Name/id: abdelkareem yousef mamdoh soubar/19110022
Assignment Title: Cloud Computing AWS
Course Tutor: Eng. Moath Malkawi
Submission Date: 26 – Jan - 2022

Task 1
video : https://drive.google.com/drive/folders/1Uyd-Lp6cTwdQBNAJczTTTfyrJJMe6vkn?usp=sharing

part11
in general:
AWS automobile Scaling ceaselessly analyzes your applications and changes capability pro re nata to produce consistent, sure performance at rock bottom possible value. It's straightforward to make up application measurability for varied resources across several services in minutes exploitation AWS automobile Scaling. Your applications can continually have the right resources at the correct time due to AWS automobile Scaling.
in this task the policy for the load blancer I use is that if a cpu uses more than 60% a new instance would show up and have traffic come to it.

An Application Load Balancer routes requests to 1 or additional ports on every instrumentality instance in your cluster, creating routing selections at the applying layer (HTTP/HTTPS). It conjointly supports path-based routing and will route requests to 1 or additional ports on every instrumentality instance in your cluster. Dynamic host port mapping is supported by Application Load Balancers.

In our case:

It also helps in case of failure or false health check that the it would keep the traffic a way from and redirect it to the working server.

As for auto scaling it is place in our case is if any of the devices need extra storage it would provide it and it keeps it eyes on the application in case it needs any more/less and it tries its best to operat under the lowest cost possible .

everything had been placed in side the video but the AWS well architected framework pillars amazon had placed methods ro insure the well architect when it comes to using their cloud.
Pillars are:

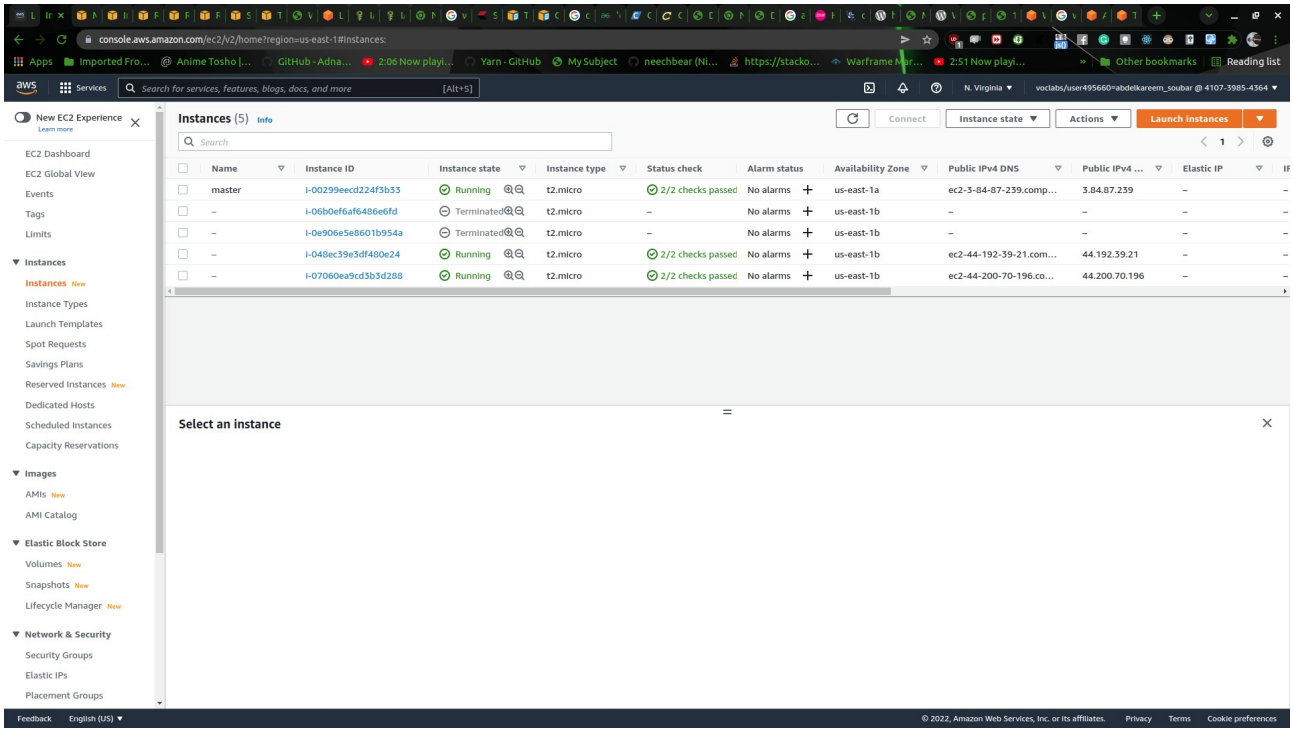
1. Operational Excellence
    - Perform operations as code
      we only used bash for code and we did setup sync through it, but we didn't change any of the settings on the cloud.
    - Make frequent, small, reversible changes
    - Refine operations procedures frequently
    - Anticipate failure
      we are using health check if it failed it would direct traffic to other servers
    - Learn from all operational failures
2. Security
    - Implement a strong identity foundation
      we were not allowed to create any IAM roles
    - Enable traceability
    - Apply security at all layers
      we did apply security layers by using vpc /security groups /subnets private and public
    - Automate security best practices

    - Protect data in transit and at rest
      the data in connection is encrypted

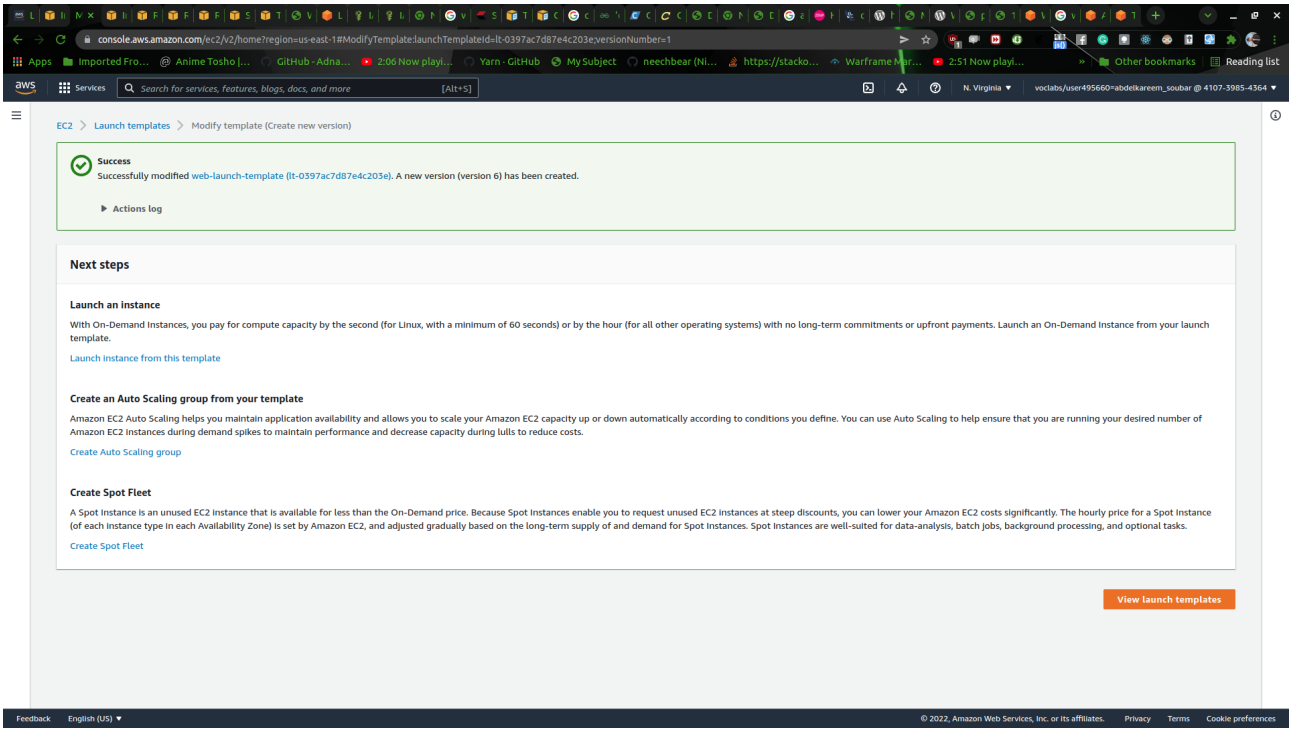
- Keep people away from data
  we are using cloudfront that hides the internal connections
- Prepare for security events
  we did place security groups that control the traffic (I only receive from my ips)

3. Reliability
   - Automatically recover from failure
     we have a load balancer in case any failure happens it lunches a new instance plus everything is placed on s3 code bucket that we made.
   - Test recovery procedures
   - Scale horizontally to increase aggregate workload availability
     when more work load occurs the load balances would launch more instances
   - Stop guessing capacity
     most of what we have auto scaling enables so if we need more we will get more and pay for it.
   - Manage change in automation
     we don't have any code related to the service it self but we did do sync in bash.
4. Performance Efficiency
   - Democratize advanced technologies
   - Go global in minutes
     yes we are using edge locations through cloudfront
   - Use serverless architectures
   - Experiment more often
     we have a heath check running all the time plus we have placed tones of effort in test our service before the final launch
   - Consider mechanical sympathy
5. Cost Optimization
   - Implement cloud financial management
     we tried to use everything that was free for this task
   - Adopt a consumption model
   - Measure overall efficiency
     its externally efficient and it uses about 1.5$ per day with task4
   - Stop spending money on undifferentiated heavy lifting
     it would use more money in case of having traffic because it opens more instances
   - Analyze and attribute expenditure
     because this is a task there would would be no traffic but in real life the first one million requests are free.

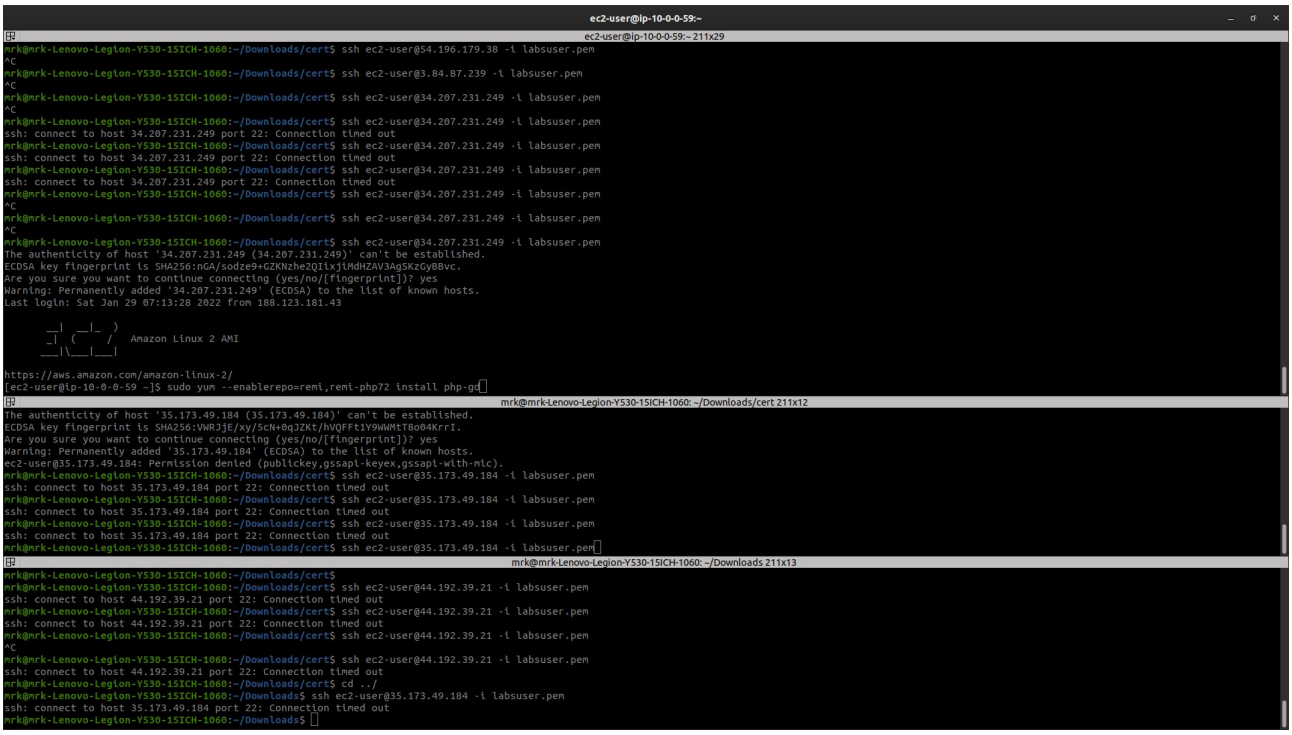
terminating 2 instances so the would load balancer would open 2 new one’s with latest update on them.



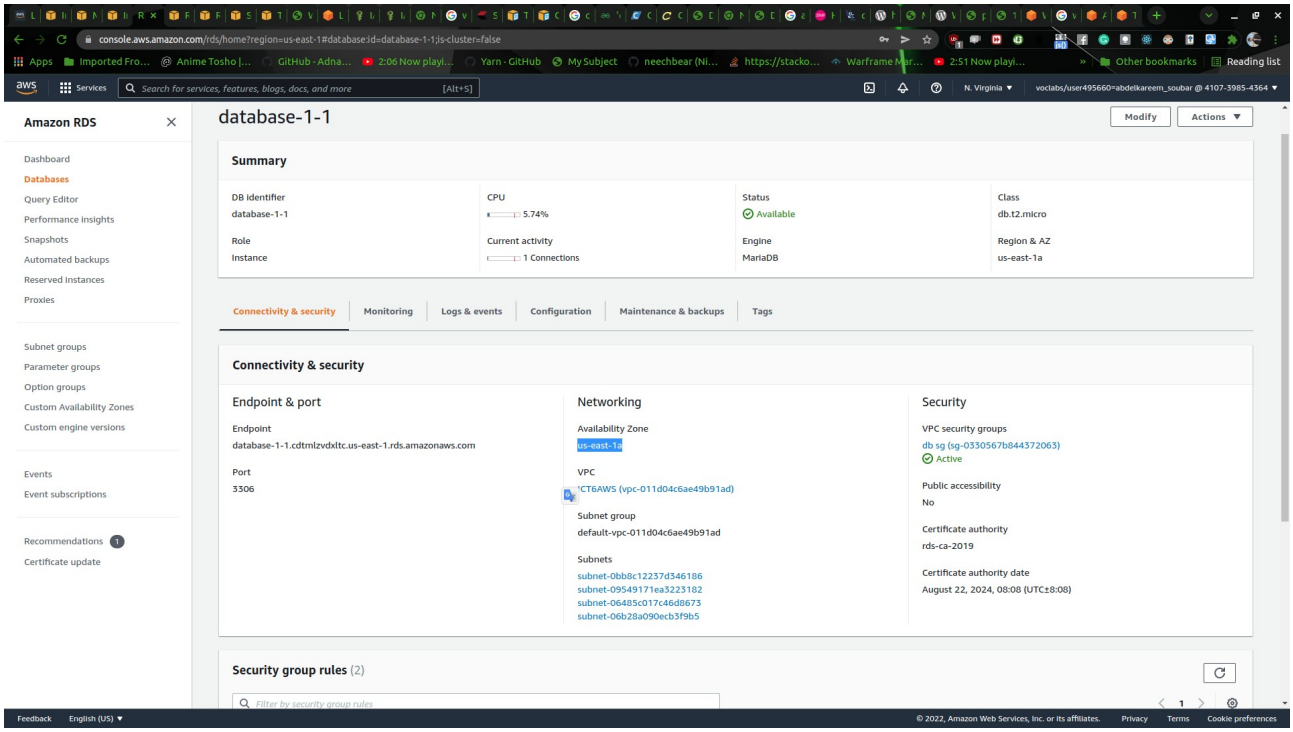
Finished creating new template (sources) while testing



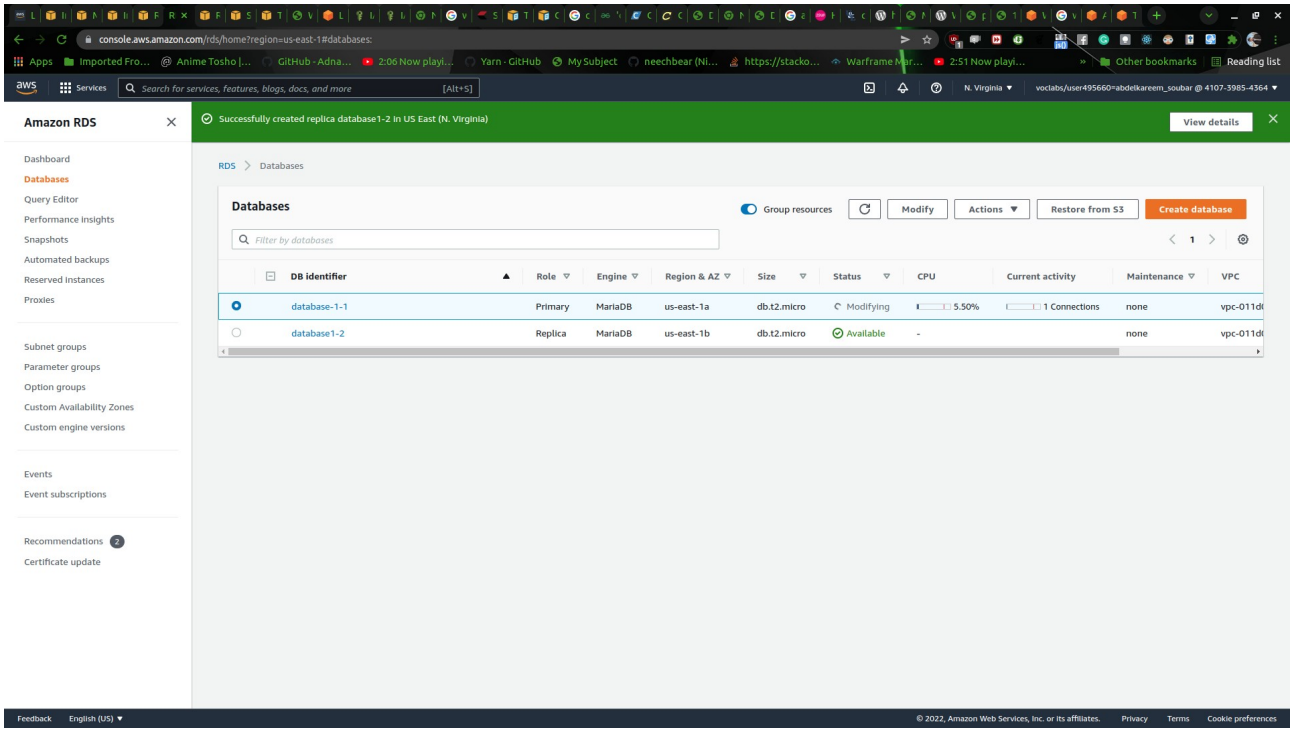
here we are testing maser and slaves



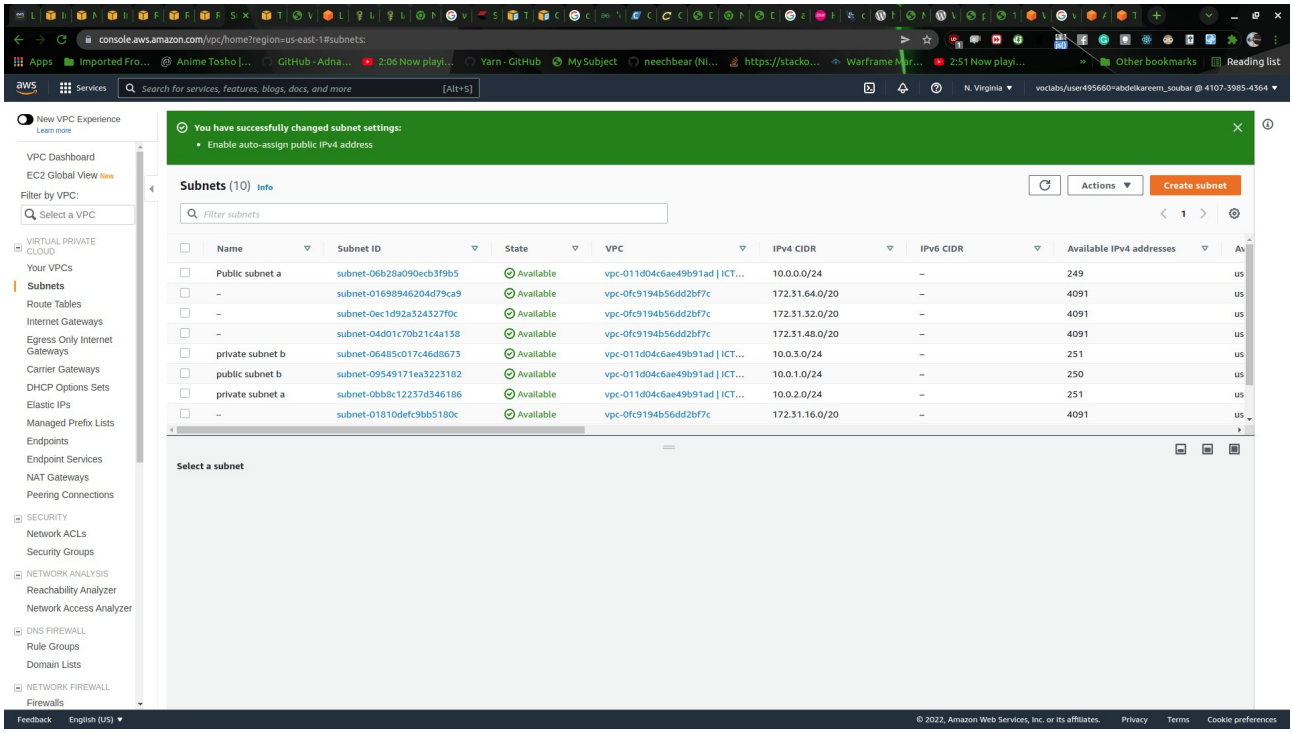
database creation



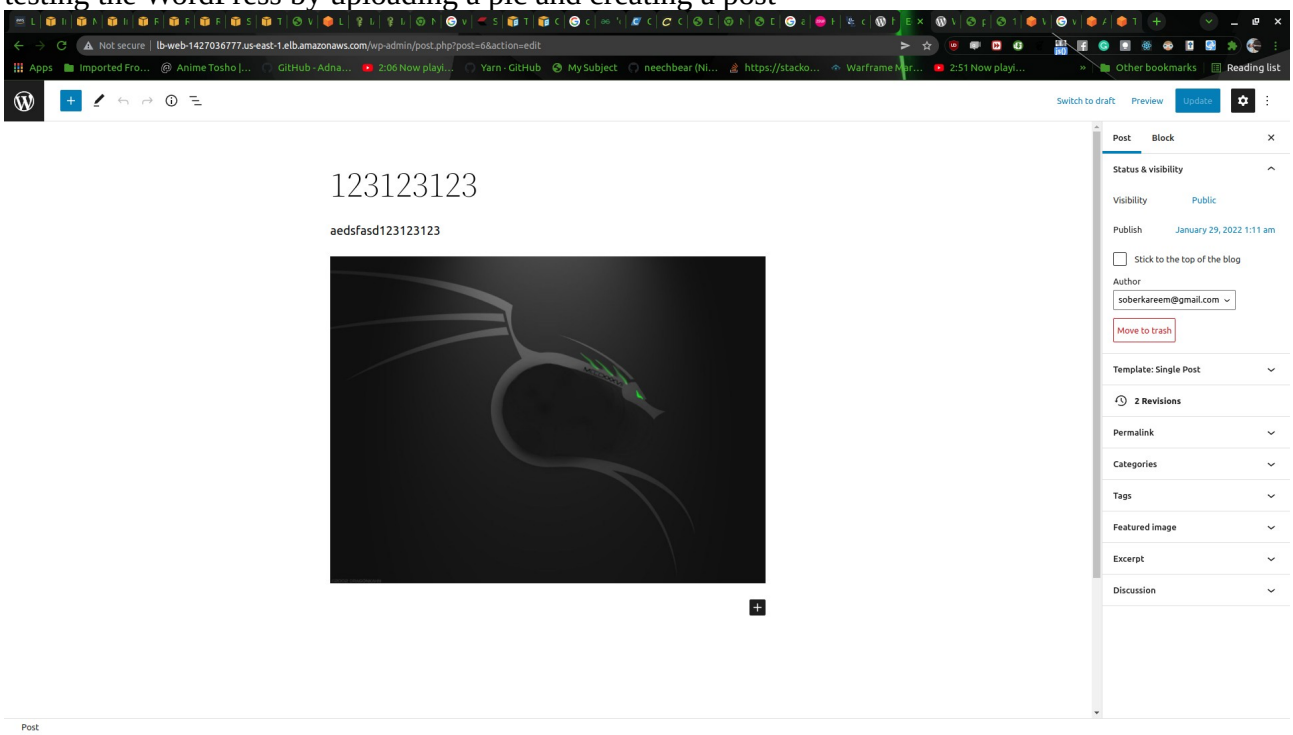
finishing off the creation of the replica



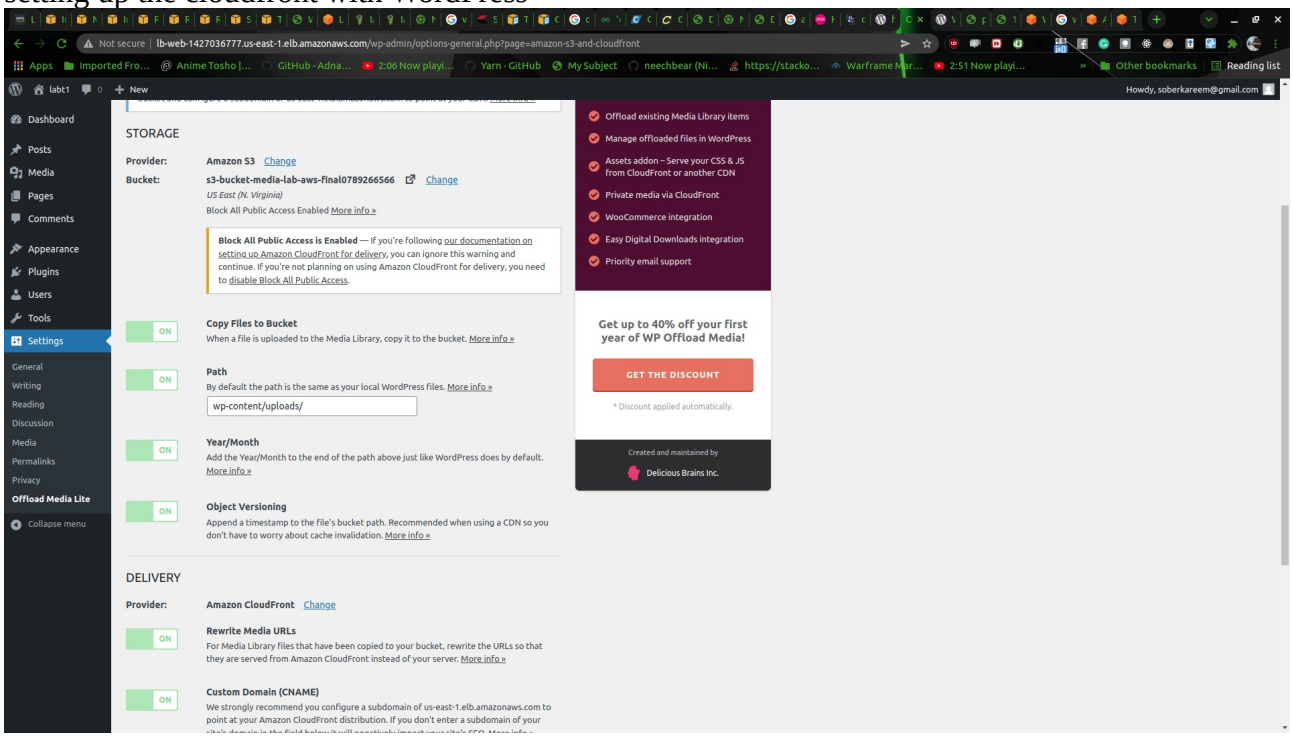
subnets

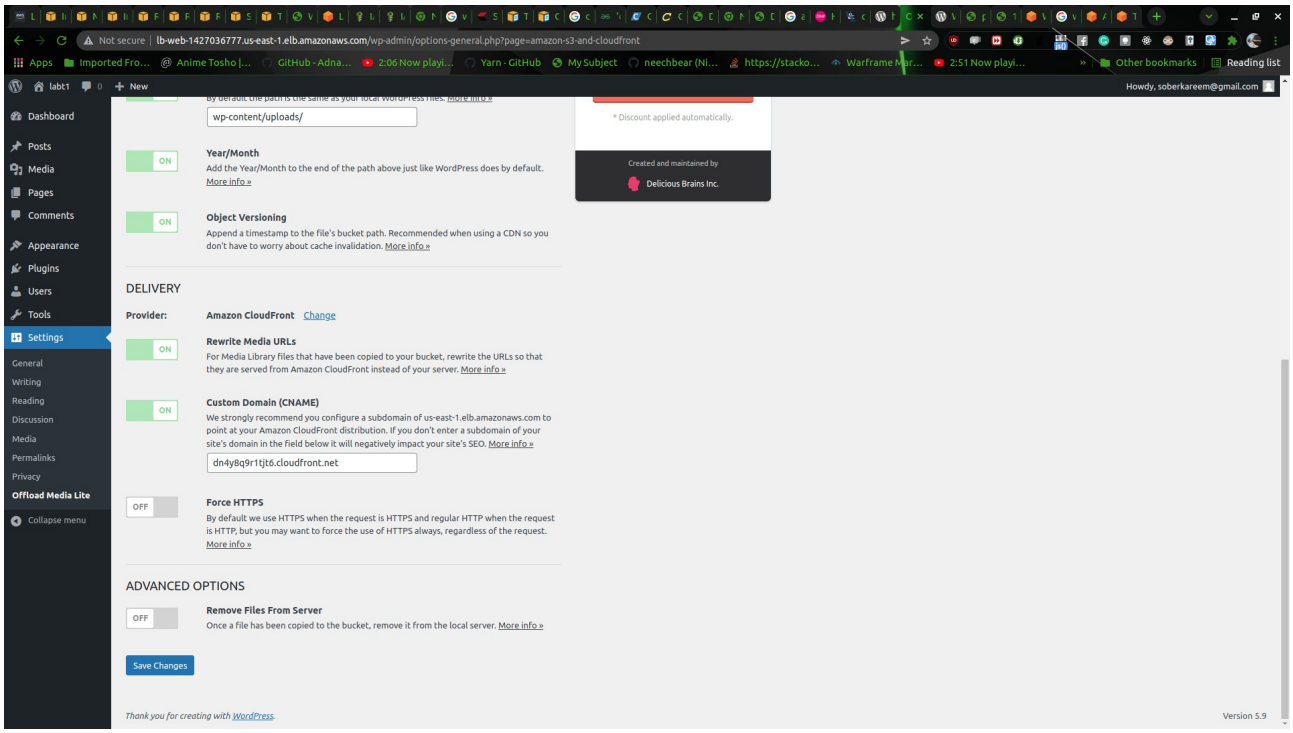


testing the WordPress by uploading a pic and creating a post



setting up the cloudfront with WordPress




Not secure | lb-web-1427036777.us-east-1.elb.amazonaws.com/wp-admin/options-general.php?page=amazon-s3-and-cloudfront
Howdy, soberkareem@gmail.com
Dashboard
Posts
Media
Pages
Comments
Appearance
Plugins
Users
Tools
Settings
General
Writing
Reading
Discussion
Media
Permalinks
Privacy
Offload Media Lite
Collapse menu
wp-content/uploads/
ON
Year/Month
Add the Year/Month to the end of the path above just like WordPress does by default.
More info »
ON
Object Versioning
Append a timestamp to the file's bucket path. Recommended when using a CDN so you don't have to worry about cache invalidation. More info »
DELIVERY
Provider:
Amazon CloudFront Change
ON
Rewrite Media URLs
For Media Library files that have been copied to your bucket, rewrite the URLs so that they are served from Amazon CloudFront instead of your server. More info »
ON
Custom Domain (CNAME)
We strongly recommend you configure a subdomain of us-east-1.elb.amazonaws.com to point at your Amazon CloudFront distribution. If you don't enter a subdomain of your site's domain in the field below it will negatively impact your site's SEO. More info »
dn4y8q9r1tjt6.cloudfront.net
OFF
Force HTTPS
By default we use HTTPS when the request is HTTPS and regular HTTP when the request is HTTP, but you may want to force the use of HTTPS always, regardless of the request.
More info »
ADVANCED OPTIONS
OFF
Remove Files From Server
Once a file has been copied to the bucket, remove it from the local server. More info »
Save Changes
* Discount applied automatically.
Created and maintained by
Delicious Brains Inc.
Thank you for creating with WordPress.
Version 5.9

Task 2

EC2

Category	Designed for
A1 (Amazon EC2 A1 instances)	It was made for saving cost while scaling out also for Arm based applications such as a web server and in extend to that it also can distribute data stores which are supported by extensive Arm ecosystem.
M4 instances	It’s designed for getting a great balance between memory , compute and network resources, and it suites workloads like web server, database...
T3 instances	T3 instances has a base line for CPU usage that has the ability to burst at any time, it was made for applications that need CPU usage and can get spikes from time to time.
T4g instances	It was made for running general purpose workloads with low cost in mind, this instance comes with a base line for CPU performance to do common workload but it can go above the baseline when for time when more performance is required.
Mac instances	It gives a stable environment high performance and secured for developers with workload, and it makes the integration with AWS easy for using AWS Lib’s and tools plus lunching configurations.

EC2

Purpose	Recommended EC2 Instance type category	Why?
Hosting a web server	M5	balances
Multiplayer gaming	c5n	High-performance and low latency
Data analytics	R5	It provides high memory plus increased CPU performance
Speech recognition	p3	It can do hardware acceleration with GPU

S3 storage

Storage Class:	Designed for:
S3 Standard	Its a storage the provides a low latency, high-performance and very reliable storage its used for web hosting and content distribution and more .
S3 Intelligent-Tiering	A cost-benefit analysis is a process used by businesses to evaluate the potential rewards and costs associated with a given decision. The

	analyst sums up the expected returns and then subtracts the total costs. The ability to store and access knowledge in a variety of ways enables organizations to work seamlessly across different platforms and applications.
S3 Standard-IA	It was not designed for connect too much to it but on the contrary its used for low usage but when its being used it give high performance and low latency and durability , and it suitable for backups or restoring data in case of disaster
S3 One Zone-IA	Its the same as s3 standers IA but its saved in one zone
S3 Glacier	long-term storage of information that's sometimes accessed and that retrieval latency times of three to five hours ar acceptable
S3 Glacier Deep Archive	Customers who maintain data sets for 7-10 years or longer to meet client needs and regulatory compliance requirements, particularly those in the financial services, healthcare, media and entertainment, and public sector.

Type of Data	Recommended Amazon storage class	Why?
Mobile game data	S3 Standard	it provides low retrieval, low latency and high throughput
Disaster recovery data	S3 IA	It has a higher data retrieval cost but its also half the s3 Standard cost
Data that may or may not be used often	S3 Intelligent-Tiering	automatically lowers your storage prices on a per-object basis by relocating information to the foremost cost-efficient access tier reckoning on access frequency.
Media archives	Glacier	highest performance, the most retrieval flexibility, the lowest cost archive storage and optimized for different access patterns

Task 3
Data availability: refers to the timeliness and consistency with that information is also accessed and used. information accessibility is a component of it.

Data persistence:meaning the data outlasts the process that generated it In AWS, information persistence is done by combining compute and storage services.

Data encryption : Data secret writing may be a security mechanism during which knowledge is encoded and solely someone with the proper encryption key could access or decipher it.

Snapshots square measure progressive backups, which suggests that they solely preserve the blocks on the device that have modified since the last pic.

EBS

Data availability : The Amazon EbS volumes square measure engineered to be extremely accessible, dependable, and long. Amazon EbS volume knowledge is duplicated across several servers in associate convenience Zone at no extra price to you, preventing knowledge loss because of the failure of any single part.

data encryption : By encrypting your knowledge volumes, boot volumes, and snapshots, EBS coding offers knowledge at rest security. Keys that are handled by Amazon or keys that you create and control using the AWS Key Management Service (KMS).

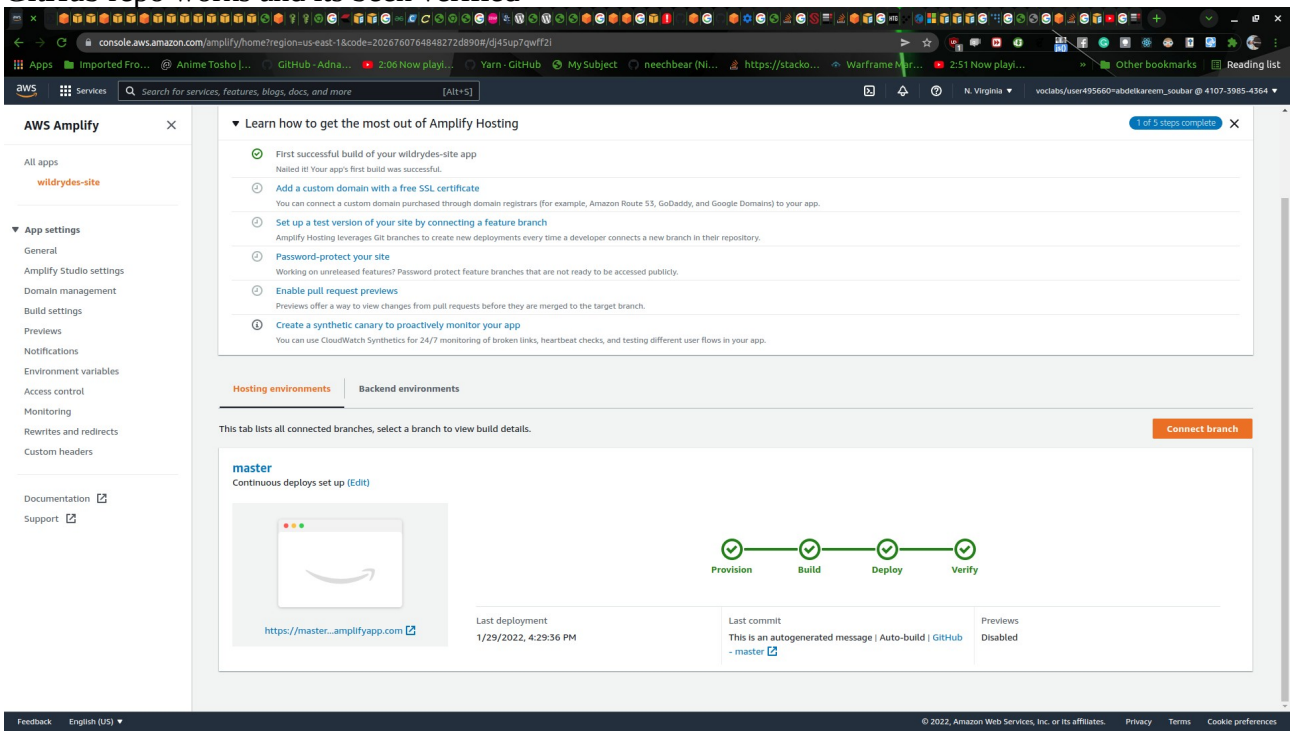
snapshot : Amazon EbS permits you to save lots of snapshots of your volumes to Amazon S3 at any moment. solely the blocks that have modified since your previous photograph area unit unbroken in Amazon EbS Snapshots, and you're solely charged for the changed blocks.

Data persistence: The Amazon compass point has a higher sturdiness volume (io2), which is meant to generate ninety nine.999 percent sturdiness with a zero.001 percent annual failure rate (AFR), where failure refers to a complete or partial loss of the quantity.

Volume Type:	Description	Use cases
General purpose SSD (gp2)	gp2 (General Purpose SSD) volumes provide cost-effective storage for a wide range of workloads.	virtual desktops, development and test environments, and low-latency interactive apps
Provisioned IOPS SSD (io1)	(io1 and io2) volumes ar meant for I/O-intensive workloads that ar sensitive to storage performance and consistency, like info workloads.	Mission-critical applications, large database such as MongoDB, Microsoft SQL Server, Cassandra, Oracle, MySQL, and PostgreSQL
Throughput optimized HDD (st1)	Low-cost HDD capacity for frequently accessed, high-output tasks.	Streaming, big data, data warehouses, log processing
Cold HDD (sc1)	For less-frequently accessible tasks, the cheapest HDD volume is used.	large volumes of data that is infrequently accessed

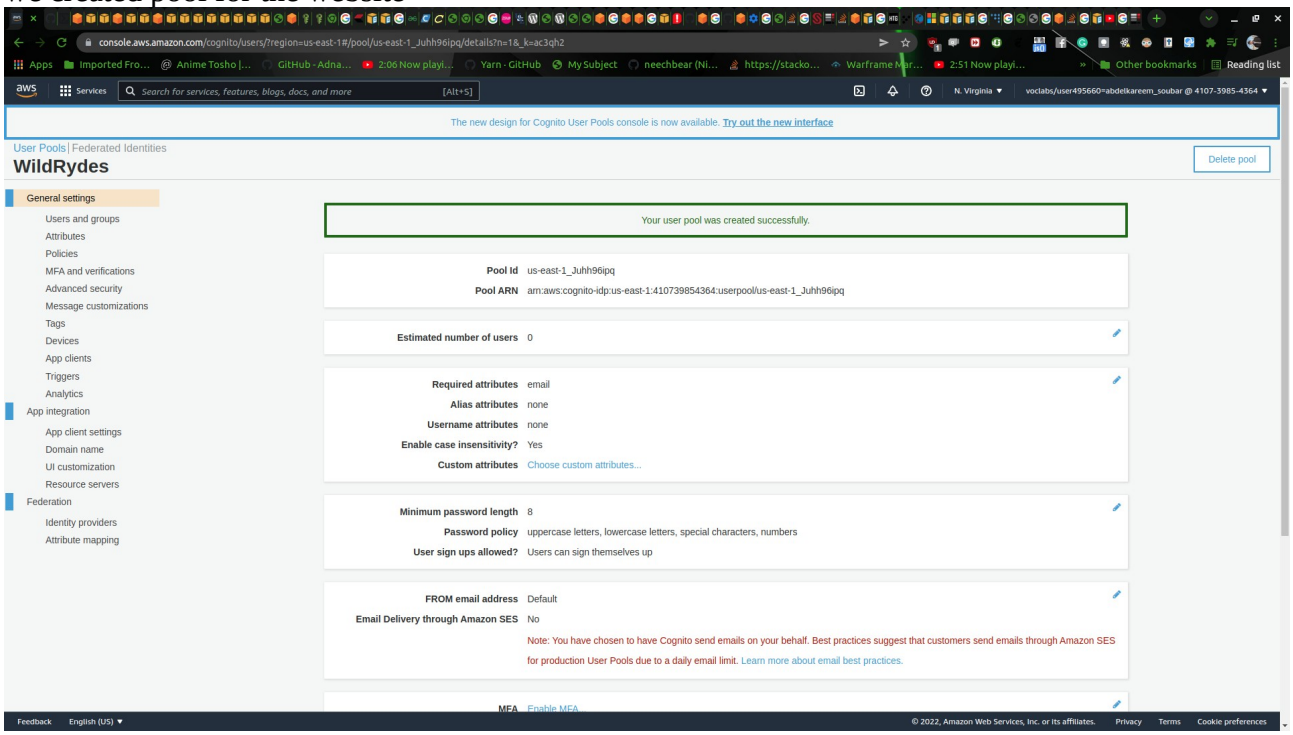
Task 4

GitHub repo works and its been verified

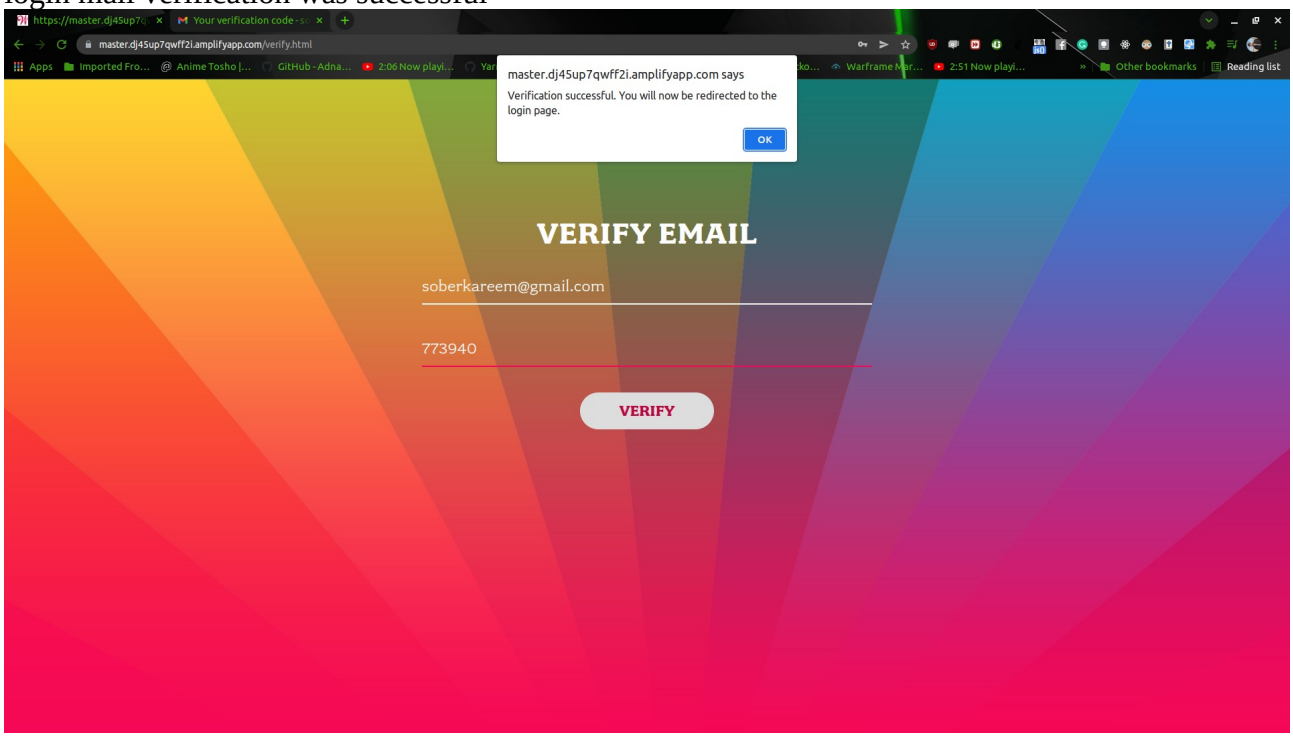


the tab title had changed

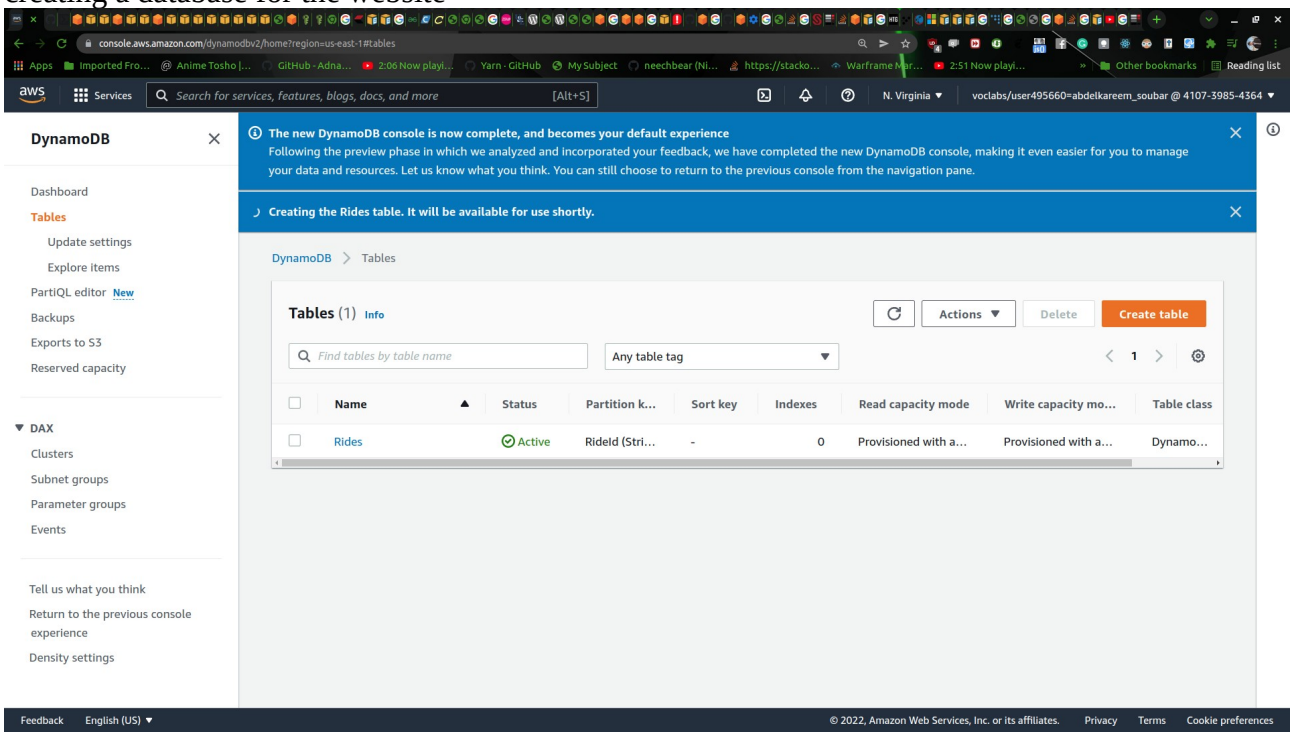
we created pool for the website



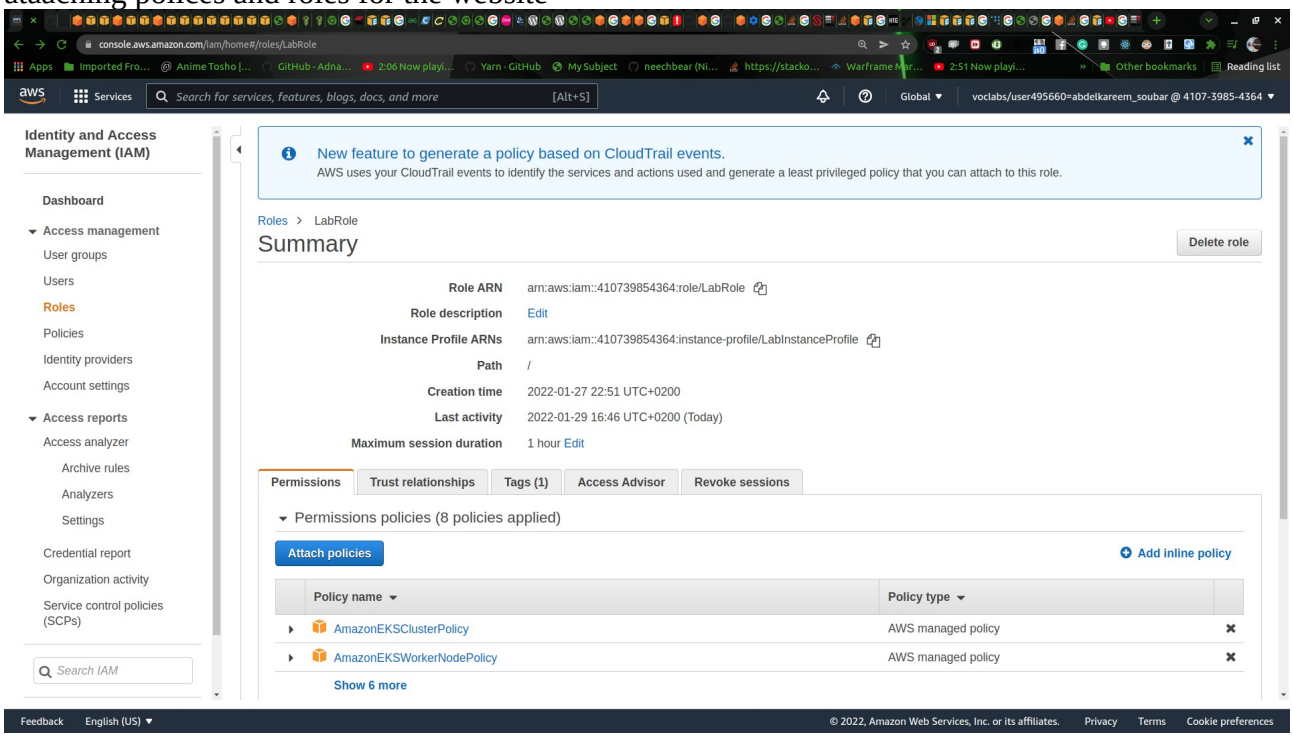
login mail verification was successful



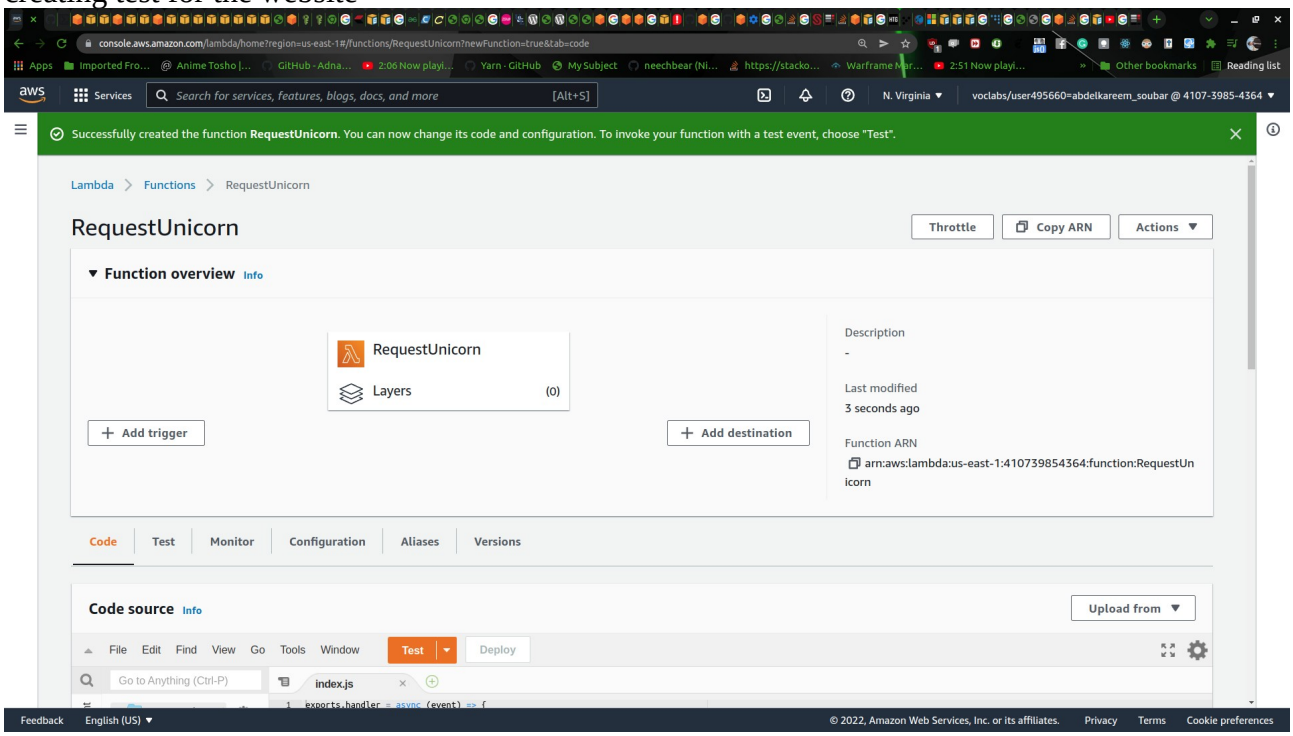
creating a database for the website



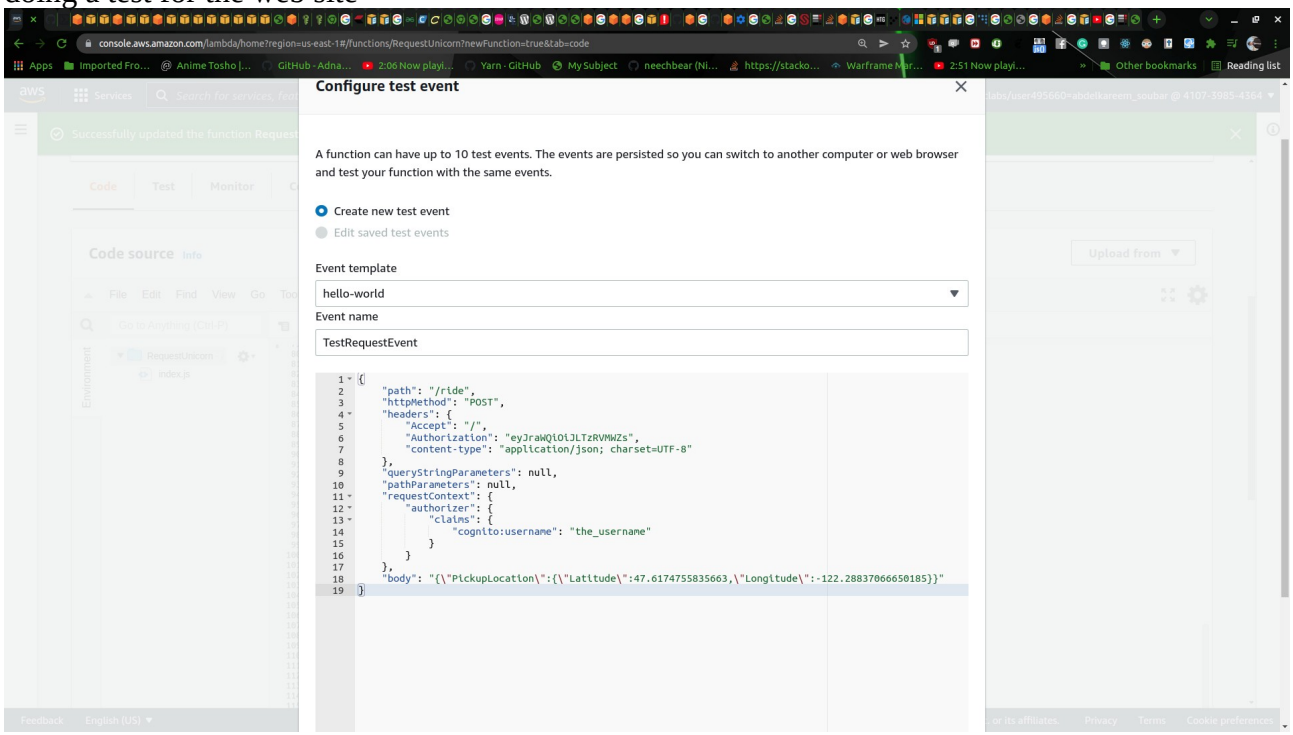
ataaching polices and roles for the website



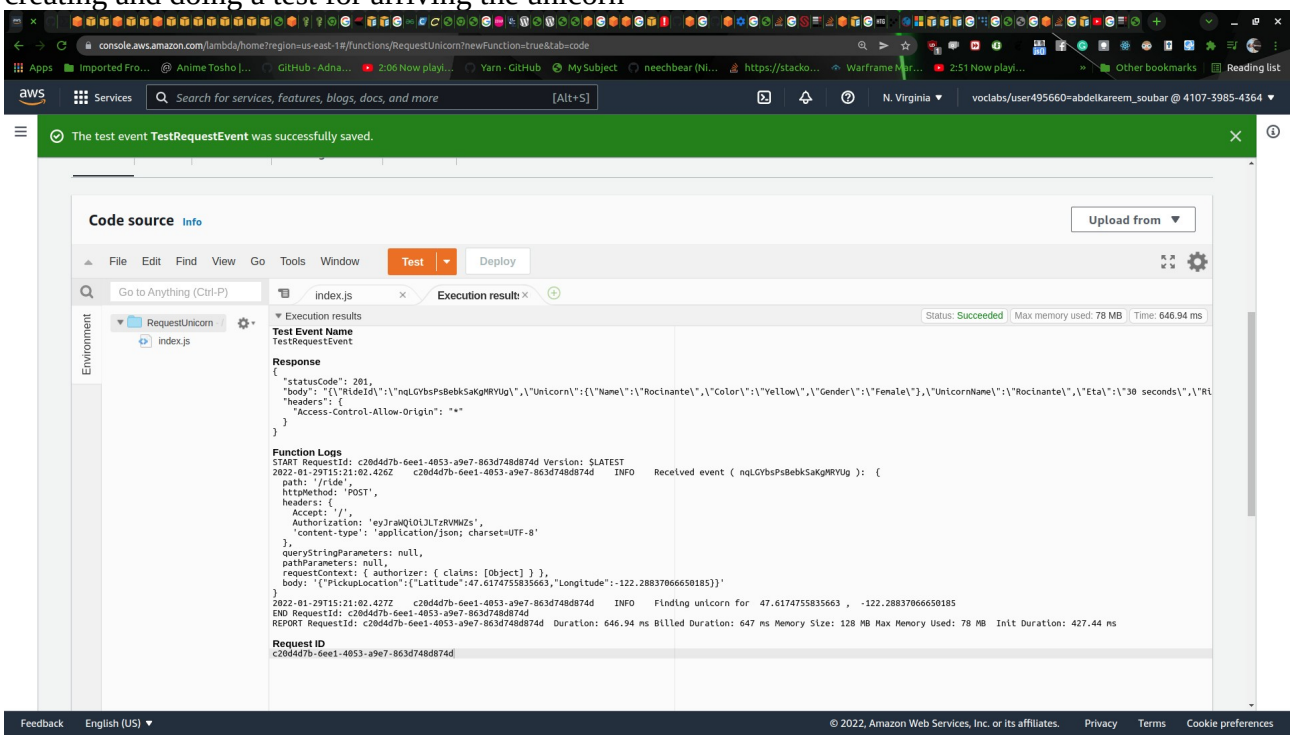
creating test for the website



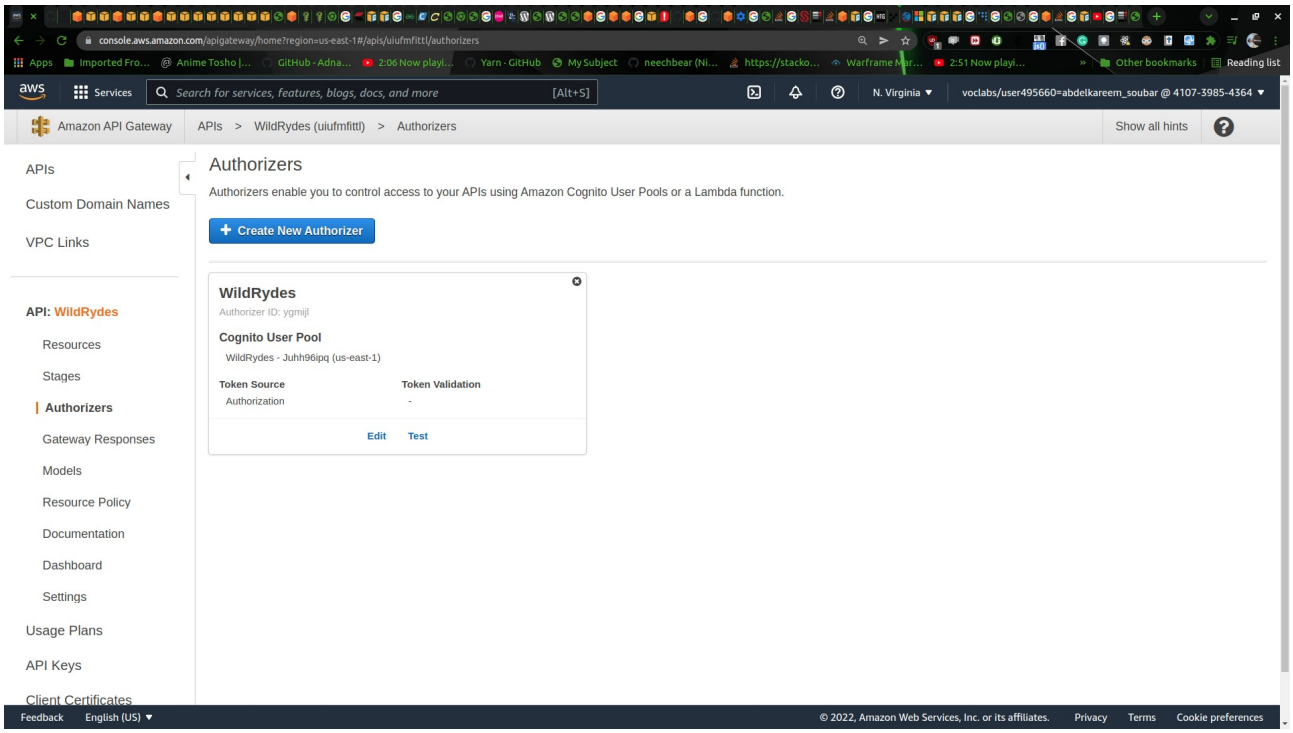
doing a test for the web site



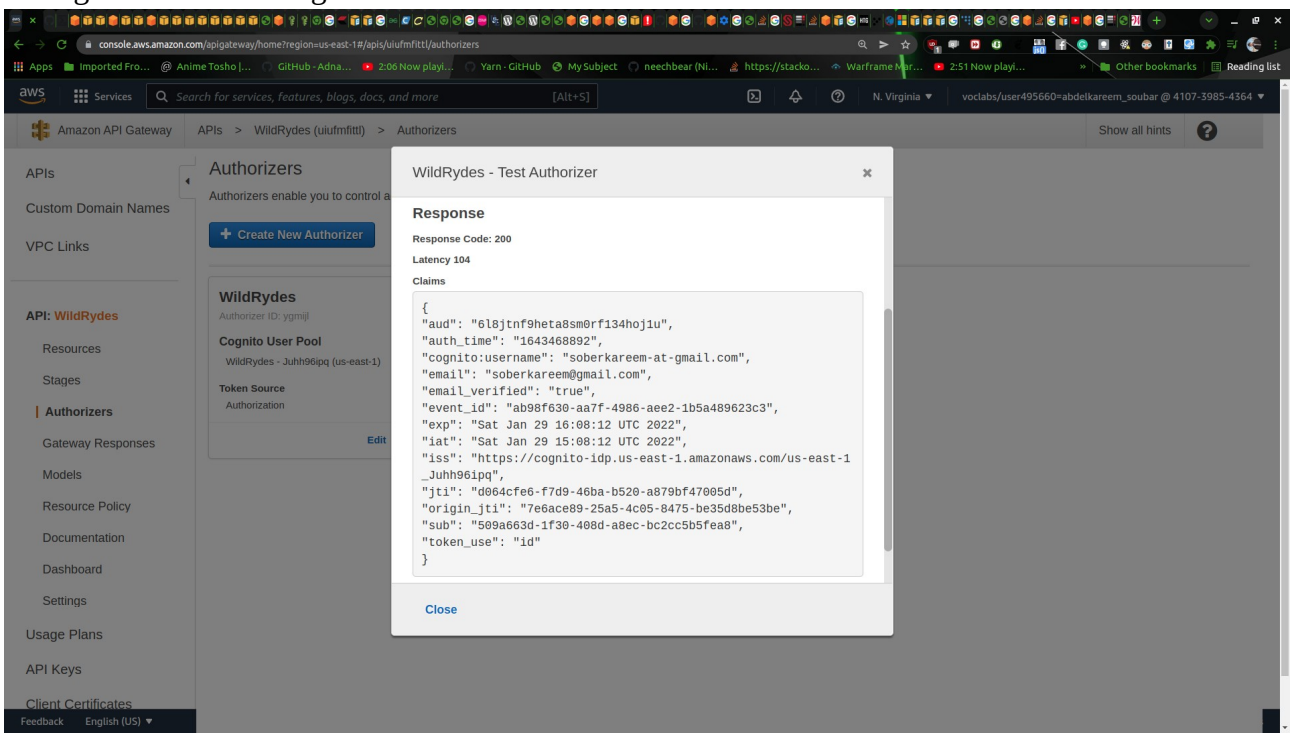
creating and doing a test for arriving the unicorn



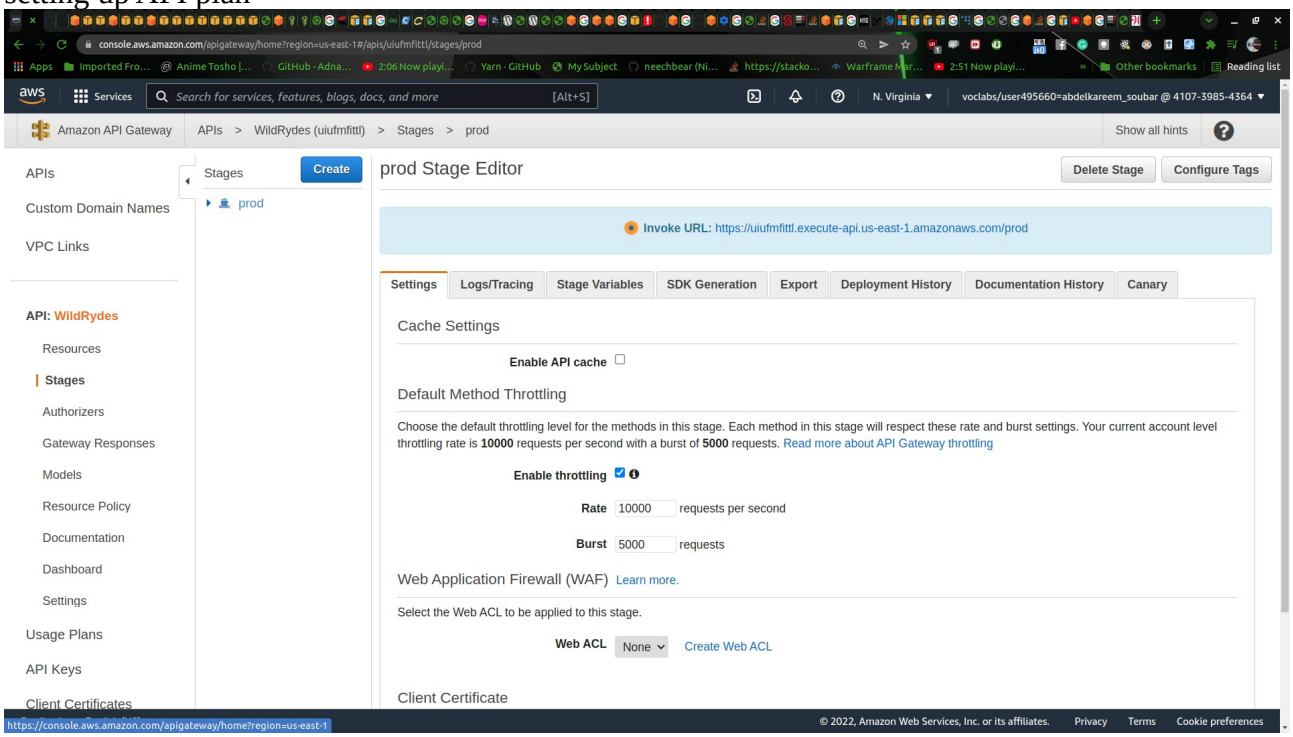
authorizing the web site to use the API



testing the control through API in relation to the website



setting up API plan



booked my ride :D