**Application Details**: Apache Web Server, Apache Tomcat application server with Active MQ and Oracle and MongoDB backend.

**SCOPE 1: To Reduce Efforts for migration as well for HA and Load balancing**

Step by step Approach for Migration:

1. Archive and take the backup on AWS S3

2. Create VPC

3. Create Template which is a JSON file that serves as a Blueprint to specify the configuration of AWS Resources that make up your infrastructure and application stack in Multi-AZs. Or we can select sample Pre-built template.

4. Run the Template in AWS Cloud formation and bring up Application.

5. For Database Migration use AWS DMS service which will migrate data from On-premises DB to AWS RDS or AWS EC2 instance.

6. You can use AWS SQS instead of Active MQ.

Used AWS Services: VPC, AWS RDS, AWS EC2, AWS S3, AWS Cloud Formation, VPC, AWS DMS, AWS ELB, AWS SQS

**SCOPE 2: To Reduce overall operational cost in AWS with less AWS services**

Step by step Approach for Migration:

1. Archive and take the backup on AWS S3

2. Create VPC and Multi-AZ on premises EC2 instance to bring up application server and webserver.

3. Use AWS EC2 Reserve instance for Database.

4. Use AWS Storage Gateway to create DB replicas across region for HA.

5. Use ELB for Load balancing.

Note: 1. Here for hosting application we are using on premises instance.

2. But the Database instances are on Reserved instance. Here only data is getting updated and written so tiny instance is enough for DB. Reserved instances are discounted and for fixed term.

Used AWS Services: VPC, AWS EC2, AWS S3, VPC, AWS Storage Gateway, AWS ELB, AWS SQS (Optional)