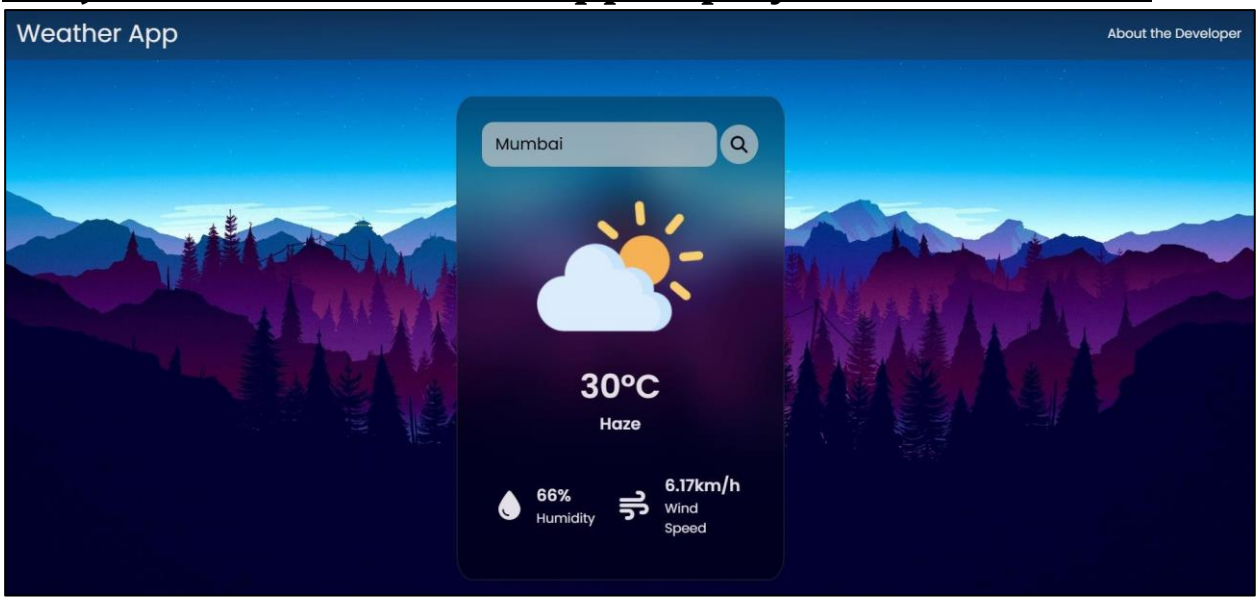
Project: Secure Weather App Deployment on AWS S3



• Project Overview:

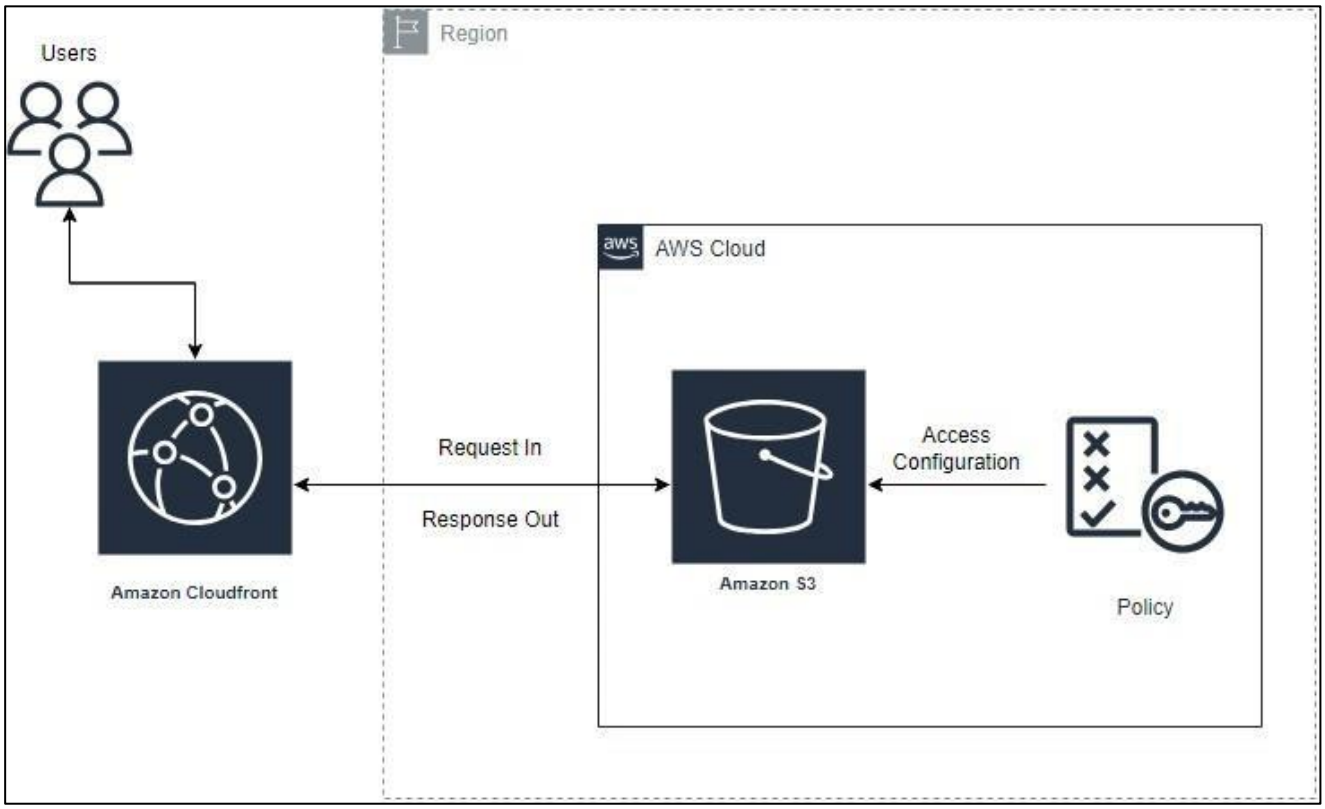
1. **Description:** This project involves deploying a static website (Weather App) on Amazon S3 to make it accessible over HTTP/HTTPS.

2. **Objective:** Set up static hosting on AWS S3, configure HTTP/HTTPS, and ensure the website is available publicly with the help of Amazon CloudFront.

• Technologies Used:

1. **AWS Services Used:**
   - Amazon S3 - Amazon CloudFront
2. **For WebApplication:**
   - HTML
   - CSS
   - JavaScript
   - OpenWeatherAPI

AWS Cloud Architecture:



- In this project, this is the architecture I have built for the website hosting.

Note: The main objective of this project is deploy the weather webapp securely on the Internet.

Part 1:

Website Creation:

- The Weather WebApp is one of my old project I built few months ago.
- As this project focuses on deployment, I will provide source code for the website.

Part 2:

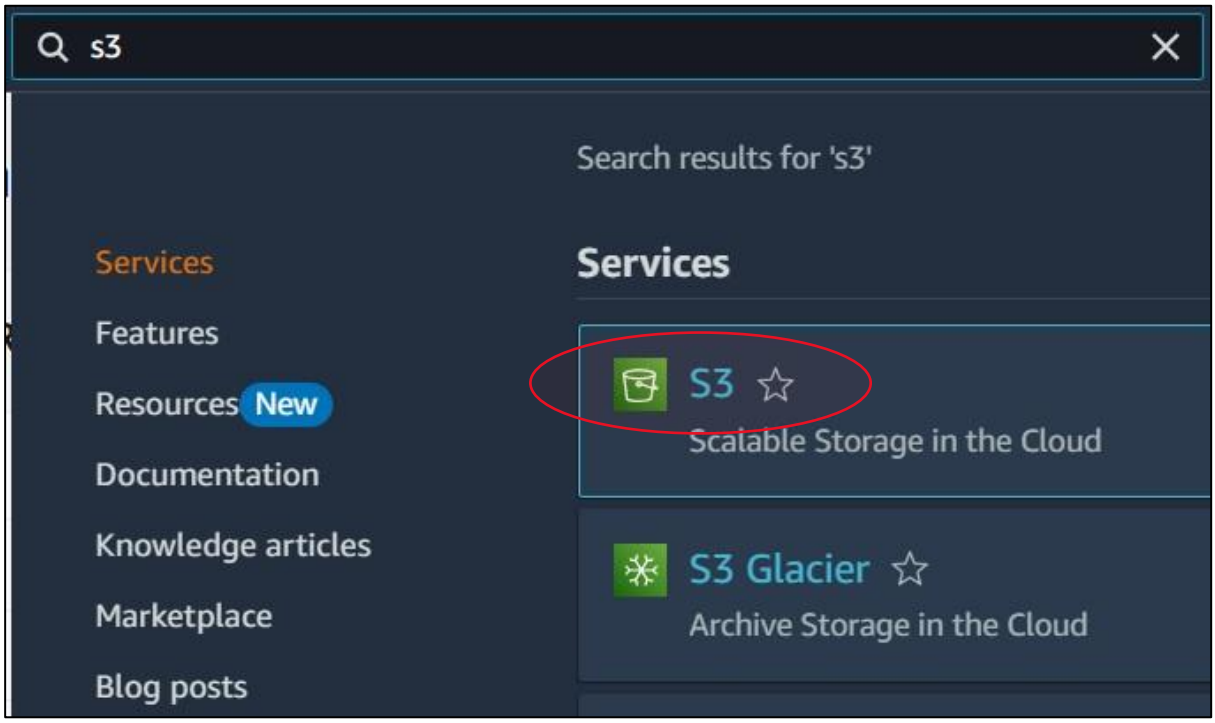
Deployment Process:

Note: if you don't have IAM user create one from the root user. with AmazonS3FullAccess and CloudFrontFullAccess policies.

Follow this guide for creating IAM User: IAM User creation

1. Creating S3 bucket using IAM user with following configurations:

a. Go to S3 Service → Create Bucket



Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region

Asia Pacific (Mumbai) ap-south-1

Bucket name Info

Unique Bucket Name!

weather_webapp

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

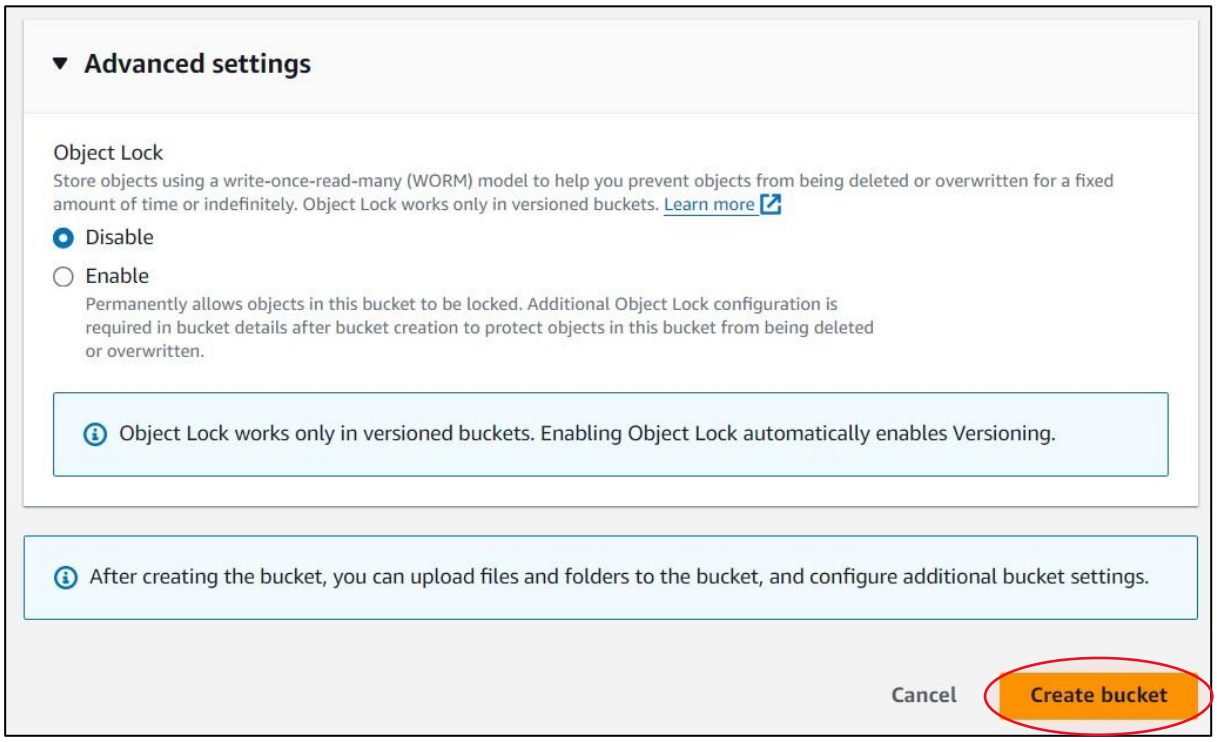
- (●) ACLs disabled (recommended)
  All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.
- ( ) ACLs enabled
  Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

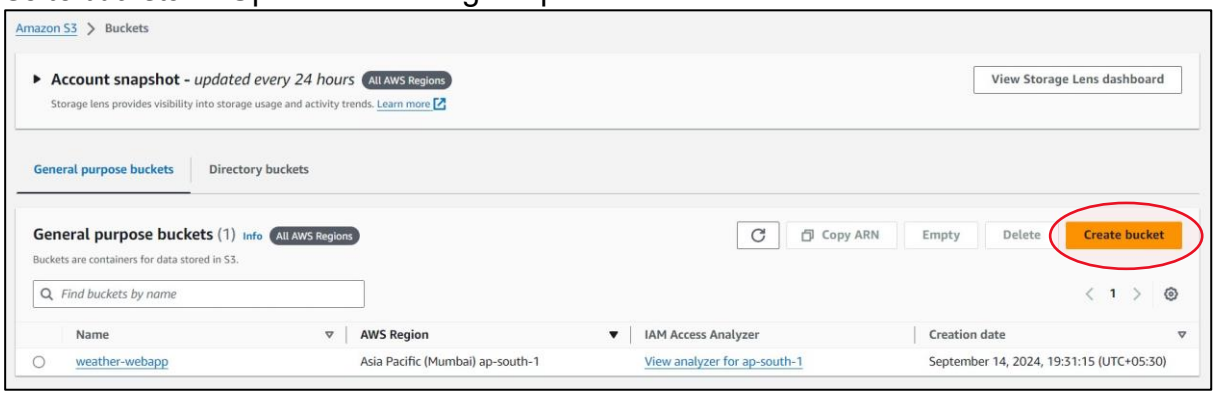
Allow Public Access

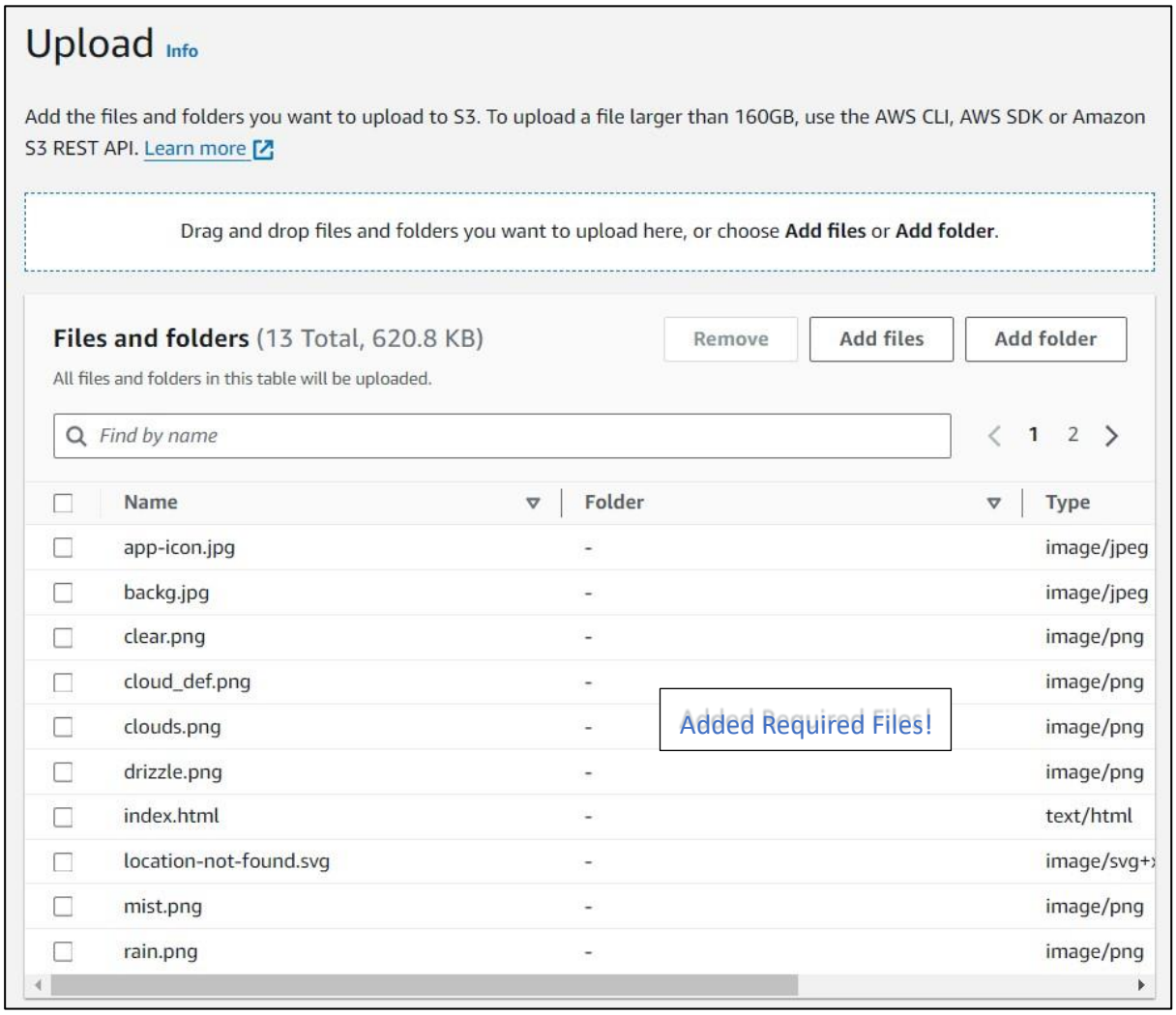
- [ ] **Block *all* public access**
  Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.
  - [ ] **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
    S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
  - [ ] **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
    S3 will ignore all ACLs that grant public access to buckets and objects.
  - [ ] **Block public access to buckets and objects granted through *new* public bucket or access point policies**
    S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
  - [ ] **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
    S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.



b. after successfully creating bucket:

Go to buckets → Upload → Adding Required Files:



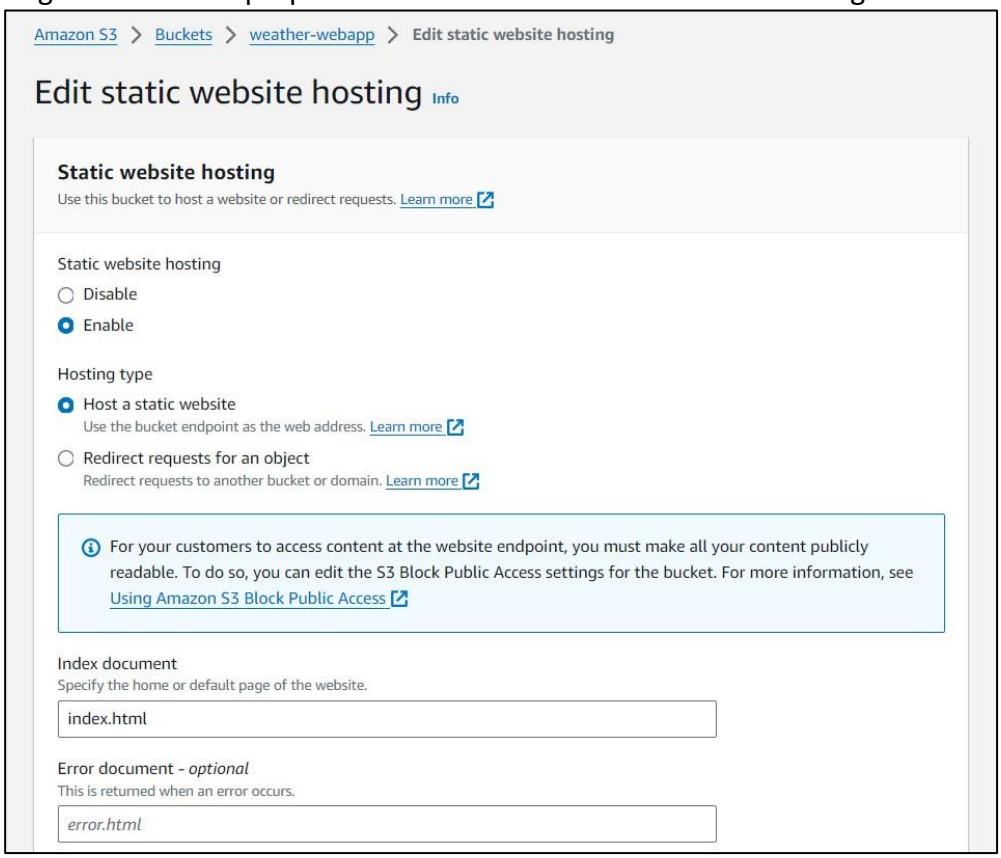


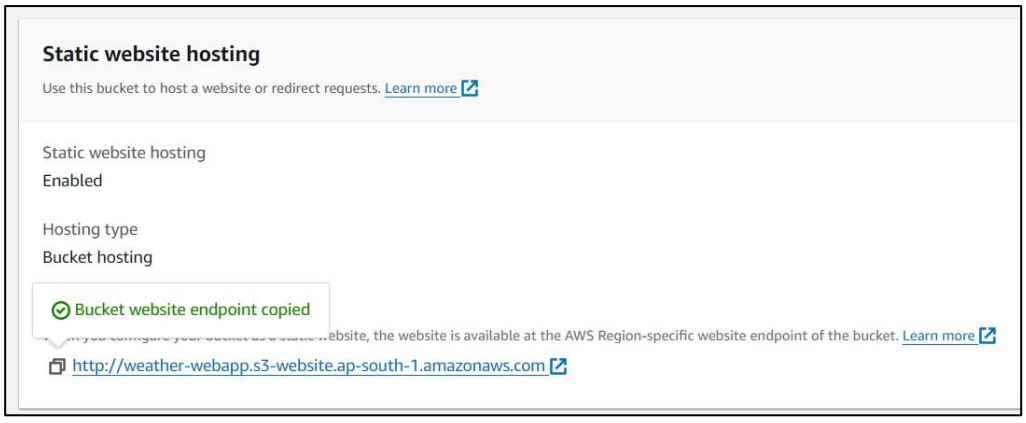
After successfully adding the required files, I kept the properties default to Standard storage type as I need to access the file frequently.

For permissions I kept them default.

Now, click on upload.

c. go to bucket → properties → scroll to enable static web hosting:



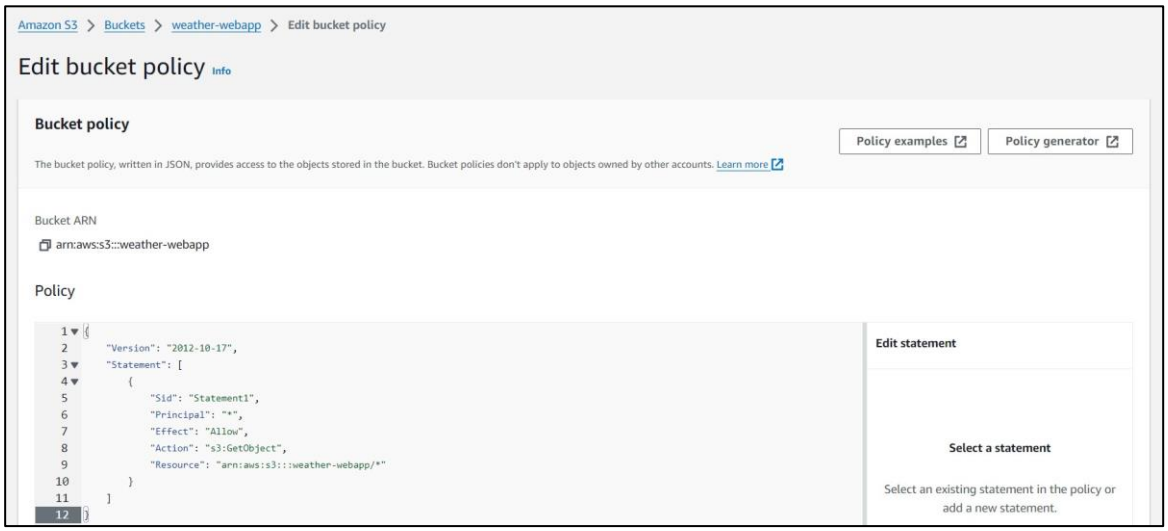


copy the given link to check whether website is working or not. paste it in new tab:

Access is Denied, as we have not configured the bucket policy.

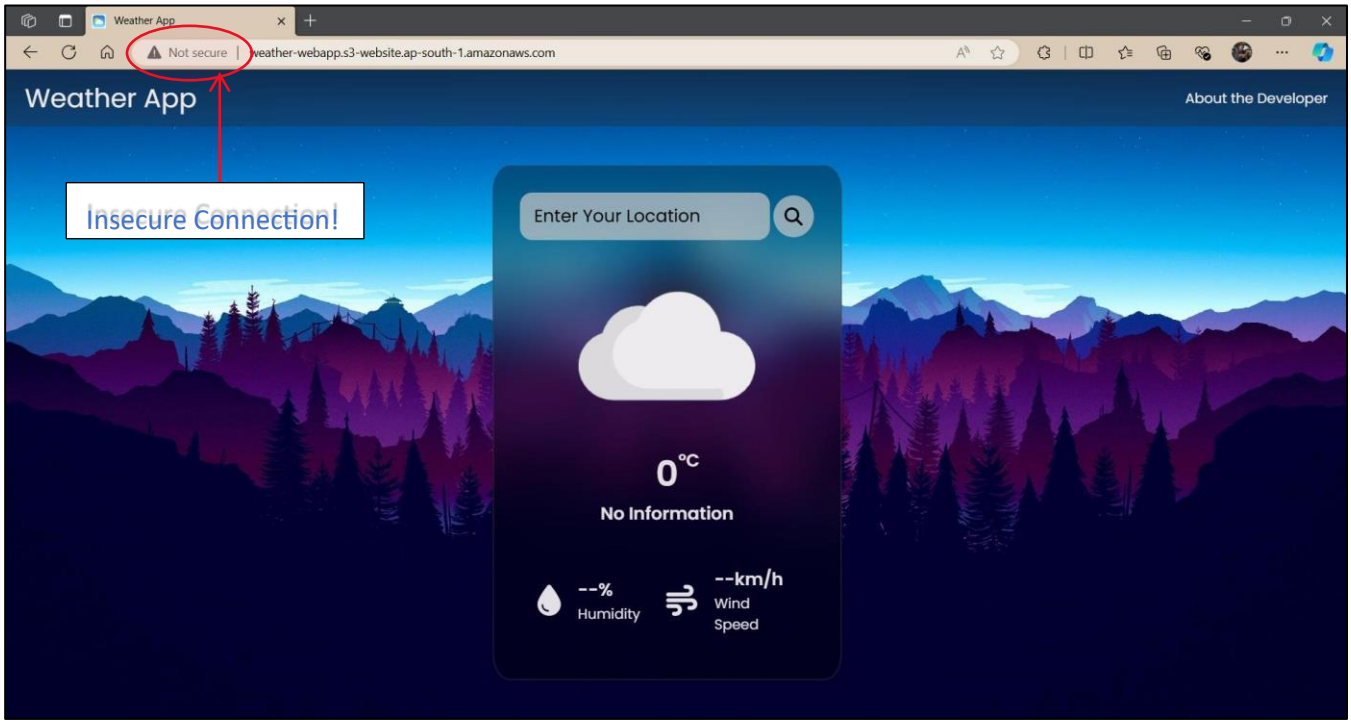
Configure bucket policy in Bucket's permissions tab and Add following Statement:

```
{
	"Version": "2012-10-17",
	"Statement": [
		{
			"Sid": "Statement1",
			"Effect": "Allow",
			"Principal": "*",
			"Action": "s3:GetObject",
			"Resource": "arn:aws:s3:::weather-webapp/*"
		}
	]
}
```



Save changes.

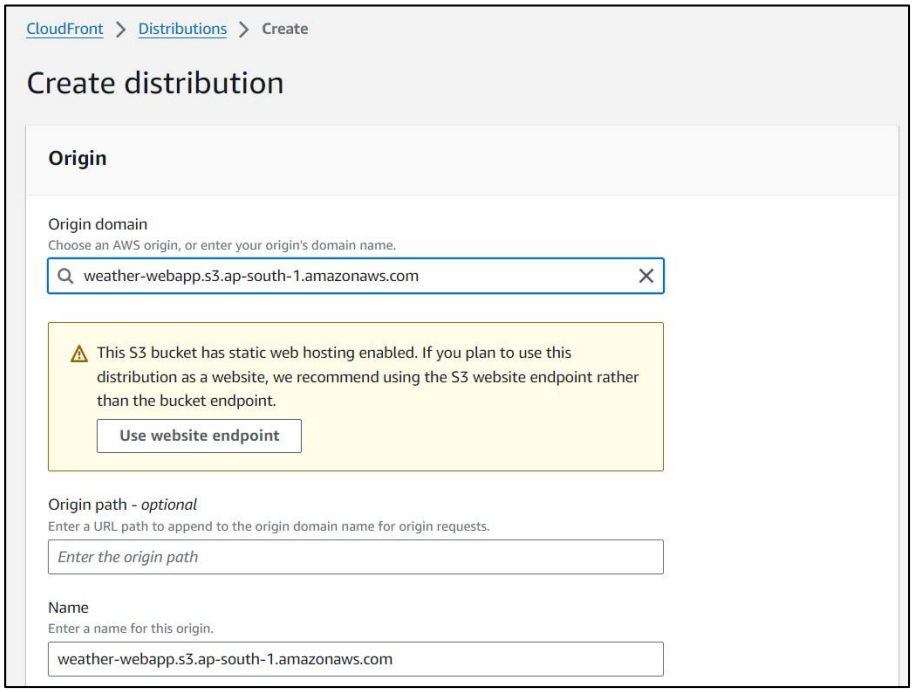
Now refresh the tab of Weather App webpage:



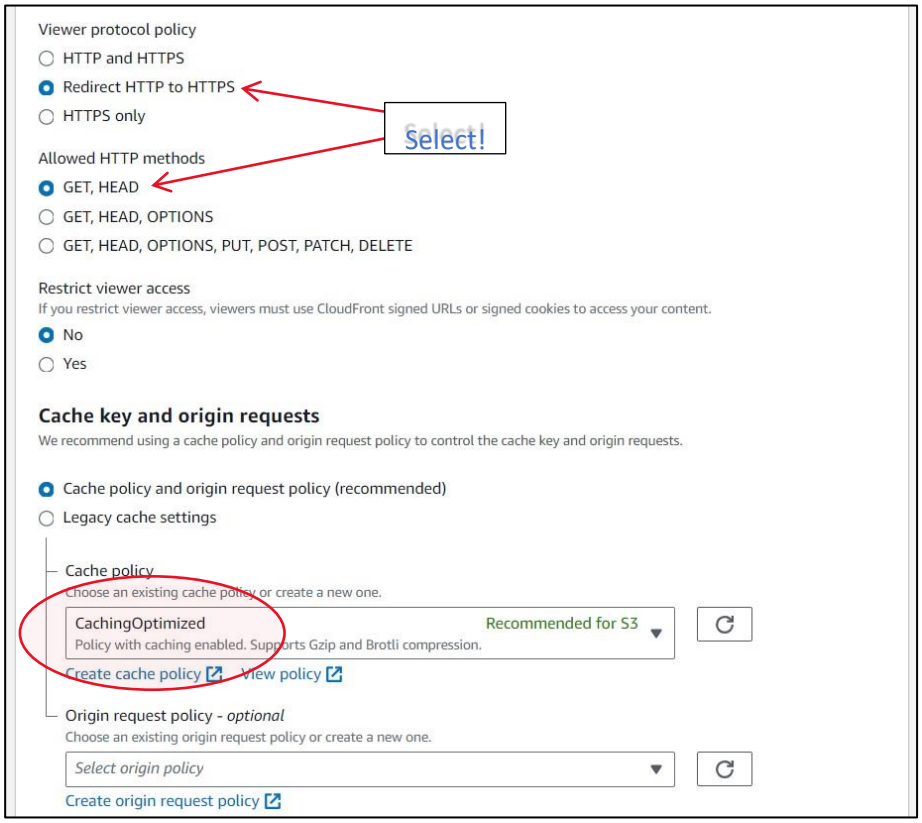
Successfully deployed the website on Amazon S3 but it is still insecure.

2. Securing the website using Amazon CloudFrount:

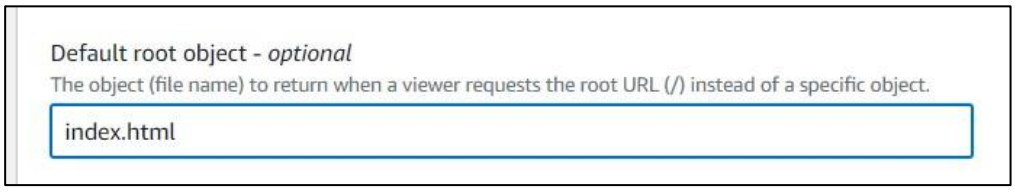
- Go to Services → CloudFront → Create Distribution



Selecting my origin domain name, i.e. same as static web url of the bucket.

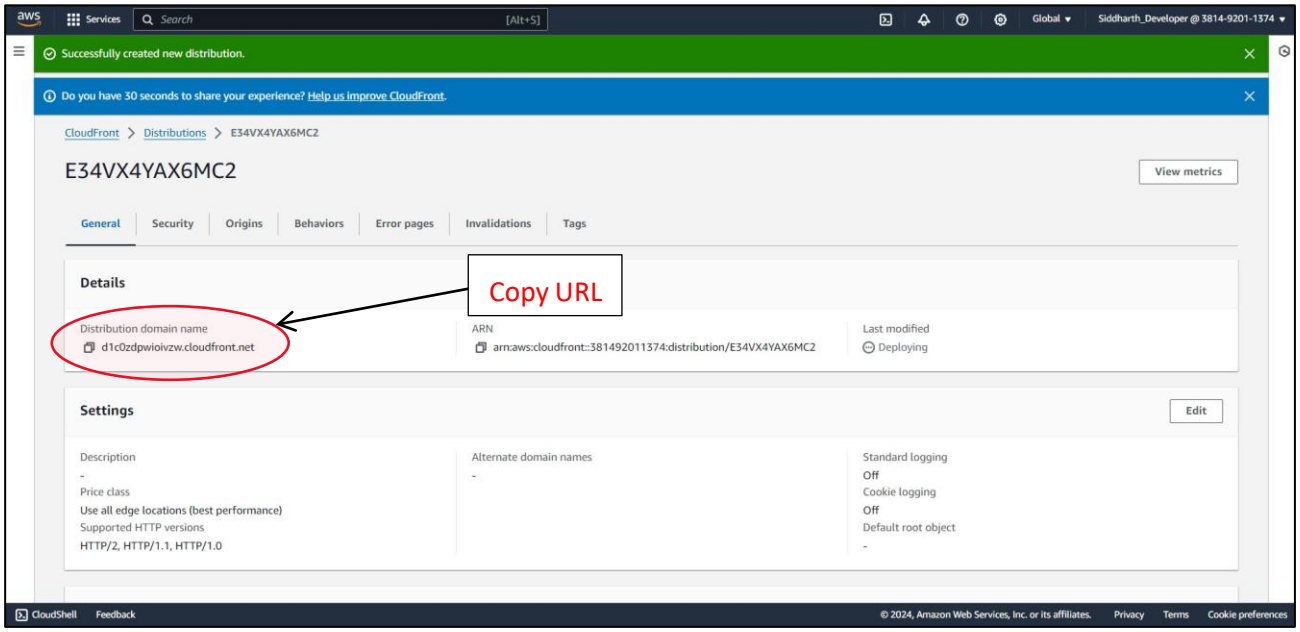


For default root object, give the name of homepage.

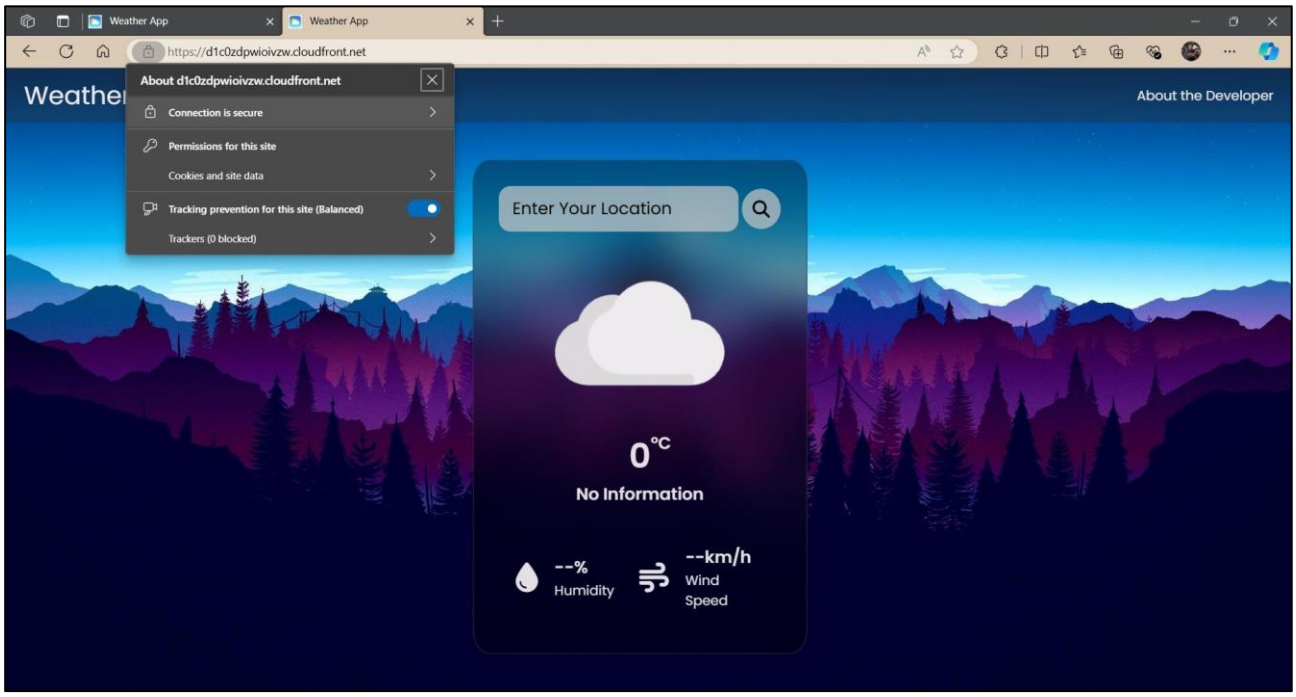


For WAF protection disable it and Create the Distribution.

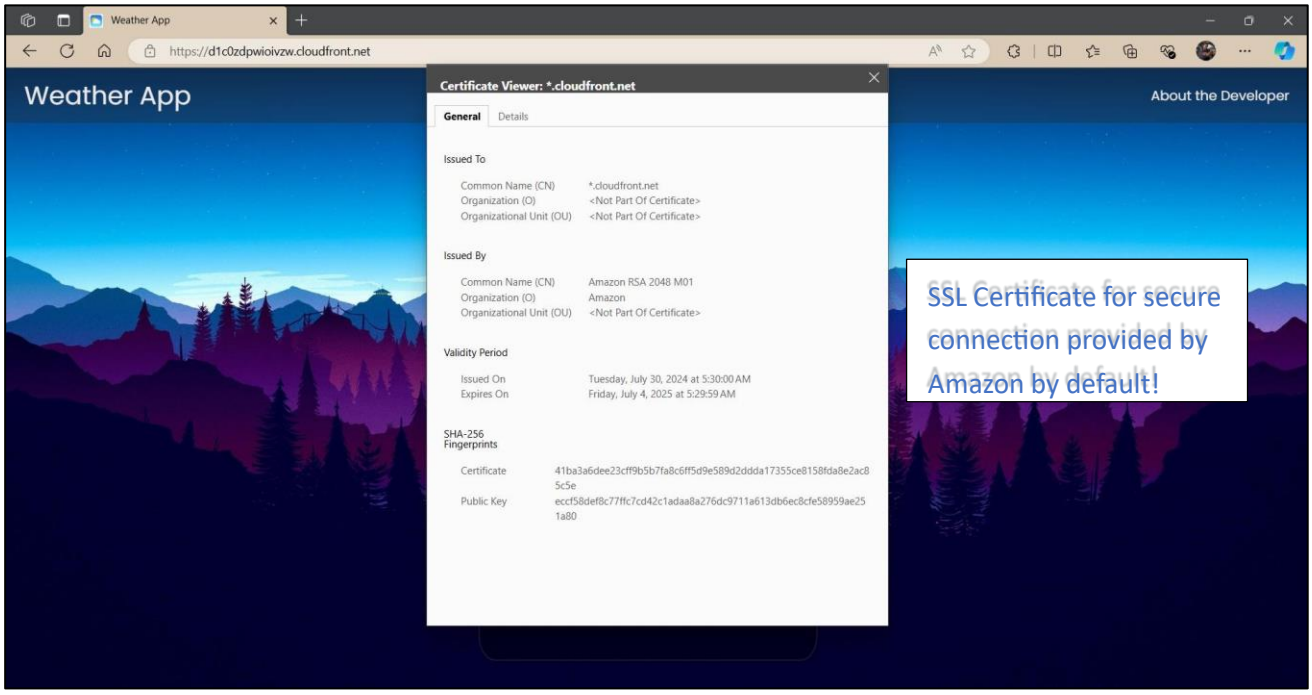
After successful creating, wait for the distribution to be available.



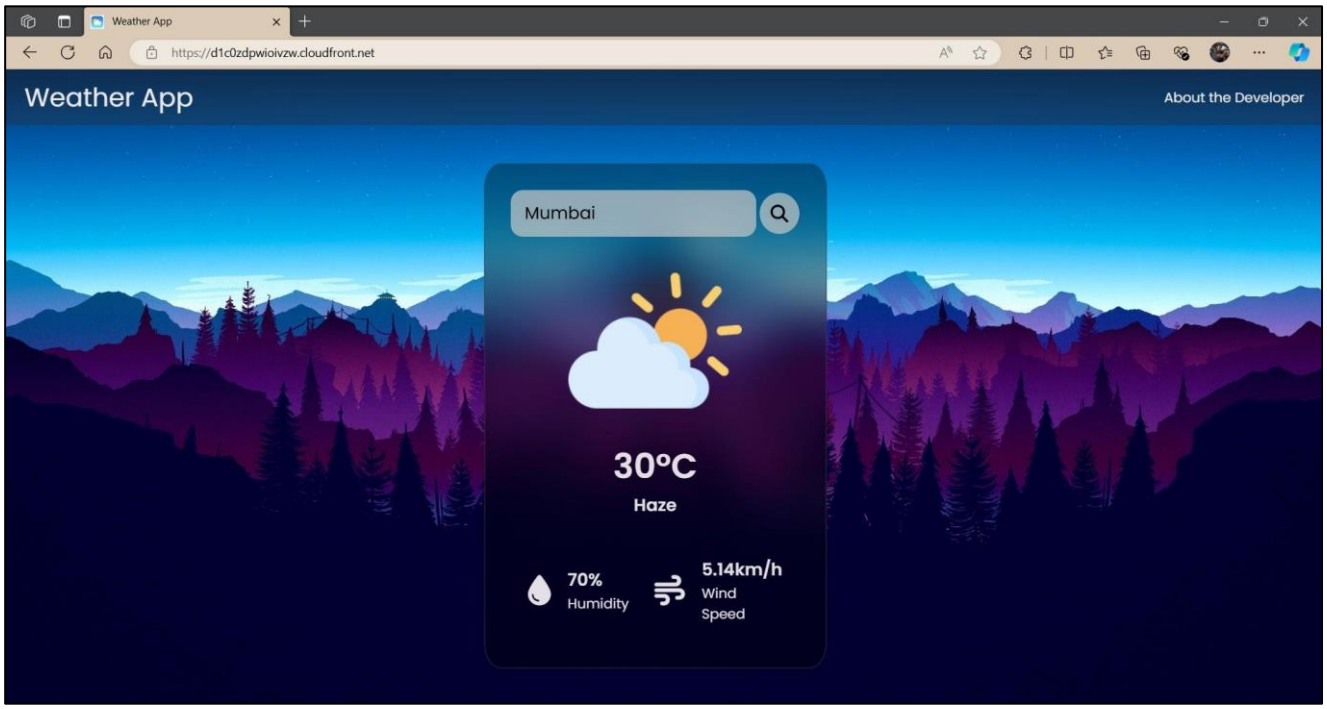
Copy the URL and paste in new tab:

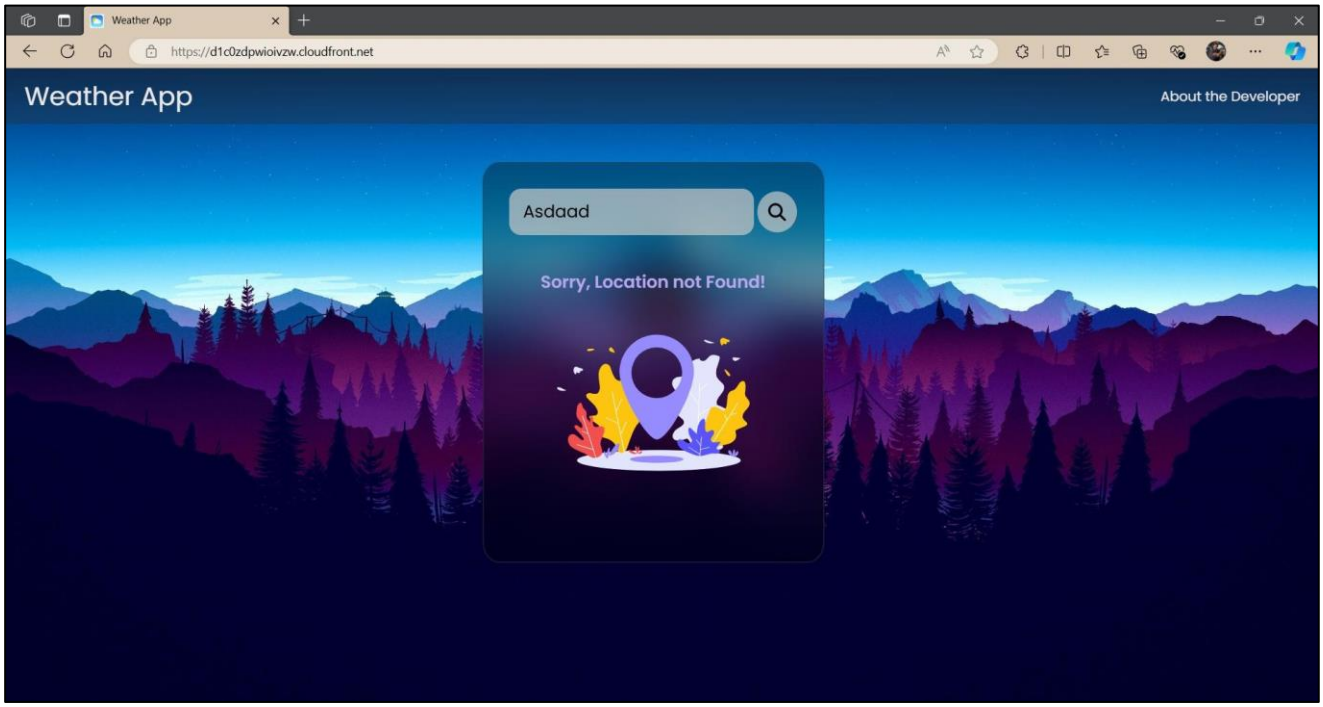


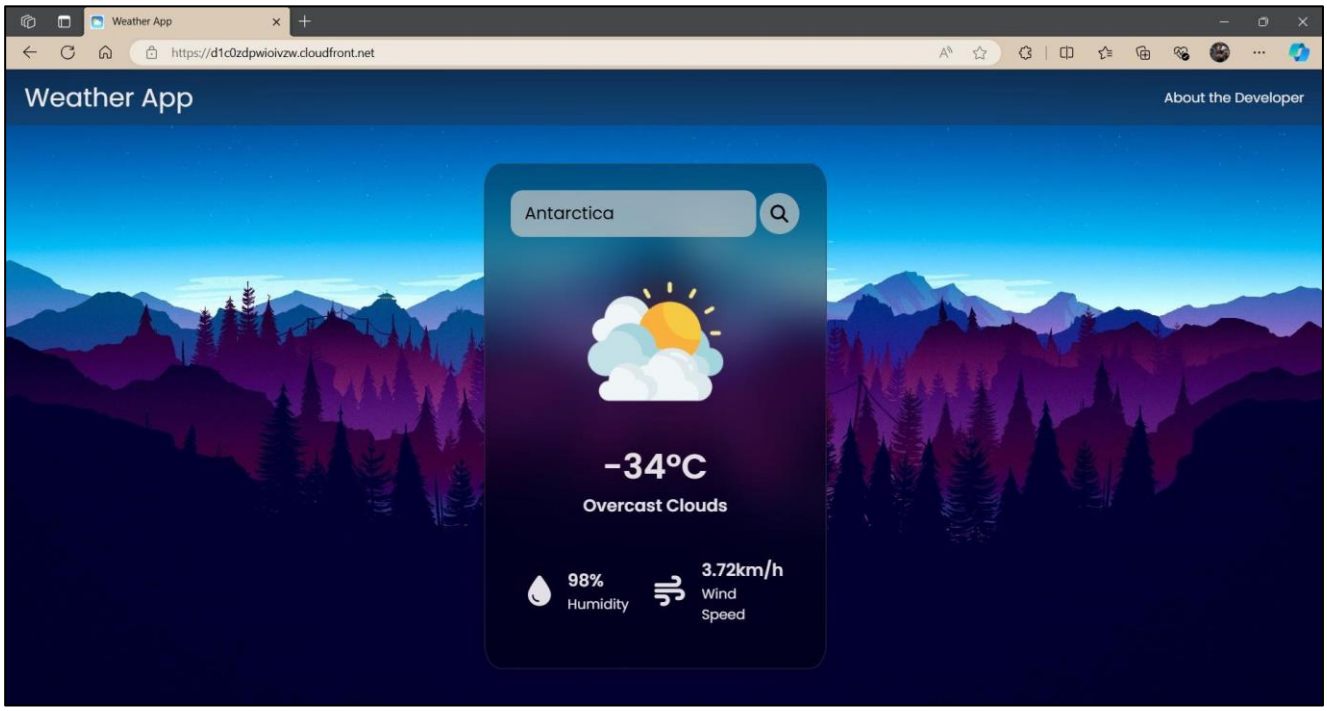
The has been launched successfully with secure HTTPS connection.



Part 3: Testing of the Weather WebApp


Weather App
https://d1c0zdpwioivzw.cloudfront.net
Weather App
About the Developer
Mumbai
30°C
Haze
70%
Humidity
5.14km/h
Wind
Speed


Weather App
https://d1c0zdpwioivzw.cloudfront.net
Weather App
About the Developer
Asdaad
Sorry, Location not Found!



Limitations:

- As you can see the domain name is by default of the AWS CloudFront, so as a free tier user with no Domain Name owning this is one of the limitations I faced.
- But to counter it we can use Amazon Route53 service to purchase/transfer owned domain to completely make website your but for that we also require SSL/TLS certificate which can be generated using Amazon ACM.
- For further implementation of Route53 and ACM certification you can visit:
- Website hosting using CloudFront and Route53
- Know more about generating certificate