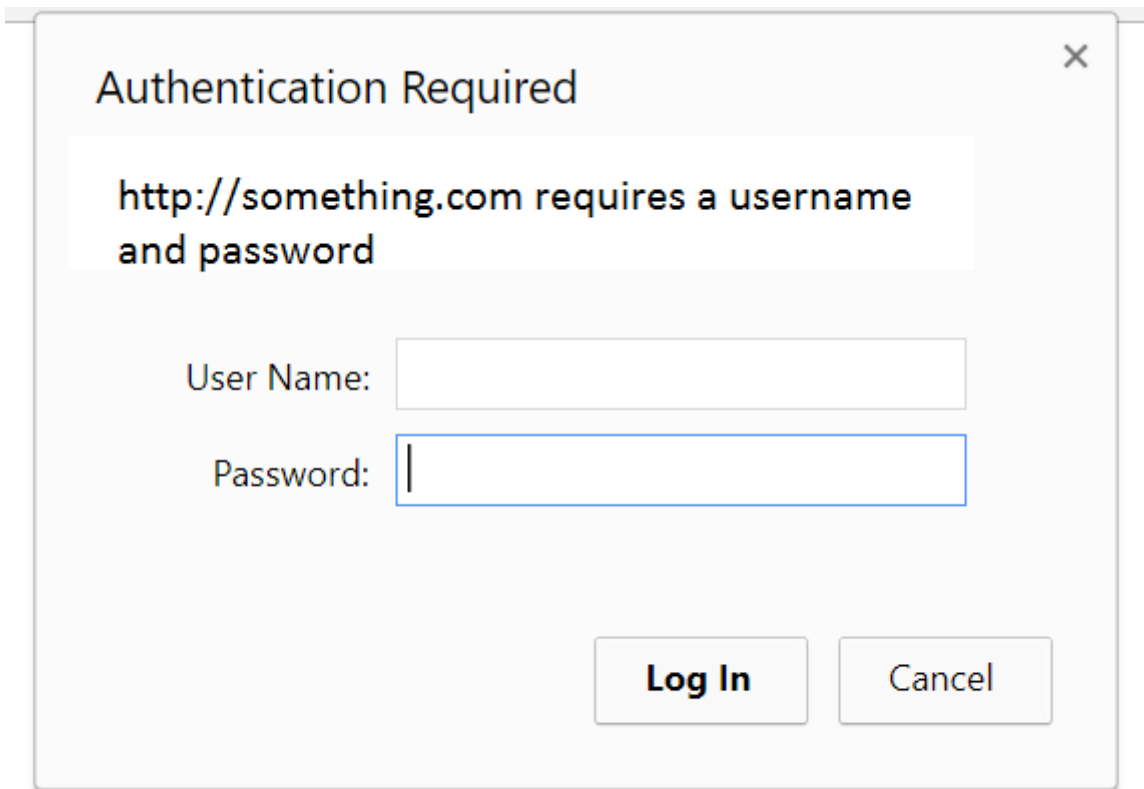
Proxy Options: Select an option ▼

Default Commit Message: Status:

4. Now you can save all your files to the selected branch if you do not have enough permissions to save changes to the master branch (which usually is restricted to many users)

Repository with multiple Authentication (Proxy Server)

When the repository needs to be configured as a multilayer authentication that needs the proxy server details to be added, this option can be chosen.



The image shows a dialog box titled "Authentication Required" with a close button (X) in the top right corner. Inside the dialog, a message states: "http://something.com requires a username and password". Below this message, there are two input fields: "User Name:" followed by a text box, and "Password:" followed by a text box. At the bottom of the dialog, there are two buttons: "Log In" and "Cancel".

Just like adding the Repository details above, with the URL and credentials, Choose Proxy Options from the drop-down, fill in the web-proxy Server, Port, and the credentials of the Proxy Server.

Click Clone Repository after providing all the valid details and credentials.

Note: Do not provide HTTP:// or HTTPS:// for the Web Proxy Server.

Repository

Reports

APM Integration

Dummy

Refresh

Last updated (yyyy-MM-dd):

Pull selected repository

Checkout Branch

Clone Repository

Type

Name (Case insensitive)

Branch: **BRANCH3**

Explore

Delete

Clear

URL

https://gitlab.com/naveen8827/dummytest.git

Username

Password / SSH (If your account uses 2FA provide the access token instead of the password)

yourusername

Password / Token

.....

Proxy Options

Proxy Server

Web Proxy Server

Web Proxy Port

Username

Password

ec2.x.x.x.us-west-2.compute.amazon

8888

appvance

.....

Default Commit Message

Status:

ENDCONFIGURE

Stacktrace:

Committing with a GIT message

You can now commit files from AIQ with a commit message while saving files, Renaming, and/or deleting files.

When you configure a GIT Repo, You will have an option to provide the default Commit Message which defaults to "Appvance Commit" which can be changed and saved while cloning.

Once the repository is configured, In all the file explorers from IDE or from anywhere in AIQ when saving a file, you should be seeing a text field with the default message which can be changed as well.

This will be seen only for Repository configurations and not for local file saves.

ROOT / REPOSITORY / 1 /

Search

Quick Access

My Computer

Name	Modified	Size
AppPenetration	05/29/2019 15:17	
AppsHarProxy	05/29/2019 15:17	
Training	05/29/2019 15:17	
aaaaaklkikijmnnjuhujuuijkuijhuasfnsdkfhaskdhfasdhfjskjsadkf.ds	05/29/2019 15:17	3.5 KB
applehar.har	05/29/2019 15:17	1.8 MB
blueprint.abp	05/29/2019 15:17	3.0 KB
blueprinttest.abp	05/29/2019 15:17	3.0 KB
dsscriptRENAMEDFILE.ds	05/29/2019 15:17	422.0 B
dummytest.ds	05/29/2019 15:17	450.0 B
ES.ds	05/29/2019 15:17	450.0 B
ExportDatasettoGIT.json	05/29/2019 15:17	522.0 B

File Name

Commit Message

Save asCancel

A commit message can also be provided when you rename or delete a file.

ROOT / REPOSITORY / 1 /

Search

Quick Access

My Computer

Name	Modified	Size
AppPenetration	05/29/2019 15:17	
AppsHarProxy	05/29/2019 15:17	
Training	05/29/2019 15:17	
aaaaaklklkijkmnjuijnjuhujuijkuijhuasfnsdkfhaskdhfasdhfjskjsadkf.ds	05/29/2019 15:17	3.5 KB
applehar.har	05/29/2019 15:17	1.8 MB
blueprint.abp	05/29/2019 15:17	3.0 KB
blueprinttest.abp	05/29/2019 15:17	3.0 KB
ds	05/29/2019 15:17	422.0 B
du	05/29/2019 15:17	450.0 B
ES.ds	05/29/2019 15:17	450.0 B
ExportDatasettoGIT.json	05/29/2019 15:17	522.0 B

File Name

OpenCancel

Clickbutton("Continue[1]")

Commit Message

Renaming DS1 Script

CommitCancel

Commit Message

Deleting DS2.ds Script

CommitCancel

All the messages will be reflected in your GitLab for the configured repository.

Last updated: March 21, 2024

If you have any feedback on this topic, please contact us at documentation@appvance.ai

Revision #4

Created 2026-07-31 16:29:00 UTC by Admin

Updated 2026-10-14 13:51:53 UTC by AI_Mananger