

Conception Phase

Hosting Static AI Website using AWS and Terraform

1: Website Preparation

A static AI landing page is created using HTML with embedded CSS. The main file, *index.html*, represents the user interface and will be hosted in the cloud.

2: S3 Bucket Creation and Configuration

Using Terraform, an Amazon S3 bucket is created to store the website files. Static website hosting is enabled, and *index.html* is configured as the entry point. Appropriate access permissions are defined to allow users to access the website.

3: Upload Website Files

Terraform uploads the website files to the S3 bucket. Once uploaded, the bucket acts as the origin server and stores the static content reliably and cost-effectively.

4: CloudFront Distribution Setup

A CloudFront distribution is created using Terraform to improve performance and availability. CloudFront acts as a Content Delivery Network (CDN) that caches and delivers the website content from the nearest edge location to users.

5: Origin Configuration

The S3 bucket is configured as the origin of the CloudFront distribution. CloudFront retrieves the website files from the S3 bucket and distributes them globally, ensuring low latency and scalability.

6: Infrastructure Deployment using Terraform

The infrastructure is defined in Terraform configuration files (*main.tf*, *variables.tf*, *outputs.tf*). Terraform is used to automate the creation and configuration of AWS resources, ensuring consistency and easy management.

7: Website Access and Availability

After deployment, Terraform provides the CloudFront domain name as output. Users can access the website securely and globally through CloudFront. This architecture ensures scalability, reliability, automation, and high performance.

