**Launching a Custom Image Windows AppStream Session.**

**Summary:-** We are creating this for the End User to Access all the UNO Bank websites & Tools inside this Virtual AppStream Session. Basically, whatever task need to be done respective to UNO Bank, need to do inside the AppStream.

**The Main Advantage of using the AppStream is:-**

* The User is having very less Access in the User Environment. They cannot install any of the Software or make any changes in windows setting.
* Admin can control everything from the admin console.
* If there is a requirement to install nay of the application in user console. Then they have to make a request to the Administrator.

**Top 6 Key Functionality used in AppStream.**

**Stack**

A *stack* consists of an associated fleet, user access policies, and storage configurations. You set up a stack to start streaming applications to users.

**Fleet**

A *fleet* consists of fleet instances (also known as streaming instances) that run the applications and desktops that you specify. Note that one user requires one instance.

**Image builder**

An *image builder* is a virtual machine that you use to create an image. You can launch and connect to an image builder by using the AppStream 2.0 console. After you connect to an image builder, you can install, add, and test your applications, and then use the image builder to create an image. You can launch new image builders by using private images that you own.

**Streaming instance**

A *streaming instance* (also known as a fleet instance) is an EC2 instance that is made available to a single user for application streaming. After the user’s session completes, the instance is terminated by EC2.

**User pool**

Use the *user pool* to manage users and their assigned stacks.

**Auto scaling rules**

*Auto scaling rules* are schedule-based and usage-based policies that you can apply to an Always-On or On-Demand fleet to automatically manage the number of streaming instances available for users to stream from.

Graphical user interface, application

Description automatically generated

If you open a AppStream, you will see this option on the left-hand side.

**How to create a Custom Image with Image builder.**

When you click on the **“Images”** on the lefthand side of the menu. You will see two options coming up. Below is the screenshot.

A picture containing box and whisker chart

Description automatically generated

* In Image registry you will find all the images which are Public to all and the Images which you have created and for that the Visibility is Private and the owner is also the one who is creating the image.
* To create a New Custom Image, you need to click on **“Image Builder”.**
* Click on the  Button.
* Select the Base Image which you want. Like you want Windows or Amazon Linux, it will also depend on your work which image you want to choose. For me I just want to have Windows server with the latest version. So, I selected the below one.

Graphical user interface, text

Description automatically generated

* Click on Next.
* Give the Image Name and Display Name. (Both can be same)

Graphical user interface, text, application, email

Description automatically generated

* Select the Instance Type.

Table

Description automatically generated with low confidence

* As I told you I just have to add some application and use it. So, I’ll use the General-Purpose Instance.
* Click On Next.
* After that you need to configure the VPC for the Image.

Graphical user interface, text, application, email

Description automatically generated

* If you don’t have the VPC just Create it by clicking on the Link given and select it.
* Review all the setting which you have selected and click on .
* Go back to the Image Builder page and see the Image is listed out there.



* It will take 10 to 15 min to launch an image.
* After it Status is Running. Click on the image name.
* You will see the these option on the right top corner.
* Click on Connect.
* It will open a New Page and you will be connected to the Image which you created.
* On the very first time you will get the Option to select before connection to the image.

Graphical user interface, text, application

Description automatically generated

* So, with this console we can install any of the software & Setting in the Admin side and replicate it into the Users side as well.
* We have an application named Image Assistant on the admin side.
* If you need to add any kind of application which user can also use. For that you have to add it in the Image Assistant.

**How to add an application in the image and test it out for the end user.**

* Click on the **Image builder** which you created and click on **“Connect”.**
* Once you enter the Image builder, a screen will pop up and ask you in which user you want to enter.
* Click on Administrator side.
* You will see the Image Assistant Application on the desktop itself. Open it.
* Whatever the Application you want to download like Chrome.
* In the Windows Server we have Firefox by default. Will open that and download the Chrome Setup file.
* After downloading it. Double Click on the setup file and install it in the Administrator side.

**Note:- While installing any of the Software make sure to copy the path or just remember the path where it is installing.**

**For Ex:-** Suppose If I’m installing **“Win Merge”** in the admin side. Below is the Screenshot in which you can see the Path where the application is installing.

All the Application is installed in **Program Files** & **Program Files(\*86).**

Graphical user interface, text, application, email

Description automatically generated Table

Description automatically generated with medium confidence

* After you Install the Application, in my case I’m installing WinMerge.
* Open the Image Assistant Application.

Graphical user interface, text, application

Description automatically generated

This application is already installed on the admin side as well as the user side . If you are staring fresh, you might see it bank.

* Click on the **“+ Add App”.**
* It will open the File Explorer and in that you need to open the same path which you have see while installing the software.

Graphical user interface, text, application, email

Description automatically generated

* After Click on Open. A New window will open. Below is the Screenshot.

Graphical user interface, text, application, email

Description automatically generated

* Click On Save. After this you can see your application will be added in your application list.

Graphical user interface, text, application, email

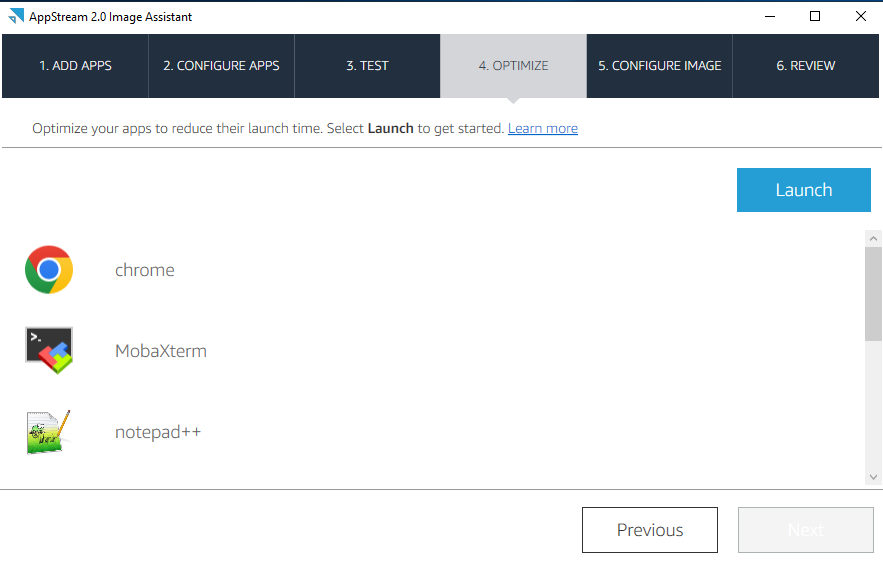
Description automatically generated

* Click On Next.

Graphical user interface, text, website

Description automatically generated

* If you read the 5 Steps given on the above Screenshot.
* It is asking to click on Switch user then go to the **Template User**  and open the application which you have installed and use it as you do normally.
* Then again open the Image Assistant in the **Template user**. And Click on the Switch user.
* And now you need to click on the **“Administrator User”.**
* You will see the same Image Assistant opened in the admin side.
* You can also see the Save setting button on that. (Refer the above Screenshot.)
* Click On the **Save Setting** button. It will take some time to save the setting.
* Click on **Next** after the setting is saved.
* And now again you need to click on **Switch User** and now we ned to click on **“Test User”** and open the application which you have installed and use it as you do normally.
* Then again open the Image Assistant in the **Test user** and Click on the Switch user. Now you are back to **Administrator user.**
* After you have tested the application in Template & Test user and come back to the Administrator user.
* Click On **Next**. Now you will see the below screen.



* You need to click on the Launch button. If will open all the application one by one.
* There will be one dialog box which you need to click on when an application will open.

Text

Description automatically generated with medium confidence **Graphical user interface, text, application, chat or text message

Description automatically generated** Graphical user interface, text, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

* Click on **Continue** if Chrome is opened. Same will happen will all the applications.
* After you opened all the application you need to click on **Next.** Then you will see below screen.

A picture containing shape

Description automatically generated

* Now you need to give the name of the Image. (Refer the Below Screenshot)

Graphical user interface, text, application

Description automatically generated

* You can give any name. I have given **Demo.** In Description you can add the application name or whatever changes you have mage in this new image.
* Click On **Next**.

Graphical user interface, text

Description automatically generated

* Click On **Disconnect and Create Image**.
* Now it will disconnect your session and it will create an image for you.

Graphical user interface, application

Description automatically generated

It will take approx. 15 min to create the Image.

* After the Creation of Image will be done. The status will automatically change to **Stopped.**
* You need to **Start** it manually.

Graphical user interface, application

Description automatically generated

If You open the Demo Image, you will see the Below Details.

A screenshot of a computer

Description automatically generated

**How we can Create a “*Fleets”* in AppStream.**

Graphical user interface, application

Description automatically generated

Click on the Fleets tab. On the right side you will see the Button of **Create Fleet.**

***Note:-*** *Whenever you are making any changes in Fleet. You must have to Restart the Fleet. To Incorporate all the Changes.*

* Click on Fleets, after that click on **Create Fleet**. A picture containing text

  Description automatically generated
* After that you will see Fleet Type.
* Select any of the Fleet type which suits your scenario. Click on Next.

Graphical user interface

Description automatically generated with medium confidence

Graphical user interface, text, application, email

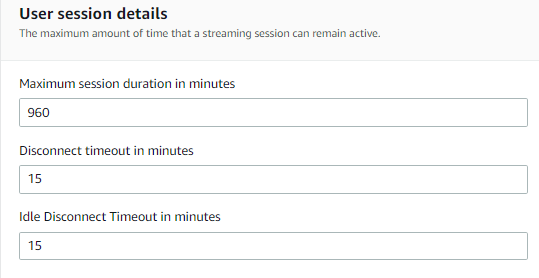
Description automatically generated

Give the name of the Fleet and the same name will be in Display Name.

Table

Description automatically generated

Select the Instance type as per the use case. You can also select the Dropdown and select the higher instance type with more vCPU’s & Memory.



The Value which is By Default. If you have to change the Number, you can change it.

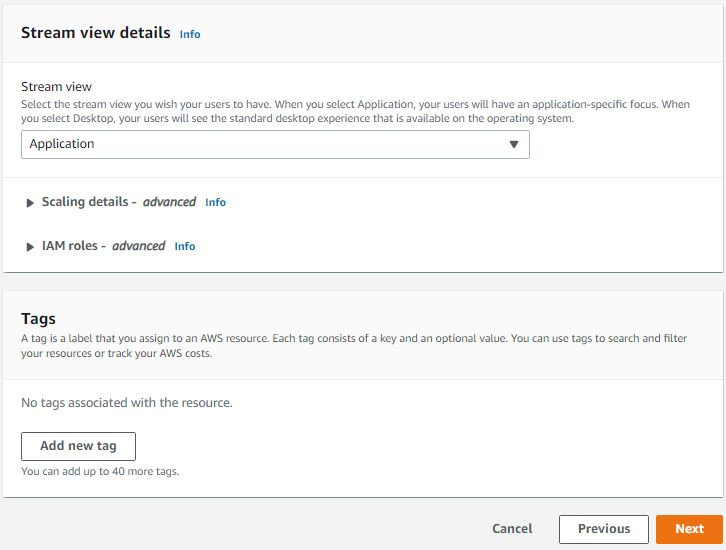
Disconnect time out is 15 Minutes. It means your session will be disconnected after 15 min if you are not using it.

Same with Idle Disconnect Timeout.

Graphical user interface, text, application, email

Description automatically generated

You need to mention the Minimum & Maximum count of how many users will be login in the AppStream. By Default, 1 Minimum & 4 Maximum user can login in AppStream.



In Streaming view, you will see 2 options.

Application, Desktop

Rest All the Setting will be By Default.

Click on Next .

Graphical user interface, text, application, email

Description automatically generated

This will be next page here you can see the multiple Public & Private images.

The One which I selected is the one which we created with the help of image builder.

Private once will be the one which you have created.

Public image is already there by default.

Total 21 images are there.

Graphical user interface, application

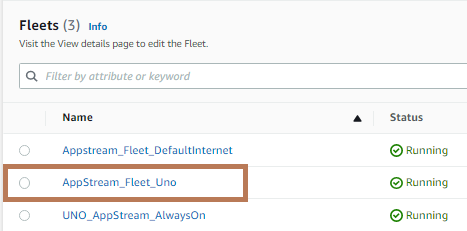
Description automatically generated

**Note:-** If you need to enable this Default internet access, then for this you have to create a public subnet connected to the bastion host.

Here you need to select the VPC and the subnets and the Security Group.

If it is not created, then create it and refresh the with this button.

* Review all the setting which you have made and click on **Create Fleet**.
* Your Fleet will be upend running in sometime. (Refer below screenshot)



**How to create *“Stack”* in AppStream.**

Graphical user interface, text, application, email

Description automatically generated

When you click on the “Stack” on the lefthand side. You will see this Stack Details page. Where you need to give the name of the Stack.

And rest of the things will be optional.

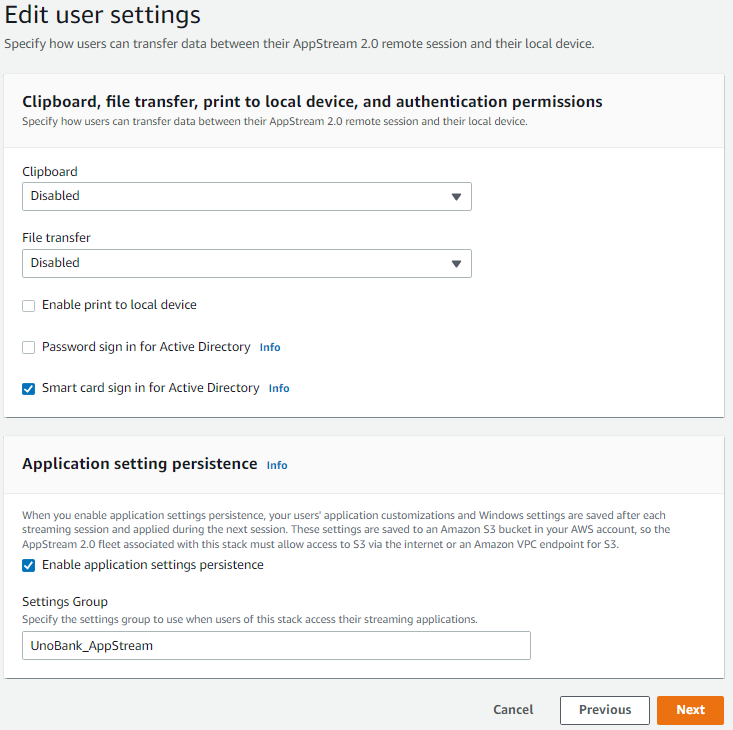
Click on Next.

Graphical user interface, text, application

Description automatically generated

By Default, this **Home folder** will be enabled. It will create a new S3 bucket for storing the data. If you want, you can disable it any time after Creating the stack as well.

If you want an additional storage attached, then you can add **OneDrive or Google drive**. You just have to add the domain for OneDrive, and it will be there in your AppStream session.



If you want to disable the copy paste access from outside into your AppStream Session, then you can Disable the Clipboard.

There is an option to upload and download file from out the AppStream Session then it can happen but if not then you can disable it in Stack.

Some More settings are their which you can enable or disable according to your requirement.

Click On Next.

* Review all the setting which you have made. If and changes required, then do it if not then Click on
* After creating the Stack. If you feel like you have to change some settings, then just go inside, and click on edit option in which section you need to edit.

**How Scale-In Policy and Scale-Out policy works in Fleet.**

**Short Summary:-** Scaling-In & Scaling-Out will help us to automatically spin up the instances if the number of logged in user are increasing and vice-versa if the users are logging out then it can remove the instance. Ultimately, we will save the cost and only Pay only for the time you have used that instance.

We have some **Fleet capacity Naming conventions** which will be used in setting up the Policy.

1. **Minimum Capacity**

The minimum number of fleet instances. The number of fleet instances can't be below this value, and scaling policies will not scale your fleet below this value. For example, if you set the minimum capacity for a fleet to 2, your fleet will never have less than 2 instances.

1. **Maximum Capacity**

The maximum number of fleet instances. The number of fleet instances can't be above this value, and scaling policies will not scale your fleet above this value. For example, if you set the maximum capacity for a fleet to 10, your fleet will never have more than 10 instances.

1. **Desired Capacity**

The total number of fleet instances that are either running or pending. This value represents the total number of concurrent streaming sessions that your fleet can support in a steady state. To set the value for **Desired Capacity**, edit **Fleet Details**. We do not recommend changing the **Desired Capacity** value manually when you use **Scaling Policies**.

If the value of **Desired Capacity** is set below the value of **Minimum Capacity** and a scale-out activity is triggered, Application Auto Scaling scales the **Desired Capacity** value up to the value of **Minimum Capacity** and then continues to scale out as required, based on the scaling policy. However, in this case, a scale-in activity does not adjust **Desired Capacity**, because it is already below the **Minimum Capacity** value.

If the value of **Desired Capacity** is set above the value of **Maximum Capacity** and a scale-in activity is triggered, Application Auto Scaling scales the **Desired Capacity** value down to the value of **Maximum Capacity** and then continues to scale in as required, based on the scaling policy. However, in this case, a scale-out activity does not adjust **Desired Capacity**, because it is already above the **Maximum Capacity** value.

**Scaling Policy Metric**

1. **Capacity Utilization**

The percentage of instances in a fleet that are being used. You can use this metric to scale your fleet based on usage of the fleet. For example, **Scaling Policy Condition**: "If Capacity Utilization < 25%" perform **Scaling Policy Action**: "Remove 25 % capacity".

1. **Available Capacity**

The number of instances in your fleet that are available for user sessions. You can use this metric to maintain a buffer in your capacity available for users to start streaming sessions. For example, **Scaling Policy Condition**: "If Available Capacity < 5" perform **Scaling Policy Action**: "Add 5 instance(s)".

1. **Insufficient Capacity Error**

The number of session requests rejected due to lack of capacity. You can use this metric to provision new instances for users who can't start streaming sessions due to lack of capacity. For example, **Scaling Policy Condition**: "If Insufficient Capacity Error > 0" perform **Scaling Policy Action**: "Add 1 instance(s)".

**Where we can Specify the Scaling Policy and add the Fleet capacity.**

* When you go in the **AppStream UI**. On the Left hand you can see the list of items there you can see the name **Fleet**. Click on that.
* You will see the running fleets over there.
* Click on the Fleet in which you have to specify the Minimum & Maximum capacity and Scaling policy as well. (Refer the below Screenshot)

Graphical user interface, application

Description automatically generated

Here you can edit the Scaling-In & Scaling-Out policies.

Click on this edit button to edit all the things.

Here you can set the Minimum capacity of fleet & maximum Capacity of Fleet.

* On the same page you will see the **Fleet Usage,** In Which you can see the below graphs side by side.

Chart

Description automatically generated with low confidence

* Above is the **Capacity Utilization** graph in which you can see on what percentage it is running. As of now it is on **0%.**
* With the help of Capacity utilization, you can set the Scaling Policy like **if Capacity utilization is greater that equal to (>=) 50% then add 2 instances.**
* Below is the real time data of the capacity you have specified in the **Fleet configuration**.

**Graphical user interface, application

Description automatically generated with medium confidence**

This no may change when users will login and logout. If scaling is also enabled, then Desired capacity will also increase.

1. **Actual Capacity:-** The total number of instances that are available for streaming.

**ActualCapacity = AvailableCapacity + InUseCapacity**

1. **AvailableCapacity**:- The number of idle instances currently available for user sessions.

**AvailableCapacity = ActualCapacity – InUseCapacity**

1. **InUseCapacity:-** The number of instances currently being used for streaming sessions. One **InUseCapacity** count represents one streaming session which is running.
2. **PendingCapacity:-** The number of instances being provisioned by AppStream 2.0. Represents the additional number of streaming sessions the fleet can support after provisioning is complete. When provisioning starts, it usually takes 10-20 minutes for an instance to become available for streaming.
3. **DesiredCapacity:-** The total number of instances that are either running or pending. This represents the total number of concurrent streaming sessions your fleet can support in a steady state.

**DesiredCapacity = ActualCapacity + PendingCapacity**

**While using the Auto Scaling of Streaming instances, it’s highly recommended that we don’t change the Desired state manually.**

**Now we’ll see how the “Auto Scaling-Out” policy will work.**

* Before stating the Scaling thing, I want to discuss some of the points which is related to Auto Scaling.
* As of now we already set the fleet size of (Refer the Fleet Configuration Screenshot above)
  + **Minimum Streaming Session is 2**
  + **Maximum Streaming Session is 5**
  + **Desired State is also at 2** but it will automatically change when Users will login and Auto-scaling happen**.**
* **Important note :-** This Minimum and Maximum value will act as a boundaries for the fleet. It means at any point of time fleet size can never fall below 2 instances which is Minimum and can never go beyond 5 instances which is Maximum.
  + **Refer the YouTube link to understand how scaling works:**- <https://www.youtube.com/watch?v=MDiQgqolZHA>
* Now if you can see the below image in which you can see **Available Capacity is 2.** This is capacity of how many users can login in total. As we have set the Minimum capacity as 2 that’s why.
* It can be increased with the help of Scaling.

Graphical user interface, application

Description automatically generated

* Now if I login into AppStream then you can see then you can see the spike in Capacity utilization.

Graphical user interface, application

Description automatically generated

* On the above screenshot. You can see the **Capacity utilization** is at **50 %.** Why it is that so, it is taking out the percentage on the basis of the **Total Actual capacity**.
  + Suppose if Actual capacity is 100 and InUseCapacity is 50 means, 50 users is logged in AppStream. Then also the Capacity Utilization will be 50 %
* If you can see the major Change in Pending capacity, it is 2 as of now. Question is how, it happened?
  + We have set the scaling-out policy that. **If Capacity Utilization >= (greater than or equal to) 50% then add 2 instance.**

A picture containing text

Description automatically generated

**Note:- Scaling-Up the instance will take 10 to 15 min to get ready, till then it will show in “Pending Capacity”, after that it will move into “Available capacity”.**

**Graphical user interface, application

Description automatically generated**

**3 Streaming sessions** are in buffer state. If any user comes in and request for login, we have the buffer capacity to fulfil the requirements.

1 user is logged in at this point of time.

* On the above Screenshot, you can see the 2 Pending Instance is moved to Available capacity. Because it is ready to use.
  + Now the **Available Capacity + InUseCapacity = 4 Actual capacity**
* You can see on the left-hand side the **Capacity utilization** is gone below. It is at **25%.**
  + When the Actual capacity limits increase the then the capacity Utilization will decrease & vice versa.

===========================================================================================================================================================================

* Now I’m going to **login 2 more** user, let’s see what will happen.

Graphical user interface, application, Word

Description automatically generated

* You can see the **InUseCapacity** is 3. Because right now 3 Streaming sessions are running.
* You can see the Capacity Utilization is 75%. Let’s calculate, out of **4 Actual capacity** the **InUseCapacity is 3**. If you will take out the percentage of it, you will get 75% only.
* Now again you can see the **Pending capacity is 1,** because.
  + The Rule which we have created in Scaling-Out. **If Capacity Utilization >= (greater than or equal to) 50% then add 2 instance.**
  + Now the question is why it is adding 1 if it is mentioned to add 2 instances.
    - Because as I told you it cannot scale up above the **Maximum limit which is 5**. Already **3 Sessions** are running 1 is in buffer and 1 is in the pending state soon it will be in buffer state.
    - As Soon as the Available capacity will increase from 1 to 2 then automatically the **Capacity Utilization** will go down. (Refer the below Screenshot)

Graphical user interface, application

Description automatically generated

* Above screenshot is the example of Scaling-out & maintaining the Capacity Utilization in control.
* Now I’ll login all the 5 users then see how it looks like.

Graphical user interface, application

Description automatically generated

* As of now you cannot scale-out because you have reached the **maximum capacity** which you set.
* Now, suppose if the 6th users try to login what will happen. (Refer the below screenshot)

Graphical user interface, application, website

Description automatically generated

* It will say that No Streaming instance is available for your session. This error will come in 2 cases.
  + If you have reached the maximum limit which you have set. You can login if in case some body logout from their session.
  + The Other case is depending on the below screenshot.

A picture containing graphical user interface

Description automatically generated

* + Here you can see the InUseCapacity is 4, it means you have 1 more session left but with the help of Scaling-out the last session is in pending state and as I told it will take 10 to 15 min to get ready.
  + As soon as it is ready you can login.

**Note:- In most of the cases you will not face the above error because we will keep the buffer instance ready as much we can for login. It will be depending on the Scaling Policy which we will create.**

**Now we’ll see how the “Auto Scaling-In” policy will work.**

**Some Couple of issues/Question which I have faced while testing AppStream.**

1. **If a user closes their AppStream Session abruptly or closed the tab, then what will happen?**
2. **If we have Available capacity or buffer instance suppose 20, they it will be chargeable for 0.025?**
3. **If I have set the Minimum capacity of 100. Then it will also charge 0.025 if it is not in use?**
4. **If an instance is used for 20 mins, then it will be charged for 1 hour or only for 20 mins?**
5. **Are there any ways to scale up the instance faster. Right now, it is taking 10 to 15 mins.**

**How we can connect AppStream to CMD and what are the Commands to perform the task.**

* If you have to connect with any of the services in AWS. There are some prerequisites for the same.
* You want the **Access key ID** & **Secret Access key** & **AWS CLI** to be installed in your machine**.**
* Or if you are working for some clients in which you don’t have the keys, then you can create an **IAM User,** or you can ask your client to create it on your behalf if you don’t have access.
* There you can create **Access key ID & Secret Access key** for that specific account.
* Steps to configure the CMD with AWS Access keys. **(You can google it for that.)**
* After configuring you have to check if it is successfully installed or not. **(Refer the below screenshot)**

**Text

Description automatically generated**

**Now we can see how we can perform the task in AppStream by CMD.**

1. How we can **Enable User** in **USERPOOL.**

**Command:-** aws appstream enable-user --user-name parthg@maveric-systems.com --authentication-type USERPOOL

1. How to **Disable User** in **USERPOOL.**

**Command:-** aws appstream disable-user --user-name parthg@maveric-systems.com --authentication-type USERPOOL

1. How we can **Delete User** in **USERPOOL.**

**Command:-** aws appstream delete-user --user-name parthg@maveric-systems.com --authentication-type USERPOOL

1. How we can **Stop Fleet** in **AppStream**.

**Command:-** aws appstream stop-fleet --name AppStream\_Fleet\_Uno

1. How to **List Out** all the Users.

**Command:-** aws appstream describe-users --authentication-type USERPOOL



1. How to **Start** **Fleet** in AppStream.

**Command:-** aws appstream start-fleet --name AppStream\_Fleet\_Uno



1. How to **Stop fleet** in AppStream.

**Command:-** aws appstream stop-fleet --name AppStream\_Fleet\_Uno

Text

Description automatically generated with medium confidence