Emirates Airlines: Solutions Architect Casestudy:

Case Study 1: Designing a Scalable E-commerce Platform

 ABC Enterprises is a well-established retail company with a strong physical presence in multiple locations. To expand their business and reach a wider customer base, ABC Enterprises has decided to launch an online e-commerce platform. As a principal solutions architect, your role is to design a scalable web application that can handle high traffic volumes, support concurrent user sessions, process a large number of transactions, and ensure a seamless shopping experience for customers.

 Scenario

ABC Enterprises is looking to create an online shopping experience that reflects the convenience and quality of their physical stores. They want to provide customers with a wide range of products, secure payment options, personalized recommendations, and a user-friendly interface. As the solutions architect, you are responsible for designing the architecture of the e-commerce platform to meet these requirements.

 Business Requirements

ABC Enterprises has identified key business requirements for the e-commerce platform, including scalability, performance and responsiveness, product catalogue and search functionality, and secure payment processing. The platform should be able to handle surges in traffic during peak seasons, deliver fast page load times, enable easy product search and filtering, and integrate with secure payment gateways.

 Your Task:

As the candidate, your task is to analyze the business requirements and design the solution architecture for the e-commerce platform.

 Consider the scalability, availability, performance, and security aspects while designing the architecture. Document your solution, explaining the various components, their interactions, and how they address the identified requirements. Include technical requirements, outlining the technologies, frameworks, and tools you would use to implement the architecture.

Prepare your solution in presentation mode (PowerPoint) and submit it before the interview. Note that you will have 20 minutes during the interview to present your solution to the panel. Consider the panel to be steering group of the project for which the solution architecture is prepared and presented.

**Document Title:** Multi-Region Web Application Architecture on AWS **Prepared by:** Tima Ali  
**Date:** October 2025 **Purpose:** To design and document a resilient, scalable, and highly available AWS architecture capable of serving users across multiple regions with automatic failover and minimal downtime.

**Deduced Requirements:**

1. Launch an ecommerce platform.
2. Multiple locations
3. Support concurrent user sessions - responsiveness
4. Scalable to handle high traffic volumes - scalability
5. Process a large number of transactions - Performance
6. Wide range of products
7. Secure payments
8. Personalized recommendations

**Overview**

This document describes the design and implementation of a **multi-region AWS architecture** for a modern web application. The goal is to ensure **high availability**, **fault tolerance**, and **low latency** for users globally.

The solution is deployed across **two AWS regions** **US East (N. Virginia)** and **EU (Ireland)** to achieve **active-passive redundancy**. Each region hosts identical infrastructure stacks including networking, compute, storage, and database components. Traffic management and failover are handled using **Amazon Route 53** and **AWS Global Accelerator** (optional enhancement).

Key objectives of the architecture include:

* Minimizing latency for global users.
* Maintaining application uptime during regional outages.
* Automating data replication and failover mechanisms.
* Ensuring security and compliance with best practices.

Design:  
  
A screenshot of a computer

AI-generated content may be incorrect.

**Considerations:**

**Overview:**

The multi-region web application is deployed in **two AWS regions** **Primary (us-east-1)** and **Secondary (eu-west-1)** each containing identical infrastructure stacks.  
The architecture follows a **highly available, fault-tolerant and scalable** design pattern, ensuring continuous service even if one region becomes unavailable.

Traffic distribution and health-based failover are managed by **Amazon Route 53**, which routes users to the healthiest region based on latency or availability.

**High Level Overview:**

**A. Global Layer**

| **Component** | **Description** |
| --- | --- |
| Amazon Route 53: | Provides global DNS resolution and latency-based or failover routing between regions. Monitors health checks to automatically redirect users to the secondary region during outages. |
| Amazon CloudFront: | Distributes static assets (images, CSS, JS) via a global content delivery network, reducing latency and offloading traffic from web servers. CloudFront origins are configured to point to the Application Load Balancer (ALB) in each region. |
|  |  |

**B. Regional Layer**

Each region (Primary and Secondary) includes identical infrastructure:

| **Component** | **Description** |
| --- | --- |
| VPC | Each region has its own isolated Virtual Private Cloud with public and private subnets across multiple Availability Zones for redundancy. |
| Internet Gateway (IGW) | Allows public subnet resources like the ALB to access the internet. |
| Application Load Balancer (ALB) | Distributes incoming traffic evenly across EC2 instances within the Auto Scaling Group. Integrated with health checks to ensure only healthy instances serve traffic. |
| Auto Scaling Group (ASG) | Ensures scalability and fault tolerance by automatically adding or removing EC2 instances based on demand. |
| EC2 Instances | Host the web application backend or API services. Configured for auto-deployment via Launch Templates or AWS Elastic Beanstalk. |
| Amazon RDS (Multi-AZ) | Primary database in us-east-1 with read replica in eu-west-1. Automatic failover to the replica in case of a regional outage. |
| Amazon S3 | Stores static files, backups, and logs. Cross-Region Replication (CRR) ensures data consistency between S3 buckets in both regions. |

**Connectivity & Routing**

* Route 53 is configured with Latency-Based Routing and Health Checks for the ALBs in both regions.
* In the event of a failure in the primary region, Route 53 detects the health check failure and reroutes traffic to the secondary region’s ALB automatically.
* Internal communication between application layers occurs through private subnets and security groups enforcing least privilege access.

**Multi-Region Failover Flow**

1. User request → Route 53 (evaluates latency & health).
2. Route 53 → CloudFront → Regional ALB (nearest healthy region).
3. ALB → EC2 (in ASG) → RDS.
4. If the primary region fails, Route 53 automatically routes to the secondary region.
5. Data synchronization occurs through S3 replication and RDS read replica promotion.