

## Serverless CI/CD Project for Contact Form

### Project Description:

We are about to create a serverless CI/CD process for a **serverless contact form**. So whenever a user hits the website, the user can able to give their details and after clicking on submit the page will return response as “*Thank you*” . After completing the project we will do integration with **some AWS services** and achieve CI/CD for to enhance better automation and reduce repeated process and time consumption for developers.

### Prerequisites:

For this project you need to have the following files

- /

- website/

- index.html

For customer details

- style.css

For the style of index file

- script.js

For s3 website hosting code logic

- lambda/

- index.js

Lambda code to create aws resources and project logic

- package.json

Dependencies of node modules

**Serverless.yaml** --→ for to create lambda, api gateway, dynamodb [after serverless deploy serverless will create a .serverless folder and within that it will create two files --→ **cloudformation-createtack.json**, **cloudformation-updatestack.json**] {you need to explain those two files }

## Procedure

- Create an EC2 instance and integrate it with vs code and aws configure with IAM user

## Install

- ✓ **git**
- ✓ **aws cli**
- ✓ **sudo yum install nodejs**
- ✓ **sudo npm install -g serverless**

In the EC2 create a folder

- **mkdir serverlesscid**

**(WE USE SAM in aws and WE USE SERVERLESS FRAMEWORK (3<sup>RD</sup> Party tool - To provision the services)**

- **vi serverless.yaml -**

```
service: serverlesscid
```

```
frameworkVersion: '3'
```

```
provider:
```

```
  name: aws
```

```
  runtime: nodejs18.x
```

```
# you can overwrite defaults here
```

```
stage: prod
```

```
region: eu-west-1
```

```
functions:
```

```
  lambda:
```

```
    handler: index.handler ( OUR LAMDA CODE – WHAT GOING TO EXECUTE )
```

```
    event ( WE ARE GOING TO CREATE REST API )
```

```
      - http: ( REST API)
```

```
path: /submit
method: POST ( 2 METHODS - GET( Get lambda invoke) AND POST )
```

➤ create index.js (This is the lambda code )

➤ **vi Index.js**

```
const AWS = require('aws-sdk');
const dynamodb = new AWS.DynamoDB.DocumentClient();
const { v4: uuidv4 } = require('uuid');

exports.handler = async (event) => {
  try {
    console.log('Raw input data:', event); // Add this line to log the raw
    input data

    const formData = {
      name: event.name,
      email: event.email,
      subject: event.subject,
      message: event.message,
    };

    const item = {
      SubmissionId: generateUUID(), // Generate a UUID
      ...formData, // Use the form data as attributes
    };

    // Store the form data in DynamoDB
    await storeFormData(item);

    return {
      statusCode: 200,
      body: JSON.stringify({ message: 'Form submitted successfully and this is
serverless cicd v1' }),
    };
  } catch (error) {
    console.error(error);
    return {
      statusCode: 500,
      body: JSON.stringify({ message: 'Error submitting the form' }),
    };
  }
}
```

```
};

// here we will provide the dynamo db table name
async function storeFormData(item) {
  const params = {
    TableName: 'ContactFormEntries',
    Item: item,
  };

  await dynamodb.put(params).promise();
}

function generateUUID() {
  return uuidv4();
}
```

- Run **npm init** --- > it will create a package.json – We are initiate the node js ( we use this file to deploy – we command deploy ,it searches the package.json and deploy it , suppose it is not there it fails )

```
[ec2-user@ip-172-31-22-233 serverlesscicd]$ npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.
```

See `npm help init` for definitive documentation on these fields and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and save it as a dependency in the package.json file.

```
Press ^C at any time to quit.
package name: (serverlesscicd)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to /home/ec2-user/serverlesscicd/package.json:
```

```
{
  "name": "serverlesscicd",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}
```

```
Is this OK? (yes) yes
```

```
[ec2-user@ip-172-31-22-233 serverlesscicd]$ ll
```

```
total 4
```

```
-rw-r--r--. 1 ec2-user ec2-user 210 Nov 15 18:46 package.json
```

```
[ec2-user@ip-172-31-22-233 serverlesscicd]$
```

---

- As it is a serverless proj we require **aws dependencies** , I've added this one line of code by seeing the error in the lambda console , add one more line in the package.json


## AWS dependencies

1. use module to use other services
- 2.

```

{
  "name": "serverlesscicd",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC",
  "dependencies": {
    "aws-sdk": "^2.1386.0"
  }
}

```



- Here the index.js is the main lambda code to be executed in simple it is the name of the code , if you are changing the code or you want to run another code means you need to change the handler name for example in the **serverless.yaml** you need to provide the handler the code name here if the code name is testservice then you need to give as testservice.handler in the serverless.yaml
- **Run --- > npm install --- >** it will create a node modules folder and package-lock.json will be created (For what it is created? =
- **Run -- > serverless deploy**
- **By default It** will create a lambda , rest api gateway based on the service+stage name+ function name – in the serverlessyaml file and also api gateway stage name plus service name
- If you again run the deploy command with the change in stage and lambda name it will create new lambda and api gateway
- The serverless deploy command will package all the code in to the lambda and it will create a .serverless folder within that two cloud formation files will be there , cloudformation-create-stack.json

- `cloudformation-update-stack.json`, by default the serverless deploy command will upload our `index.js` code to s3 by as it creates two cloud formation files one to create s3 bucket and value of the output , that bucket will be imported in another update stack , and on this bucket only our `index.js` code and all our code will be uploaded , on the template you can find the s3 key , in that location the code will be there

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Description": "The AWS CloudFormation template for this Serverless application",
  "Resources": {
    "ServerlessDeploymentBucket": {
      "Type": "AWS::S3::Bucket",
      "Properties": {
        "BucketEncryption": {
          "ServerSideEncryptionConfiguration": [
            {
              "ServerSideEncryptionByDefault": {
                "SSEAlgorithm": "AES256"
              }
            }
          ]
        }
      }
    },
    "ServerlessDeploymentBucketPolicy": {
      "Type": "AWS::S3::BucketPolicy",
      "Properties": {
        "Bucket": {
          "Ref": "ServerlessDeploymentBucket"
        },
        "PolicyDocument": {
          "Statement": [
            {
              "Action": "s3:*",
              "Effect": "Deny",
              "Principal": "*"
            }
          ]
        }
      }
    }
  }
}
```

Creating s3 bucket

Creating s3 bucket policy





In *serverless.yaml*

```
13
14
15 functions:
16   cird:
17     handler: index.handler
18     events:
19       - http:
20         path: /submit
21         method: POST
22
```

- Even though we are giving http it will create rest api only , that is the latest update if you give rest instead of http , still it would rest api gateway but it will be older version of the rest api.
- So know about how to create http

### Create events in lambda

```
{
  "name": "test",
  "email": "abd@example.com",
  "subject": "This is you",
  "message": "A message from you"
}
```

### Test event action

☒ Create new event

### Event name

test

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

### Event sharing settings

☒ Private

This event is only available in the Lambda console and to the event creator. You can configure a total of 10. [Learn more](#)

☐ Shareable

This event is available to IAM users within the same account who have permissions to access and use shareable events. [Learn more](#)

### Template - optional

hello-world

### Event JSON

```
1 {  
2   "name": "test",  
3   "email": "abd@example.com",  
4   "subject": "This is you",  
5   "message": "A message from you"  
6 }  
7
```

And click on **save**

- Goto dynamo db and create a table as given in the same name in index.js

[DynamoDB](#) > [Tables](#) > Create table

## Create table

**Table details** [Info](#)

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

**Table name**  
This will be used to identify your table.

Between 3 and 255 characters, containing only letters, numbers, underscores (\_), hyphens (-), and periods (.).

**Partition key**  
The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

1 to 255 characters and case sensitive.

**Sort key - optional**  
You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

1 to 255 characters and case sensitive.

- Edit the existing lambda role to have access to dynamo db , if you didn't attach the role then the permission error will be visible in cloud watch logs
- Under *configuration -- > permission --- > execution role -- > add permission -- > attach policy --- > dynamo db full access/ required access.*

Code Test Monitor **Configuration** Aliases Versions

General configuration  
Triggers  
**Permissions**  
Destinations  
Function URL  
Environment variables  
Tags

**Execution role** [Refresh](#) [Edit](#) [View role document](#)

Role name

**Resource summary**

To view the resources and actions that your function has permission to access, choose a service.

Amazon CloudWatch Logs  
4 actions, 2 resources

➤ *Now run the test event, you should see*

☑ The test event test was successfully saved.

Code **Test** Monitor Configuration Aliases Versions

☑ Executing function: succeeded ([logs](#))

▼ Details

The area below shows the last 4 KB of the execution log.

```
{
  "statusCode": 200,
  "body": "{\"message\":\"Form submitted successfully and this is serverless ci3d3\"}"
}
```

Summary

Code SHA-256 1KAz9hbXSvm7X3YJZLCxfoBQiZ4GS3ALr8CdrbQUZ0=	Execution time 23 seconds ago (November 16, 2023 at 01:26 AM GMT+5:30)
Request ID daa8a108-a717-47b7-82fd-9b0b5794927d	Function version \$LATEST
Init duration 712.52 ms	Duration 86.25 ms
Billed duration 87 ms	Resources configured 1024 MB
Max memory used 93 MB	

➤ *In the dynamodb you should see the content*

CONTACTFORMENTRIES2 Autopreview View table details

▼ Scan or query items

☒ Scan ☐ Query

Select a table or index: Table - ContactFormEntries2

Select attribute projection: All attributes

► Filters

Run Reset

☑ Completed. Read capacity units consumed: 0.5

Items returned (1) Refresh Actions Create item

< 1 > Settings Fullscreen

<input type="checkbox"/>	SubmissionId (String)	email	message	name	subject
<input type="checkbox"/>	<a href="#">7dacbd76-c4da-4c2d-90e...</a>	abd@exam...	A message ...	test	This is you

- Go to api gateway
- Click on Post--- > Integration request

Amazon API Gateway

APIs > prod-serverlessci (b1og0opyv2) > Resources > /submit (78zykh)

APIs

Custom Domain Names

VPC Links

API: prod-serverless...

Resources

Stages

Authorizers

Gateway Responses

Models

Resource Policy

Documentation

Dashboard

Settings

Usage Plans

The new API Gateway console experience is now available  
We've redesigned the API Gateway console for REST APIs and WebSocket APIs. [Try out the new console](#). The old console experience will no longer be available starting late November 2023.

Resources Actions

/submit

POST

/submit Methods

POST

Authorization None

API Key Not required

The new API Gateway console experience is now available  
We've redesigned the API Gateway console for REST APIs and WebSocket APIs. [Try out the new console](#). The old console experience will no longer be available starting late November 2023.

Resources Actions

/submit - POST - Method Execution

TEST

Client

Method Request

Auth: NONE

ARN: arn:aws:execute-api:eu-west-1:756168232951:b1og0opyv2/"POST/submit

Integration Request

Type: LAMBDA\_PROXY

Integration Response

Proxy integrations cannot be configured to transform responses.

Lambda serverlessci-prod-lambda

## ➤ *Uncheck the lambda proxy integration*

Resources

Actions

/

/submit

POST

Method Execution

/submit - POST - Integration Request

Provide information about the target backend that this method will call and whether the incoming request data should be modified.

Integration type

☒ Lambda Function

☐ HTTP

☐ Mock

☐ AWS Service

☐ VPC Link

Use Lambda Proxy integration

☒

Lambda Region

eu-west-1

Lambda Function

serverlesscicd-prod-lambda

Execution role

Invoke with caller credentials

☐


Credentials cache


Do not add caller credentials to cache key


Use Default Timeout


☒


➤ *Scroll down and click on mapping template and verify like configured below*

**Lambda Function** serverlesscicd-prod-lambda 

**Execution role** 

**Invoke with caller credentials** ☐ 

**Credentials cache** Do not add caller credentials to cache key 

**Use Default Timeout** ☒ 

▶ URL Path Parameters

---

▶ URL Query String Parameters


---


▶ HTTP Headers


---

▼ Mapping Templates

---


**Request body passthrough** ☒ When no template matches the request Content-Type header 

☐ When there are no templates defined (recommended) 

☐ Never 

Content-Type
No mapping templates defined. The request body will be passed through to the integration endpoint

---

 [Add mapping template](#)

- Create an *s3 bucket* for website hosting and make the ***bucket public***

### General configuration

Bucket name

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming.](#)

AWS Region

Europe (Ireland) eu-west-1 ▼

Copy settings from existing bucket - *optional*  
Only the bucket settings in the following configuration are copied.

Choose bucket

### 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.

Object Ownership

Bucket owner enforced



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](#)

☐ **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.



**Turning off block all public access might result in this bucket and the objects within becoming public**

AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☒ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Bucket Versioning

[Amazon S3](#) > [Buckets](#) > serverlesscicddabd

serverlesscicddabd [Info](#)

Publicly accessible

Objects

Properties

**Permissions**

Metrics

Management

Access Points

### Permissions overview

Access

Public

by using policy generator



➤ Actions -- > enable cors

➤ On that Allow access origin paste the s3 static website url

Resources Actions ▾ Enable CORS

Gateway Responses for prod-serverlesscicdd API ☐ DEFAULT 4XX ☐ DEFAULT 5XX ⓘ

Methods ☒ POST ☐ OPTIONS ⓘ

Access-Control-Allow-Methods OPTIONS, POST ⓘ

Access-Control-Allow-Headers 'Content-Type,X-Amz-Date,Authorization' ⓘ

Access-Control-Allow-Origin 'http://serverlesscicddabds3-website-ei' ⓘ

Advanced

Enable CORS and replace existing CORS header

➤ It would throw a error

Resources Actions ▾ Enable CORS

✓ Add Access-Control-Allow-Headers, Access-Control-Allow-Methods, Access-Control-Allow-Origin Method Response Headers to OPTIONS method

✓ Add Access-Control-Allow-Headers, Access-Control-Allow-Methods, Access-Control-Allow-Origin Integration Response Header Mappings to OPTIONS method

✗ Add Access-Control-Allow-Origin Method Response Header to POST method ⚠

✗ Add Access-Control-Allow-Origin Integration Response Header Mapping to POST method ⚠

➤ Now click on options--- > integration request

The new API Gateway console experience is now available  
We've redesigned the API Gateway console for REST APIs and WebSocket APIs. Try out the new console. The old console experience will no longer be available starting late November 2023.

Resources Actions ▾ /submit - OPTIONS - Method Execution

Client

Method Request

Auth: NONE

ARN: arn:aws:execute-api:eu-west-1:756168232951:b1og0opyv2/"OPTIONS/su

Integration Request

Type: MOCK

Method Response

HTTP Status: 200

Models: application/json => Empty

Integration Response

HTTP status pattern: 200

Output passthrough: No

Mock Endpoint

➤ *Configure like below*

Resources Actions ▾

Method Execution /submit - OPTIONS - Integration Request

Provide information about the target backend that this method will call and whether the incoming request data should be modified.

Integration type ☒ Lambda Function ⓘ  
☐ HTTP ⓘ  
☐ Mock ⓘ  
☐ AWS Service ⓘ  
☐ VPC Link ⓘ

Use Lambda Proxy integration ☐ ⓘ

Lambda Region eu-west-1 ▾

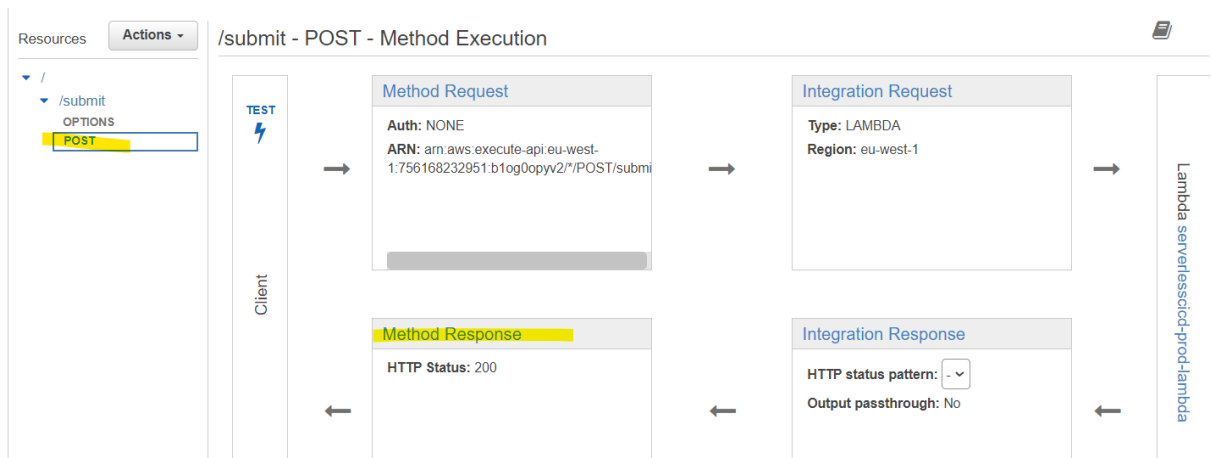
Lambda Function serverlesscicd-prod-lambda ⓘ

Use Default Timeout ☒ ⓘ

Save

Mapping Templates

➤ *Click on POST---- > method response*



➤ *Type 200*

Resources Actions ▾

Method Execution /submit - POST - Method Response

Provide information about this method's response types, their headers and content types.

HTTP Status

200

➤ *Now again enable CORS*

Pls > prod-serverlessciid (b1og0opyv2) > Resources > /submit (7ôzykh) > Enable CORS Show hints ?

**The new API Gateway console experience is now available**  
We've redesigned the API Gateway console for REST APIs and WebSocket APIs. Try out the new console. The old console experience will no longer be available starting late November 2023.

Resources Actions ▾ Enable CORS

▼ /  
▼ /submit  
OPTIONS  
POST

- ✓ Add Access-Control-Allow-Headers, Access-Control-Allow-Methods, Access-Control-Allow-Origin Method Response Headers to OPTIONS method
- ✓ Add Access-Control-Allow-Headers, Access-Control-Allow-Methods, Access-Control-Allow-Origin Integration Response Header Mappings to OPTIONS method
- ✓ Add Access-Control-Allow-Origin Method Response Header to POST method
- ✓ Add Access-Control-Allow-Origin Integration Response Header Mapping to POST method

➤ *Now **Deploy API** under prod stage and copy the api url*

➤ ***Goto resources -- > test***

➤ *Under request body*

### Stage Variables

No [stage variables](#) exist for this method.

### Request Body

```
1 {  
2   "name": "abdul",  
3   "email": "abd@gmail.com",  
4   "subject": "2ndsub",  
5   "message": "2ndmessage"  
6 }  
7
```

✓ You should see the below response

Resources

Actions

Method Execution

/submit - POST - Method Test

Make a test call to your method. When you make a test call, API Gateway skips authorization and directly invokes your method

Path

No path parameters exist for this resource. You can define path parameters by using the syntax **(myPathParam)** in a resource path.

Query Strings

{submit}

Headers

{submit}

Stage Variables

Request: /submit

Status: 200

Latency: 1226 ms

Response Body

```
{ "statusCode": 200, "body": "{\\\"message\\\": \\\"Form submitted successfully and this is serverless ci3d3\\\"}" }
```

Response Headers

```
{ "Access-Control-Allow-Origin": [ "http://serverlessci3d3.s3-website-eu-west-1.amazonaws.com" ], "Content-Type": [ "application/json" ], "X-Amzn-Trace-Id": [ "Root=1-6555b07a-3a00b3528b99603462c40ee4;Sampled=0;lineage=088b8c28:0" ] }
```

Logs

Execution log for request 6a75a1b6-42d2-4a3a-a366-22a281d9ec57

Thu Nov 16 06:02:34 UTC 2023 : Starting execution for request: 6a75a1

✓ In dynamo db you should see the below response

ContactFormEntries2

Select a table or index

Table - ContactFormEntries2

Select attribute projection

All attributes

Filters

Run

Reset

Completed. Read capacity units consumed: 0.5

Items returned (2)

Actions

Create item

	SubmissionId (String)	email	message	name	subject
<input type="checkbox"/>	17692f59-df33-4c0c-872f...	abd@gmail...	2ndmessage	abdul	2ndsub
<input type="checkbox"/>	7dacbd76-c4da-4c2d-90e...	abd@exam...	A message ...	test	This is you

➤ In the script.js give the api gateway url

```
//  
  
function submitForm(formData) {  
  // Make an API request to the backend (API Gateway) for form submission  
  fetch('https://blog0opyv2.execute-api.eu-west-1.amazonaws.com/prod/submit', {  
    method: 'POST',  
    headers: {  
      'Content-Type': 'application/json'  
    },  
  
    body: JSON.stringify(formData)  
  })  
  .then(function(response) {  
    if (response.ok) {  
      // Redirect to the thank you page  
      window.location.href = 'thank-you.html';  
    } else {  
      throw new Error('Form submission failed.');    }  
  })  
  .catch(function(error) {  
    console.error(error);  
    alert('Form submission failed. Please try again later.');  });  
}
```

➤ *Go to s3 and upload the below files*

Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

**Files and folders** (4 Total, 3.1 KB)

Remove

Add files

Add folder

All files and folders in this table will be uploaded.

Find by name

< 1 >

<input type="checkbox"/>	Name	Folder	Type	Size
<input type="checkbox"/>	script.js	-	text/javascript	1.6 KB
<input type="checkbox"/>	thank-you.html	-	text/html	271.0 B
<input type="checkbox"/>	style.css	-	text/css	466.0 B
<input type="checkbox"/>	index.html	-	text/html	786.0 B

**Destination**

Destination

s3://serverlesscicddabd

► Destination details

Bucket settings that impact new objects stored in the specified destination.

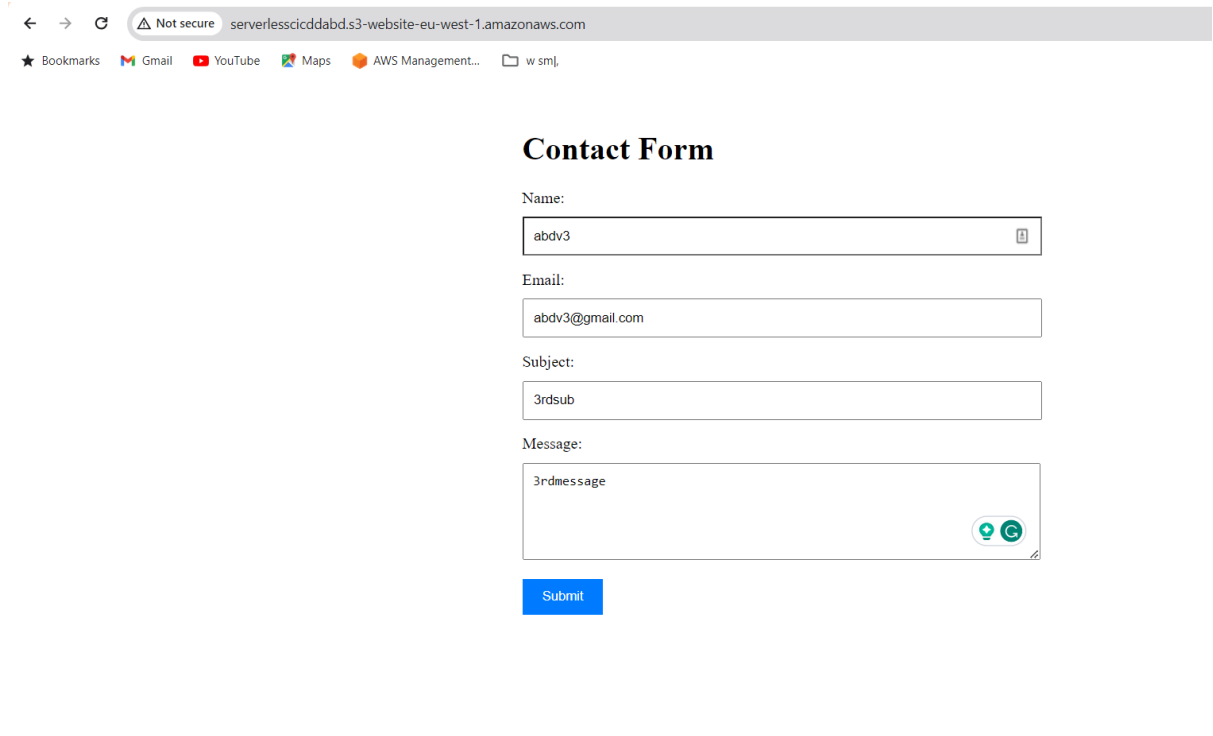
➤ *In s3 paste the following cors under cors configuration*

```
[
  {
    "AllowedHeaders": [
      "*"
    ],
    "AllowedMethods": [
      "POST"
    ],
    "AllowedOrigins": [
      "your-static-website-url"
    ],
    "ExposeHeaders": []
  }
]
```



➤ Hit the S3 website url under properties

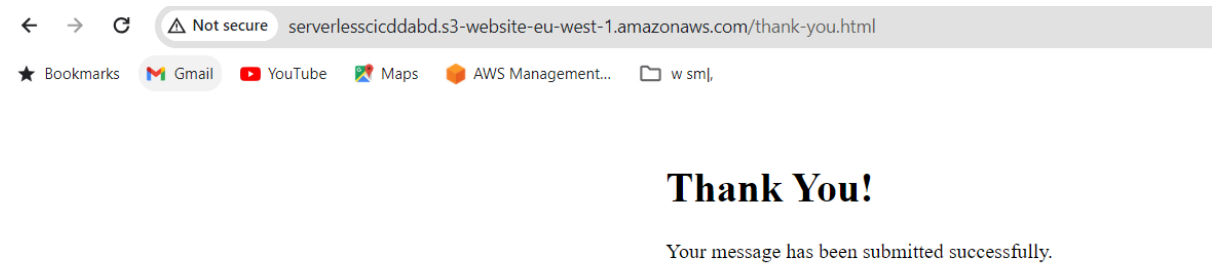
➤ *You should see the below success response*



The screenshot shows a web browser window with the address bar displaying "Not secure serverlesscicddabd.s3-website-eu-west-1.amazonaws.com". The browser's bookmark bar includes "Bookmarks", "Gmail", "YouTube", "Maps", "AWS Management...", and "w smj,". The main content area features a "Contact Form" with the following fields:

- Name:** A text input field containing "abdv3".
- Email:** A text input field containing "abdv3@gmail.com".
- Subject:** A text input field containing "3rdsub".
- Message:** A large text area containing "3rdmessage".

Below the message field is a blue "Submit" button. A small green circular icon with a white plus sign is visible in the bottom right corner of the message field.



The screenshot shows a web browser window with the address bar displaying "Not secure serverlesscicddabd.s3-website-eu-west-1.amazonaws.com/thank-you.html". The browser's bookmark bar includes "Bookmarks", "Gmail", "YouTube", "Maps", "AWS Management...", and "w smj,". The main content area displays a "Thank You!" message:

**Thank You!**

Your message has been submitted successfully.

✓ *In dynamo db you should see the below response*

The screenshot shows the AWS DynamoDB console interface. On the left, the table 'ContactFormEntries2' is selected. The main area displays a successful query result with a green status bar indicating 'Completed. Read capacity units consumed: 0.5'. Below this, a table titled 'Items returned (4)' shows four items. The last item is highlighted with a red checkmark.

SubmissionId (String)	email	message	name	subject
17692f59-df33-4c0c-872f...	abd@gmail...	2ndmessage	abdul	2ndsub
e49aac84-fee3-4468-b40...				
7dacbd76-c4da-4c2d-90e...	abd@exam...	A message ...	test	This is you
5505988e-37ea-4432-87a...	abdv3@gm...	3rdmessage	abdv3	3rdsub

✓ Thus we have integrated *LAMBDA -- > API GATEWAY -- > DYNAMODB -- > S3*

- Now we are about to initiate CI/CD
- Initially we need to integrate our working repo with github
- Create a new repository in **git hub**

The screenshot shows the GitHub Dashboard. At the top, there are links for Bookmarks, Gmail, YouTube, and Maps. Below the navigation bar, the 'Top Repositories' section is visible, featuring a search bar and a list of repositories. A green 'New' button is also present.

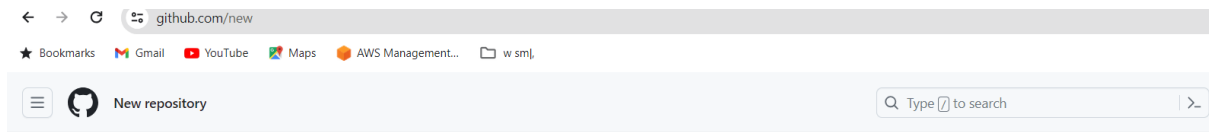
★ Bookmarks   Gmail   YouTube   Maps

Dashboard

Top Repositories New

Find a repository...

- abdulrahman0108/serverlessci
- abdulrahman0108/testdemo



- cd serverlesiccd/
- ll
- Initiate Git --- > **git init** --- in the working repository
- git add\*
- git config username and email
- git commit -m "first commit"
- **git remote add abd**  
<https://github.com/abdurahman0108/serverlesiccd.git>
  
- run -- > **ssh-keygen -t ed25519 -C your\_email@example.com**
  
- **Press two enter**
- cd
- cd .ssh
- ll
- it displays the all keys
- cat id\_pub – copy the key
- Run -- > **eval "\$(ssh-agent -s)"**

```

git@github.com: Permission denied (publickey).
[ec2-user@ip-172-31-22-233 serverlesscicd]$ ssh-keygen -t ed25519 -C "abdul.ib96@gmail.com"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ec2-user/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ec2-user/.ssh/id_ed25519
Your public key has been saved in /home/ec2-user/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:23WPaLDqA7AkGt0DlXNFgWM4MEW1BvM9fz9nD9+kpRo abdul.ib96@gmail.com
The key's randomart image is:
+--[ED25519 256]--+
| o=*+.++.      |
| ..+=+o        |
| . o =+o.      |
| . o =.   o    |
| o o + S o o . |
| . . . o = + o |
| .. o E = *    |
| .. . . @o    |
| .o. .o +    |
+----[SHA256]-----+
[ec2-user@ip-172-31-22-233 serverlesscicd]$ eval "$(ssh-agent -s)"
Agent pid 32495
[ec2-user@ip-172-31-22-233 serverlesscicd]$ ssh-add ~/.ssh/id_ed25519
Identity added: /home/ec2-user/.ssh/id_ed25519 (abdul.ib96@gmail.com)
[ec2-user@ip-172-31-22-233 serverlesscicd]$ cd
[ec2-user@ip-172-31-22-233 ~]$ cd .ssh/
[ec2-user@ip-172-31-22-233 .ssh]$ ll
total 16
-rw-----. 1 ec2-user ec2-user 385 Nov 12 19:18 authorized_keys
-rw-----. 1 ec2-user ec2-user 411 Nov 16 10:25 id_ed25519
-rw-r--r--. 1 ec2-user ec2-user 102 Nov 16 10:25 id_ed25519.pub
-rw-r--r--. 1 ec2-user ec2-user 92 Nov 16 10:24 known_hosts
[ec2-user@ip-172-31-22-233 .ssh]$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIM0SLbtj2qoUS/j1lBUuE2xH/1P2JiNrQo9M3QxwgQuc abdul.ib96@gmail.com
[ec2-user@ip-172-31-22-233 .ssh]$ ^C
[ec2-user@ip-172-31-22-233 .ssh]$ ssh -v git@github.com
OpenSSH_8.7p1, OpenSSL 3.0.8 7 Feb 2023
debug1: Reading configuration data /etc/ssh/ssh_config

```



**ssh-add ~/.ssh/id\_ed25519**

**Go to Git hub and add the public key in the ssh keys and add it**

**ssh -v [git@github.com](https://github.com)**

**hi 7598273777 – it shows**

```

total 44
-rw-r--r--. 1 ec2-user ec2-user 1170 Nov 15 19:01 index.js
drwxr-xr-x. 37 ec2-user ec2-user 16384 Nov 15 19:00 node_modules
-rw-r--r--. 1 ec2-user ec2-user 14158 Nov 15 19:00 package-lock.json
-rw-r--r--. 1 ec2-user ec2-user 270 Nov 15 18:51 package.json
-rw-r--r--. 1 ec2-user ec2-user 287 Nov 15 19:08 serverless.yaml
[ec2-user@ip-172-31-22-233 serverlessci]$ git push -u abd master
Username for 'https://github.com': abdulrahman0108
Password for 'https://abdulrahman0108@github.com':
remote: Support for password authentication was removed on August 13, 2021.
remote: Please see https://docs.github.com/en/get-started/getting-started-
tication.
fatal: Authentication failed for 'https://github.com/abdulrahman0108/serve
[ec2-user@ip-172-31-22-233 serverlessci]$ ^C
[ec2-user@ip-172-31-22-233 serverlessci]$ git push -u abd master
Username for 'https://github.com': abdulrahman0108
Password for 'https://abdulrahman0108@github.com':
Enumerating objects: 2283, done.
Counting objects: 100% (2283/2283), done.
Compressing objects: 100% (2231/2231), done.
Writing objects: 100% (2283/2283), 11.39 MiB | 3.14 MiB/s, done.
Total 2283 (delta 1178), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1178/1178), done.
To https://github.com/abdulrahman0108/serverlessci.git
 * [new branch]      master -> master
branch 'master' set up to track 'abd/master'.
[ec2-user@ip-172-31-22-233 serverlessci]$

```

This creates a new SSH key, using the provided email as a label.

```
> Generating public/private ALGORITHM key pair.
```

When you're prompted to "Enter a file in which to save the key", you can press **Enter** to accept the default file location. Please note that if you created SSH keys previously, ssh-keygen may ask you to rewrite another key, in which case we recommend creating a custom-named SSH key. To do so, type the default file location and replace id\_ALGORITHM with your custom key name.

```
> Enter a file in which to save the key (/home/YOU/.ssh/ALGORITHM):[Press enter]
```

- 3 At the prompt, type a secure passphrase. For more information, see "[Working with SSH key phrases](#)."

```
> Enter passphrase (empty for no passphrase): [Type a passphrase]
> Enter same passphrase again: [Type passphrase again]
```

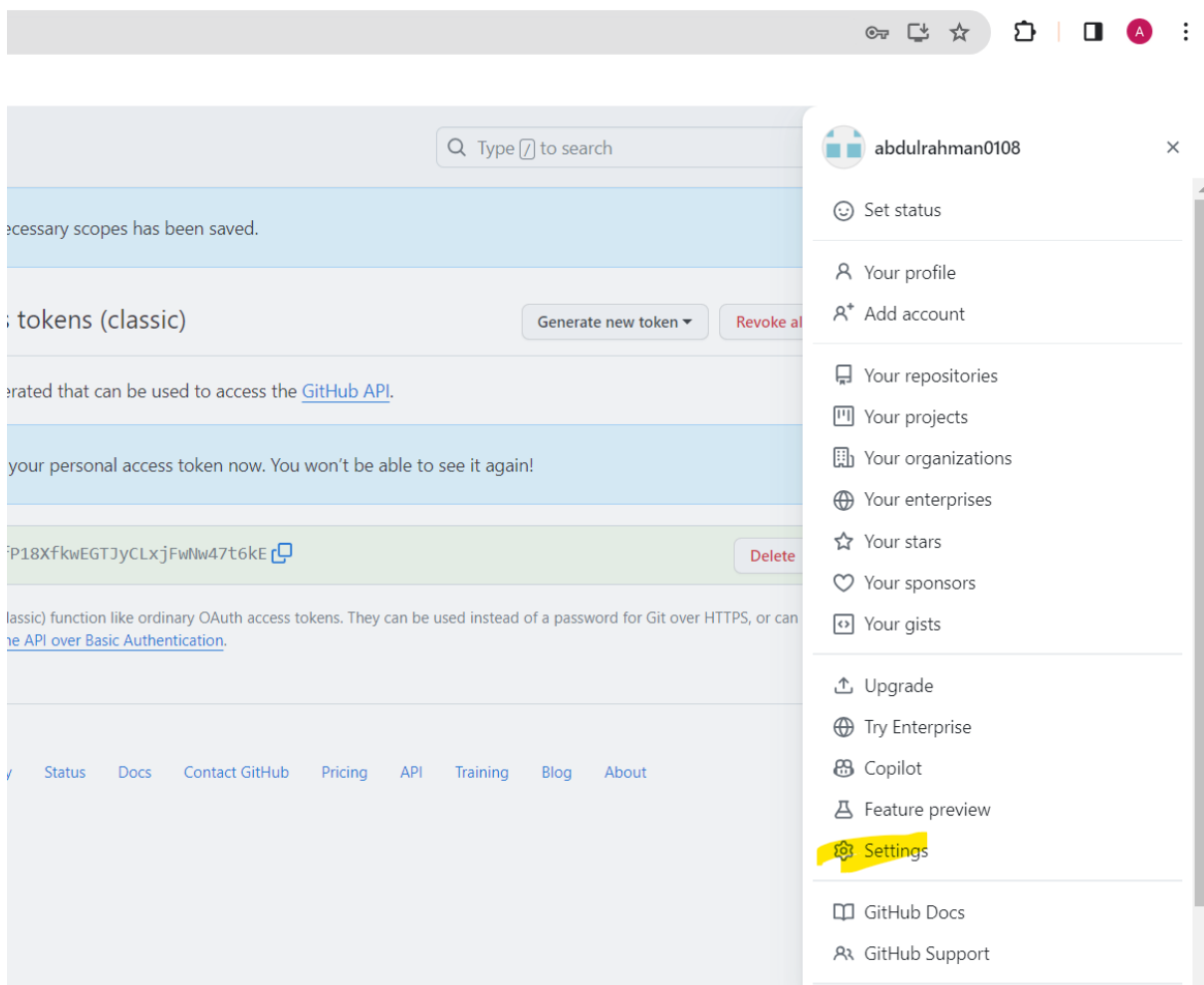
➤ *for more clarity on above github steps follow the url*

[ <https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent?platform=linux> ]

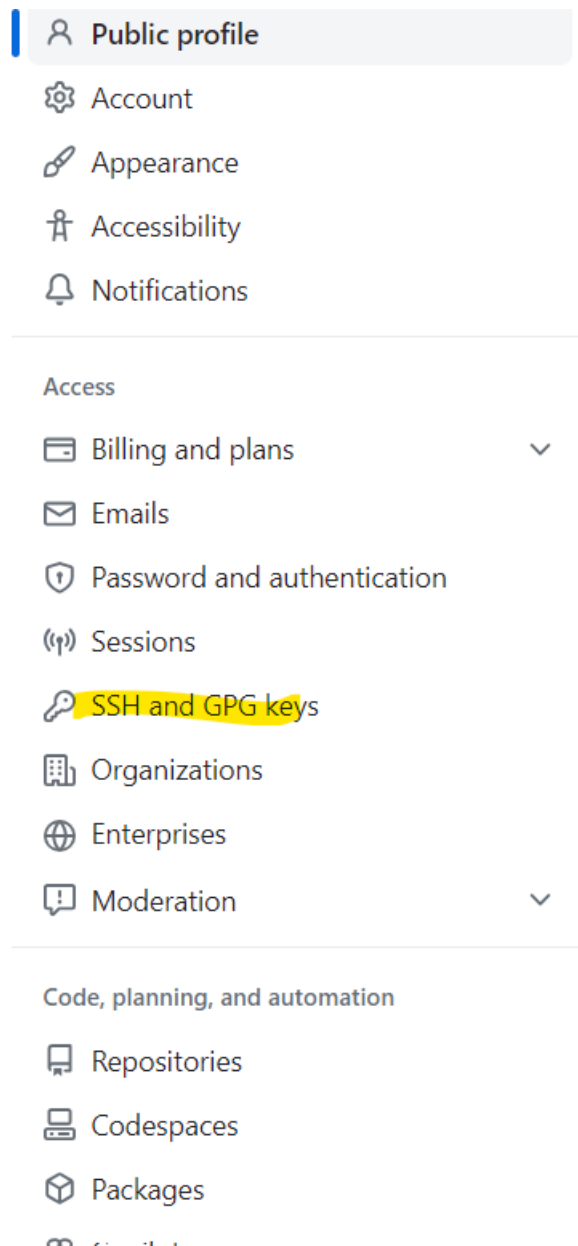
<https://docs.github.com/en/authentication/connecting-to-github-with-ssh/adding-a-new-ssh-key-to-your-github-account> ]

➤ Now we have created the ssh public key, now we need to add the public key in github

➤ ***Goto settings in github***




➤ ***Click on add ssh and gpg keys***



➤ ***Copy the public key from .ssh/***

➤ ***Run -- > `cat id_ed25519.pub`***

➤ ***Copy the ssh and paste on the github ssh key***



**abdulrahman0108** (abdulrahman0108)

Your personal account

[Go to your personal profile](#)

- Public profile
- Account
- Appearance
- Accessibility
- Notifications


Access

- Billing and plans
- Emails
- Password and authentication

## SSH keys

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

### Authentication Keys



**serverlesscid**

SHA256: 23WPaLDqA7AkGt0D1XNFgWM4MEW1BvM9fz9nD9+KpRo

Added on Nov 16, 2023

Never used — Read/write


[Delete](#)

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH problems](#).


Now run -- > ***ssh -v git@github.com*** ,, you should see

```
debug1: Will attempt key: /home/ec2-user/.ssh/id_rsa
debug1: SSH2_MSG_EXT_INFO received
debug1: kex_input_ext_info: server-sig-algs=<ssh-ed25519-cert-v01@openssh.com,ecdsa-sha2-nistp521-cert-v01@openssh.com,ecdsa-sha2-nistp384-cert-v01@openssh.com,sk-ssh-ed25519-cert-v01@openssh.com,sk-ecdsa-sha2-nistp256-cert-v01@openssh.com,rsa-sha2-512-cert-v01@openssh.com,rsa-sha2-256-cert-v01@openssh.com,sk-ssh-ed25519@openssh.com,sk-ecdsa-sha2-nistp256@openssh.com,ssh-ed25519,ecdsa-sha2-nistp521,ecdsa-sha2-nistp384,ecdsa-sha2-nistp256,rsa-sha2-512,rsa-sha2-256>
debug1: SSH2_MSG_SERVICE_ACCEPT received
debug1: Authentications that can continue: publickey
debug1: Next authentication method: publickey
debug1: Offering public key: /home/ec2-user/.ssh/id_ed25519 ED25519 SHA256:23WPaLDqA7AkGt0D1XNFgWM4MEW1BvM9fz9nD9+KpRo agent
debug1: Server accepts key: /home/ec2-user/.ssh/id_ed25519 ED25519 SHA256:23WPaLDqA7AkGt0D1XNFgWM4MEW1BvM9fz9nD9+KpRo agent
Authenticated to github.com ([140.82.121.4]:22) using "publickey".
debug1: pkcs11_del_provider: called, provider_id = (null)
debug1: channel 0: new [client-session]
debug1: Entering interactive session.
debug1: pledge: filesystem full
debug1: client_input_global_request: rtype hostkeys-00@openssh.com want_reply 0
debug1: client_input_hostkeys: searching /home/ec2-user/.ssh/known_hosts for github.com / (none)
debug1: client_input_hostkeys: searching /home/ec2-user/.ssh/known_hosts2 for github.com / (none)
debug1: client_input_hostkeys: hostkeys file /home/ec2-user/.ssh/known_hosts2 does not exist
Learned new hostkey: RSA SHA256:uNiVztkCsDhcc0u9e8BujQXVUpKZIDTMczCvj3tD2s
Learned new hostkey: ECDSA SHA256:p2QAMXNIC1TJYWeIottrVc98/R1BUFWu3/LiyKgUfQM
Adding new key for github.com to /home/ec2-user/.ssh/known_hosts: ssh-rsa SHA256:uNiVztkCsDhcc0u9e8BujQXVUpKZIDTMczCvj3tD2s
Adding new key for github.com to /home/ec2-user/.ssh/known_hosts: ecdsa-sha2-nistp256 SHA256:p2QAMXNIC1TJYWeIottrVc98/R1BUFWu3/LiyKgUfQM
debug1: update_known_hosts: known_hosts file /home/ec2-user/.ssh/known_hosts2 does not exist
PTY allocation request failed on channel 0
Hi abdulrahman0108! You've successfully authenticated, but GitHub does not provide shell access.
debug1: client_input_channel_req: channel 0 rtype exit-status reply 0
debug1: channel 0: free: client-session, nchannels 1
Connection to github.com closed.
Transferred: sent 2928, received 3228 bytes, in 0.2 seconds
Bytes per second: sent 12356.2, received 13622.2
debug1: Exit status 1
[ec2-user@ip-172-31-22-233 .ssh]$ ll
total 20
-rw-r--r-- 1 ec2-user ec2-user 205 Nov 16 10:10 authorized_keys
```

➤ Again go to github clic on ***settings -- > developer settings-- > generate classic tokens***



Settings / Developer Settings



**GitHub Apps**

[OAuth Apps](#)

[Personal access tokens](#)

[Fine-grained tokens](#)

[Tokens \(classic\)](#)

**GitHub Apps**

[New GitHub App](#)

Want to build something that integrates with and extends GitHub? [Register a new GitHub App](#) to get started developing on the GitHub API. You can also read more about building GitHub Apps in our [developer documentation](#).

➤ Give all the access and generate token and set expiration as **never expire** and copy the token id



<input type="checkbox"/> write:org	Read and write org and team membership, read and write org projects
<input type="checkbox"/> read:org	Read org and team membership, read org projects
<input type="checkbox"/> manage_runners:org	Manage org runners and runner groups
<input checked="" type="checkbox"/> admin:public_key	Full control of user public keys
<input type="checkbox"/> write:public_key	Write user public keys
<input type="checkbox"/> read:public_key	Read user public keys
<input checked="" type="checkbox"/> admin:repo_hook	Full control of repository hooks
<input type="checkbox"/> write:repo_hook	Write repository hooks
<input type="checkbox"/> read:repo_hook	Read repository hooks
<input checked="" type="checkbox"/> admin:org_hook	Full control of organization hooks
<input checked="" type="checkbox"/> gist	Create gists
<input checked="" type="checkbox"/> notifications	Access notifications
<input checked="" type="checkbox"/> user	Update ALL user data
<input type="checkbox"/> read:user	Read ALL user profile data
<input type="checkbox"/> user:email	Access user email addresses (read-only)
<input type="checkbox"/> user:follow	Follow and unfollow users
<input checked="" type="checkbox"/> delete_repo	Delete repositories
<input checked="" type="checkbox"/> write:discussion	Read and write team discussions
<input type="checkbox"/> read:discussion	Read team discussions
<input checked="" type="checkbox"/> admin:enterprise	Full control of enterprises
<input type="checkbox"/> manage_runners:enterprise	Manage enterprise runners and runner groups
<input type="checkbox"/> manage_billing:enterprise	Read and write enterprise billing data
<input type="checkbox"/> read:enterprise	Read enterprise profile data
<input checked="" type="checkbox"/> audit_log	Full control of audit log
<input type="checkbox"/> read:audit_log	Read access of audit log
<input checked="" type="checkbox"/> codespace	Full control of codespaces
<input type="checkbox"/> codespace:secrets	Ability to create, read, update, and delete codespace secrets
<input checked="" type="checkbox"/> copilot	Full control of GitHub Copilot settings and seat assignments
<input type="checkbox"/> manage_billing:copilot	View and edit Copilot for Business seat assignments
<input checked="" type="checkbox"/> project	Full control of projects
<input type="checkbox"/> read:project	Read access of projects
<input checked="" type="checkbox"/> admin:gpg_key	Full control of public user GPG keys
<input type="checkbox"/> write:gpg_key	Write public user GPG keys
<input type="checkbox"/> read:gpg_key	Read public user GPG keys
<input checked="" type="checkbox"/> admin:ssh_signing_key	Full control of public user SSH signing keys
<input type="checkbox"/> write:ssh_signing_key	Write public user SSH signing keys
<input type="checkbox"/> read:ssh_signing_key	Read public user SSH signing keys

- Run -- > ***git push -u jegan master***
- username : jegan
- Password : Token of git

**Given the commands to push to github for reference**

**git init**

**35 git add \***

**36 git config --global user.name "abduhrahman0108"**

**37 git config --global user.mail "abduh.ib96@gmail.com"**

**38 git commit -m "1st commit"**

**39 ll**

**40 git remote add abd**

**<https://github.com/abduhrahman0108/serverlessiccd.git>**

**44 ssh -v git@github.com**

**45 ssh-keygen -t ed25519 -C "abduh.ib96@gmail.com"**

**46 eval "\$(ssh-agent -s)"**

**47 ssh-add ~/.ssh/id\_ed25519**

**49 cd .ssh/**

**51 cat id\_ed25519.pub**

**52 ssh -v git@github.com**

**54 ssh-keygen -E md5 -lf ~/.ssh/id\_ed25519.pub**

**56 cd serverlessiccd/**

**58 git push -u abd master**

✓ *You should be able to push from local to github*

```
[ec2-user@ip-172-31-22-233 serverlessiccd]$ git push -u abd master
Username for 'https://github.com': abdulrahman0108
Password for 'https://abdulrahman0108@github.com':
Enumerating objects: 2283, done.
Counting objects: 100% (2283/2283), done.
Compressing objects: 100% (2231/2231), done.
Writing objects: 100% (2283/2283), 11.39 MiB | 3.14 MiB/s, done.
Total 2283 (delta 1178), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1178/1178), done.
To https://github.com/abdulrahman0108/serverlessiccd.git
 * [new branch]      master -> master
branch 'master' set up to track 'abd/master'.
[ec2-user@ip-172-31-22-233 serverlessiccd]$ history
1  11
```

➤ *Now go to code build*


➤ *Build the project*

[Developer Tools](#) > [CodeBuild](#) > [Build projects](#) > Create build project

## Create build project

### Project configuration

Project name



A project name must be 2 to 255 characters. It can include the letters A-Z and a-z, the numbers 0-9, and the special characters - and \_.

► **Additional configuration**  
Description, Build badge, Concurrent build limit, tags

### Source

Add source

#### Source 1 - Primary

Source provider

GitHub ▼

Repository

AWS CodeBuild needs access to your GitHub account to display available repositories.

➤ *Save Token*

## Source 1 - Primary

Source provider

GitHub

### Repository

AWS CodeBuild needs access to your GitHub account to display available repositories.

☐ Connect using OAuth

☒ Connect with a GitHub personal access token

GitHub personal access token

ghp\_jlpF20EYPfP18XfkwEGTJyCLxjFwNw47t6kE

Save token

Source version - *optional* [Info](#)

Enter a pull request, branch, commit ID, tag, or reference and a commit ID.

### ► Additional configuration

Git clone depth, Git submodules

## Environment

➤ ***Give the repository url***

**Source 1 - Primary**

Source provider

GitHub ▼

Repository

☒ Public repository

☐ Repository in my GitHub account

Repository URL

https://github.com/abduhrahman0108/serverlesiccd.git

https://github.com/<user-name>/<repository-name>

Connection status

You are connected to GitHub using a personal access token.

Disconnect from GitHub

Source version - optional [Info](#)

Enter a pull request, branch, commit ID, tag, or reference and a commit ID.

► **Additional configuration**

Git clone depth, Git submodules

➤ ***Attach the existing role or create the role with the following permissions***

#### Service role

☐ New service role  
Create a service role in your account

☒ Existing service role  
Choose an existing service role from your account

#### Role ARN

Q arn:aws:iam::756168232951:role/service-role/codebuild-slscicdv3-service-role X

☒ Allow AWS CodeBuild to modify this service role so it can be used with this build project

#### ► Additional configuration

Timeout, certificate, VPC, compute type, environment variables, file systems

### Buildspec

#### Build specifications

☒ Use a buildspec file  
Store build commands in a YAML-formatted buildspec file

☐ Insert build commands  
Store build commands as build project configuration

#### Buildspec name - *optional*

By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or

[Docker images provided by CodeBuild - AWS CodeBuild \(amazon.com\)](#)

➤ ***Configure the environment like below***

#### Environment image



##### Managed image

Use an image managed by AWS CodeBuild



##### Custom image

Specify a Docker image

#### Compute



##### EC2

Optimized for flexibility during action runs



##### Lambda

Optimized for speed and minimizes the start up time of workflow actions

#### Operating system

Amazon Linux



#### Runtime(s)

Standard



#### Image

aws/codebuild/amazonlinux2-x86\_64-standard:5.0



#### Image version

Always use the latest image for this runtime version



☐ Use GPU-enhanced compute

#### Privileged

☐ Enable this flag if you want to build Docker images or want your builds to get elevated privileges

- ***Cause the code build to avoid permission issues attach the following permission***

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

Permissions policies (5) [Info](#)

↻

Simulate [↗](#)

Remove

Add permissions ▼

You can attach up to 10 managed policies.

Search

Filter by Type

All types ▼

< 1 > ⚙

<input type="checkbox"/>	Policy name <a href="#">↗</a>	Type	Attached entities
<input type="checkbox"/>	<a href="#">AdministratorAccess</a>	AWS managed - job function	<a href="#">3</a>
<input type="checkbox"/>	<a href="#">CodeBuildBasePolicy-serverlesscid-eu-c...</a>	Customer managed	<a href="#">1</a>
<input type="checkbox"/>	<a href="#">CodeBuildBasePolicy-slscicdv3-eu-west-1</a>	Customer managed	<a href="#">1</a>
<input type="checkbox"/>	<a href="#">CodeBuildCloudWatchLogsPolicy-serverl...</a>	Customer managed	<a href="#">1</a>
<input type="checkbox"/>	<a href="#">CodeBuildCloudWatchLogsPolicy-slscicd...</a>	Customer managed	<a href="#">1</a>

- ***Give the **`buildspec.yaml`** name***



## Buildspec

### Build specifications



Use a buildspec file

Store build commands in a YAML-formatted buildspec file



Insert build commands

Store build commands as build project configuration

### Buildspec name - *optional*

By default, CodeBuild looks for a file named buildspec.yml in the source code root directory. If your buildspec file uses a different name or location, enter its path from the source root here (for example, buildspec-two.yml or configuration/buildspec.yml).

## Batch configuration

You can run a group of builds as a single execution. Batch configuration is also available in advanced option when starting build.



Define batch configuration - *optional*

You can also define or override batch configuration when starting a build batch.

## Artifacts

[Add artifact](#)

### Artifact 1 - Primary

Type



You might choose no artifacts if you are running tests or pushing a Docker image to Amazon ECR.

➤ *Under additional configuration*

## Edit Environment

### Environment

Current environment image

aws/codebuild/amazonlinux2-x86\_64-standard:5.0

Override image

Service role

Choose an existing service role from your account

Q arn:aws:iam::756168232951:role/service-role/codebuild-slscicdv3-service-role X

☒ Allow AWS CodeBuild to modify this service role so it can be used with this build project

### ▼ Additional configuration

Timeout, certificate, VPC, compute type, environment variables, file systems

Timeout

Default timeout is 1 hour

Hours

1

Minutes

0

Timeout must be between 5 minutes and 8 hours

Queued timeout

➤ *Set the stage variable as given below*

### Compute

- ☒ 3 GB memory, 2 vCPUs
- ☐ 7 GB memory, 4 vCPUs
- ☐ 15 GB memory, 8 vCPUs
- ☐ 145 GB memory, 72 vCPUs

### Environment variables

Name	Value	Type	
STAGE_NAME	prod	Plaintext ▼	Remove

Add environment variable

Create parameter

### File systems

Identifier	ID	
	Q	Remove

Directory path - optional

Mount point

Mount options - optional

Add file system

➤ *Export the logs to cloudwatch if require s3 also and create build project*

### Logs

#### CloudWatch

- ☒ CloudWatch logs - optional  
Checking this option will upload build output logs to CloudWatch.

Group name

serverless

Stream name

cicd

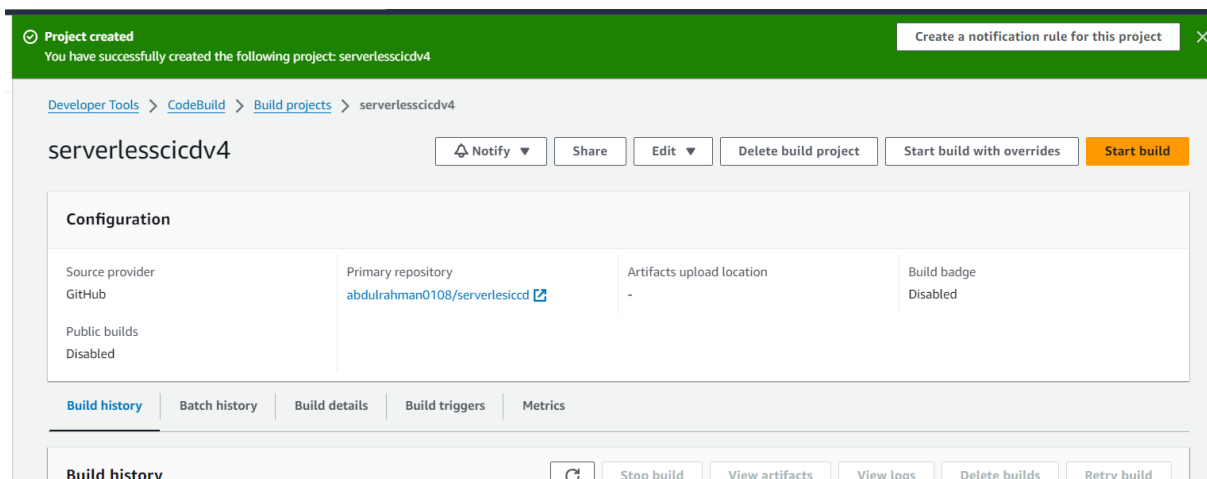
#### S3

- ☐ S3 logs - optional  
Checking this option will upload build output logs to S3.

Cancel

Create build project

➤ *Start the build and check for the logs*



- *The build will be expected to be failure cause we haven't push the **buildspec.yaml***
- *Create **buildspec.yaml** in the working repository inside the project folder*

version: 0.2

run-as: root

phases:

install:

runtime-versions:

nodejs: 18

commands:

- npm install -g serverless
- npm install

build:

commands:

- serverless deploy --stage \${STAGE\_NAME}

cache:

paths:

- node\_modules

- *Push the **buildspec.yaml** to github*

```
[ec2-user@ip-172-31-22-233 serverlesscicd]$ git push -u abd master
Username for 'https://github.com': abdulrahman0108
Password for 'https://abdulrahman0108@github.com':
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 459 bytes | 459.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/abdulrahman0108/serverlesscicd.git
6c60b7e..0235237 master -> master
branch 'master' set up to track 'abd/master'.
[ec2-user@ip-172-31-22-233 serverlesscicd]$
```

## Retry the build

## If error – Give the Roles full Admin Access

Build started
You have successfully started the following build: serverlesscicdv4:f4d94102-4817-4cc9-8660-0895b7ac76ab

Developer Tools > CodeBuild > Build projects > serverlesscicdv4 > serverlesscicdv4:f4d94102-4817-4cc9-8660-0895b7ac76ab

serverlesscicdv4:f4d94102-4817-4cc9-8660-0895b7ac76ab

Stop build

Retry build

Build status

Status	Initiator	Build ARN	Resolved source version
Failed	root	arn:aws:codebuild:eu-west-1:756168232951:build/serverlesscicdv4:f4d94102-4817-4cc9-8660-0895b7ac76ab	-
Start time	End time	Build number	

- **You should the build succeeded state along with the api gateway url**
- **Build gives the POST – url and it cant be browsed**

Build logs

Succeeded

Start time: 1 minute ago

Current phase: COMPLETED

```

39 found 0 vulnerabilities
40
41 [Container] 2023/11/16 11:41:09.870594 Phase complete: INSTALL State: SUCCEEDED
42 [Container] 2023/11/16 11:41:09.870609 Phase context status code: Message:
43 [Container] 2023/11/16 11:41:09.899371 Entering phase PRE_BUILD
44 [Container] 2023/11/16 11:41:09.900948 Phase complete: PRE_BUILD State: SUCCEEDED
45 [Container] 2023/11/16 11:41:09.900962 Phase context status code: Message:
46 [Container] 2023/11/16 11:41:09.928428 Entering phase BUILD
47 [Container] 2023/11/16 11:41:09.929041 Running command serverless deploy --stage ${STAGE_NAME}
48
49 Deploying serverlesscicd to stage prod (eu-west-1)
50
51 ✓ Service deployed to stack serverlesscicd-prod (30s)
52
53 endpoint: POST - https://blog0qqyv2.execute-api.eu-west-1.amazonaws.com/prod/submit
54 functions:
55   lambda: serverlesscicd-prod-lambda (16 MB)
56
57 [Container] 2023/11/16 11:41:45.724345 Phase complete: BUILD State: SUCCEEDED
58 [Container] 2023/11/16 11:41:45.724372 Phase context status code: Message:
59 [Container] 2023/11/16 11:41:45.752395 Entering phase POST_BUILD
60 [Container] 2023/11/16 11:41:45.753960 Phase complete: POST_BUILD State: SUCCEEDED
61 [Container] 2023/11/16 11:41:45.753975 Phase context status code: Message:
62

```

- **Now integrate the project with code pipeline**
- **Create the code pipeline like below**

[Developer Tools](#) > [CodePipeline](#) > [Pipelines](#) > Create new pipeline

Step 1

**Choose pipeline settings**

Step 2

Add source stage

Step 3

Add build stage

Step 4

Add deploy stage

Step 5

Review

## Choose pipeline settings Info

Step 1 of 5

### Pipeline settings

**Pipeline name**  
Enter the pipeline name. You cannot edit the pipeline name after it is created.

No more than 100 characters

**Pipeline type**  
The pipeline type determines the pipeline structure and availability of parameters such as triggers. Pipeline type selection will impact features and pricing. [Which pipeline is right for me?](#)

☐ V1
 ☒ V2

**Service role**

☒ **New service role**  
Create a service role in your account
 ☐ **Existing service role**  
Choose an existing service role from your account

**Role name**

Type your service role name

☒ Allow AWS CodePipeline to create a service role so it can be used with this new pipeline

➤ **Choose the source provider as github v1 and connect to github**

### Source

**Source provider**  
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

Grant AWS CodePipeline access to your GitHub repository. This allows AWS CodePipeline to upload commits from GitHub to your pipeline.

Connecting

**The GitHub (Version 1) action is not recommended**

The selected action uses OAuth apps to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub (Version 2) action to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources. [Learn more](#)

**Change detection options**  
Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

☐ **GitHub webhooks (recommended)**  
Use webhooks in GitHub to automatically start my pipeline when a change occurs

☒ **AWS CodePipeline**  
Use AWS CodePipeline to check periodically for changes

Cancel

Previous

Next

Correction : Give webhook not aws codepipeline

➤ **Give a connection name**

- *Click authorize aws-codesuite*
- **It will prompt to authorize – click on authorize**

Developer Tools > CodePipeline > Pipelines > Create new pipeline

Step 1  
Choose pipeline settings

Step 2  
**Add source stage**

Step 3  
Add build stage

Step 4  
Add deploy stage

Step 5  
Review

## Add source stage [Info](#)

Step 2 of 5

### Source


**Source provider**  
This is where you stored your input artifacts for your pipeline. Choose the provider and then provide the connection details.

GitHub (Version 1) ▼

Grant AWS CodePipeline access to your GitHub repository. This allows AWS CodePipeline to upload commits from GitHub to your pipeline.

Connected

✔ You have successfully configured the action with the provider. ✕


**The GitHub (Version 1) action is not recommended**  
The selected action uses OAuth apps to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub (Version 2) action to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources. [Learn more](#)

Repository

○

➤ **Choose your repository with master branch**

✔ You have successfully configured the action with the provider. ✕

 **The GitHub (Version 1) action is not recommended**  
The selected action uses OAuth apps to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub (Version 2) action to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources. [Learn more](#)

Repository


abdu <span>l</span> rahman0108/25082020
abdu <span>l</span> rahman0108/git---1608
abdu <span>l</span> rahman0108/git-1408
abdu <span>l</span> rahman0108/java-web-app-docker
abdu <span>l</span> rahman0108/javaproject
abdu <span>l</span> rahman0108/mavenproject
abdu <span>l</span> rahman0108/new1
abdu <span>l</span> rahman0108/new2
<b>abdu<span>l</span>rahman0108/serverlessicd</b>
abdu <span>l</span> rahman0108/serverlessicd
abdu <span>l</span> rahman0108/simplewebproject
abdu <span>l</span> rahman0108/testdemo

check periodically for changes

Previous

Next



**The GitHub (Version 1) action is not recommended**

The selected action uses OAuth apps to access your GitHub repository. This is no longer the recommended method. Instead, choose the GitHub (Version 2) action to access your repository by creating a connection. Connections use GitHub Apps to manage authentication and can be shared with other resources. [Learn more](#)

Repository

Branch

Change detection options  
Choose a detection mode to automatically start your pipeline when a change occurs in the source code.

☒ **GitHub webhooks (recommended)**  
Use webhooks in GitHub to automatically start my pipeline when a change occurs

☐ **AWS CodePipeline**  
Use AWS CodePipeline to check periodically for changes

Cancel

Previous

Next

➤ ***Give the build provider name***

### Build - optional

#### Build provider

This is the tool of your build project. Provide build artifact details like operating system, build spec file, and output file names.

AWS CodeBuild

#### Region

Europe (Ireland)

#### Project name

Choose a build project that you have already created in the AWS CodeBuild console. Or create a build project in the AWS CodeBuild console and then return to this task.

serverlesscicdv4



or

Create project

#### Environment variables - optional

Choose the key, value, and type for your CodeBuild environment variables. In the value field, you can reference variables generated by CodePipeline. [Learn more](#)

Add environment variable

#### Build type



Single build

Triggers a single build.



Batch build

Triggers multiple builds as a single execution.

Cancel

Previous

Skip build stage

Next

➤ ***As we have given the deploy in the buildspec yaml itself we can skip the integration with code deploy***

## Add deploy stage [Info](#)

Step 4 of 5

### Deploy - optional

#### Deploy provider

Choose how you deploy to instances. Choose the provider, and then provide the configuration details for that provider.

Cancel

Previous

Skip deploy stage

Next

## ➤ **Create pipeline**

### Step 3: Add build stage

#### Build action provider

Build action provider

AWS CodeBuild

ProjectName

serverlesscicdv4

### Step 4: Add deploy stage

#### Deploy action provider

Deployment stage

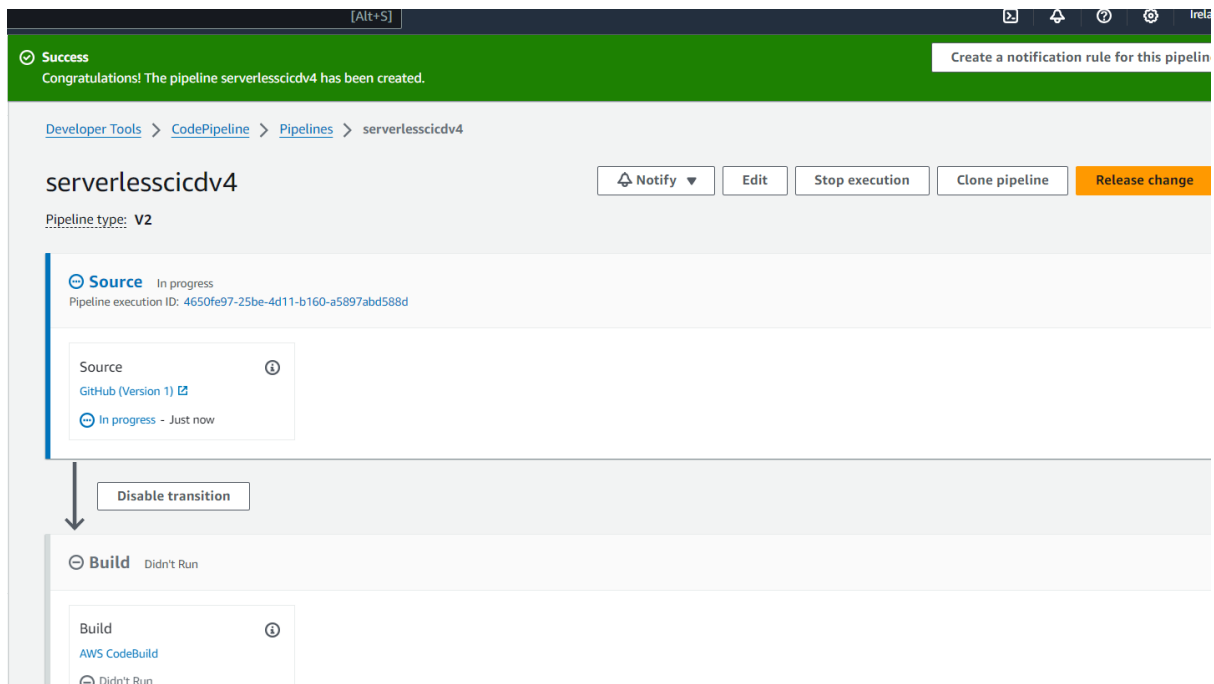
No deploy

Cancel

Previous

Create pipeline

➤ **After creating the pipeline , the pipeline will run a dry build.**



- *Thus we have successfully integrated the project with lambda, api gateway, dynamodb,s3, github,codebuild, code pipeline*
- *Open the index.js in the local repository and edit the lambda code*

```

try {
  console.log('Raw input data:', event); // Add this line to log the raw input data

  const formData = {
    name: event.name,
    email: event.email,
    subject: event.subject,
    message: event.message,
  };

  const item = {
    SubmissionId: generateUUID(), // Generate a UUID
    ...formData, // Use the form data as attributes
  };

  // Store the form data in DynamoDB
  await storeFormData(item);

  return {
    statusCode: 200,
    body: JSON.stringify({ message: 'Form submitted successfully and this is serverless cicdv5' }),
  };
} catch (error) {
  console.error(error);
  return {
    statusCode: 500,
    body: JSON.stringify({ message: 'Error submitting the form' }),
  };
}
};
// here we will provide the dynamodb table name
async function storeFormData(item) {
  const params = {
    TableName: 'ContactFormEntries2',
    Item: item,
  };

  await dynamodb.put(params).promise();
}

function generateUUID() {
  return uuidv4();
}

```

### ➤ When you push the changes

```

[ec2-user@ip-172-31-22-233 serverlesscicd]$ git push -u abd master
Username for 'https://github.com': abdulrahman0108
Password for 'https://abdulrahman0108@github.com':
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 316 bytes | 316.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/abdulrahman0108/serverlessiccd.git
  0235237..b2ea13d  master -> master
branch 'master' set up to track 'abd/master'.
[ec2-user@ip-172-31-22-233 serverlesscicd]$

```

➤ The build will be started again

The screenshot displays the AWS CodePipeline console for a pipeline with execution ID 5b275cb8-ef5a-4bcf-bf6b-1a507bdacfe4. The **Source** stage, using **GitHub (Version 1)**, has completed successfully. Below it, the **Build** stage, using **AWS CodeBuild**, is currently in progress. A **Disable transition** button is visible between the stages. On the right, a vertical status bar shows a green checkmark for the Source stage and a blue circle for the Build stage. The commit ID b2ea13d6 is shown at the bottom of each stage.

**Source** Succeeded  
Pipeline execution ID: 5b275cb8-ef5a-4bcf-bf6b-1a507bdacfe4

Source  
GitHub (Version 1) [🔗](#)  
Succeeded - Just now  
[b2ea13d6](#) [🔗](#)

b2ea13d6 [🔗](#) Source: 3rd commit

Disable transition

**Build** In progress  
Pipeline execution ID: 5b275cb8-ef5a-4bcf-bf6b-1a507bdacfe4

Build  
AWS CodeBuild  
In progress - Just now  
[Details](#)  
[View logs](#)

b2ea13d6 [🔗](#) Source: 3rd commit

➤ ***Go to lambda and edit the saved test event***

This is a duplicate of the previous screenshot, showing the same AWS CodePipeline console view. The Source stage is successful, and the Build stage is in progress. The interface elements, including the stage details, transition button, and status bar, are identical to the first image.

**Source** Succeeded  
Pipeline execution ID: 5b275cb8-ef5a-4bcf-bf6b-1a507bdacfe4

Source  
GitHub (Version 1) [🔗](#)  
Succeeded - Just now  
[b2ea13d6](#) [🔗](#)

b2ea13d6 [🔗](#) Source: 3rd commit

Disable transition

**Build** In progress  
Pipeline execution ID: 5b275cb8-ef5a-4bcf-bf6b-1a507bdacfe4

Build  
AWS CodeBuild  
In progress - Just now  
[Details](#)  
[View logs](#)

b2ea13d6 [🔗](#) Source: 3rd commit

Code **Test** Monitor Configuration Aliases Versions

**Test event** [Info](#) Delete Save Test

To invoke your function without saving an event, modify the event, then choose Test. Lambda uses the modified event to invoke your function, but does not overwrite the original event until you choose Save changes.

Test event action

☐ Create new event ☒ Edit saved event

Event name

test ↻

**Event JSON** Format JSON


```

1 {
2   "name": "abdv6",
3   "email": "abdv6@example.com",
4   "subject": "v6sub",
5   "message": "v6message"
6 }

```

## ➤ **Test it**

➤ **Here the lambda response code is changed to v6**

 **Executing function: succeeded** ([logs](#))

▼ Details

The area below shows the last 4 KB of the execution log.

```

{
  "statusCode": 200,
  "body": "{\"message\":\"Form submitted successfully and this is serverless cicdv5\"}"
}

```

**Summary**

Code SHA-256	Execution time
kp/snbHN+mmDqIBP2hzVTFZVnaplKtX8rsbn3YoS/Hk=	2 seconds ago (November 16, 2023 at 05:52 PM GMT+5:30)
Request ID	Function version
c20d0ab4-e1c2-4eb1-a7a1-9cacb4f28046	\$LATEST
Init duration	Duration
682.85 ms	84.39 ms
Billed duration	Resources configured
85 ms	1024 MB
Max memory used	
93 MB	

**Log output**

✓ *Thus if any changes in github repository will identified by githook in the code pipeline and it trigger the code build to build the project from the github repository and deploy the changes in the code. **Thus we have achieved CI/CD integration.***

## Interview

Frame the questions with integrating with services

1. **ECS - deploy and usage , procedure**
2. **IAM Roles**
3. ***List the aws services u know***
4. ***say something about hobby and yourself***
5. ***How do you know about AWS and why do u choose it***
6. ***How to intregrate ECS and DynamoDb -***
7. ***Aws SDK***
8. ***What are services and Database U worked with list ? ---? ?***
9. ***How to install postgre in EC2 ( Browse)***
10. ***How to change website http to https***
11. ***400 – Client request error***
12. ***503 error – file not inside so check the file/permission/handler***
13. ***200 – success***