**Cloud Computing (ITCS-6190) Project Deliverable 4**

Group No: 16

Team members:

Akhil Rayapati

Hemanth Anumolu

Mahesh Hasbi

Prashanth Sammanu

Saandeep Ippagunta

**Optimization, Scalability, and Deployment for Heart Attack Prediction Project**

**Introduction**

This document details Phase 4 of the Heart Attack Prediction Project, focusing on AWS cost analysis and optimization, pipeline scalability, and deployment strategies. The objective is to ensure the project is cost-effective, scalable, and efficiently deployed in the AWS cloud environment.

**AWS Cost Analysis and Optimization**

**Objective**: To optimize AWS spending related to the project without compromising on performance and reliability.

**1. Analyzing AWS Costs**

* **Methodology**: Employ AWS Cost Explorer to gain insights into the project's AWS spending pattern. Regularly monitor the monthly and daily costs, and analyze the usage reports for each service.
* **Action Steps**:
  + Periodically review the AWS Cost Explorer dashboard.
  + Identify high-cost services and assess if they align with project needs.
  + Investigate unexpected spikes in usage or cost.

A screenshot of a computer

Description automatically generated

**2. Cost Optimization Strategies**

* **Right-Sizing Resources**: Assess EC2 instances, RDS databases, and other AWS services used in the project for their utilization patterns. Resize them to match the actual workload requirements.
* **Reserved Instances and Savings Plans**: For consistent and predictable workloads, like database instances, consider purchasing Reserved Instances or Savings Plans.
* **Spot Instances**: Implement EC2 Spot Instances for any flexible, non-critical processing tasks like batch jobs, thereby reducing the compute cost significantly.
* **Auto Scaling**: Integrate Auto Scaling to dynamically adjust the capacity, ensuring cost-effective handling of workload fluctuations.
* **S3 Lifecycle Management**: Implement lifecycle policies to move older data to more cost-effective storage classes like S3 Standard-IA or Glacier.
* **AWS Trusted Advisor**: Regularly consult AWS Trusted Advisor to receive recommendations on how to reduce costs by eliminating unused or idle resources.

**Scaling the Pipeline**

**Objective**: To automate and scale the data processing pipeline for the Heart Attack Prediction project efficiently.

* **AWS Lambda for ETL Jobs**: Leverage AWS Lambda for handling ETL tasks, triggered by events such as new data uploads to S3 buckets. This serverless approach allows for handling data processing loads efficiently without managing servers.
  + **Automating ETL Pipeline**: Use Lambda functions to automate the ETL process, triggered when new data is available, ensuring the data is processed and stored in a timely manner.
  + **Integration with AWS Services**: Seamlessly integrate Lambda with other AWS services such as S3 for data storage and RDS or DynamoDB for database management.
* **AWS Step Functions**: Utilize AWS Step Functions for orchestrating complex workflows, particularly when the ETL process involves multiple steps or conditional logic.

**Deployment**

**AWS SageMaker for Model Deployment**

* **Objective**: To deploy the heart attack prediction model in a scalable, managed environment.
* **SageMaker Features**: Utilize SageMaker's capabilities for model deployment, including easy scaling, setting up real-time inference endpoints, and conducting A/B testing for different model versions.
* **Deployment Process**:
  + Prepare the machine learning model in a SageMaker-compatible format.
  + Configure and deploy the model on SageMaker endpoints.
  + Implement monitoring using SageMaker’s monitoring tools to ensure model performance and reliability.

**Alternative Deployment Options**

* **Amazon ECS/EKS**: For more granular control over the environment, consider deploying the model within containerized environments using Amazon ECS or EKS.
* **CI/CD Pipelines**: Establish a CI/CD pipeline using AWS CodePipeline and AWS CodeBuild for continuous deployment and integration of the model and associated applications.

**Monitoring and Logging**

* Implement Amazon CloudWatch for comprehensive monitoring and logging. This includes tracking application and model performance, setting alarms for unusual activity or errors, and logging for auditing and debugging purposes.

Conclusion

In Phase 4, we aim to optimize costs, ensure scalability, and deploy the heart attack prediction model effectively in the AWS cloud. This phase requires ongoing attention to adapt to changing requirements and usage patterns, ensuring the project remains cost-effective, scalable, and efficient.