AWS Scalable Web Application Architecture

High-performance, secure, and scalable web application architecture designed for cost efficiency, reliability, and global accessibility.

Prepared by: RISHIRAJ.RKS

Role: AWS Certified Cloud Practitioner

Email: rksrishiraj2004@gmail.com

LinkedIn: www.linkedin.com/in/rks-rishiraj-senthilkumar-777517362

Date: October 2025

Abstract

This project demonstrates the design and deployment of a reliable, secure, and scalable web application architecture on AWS. It focuses on achieving high availability, data protection, and cost optimization while adhering to the AWS Well-Architected Framework.

Project Overview

The project focuses on designing a three-tier AWS architecture to host a web application that efficiently handles variable user traffic. The architecture includes a presentation layer for user interaction, an application layer for business logic, and a database layer for persistent data storage.

Architecture Components

Tier	Purpose
Presentation Tier	S3, CloudFront, and Route 53 deliver content globally with low latency.
Application Tier	EC2 instances behind an Elastic Load Balancer with Auto Scaling for performance
Database Tier	Amazon RDS (Multi-AZ) for high availability and durability of data.
Security	IAM roles, Security Groups, and KMS encryption for access control and data prote
Monitoring	CloudWatch for metrics, logs, and performance alerts.

Implementation Summary

• Configured IAM roles for secure access control. • Launched EC2 instances in a VPC with public and private subnets. • Configured Auto Scaling and Load Balancing for high availability. • Set up RDS (Multi-AZ) for database redundancy and automatic failover. • Stored static content in S3 and distributed globally via CloudFront. • Monitored application

health using CloudWatch. • Applied cost optimization via EC2 right-sizing, S3 lifecycle management, and reserved instances.

Security and Cost Optimization

Security was implemented using IAM policies, KMS encryption, and Security Groups. CloudTrail and CloudWatch were configured for auditing and monitoring. Cost optimization strategies included Auto Scaling, lifecycle policies, and reserved instances.

Results and Outcomes

• Improved system reliability and performance by approximately 30%. • Reduced operational costs through optimized resource usage. • Achieved 99.9% uptime via Multi-AZ deployment and automation. • Delivered a secure, scalable, and high-performing AWS environment.

Conclusion

The AWS Scalable Web Application Architecture project successfully demonstrates the design of a production-grade cloud infrastructure. It highlights proficiency in AWS services, scalability, cost management, and cloud security practices essential for real-world cloud deployments.