Modulo 9

Actividad 4 **Servicios de Bases de Datos en la Nube**

Nombre: Marco Neira

**1. Diagnóstico del entorno actual**

La empresa de educación online enfrenta varias limitaciones en su infraestructura **on-premise**:

* **Escalabilidad limitada:** el servidor físico no soporta el rápido crecimiento de usuarios.
* **Problemas de rendimiento y disponibilidad:** la base relacional centralizada genera lentitud e interrupciones en momentos de alta demanda.
* **Carga operativa alta:** el equipo de TI debe encargarse de parches, actualizaciones, backups y hardware.
* **Riesgos y costos elevados:** mantener servidores locales implica gastos de mantenimiento, reposición de hardware y riesgo de pérdida de datos ante fallas críticas.

**2. Propuesta de arquitectura en la nube**

Para resolver los desafíos, se recomienda una **arquitectura híbrida en AWS**:

1. **Base de datos relacional gestionada:**
   * **Amazon Aurora (compatible con MySQL/PostgreSQL)** para la aplicación principal (cursos, usuarios, evaluaciones).
   * Alta disponibilidad con **Multi-AZ** y escalado automático de lectura mediante **Aurora Read Replicas**.
2. **Base de datos NoSQL:**
   * **Amazon DynamoDB** para gestionar sesiones de usuarios, logs de acceso e interacciones en tiempo real.
   * Latencia de milisegundos y escalabilidad automática.
3. **Data Warehouse analítico:**
   * **Amazon Redshift** para análisis histórico y generación de reportes sin afectar la operación principal.
   * Integración con **AWS Glue** para ETL y con **Amazon QuickSight** para visualización.
4. **Flujo de datos propuesto:**
   * **Aurora/DynamoDB → AWS Data Pipeline / AWS Glue → Redshift.**

**3. Ventajas esperadas**

* **Escalabilidad:** Aurora y DynamoDB se ajustan automáticamente a la demanda.
* **Resiliencia y disponibilidad:** replicación en múltiples zonas de disponibilidad.
* **Seguridad:** encriptación en tránsito y en reposo, control granular con **IAM**, backups automáticos.
* **Menor carga operativa:** AWS gestiona parches, mantenimiento y escalado, liberando al equipo de TI.
* **Experiencia de usuario mejorada:** tiempos de respuesta más rápidos y sin interrupciones.
* **Soporte a analítica avanzada:** Redshift permitirá análisis masivos sin afectar al sistema productivo.

**4. Consideraciones técnicas**

* **Seguridad:**
  + Uso de **IAM Roles** para acceso seguro.
  + Encriptación con **KMS**.
  + Copias de seguridad automáticas en S3.
* **Escalabilidad:**
  + Aurora Auto Scaling para instancias de lectura.
  + DynamoDB con **On-Demand Capacity Mode**.
  + Redshift con escalado de nodos según carga.
* **Costos estimados (nivel general):**
  + Aurora: pago por hora de instancia + almacenamiento.
  + DynamoDB: pago por capacidad provisionada o por uso (lecturas/escrituras).
  + Redshift: pago por hora/nodo y almacenamiento gestionado.

**5. Buenas prácticas recomendadas**

* Separar ambientes: **producción, staging y pruebas**.
* Configurar **backups automáticos y snapshots**.
* Implementar **CloudWatch + AWS CloudTrail** para monitoreo y auditoría.
* Optimizar consultas en **Redshift** y configurar **WLM (Workload Management)**.
* Aplicar **control de costos en DynamoDB** (mediante auto scaling o límites de uso).
* Uso de **ciclo de vida en S3** para mover backups antiguos a Glacier y reducir costos.

**6. Conclusiones**

Con esta arquitectura en AWS, la empresa:

* Mejora su rendimiento y disponibilidad para usuarios activos.
* Escala automáticamente a medida que crece su comunidad.
* Incorpora analítica avanzada con Redshift, sin afectar operaciones en tiempo real.
* Reduce costos y riesgos de mantener infraestructura física.
* Optimiza la carga de trabajo del equipo de TI, que puede enfocarse en innovación y no en mantenimiento.