Architecture & Justication of project

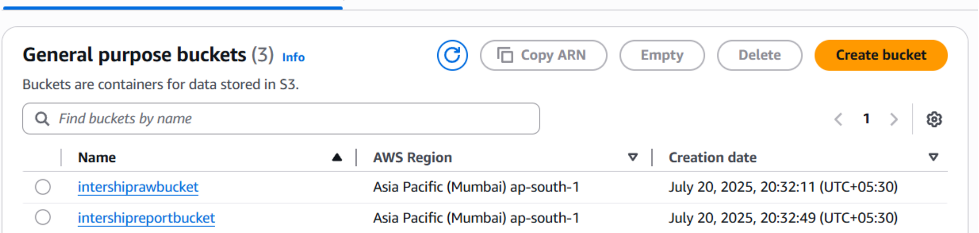
Justification of Design Choices

Serverless, Cloud-Native Foundation

- **AWS-managed services** (S3, Lambda, SNS, EventBridge) minimize operational overhead, automatically scale, and fit AWS Free Tier for cost-effective, beginner-friendly deployment.

Configure an S3 bucket:-

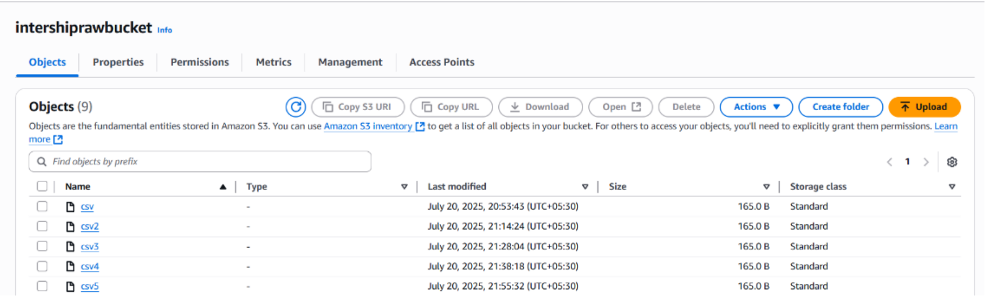
S3 is the landing zone for all raw data uploads (e.g., CSV files). S3 can automatically trigger AWS Lambda or Event Bridge when a new file is uploaded, enabling automated, event-driven processing. S3 can automatically trigger AWS Lambda or Event Bridge when a new file is uploaded, enabling automated, event-driven processing



Buckets are containers for data stored in S3.

	Name	AWS Region	Creation date
○	intershiprawbucket	Asia Pacific (Mumbai) ap-south-1	July 20, 2025, 20:32:11 (UTC+05:30)
○	intershipreportbucket	Asia Pacific (Mumbai) ap-south-1	July 20, 2025, 20:32:49 (UTC+05:30)

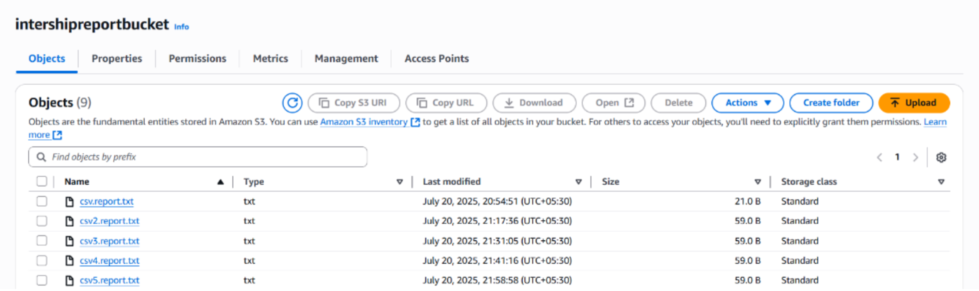
Raw Bucket data :-



Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	csv	-	July 20, 2025, 20:53:43 (UTC+05:30)	165.0 B	Standard
□	csv2	-	July 20, 2025, 21:14:24 (UTC+05:30)	165.0 B	Standard
□	csv3	-	July 20, 2025, 21:28:04 (UTC+05:30)	165.0 B	Standard
□	csv4	-	July 20, 2025, 21:38:18 (UTC+05:30)	165.0 B	Standard
□	csv5	-	July 20, 2025, 21:55:32 (UTC+05:30)	165.0 B	Standard

Report Bucket data:-



intershipreportbucket Info

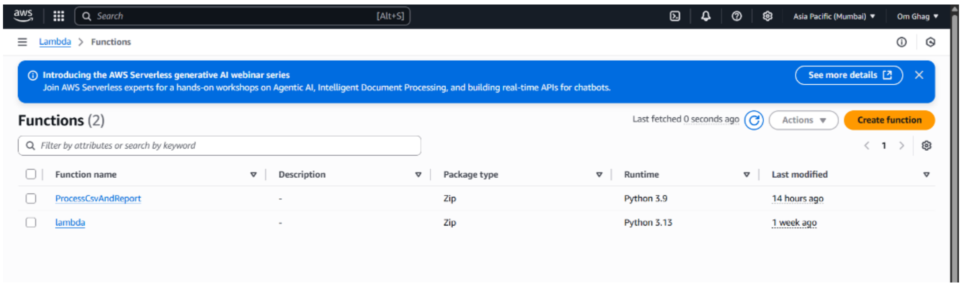
Objects (9)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class
csv.report.txt	txt	July 20, 2025, 20:54:51 (UTC+05:30)	21.0 B	Standard
csv2.report.txt	txt	July 20, 2025, 21:17:36 (UTC+05:30)	59.0 B	Standard
csv3.report.txt	txt	July 20, 2025, 21:31:05 (UTC+05:30)	59.0 B	Standard
csv4.report.txt	txt	July 20, 2025, 21:41:16 (UTC+05:30)	59.0 B	Standard
csv5.report.txt	txt	July 20, 2025, 21:58:58 (UTC+05:30)	59.0 B	Standard

Configrue an lambda Function (processcsvandreport) :-

AWS Lambda is used to automatically processes data files whenever a new file is uploaded to the S3 bucket. It acts as the “brains” of the pipeline: whenever triggered, Lambda reads the file, processes or summarizes the data, saves the report to another S3 bucket, and sends a notification via SNS. This enables fully automated, serverless data processing without any manual intervention or the need to manage servers.



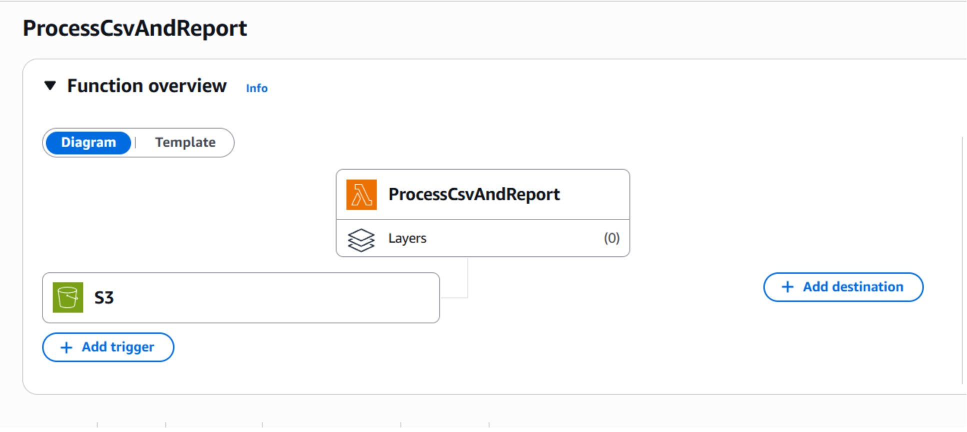
Introducing the AWS Serverless generative AI webinar series
Join AWS Serverless experts for a hands-on workshops on Agentic AI, Intelligent Document Processing, and building real-time APIs for chatbots.

Functions (2)

Last fetched 0 seconds ago

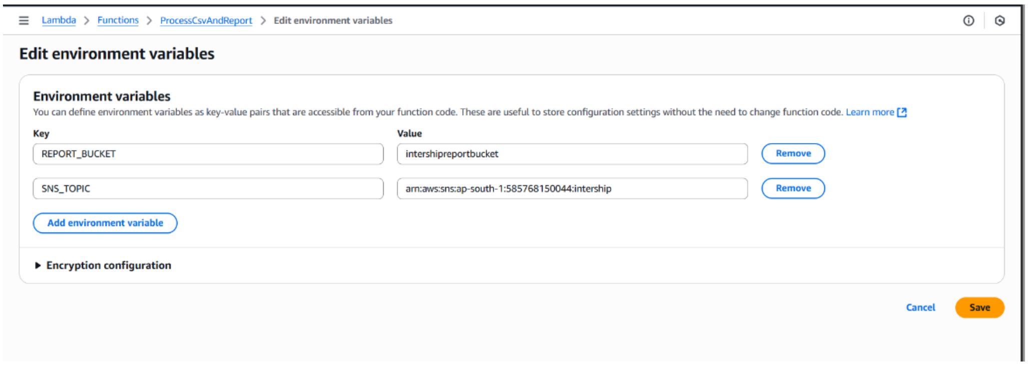
Function name	Description	Package type	Runtime	Last modified
ProcessCsvAndReport	-	Zip	Python 3.9	14 hours ago
lambda	-	Zip	Python 3.13	1 week ago

Lambda Trigger S3 bucket:-



ProcessCsvAndReport

Edited Environment Variable:-



Edit environment variables

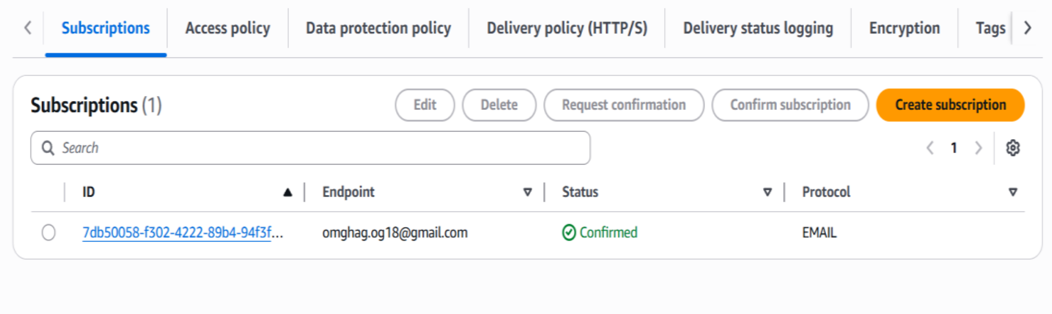
Environment variables

You can define environment variables as key-value pairs that are accessible from your function code. These are useful to store configuration settings without the need to change function code. Learn more

Key	Value
REPORT_BUCKET	intershipreportbucket
SNS_TOPIC	arn:aws:sns:ap-south-1:585768150044:intership

Configure a Simple Notification Service(SNS) service:-

Amazon SNS (Simple Notification Service) is used to automatically send real-time email notifications to users whenever important events occur—such as when a new data file is processed and a report is generated. After AWS Lambda completes processing a file and storing the result, it publishes a message to an SNS topic, which immediately delivers alerts (like emails) to all subscribed recipients. This ensures stakeholders are promptly informed about pipeline outcomes, errors, or scheduled reports, enabling seamless, automated communication without manual effort

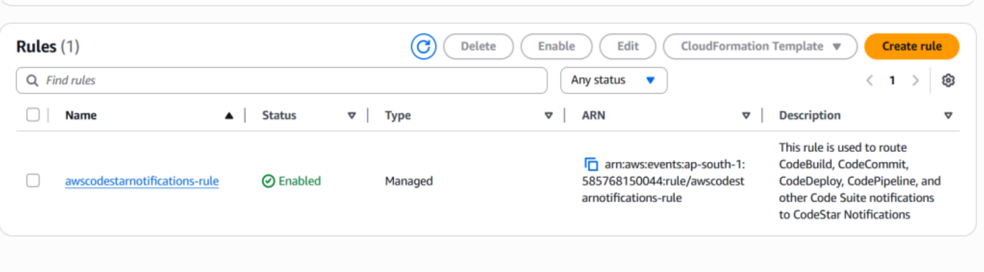


Subscriptions (1)

	ID	Endpoint	Status	Protocol
○	7db50058-f302-4222-89b4-94f3f...	omghag.og18@gmail.com	✓ Confirmed	EMAIL

Amazon EventBridge:-

Schedules automated executions (e.g., daily reports) and routes events to Lambda functions for scheduled or advanced automation.

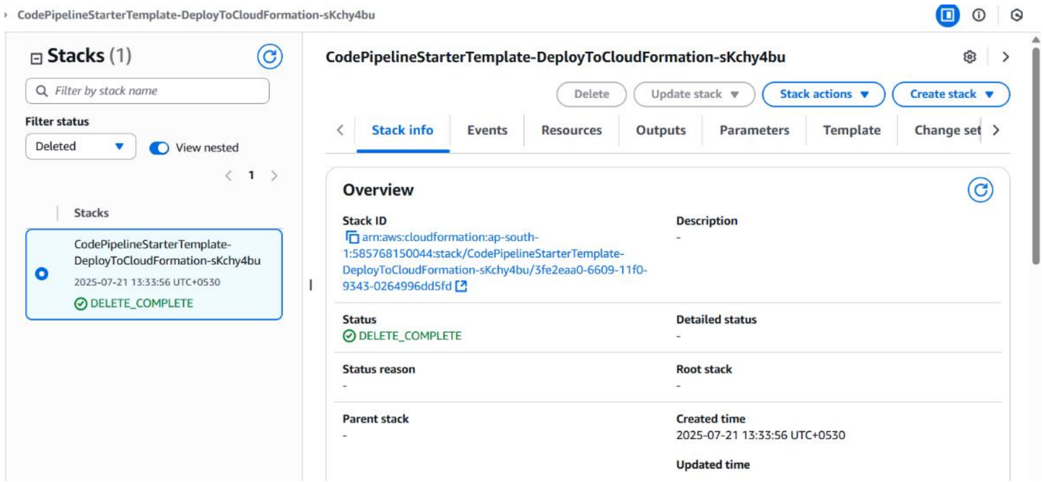


Rules (1)

□	Name	Status	Type	ARN	Description
□	awscodestarnotifications-rule	✓ Enabled	Managed	arn:aws:events:ap-south-1:585768150044:rule/awscodestarnotifications-rule	This rule is used to route CodeBuild, CodeCommit, CodeDeploy, CodePipeline, and other Code Suite notifications to CodeStar Notifications

AWS CloudFormation :-

Infrastructure as Code (IaC) service to automatically provision and manage all AWS resources declaratively through template files.



CodePipelineStarterTemplate-DeployToCloudFormation-sKchy4bu

Filter status: Deleted ▼

View nested

Stacks

● CodePipelineStarterTemplate-DeployToCloudFormation-sKchy4bu
2025-07-21 13:33:56 UTC+0530
✓ DELETE_COMPLETE

CodePipelineStarterTemplate-DeployToCloudFormation-sKchy4bu

Overview

Stack ID
arn:aws:cloudformation:ap-south-1:585768150044:stack/CodePipelineStarterTemplate-DeployToCloudFormation-sKchy4bu/3fe2eaa0-6609-11f0-9343-0264996dd5fd

Description
-

Status
✓ DELETE_COMPLETE

Detailed status
-

Status reason
-

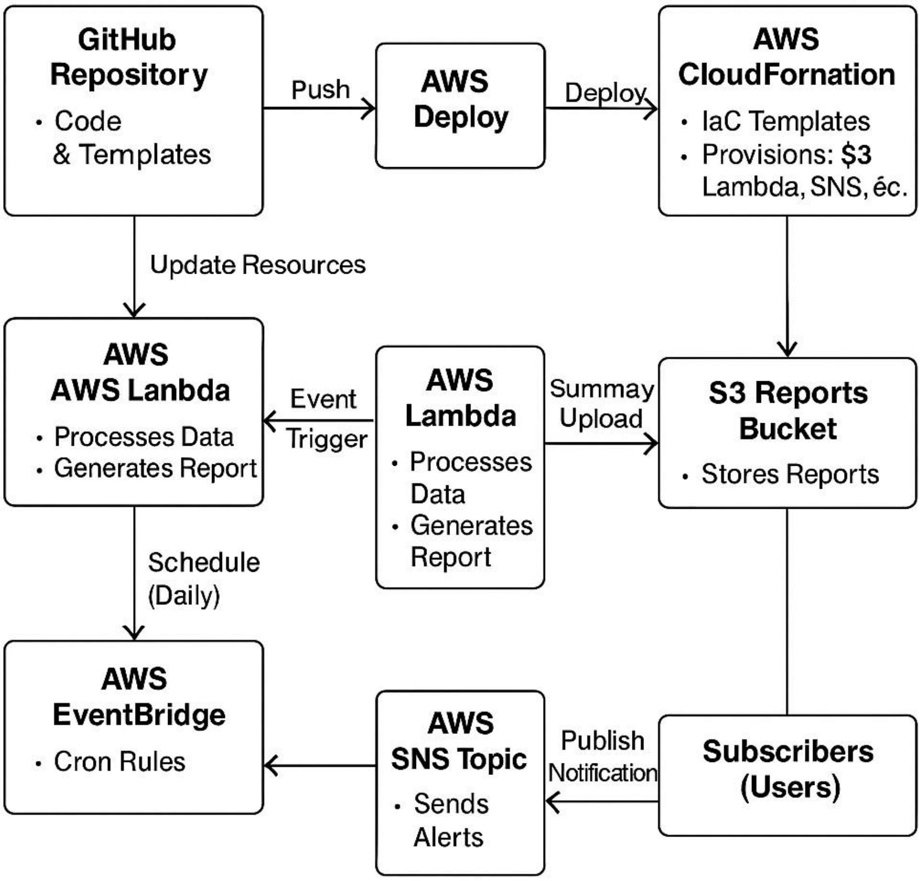
Root stack
-

Parent stack
-

Created time
2025-07-21 13:33:56 UTC+0530

Updated time

Architecture Diagram:-



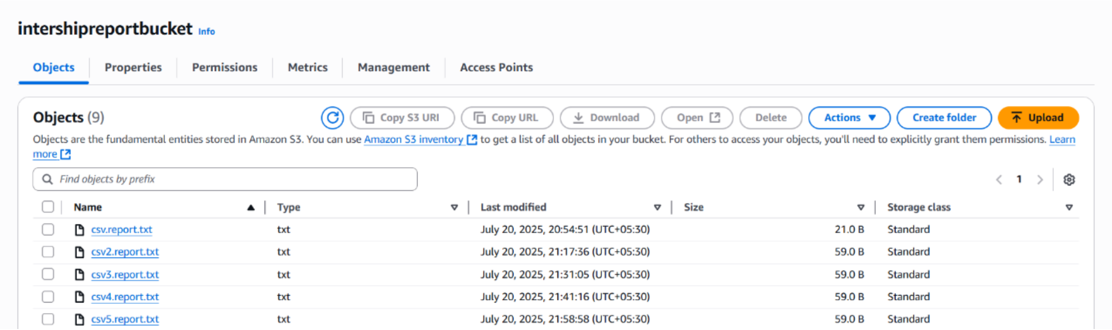
The above architecture diagram implementing an event-driven automated reporting system on AWS, using serverless architecture, Infrastructure as Code (IaC), and Continuous Integration/Continuous Deployment (CI/CD) practices. The solution begins with a GitHub repository that contains all source code and AWS CloudFormation templates. Upon code changes, AWS Deploy automatically triggers the deployment process, and AWS CloudFormation provisions the necessary resources, including Lambda functions, an S3 bucket, and an SNS topic. The core processing is handled by AWS Lambda, which is scheduled to run daily using AWS EventBridge with cron rules. This Lambda function processes the incoming data and generates summary reports, which are then uploaded to an Amazon S3 bucket for storage and future access. Once the report is successfully uploaded, AWS SNS is used to send alert notifications to subscribed users, informing them of report availability. This end-to-end automated pipeline ensures reliability, scalability, and maintainability, while reducing manual effort in report generation and distribution.

Final Output :-

when we upload file in S3 bucket:-

When we upload a CSV file in a S3 bucket it will trigger the lambda function and give a email notification and generate a report file to another bucket.

Report file created in another bucket:-



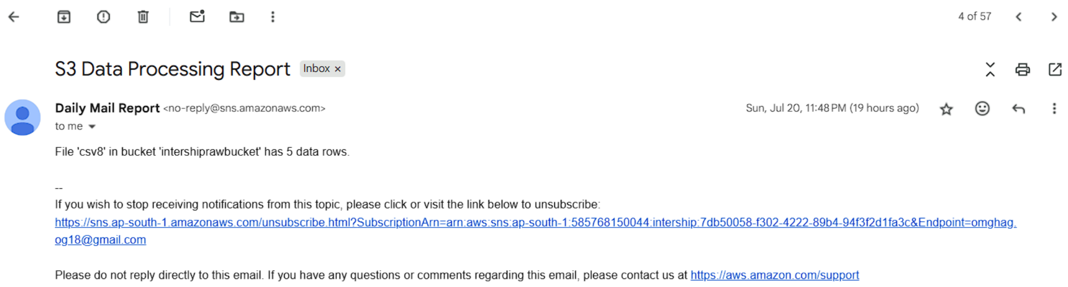
intershipreportbucket

Objects (9)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

□	Name	Type	Last modified	Size	Storage class
□	csv.report.txt	txt	July 20, 2025, 20:54:51 (UTC+05:30)	21.0 B	Standard
□	csv2.report.txt	txt	July 20, 2025, 21:17:36 (UTC+05:30)	59.0 B	Standard
□	csv3.report.txt	txt	July 20, 2025, 21:31:05 (UTC+05:30)	59.0 B	Standard
□	csv4.report.txt	txt	July 20, 2025, 21:41:16 (UTC+05:30)	59.0 B	Standard
□	csv5.report.txt	txt	July 20, 2025, 21:58:58 (UTC+05:30)	59.0 B	Standard

Email Notification to subscriber:-



S3 Data Processing Report

Daily Mail Report <no-reply@sns.amazonaws.com>

Sun, Jul 20, 11:48 PM (19 hours ago)

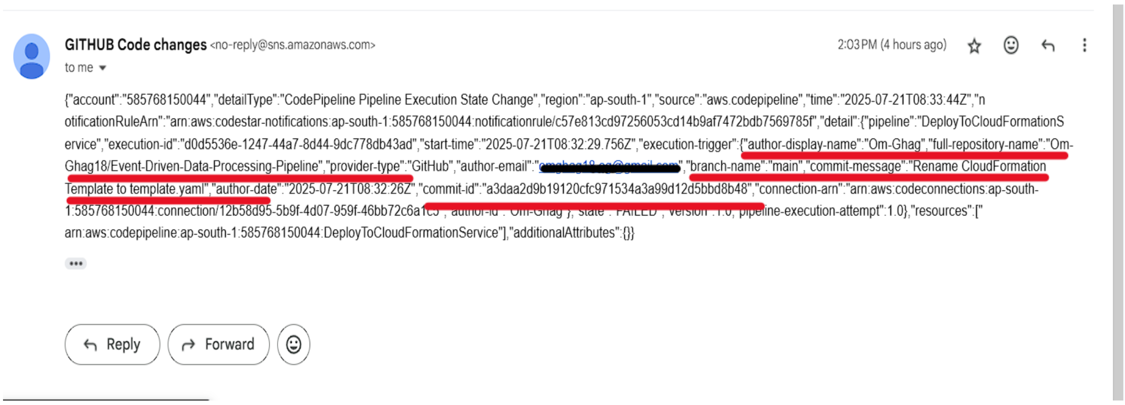
File 'csv8' in bucket 'intershiprawbucket' has 5 data rows.

--
If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
https://sns.ap-south-1.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:ap-south-1:585768150044:intership:7db50058-f302-4222-89b4-94f3f2d1fa3c&Endpoint=omghag.og18@gmail.com

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at https://aws.amazon.com/support

when we commit or push code from GITHUB:-

When we commit any change in code or push a code we get an email notification of the following. In the below Image I have done changes is a file name in my Github repository. which result in sending an email notification to mail.



GITHUB Code changes <no-reply@sns.amazonaws.com>

to me

2:03 PM (4 hours ago)

```
{"account":"585768150044","detailType":"CodePipeline Pipeline Execution State Change","region":"ap-south-1","source":"aws.codepipeline","time":"2025-07-21T08:33:44Z","n
otificationRuleArn":"arn:aws:codestar-notifications:ap-south-1:585768150044:notificationrule/c57e813cd97256053cd14b9af7472bdb7569785f","detail":{"pipeline":"DeployToCloudFormationS
ervice","execution-id":"d0d5536e-1247-44a7-8d44-9dc778db43ad","start-time":"2025-07-21T08:32:29.756Z","execution-trigger":{"author-display-name":"Om-Ghag","full-repository-name":"Om-
Ghag18/Event-Driven-Data-Processing-Pipeline","provider-type":"GitHub","author-email":"[redacted]","branch-name":"main","commit-message":"Rename CloudFormation
Template to template.yaml","author-date":"2025-07-21T08:32:26Z","commit-id":"a3daa2d9b19120cfc971534a3a99d12d5bbd8b48","connection-arn":"arn:aws:codeconnections:ap-south-
1:585768150044:connection/12b58d95-5b9f-4d07-959f-46bb72c6a1c5","author-id":"Om-Ghag"},"state":"FAILED","version":1.0,"pipeline-execution-attempt":1.0},"resources":["
arn:aws:codepipeline:ap-south-1:585768150044:DeployToCloudFormationService"],"additionalAttributes":{}}
```

Problem statement :-

Managing Permissions and Access Control

Issue:

Configuring the correct permissions for Lambda, S3, CodePipeline, and CloudFormation was challenging. Overly broad or missing IAM policies could cause failures — for example, Lambda couldn't read from S3 or write reports, or CodePipeline couldn't deploy updates.

Solution:

Carefully designed and assigned IAM roles with the least-privilege principle. Explicitly granted Lambda access to S3 buckets and SNS, CloudFormation permissions to manage stack resources, and CodePipeline trust relationships.

No Notification on Failures or Deployments

Issue:
Initially, there were no alerts when Lambda processing failed, or when a CodePipeline deployment succeeded/failed, leading to delays in troubleshooting.

Solution:
Configured SNS topics for Lambda and CodePipeline to push email notifications on job completion or error events. This enabled rapid awareness and response to pipeline status.

Debugging Lambda Failures

Issue:
When Lambda failed (due to code bugs, unhandled file formats, or timeouts).

Solution:
Enabled detailed CloudWatch logs for Lambda, wrote error-handling in function code. Regularly reviewed logs and set up alerts for repeated failures.

Ensuring Reliable Scheduling and Event Triggers

Issue:
Event Bridge schedules or S3 triggers sometimes didn’t fire (often due to misconfiguration or missing permissions), causing missed processing runs.

Solution:
Validated all event sources and target configurations. Used the AWS console to test triggers, and monitored Event Bridge invocation logs to confirm regular operation.