Harshil Vyas

CMPE 281 - Project #1

Website:

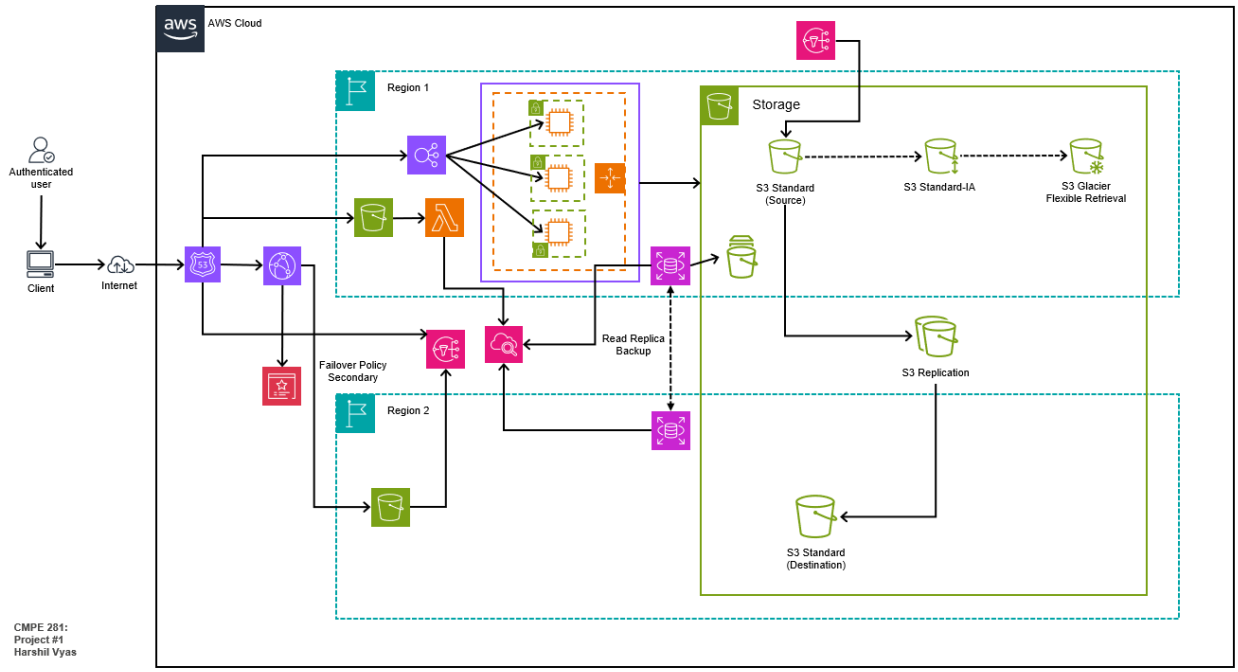
http://frontend.harshilvyas.com/

GitHub URL:

https://github.com/Harshil-V/CMPE281-Cloud-Project-1

Demo Video:
https://drive.google.com/file/d/1t6Ndc_xBMRXhCd9y1Ll69MatW16a9PGx/view?usp=sharing

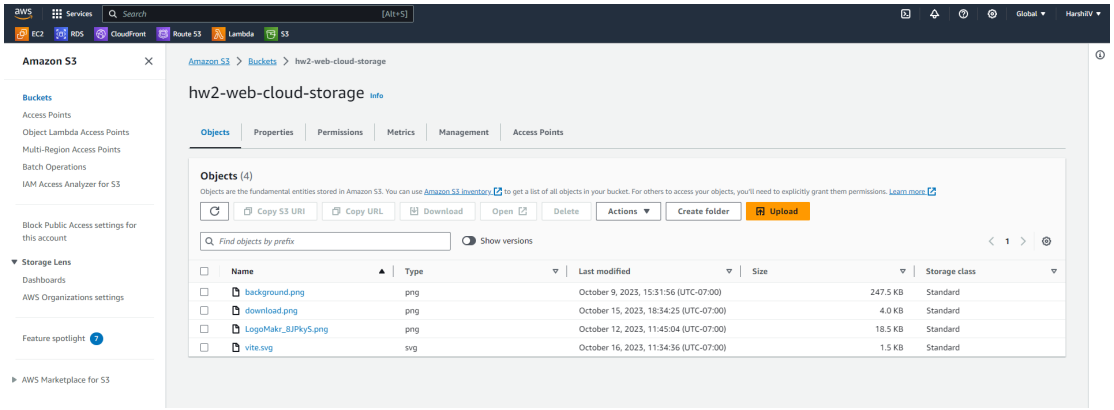
System Architecture:

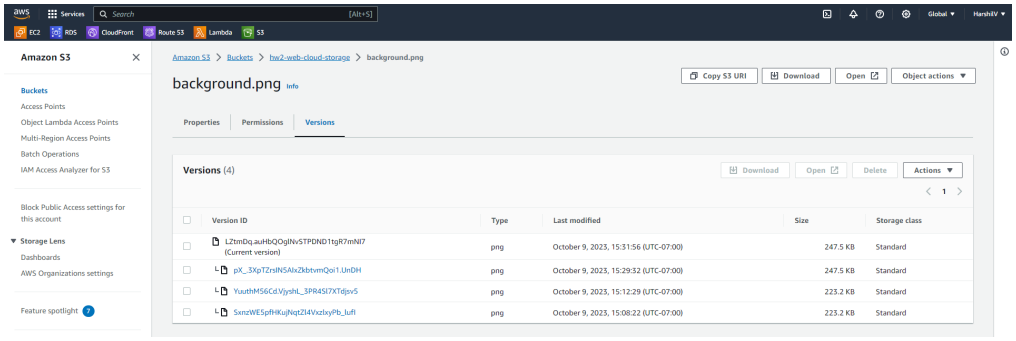


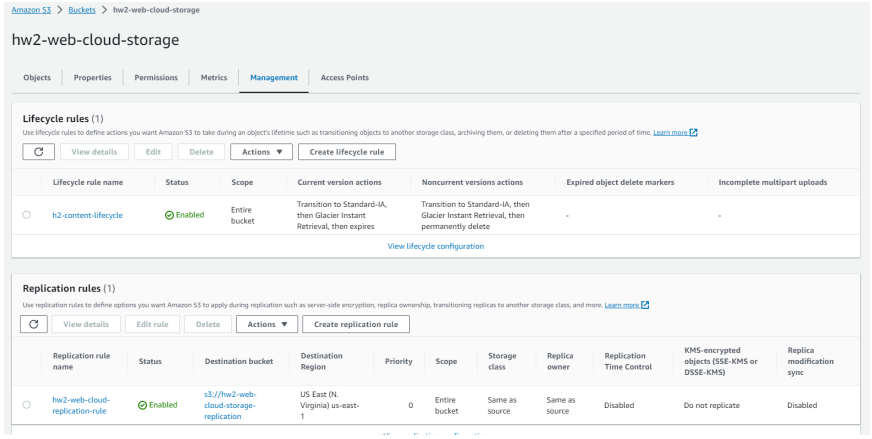
S3 Buckets Image:

Here the image where all the files are getting upload to, and the following image shows the different versions of file that were uploaded, for high durability.

Harshil Vyas



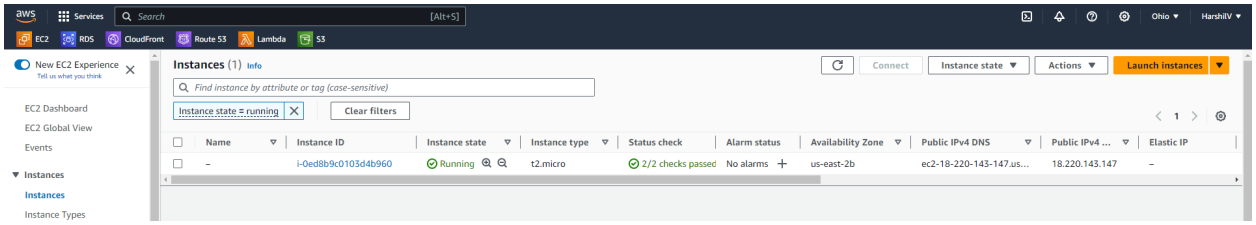




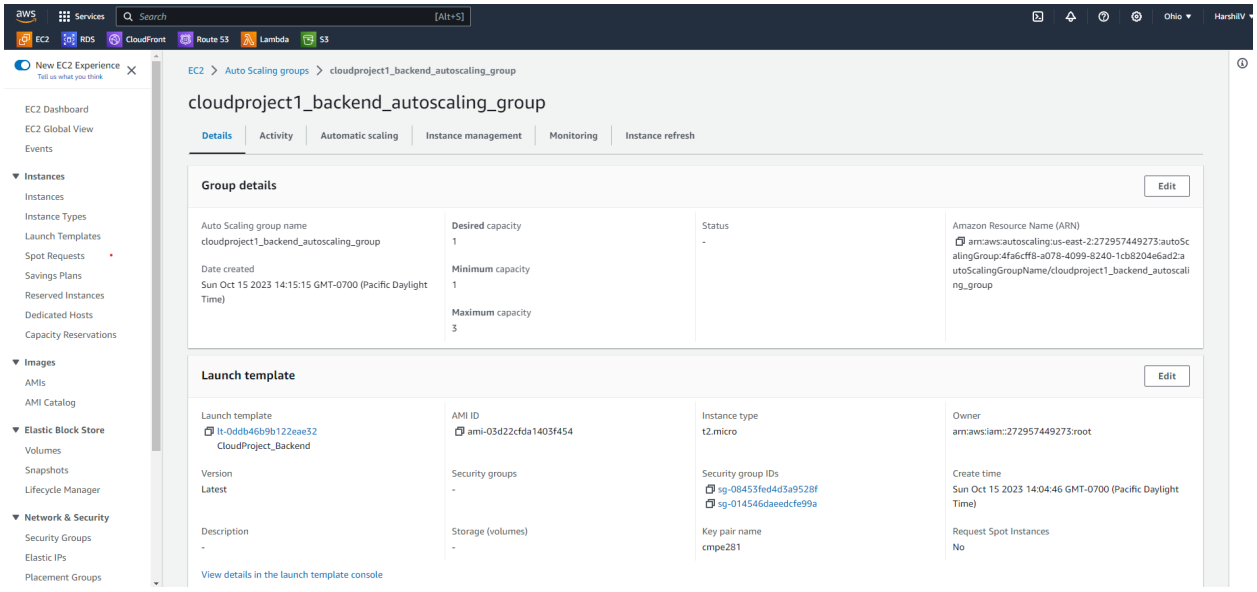
Harshil Vyas

EC2 (Backend):

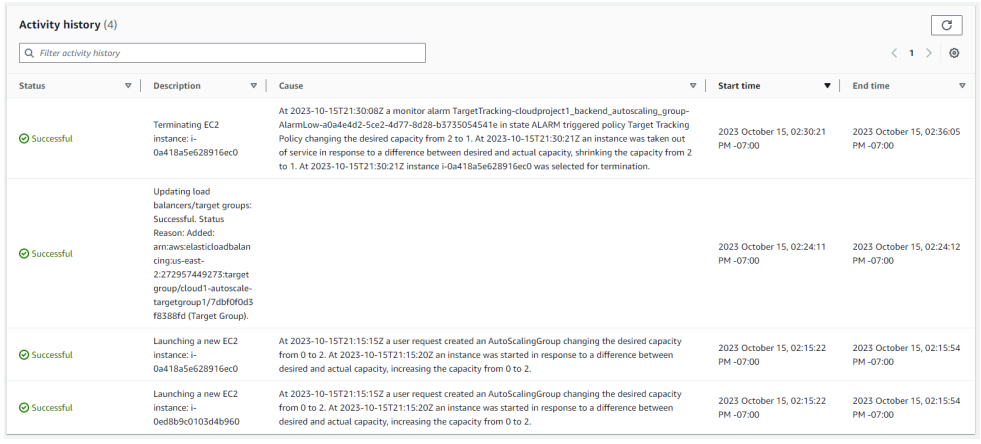
The Image below is one of the EC2 Instances that will be running my backend service.



Here is the image of the configuration for the autoscaling group of my EC2 instances, depending on the workload of the instance it will spin up more instances for the traffic.



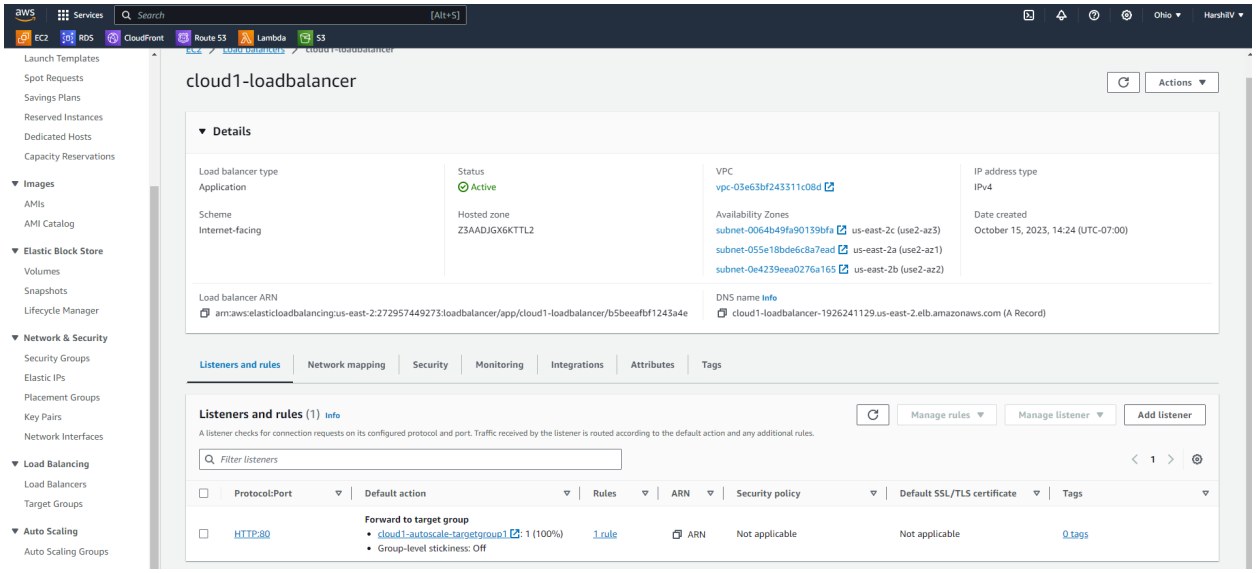
Here is the activity history that launched more EC2 instances and terminated as required.



Harshil Vyas

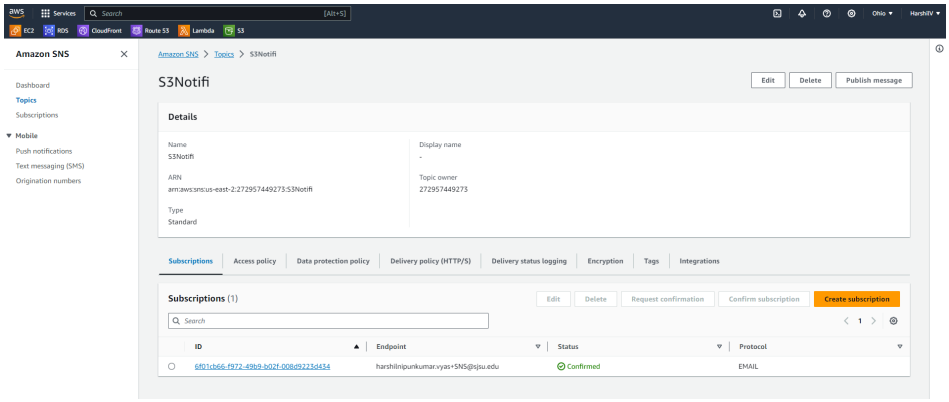
Load Balancer:

I have high availability and durability, we have a load balancer for the EC2 Instances, and we have 3 availability zones to make the application easily accessible.



SNS:

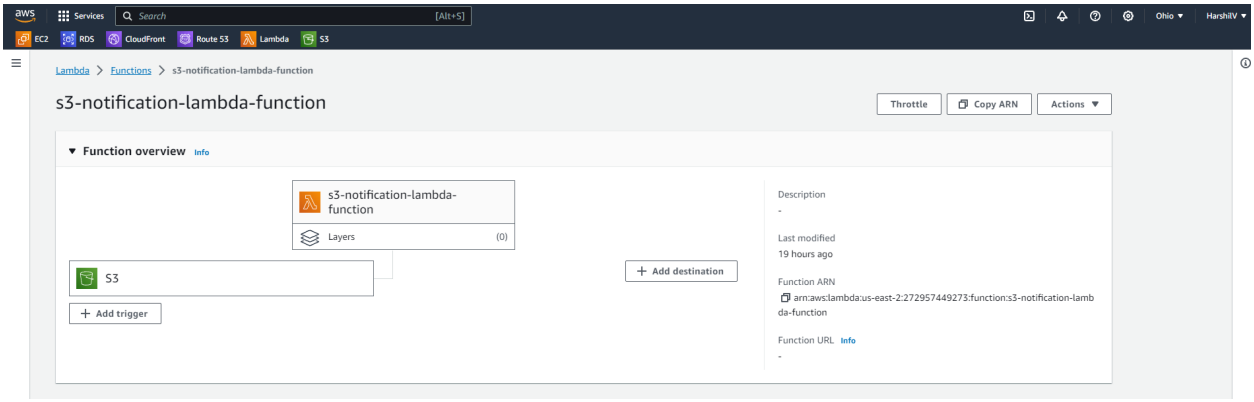
When a file operation occurs in the target S3 bucket, an SNS email notification is sent out, as seen below.



Harshil Vyas

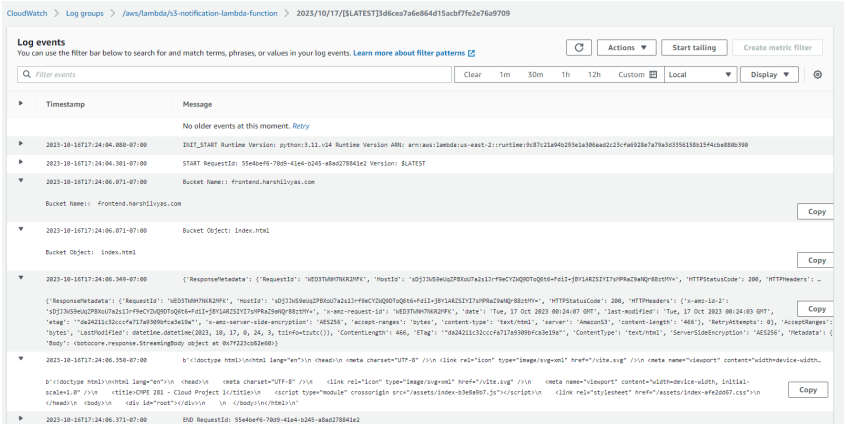
Lambda:

Whenever an S3 Bucket hosting the frontend website gets an update, the uploaded content triggers a lambda function that logs which file was added into which bucket and reads the file content.



CloudWatch:

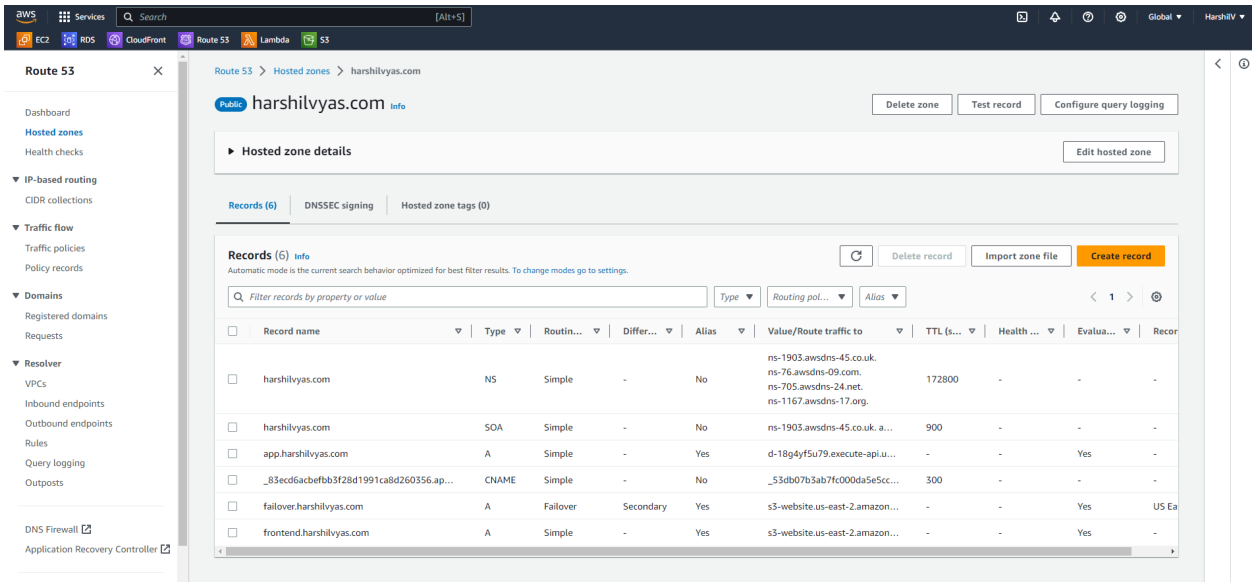
Here is an example of CloudWatch logs that are triggered when the lambda function runs.



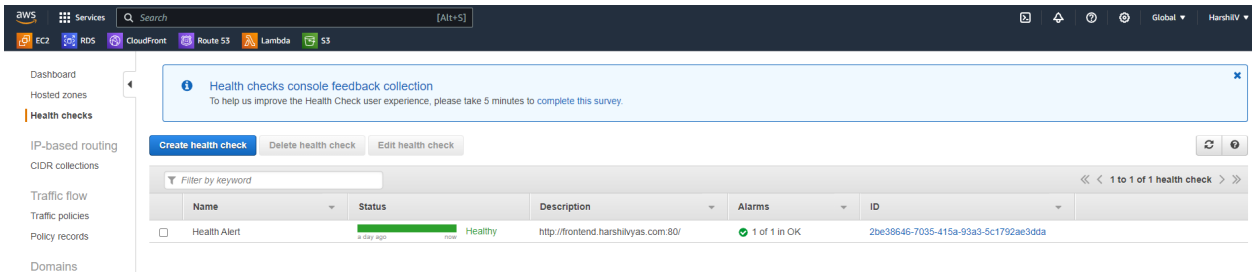
Harshil Vyas

Route 53:

Below is the image of the hosted zone for my website with all the records, for my primary and secondary websites.

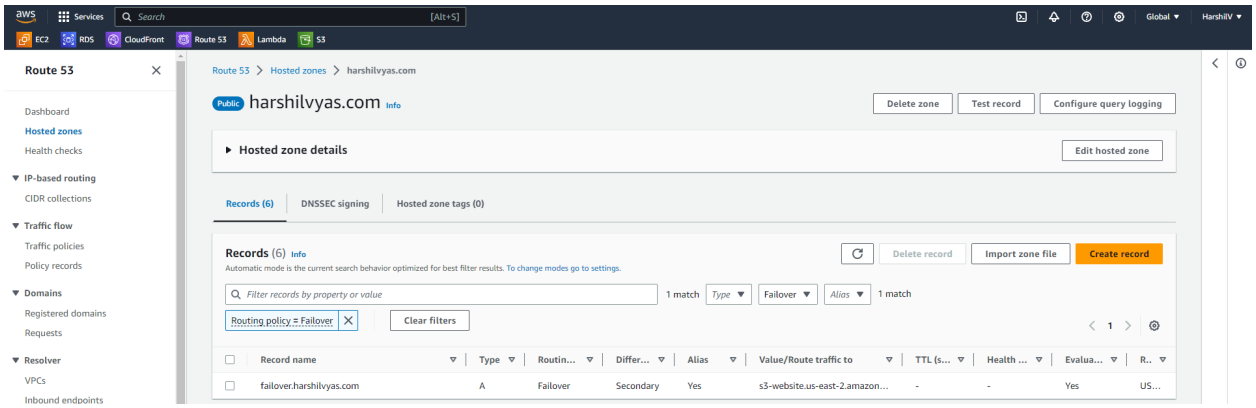


Here is the Health Check to make sure the website is functioning, otherwise an alert is sent out.



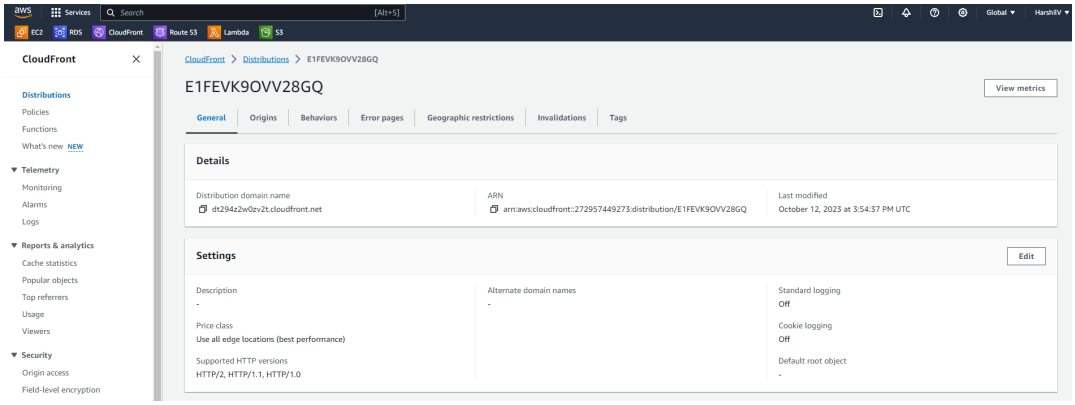
Harshil Vyas

Failover Policy:

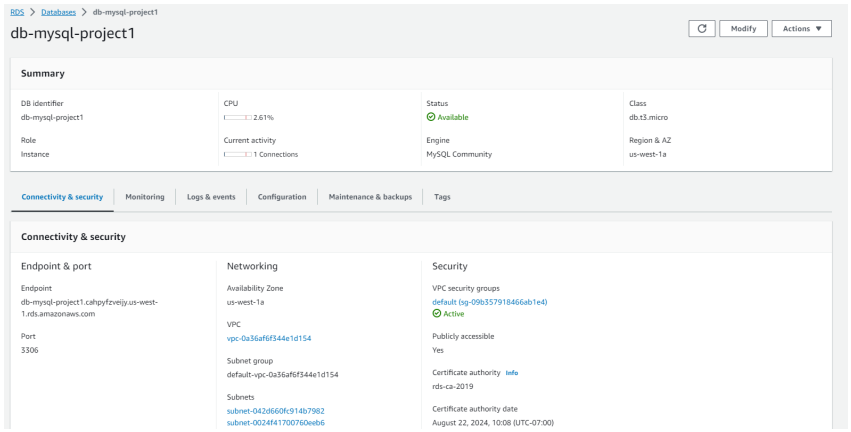


CloudFront:

Here is my CloudFront distribution for my S3 bucket to allow high availability, to make sure that we can reduce the load on the origin when requesting for images.

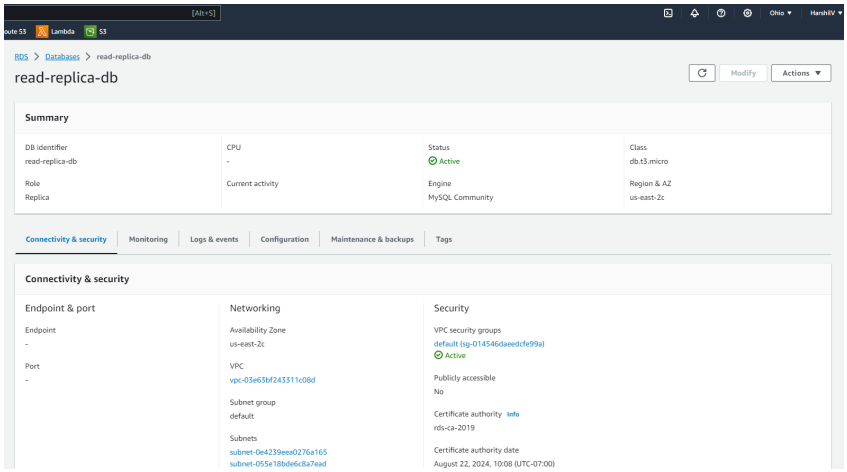


RDS (MYSQL):

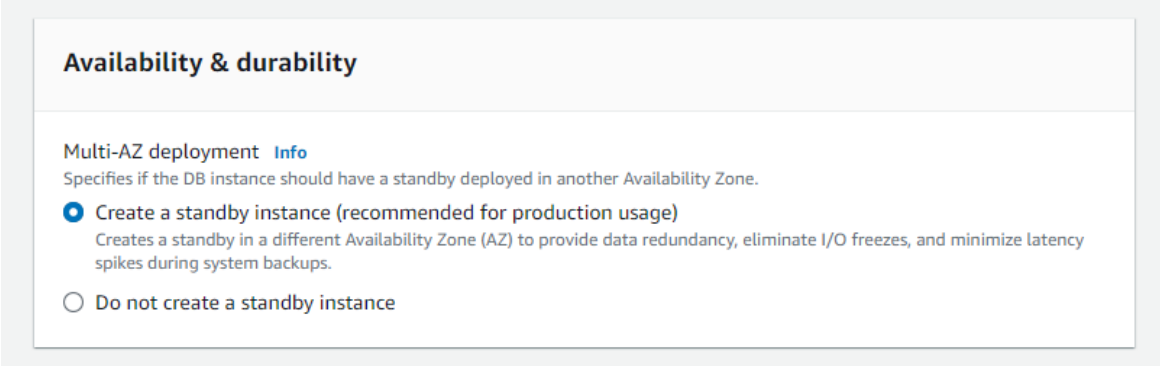


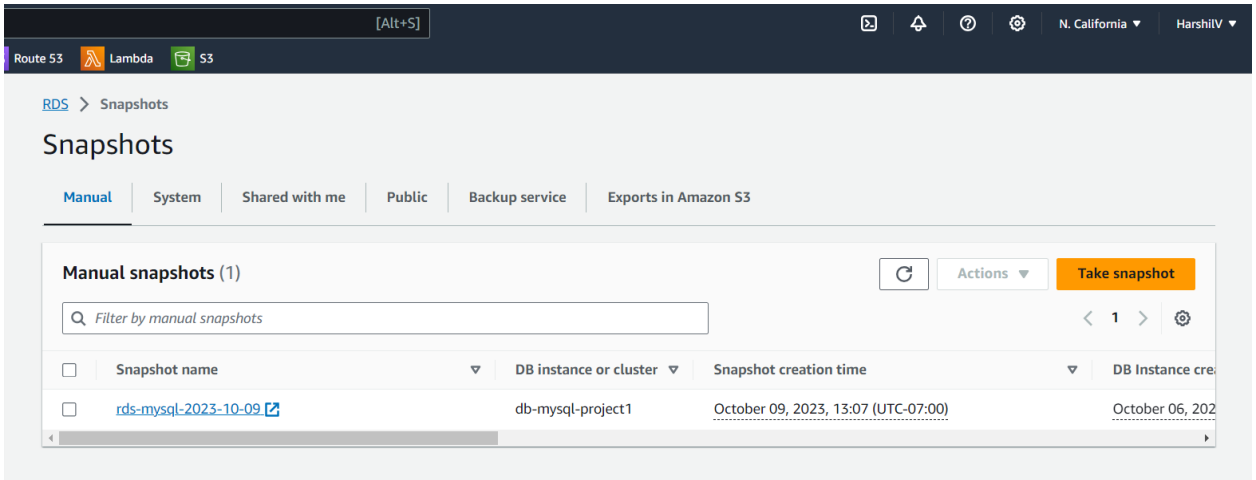
Harshil Vyas

To make my application data highly available, I have a read replica in a different region.



The duplicated database is used to ensure the availability and durability of data, making it accessible in different regions. It can be helpful when the production database has issues or fails.



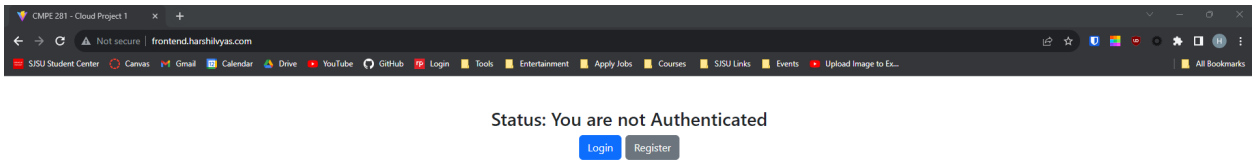


Harshil Vyas

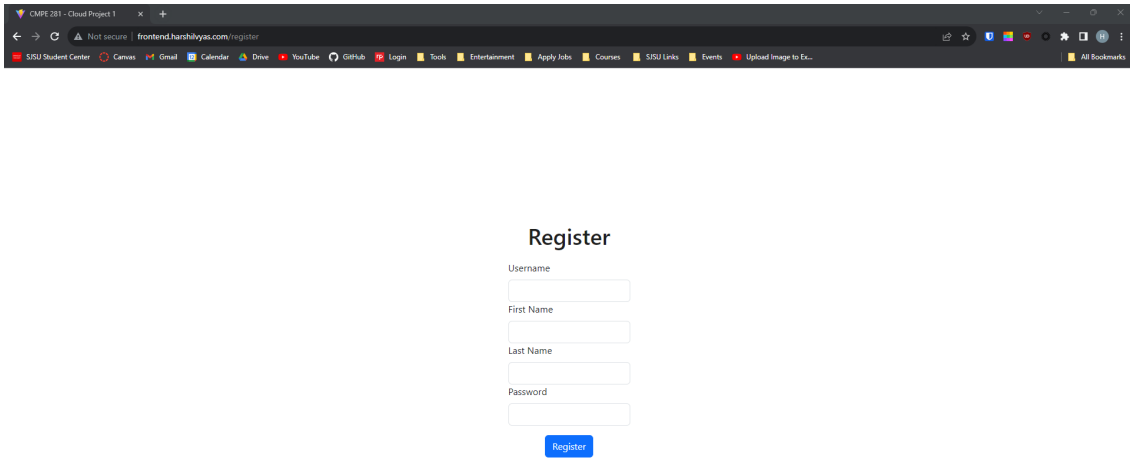
DR Measures:

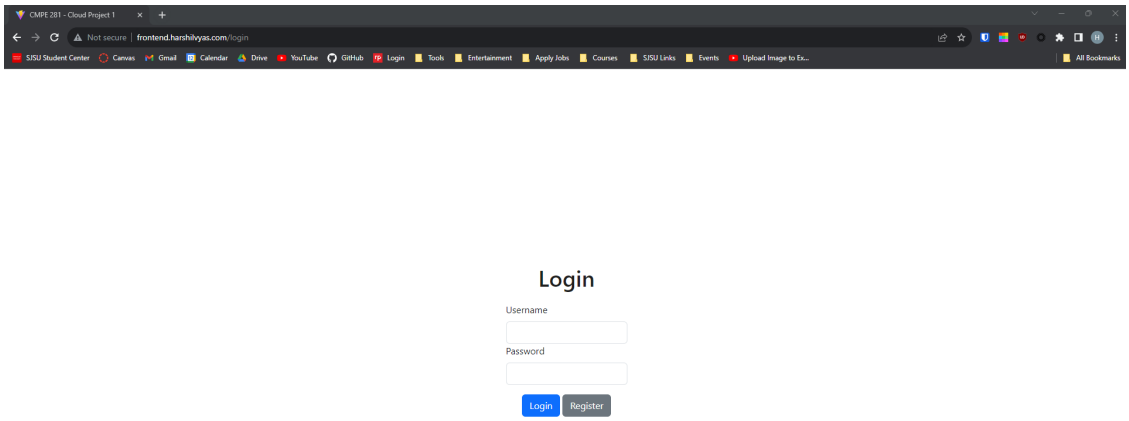
- We have set up the RDS Database to have read replicas with multi-AZ deployment in cases where the production data fails, or other problems.
- The RDS Database has a daily backup copy policy to easily restore data, plus I have taken some manual snapshots whenever I want a backup of a particular point in time.
- If the primary website (frontend.harshilvyas.com) fails, there is a failover policy in Route53 to point to another website (failover.harshilyas.com) in such cases.
- In S3 Buckets versioning enables tracking of all the different files with the same name that are uploaded.
    - Alongside the S3 buckets is the lifecycle and replication policy of data in another bucket in a different region (based on Homework #2)

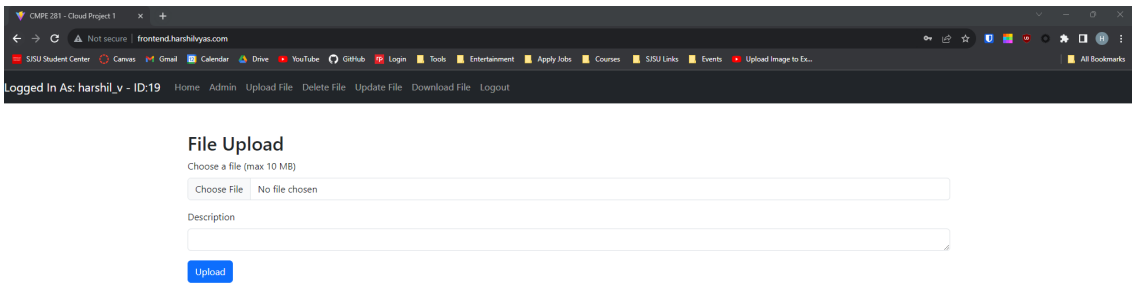
Screenshot of Web Application:



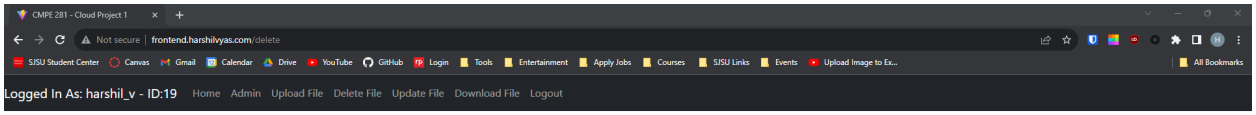
Harshil Vyas







Harshil Vyas

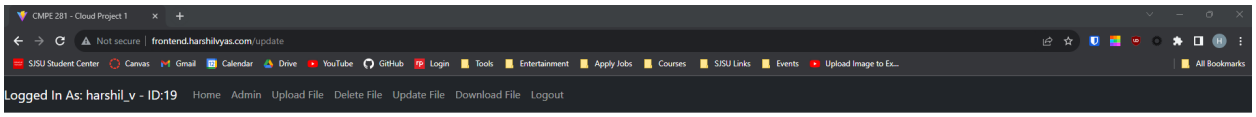


Delete File From S3 Bucket

File Name / Object Key

Please Enter File Name / Key

Delete File/Object



File Update

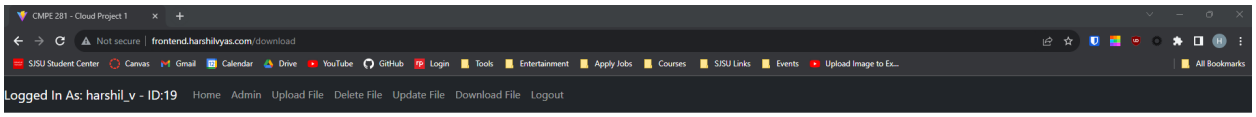
Choose a file (max 10 MB)

Please Make Sure File Name / Key Already Exists

Choose File | No file chosen

Description

Upload

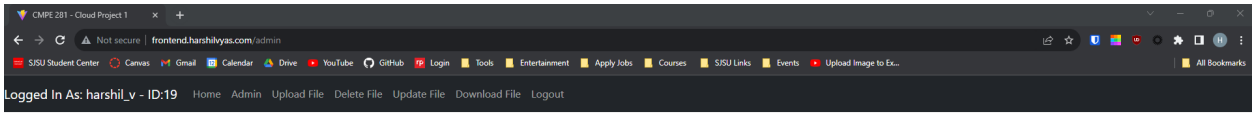


Download File

File Name / Object Key

demo_image.jpg

Request File | Download File



Admin Page

User ID	User Name	Full Name	Created At	Updated At	File Name / KEY	Description	Action
7	TestAcc	Test Account	10/9/2023, 8:12:28 AM	10/9/2023, 3:29:31 PM	background.png	new new	Delete
8	harshil	Harshil Vyas	10/12/2023, 4:45:02 AM	10/12/2023, 4:45:02 AM	LogoMakr_8JPkyS.png	logo	Delete
7	TestAcc	Test Account	10/15/2023, 6:33:54 PM	10/15/2023, 6:33:54 PM	download.png	home	Delete
7	TestAcc	Test Account	10/16/2023, 11:32:29 AM	10/16/2023, 11:34:36 AM	vite.svg	new	Delete