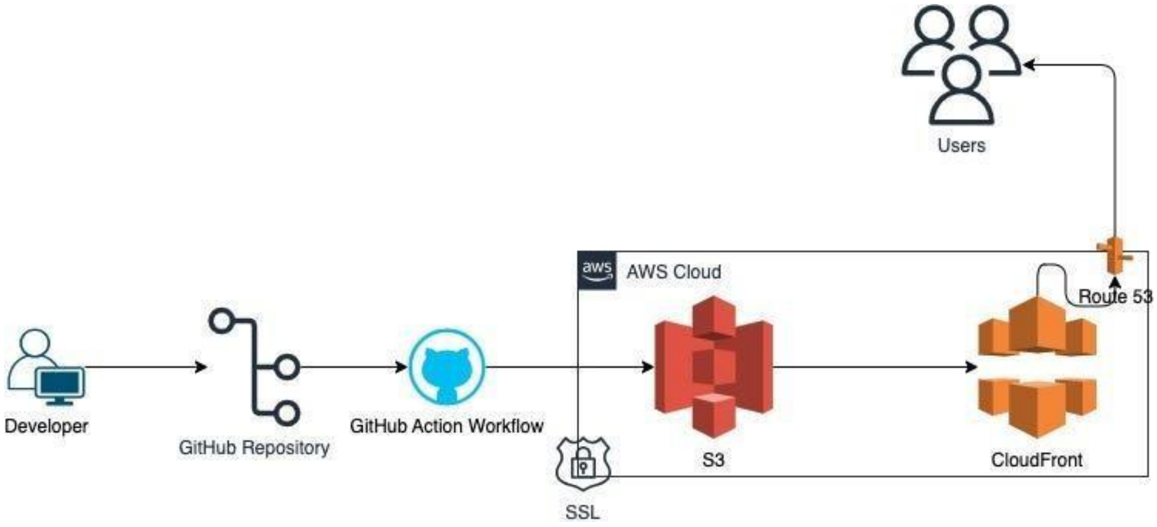
Project Topic :- Securely Deploying a Weather App on AWS S3



- **Project Overview**

1. Description: - Deploy a static Weather App on Amazon S3, ensuring secure public access over HTTPS.
2. Objective: - Set up AWS S3 for static website hosting.
3. Technologies Used: - AWS Services: Amazon S3, Amazon CloudFront.
4. Web Application: - HTML, CSS, JavaScript, OpenWeatherAPI.

- ## Deployment Process

1. Create an S3 Bucket

❖ Navigate to S3 Service → Create Bucket.

❖ Provide a unique bucket name and select the desired AWS region.

❖ Keep the default storage type unless specific requirements exist.

❖ Allow public access by configuring the bucket permissions after creation.

2. Upload Files to the Bucket

❖ Go to your bucket and click Upload.

❖ Add the required files (HTML, CSS, JS, images).

❖ For permissions, ensure files are set to public read for access.

3. Enable Static Website Hosting

❖ In the bucket's Properties tab, enable Static Website Hosting.

❖ Specify the index document (e.g., index.html).

❖ Copy the provided static website URL for testing.

Securing the Deployment

1. Configure Bucket Policy

- Access the bucket's Permissions tab and add the following policy:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::your-bucket-name/*"
    }
  ]
}
```

- Save changes and test the bucket URL.

2. Set Up CloudFront

- Navigate to CloudFront → Create Distribution.
- Use your S3 bucket URL as the Origin Domain Name.
- Specify the index document as the default root object.
- Leave Web Application Firewall (WAF) disabled for now.
- Create the distribution and wait for the status to change to Deployed.
- Copy the CloudFront URL and test it for HTTPS connectivity.

- **Optional Enhancements**

1. Use a Custom Domain

- ❖ Register a domain with Amazon Route 53 or any registrar.
- ❖ Map your domain to the CloudFront distribution using a CNAME record.

2. Enable SSL/TLS

- ❖ Use AWS Certificate Manager (ACM) to generate a free SSL/TLS certificate.
- ❖ Attach the certificate to your CloudFront distribution for secure communication.

3. Testing and Monitoring

- ❖ Test the app using the CloudFront URL to ensure secure HTTPS functionality.
- ❖ Monitor performance and logs via AWS CloudWatch.

4. Limitations

- ❖ - By default, the CloudFront URL uses AWS domain names. For complete branding, use Route 53 for custom domains and ACM for certificates.