Deployment Of Application:

Architecture:

We have developed a Machine learning application using Google Colab. Implemented various data cleaning techniques, visualization and Model training and testing. Then we used Streamlit to host our application. Streamlit is an open source platform for machine learning application development, it uses Python as core language.

We decided to host our application using AWS cloud services.

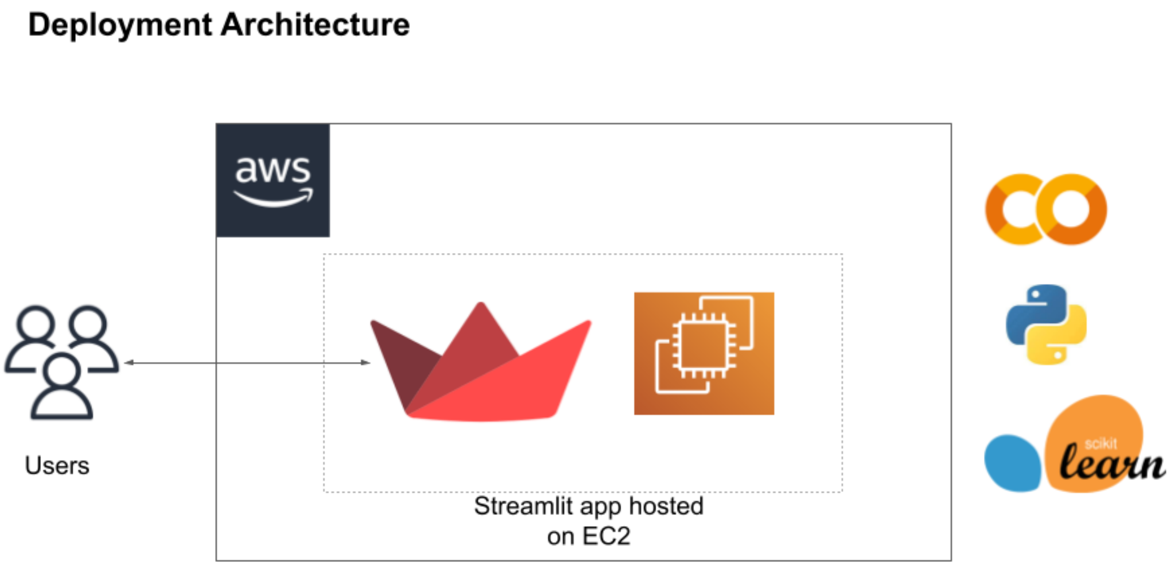
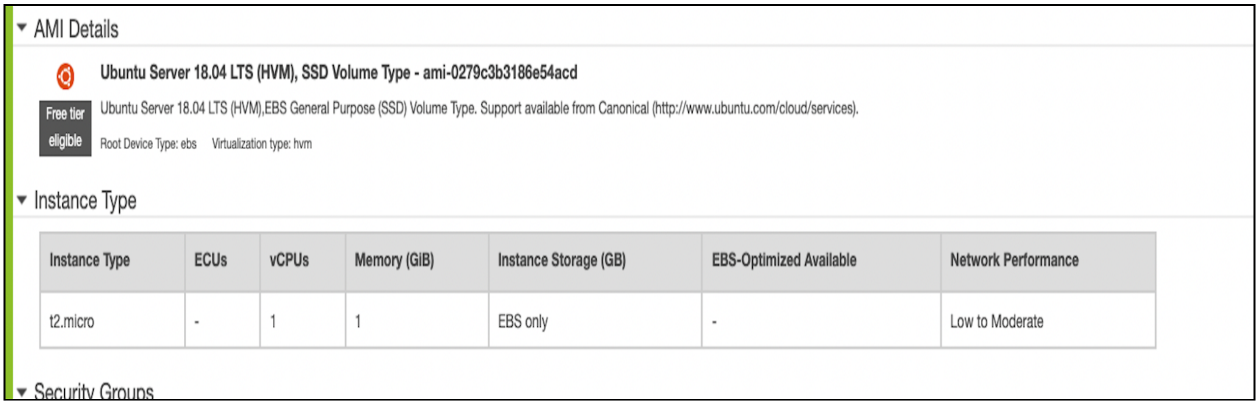


Figure 1 Deployment Architecture

For deployment of our application, we decided to go with Amazon EC2 free tier instance. Used Ubuntu operating system for hosting the application. We follower following implementation steps:

- Opened EC2 console on AWS.
- On the Console, selected “Launch a Virtual Machine”.
- We selected **t2.micro** type of instance, with AMI : **ami-0279c3b3186e54acd (Ubuntu Machine).**



▾ AMI Details

Ubuntu Server 18.04 LTS (HVM), SSD Volume Type - ami-0279c3b3186e54acd

Free tier eligible

Ubuntu Server 18.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (http://www.ubuntu.com/cloud/services).

Root Device Type: ebs   Virtualization type: hvm

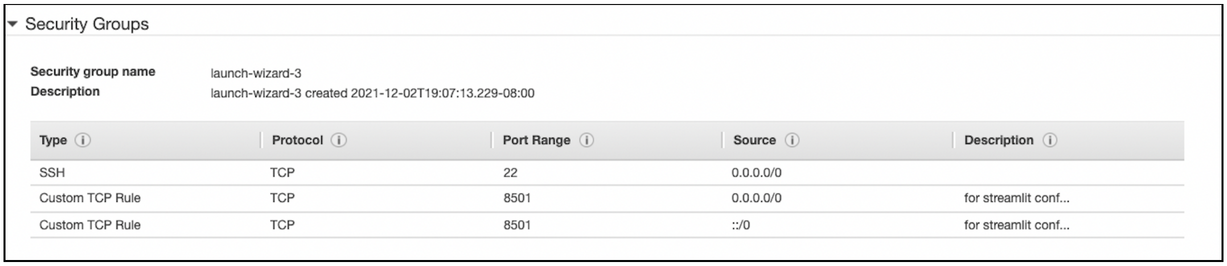
▾ Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

▾ Security Groups

Figure 2 EC2 instance AMI

- On Configure Security Group tab, we opened port 8501
- The streamlit application uses 8501 port number , in the security group we have changed the TCP rules as shown in the diagram below.



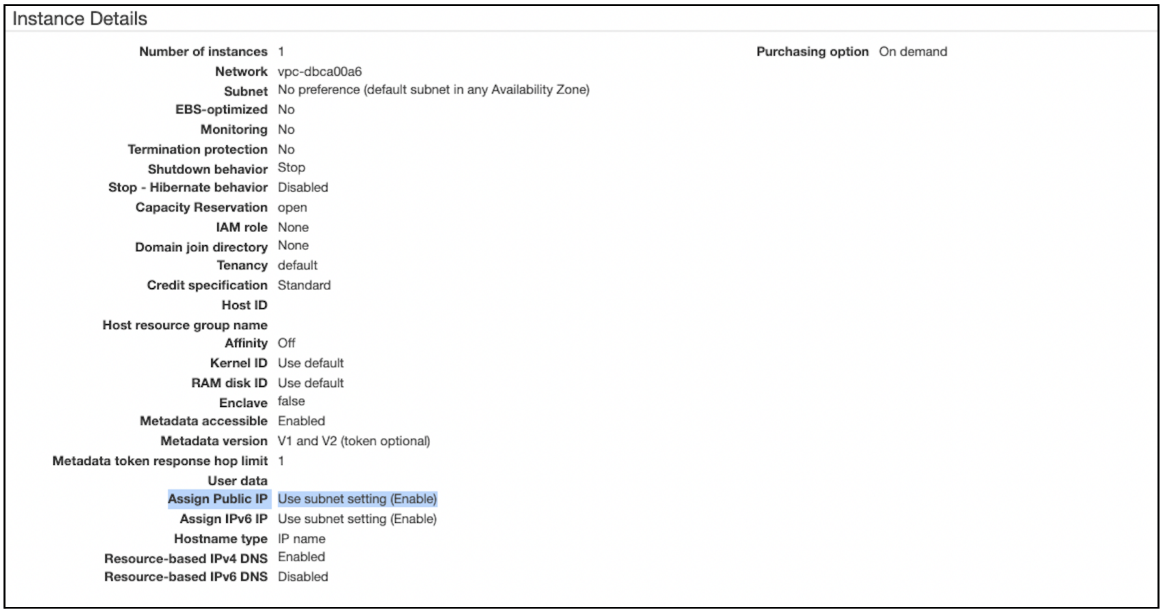
▾ Security Groups

Security group name launch-wizard-3
Description launch-wizard-3 created 2021-12-02T19:07:13.229-08:00

Type ⓘ	Protocol ⓘ	Port Range ⓘ	Source ⓘ	Description ⓘ
SSH	TCP	22	0.0.0.0/0	
Custom TCP Rule	TCP	8501	0.0.0.0/0	for streamlit conf...
Custom TCP Rule	TCP	8501	::/0	for streamlit conf...

Figure 3 Security Group Set-up

- Below figure shows the configuration of the EC2 instance.

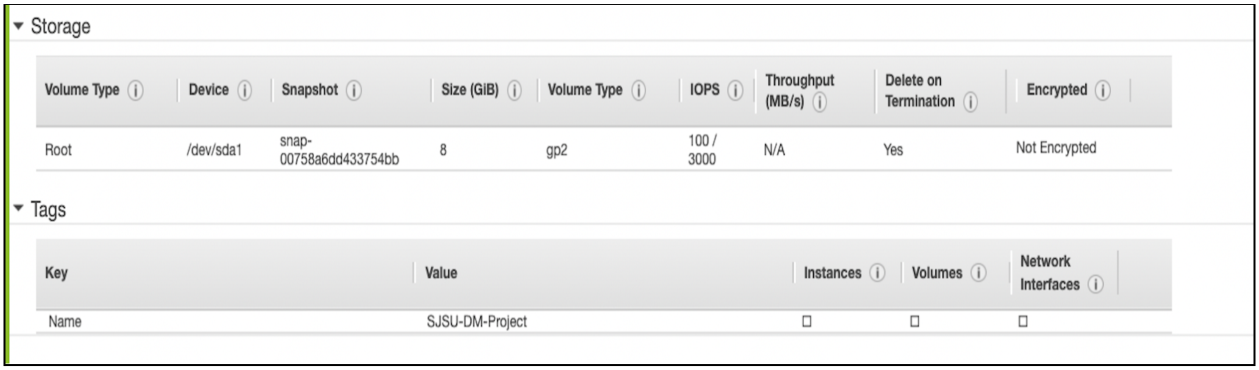


Instance Details

Number of instances	1	Purchasing option	On demand
Network	vpc-dbca00a6		
Subnet	No preference (default subnet in any Availability Zone)		
EBS-optimized	No		
Monitoring	No		
Termination protection	No		
Shutdown behavior	Stop		
Stop - Hibernate behavior	Disabled		
Capacity Reservation	open		
IAM role	None		
Domain join directory	None		
Tenancy	default		
Credit specification	Standard		
Host ID			
Host resource group name			
Affinity	Off		
Kernel ID	Use default		
RAM disk ID	Use default		
Enclave	false		
Metadata accessible	Enabled		
Metadata version	V1 and V2 (token optional)		
Metadata token response hop limit	1		
User data			
Assign Public IP	Use subnet setting (Enable)		
Assign IPv6 IP	Use subnet setting (Enable)		
Hostname type	IP name		
Resource-based IPv4 DNS	Enabled		
Resource-based IPv6 DNS	Disabled		

Figure 4 EC2 Configurations

- Selected the EBS storage of 8 GB.



Storage

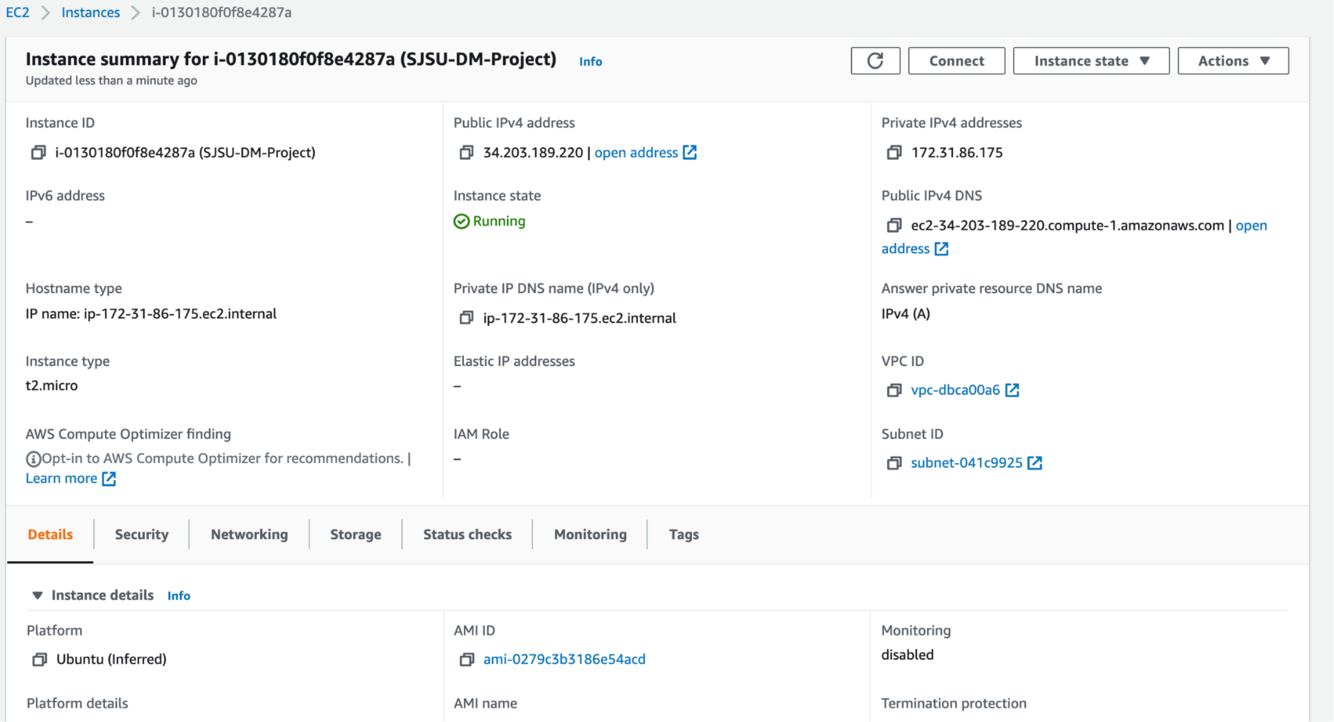
Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-00758a6dd433754bb	8	gp2	100 / 3000	N/A	Yes	Not Encrypted

Tags

Key	Value	Instances	Volumes	Network Interfaces
Name	SJSU-DM-Project	□	□	□

Figure 5 Storage Allocated to server

- Before launching the EC2 instance we need to generate key-pair to connect to our Server. We downloaded the .pem file.



Instance summary for i-0130180f0f8e4287a (SJSU-DM-Project) Info

Updated less than a minute ago

Instance ID
i-0130180f0f8e4287a (SJSU-DM-Project)

Public IPv4 address
34.203.189.220 | open address

Private IPv4 addresses
172.31.86.175

IPv6 address
–

Instance state
Running

Public IPv4 DNS
ec2-34-203-189-220.compute-1.amazonaws.com | open address

Hostname type
IP name: ip-172-31-86-175.ec2.internal

Private IP DNS name (IPv4 only)
ip-172-31-86-175.ec2.internal

Answer private resource DNS name
IPv4 (A)

Instance type
t2.micro

Elastic IP addresses
–

VPC ID
vpc-dbca00a6

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more

IAM Role
–

Subnet ID
subnet-041c9925

▼ Instance details Info

Platform
Ubuntu (Inferred)

AMI ID
ami-0279c3b3186e54acd

Monitoring
disabled

Platform details

AMI name

Termination protection

Figure 6 Summary of EC2 instance

- We have created the .pem file to connect to our EC2 instance. After downloading the file, we could connect to EC2 by using following command:

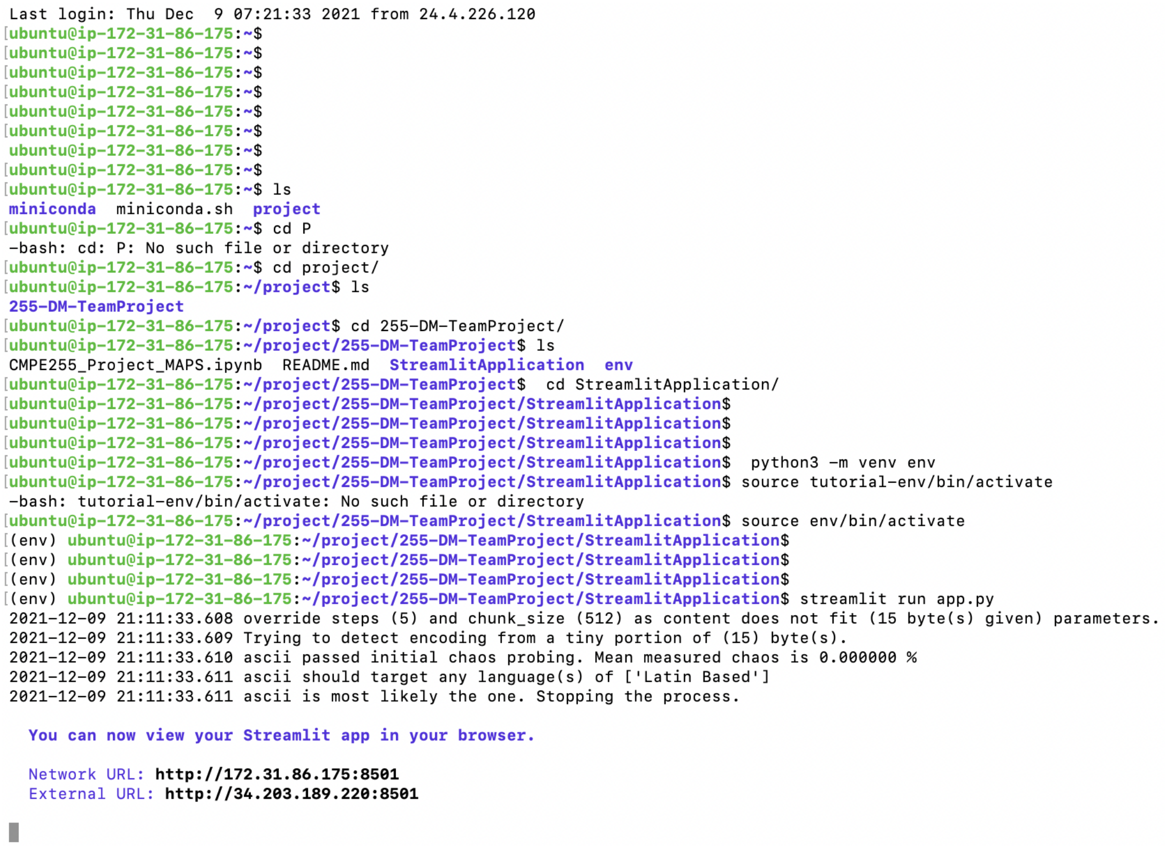
  ssh -i "dm-pro.pem" ubuntu@34.203.189.220

- We installed the required libraries used by Streamlit :

  pip install streamlit

- For running the streamlit application, we cloned the project repo and ran the following command:

  steamlit run app.py



```
 Last login: Thu Dec  9 07:21:33 2021 from 24.4.226.120
[ubuntu@ip-172-31-86-175:~$
[ubuntu@ip-172-31-86-175:~$
[ubuntu@ip-172-31-86-175:~$
[ubuntu@ip-172-31-86-175:~$
[ubuntu@ip-172-31-86-175:~$
[ubuntu@ip-172-31-86-175:~$
 ubuntu@ip-172-31-86-175:~$
[ubuntu@ip-172-31-86-175:~$
[ubuntu@ip-172-31-86-175:~$ ls
 miniconda  miniconda.sh  project
[ubuntu@ip-172-31-86-175:~$ cd P
 -bash: cd: P: No such file or directory
[ubuntu@ip-172-31-86-175:~$ cd project/
[ubuntu@ip-172-31-86-175:~/project$ ls
 255-DM-TeamProject
[ubuntu@ip-172-31-86-175:~/project$ cd 255-DM-TeamProject/
[ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject$ ls
 CMPE255_Project_MAPS.ipynb  README.md  StreamlitApplication  env
[ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject$  cd StreamlitApplication/
[ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$
[ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$
[ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$
[ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$  python3 -m venv env
[ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$ source tutorial-env/bin/activate
 -bash: tutorial-env/bin/activate: No such file or directory
[ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$ source env/bin/activate
[(env) ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$
[(env) ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$
[(env) ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$
[(env) ubuntu@ip-172-31-86-175:~/project/255-DM-TeamProject/StreamlitApplication$ streamlit run app.py
 2021-12-09 21:11:33.608 override steps (5) and chunk_size (512) as content does not fit (15 byte(s) given) parameters.
 2021-12-09 21:11:33.609 Trying to detect encoding from a tiny portion of (15) byte(s).
 2021-12-09 21:11:33.610 ascii passed initial chaos probing. Mean measured chaos is 0.000000 %
 2021-12-09 21:11:33.611 ascii should target any language(s) of ['Latin Based']
 2021-12-09 21:11:33.611 ascii is most likely the one. Stopping the process.

   You can now view your Streamlit app in your browser.

   Network URL: http://172.31.86.175:8501
   External URL: http://34.203.189.220:8501

```

Figure 7 Deployment of streamlit

- Figure 7 is the screenshot of our machine, where we ran the application.

  And Streamlit app can be opened by using

  http://34.203.189.220:8501

Reference used:

Streamlit Deployment on AWS Cloud