Serverless Labs

1. Building a Serverless Web Application

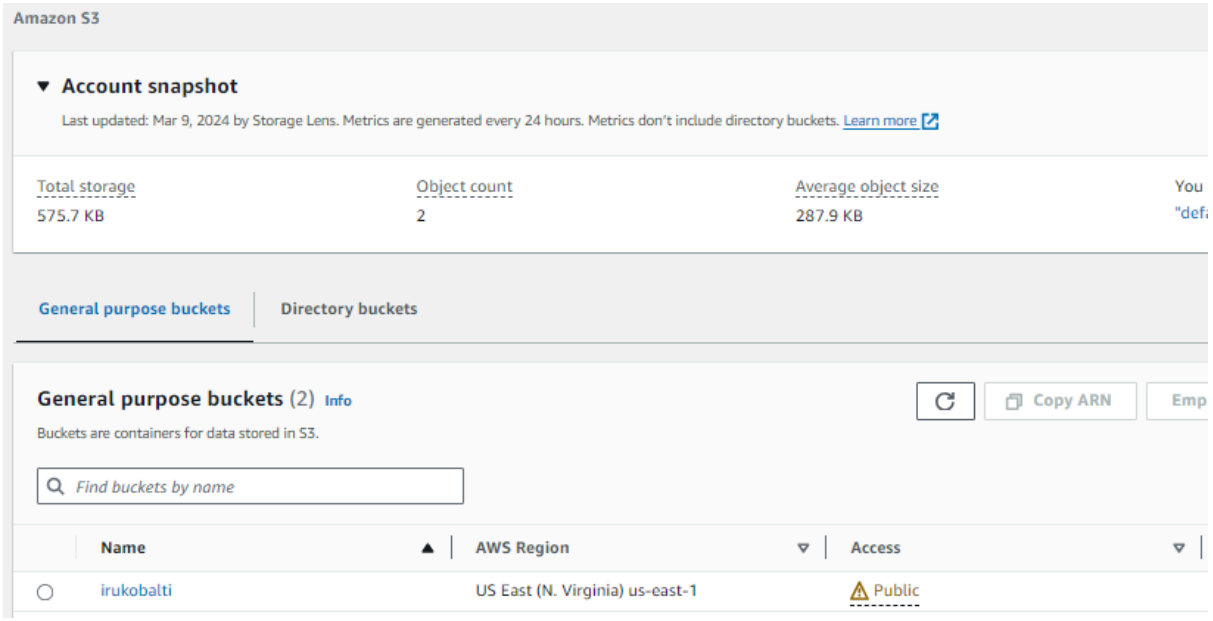
Objective: Create a serverless web application using AWS Lambda, API Gateway, S3, and DynamoDB.

Approach:

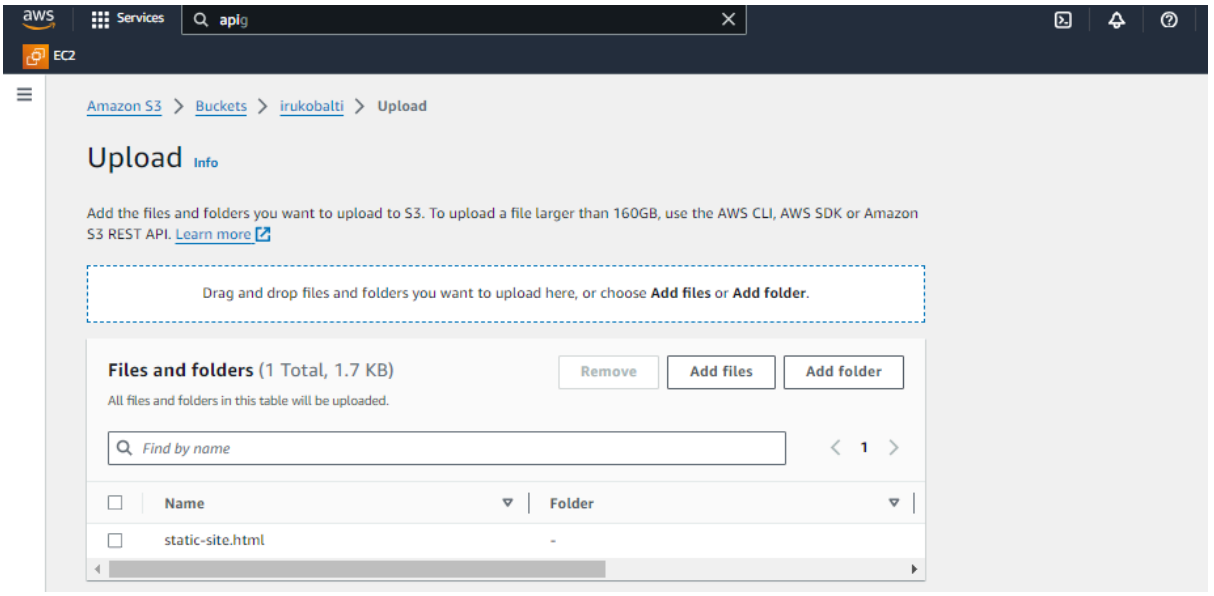
- **Set Up Backend**: Create Lambda functions to handle backend logic. These functions will interact with a DynamoDB table for data storage.
- **API Gateway**: Set up API Gateway to create RESTful endpoints that trigger the Lambda functions.
- **Frontend Hosting**: Host a static website on S3 that interacts with the backend via API Gateway.
- **Integration**: Ensure that the frontend can successfully send requests to the backend and display responses.

Goal: Understand the basics of building and connecting serverless backend services with a static frontend, enabling a fully serverless web application.

Start by creating an S3 bucket

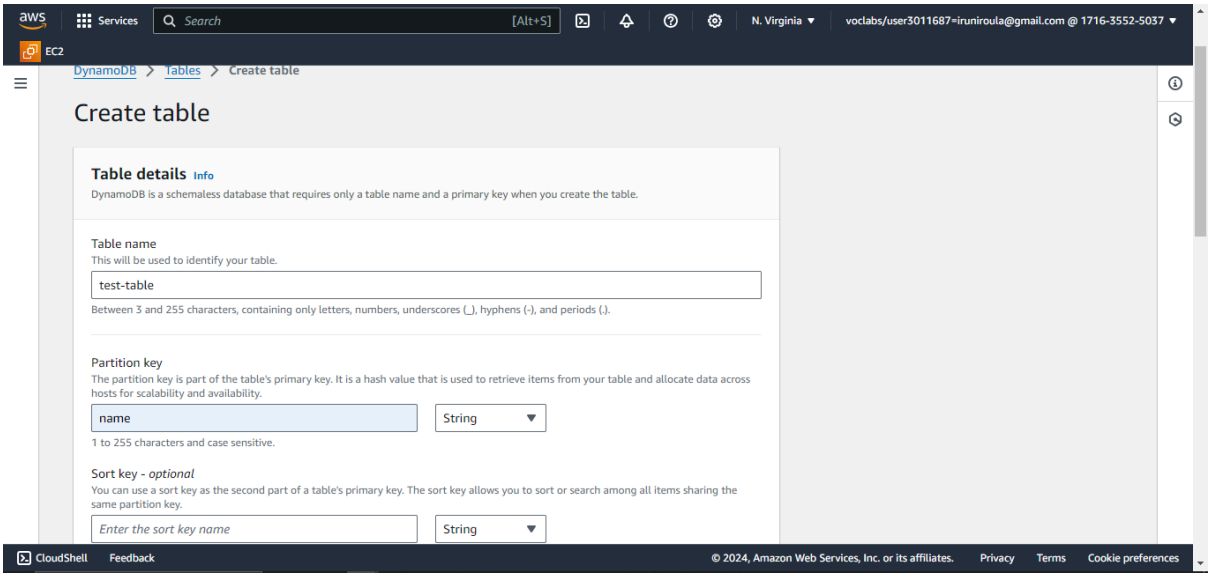


Upload static file created in the bucket

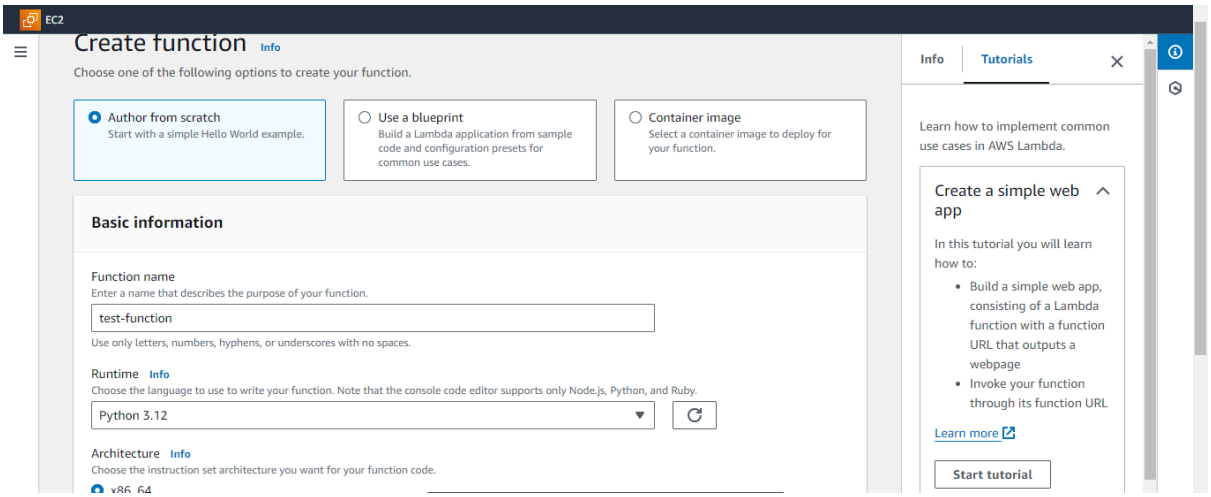


Our static site looks like this

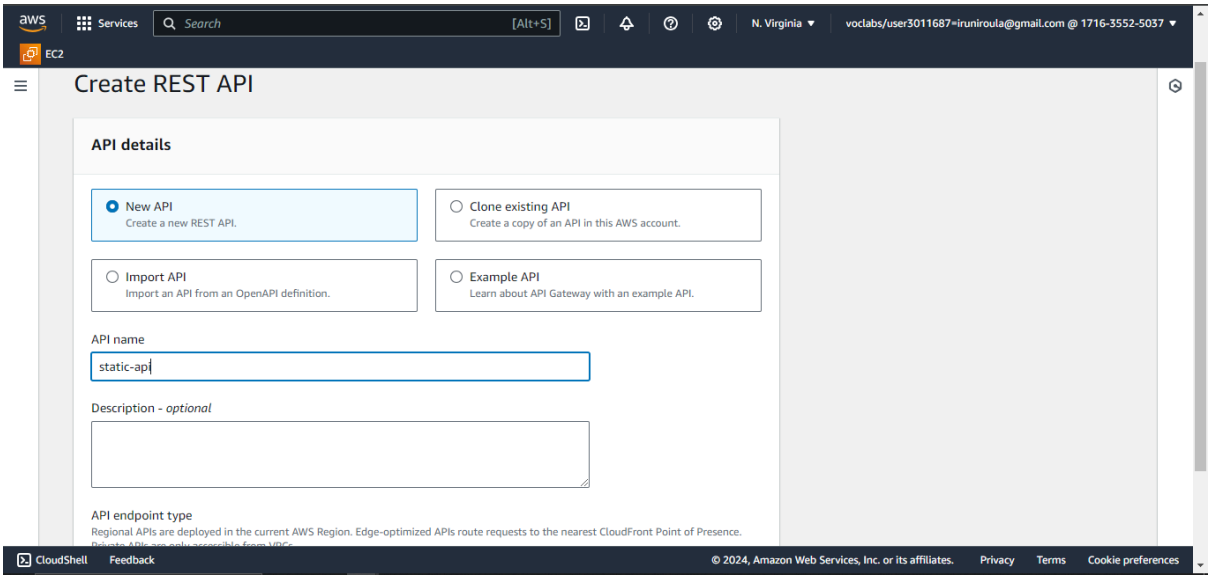
Now a DynamoDB table is created

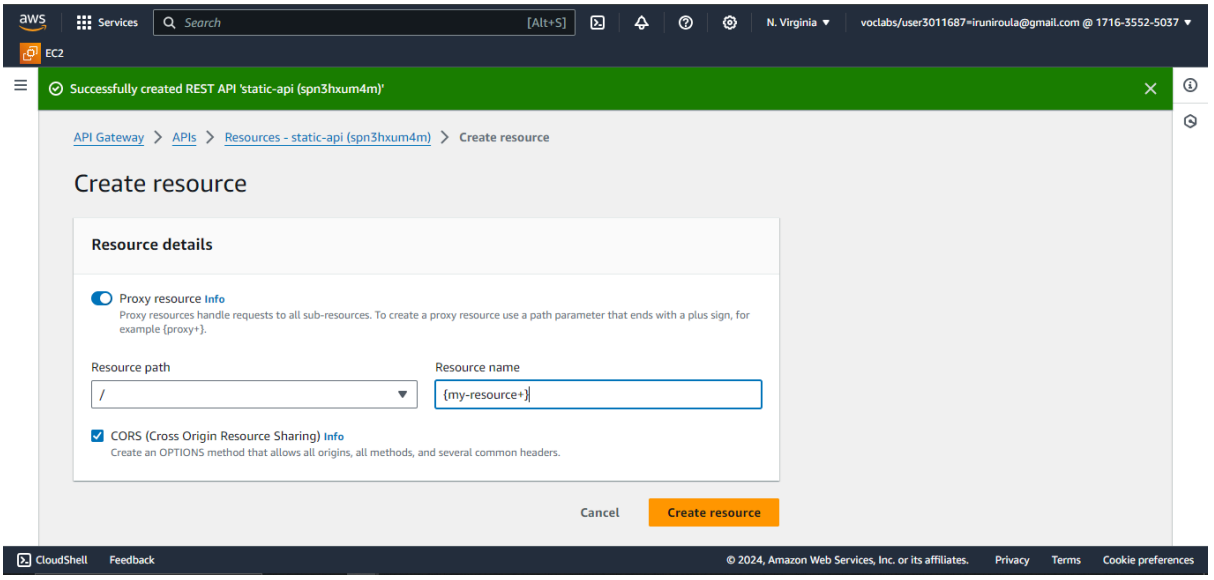


Then a lambda function is created and python code is written and deployed in it a trigger is also added.



Create Rest API, then create a Resource, then a method and stage within it and finally deploy it





Resource details

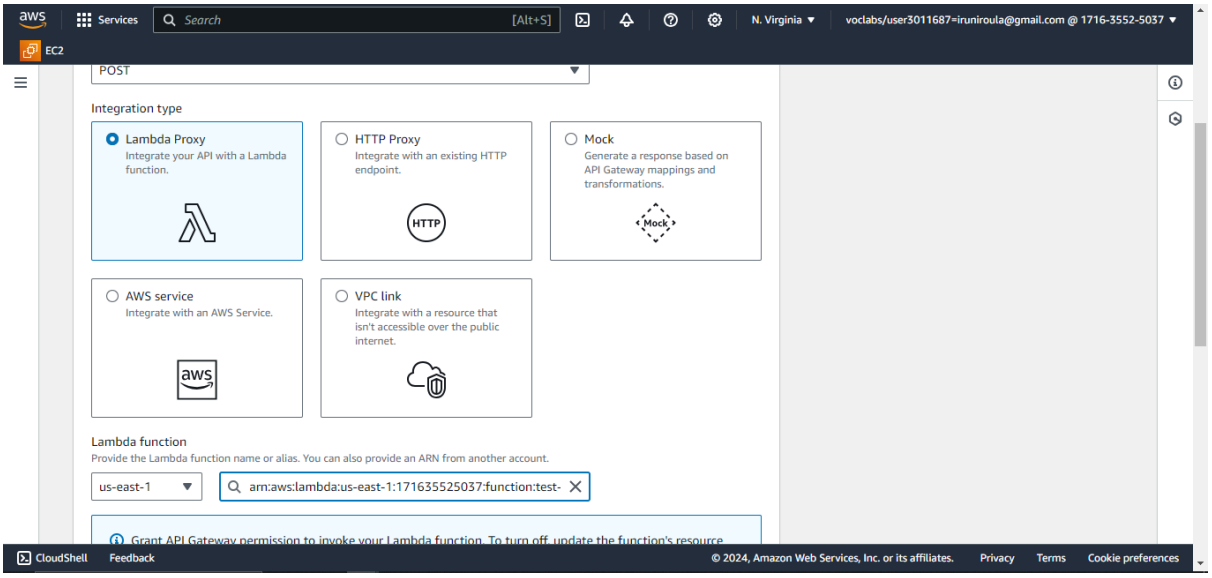
Delete | Update documentation | Enable CORS

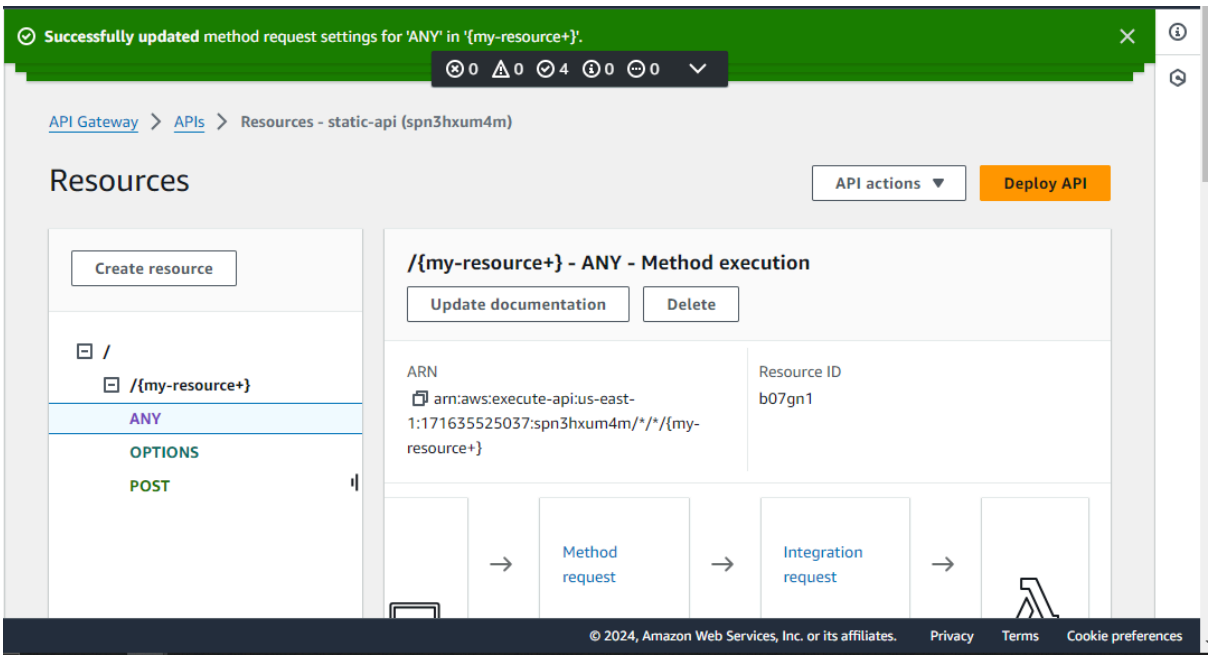
Path	Resource ID
/{my-resource+}	b07gn1

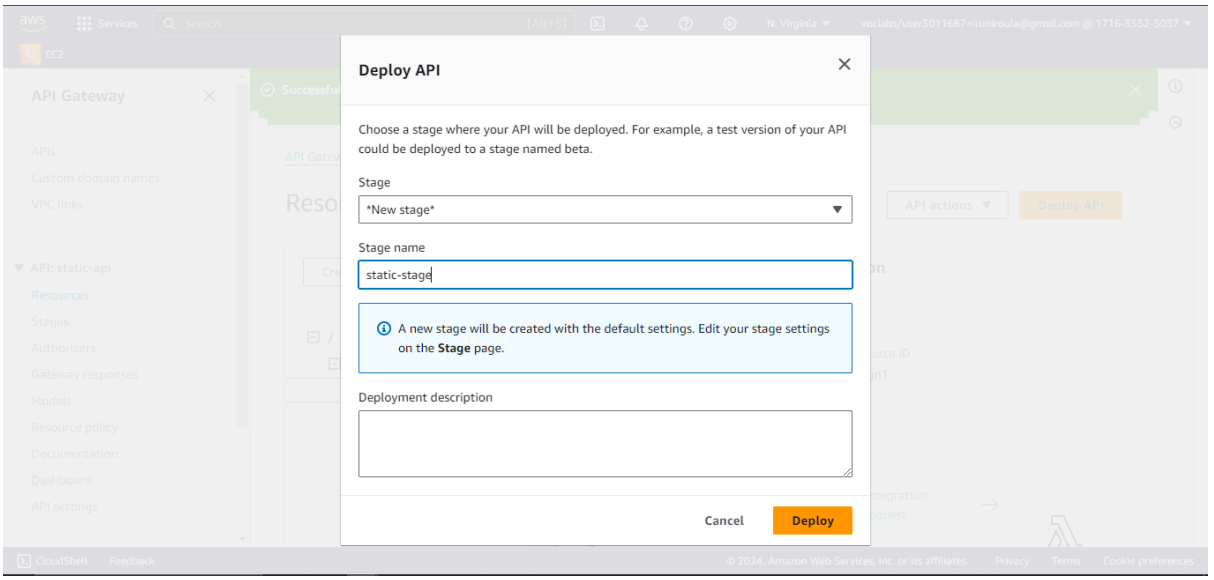
Methods (2)

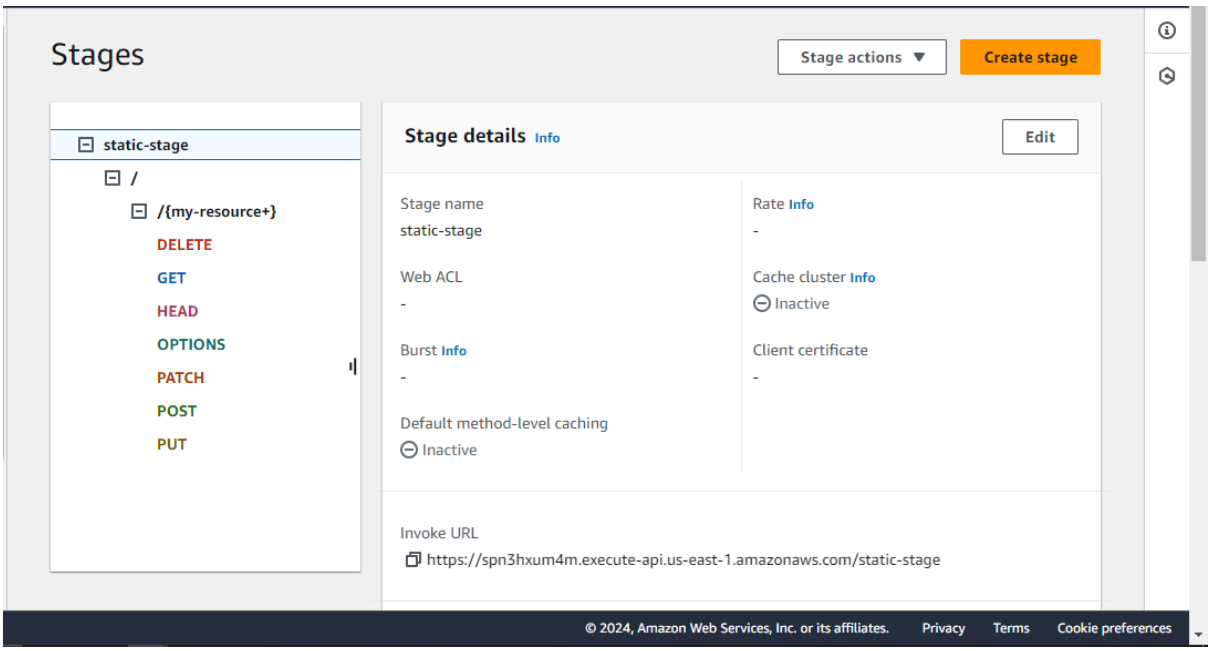
Delete | Create method

	Method type	Integration type	Authorization	API key
○	ANY	Not setup	None	Not required
○	OPTIONS	Mock	None	Not required

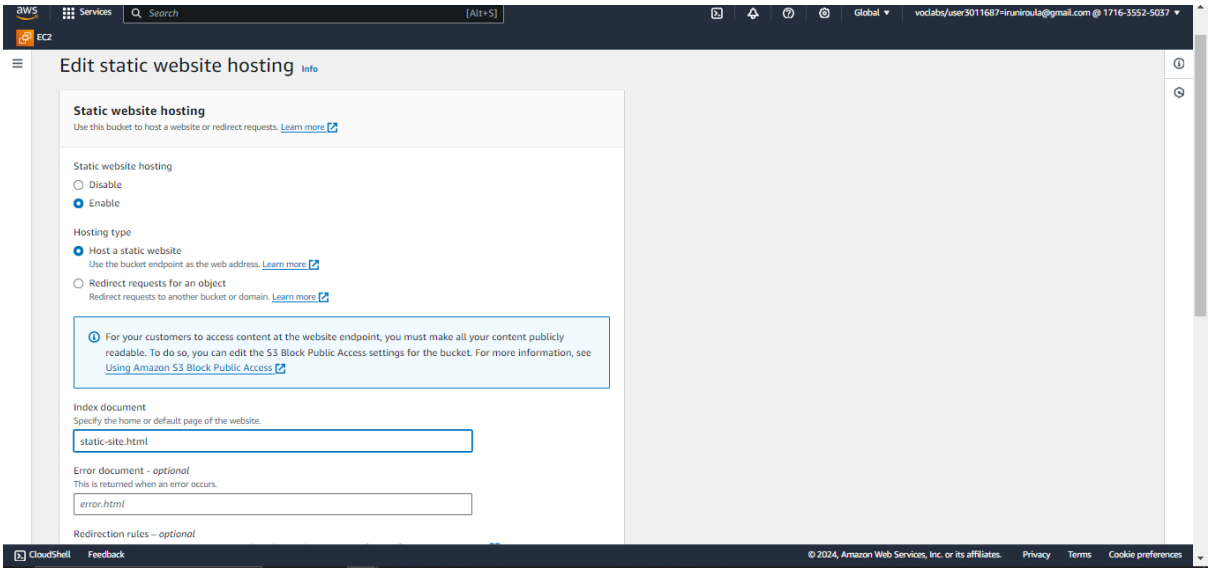



Successfully updated method request settings for 'ANY' in '{my-resource+}'.
API Gateway
APIs
Resources - static-api (spn3hxum4m)
Resources
API actions
Deploy API
Create resource
/{my-resource+} - ANY - Method execution
Update documentation
Delete
/
/{my-resource+}
ANY
OPTIONS
POST
ARN
arn:aws:execute-api:us-east-1:171635525037:spn3hxum4m/*/*/{my-resource+}
Resource ID
b07gn1
Method request
Integration request
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Deploy API
Choose a stage where your API will be deployed. For example, a test version of your API could be deployed to a stage named beta.
Stage
New stage
Stage name
static-stage
A new stage will be created with the default settings. Edit your stage settings on the Stage page.
Deployment description
Cancel
Deploy



Edit necessary policies within the bucket to allow static hosting



Policy

```
{
    "Version": "2012-10-17",
    "Id": "Policy1709791241957",
    "Statement": [
        {
            "Sid": "Stmt1709791239327",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::irusbalti/*"
        }
    ]
}
```

When information is filled in this contact form,

It is now recorded in the DynamoDB as follows: