# Capstone Project – Architecting Solutions on AWS

**Scenario:** You are working for a customer that runs their workloads on premises. Your customer has two workloads:

* A three-tier architecture composed of a frontend (HTML, CSS, JavaScript), backend (Apache Web Server and a Java application), and database (MySQL). The three-tier application hosts a dynamic website that accepts user traffic from the internet.
* A data analytics workload that runs Apache Hadoop. The analytics workload analyzes a massive amount of data that stored on premises and it also uses visualization tools to derive insights.

These components are currently running in the data center on physical servers. Currently, if a power outage occurred in the data center, all systems would be brought offline. Because of this issue (in addition to other benefits of the cloud), your customer wants to migrate all components to the cloud and, when possible, use AWS services to replace on-premises components.

**Solution**:

Graphical user interface, application

Description automatically generated

Application:

* We can use **AWS Elastic Beanstalk** for the front-end and back-end application layers. Elastic Beanstalk is a fully managed service that supports Java and Apache web server, making it a good fit for this project.
* For the database, we can use **Amazon RDS**, a managed relational database service that supports MySQL. We can use multi-AZ deployment for high availability.
* **Amazon Route 53** is a Domain Naming System web service. We can use Route 53 to map a custom domain name to our Elastic Beanstalk environment.
* **Amazon CloudFront** can be used to improve performance by caching content closer to the end-users, thereby reducing latency and load on the Elastic Beanstalk environment.

Data Analytics:

* The Data for analytics can be stored in **Amazon S3.**
* The data can be processed by **Amazon EMR** using Hadoop.
* The output can be stored in **Amazon S3** again for further analysis or use.
* **Amazon QuickSight** can be used to visualize the data processed by Amazon EMR to provide powerful visualizations and insights.
* **Amazon CloudWatch** can be used to monitor and troubleshoot all the AWS services.

**Cloud-Native vs. Lift and Shift Approach:**

This solution is cloud-native and has benefits like **improved scalability, reliability, and cost-efficiency**. For a lift and shift approach, we can use AWS services such as Amazon Elastic Compute Cloud (EC2) to host virtual machines, Amazon Simple Storage Service (S3) to store data, Amazon Relational Database Service (RDS) to host databases, and Elastic Load Balancing (ELB) to distribute incoming traffic among multiple instances. The lift and shift approach may not be the best choice here as it may limit the ability to take advantage of cloud-native technologies and optimizations. Additionally, the migrated application may still require significant maintenance and operational effort, reducing the potential benefits of moving to the cloud.