**CREATING EC2 INSTANCES USING LAMBDA AND S3 BUCKET AS**

**A LAMBDA TRIGGER**

*Prepared in the partial fulfilment of the Summer Internship Program on AWS*

AT



*Under the guidance of*

***Mrs. Sumana Bethala, APSSDC***

***Mr. Rama Krishna, APSSDC***

***Done By***

*AP21110010451 Ramcharan Badagala*

***SRM University AP***

*(August, 2023)*

## ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to all those who have contributed to the successful completion of my summer internship project at **Andhra Pradesh Skill Development Corporation (APSSDC)**. This opportunity has been an enriching and transformative experience for me, and I am truly thankful for the support, guidance, and encouragement I have received along the way.

First and foremost, I extend my sincere regards to Mrs Sumana Bethala and Mr Rama Krishna, my supervisors and mentors, for providing me with valuable insights, constant guidance, and unwavering support throughout the duration of the internship. Their expertise and encouragement have been instrumental in shaping the direction of this project.

I would like to thank the entire team at **Andhra Pradesh Skill Development Corporation (APSSDC)** for fostering a collaborative and innovative environment. The camaraderie, knowledge sharing, and feedback I received from my colleagues significantly contributed to the development and success of this project.

In conclusion, I am honoured to have been a part of this internship program, and I look forward to leveraging the skills and knowledge gained to contribute positively to future endeavours.

Thank you.

Sincerely,

## ABSTRACT

Amazon Web Services (AWS) provides a powerful set of tools for creating and managing cloud-based infrastructure. One of these tools is AWS Lambda, a serverless computing service that allows you to run code in response to events. Another tool is Amazon Simple Storage Service (S3), a highly scalable and durable object storage service.

You can use these two services together to create an EC2 instance using a Lambda function triggered by an S3 bucket. This can be useful for automating the process of launching EC2 instances in response to certain events, such as the upload of a new file to an S3 bucket.

To set this up, you'll need to create an S3 bucket and a Lambda function that returns the object type of objects in the S3 bucket. Then, you'll need to configure a Lambda trigger that invokes your function when objects are uploaded to your bucket. You can test your function first with a dummy event and then using the trigger.

You can also use a Lambda function to create an EC2 instance. This can be done by writing a Lambda function in Python using the Boto3 library. You'll also need to create a custom Lambda execution policy for your IAM role. Once you've done this, you'll be able to log in to the new EC2 instance via SSH.

To get started, you'll need to create an EC2 key pair and then create a Lambda function with the necessary settings. You'll also need to navigate to IAM and edit the policy for your newly created role. Then, you can paste in the Python source code for your Lambda function and set the necessary environment variables. Finally, you can test your Lambda function and observe that an EC2 instance is initializing.

Once your EC2 instance is up and running, you can connect to it via SSH using the .pem file that you downloaded earlier. This setup allows you to automate the process of launching EC2 instances in response to events in your S3 bucket, providing a powerful and flexible way to manage your cloud infrastructure.

This process can be automated so that every time you upload a new version of your web page folder to the S3 bucket, the Lambda function will automatically create a new EC2 instance and host the updated website.

## TABLE OF CONTENTS

### S.No Content Pg.No

[1. Introduction 6,7](#_Toc13262)

[2. Methodology 7](#_Toc13263)

[3. System Design / Architecture 8,9](#_Toc13264)

[4. Implementation 9-15](#_Toc13265)

[5. Results 16](#_Toc13266)

[6. Conclusion 16,17](#_Toc13267)

# INTRODUCTION

You can use an Amazon S3 trigger to invoke a Lambda function. This can be done by configuring a trigger for an Amazon Simple Storage Service (Amazon S3) bucket. Every time that you add an object to your Amazon S3 bucket, your function runs and outputs the object type to Amazon CloudWatch Logs. You can use a Lambda function with an Amazon S3 trigger to perform many types of file processing tasks. For example, you can use a Lambda function to create a thumbnail whenever an image file is uploaded to your Amazon S3 bucket, or to convert uploaded documents into different formats.

To complete this process, you need to carry out the following steps:

1. Create an Amazon S3 bucket.

2. Create a Lambda function that returns the object type of objects in an Amazon S3 bucket.

3. Configure a Lambda trigger that invokes your function when objects are uploaded to your bucket.

4. Test your function, first with a dummy event, and then using the trigger.

By completing these steps, you’ll learn how to configure a Lambda function to run whenever objects are added to or deleted from an Amazon S3 bucket. You can complete this tutorial using only the AWS Management Console. If you do not have an AWS account, you can sign up for one. After signing up for an AWS account, you should create an administrative user so that you don't use the root user for everyday tasks. You can also turn on multi-factor authentication (MFA) for your root user for added security.

The following AWS services utilized in this project:

*Amazon EC2 (Elastic Compute Cloud):* The application is hosted on Amazon EC2 instance, which provide resizable compute capacity in the cloud. EC2 instances act as the application servers, handling user requests and managing the system's backend processes.

# 

# METHODOLOGY

# Requirements Gathering

To create an EC2 instance using a Lambda function triggered by an Amazon S3 bucket, you need to gather the following requirements:

1. An Amazon Web Services (AWS) account.

2. An Amazon S3 bucket to use as a trigger.

3. An IAM role with permissions to create EC2 instances and access the Amazon S3 bucket.

4. A Lambda function that creates an EC2 instance when triggered by an Amazon S3 event.

Once you have gathered these requirements, you can configure the Amazon S3 bucket to trigger the Lambda function whenever an object is added or deleted from the bucket. The Lambda function will then create an EC2 instance based on the specified configuration.

# 3. SYSTEM DESIGN / ARCHITECTURE

### *Amazon EC2 (Elastic Compute Cloud)*

Amazon EC2 is at the heart of the system's hosting infrastructure. It provides resizable compute capacity in the cloud, allowing the deployment of virtual server instances. The EC2 instance acts as the application server for the Student Portfolio Management System. Key aspects of its usage include:

*Instance Provisioning:* An Amazon EC2 instance was created and configured with an Ubuntu Amazon Machine Image (AMI). The choice of AMI ensured compatibility and ease of setup for the Ubuntu operating system.

*Resource Scaling:* EC2 instances can be scaled vertically or horizontally to accommodate varying levels of user traffic. This scalability ensures that the system remains responsive and available, even during peak usage periods.

### *Security Groups*

AWS Security Groups provide essential security measures to control inbound and outbound traffic to the EC2 instance. Key aspects include:

*Inbound Access Control:* Security Groups define rules that control incoming traffic to the EC2 instance. These rules specify the allowed IP ranges, ports, and protocols, ensuring that only authorized traffic reaches the instance.

### *Continuous Monitoring and Scaling*

AWS provides tools and services for monitoring the health, performance, and utilization of resources. CloudWatch, an AWS monitoring service, is used to monitor the system and trigger automatic scaling when needed. Key aspects include:

*Performance Monitoring:* CloudWatch monitors key metrics such as CPU utilization, memory usage, and network traffic of the EC2 instance. This monitoring helps identify performance bottlenecks and resource utilization trends.

*Automatic Scaling:* Based on defined thresholds and rules, CloudWatch can trigger automatic scaling actions. For example, if CPU utilization surpasses a certain threshold, additional EC2 instances can be launched to handle the increased load.

# 4. IMPLEMENTATION

Creating s3 bucket

You can create an Amazon S3 bucket using the AWS Management Console, the AWS CLI, or the AWS SDKs. Here are the steps to create an S3 bucket using the AWS Management Console:

1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.

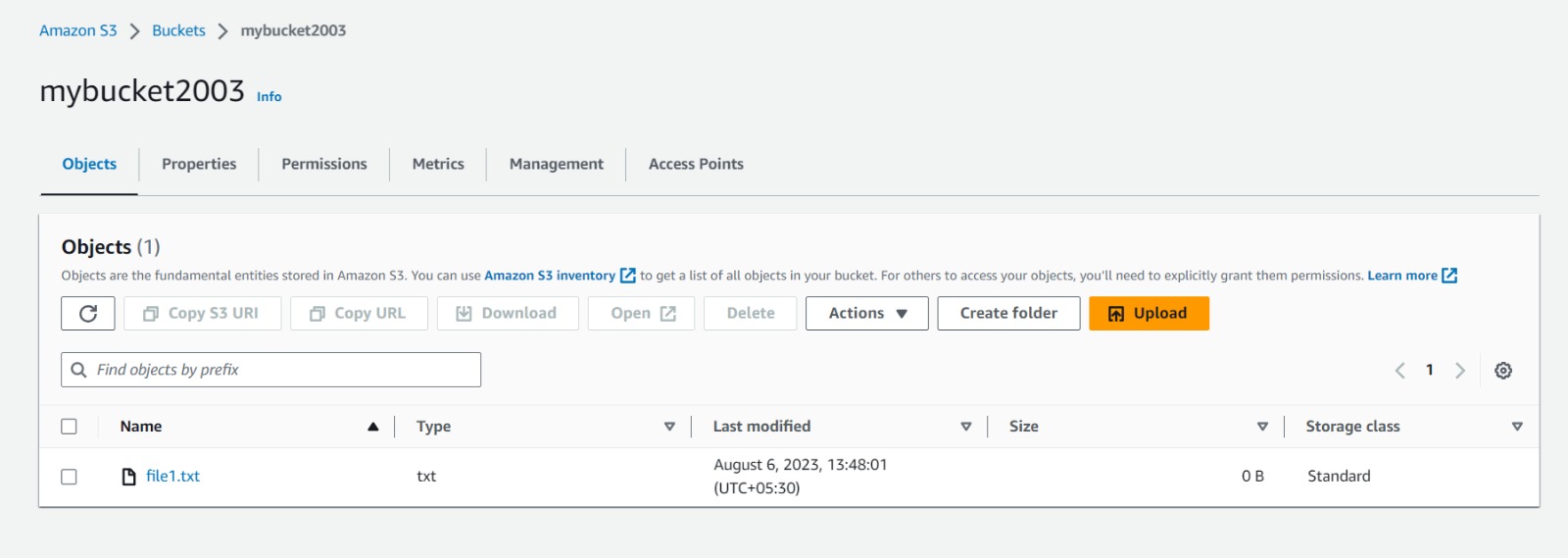
2. Choose \*Create bucket\*.

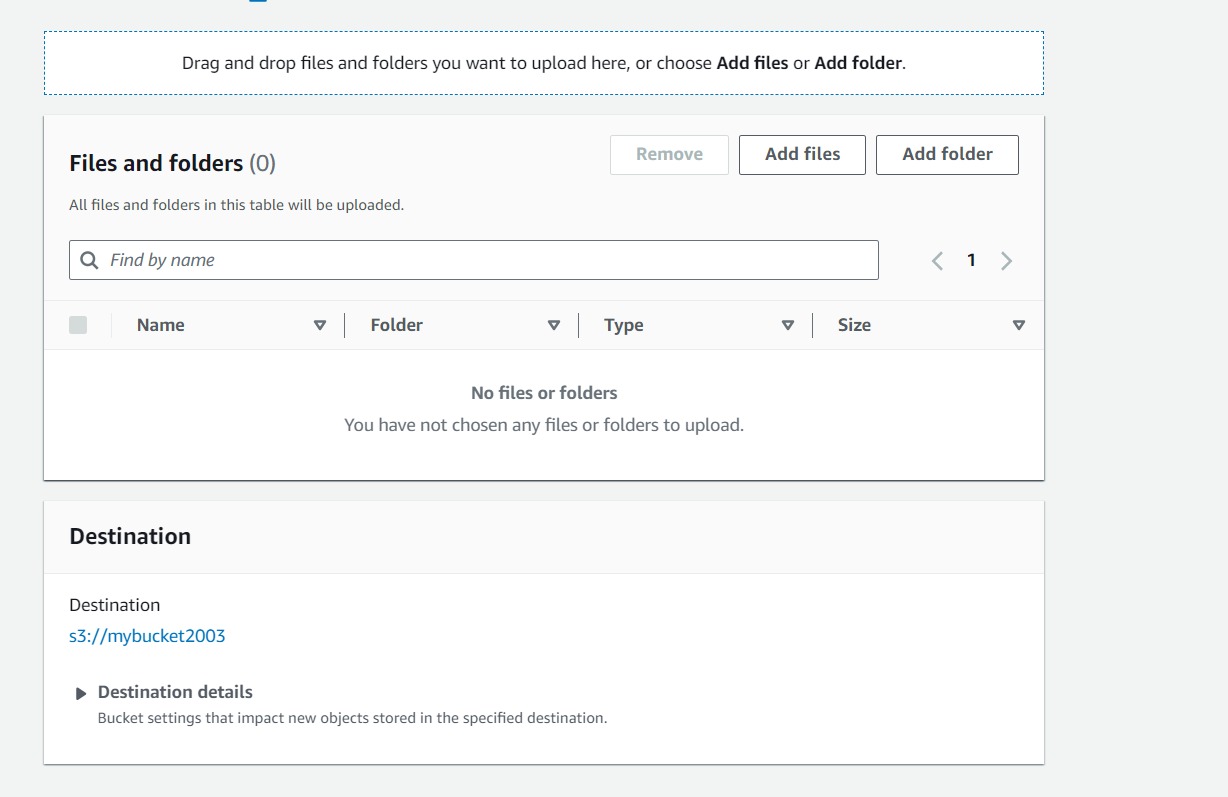
3. In the \*Bucket name\* field, type a unique DNS-compliant name for your new bucket.

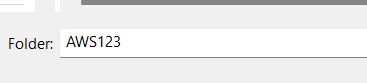
4. In the \*Region\* field, choose the AWS Region where you want the bucket to reside.

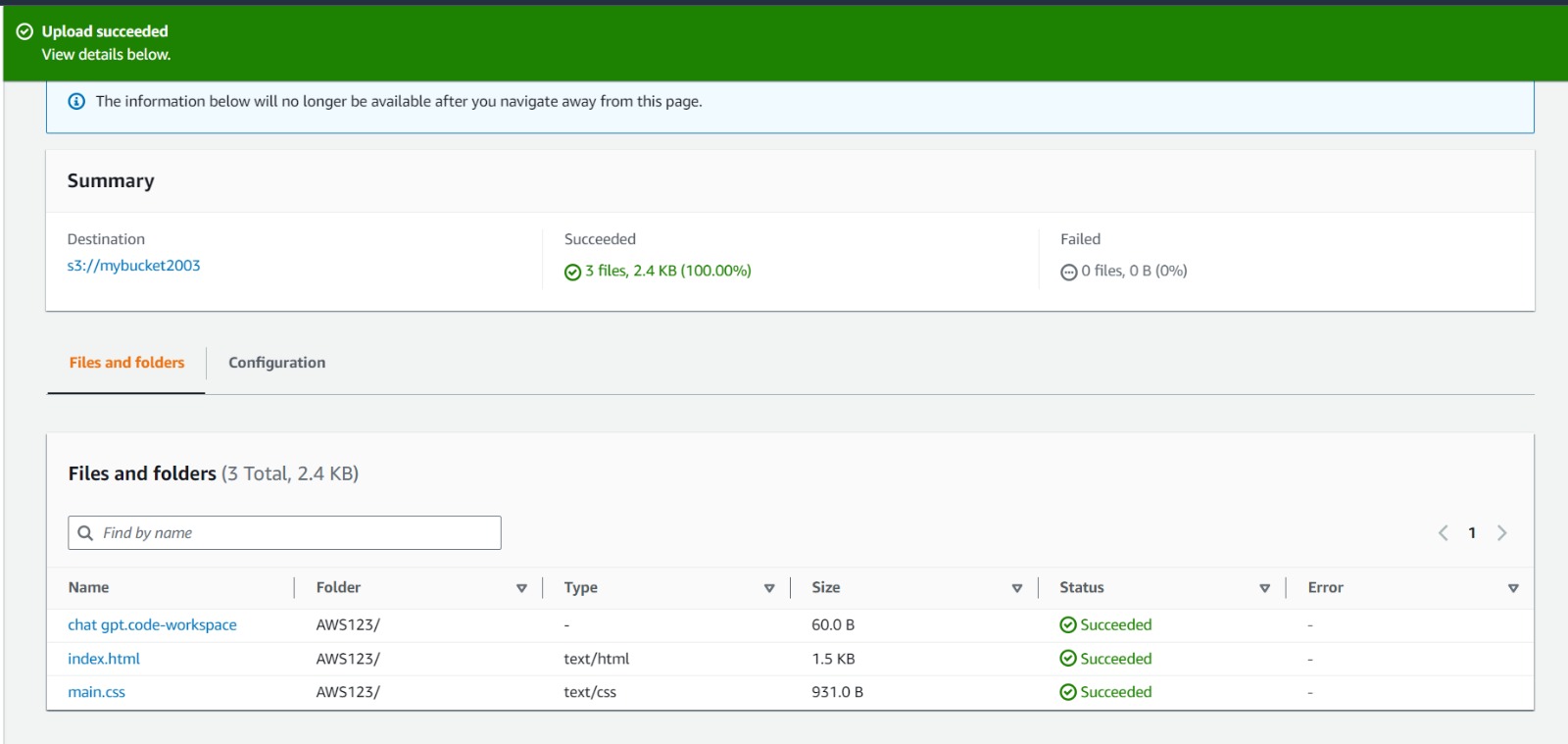
5. Choose \*Create\* in the lower right of the console.

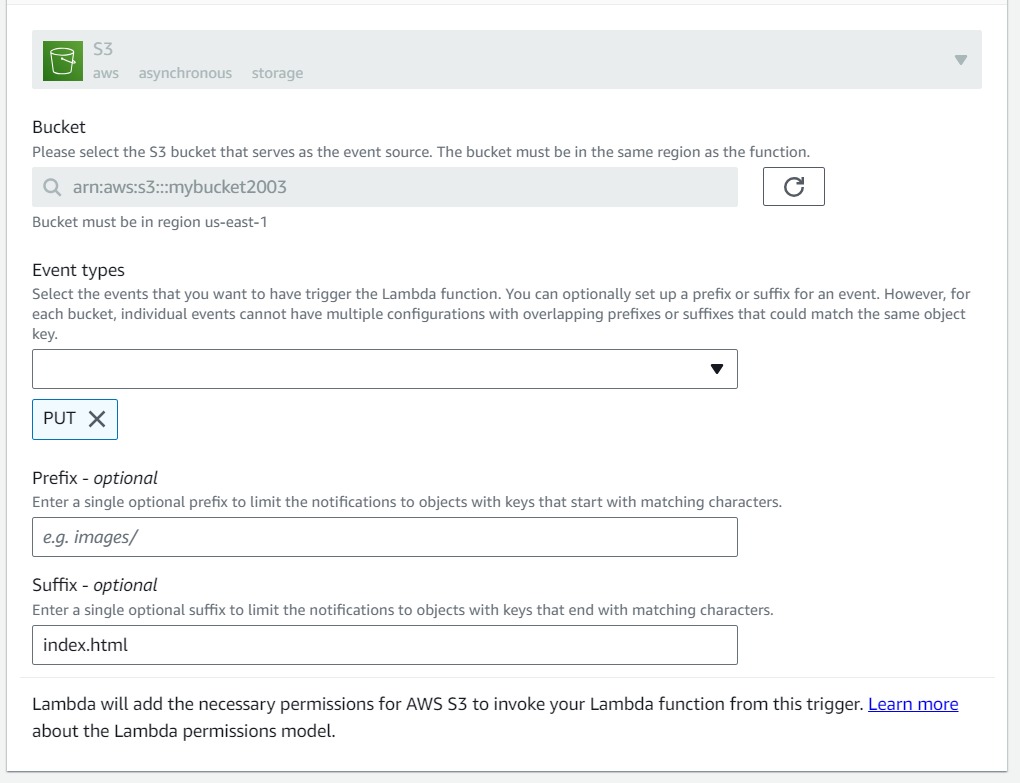
After you create a bucket, you can upload an unlimited number of data objects to it. Each object can contain up to 5 terabytes of data. Each object is stored and retrieved using a unique developer-assigned key.











Creating lambda function

You can create a Lambda function using the AWS Management Console, the AWS CLI, or the AWS SDKs. Here are the steps to create a Lambda function using the AWS Management Console:

1. Open the [AWS Lambda console].

2. Choose \*Create function\*.

3. Select \*Use a blueprint\*.

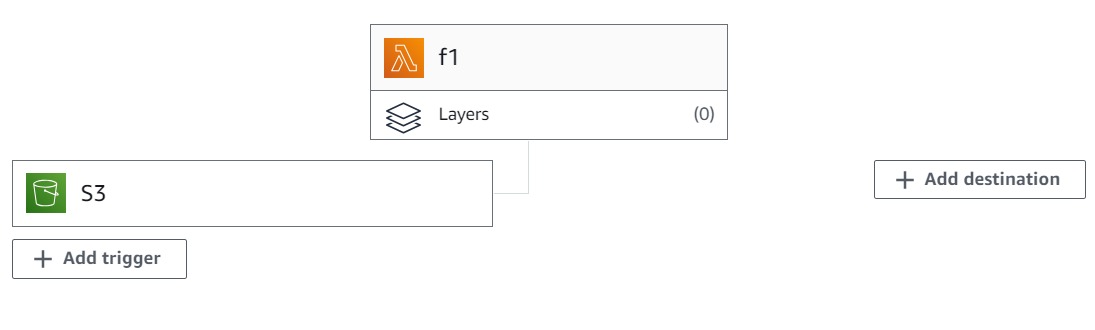
4. Open the \*Select blueprint\* dropdown list and search for \*Hello world function\*.

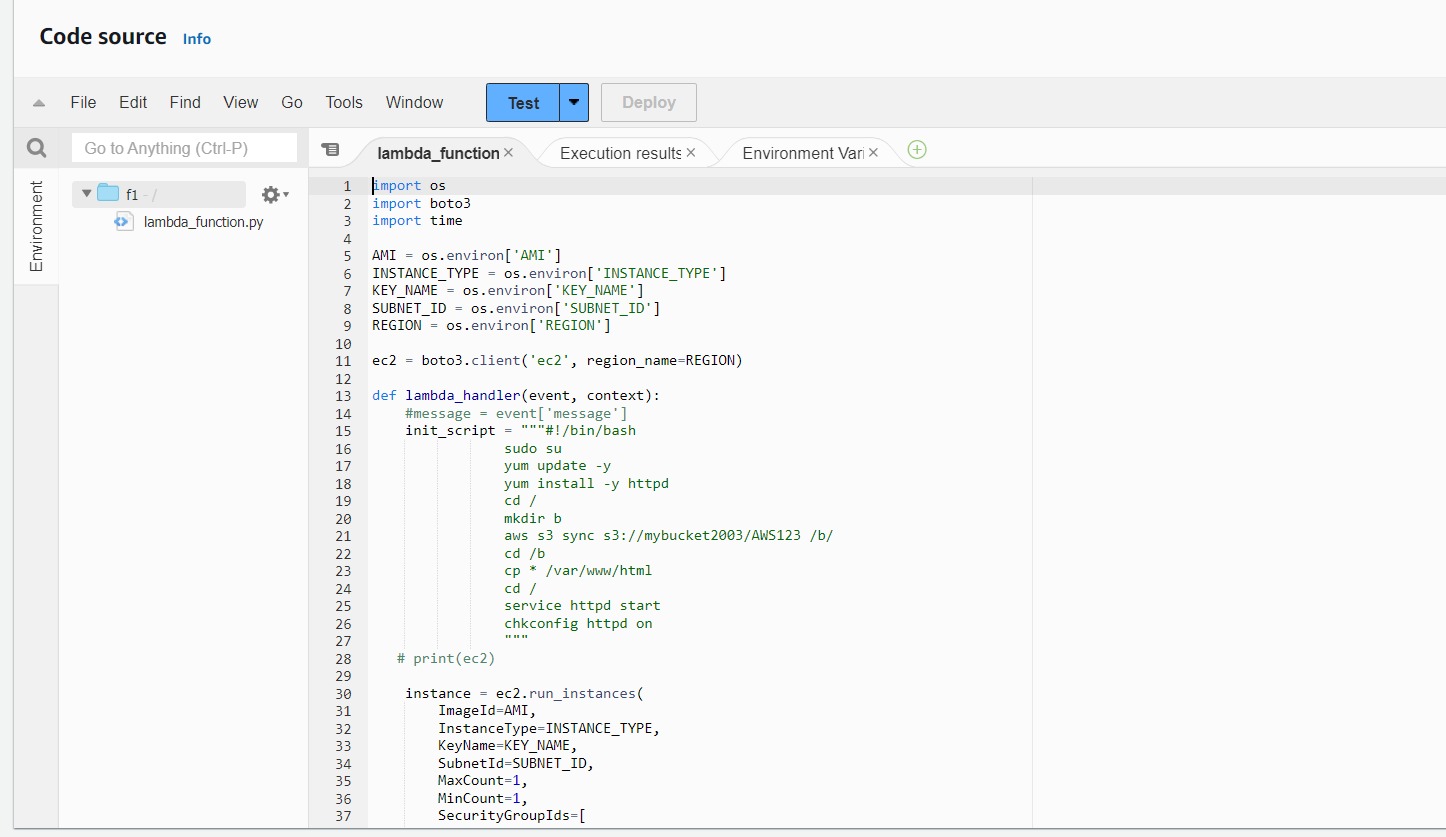
5. Select \*nodejs\* or \*python\*.

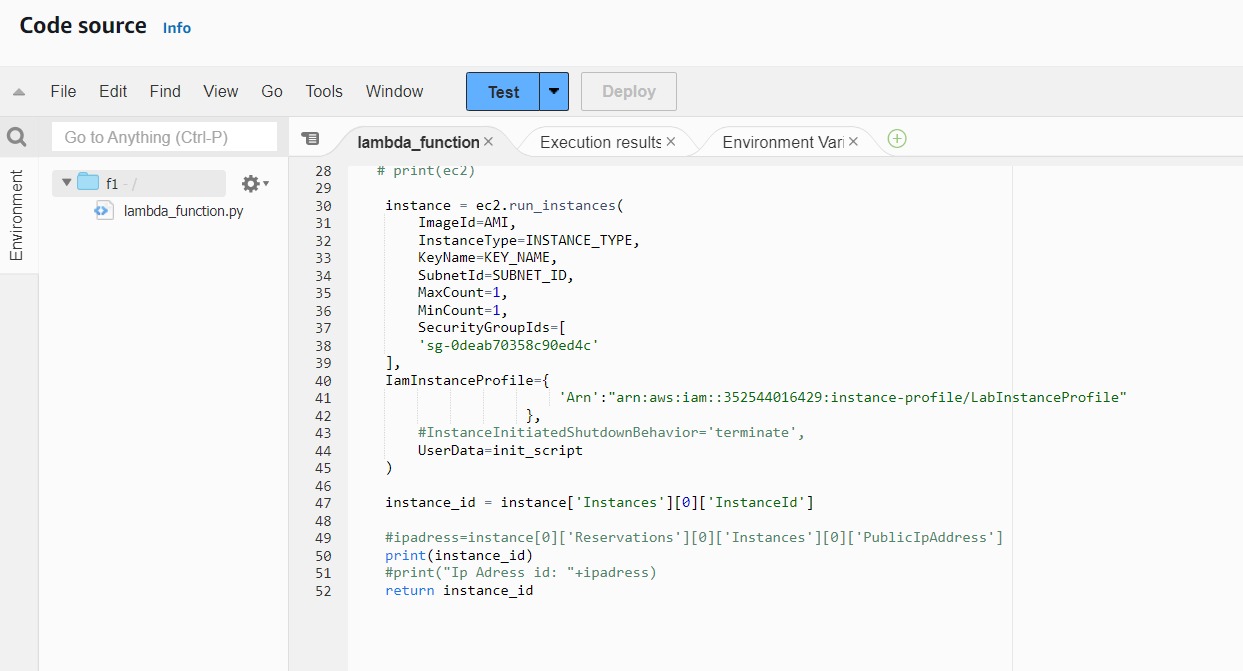
6. Enter a \*Function name\*.

7. For \*Execution role, choose \*\*Create a new role with basic Lambda permissions\*.

After you create your Lambda function, you can test it by choosing \*Test\* on the function configuration page. You can also view the logs and metrics for your function to see how it is performing.







Creating ec2 instance

Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage.

To create an EC2 instance, you can use the AWS Management Console, the AWS CLI, or the AWS SDKs. Here are the steps to create an EC2 instance using the AWS Management Console:

1. Open the Amazon EC2 console at https://console.aws.amazon.com/ec2/.

2. In the navigation pane, choose \*Instances, and then choose \*\*Launch Instance\*.

3. Choose an Amazon Machine Image (AMI), which is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance.

4. Choose an instance type, which determines the hardware of the host computer for your instance.

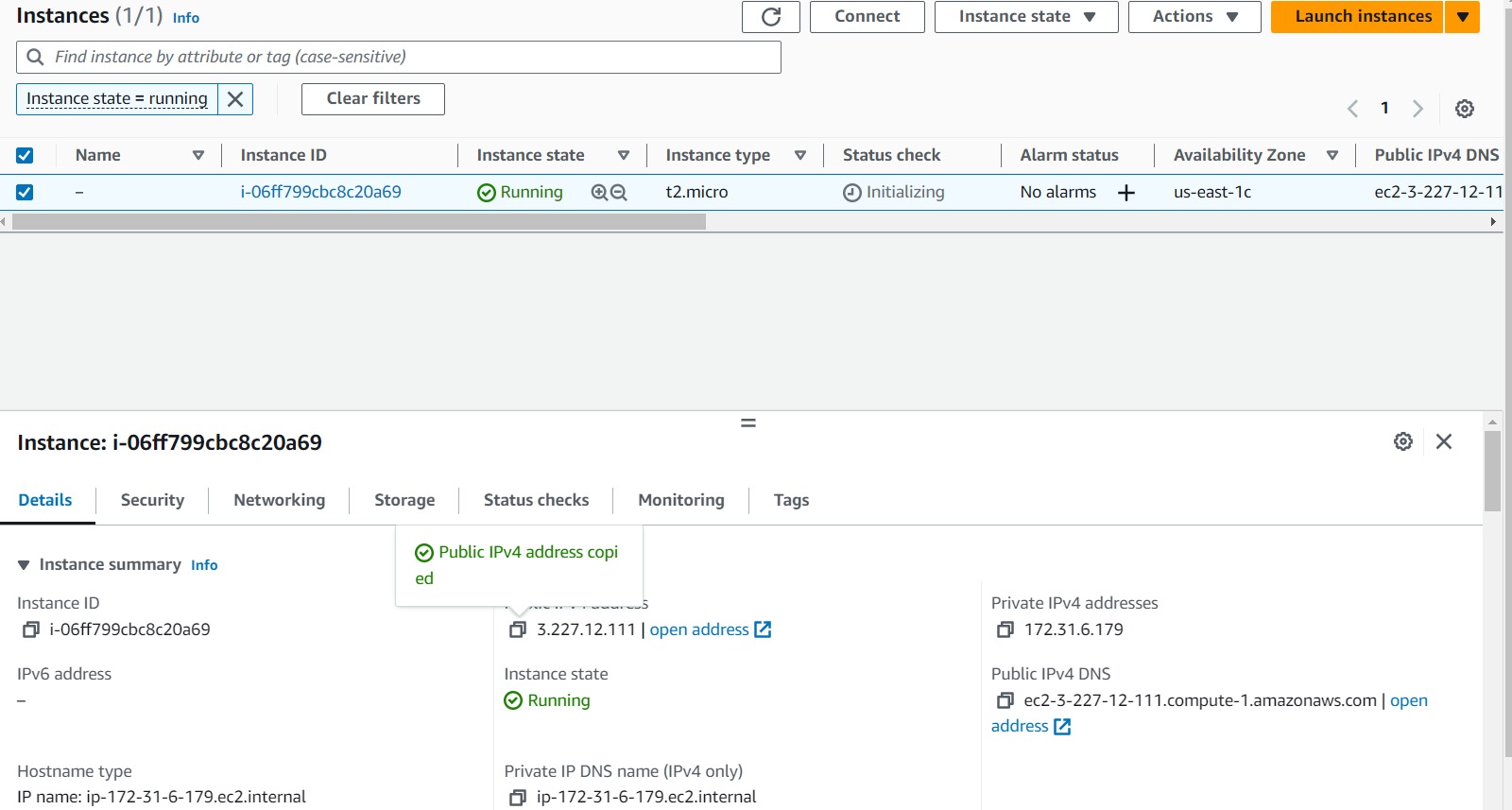
5. Configure the instance details as needed.

6. Add storage to your instance.

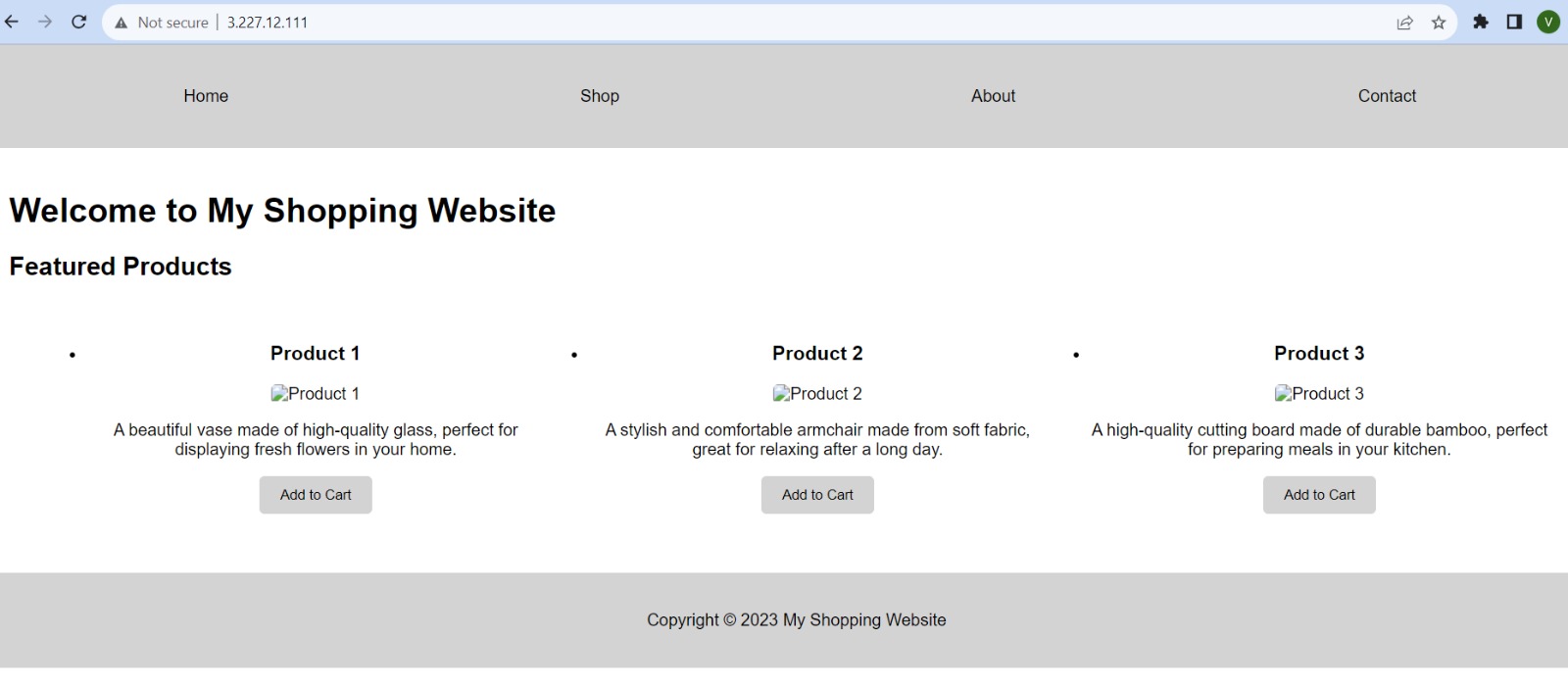
7. Tag your instance.

8. Configure the security group for your instance.

9. Review and launch your instance.



# 5. RESULTS



# 6. CONCLUSION

In conclusion, using AWS Lambda and Amazon S3 together can provide a powerful and flexible way to automate the process of launching EC2 instances. By creating a Lambda function that is triggered by an S3 bucket, you can automatically launch EC2 instances in response to certain events, such as the upload of a new file to your S3 bucket.

To set this up, you'll need to create an S3 bucket and a Lambda function that returns the object type of objects in the S3 bucket. Then, you'll need to configure a Lambda trigger that invokes your function when objects are uploaded to your bucket. You can test your function first with a dummy event and then using the trigger.

You can also use a Lambda function to create an EC2 instance. This can be done by writing a Lambda function in Python using the Boto3 library. You'll also need to create a custom Lambda execution policy for your IAM role. Once you've done this, you'll be able to log in to the new EC2 instance via SSH.

Overall, using AWS Lambda and Amazon S3 together provides a powerful and flexible way to automate the process of launching EC2 instances. This can save time and effort, allowing you to focus on other aspects of managing your cloud infrastructure.

**This allows you to easily deploy new versions of your web application by simply uploading the files to your S3 bucket.**

**REFRENCE LINKS**

<https://docs.aws.amazon.com/lambda/latest/dg/with-s3-example.html>

<https://www.pluralsight.com/cloud-guru/labs/aws/creating-an-ec2-instance-with-lambda-in-aws>

<https://stackoverflow.com/questions/66433454/how-do-i-copy-a-file-from-s3-bucket-to-ec2-instance-using-lambda-function>

**VIDEO LINK**

<https://www.youtube.com/watch?v=GzqHxjLC3Q4&pp=ygVFQ1JFQVRJTkcgRUMyIElOU1RBTkNFUyBVU0lORyBMQU1CREEgQU5EIFMzIEJVQ0tFVCBBUyBBIExBTUJEQSBUUklHR0VS>