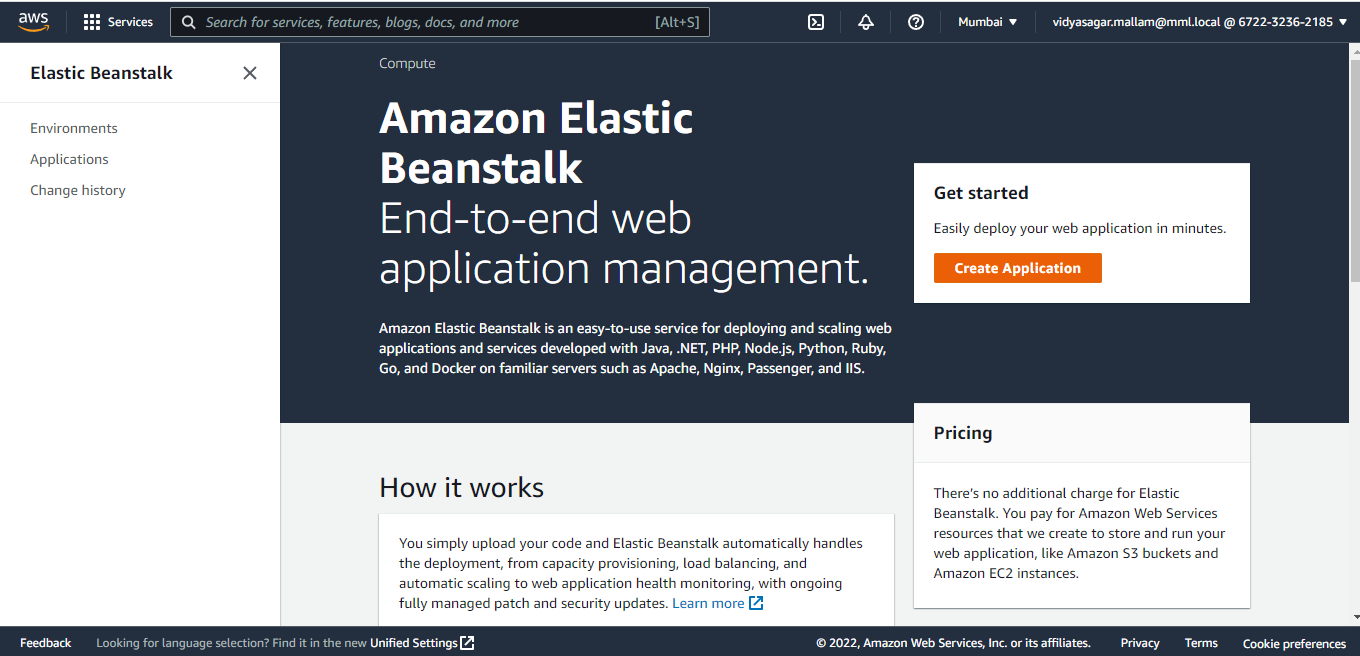
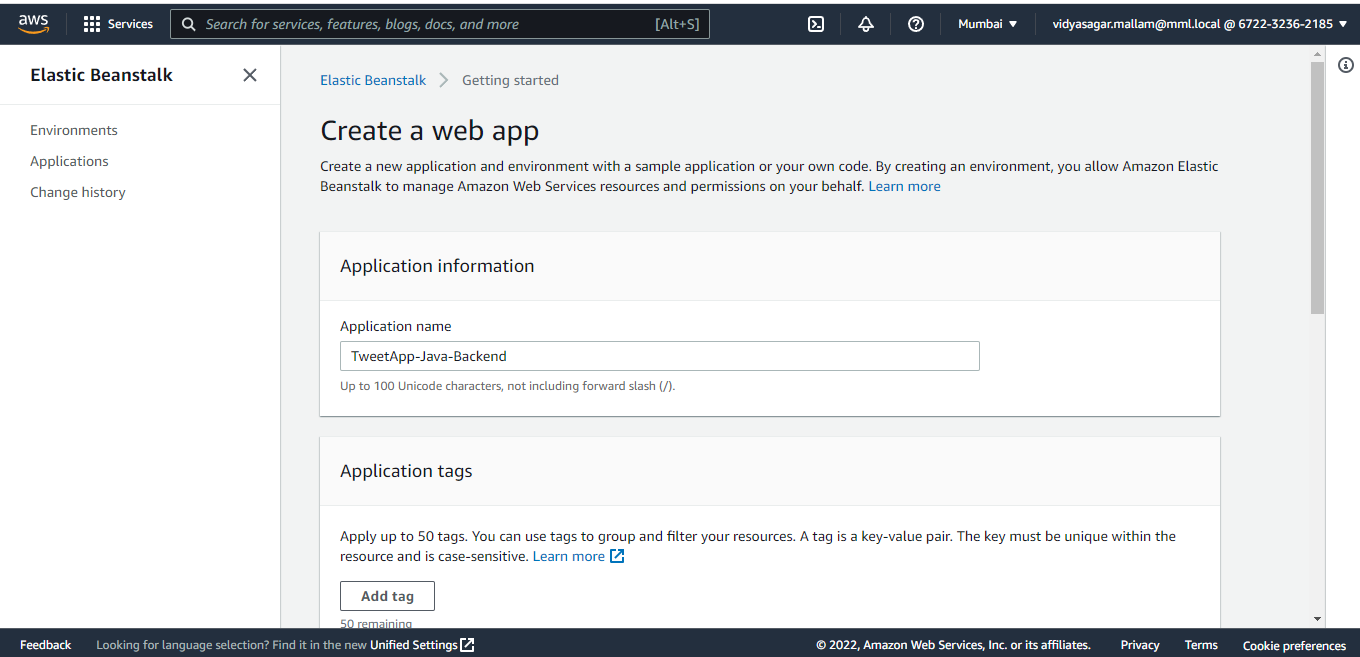
AWS Deployment

**Backend Deployment**

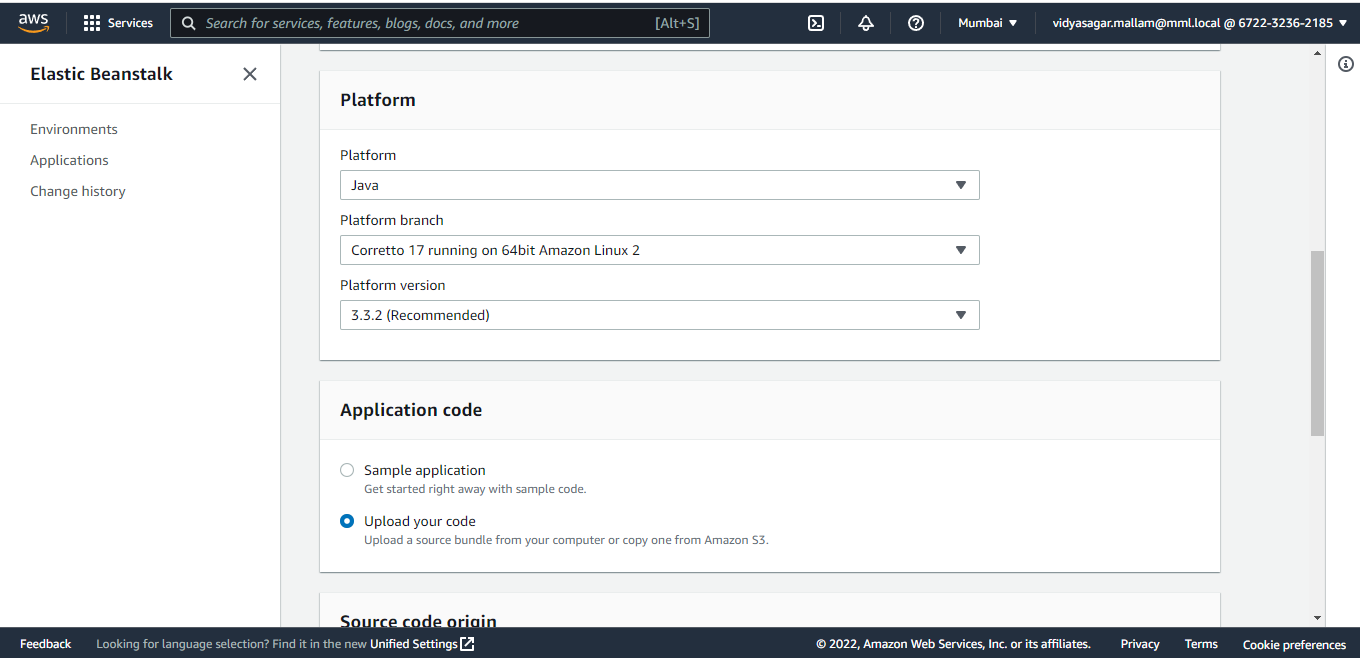
**Step-1:** Open EBS (Elastic Beanstalk) and Click on Create Application



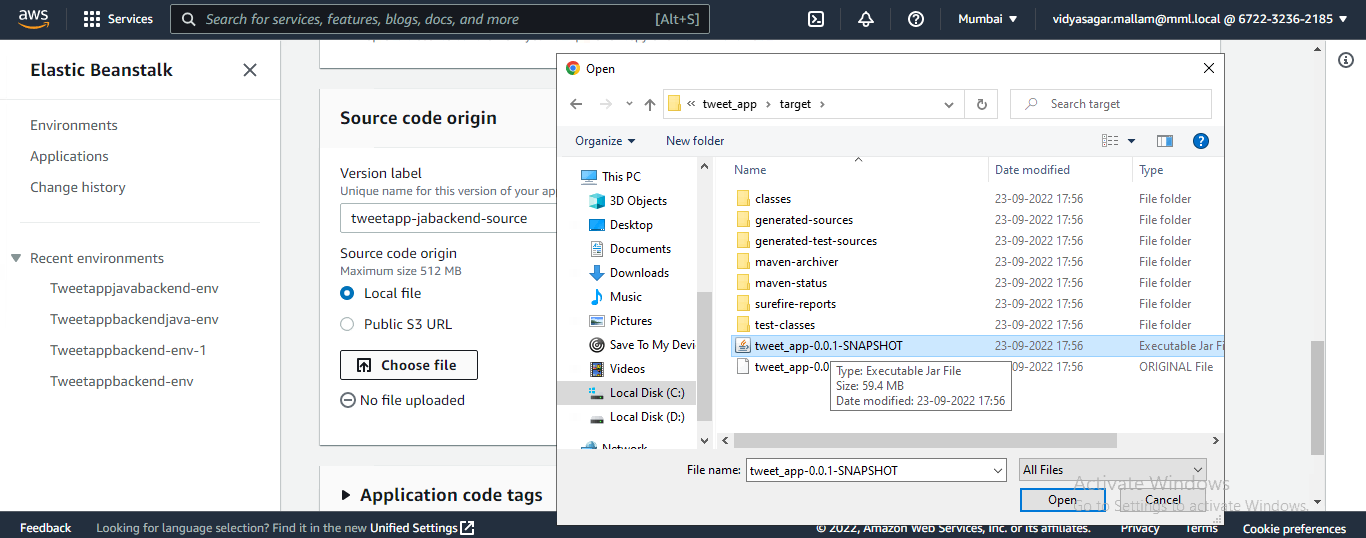
**Step-2:** Fill the mentioned Details (Application name) and tags are optional

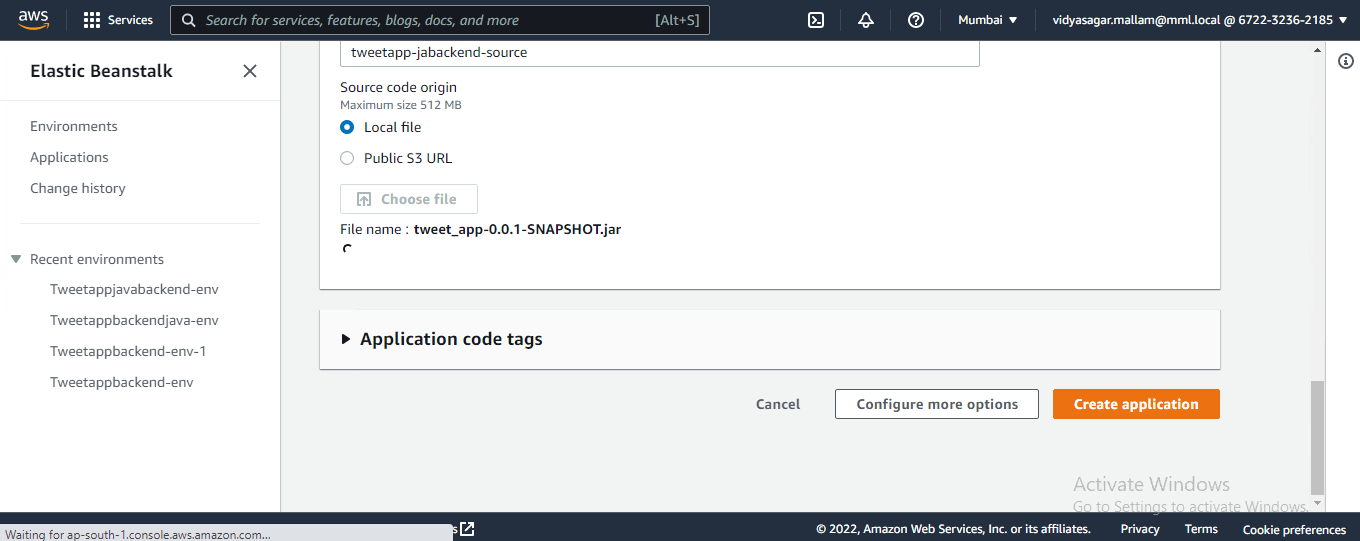


**Step-3:** Select Preferred platform(java), platform branch and version

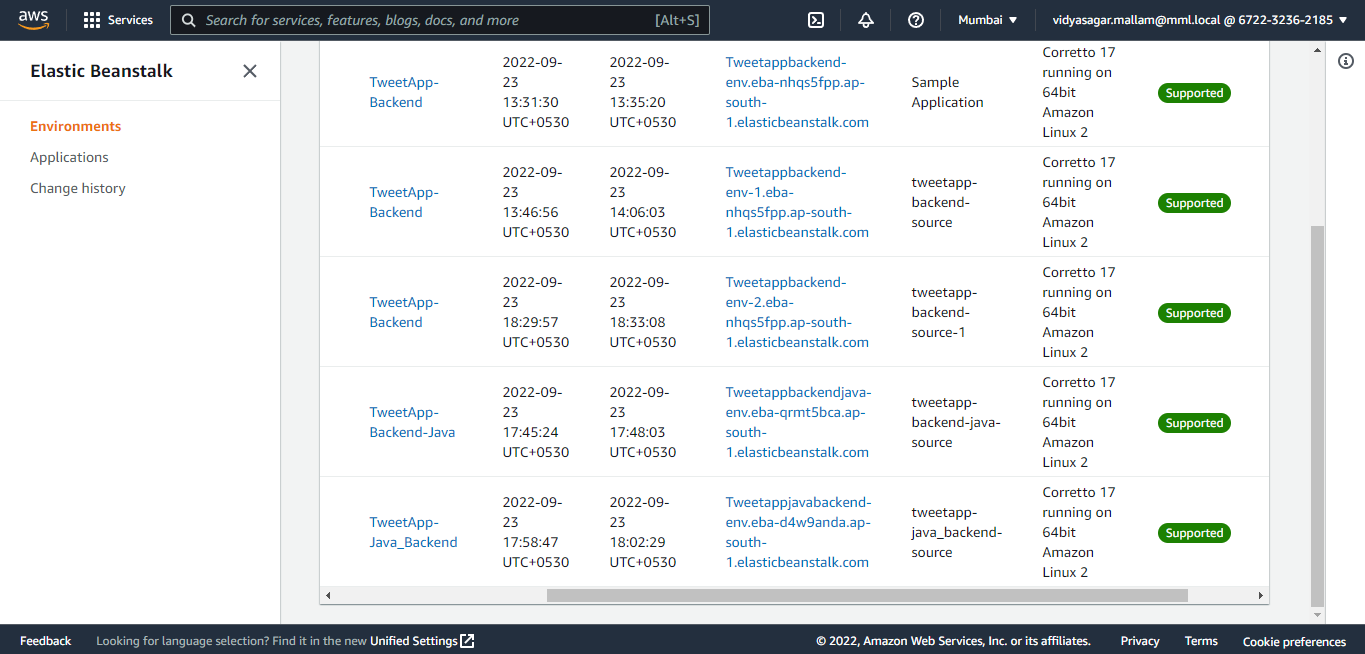


**Step-4:** Upload jar file and click on create application

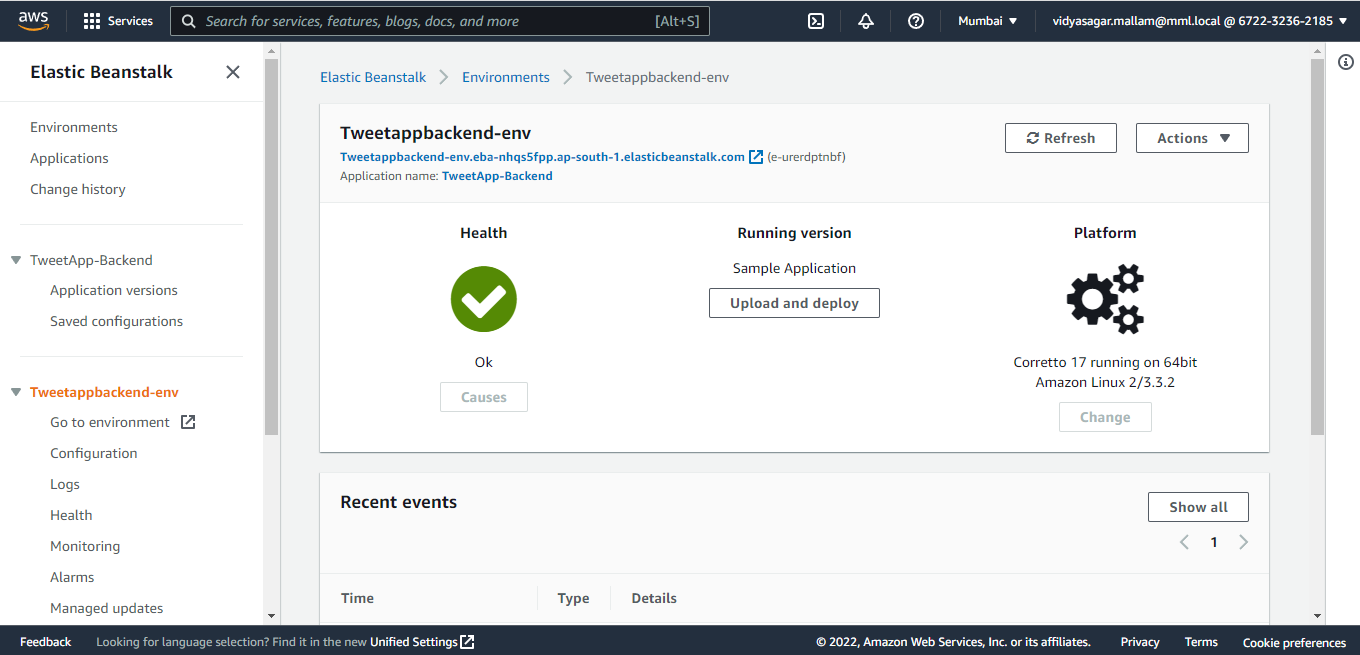


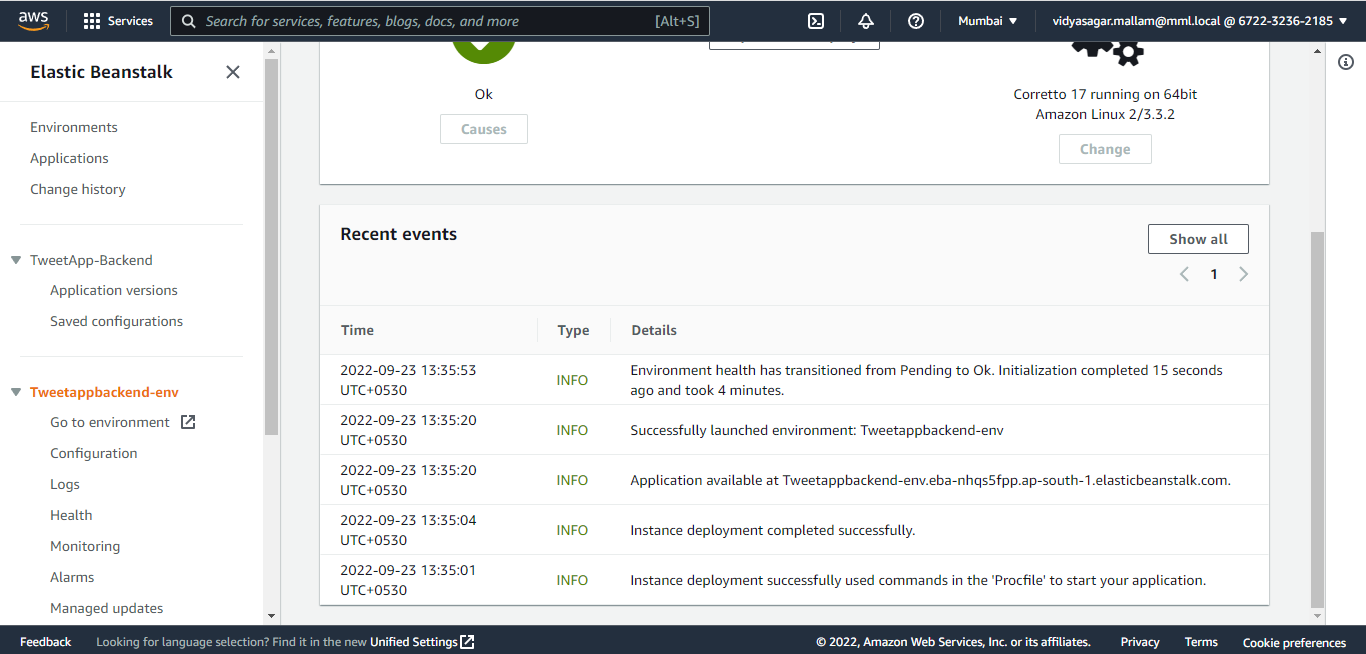


**Step-5:** We can see the list of applications we have created

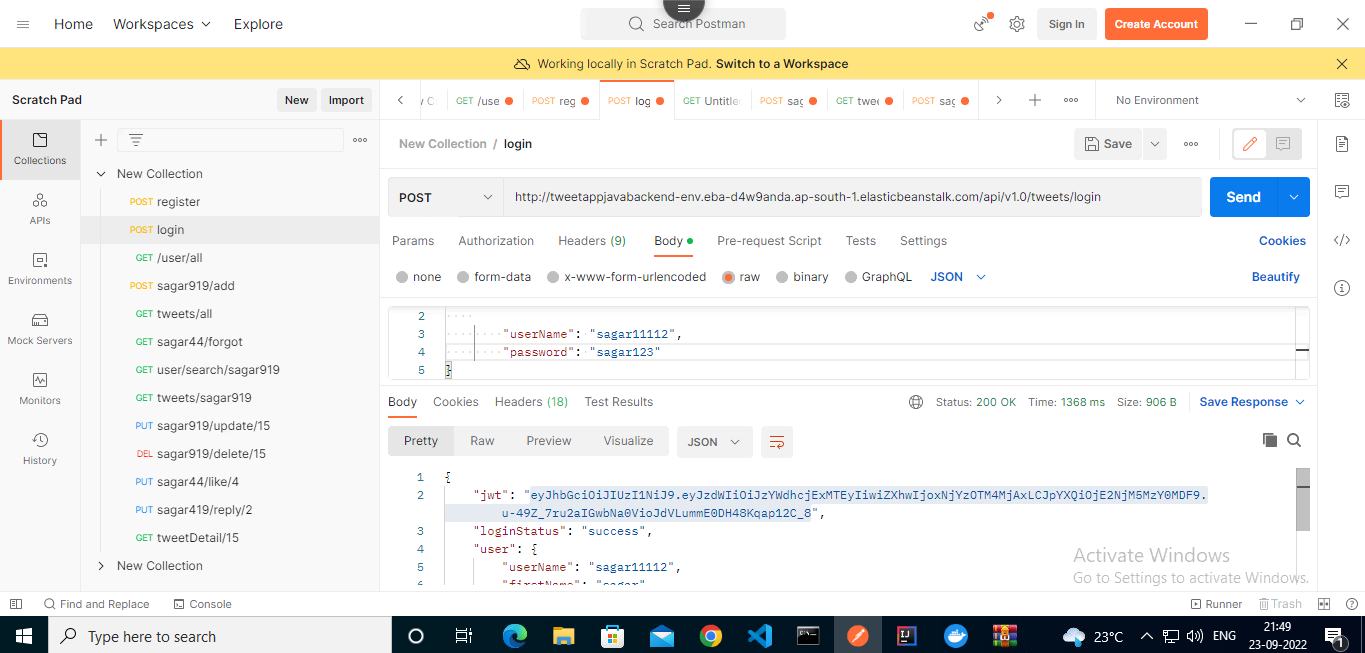


**Step-6:** deployment process will happen once jar is uploaded. Once deployment is done, the link will be available for use. And if any changes required in backend, we could upload new jar file and deploy again in the same page.

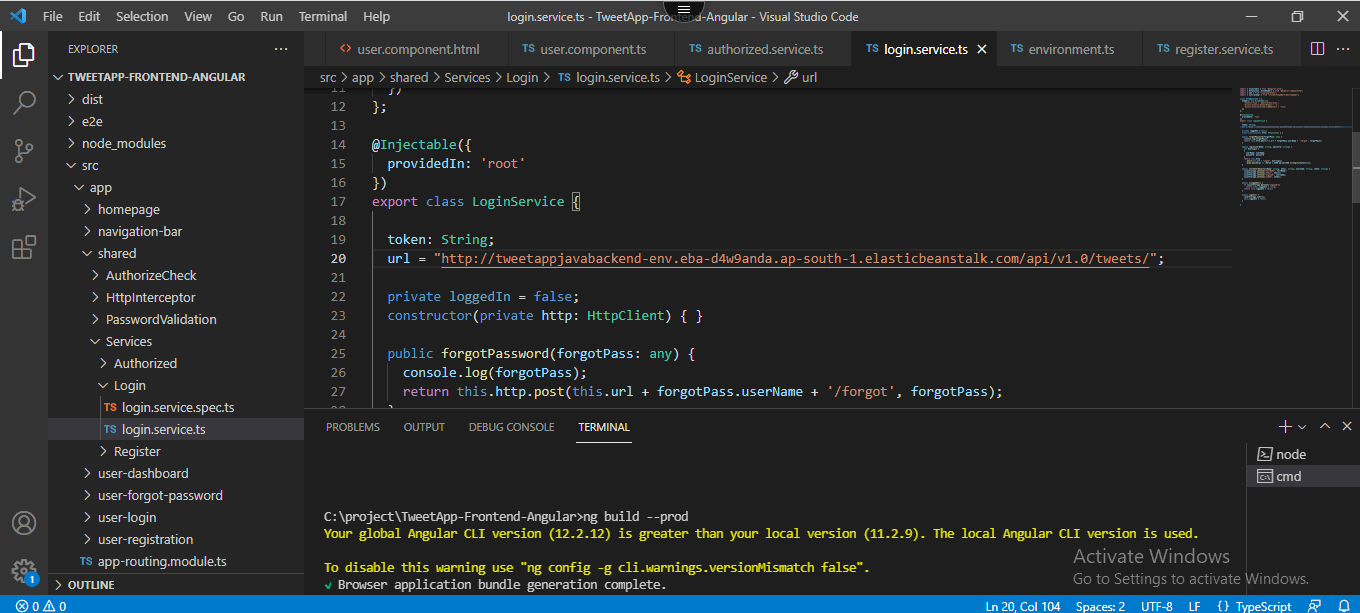




**Step-7:** Same link we can paste in place of localhost and run-in postman, and we can see the response.

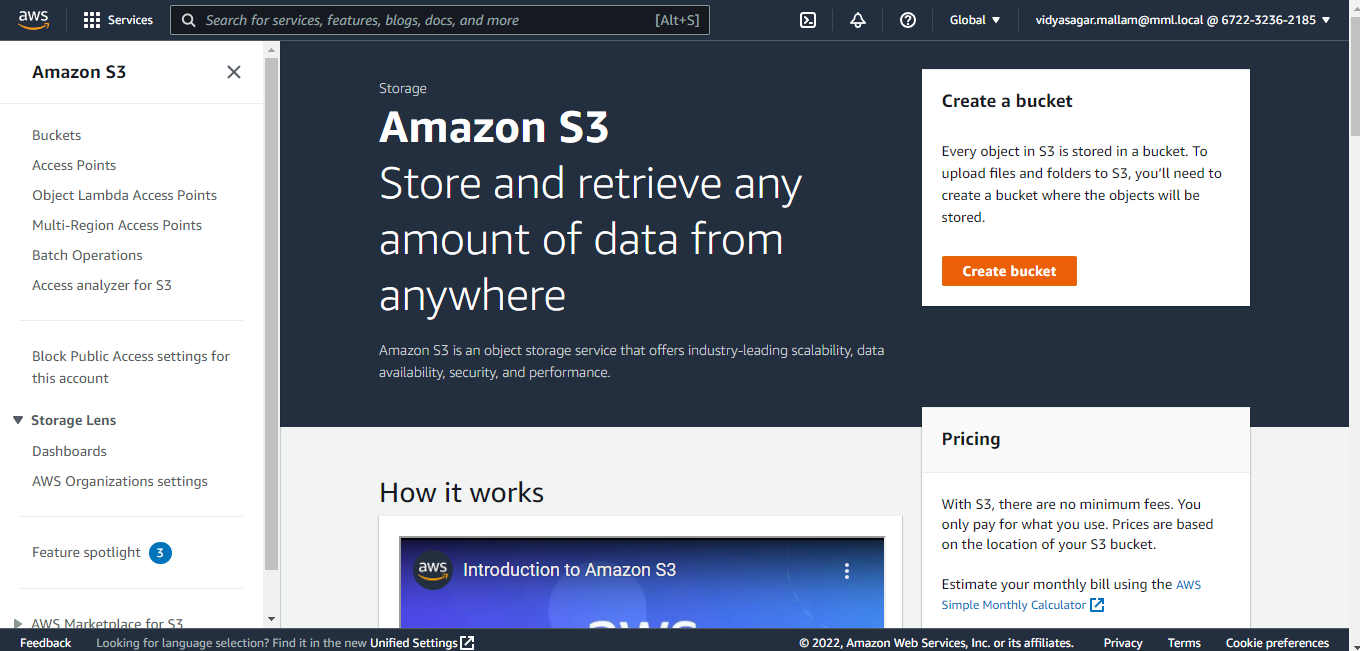


**Step-8:** The same link will replace localhost in url in frontend.

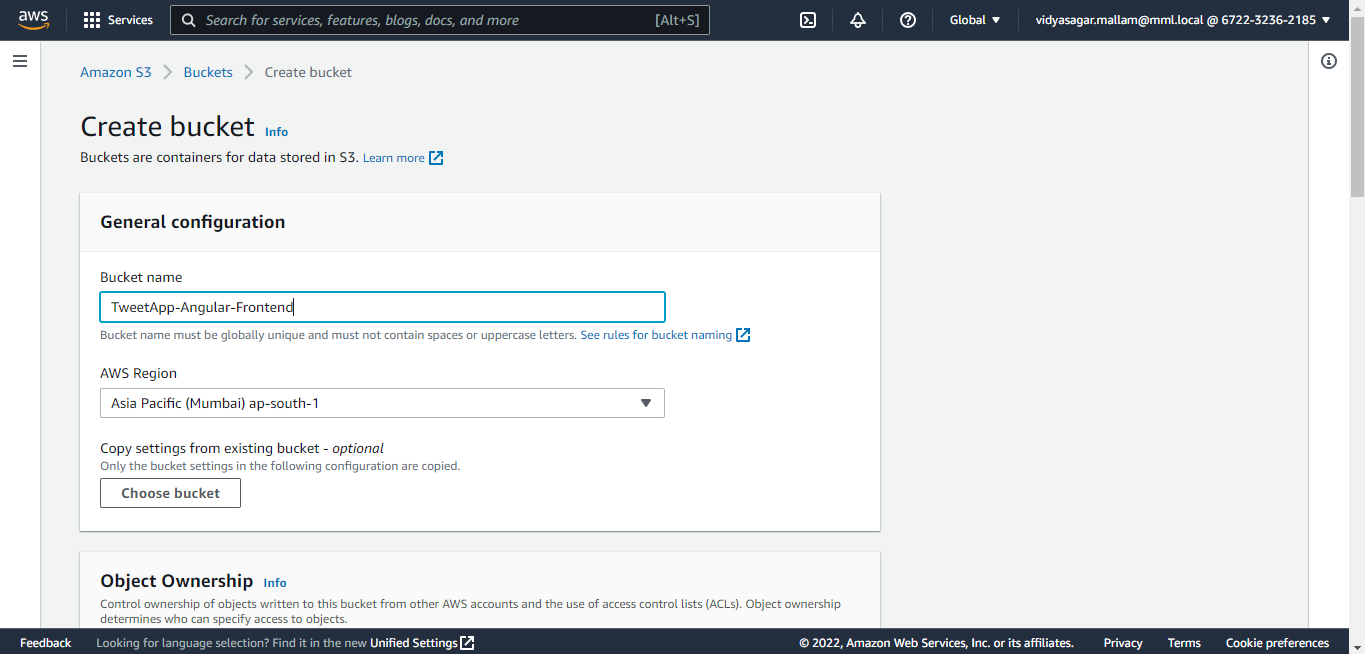


**Front-end Deployment**

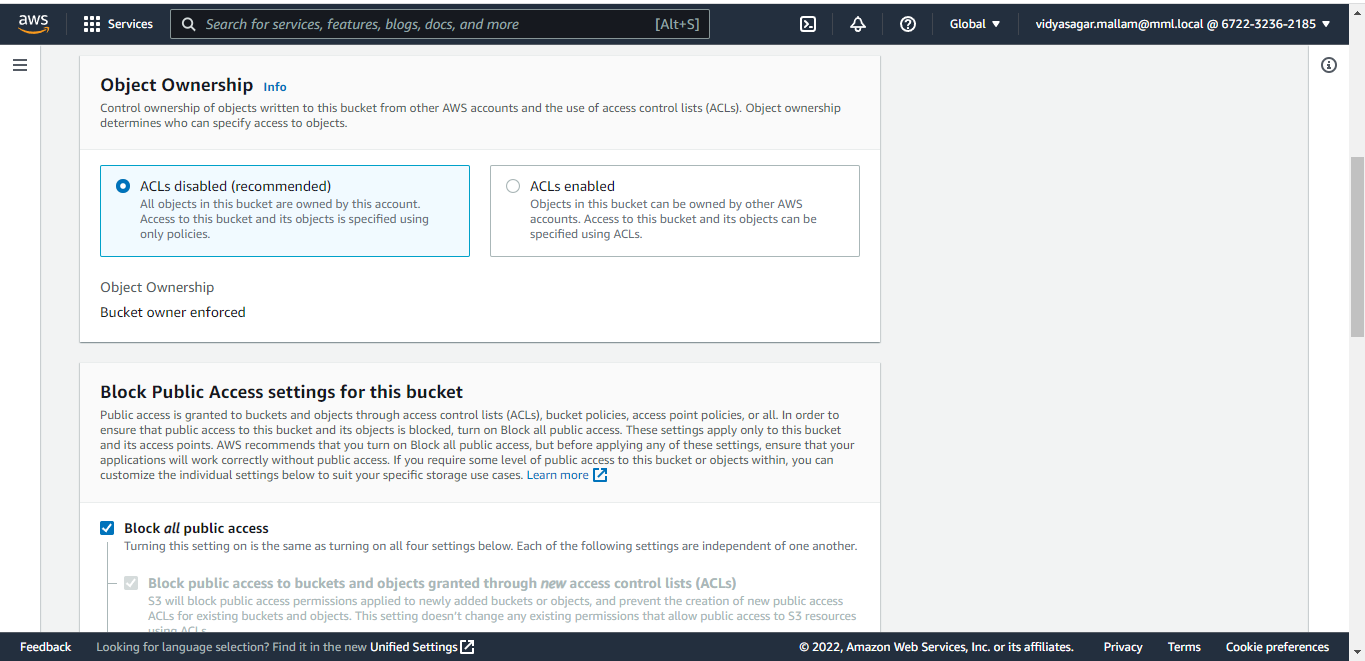
**Step-1:** Open S3 and click on create bucket



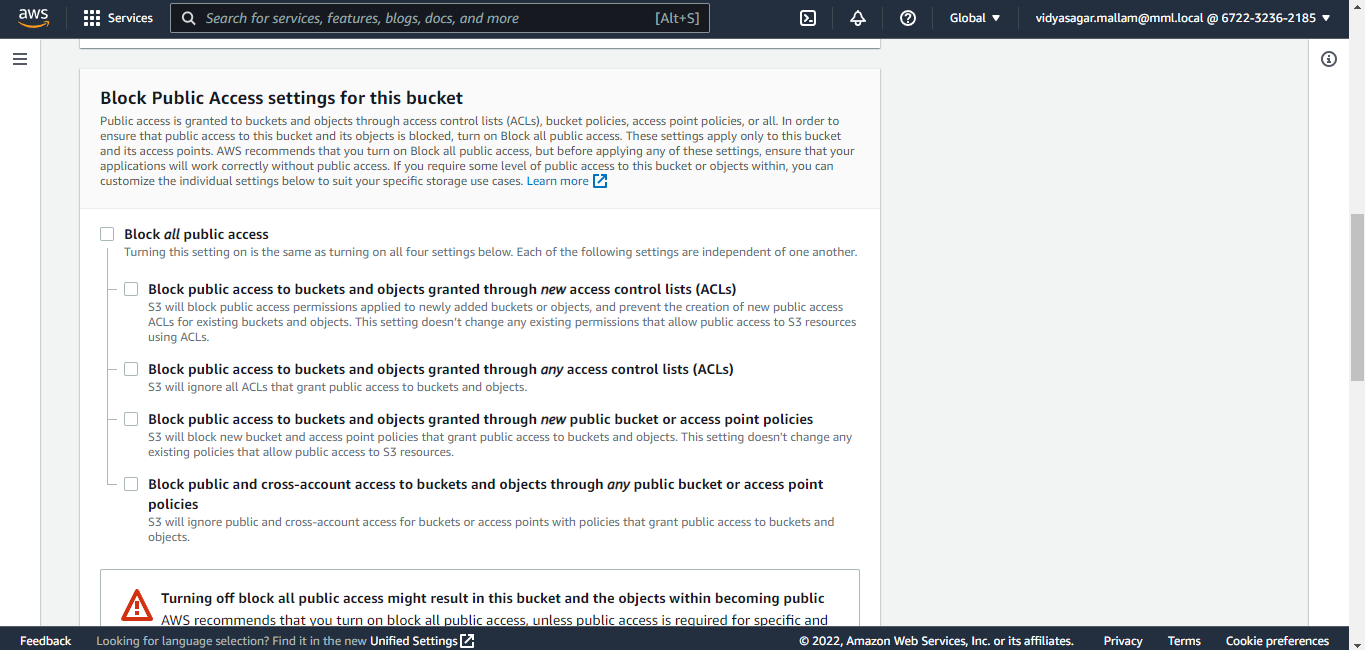
**Step-2:** Fill the details (Bucket name) and region



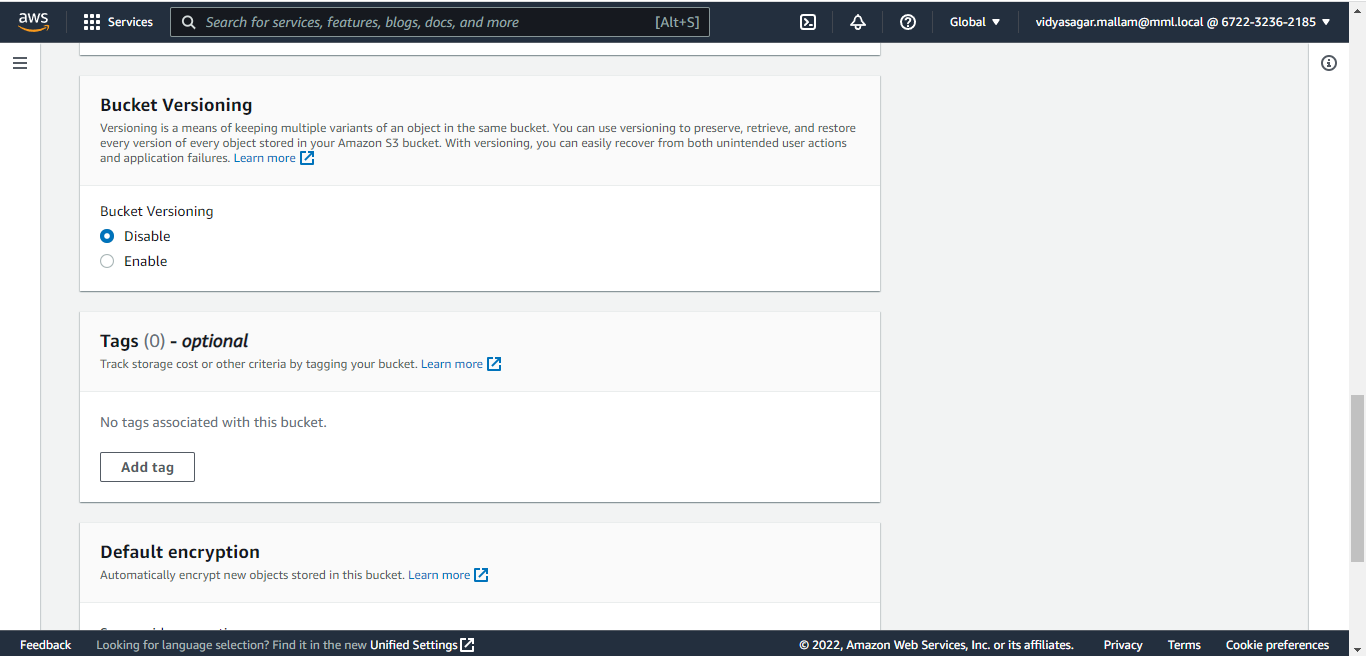
**Step-3:** Select ACLs disabled



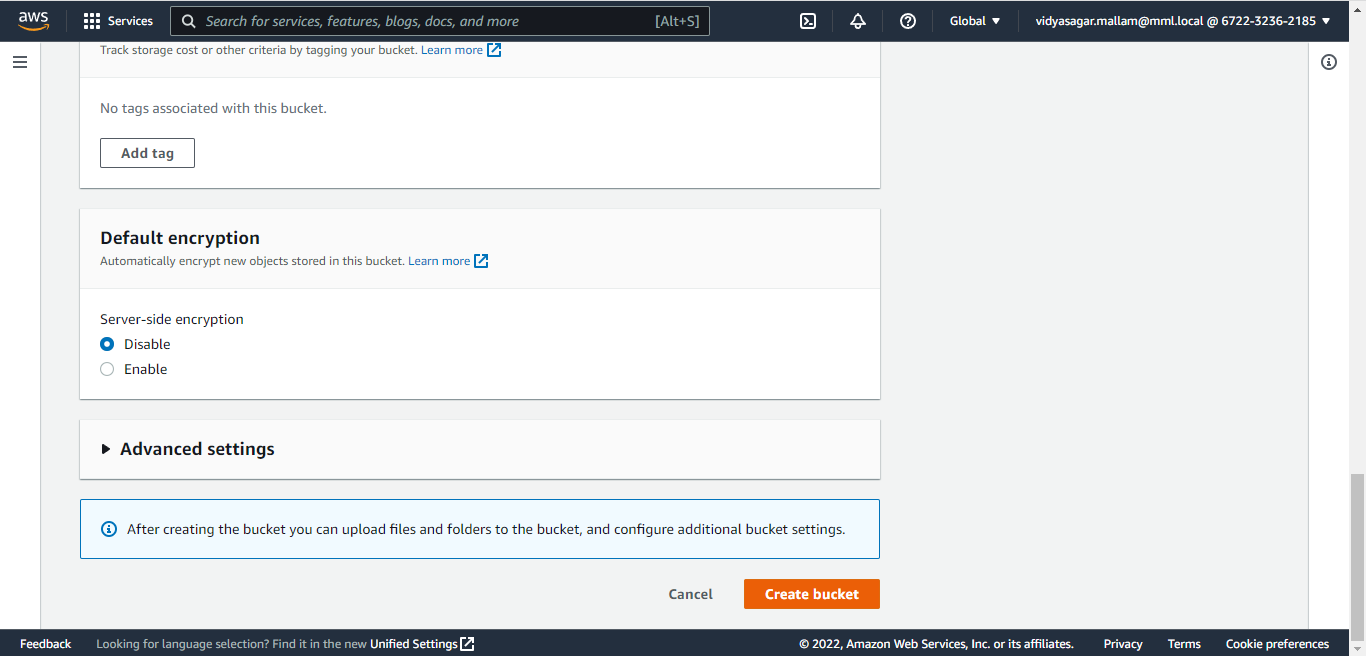
**Step-4:** Uncheck the block all public access



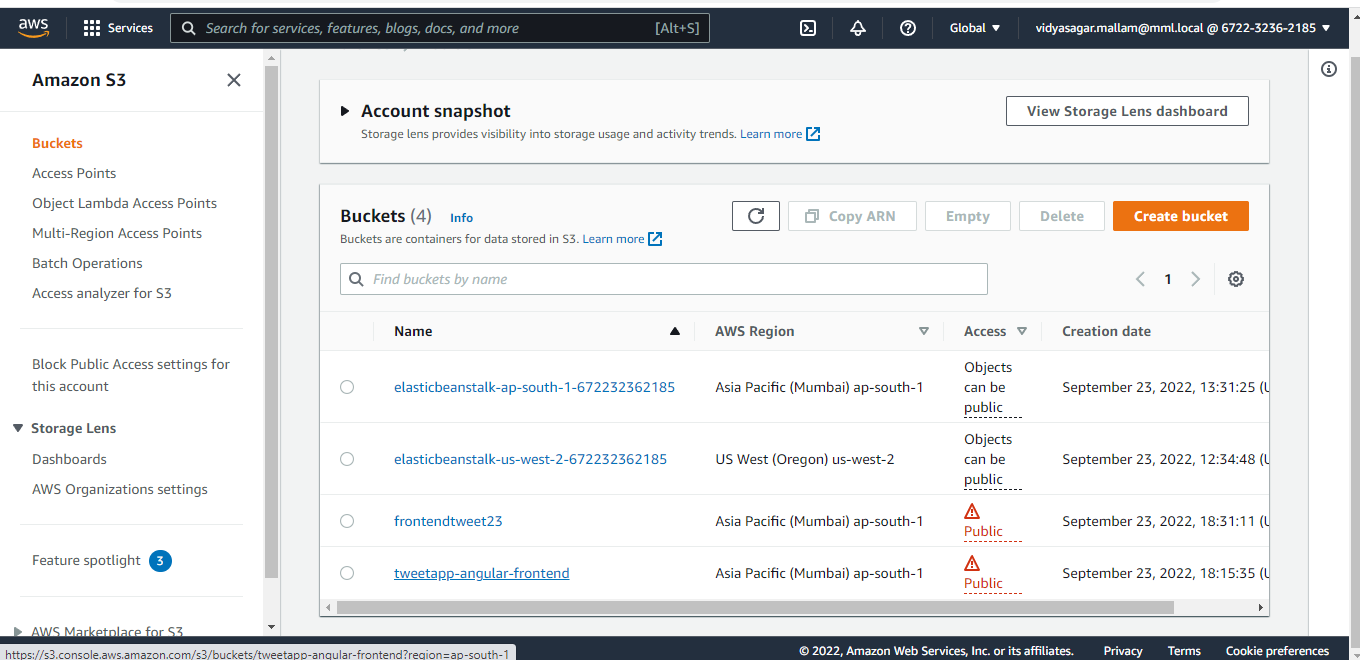
**Step-5:** Disable Bucket Versioning



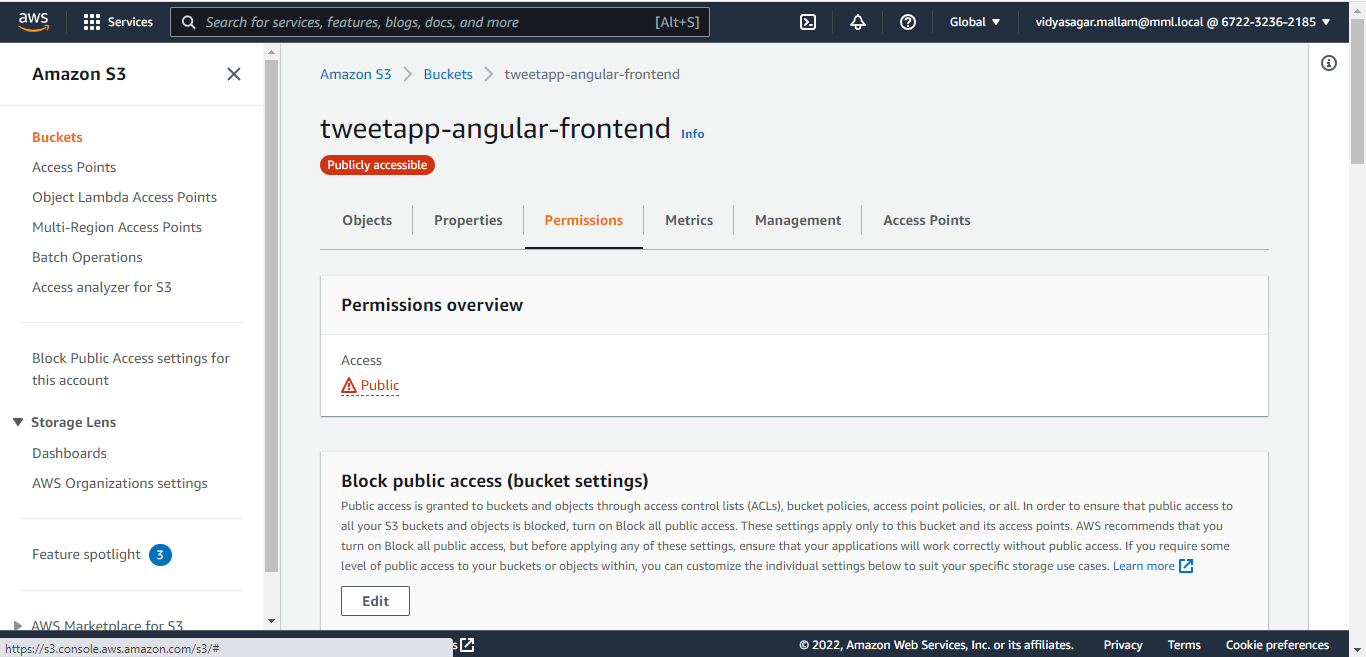
**Step-6:** Disable Default encryption and click on create application



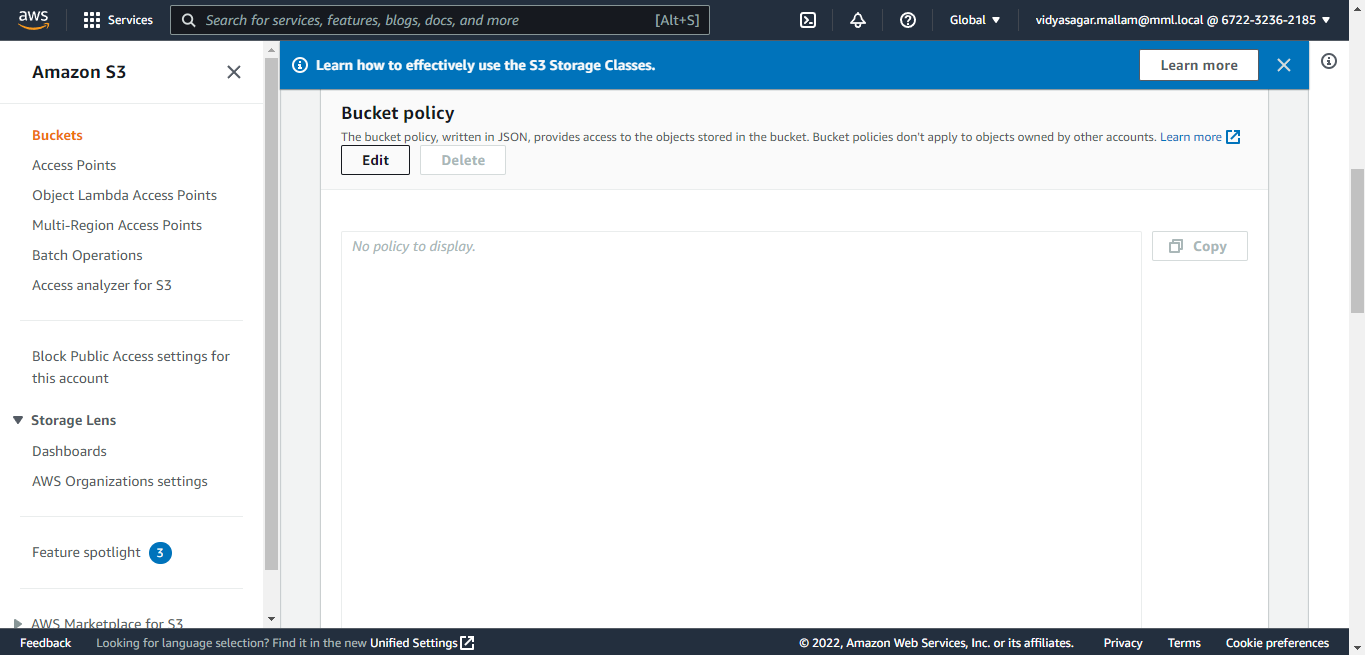
**Step-7:** We can see the list of buckets created



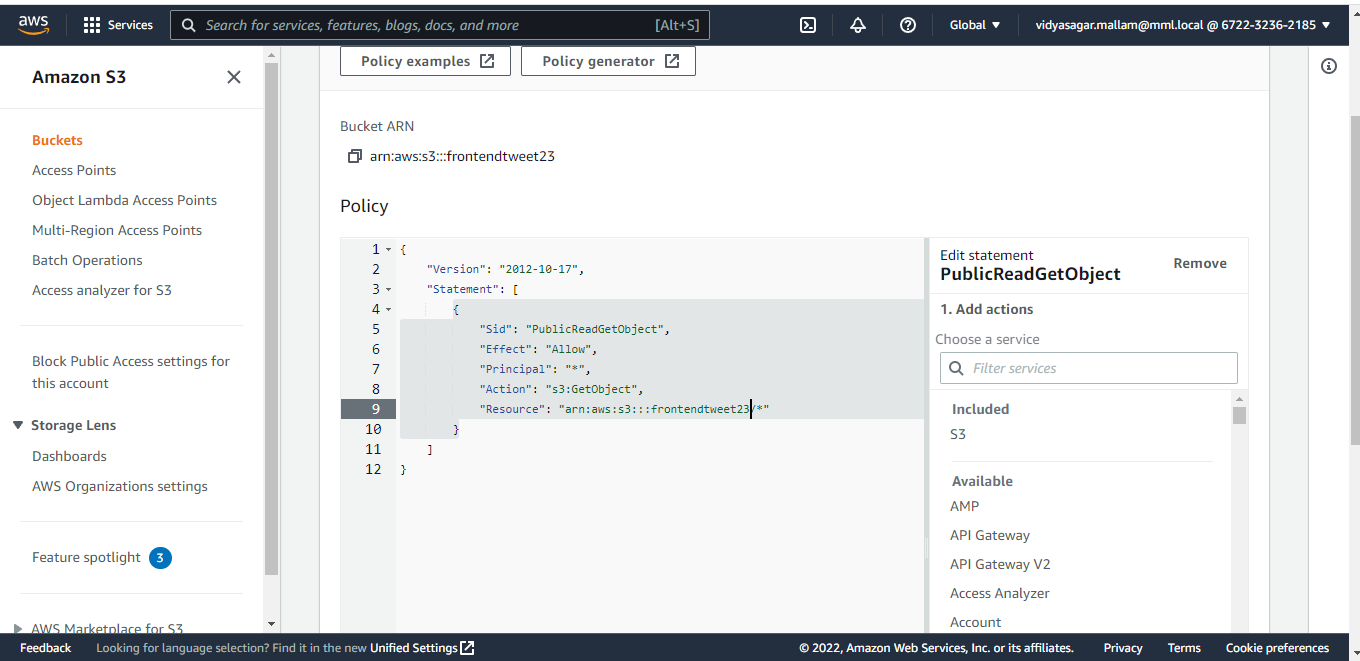
**Step-8:** Click on the bucket and go to permissions tab

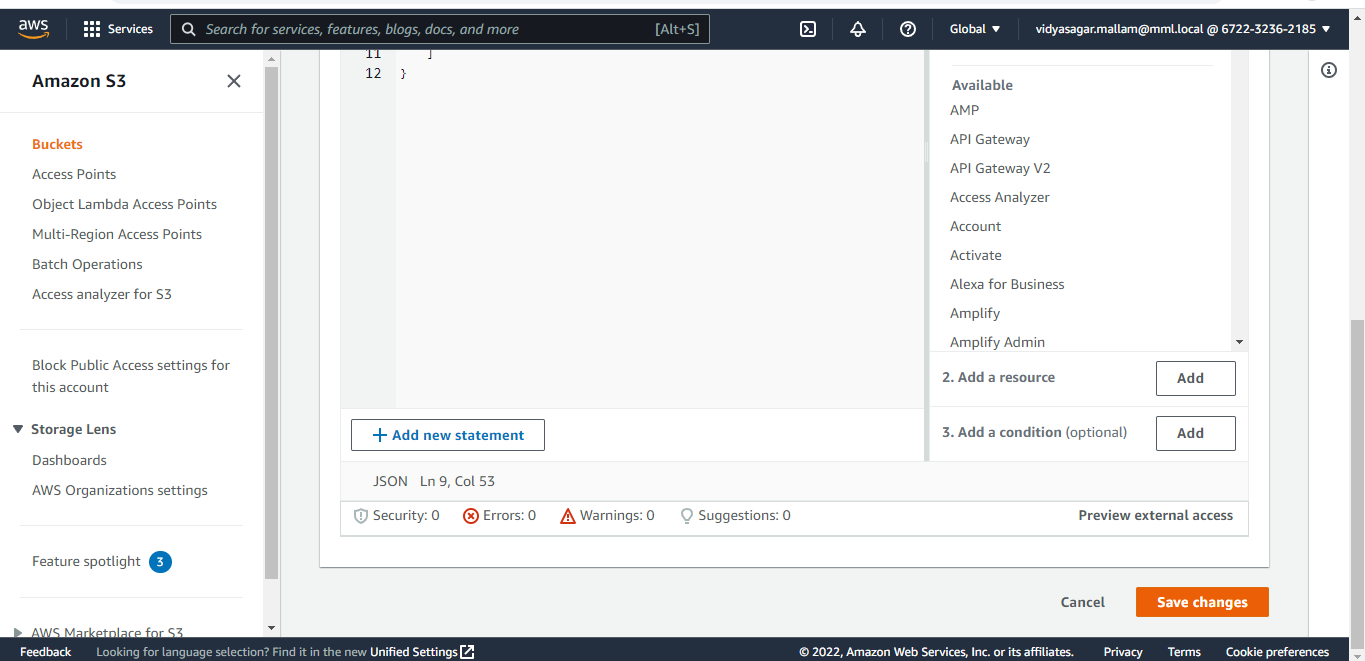


**Step-9:** Scroll down and click on edit Bucket policy

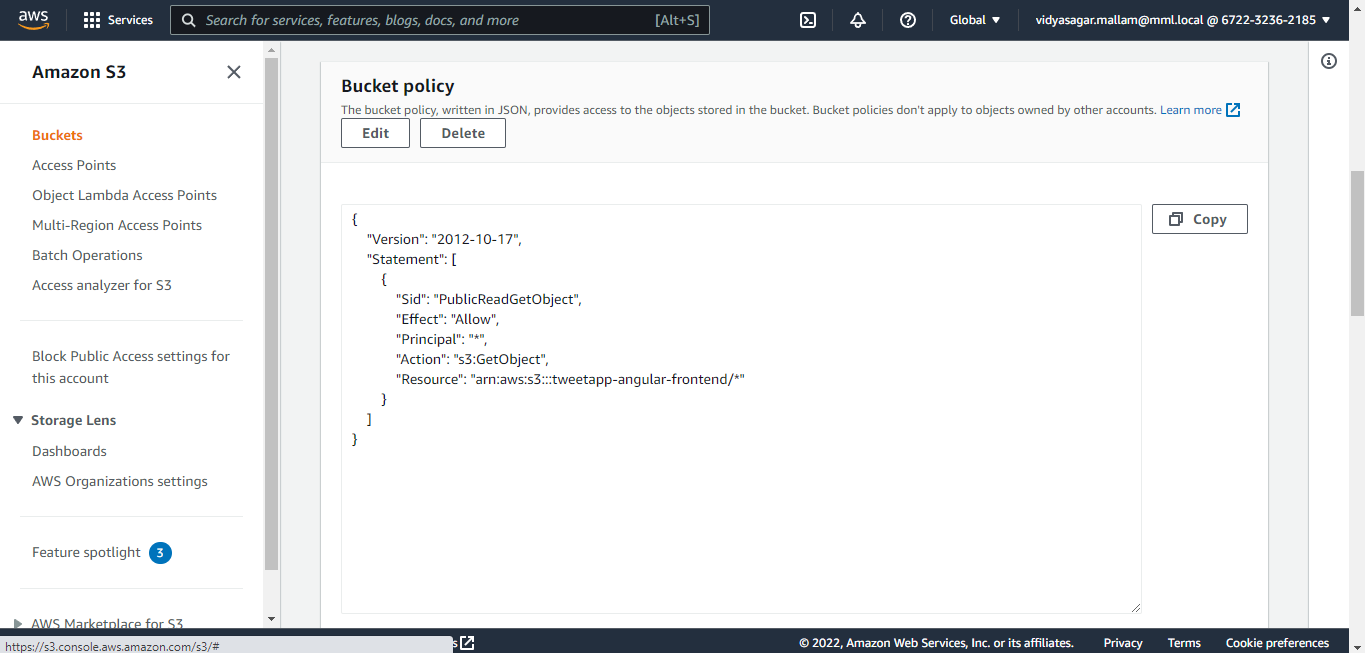


**Step-10:** paste the ARN link (provided above) in resource and add /\* at the end and save the changes.

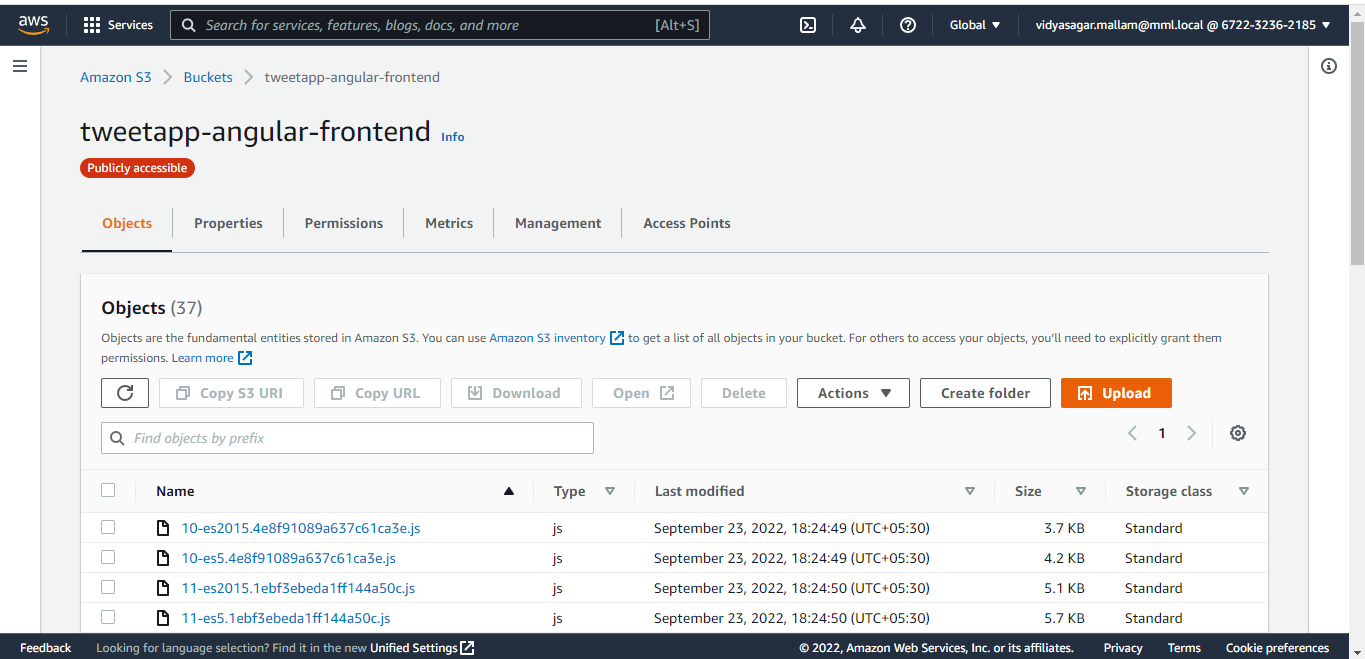




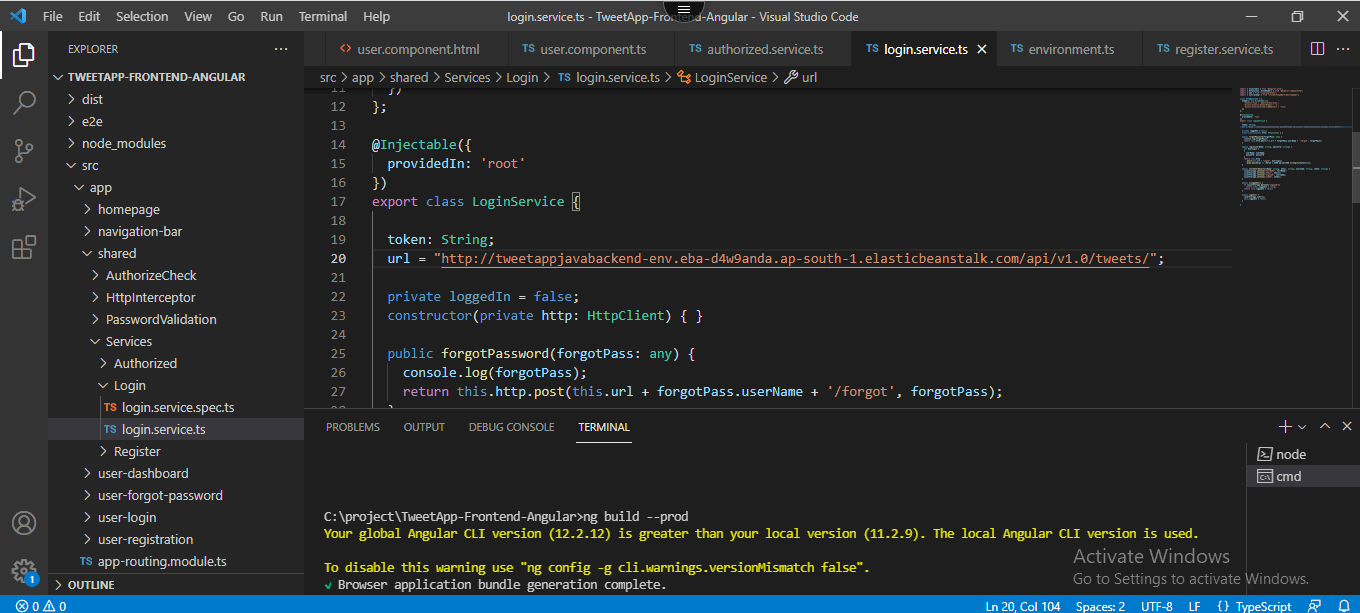
We can see the edited policy here



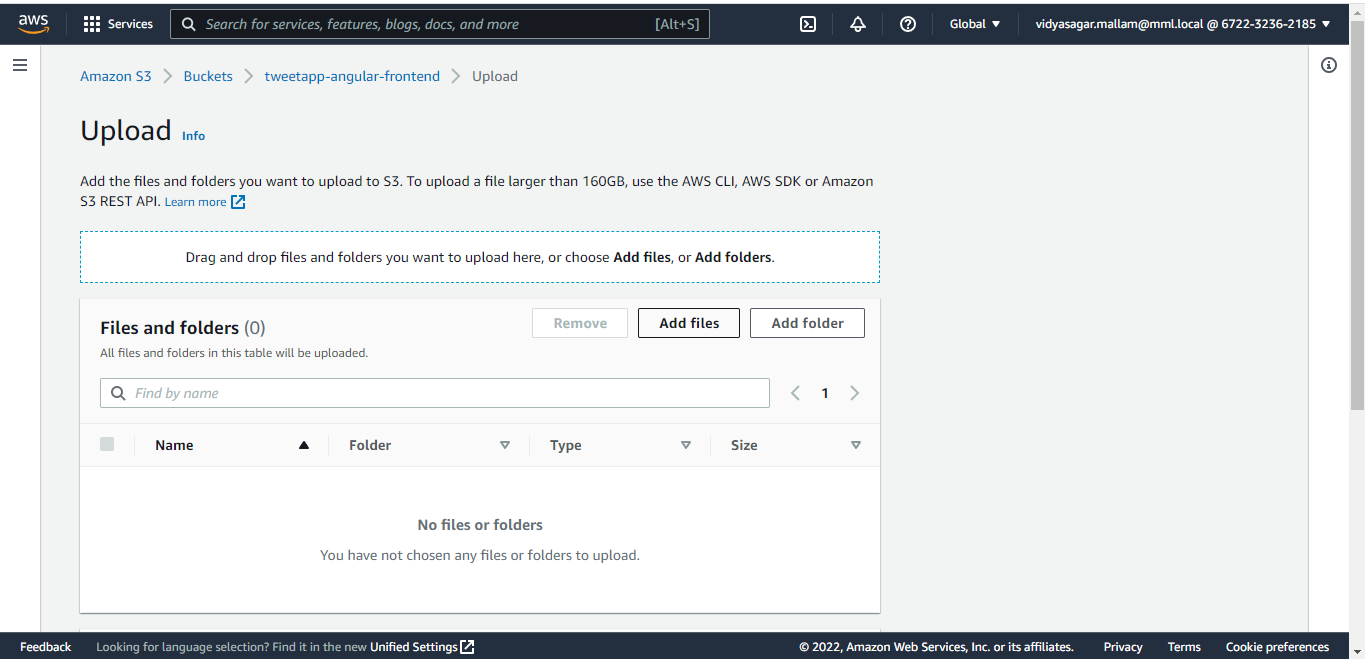
**Step-11:** Open objects tab

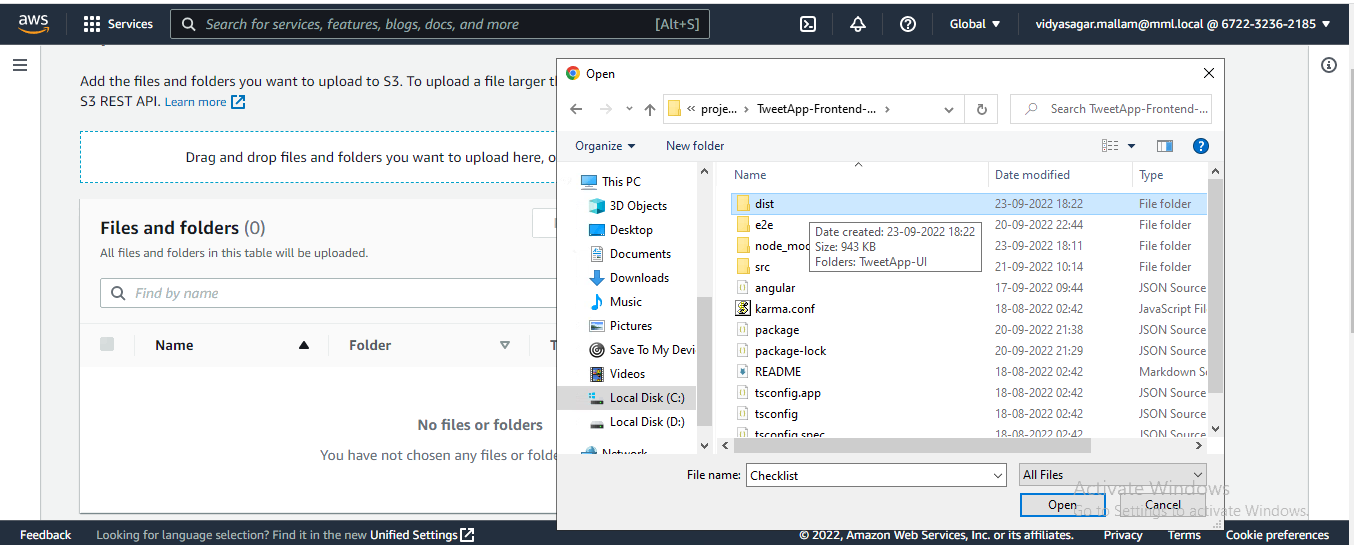


**Step-12:** Now go to Angular code and run command “ng build –prod”. This will create a dist folder in application folder.

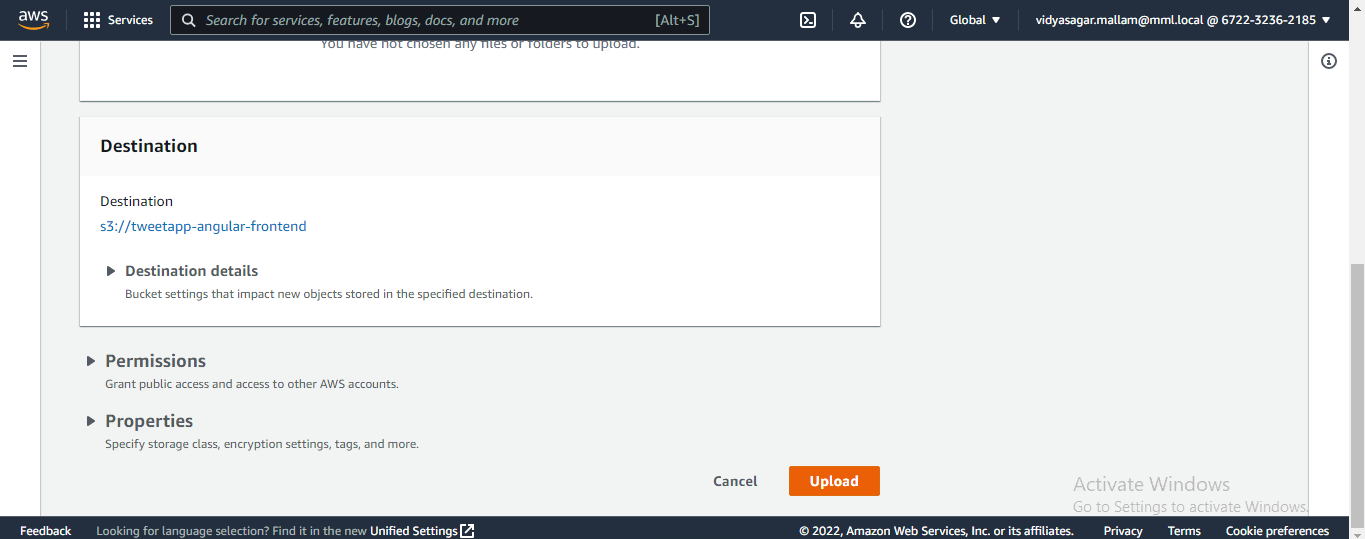


**Step-13:** click on add files and upload dist folder

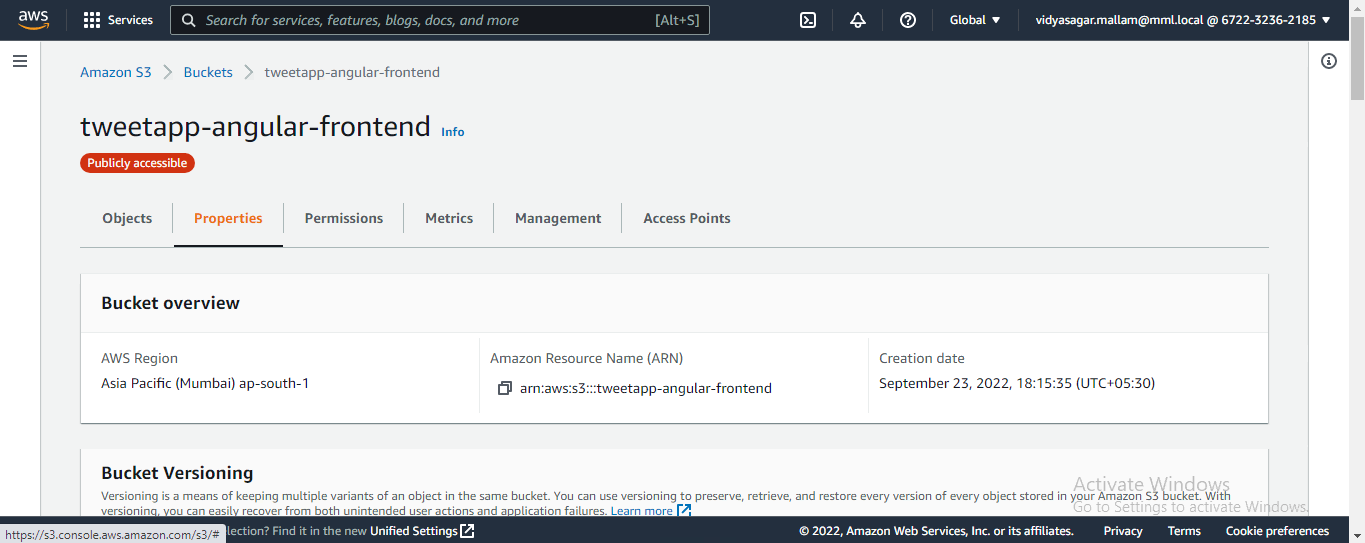




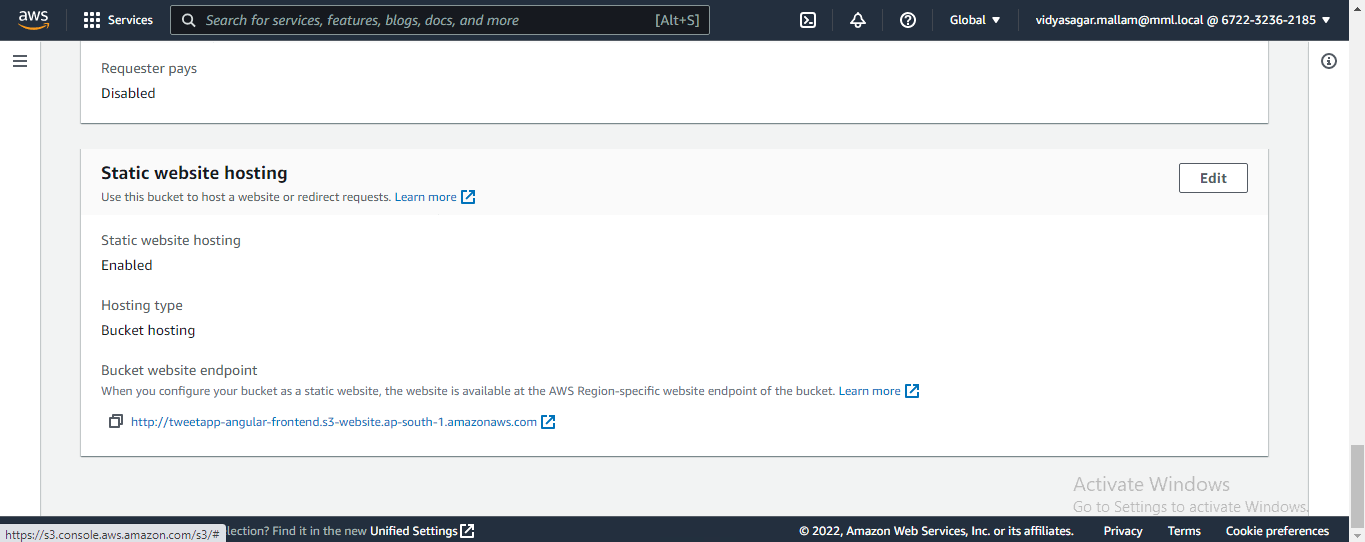
**Step-14:** Once files are uploaded click on the upload

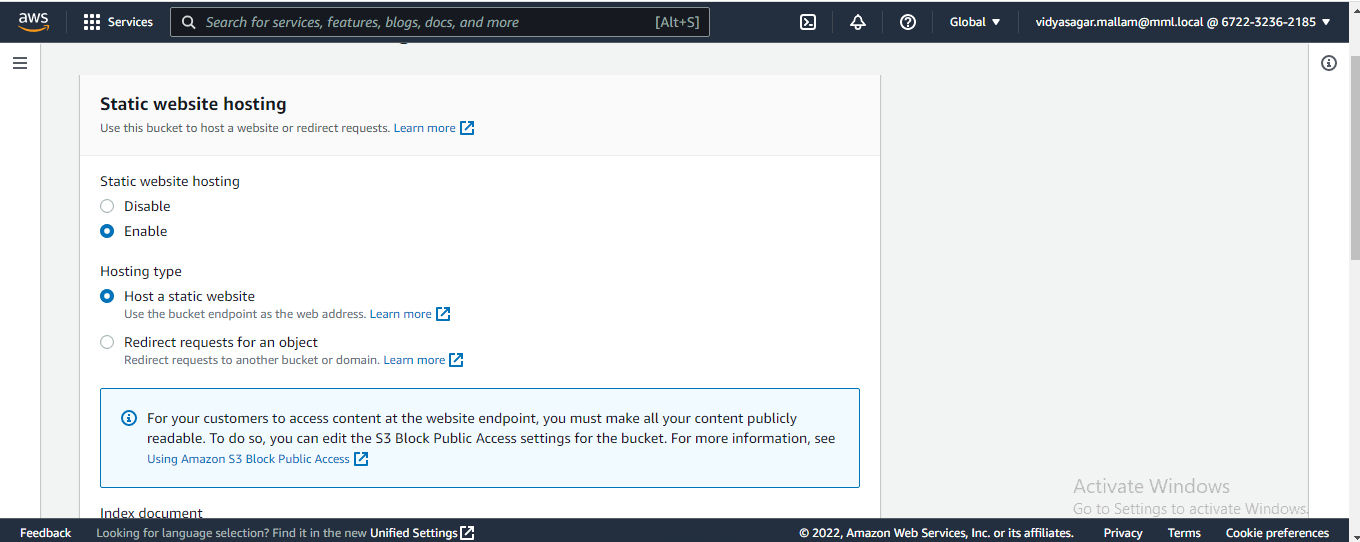


**Step-15:** Next go to properties tab

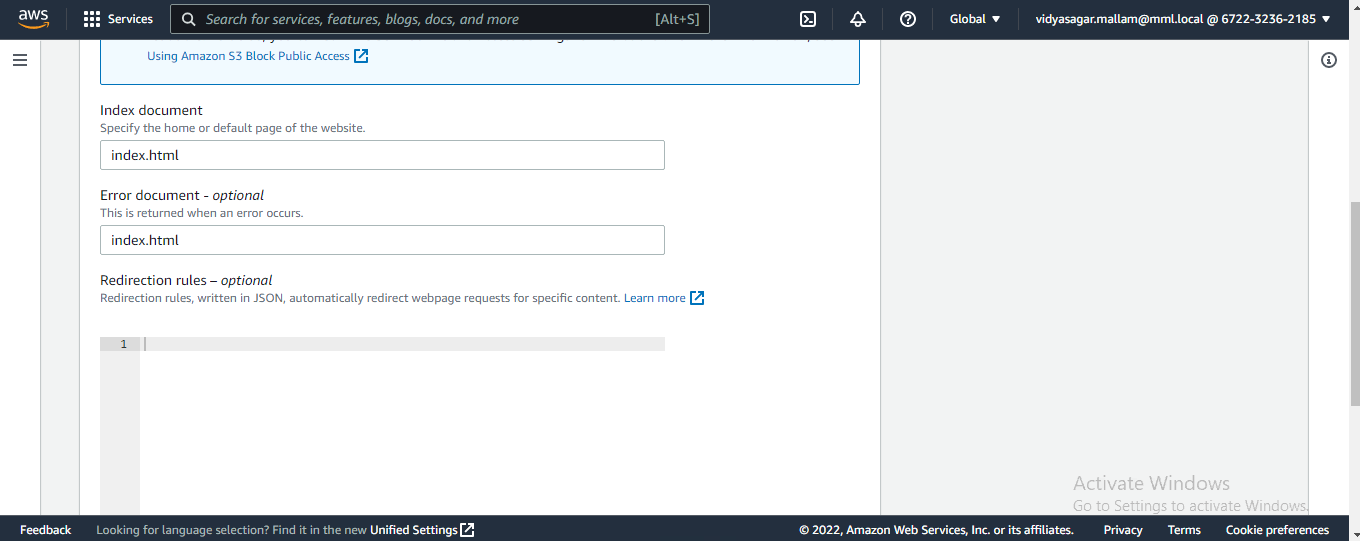


**Step-16:** go to static website hosting and click on edit and enable static website hosting

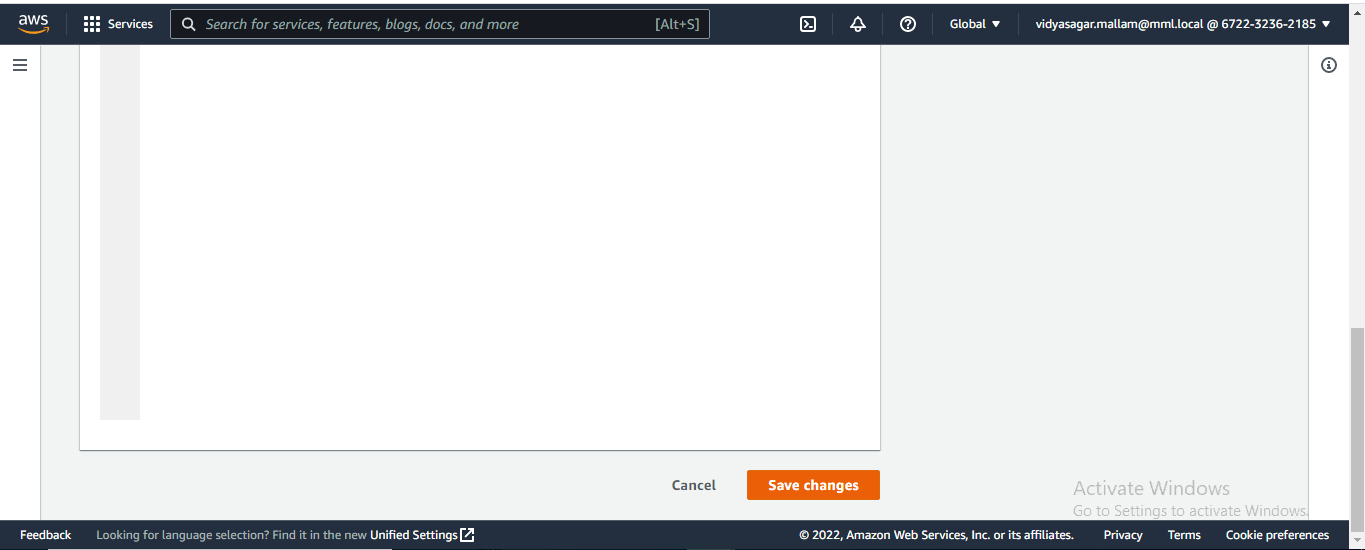




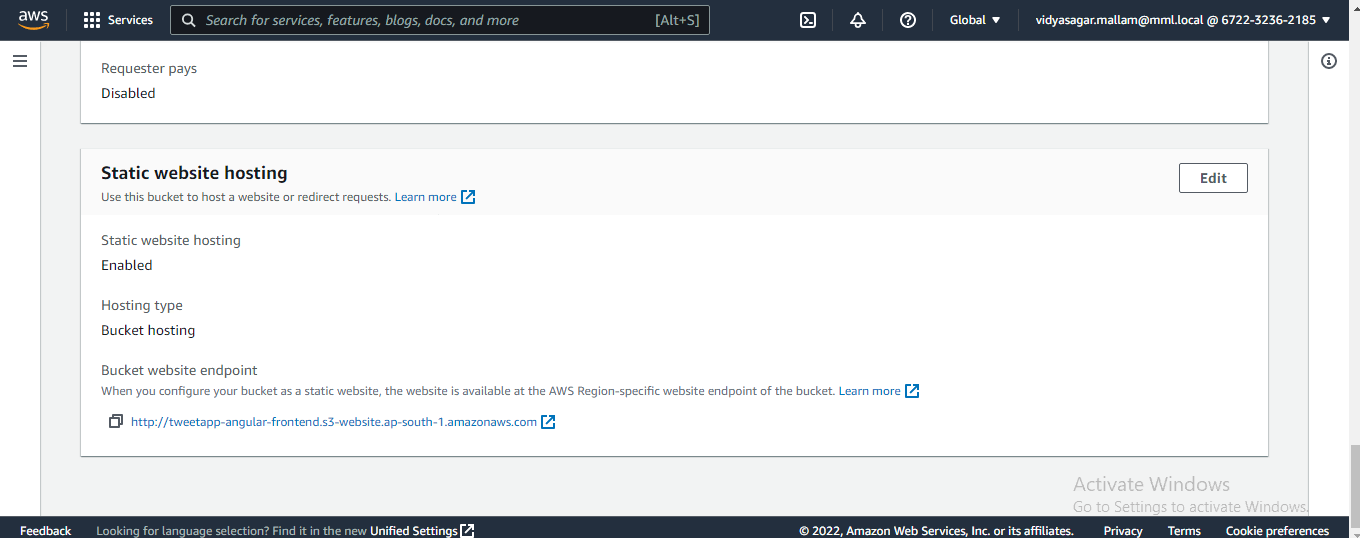
**Step-18:** now add index.html in index document



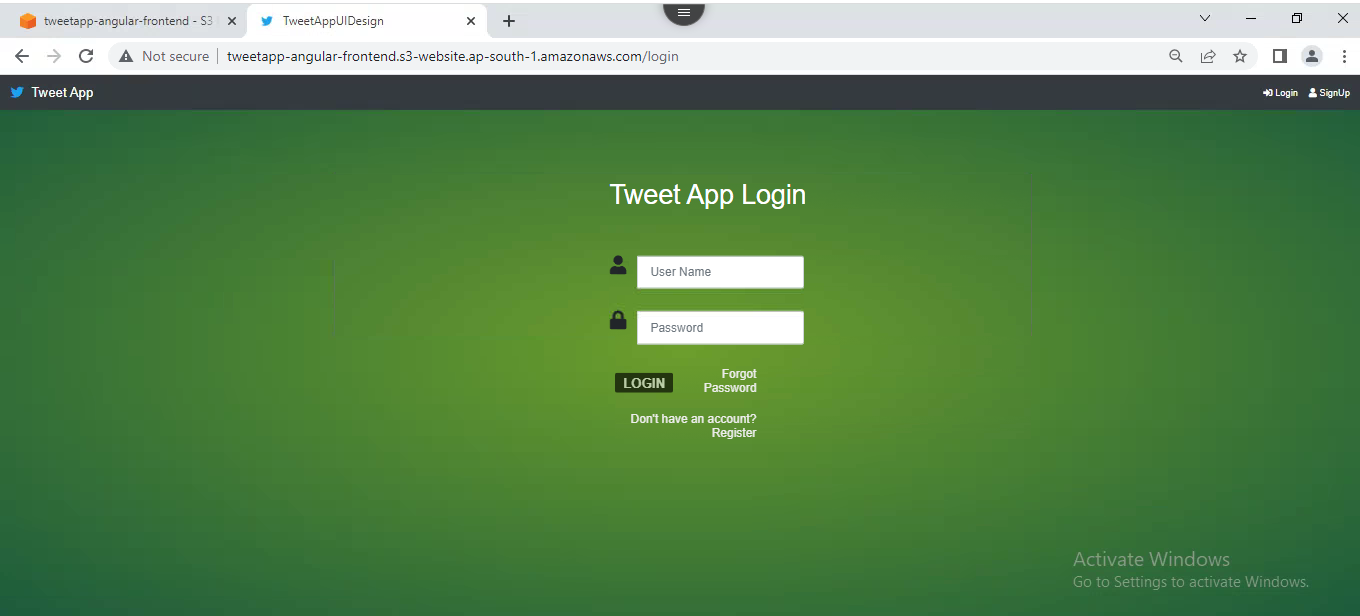
**Step-19:** And save changes



**Step-20:** now we can use the below link for our tweetApp (application will be running on this link)



**Step-21:** We can use tweetApp with this link



Mallam Vidya Sagar

[VidyaSagar.Mallam@cognizant.com](mailto:VidyaSagar.Mallam@cognizant.com)

921463

Fse-1 (TweetApp)