

*Project: Deploy a Static Website on*

*AWS*

**Table of Contents**

|  |  |  |  |
| --- | --- | --- | --- |
| **1** | [**Introduction**](#page3) | | **3** |
|  | 1.1 [3](#page4) Setup Checklist for Mini Project | | 4 |
|  | 1.2 | [Instructions](#page4) | 4 |
| **2** | [**Problem Statement**](#page5) | | **5** |
|  | 2.1 | [Objective](#page5) | 5 |
|  | 2.2 | [Technology used](#page5) | 5 |
| **3** | [**Implementation**](#page6) | | **6** |
|  | 3.1 | [Steps for deploying static website](#page6) | 6 |
|  | 3.2 | Functional requirement | 10 |

* Introduction

In the modern era of web development, hosting static websites has become more efficient, scalable, and cost-effective with the help of cloud platforms. This project focuses on deploying a static website using Amazon Web Services (AWS), leveraging its powerful services such as Amazon S3, CloudFront, and AWS Certificate Manager .

A **static website** consists of fixed content—HTML, CSS, JavaScript, and media files—which does not change dynamically based on user input

**.1 SETUP CHECKLIST FOR MINIPROJECT**

* Hardware:
  + Intel processor i3
  + Microsoft Windows
  + Memory of 4GB RAM (4GB or more recommended)

➢

Software:

* Chatgpt
* Google
* AWS service
* S3 bucket
* HTML , CSS , Javascript
* VS code

**.2** **INSTRUCTIONS**

* Here is a step-by-step instruction guide to deploy a static website on AWS using Amazon S3 (Simple Storage Service), which is the most common and cost-effective way for hosting static sites.
* An AWS account
* A static website (HTML/CSS/JS files)
* (Optional) Domain name registered (can be with Route 53 or other)

* Problem Statement

**.1 OBJECTIVE**

**Key Points in the Objective:**

* **Use AWS S3 to host static website files.**
* **Ensure public access with appropriate security settings.**
* **Enable fast loading using CloudFront (optional).**
* **Deploy a site that is easily maintainable and scalable.**
* **Gain hands-on experience with cloud infrastructure for static hosting.**

**.2** **ABSTRACT OF THE PROJECT**

To deploy a static website on AWS for a project, you can leverage Amazon S3 for cost-effective and scalable hosting. The process involves creating an S3 bucket, enabling static website hosting, uploading your website files (HTML, CSS, JavaScript, etc.), and configuring public access. This approach provides a simple and efficient way to make your static website accessible to users.

**.3 TECHNOLOGY USED:**

* Github

➢

AWS (S3 Bucket

➢

ChatGPT

➢

Google

➢

Napkin AI

* Implementation

**.1** **Steps for deploying static website:**

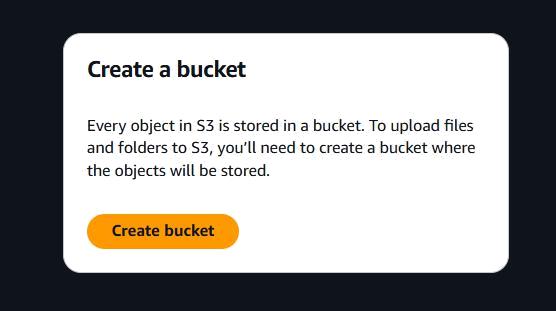
**Lab:** Deploy a Static Website on AWS S3

AWS account

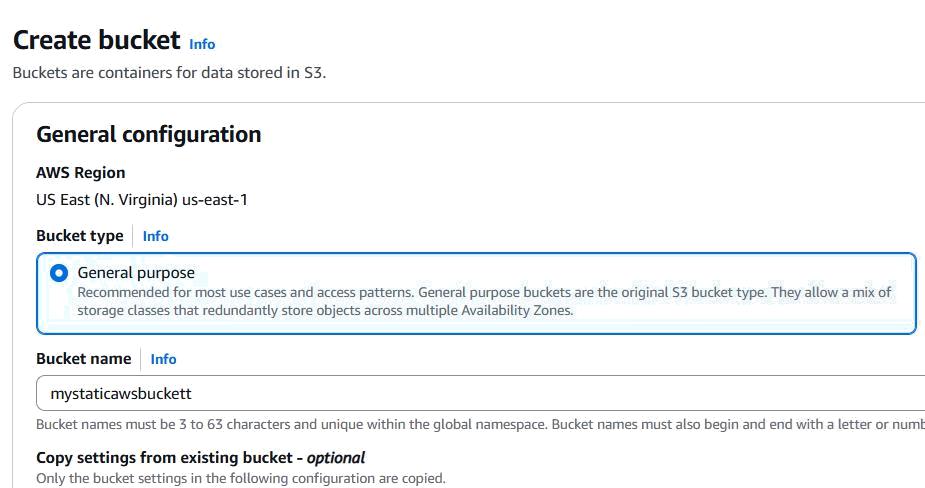
**Basic static website files (index.html, style.css, etc.)**

Step 1: Create an S3 Bucket

* Go to the S3 service in the AWS Management Console.
* Click on “Create bucket”.



* Enter a unique bucket name (e.g., **mystaticawsbuckett**).



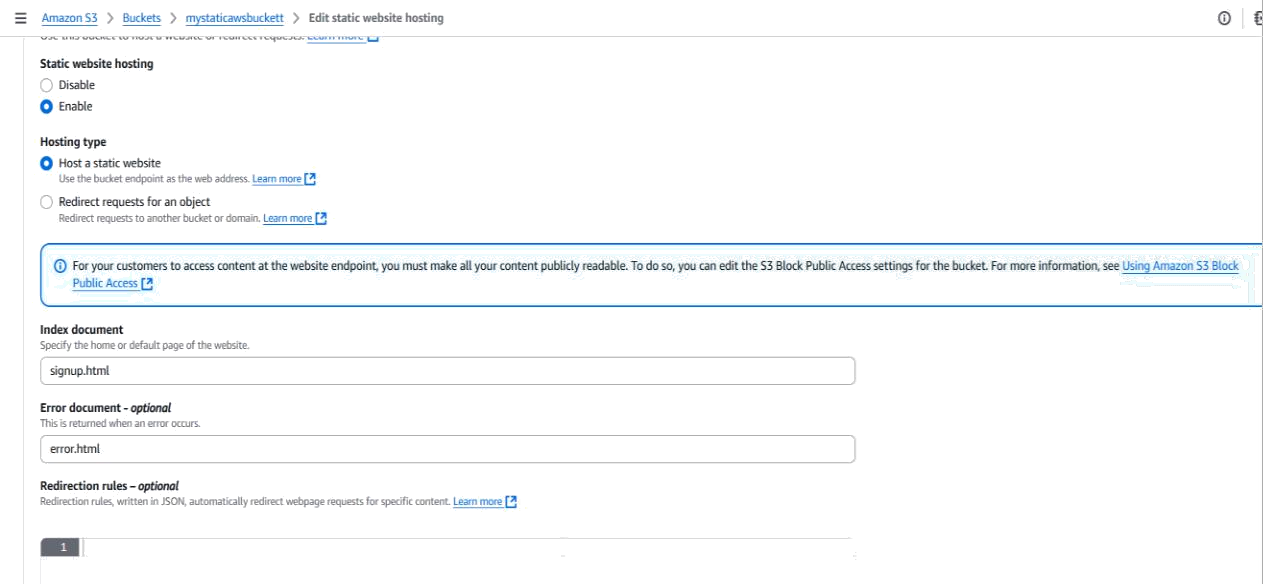
* Select a region.
* Uncheck “Block all public access” under Permissions.
* Acknowledge the warning and click Create bucket.

Step 2: Upload Website Files

* Click the newly created bucket.
* Go to the “Objects” tab.
* Click “Upload” → Add files (upload index.html, style.css, etc.).
* Click Upload.

Step 3: Enable Static Website Hosting

* In the bucket, go to the “Properties” tab.
* Scroll to “Static website hosting”.

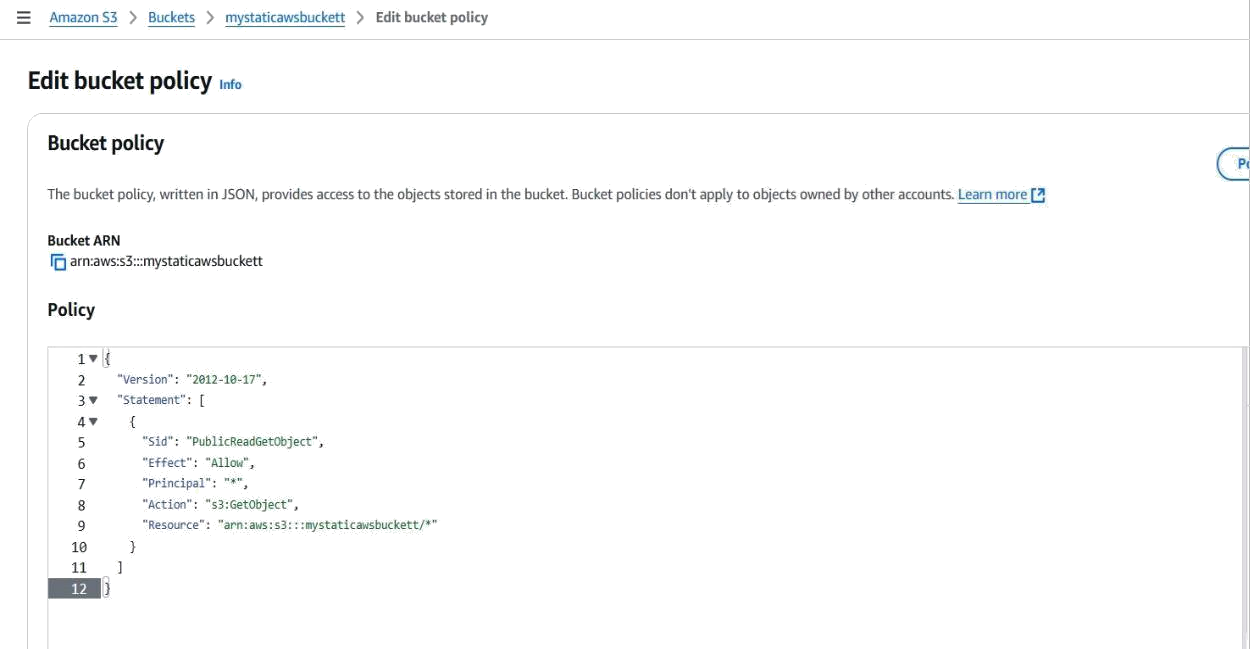
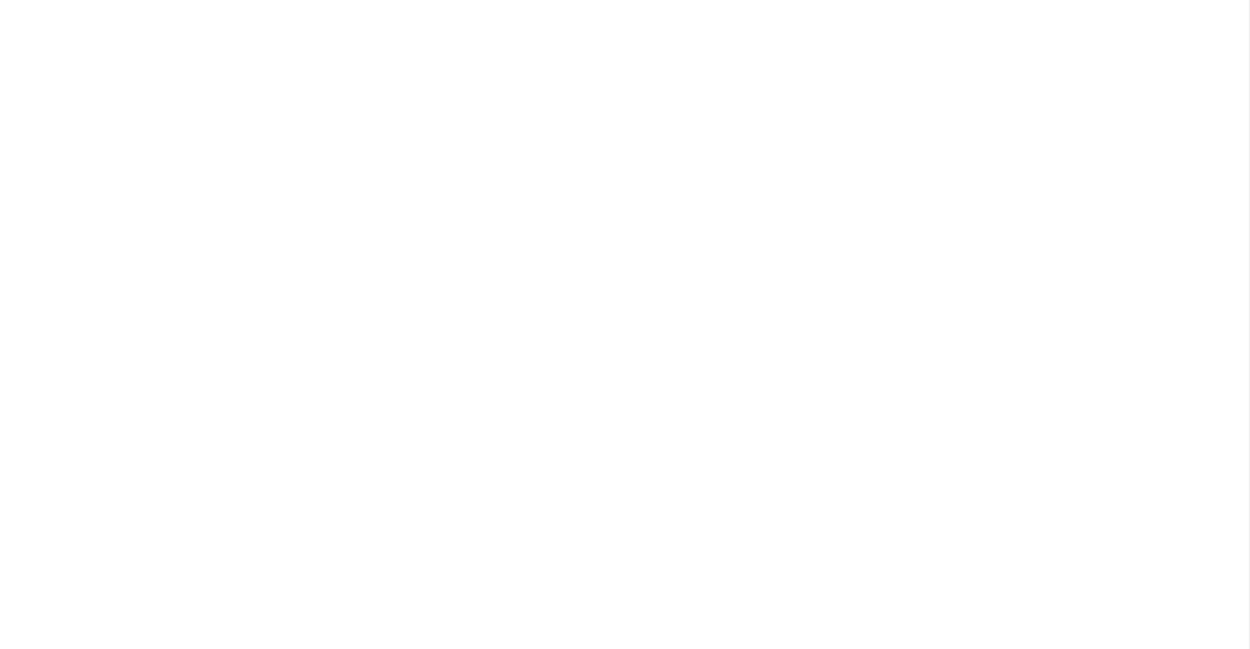


* Click “Edit”.

* Enable it:
* Select “Enable”.
* Enter signup.html as the Index document. (Optional) Enter error.html if available.
* Save changes.

Step 4: Set Permissions to Make the Site Public

* Go to the Permissions tab.
* Scroll to “Bucket policy” → Click Edit.



* Add this JSON policy, replacing YOUR\_BUCKET\_NAME with your bucket's name:

{

"Version": "2012-10-17",

"Statement": [

{

"Sid": "PublicReadGetObject",

"Effect": "Allow",

"Principal": "\*",

"Action": "s3:GetObject",

"Resource": "arn:aws:s3:::YOUR\_BUCKET\_NAME/\*"

}

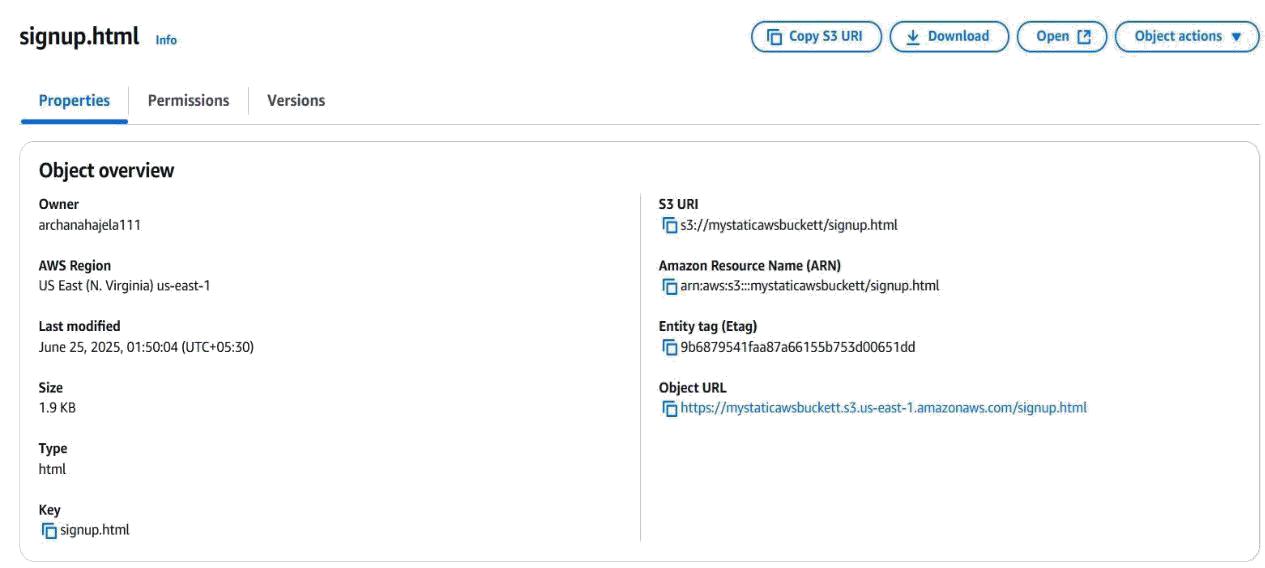
]

}

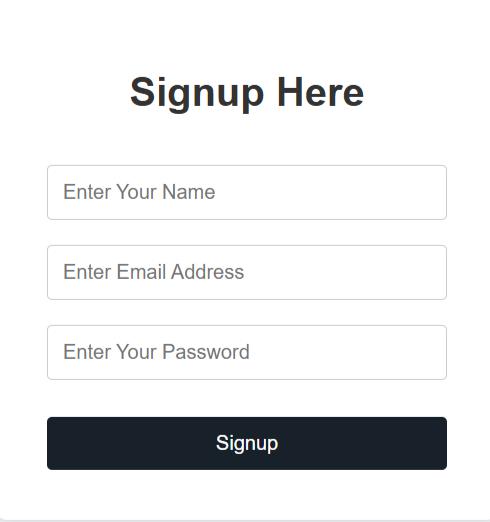
* Save changes.

Step 5: Access Your Static Website

* Go back to Properties → Static website hosting.



* Copy the “Bucket website endpoint” URL.
* Paste it in a browser to view your website.



**FUNCTIONAL REQUIREMENTS**

To deploy a static website on AWS, the functional requirements include creating an

S3 bucket, configuring it for static website hosting, uploading website files, setting appropriate permissions, and potentially integrating with CloudFront and Route 53 for enhanced performance and domain management. Specifically, you need to ensure the website is accessible via a custom domain and supports HTTPS.