Module 2.14 Assignment

NTU_CE7

28 Aug 2024

CE7 Group 2 Members

1. Chua Lai Chwang
2. Dijay Kumar
3. Tan Yuan
4. Lovell Tan
5. Khai Razali
6. Shashipal Singh
7. Wong Teck Choy

What is Cloud Architecture Design – Cost Optimization?

Chat generated reference

Cloud architecture design focused on Cost Optimization, as outlined in the AWS Well-Architected Framework, emphasizes architecting workloads with the most effective use of services and resources to achieve business outcomes at the lowest price point.

Key Design Principles

- Practice cloud financial management to develop a culture of cost awareness
- Maintain expenditure and usage awareness to understand how choices impact costs
- Use cost-effective resources by selecting the right services and configurations
- Manage demand and supply resources to balance usage with actual demand
- Optimize over time by continuously refining cost management practices

Best Practices

- Right-size resources by making informed decisions about instance types and storage
- Utilize reserved and spot instances to take advantage of cost-saving pricing models
- Implement auto scaling to dynamically adjust resources based on demand
- Monitor and analyze costs using tools like AWS Cost Explorer and AWS Budgets
- Optimize storage costs by reviewing solutions and using lifecycle policies

Cloud Architecture Design – Cost Optimization Implementation

How do you implement cloud financial management?

The following steps can be taken to implement Cloud Financial management:

1. See (Measurement and Accountability)

- Consider what cloud services and products are in use and where costs are coming from. Establish usage and cost visibility.

1. Save (Cost Optimization)

- Establish cost optimizations; a good way is to utilize reserved and spot instances to take advantage of cost-saving pricing models. Consider implementing auto scaling to dynamically adjust resources based on demand.

1. Plan (Planning and Forecasting)

- Monitor and analyze costs proactively using tools like AWS Cost Explorer and AWS Budgets. Plan budgets for upcoming cloud projects and

1. Run (Cloud Financial Operations)

- Optimize storage costs by reviewing solutions and using lifecycle policies. Use tools to set threshold alerts and detect anomalies. Ensure proper ownership and access management practices.

How do you monitor usage and cost?

How we can leverage AWS tools to stay on top of usage and expenses:

Estimate Costs Before Launch

- AWS Pricing Calculator: Before implementing new services, use this tool to get a detailed cost estimate. This helps in setting a realistic budget and avoiding unexpected expenses.

Track and Analyze Costs Over Time

- AWS Cost Explorer: Visualize your spending patterns and track costs by service. This tool helps you understand past expenditures and forecast future costs, allowing for better resource management.
- Amazon QuickSight & Athena: Build a Cost and Usage Dashboard for clear, interactive visualizations of your spending. This helps in spotting trends and making informed financial decisions.

Set Budgets and Receive Alerts

- AWS Budgets: Configure alerts to notify you when your spending approaches predefined thresholds. This proactive approach helps you take corrective actions to manage costs effectively.

Optimize Resource Utilization

- AWS Trusted Advisor: Get actionable recommendations based on AWS best practices. Identify and eliminate underutilized resources or resize them to save costs.
- AWS Compute Optimizer: Analyze your EC2 instances, Lambda functions, and EBS volumes to receive tailored rightsizing recommendations. This ensures you're using the right resources for your needs, optimizing both performance and cost.

Effective Monitoring

- Regularly Review Cost Reports: Make it a habit to check your cost reports weekly or monthly.
- Leverage Automation: Use automated alerts and scaling policies to manage costs in real-time.
- Educate Your Team: Ensure that everyone involved understands how to use these tools effectively.

How do you evaluate new services?

Evaluating New Services: A Step-by-Step Approach

1. **Define Requirements**
    - **Business and Technical Needs**: Clearly identify and understand your specific needs and goals.
2. **Explore Service Options**
    - **Service Comparison**: Investigate AWS services that align with your defined requirements to find the optimal match.
3. **Analyze Costs**
    - **Cost Estimation**: Project and compare costs to ensure the service is cost-effective and fits within your budget.
4. **Conduct Performance Testing**
    - **Benchmarking**: Evaluate the service's performance under different scenarios to ensure it meets your performance criteria.
5. **Verify Security and Compliance**
    - **Standards Compliance**: Ensure the service adheres to necessary security protocols and regulatory requirements.
6. **Assess Integration and Compatibility**
    - **System Fit**: Check how well the service integrates with your current systems and processes.
7. **Evaluate Scalability and Flexibility**
    - **Growth Potential**: Determine the service's capability to scale and adapt to future needs.
8. **Consider Vendor Lock-in**
    - **Portability and Interoperability**: Assess the implications of vendor lock-in and the service's ability to work with other systems.
9. **Review Support and Documentation**
    - **Resources and Assistance**: Examine the support options and documentation available to ensure you have adequate resources for implementation.
10. **Implement Proof of Concept**
    - **Pilot Testing**: Run a controlled test to validate the service's suitability and performance.
11. **Monitor and Optimize**
    - **Ongoing Evaluation**: Use AWS tools to monitor the service's performance and make necessary adjustments for optimization.
12. **Make Final Decision**
    - **Presentation and Approval**: Compile findings, present them for stakeholder approval, and plan for full deployment.