**Web Application – Box as a service**

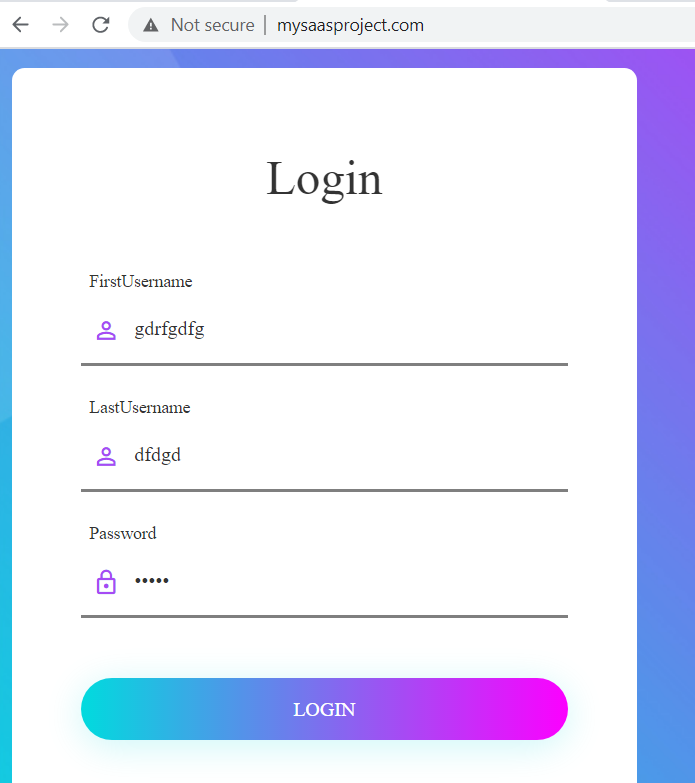
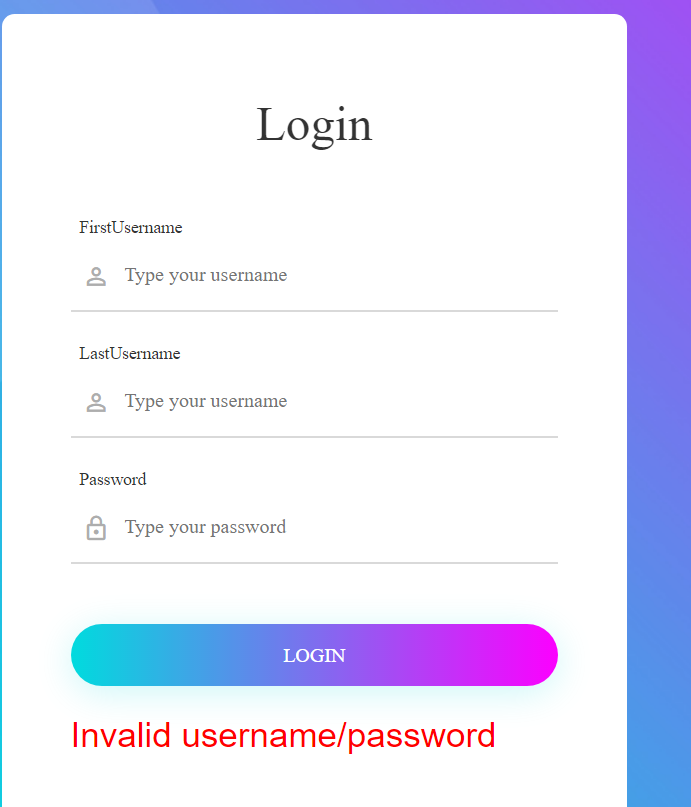
**Project Link:** <http://www.mysaasproject.com/>

**Github Link:** <https://github.com/AnishaA-git/BoxASAService>

**Youtube Link:** <https://youtu.be/cLwG2JNs0Rw>

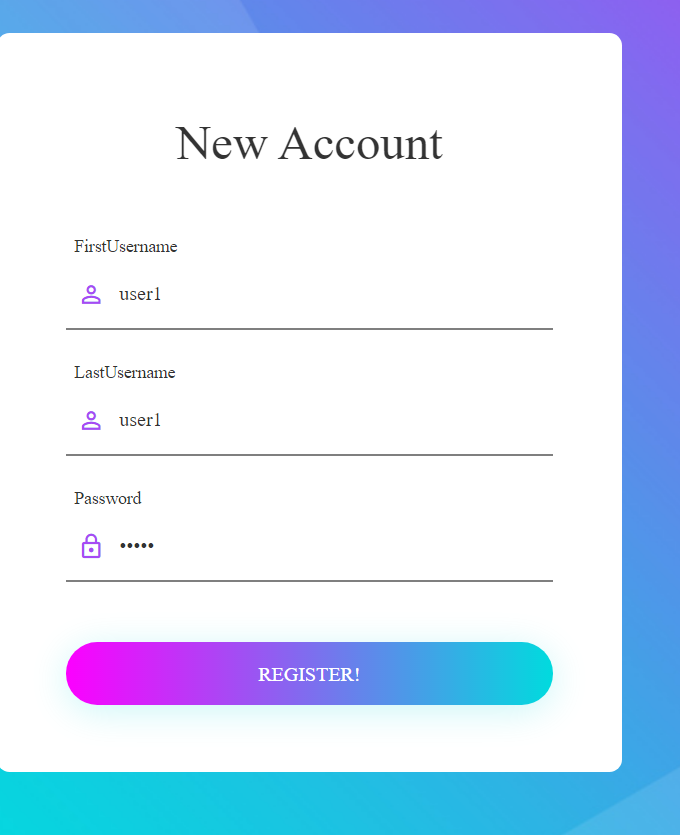
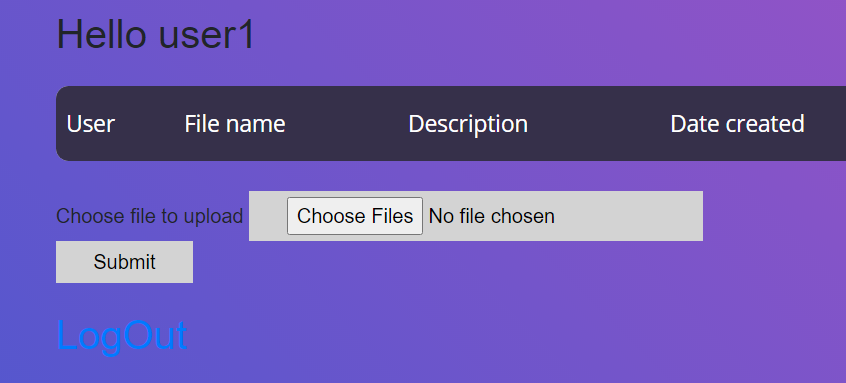
1. **Authorized users for the application**

type any random user name and click on login, it says invalid username/password.

** **

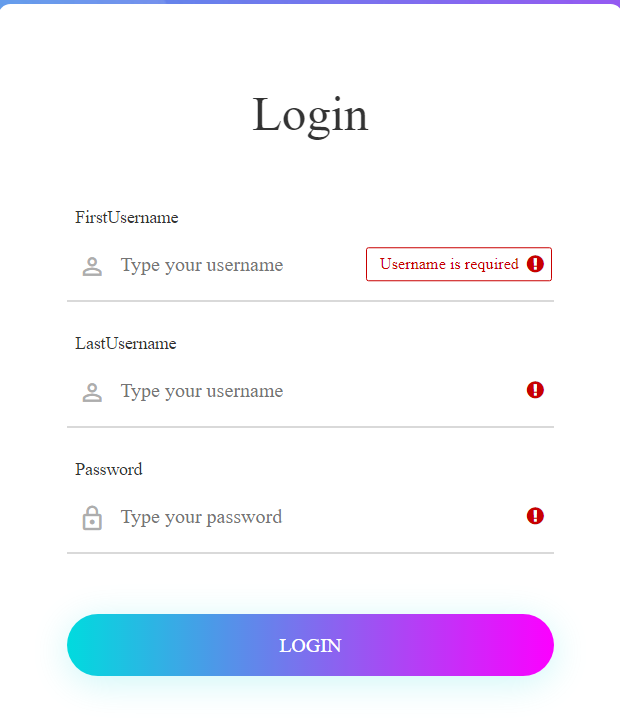
1. **Register Users –**

Click on Register, type the details of the user. Click on register you will be navigated to Login page if the registration is successful. Then enter the details on login page you will be able to successfully login as an authorized user.

** **

1. **Login page Validation –**

Just click on login button without entering any registered user details, you can see an error symbol. Just hover on it and you can find what is the error.

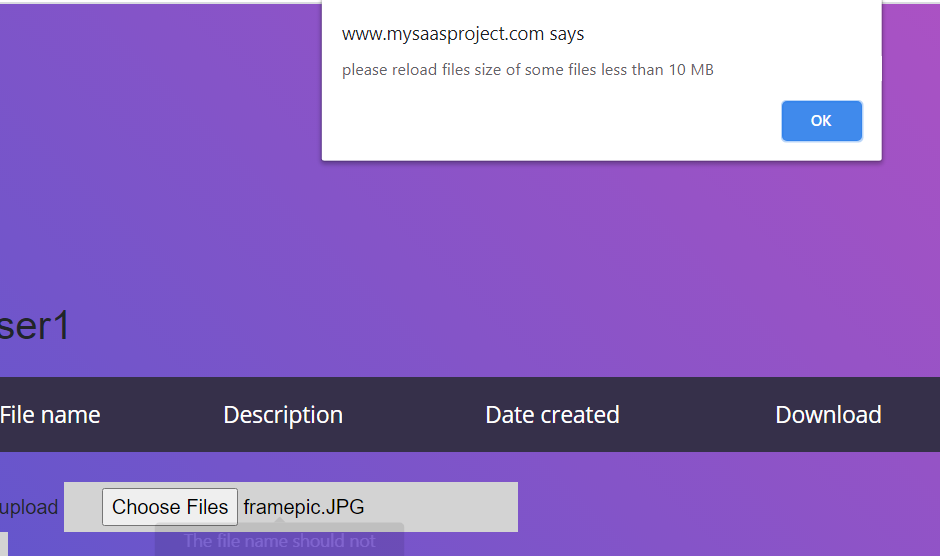
****

1. **Operations –**

* **Upload new files. (max size 10 MB per file)** –

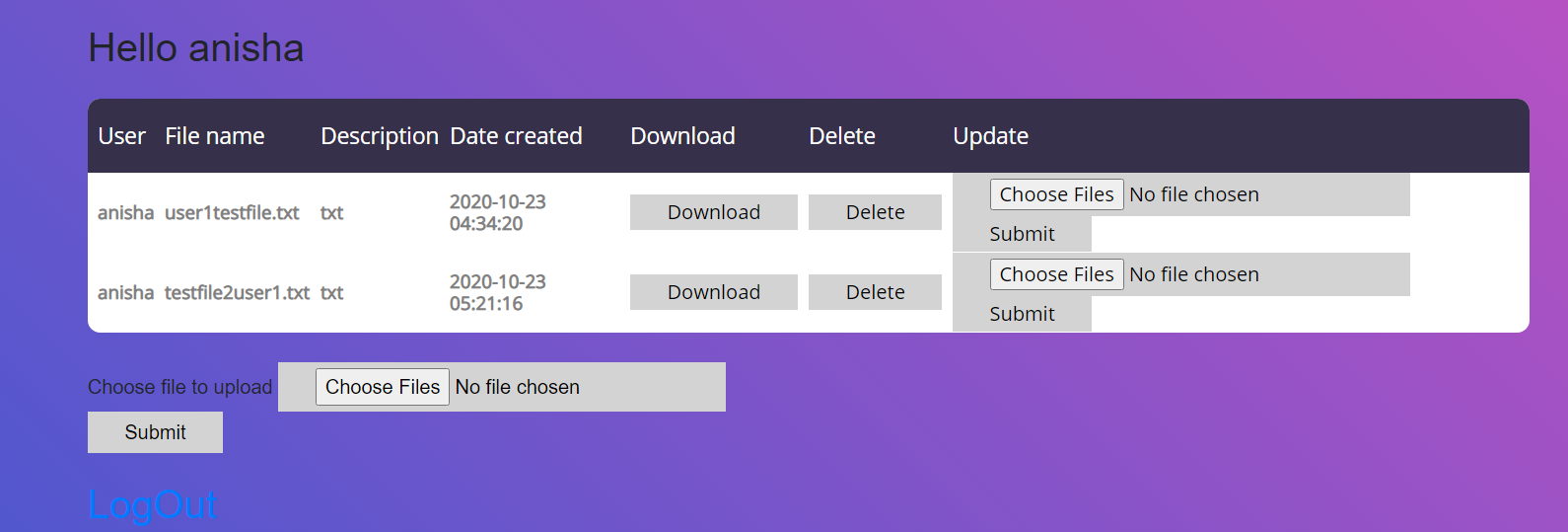
Select a file greater than 10MB as shown below, you can see dialog appears saying file size should be less than 10MB**.**

****

****

* **Browse through already uploaded list of files** –

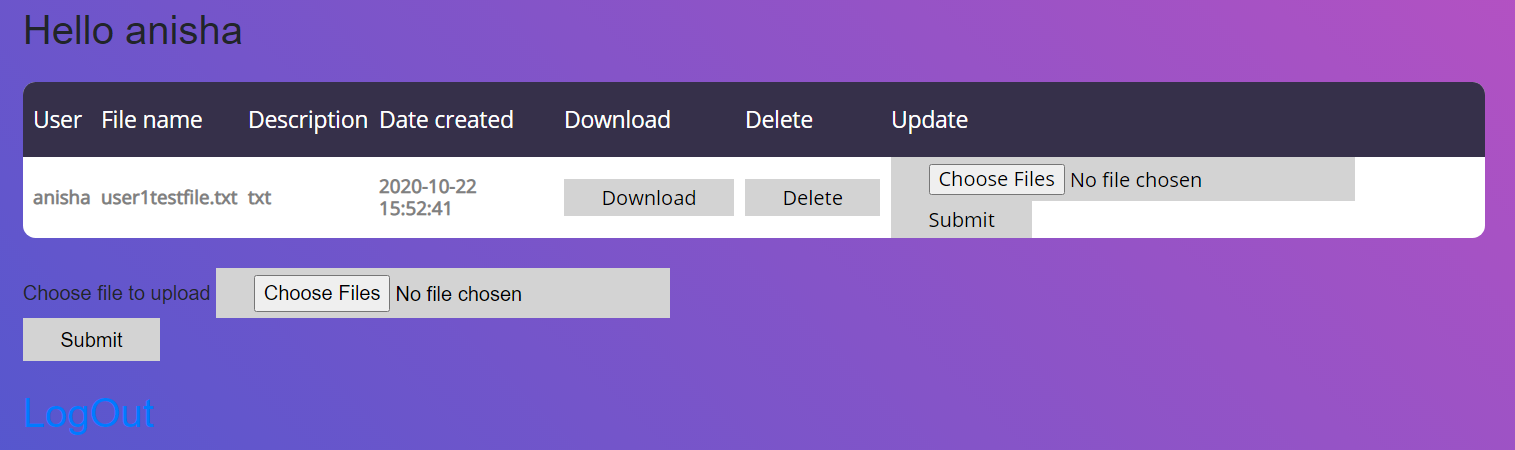
You can go through the list of file user has already uploaded with a download button added where the user can download and also reupload the updated files:

****

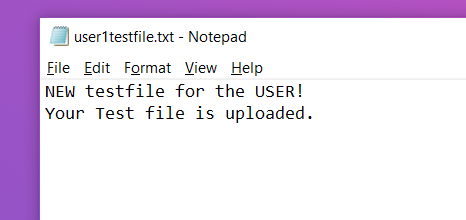
* **Update already uploaded files –**

First upload a new file to the user, and then reupload the file in order to see how update to already updated files takes place.

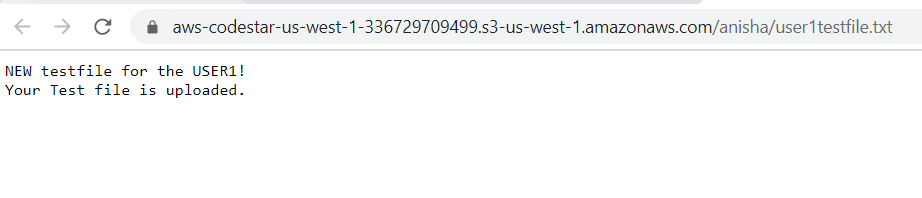
First upload to the user:

****

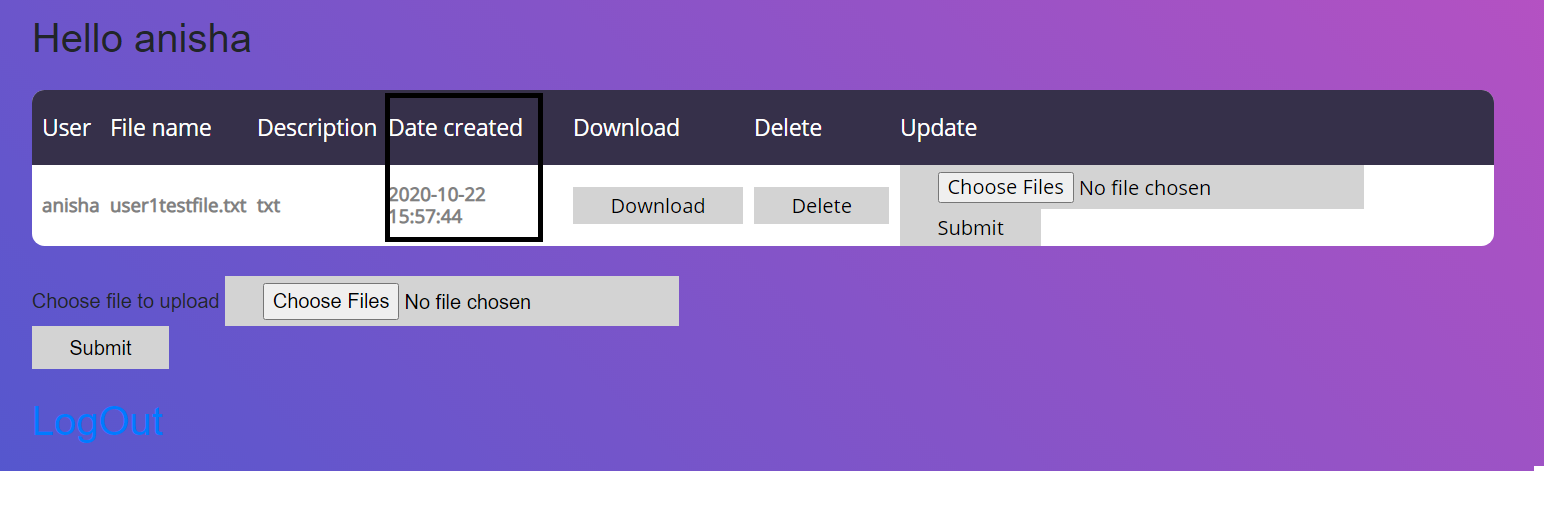
Download the file from the application to see the contents:

****

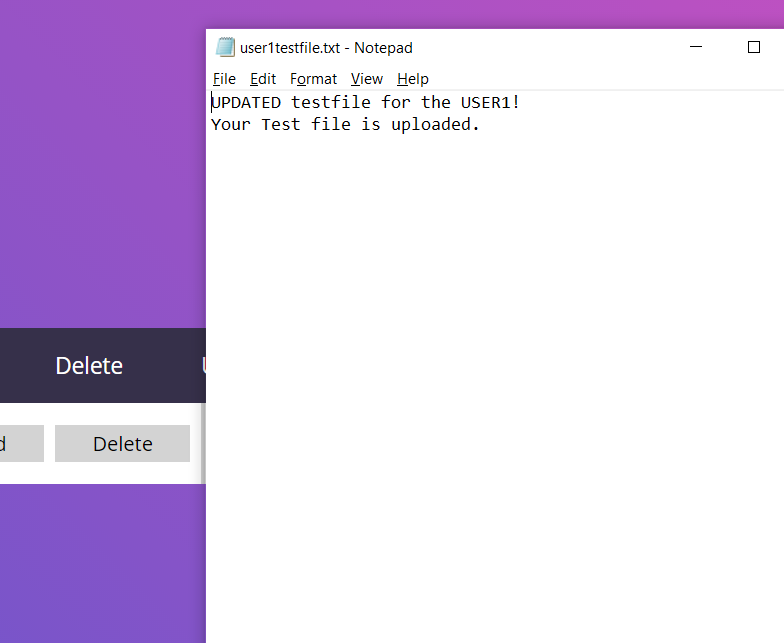
Contents of the file uploaded for the first time:

****

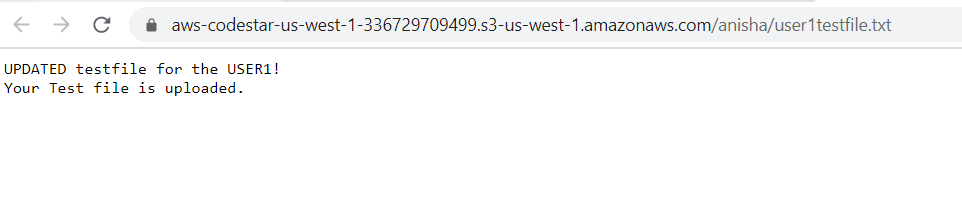
Now update this file with some change in content so that we can see if the update is happening. You can see date created is changed and updated to the new time of the file update:

****

download from the application and see the updated contents:

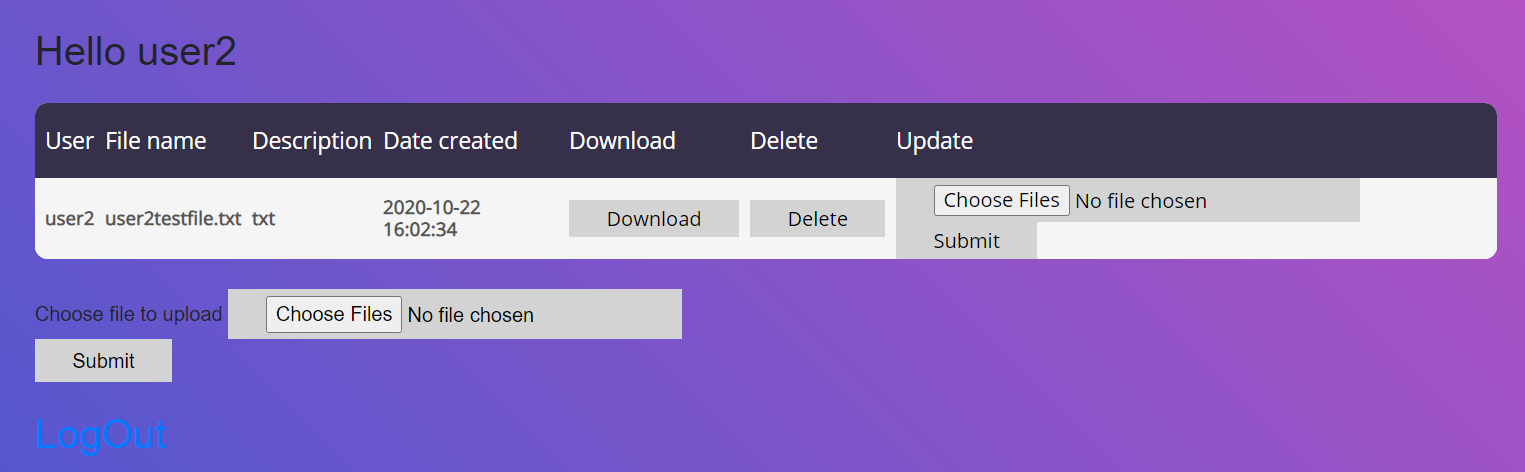
****

Updated contents from S3:

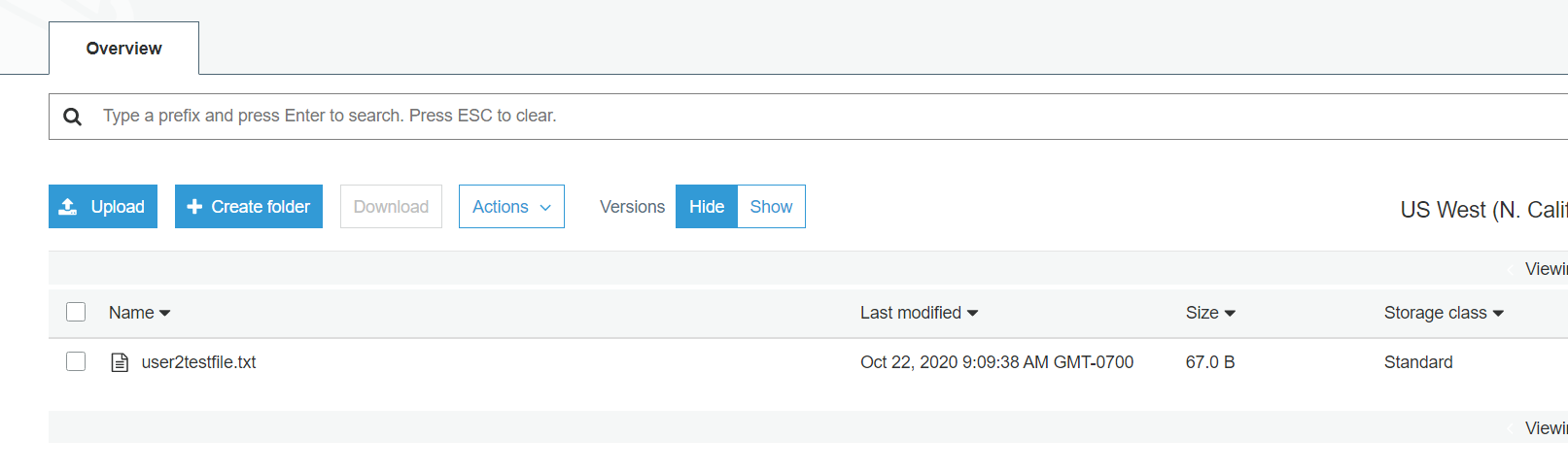
****

* **Delete already uploaded file:**

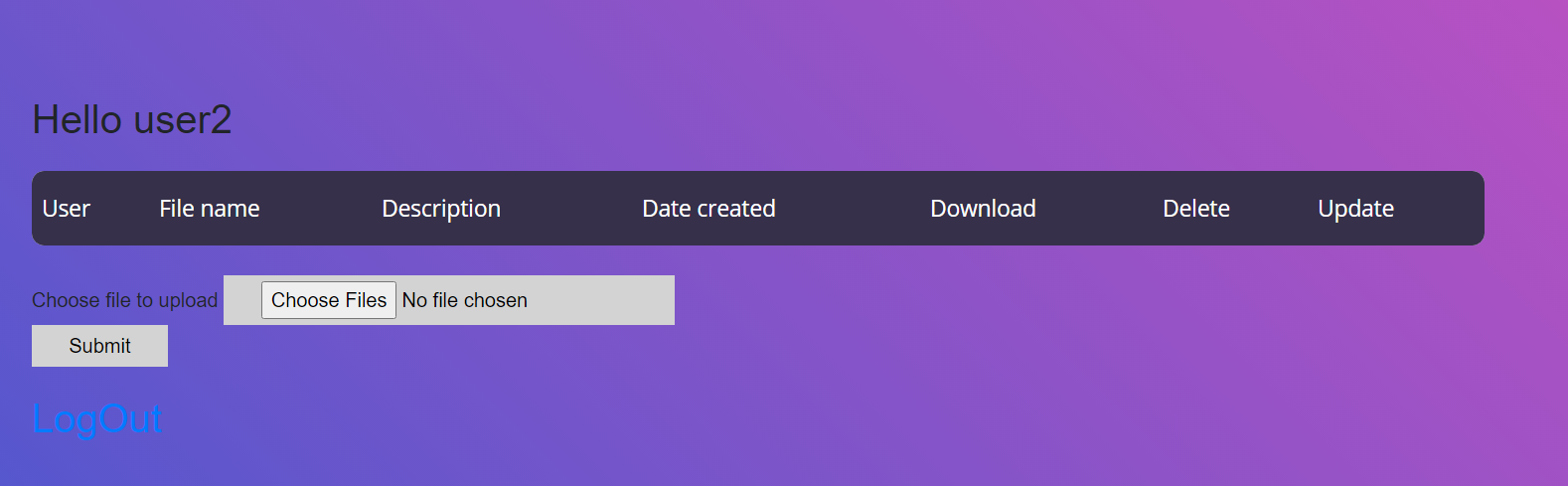
User2 has already a file uploaded:

****

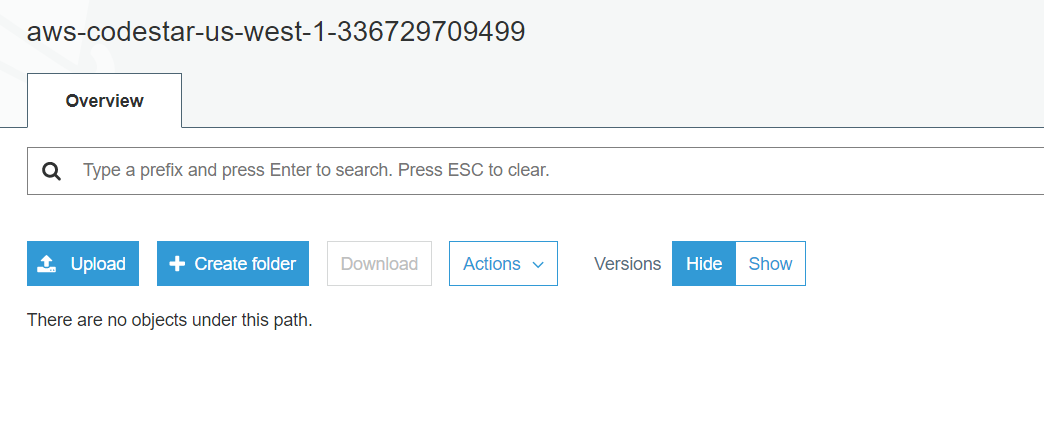
File in s3:

****

Now click on delete to see if delete functionality is working:

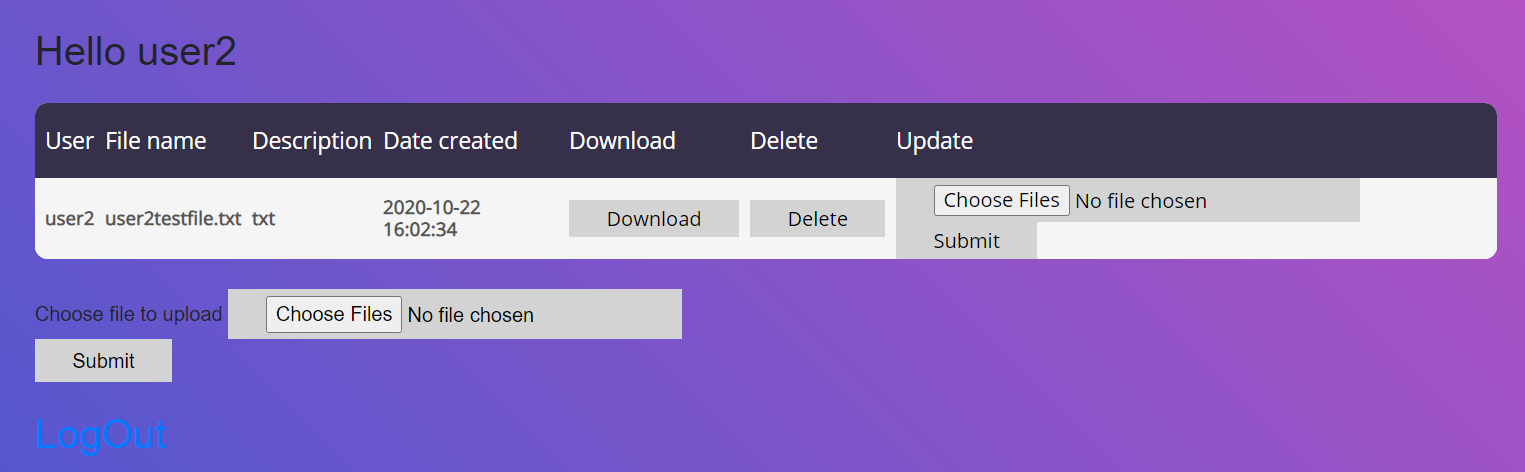
****

Delete in s3:

****

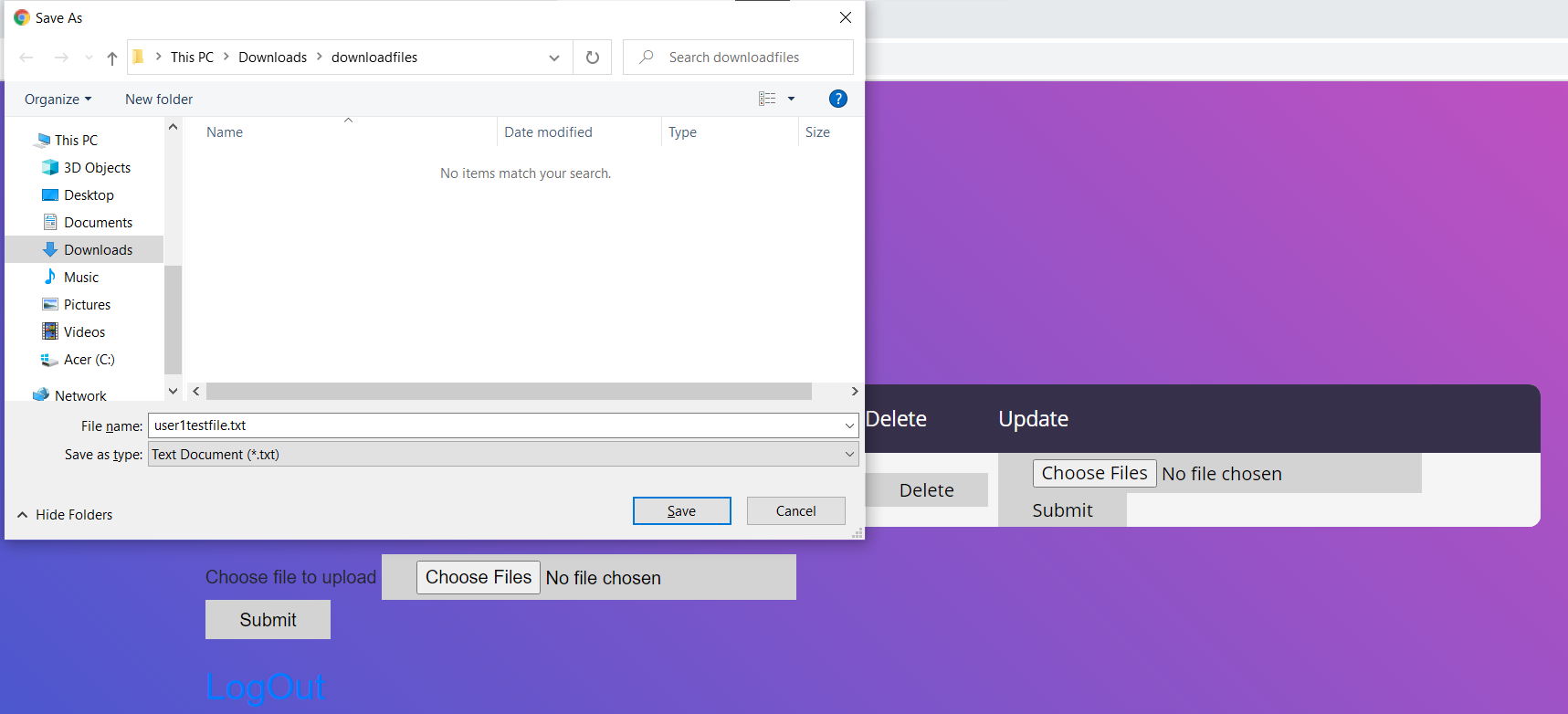
1. **For each file upload, application tracks the USER details:**

Please find all the track of the following fields for the user:

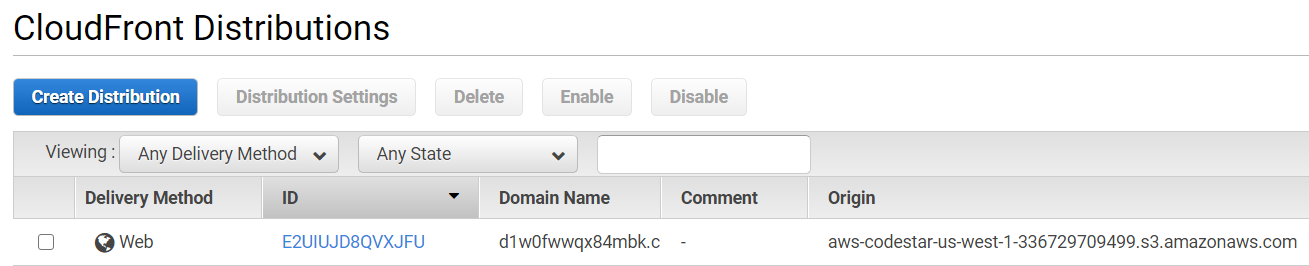
****

1. **Download functionality for user update files, leveraging cloudfront service.**

Click on Download button to download the file:

****

Have used cloudFront distribution for download**:**

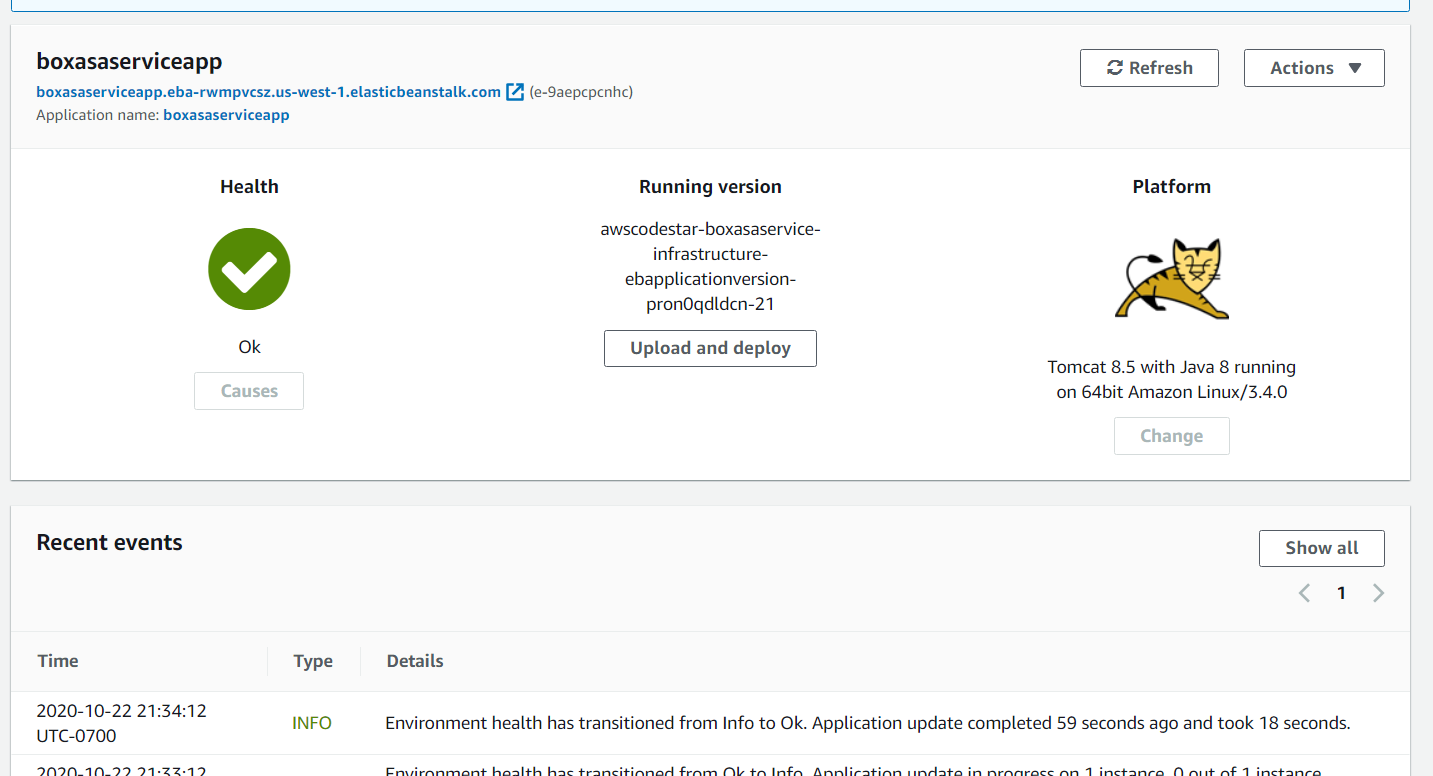
****

1. **AWS Services used for the Application:**

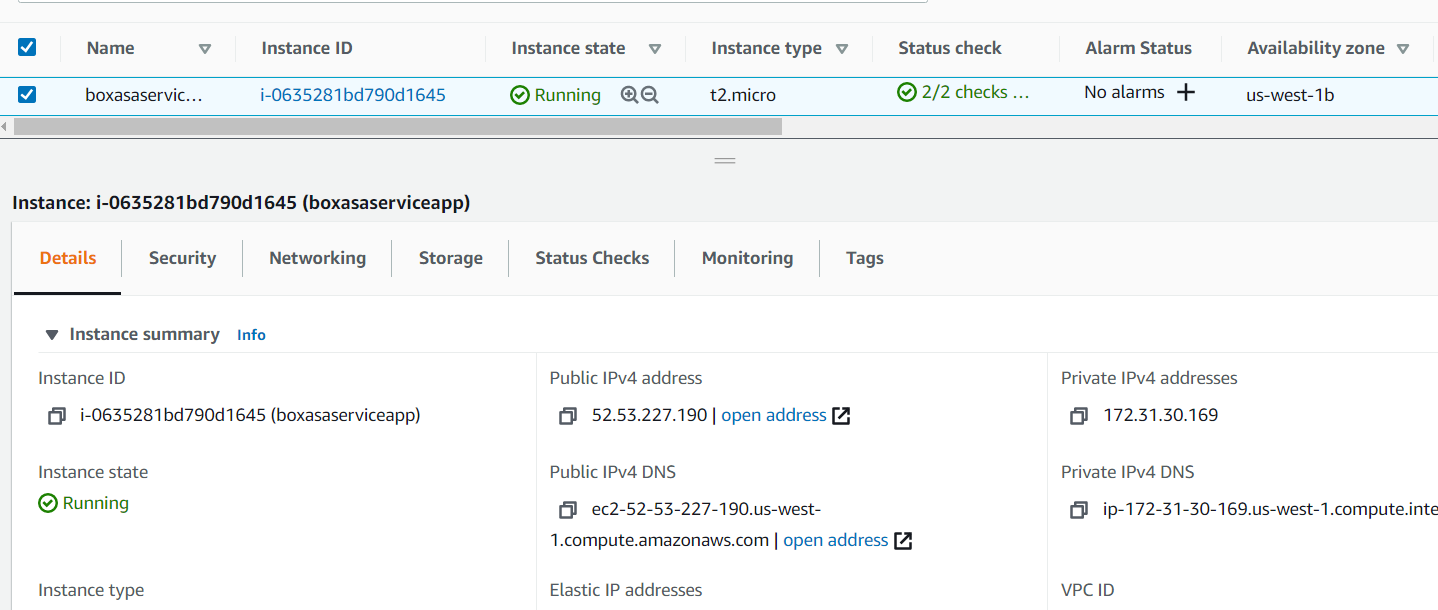
* **EC2**

As I have deployed my application on Elastic beanstalk, which has the functionality to create ec2 instances for the application. So for now I have considered only one instance but I can change this to minimum of 4 instances any time as per the requirement.

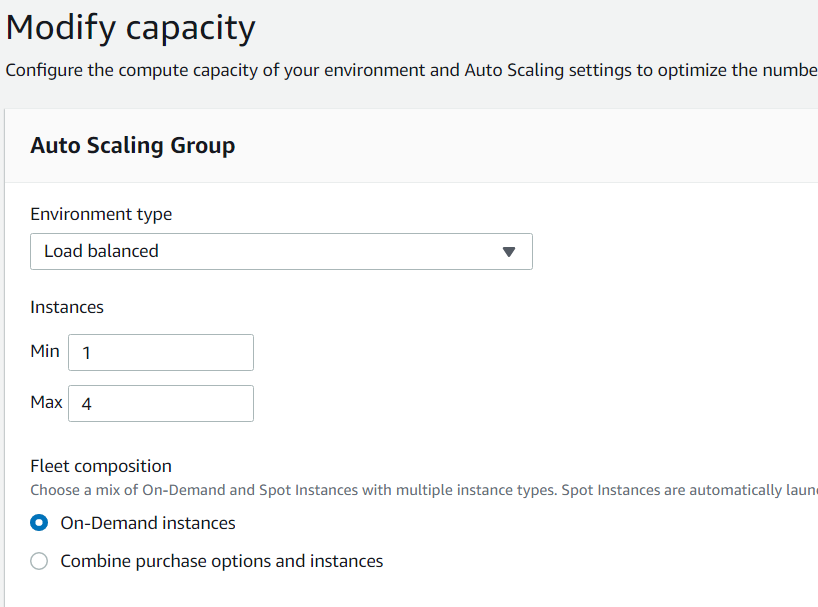
Application Deployed on Elastic Beanstalk:

****

**EC2 Instance:**

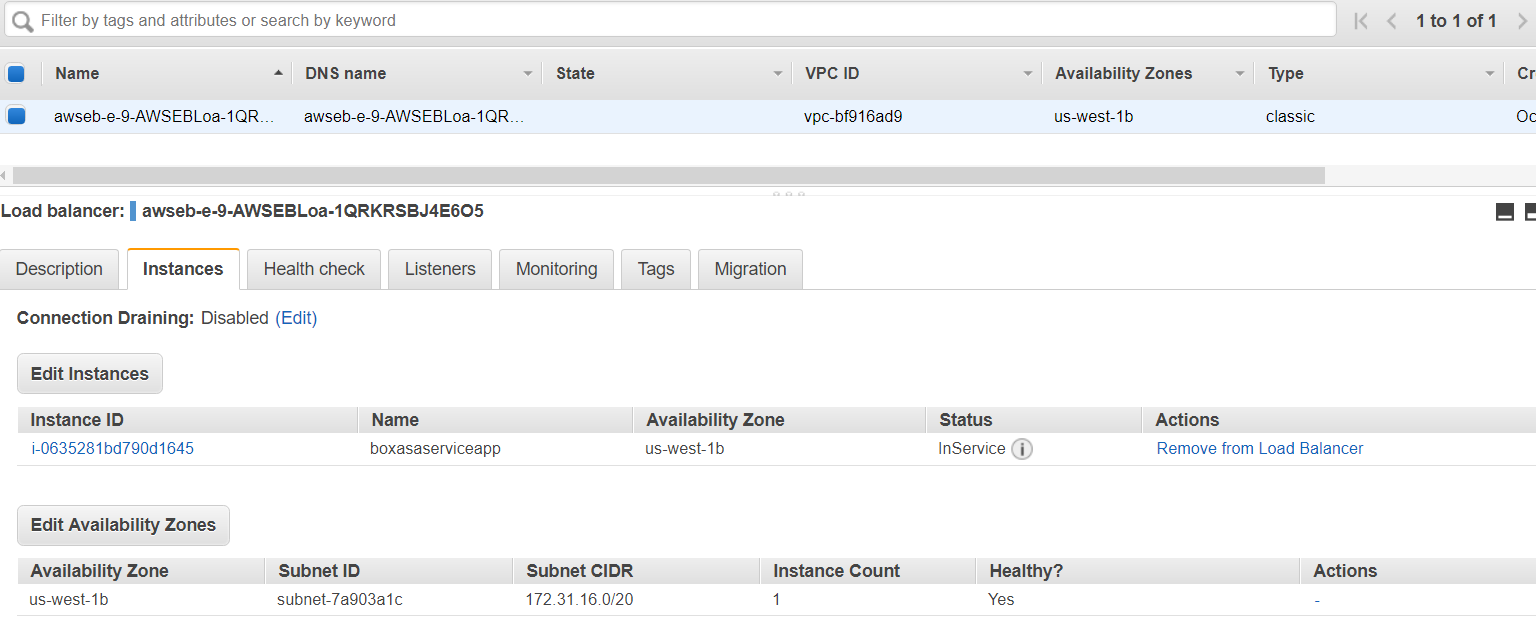
****

We can edit and add max of 4 instances at any point of time:

****

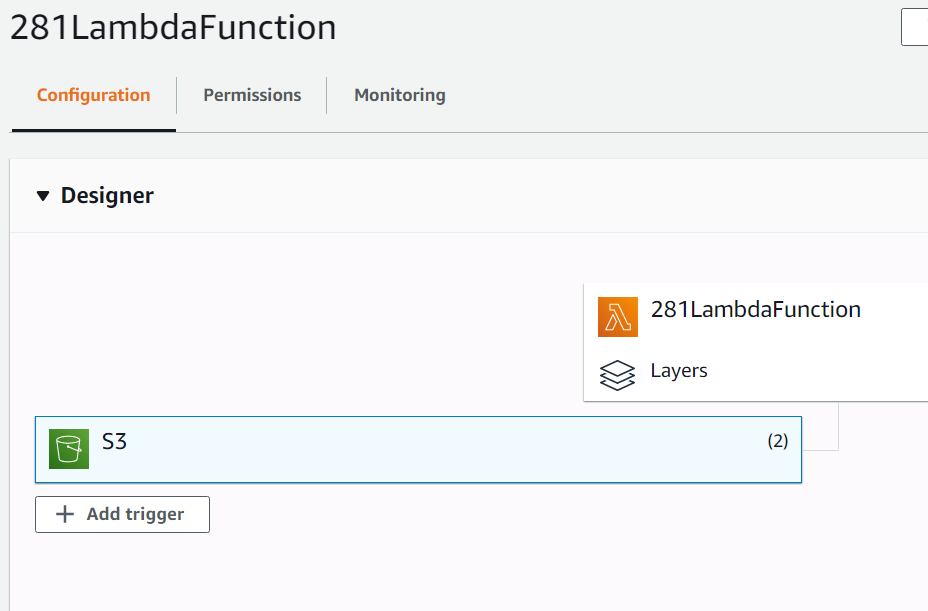
* **ELB –**

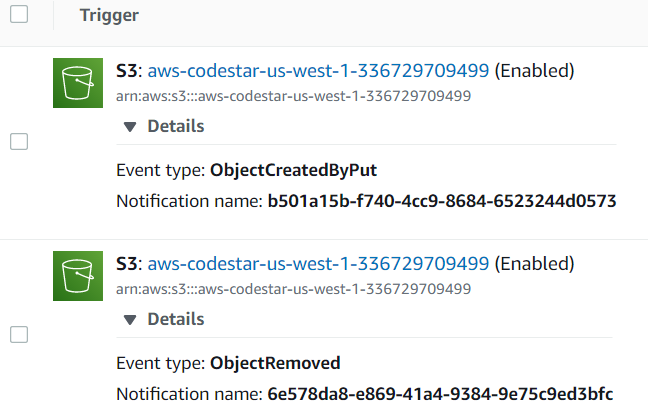
Have used a load balancer which can be modified as per our requirement if we want to increase the instances:

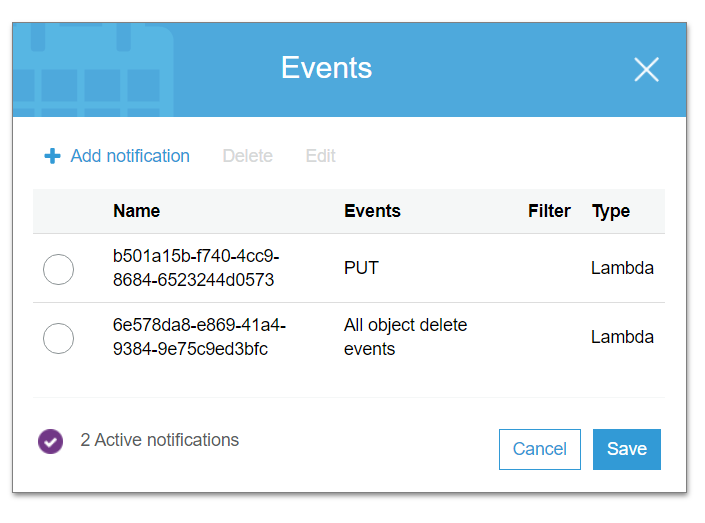
****

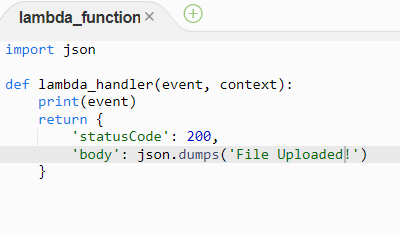
* **Lambda –**

Created a Lambda Function which triggers S3 Events like PUT and DELETE object

****

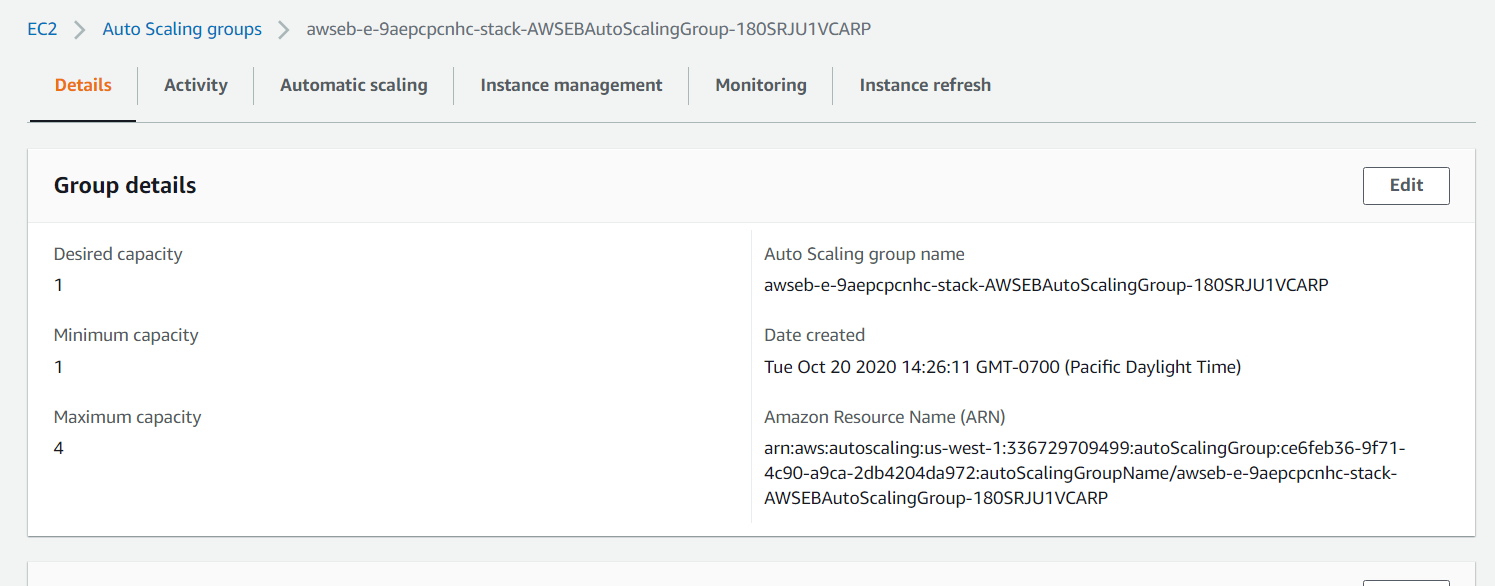
****

****

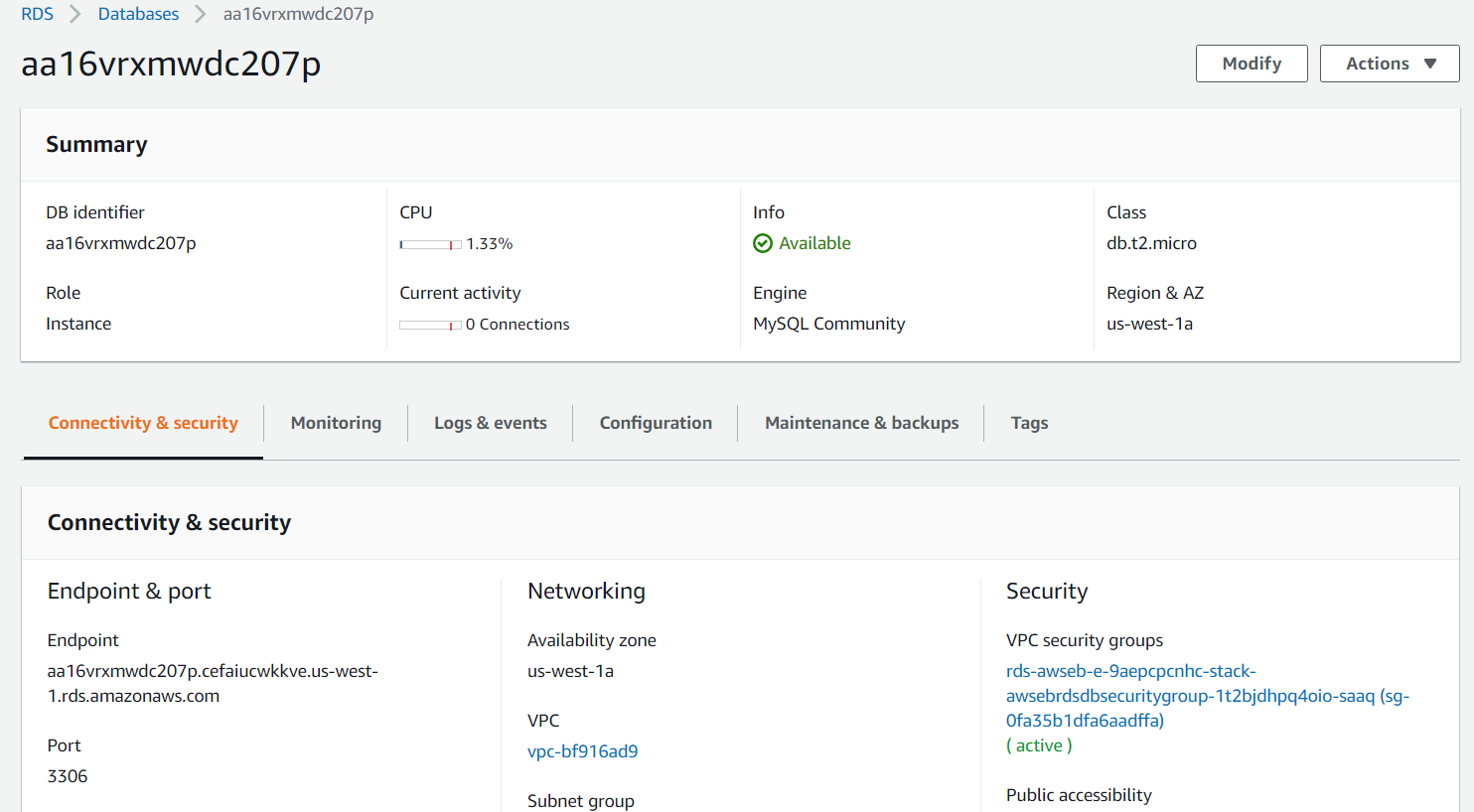
****

* **AutoScaling Group –**

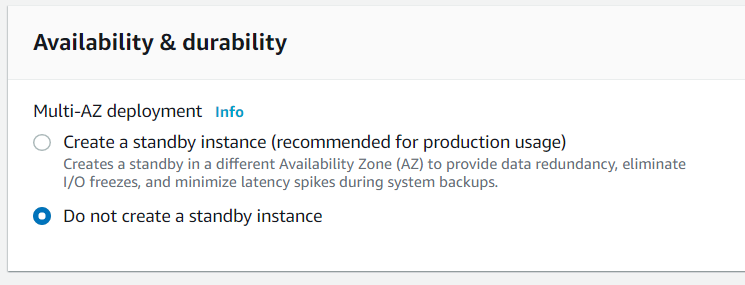
Have created Auto scaling which can be changed to any number of instances as per the requirement:

****

* **Single AZ RDS as well as procedure to convert it into multi AZ:**

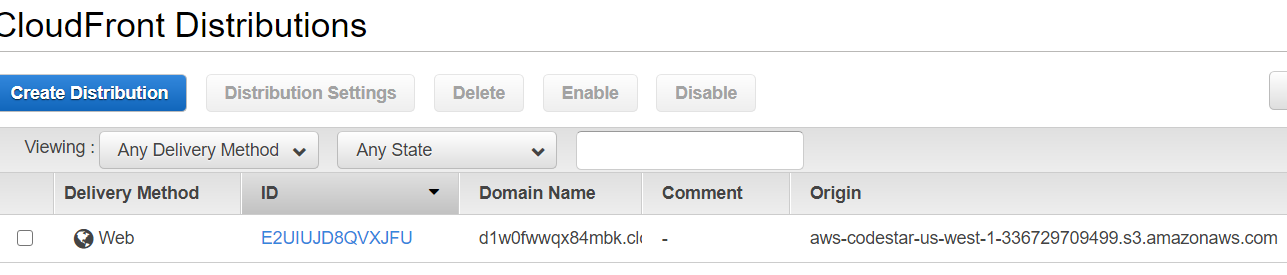
****

We can any time change the DB to multi AZ as per the requirement:

****

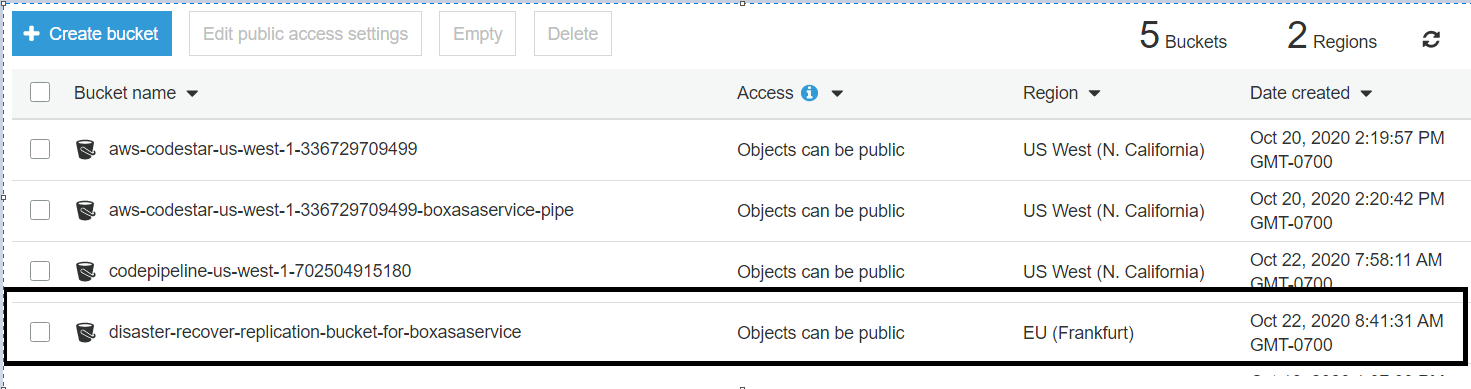
* **CloudFront –**

Have created CloudFront which gives a faster download facility from its edge location:

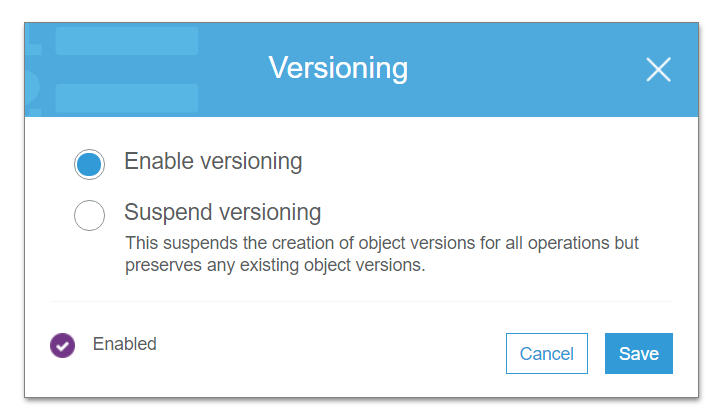
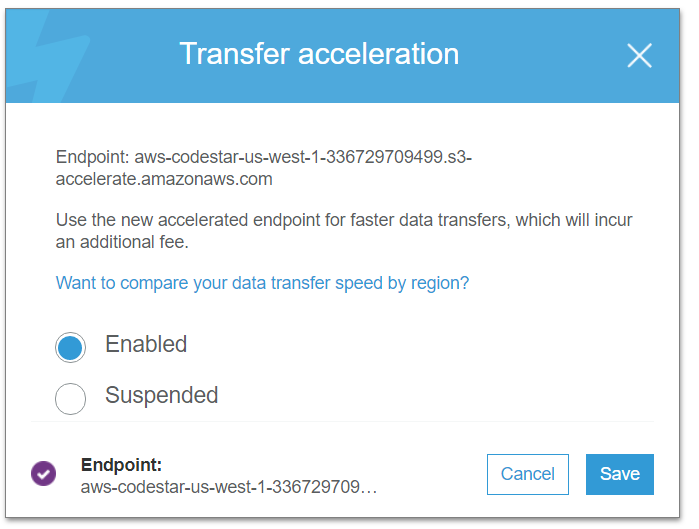
****

* **S3 –**

S3 buckets are created where the operations of the web application happen. There is one Disaster Recovery Bucket which is the replication of the primary bucket**:**

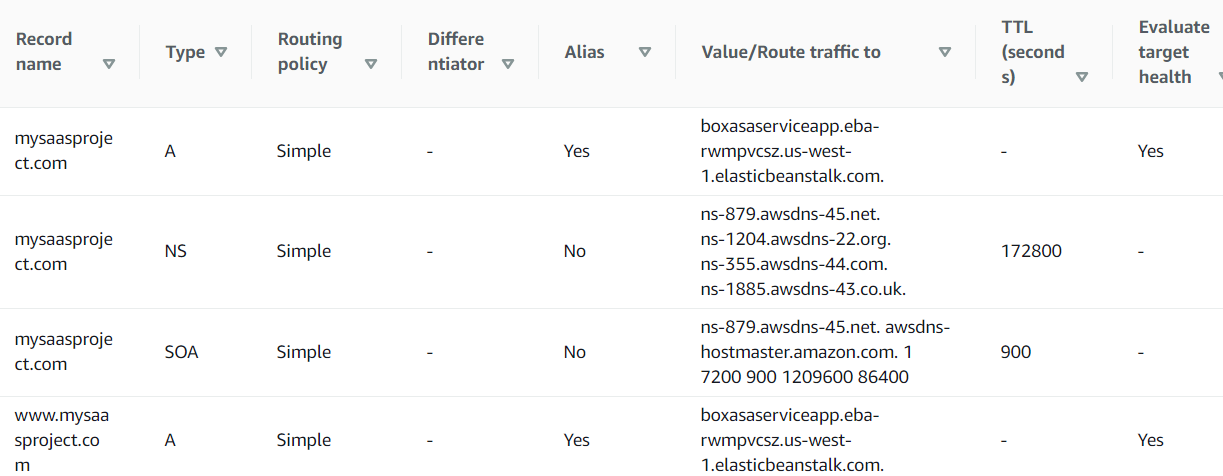
****

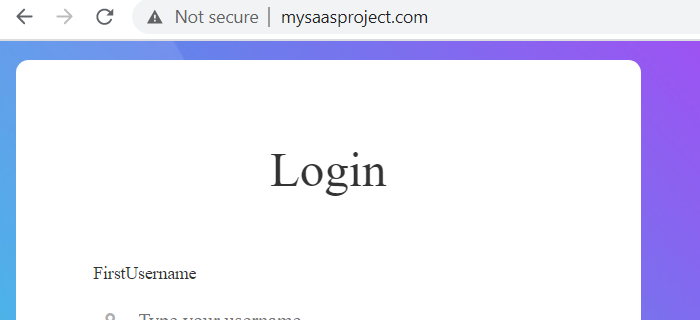
* **S3 Transfer Acceleration:**

****

* **R53 –**

R53 is created as “mysaasproject” with all the health checks enabled. Simple routing can be modified to any of the other routing like Latency based, geo based at any point of time according to requirement:

****

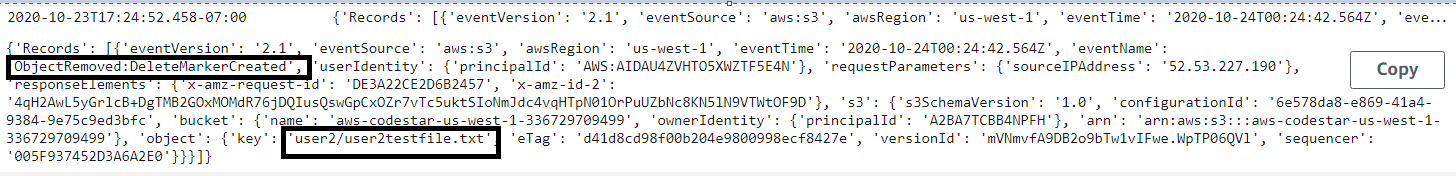
****

* **CloudWatch –**

Logs triggered when the object is uploaded to S3 bucket:

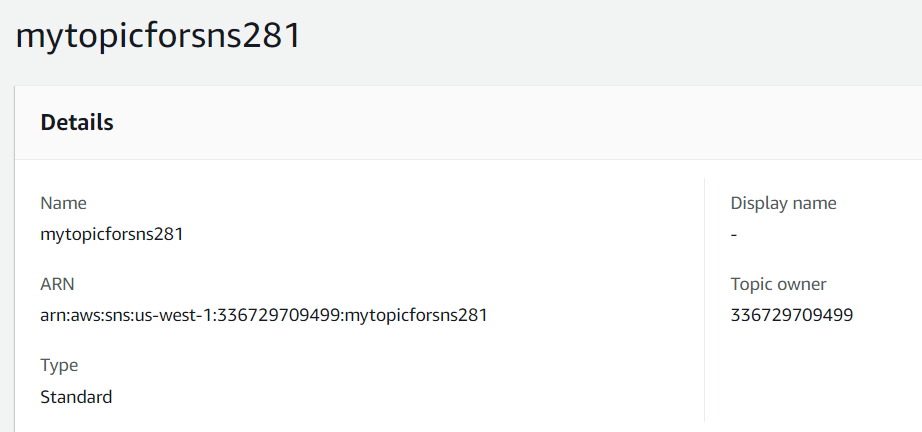
****

Logs triggered when the object is deleted from S3 bucket

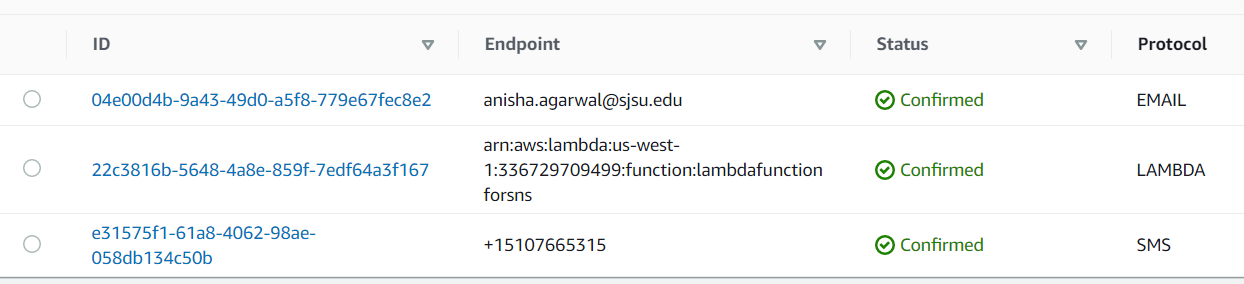
****

* **SNS –**

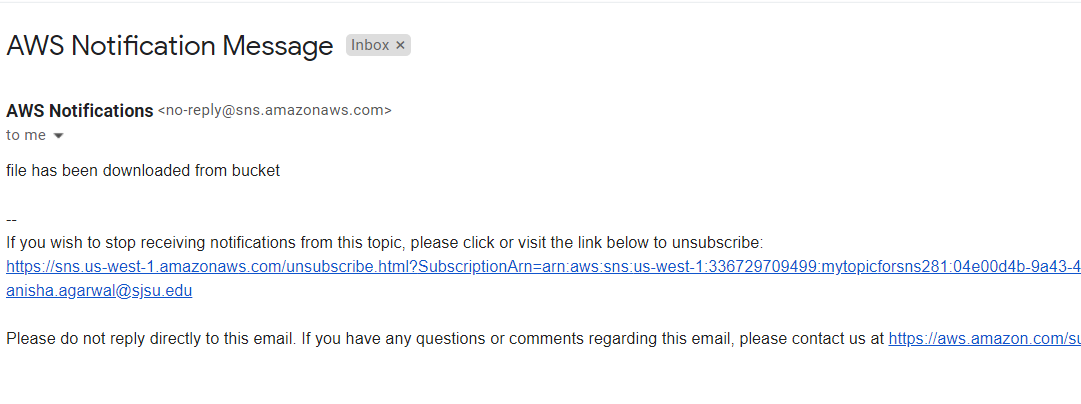
Topic is Created:

****

Subscriptions are created:

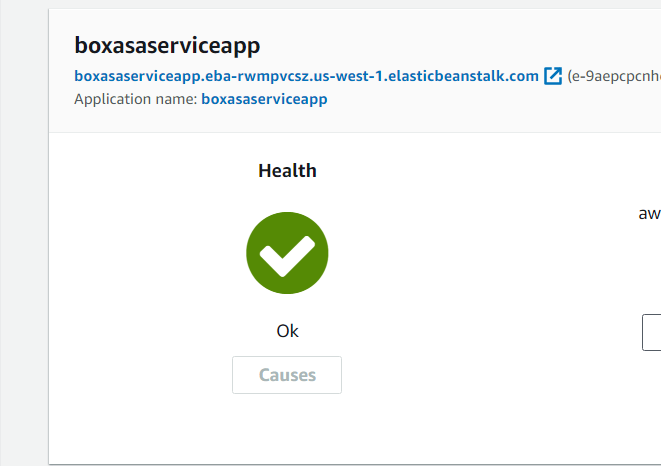
****

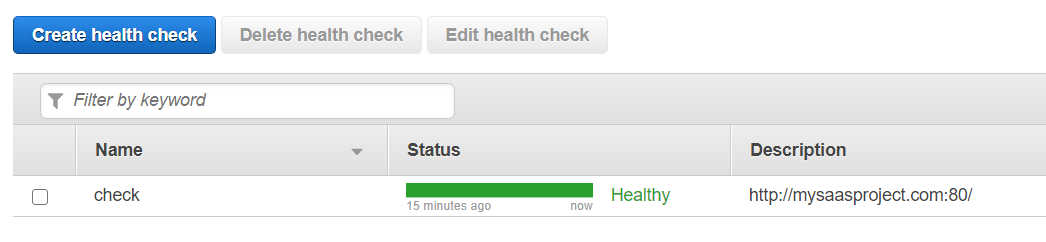
Mail is triggered saying “File has been downloaded”:

****

1. **Proper Health checks are created for the application.**

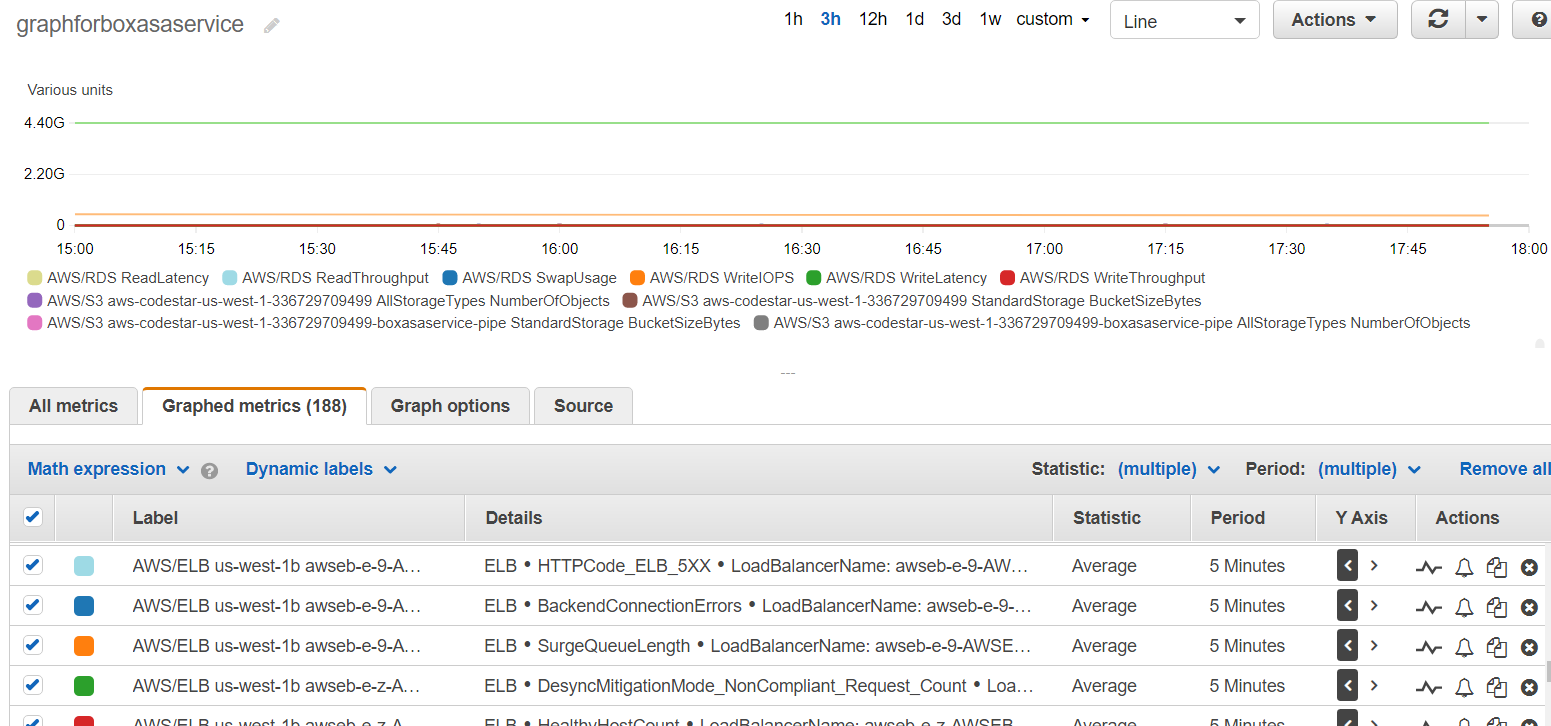
**Health check of overall Elastic Beanstalk is checked:**

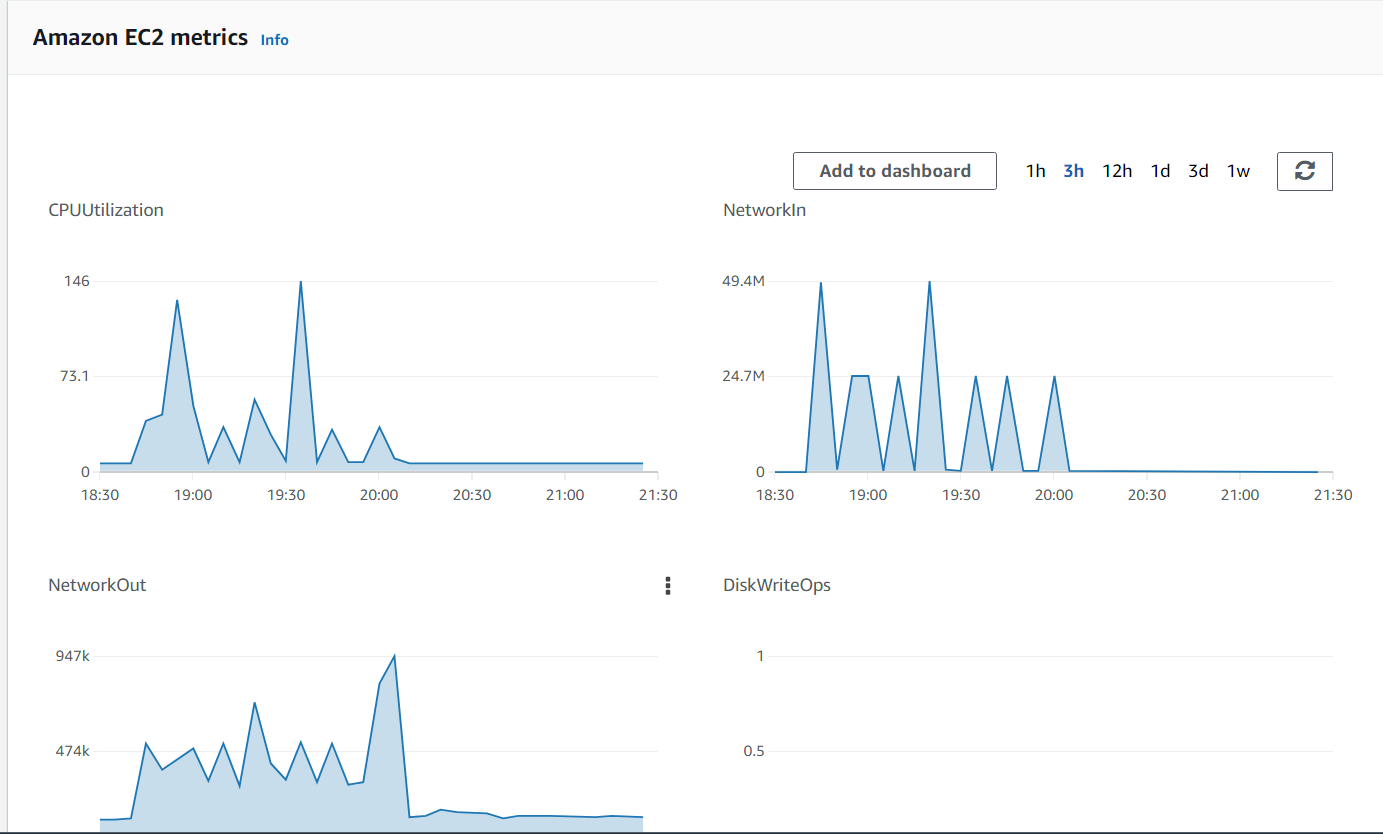
****

****

metrics monitoring and alarms are set on Cloudwatch:

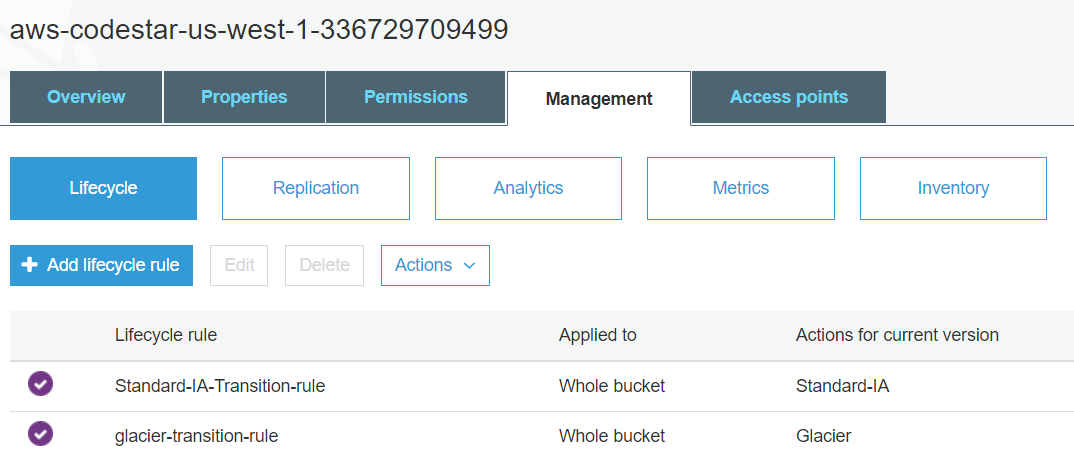
****

****

****

1. **S3 and CloudFront backend architecture –**

Lifecycle policy created for both Standard IA and Glacier storage for 75 and 1year respectively:

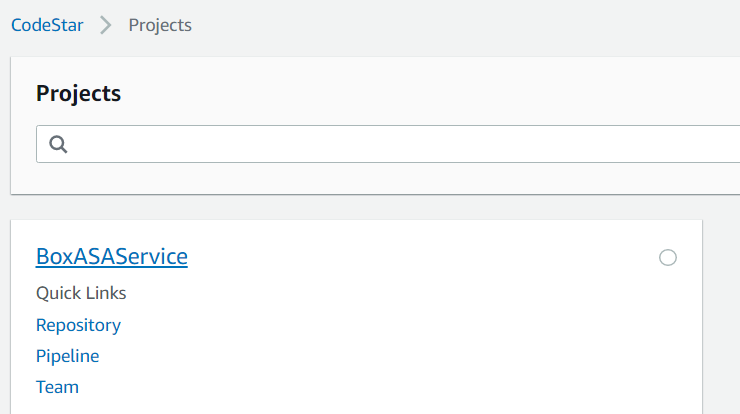
****

CloudFront created:

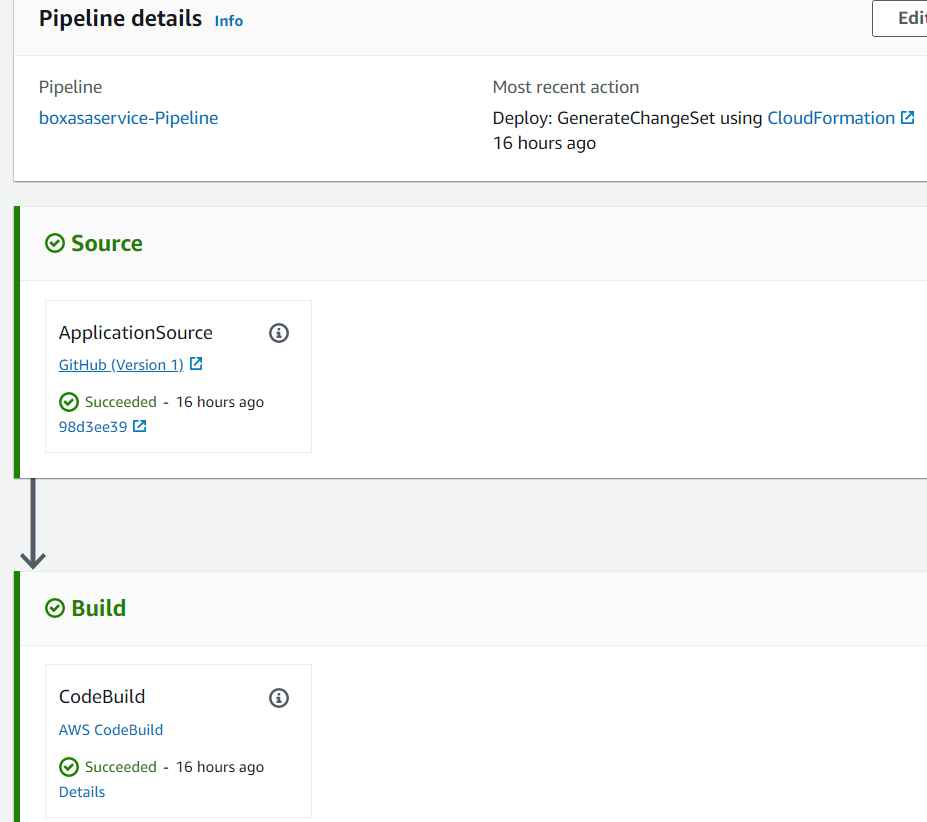
****

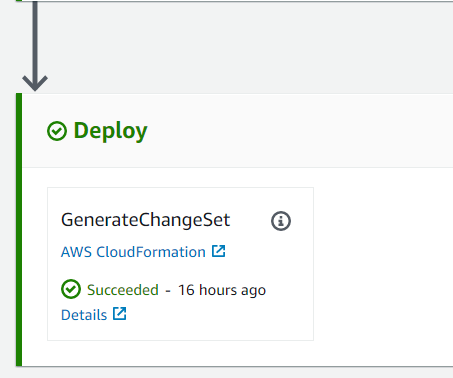
1. **CodeStar Project-**

Have created a codestar project:

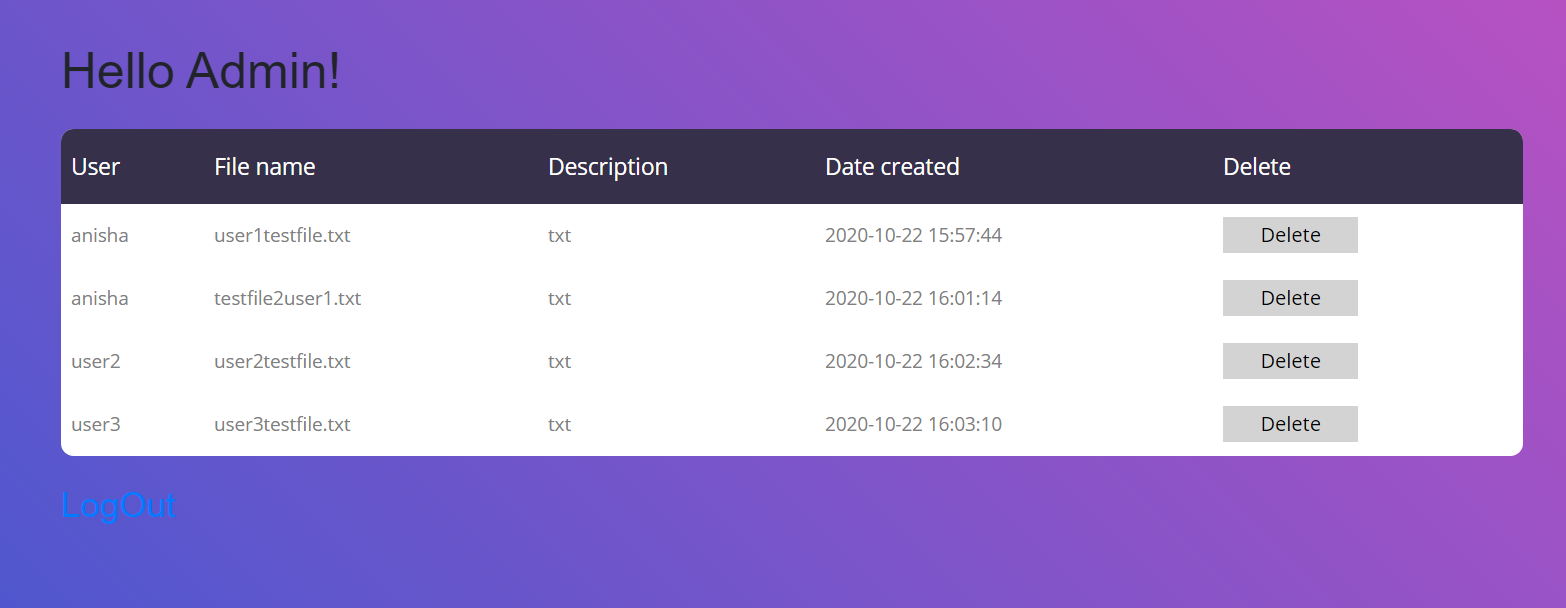
****

1. **CI/CD using CodeStar Pipeline:**

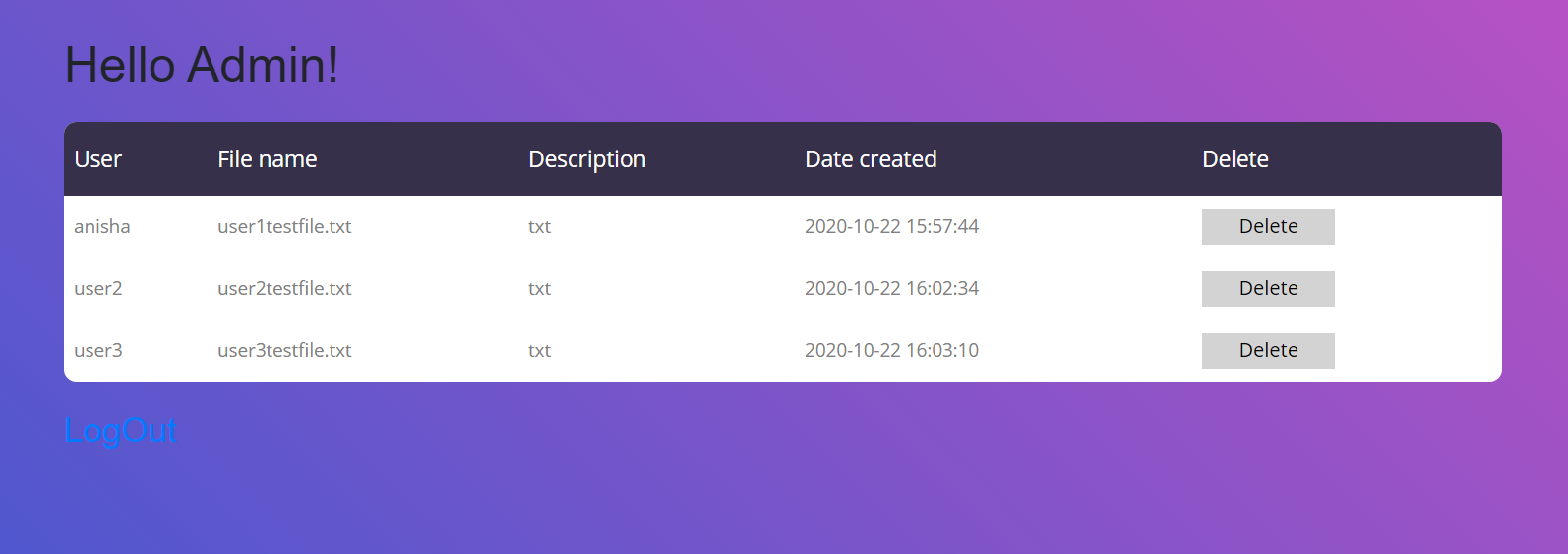
****

****

1. **Admin Panel where Admin has an authority to delete all files:**

****

Leverage/rights to delete the file of the user:

****

**REFERENCES:**

UI Template:

1. <https://colorlib.com/etc/lf/Login_v4/index.html>

AWS Services:

1. <https://www.youtube.com/watch?v=BeOKTpFsuvk>
2. <https://www.youtube.com/watch?v=Tkkb4mcuGSY>
3. <https://www.youtube.com/watch?v=2wyi70l3WM0>