DEVELOP Virtual Machine Guidebook

Your virtual machine (VM) is a NASA-sponsored Amazon Web Services (AWS) Science Data Managed Cloud Environment (SMCE) Elastic Compute Cloud (EC2) instance. Here's a quick guide on getting it started and using some of its helpful features. Navigate to any section directly by clicking on its name in the Table of Contents.

Troubleshooting note: If you are experiencing any difficulties with your VM, reach out to James (Jim) Davis via Microsoft Teams or email (james.davis@ssaihq.com).

Section 1.   Creating your EC2 and starting your VM (for the first time)

Note: If you have already created the EC2 previously, you can skip this section and proceed to Section 2. Starting your VM.

1. Navigate to the Jenkins server login page: https://jenkins.developprogram.org

Note: If you are logged into a NASA VPN, the page will not load. Disconnect from the VPN and reload the Jenkins server login page. Most participants will not have VPN access, which mainly applies to Fellows.

2. Sign in with the password provided to you by James (Jim) Davis. The username should look something like `firstname.lastname`, and the password will be pretty long!

    Note: Passwords are set to expire every 60 days. Refer to Section III on how to check and change your password.
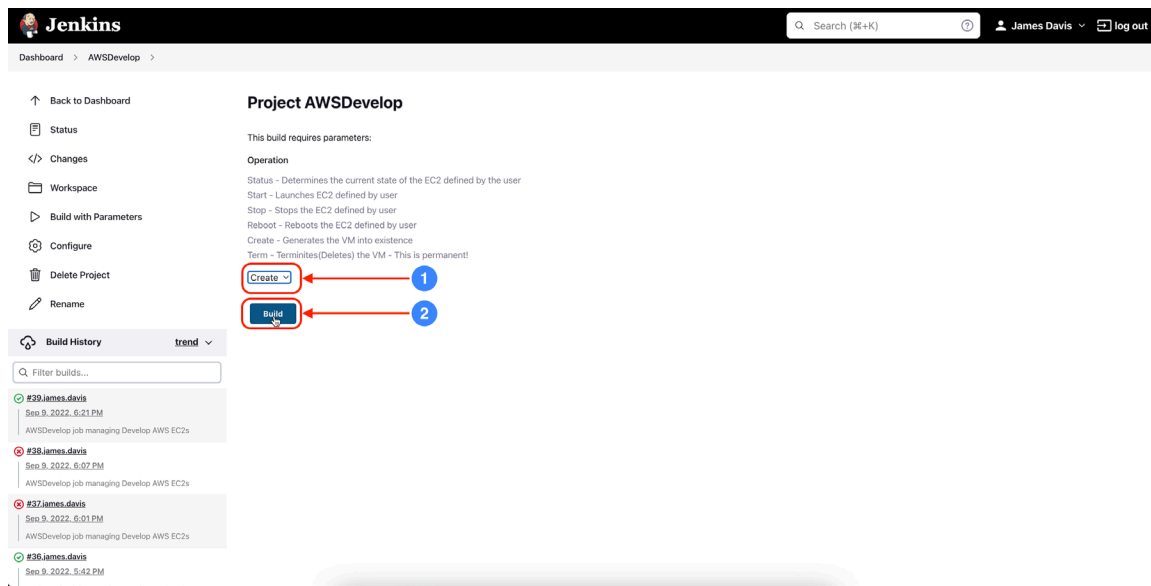
3. Select `AWSDevelop`. This is where you can view the job build history and start a new job.
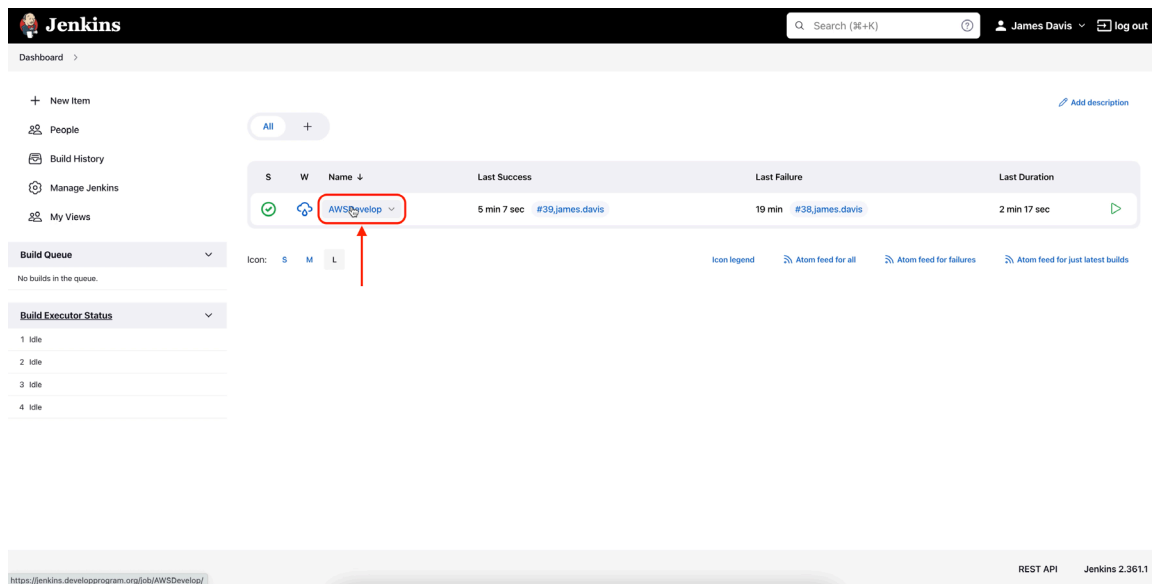
4. On the left-hand panel, select `Build with Parameters`. This is where you will first create your EC2 instance.
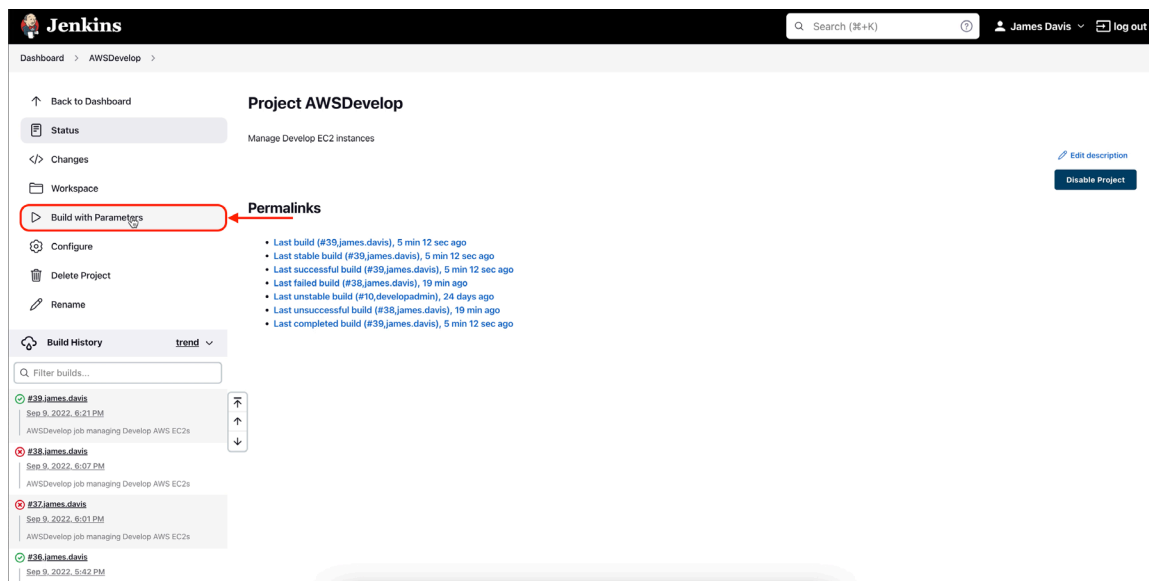
5. To create your EC2 Instance for the first time, you must select `Create` from the `Operation` dropdown menu. From that, click the blue `Build` button.

6. Once you click the **Build** button, your job will be submitted into the Jenkins queue. Your **Create** job will always be the first one in the queue. Click on the progress bar of your **Create** job in the queue to view the status in the **Console Output** screen.
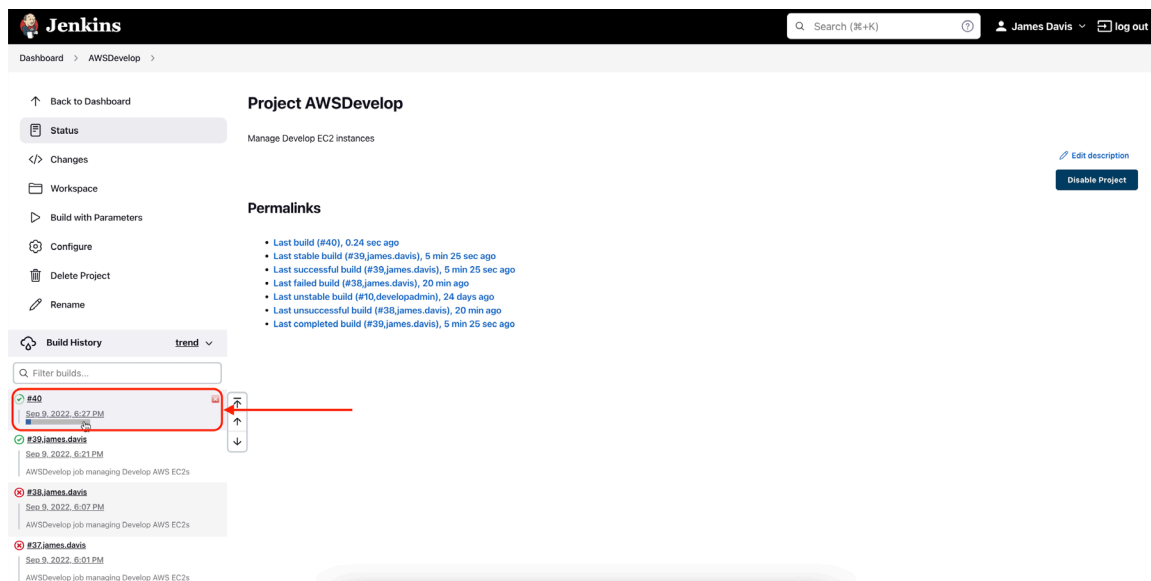
7. On the left-hand panel, select **Console Output** The **Console Output** screen will show the status of your **Create** job.
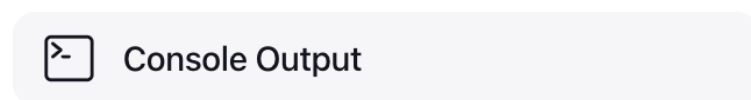
8. *The first time you create your EC2 instance may take a while, so be patient and let it run until the job status indicates "Finished: SUCCESS."*

9. You are going next want to go back to the **Project AWSDevelop** screen to get the status of your build. Click on the **AWSDevelop** link on the upper left of the screen.

10. On the left-hand panel, select **Build with Parameters**.

11. Select **Status** from the **Operation** dropdown menu. From that, click the blue **Build** button.

12. Click on the progress bar of your **Status** job in the queue to view the status in the **Console Output** screen.

13. On the left-hand panel, select **Console Output**. The **Console Output** screen will show the status of your **Status** job.

14. Allow the **Status** job to complete until the job status indicates "Finished: SUCCESS."

Once complete, locate the header in the Console Output log indicated by Owner/State/ImageId/Ip Addr/InstanceId/Type/Purpose/PublicIp.

Note and write down the **PublicIp**; This will be the IP address you will use to login remotely to your VM.

15. At this point, you can log out of Jenkins and proceed to Section 3. Remote into Your VM.

Section 2.   Starting your VM

1. Navigate to the Jenkins server login page: https://jenkins.developprogram.org

Welcome to Jenkins!

| Username |
| Password |
| Sign in |
☐ Keep me signed in

2. Sign in with the password provided to you by James (Jim) Davis. The username should look something like `firstname.lastname`, and the password will be pretty long!

3. Select `AWSDevelop`. This is where you can view the job build history and start a new job.

4. On the left-hand panel, select `Build with Parameters`. This is where you will start your EC2 instance.
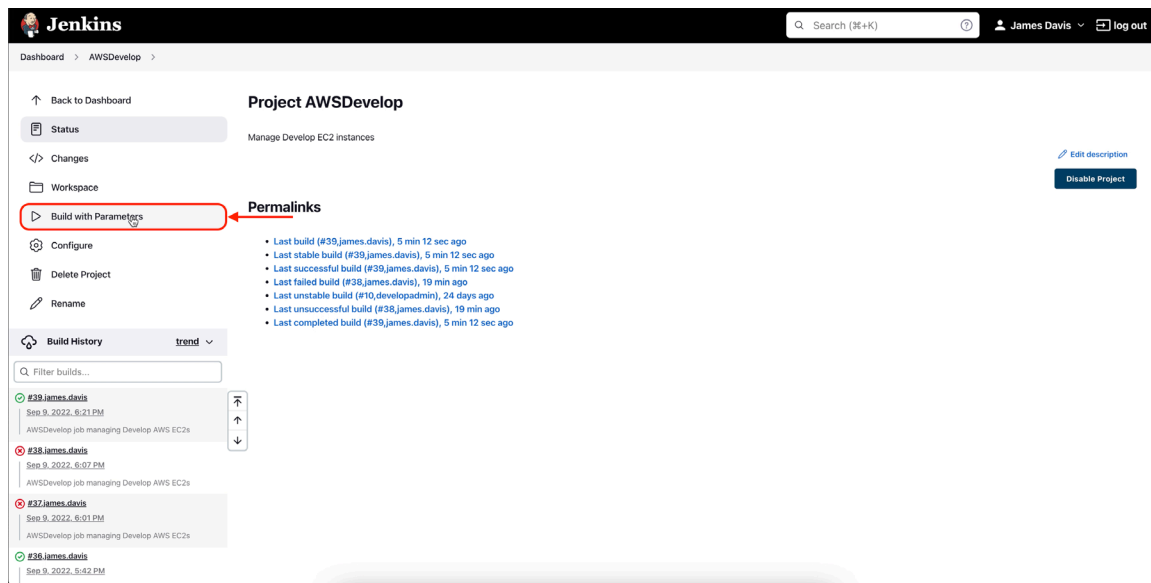
5. To start your EC2 Instance, select **Start** from the **Operation** dropdown menu. From that, click the blue **Build** button.
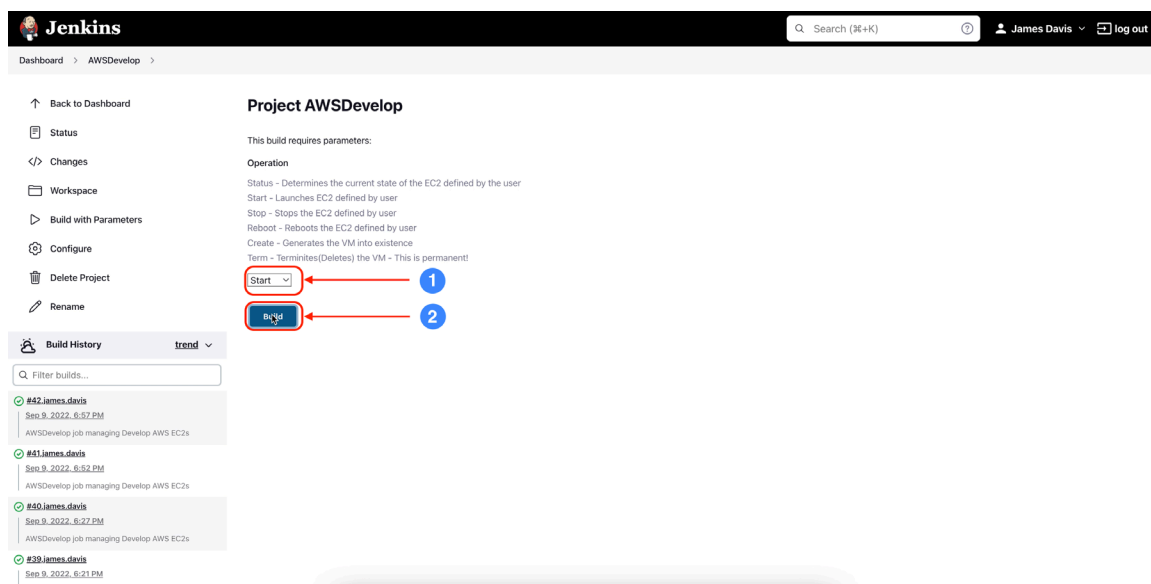
6. Once you click the **Build** button, your job will be submitted into the Jenkins queue. Your **Start** job will always be the first one in the queue. Click on the progress bar of your **Start** job in the queue to view the status in the **Console Output** screen.

7. On the left-hand panel, select **Console Output**. The **Console Output** screen will show the status of your **Start** job.

8. *Starting your EC2 instance may take a while, so be patient and let it run until the job status indicates "Finished: SUCCESS."*

9. You are going next want to go back to the **Project AWSDevelop** screen to get the status of your build. Click on the **AWSDevelop** link on the upper left of the screen.

10. On the left-hand panel, select **Build with Parameters**.

11. Select **Status** from the **Operation** dropdown menu. From that, click the blue **Build** button.

12. Click on the progress bar of your **Status** job in the queue to view the status in the **Console Output** screen.

13. On the left-hand panel, select **Console Output**. The **Console Output** screen will show the status of your **Status** job.

14. Allow the **Status** job to complete until the job status indicates "Finished: SUCCESS."

    Once complete, locate the header in the Console Output log indicated by Owner/State/ImageId/Ip Addr/InstanceId/Type/Purpose/PublicIp.

    Note and write down the **PublicIp**; This will be the IP address you will use to login remotely to your VM.

15. At this point, you can log out of Jenkins and proceed to Section 3. Remote into Your VM.

Section 3.   Remote into Your VM

1. You will need a remote desktop client.

   For Windows:
   The `Remote Desktop Connection` client is provided as part of the Windows operating systems. Use the Windows Start menu and search for "remote" and the `Remote Desktop Connection` client is usually the first application listed.

   For macOS:
   macOS users will need to download/install the `Microsoft Remote Desktop` app from the Mac App Store. In the Mac App Store do a search for "Microsoft Remote Desktop."

2. Launch remote desktop client. Enter the IP address (`IPAddr`) you wrote down in Step 14 from Section 1. or Section 2.

   a. On Windows

        1) Populate the IP address in the `Computer` field:
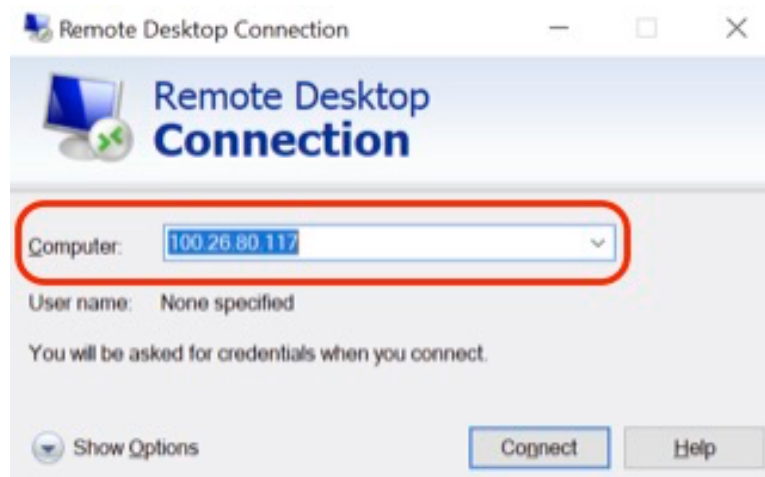
        2) Click `Connect`.

        3) Enter your login information. Your username is `develop\firstname.lastname`, and the password is the same you used to get into the Jenkins server in Section 2. , Step 2 You may need to select `Use a different account` to login using the develop domain.
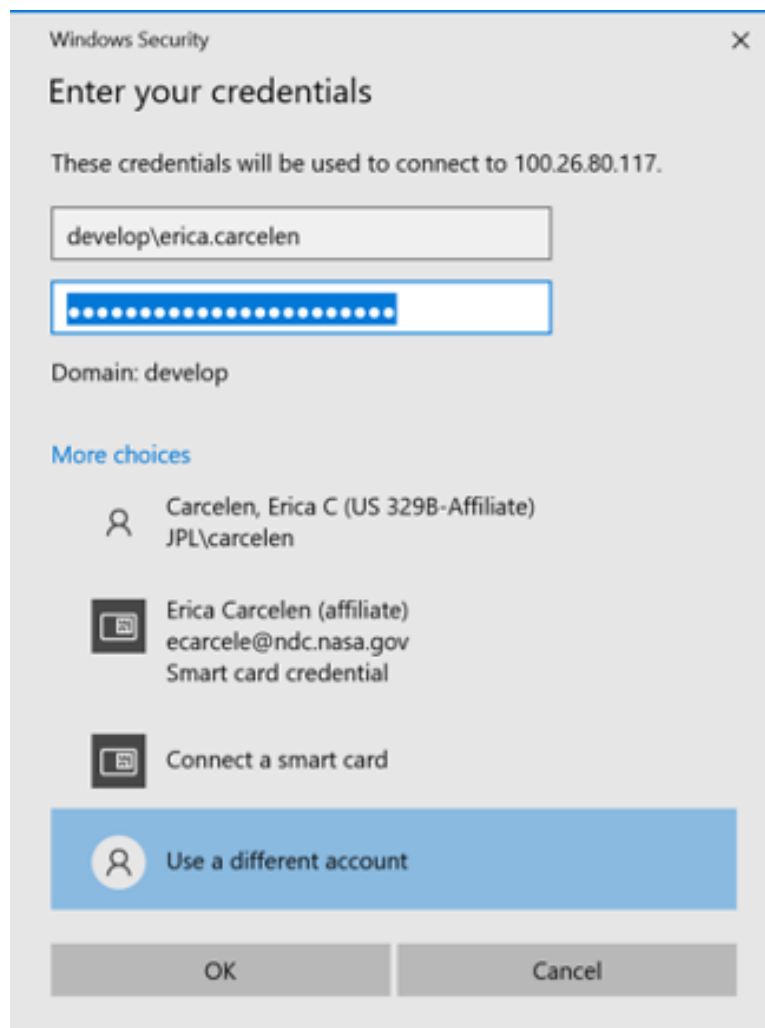
  4) Click **OK**.

b. On macOS:

  1) Populate the IP address in the **PC Name** field:
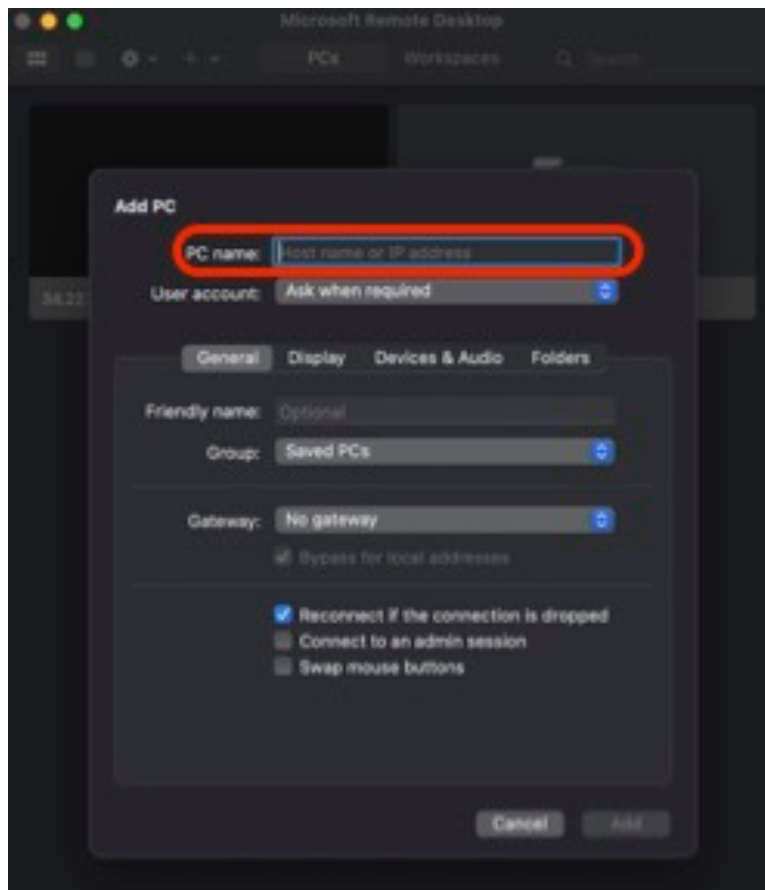
2) Click the `Add` button, then double-click on the PC you just added.

3) You may get a pop-up about the validity of certificate like the example below. Click `Continue`, as this is normal for remote desktop connections to EC2 instances.
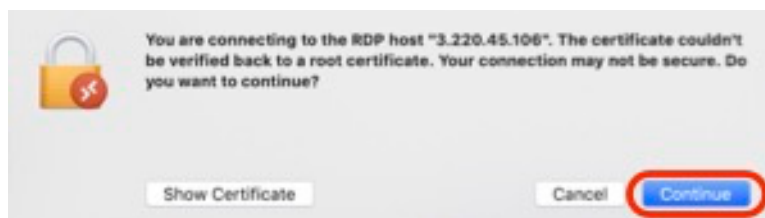
3. You should be greeted with your VM desktop!
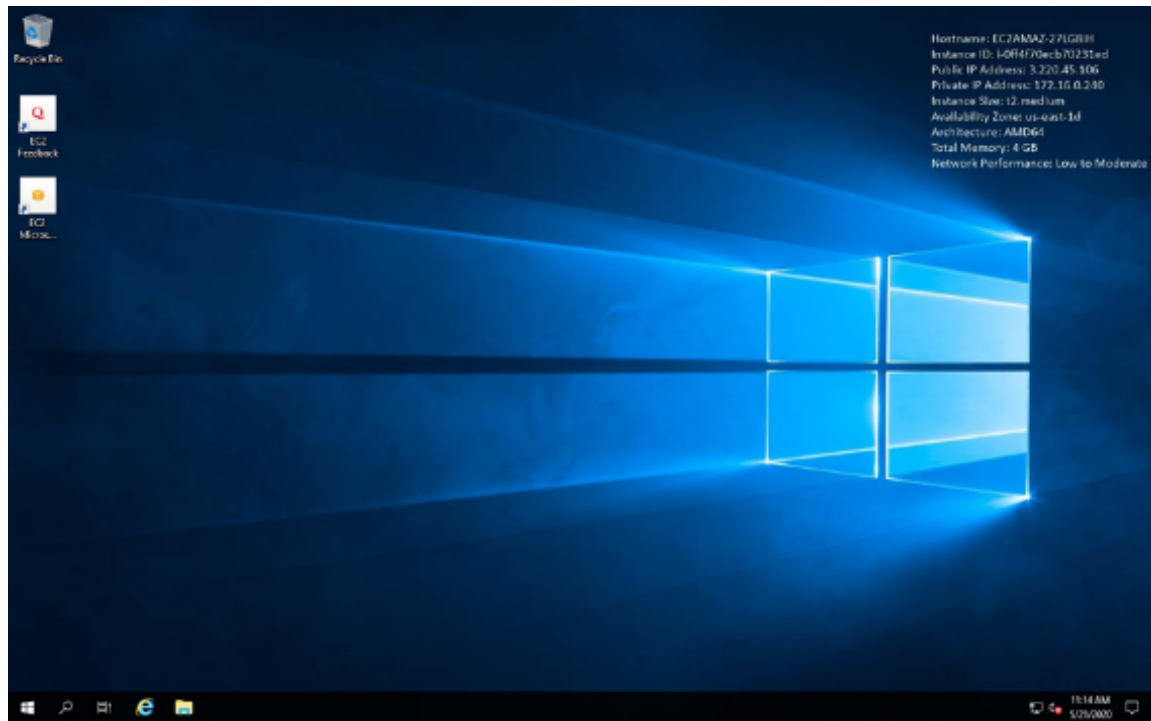
Section 4. Shutting down your VM

You **MUST** shut down your VM and terminate your EC2 every single time you are done using it! Failure to do so will waste energy and money on an unused virtual machine.

1. If you are remotely logged into your VM, bring up the `Start` menu by pressing the Win key on your keyboard or by clicking the Win button on the lower left corner of the Windows desktop, then click the `Power` button, and then click `Shut down`.
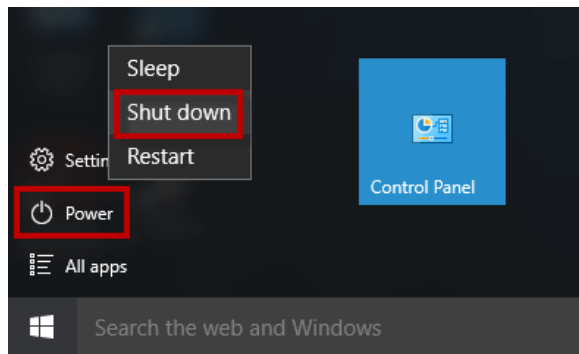
   Wait until your VM completely shuts down and returns to your remote desktop client.

2. If you are not currently logged in to Jenkins, you will need to log back into the Jenkins server: https://jenkins.developprogram.org/

Welcome to Jenkins!

| Username |
| Password |
| Sign in |
| ☐ Keep me signed in |

3. Click on `AWSDevelop` to navigate back to the DEVELOP AWS project page.
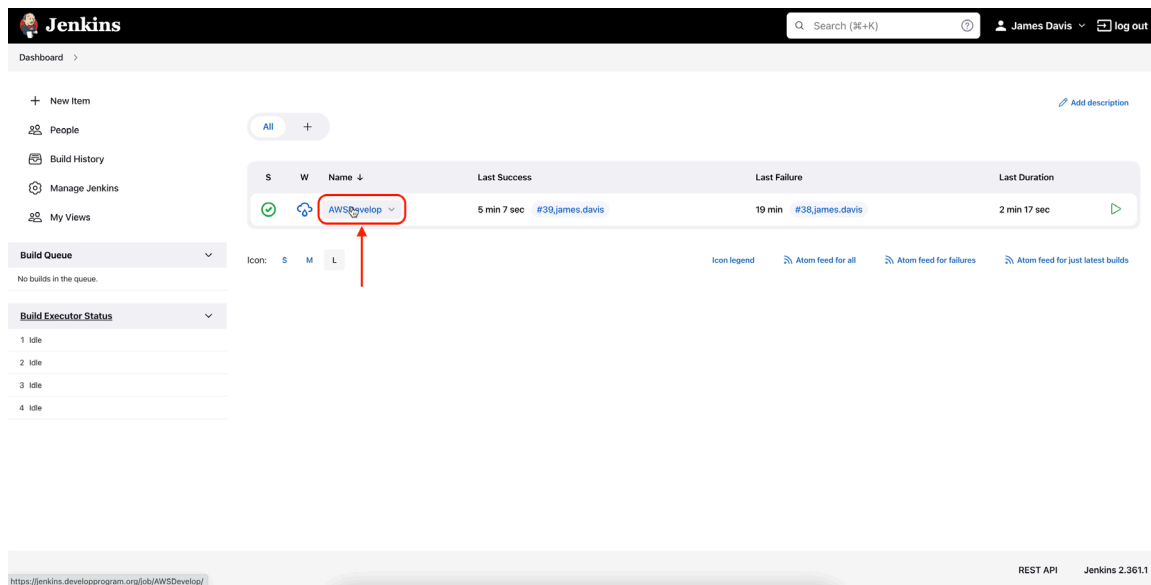
4. Select **Build with Parameters** on the left-hand side panel.
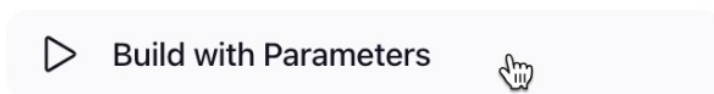
5. In the **Operation** dropdown menu, select the option **Stop** and click the blue **Build** button.
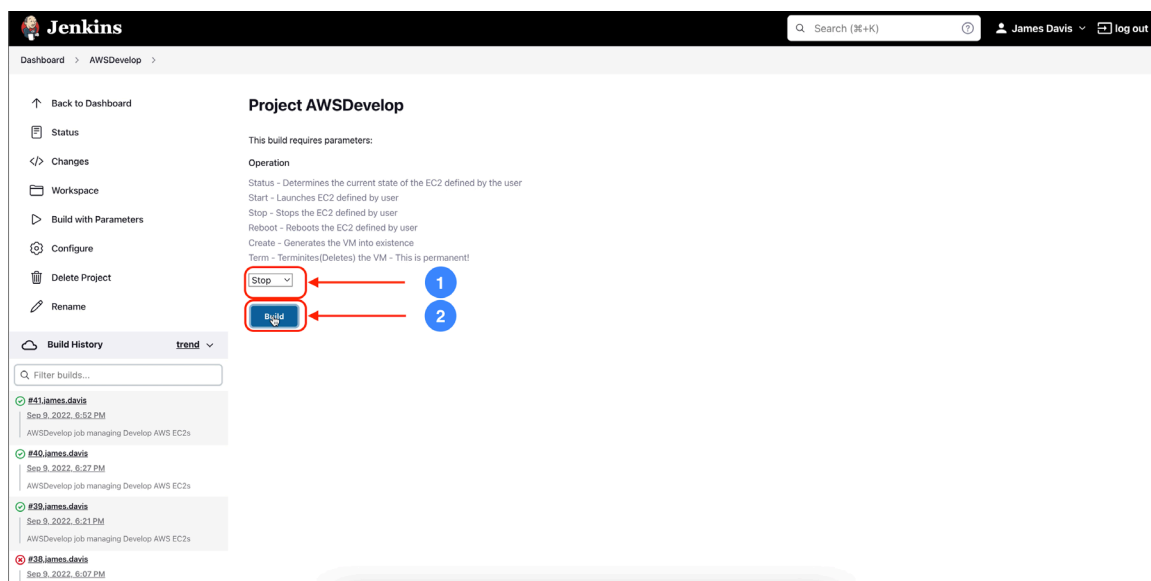
6. You can view the progress of your **Stop** job in the bottom left-hand panel under **Build History**. Your job will always be the first one listed.
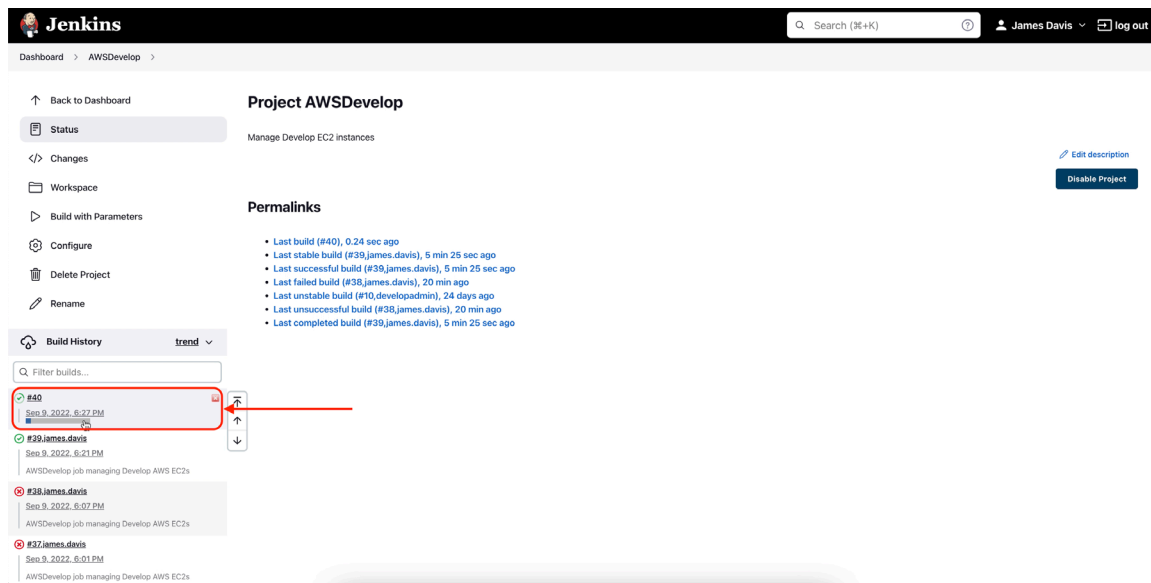
7. Click on the progress bar of your `Stop` job in the queue to view the status in the `Console Output` screen.
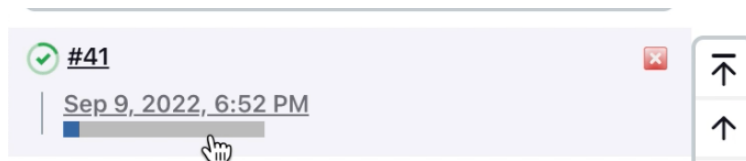
8. On the left-hand panel, select `Console Output`. The `Console Output` screen will show the status of your `Stop` job.
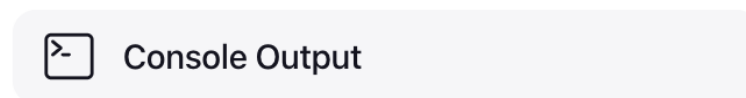
9. The Console Output screen will show the status of your `Stop` job.

   Stopping your EC2 instance may take a while, so be patient and let it run until the job status indicates "Finished: SUCCESS."

10. Once you see the *"Finished: SUCCESS"* in the `Console Output` log, you know that your EC2 has stopped, and you are good to log off for the night. At this point, you can log out of Jenkins.

Section 5.   Storing Files

At DEVELOP, there are several resources available to store your files, which are detailed below. Discuss these options with your Fellow, who may have preferences for the node or suggestions for what would work best for your team!

<u>**Local Storage (C:)**</u>

The `C: drive` on your virtual machine is your local drive, meaning any files stored here are only accessible on your virtual machine. If you start running low on space in your `C: drive`, work with your Fellow to request an increase in space from IT. Files stored on your `C: drive` will be lost at the end of the term, so make sure you have sent files to your Fellow or uploaded them to the shared drive. These should only be files that cannot be re-downloaded or files that are being sent to your partners. Use a `README` to document your folder's content and structure!
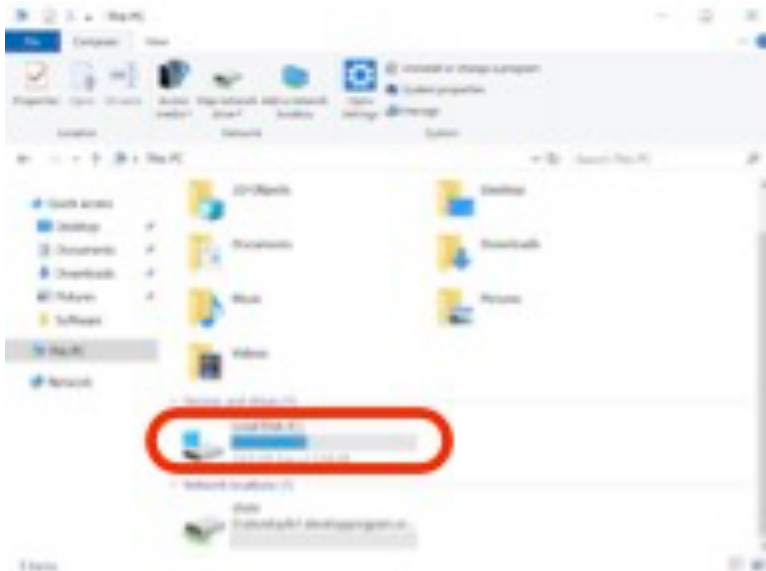
<u>**Shared Drive (D:)**</u>

The program's shared drive is accessible through the Virtual Machine and is already mapped to the `D: drive`, no additional steps are needed to connect to it. Because this folder is accessible by everyone in the program, you can use it to transfer files among your team members or back-up your work. Please keep your files organized in a team folder with the following naming format:

`YearTerm_Node_ProjectShortTitle`

Remember that this is a shared space for the program, so be mindful of the amount of space you are using.

If you have several large files that need to be shared with your team members, contact IT about options to set-up a server for easy transfer!
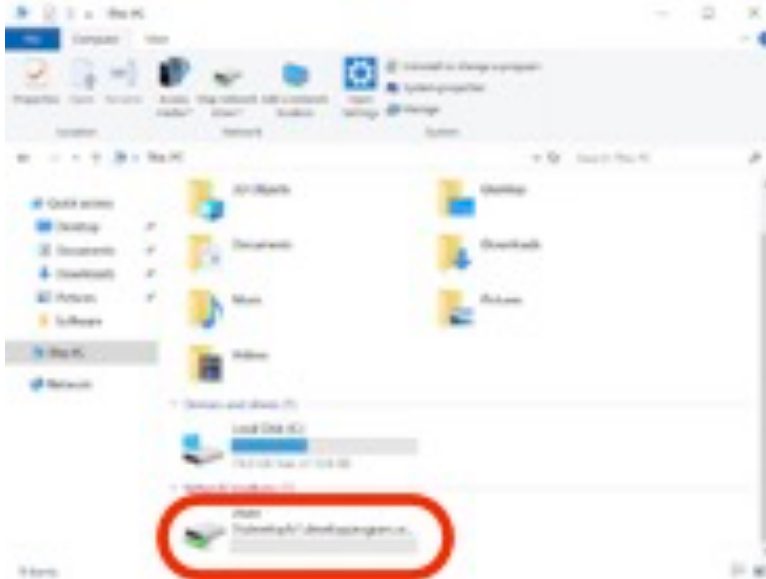
AWS WorkDocs

AWS WorkDocs is a cloud-based file-sharing platform that can be accessed both inside and outside the Virtual Machine. It is a great option for teams that are not requesting Virtual Machines and need space to share files outside of Office 365.

Please access AWS WorkDocs through the browser only, <u>do not download</u> and install the app on your virtual machine.

1. Navigate to the AWS WorkDocs page:

    https://developprogram.awsapps.com/workdocs/loginv2/index.html#/emailSelect?sitename=developprogram

2. Login with the username (`firstname.lastname`) and password that Jim Davis sent you. The username should be formatted as `username@develop.environments.com`. It should be the same username and password used for Jenkins and GitLab.
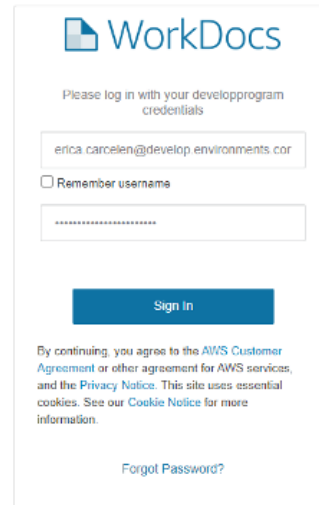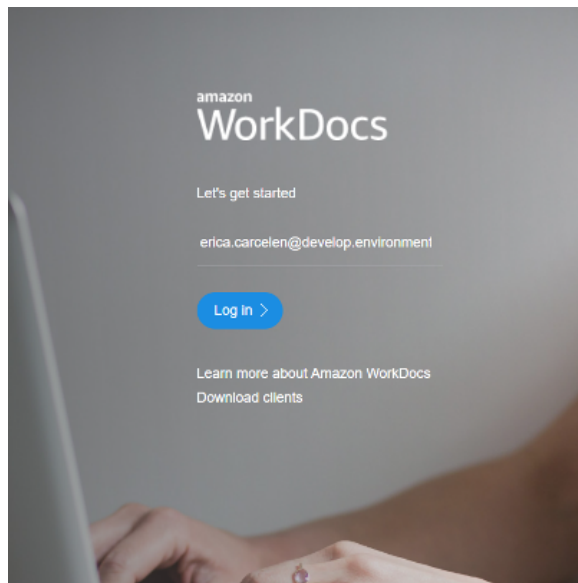
3. The home page defaults to `MyDocs`, which are the folders and files you have uploaded to AWS WorkDocs. If this is your first-time logging in, there should be no files since you have not added any files yet! Navigate to Collaborate to see folders shared among other users in DEVELOP's AWS WorkDocs and `Activity` to see recent actions taken in your files.
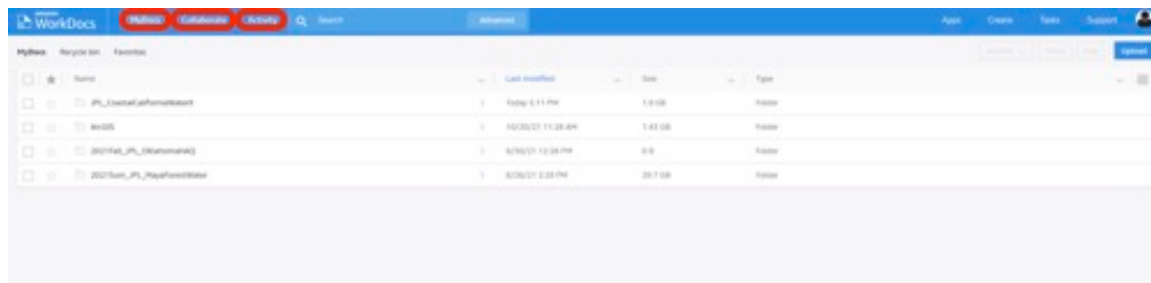
4. To create a new folder, use the `Create` button at the top right. Once you're ready to upload files to your folder, navigate to the folder and use the blue `Upload` button to upload folders or individual files.
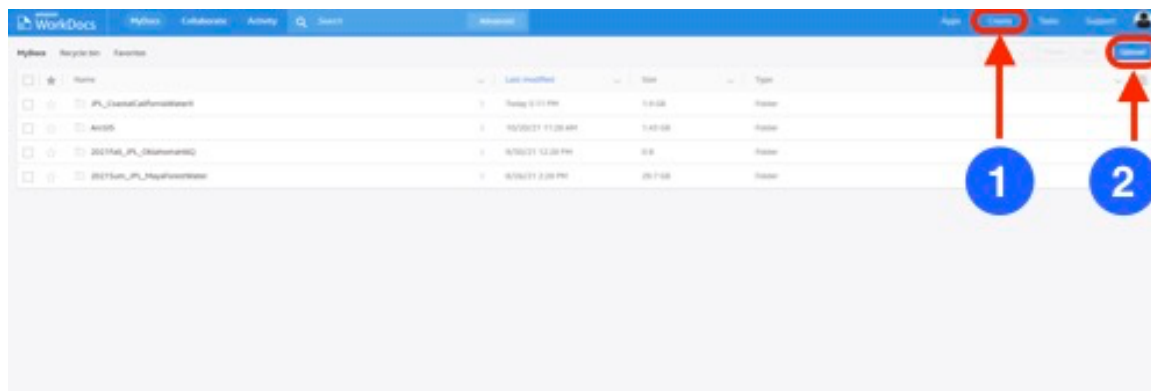
5. If you'd like to share the folder with your teammates via AWS WorkDocs, select the file or folder you'd like to share. Click the **Share** button that becomes available on the top right, near the blue **Upload** button.

6. You can create a link by selecting the Share a link option and selecting the appropriate option.
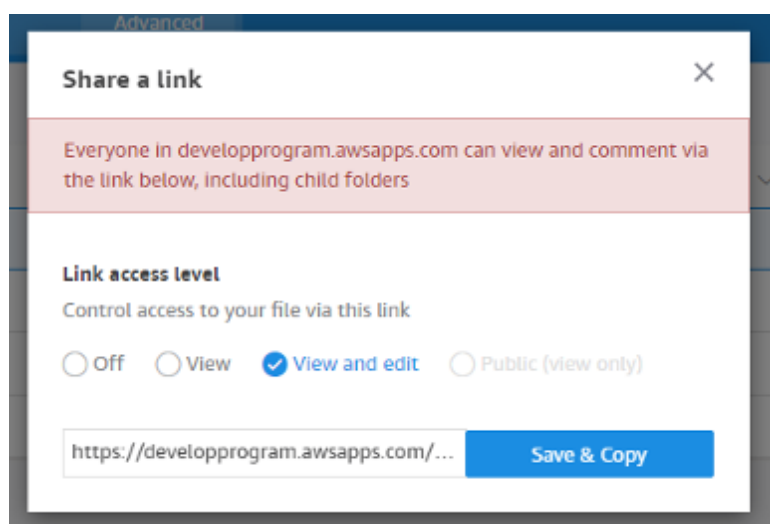
You can also add people to the folder by selecting Invite people and entering their emails (typically **firstname.lastname@ssaihq.com**). Any folders shared with you will show up in the **Collaborate** tab.
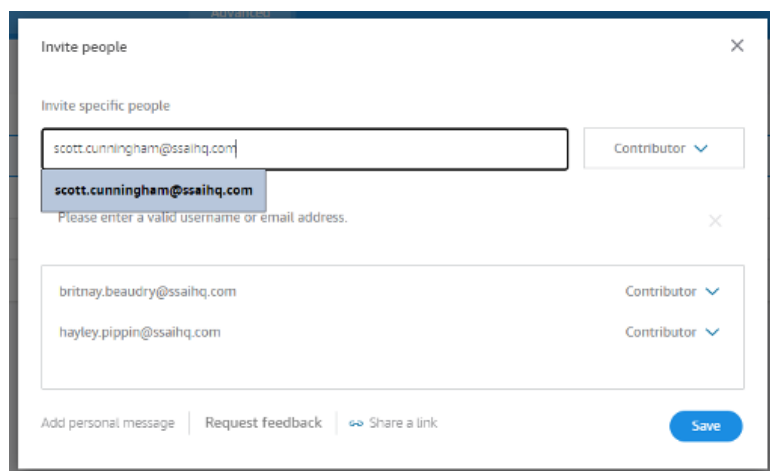
Note: Only those with access to DEVELOP's AWS WorkDocs can be added or view these folders. If you need to share a folder or file externally, discuss any limitations and options with your Fellow and IT.

7. When you select a file or folder, the `Actions` drop-down menu also becomes available. Use this drop-down to download, edit, or delete the item.
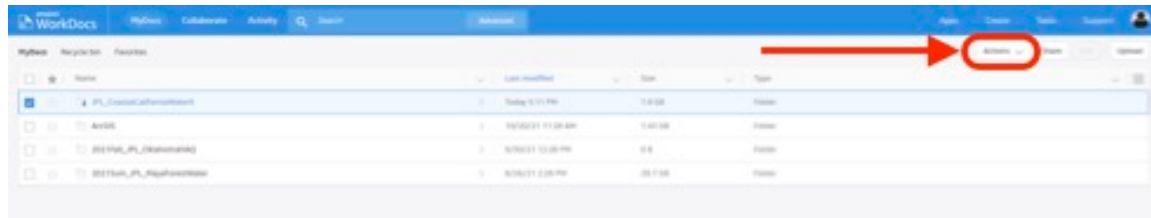
Section 6.   SMCE Passwords

For security reasons, your SMCE account password expires every 90 days. *Please keep track of when your password expires, and remember to change it before its expiration date!*

1. If you did not request a VM for the term and needed to change your password before it expires to access GitLab or AWS WorkDocs, send a message to Jim Davis to request that your password be reset.

2. To check your password expiration date before your password expires, follow the steps in the subsection `Check your password expiration`.

3. To change your password before your password expiration date, follow the steps in the subsection `Changing your password`.

<u>Check your password expiration</u>
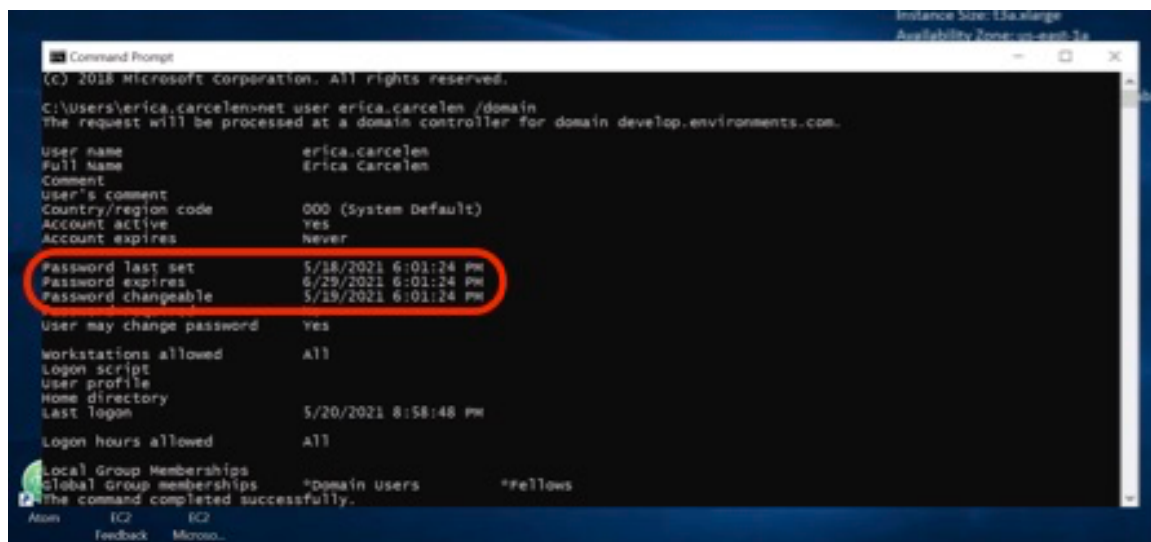
When you remotely log into your VM, a dialog box should appear within 10 days of an expiration with a warning that your password will expire soon. You can, however, check the expiration date anytime.

1. To check the expiration date of your password, open the `Command Prompt` application on your VM by pressing the Windows key, and on the search field, type "command prompt" to locate the `Command Prompt` application.

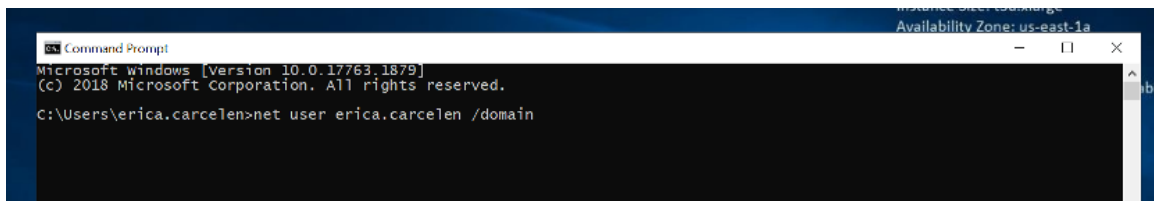   In the `Command Prompt` application, type `net user <your_login> /domain` replacing `<your_login>` with your AWS username (typically `firstname.lastname`) and hit `Enter`. Make sure there is a space between your `<your_login>` and `/domain`.

2. A report will then show up with information about your account. Refer to the "Password expires" field for the exact day your current password is set to expire.

Changing your password

If you attempt to remotely log into your VM **after your password has expired**, you will not be able to log in and will need to message Jim Davis james.a.davis@ssaihq.com, to reset your password.

To update your password, you must connect to the `Change Password Host` (IP address 52.203.105.43) from your VM between NOON and 6 PM Eastern (9 AM and 3 PM Pacific). The `Change Password Host` will only be available at those times and can only be used by one person at a time, so be patient during peak hours.

1. To change your password before its expiration date, open the `Command Prompt` application on your VM by pressing the Windows key, and on the search field, type "command prompt" to locate the `Command Prompt` application.

2. In the `Command Prompt` application, type `osk` and hit `Enter`.

3. A virtual keyboard will appear. On your local keyboard, press the `Ctrl-Alt` keys (these should turn blue on the virtual keyboard) and then click on the `Del` key on the virtual keyboard.

    Make sure you click on the `Del` key on the virtual keyboard and __do not__ press `Del` on your local keyboard. If you press `Del` on your local keyboard instead of clicking `Del` on the virtual keyboard, you may change the password on your local machine, not the VM!
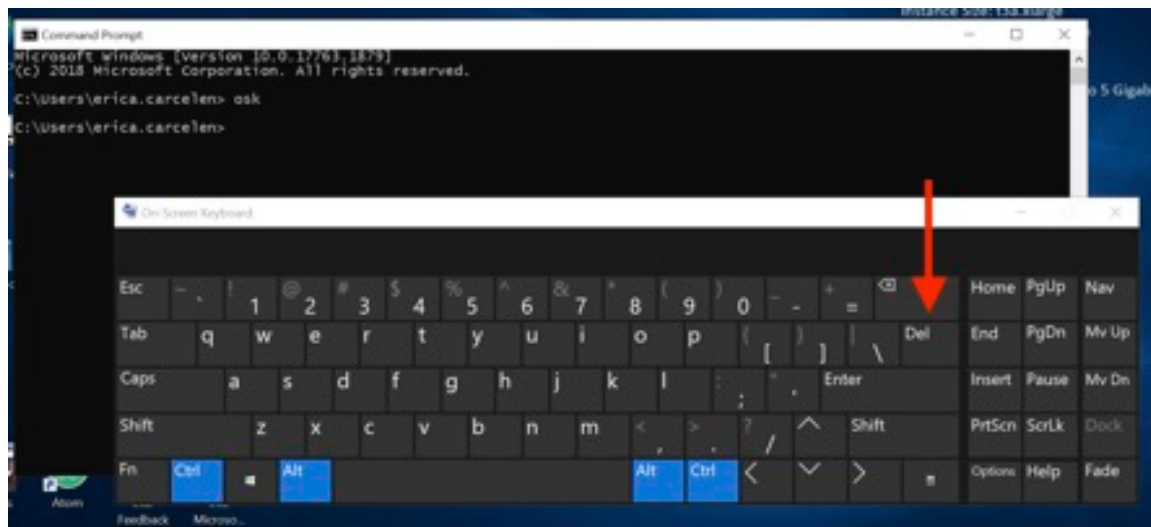
4. Select **Change a password** and fill out the sections accordingly. Passwords must meet the following requirements:

   • at least 15 characters
   • 1 upper case
   • 1 lower case
   • 1 special character,
   • 1 numeric character
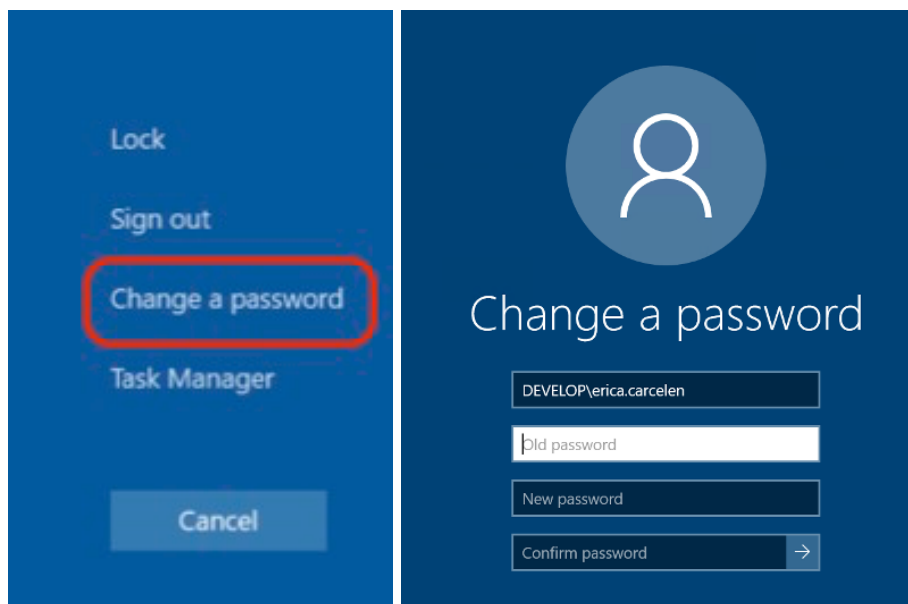
   The same character cannot be repeated more than 3 times in a row.

5. Once you click on the **Confirm password** button, your password is set for the next 90 days!