MODULE – 6

CASE STUDY DATABASE

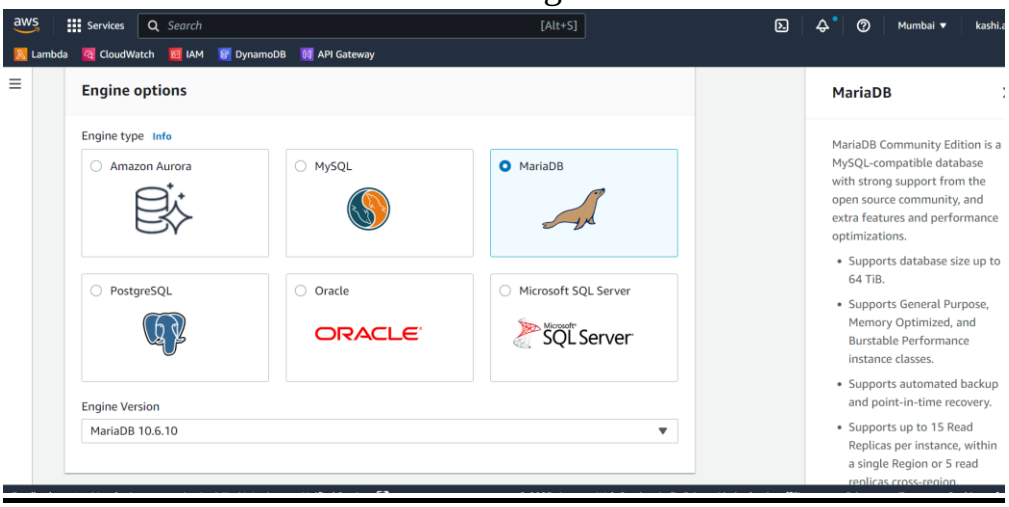
1. Build a highly scalable database and make sure that it's always available and accessible. Manage automated backup for this database with retention period of 2 days.
2. Build a Database architecture which lets your application read data from different regions
3. Establish a Database Architecture to process the data collected for near real-time analysis

As the user base of the application increases, your corporation observed a significant latency in the transfer of data to the branches. Your company realized that most of the time of the time, same kind of data is being fetched and sent to the branches. You are asked to:

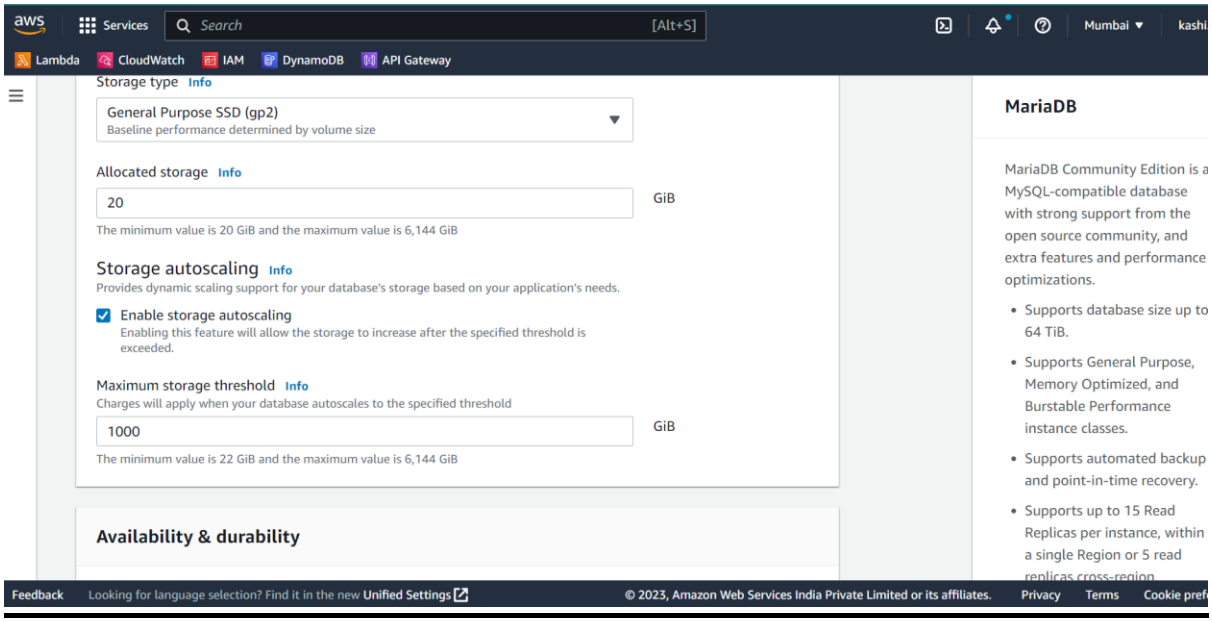
1. Find and implement the solution in order to resolve the latency issues

---

1. Highly Scalable Database and always available and accessible, RDS would be a solution with Autoscaling and MultiAZ.

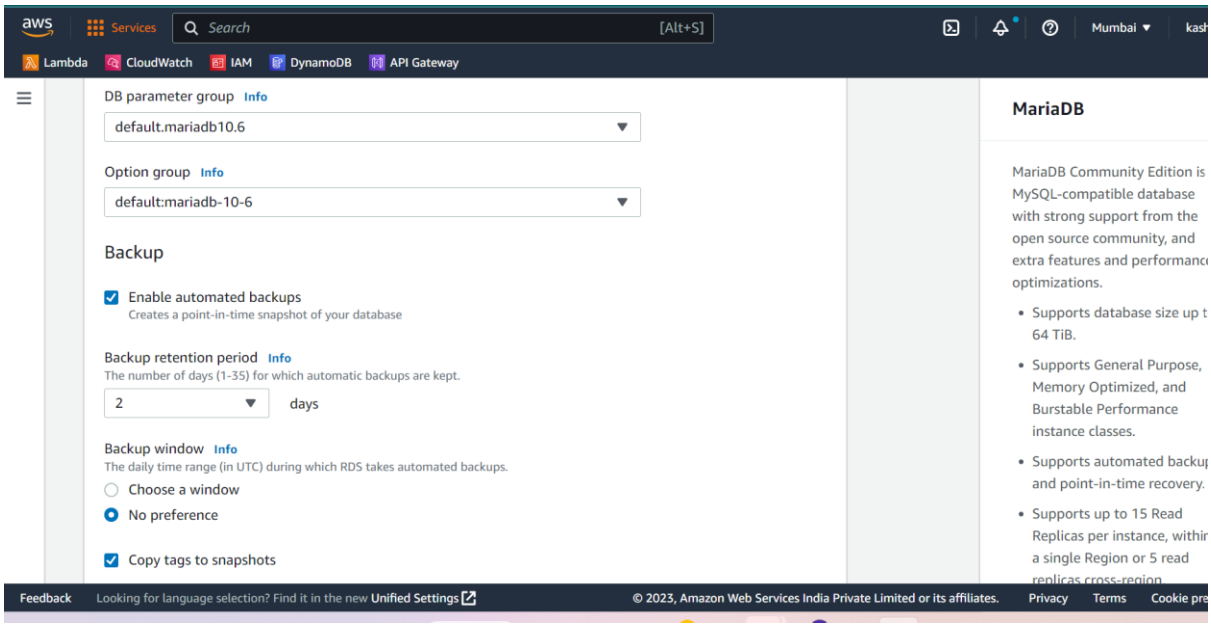


➔ MariaDB is a database engine is used with latest version

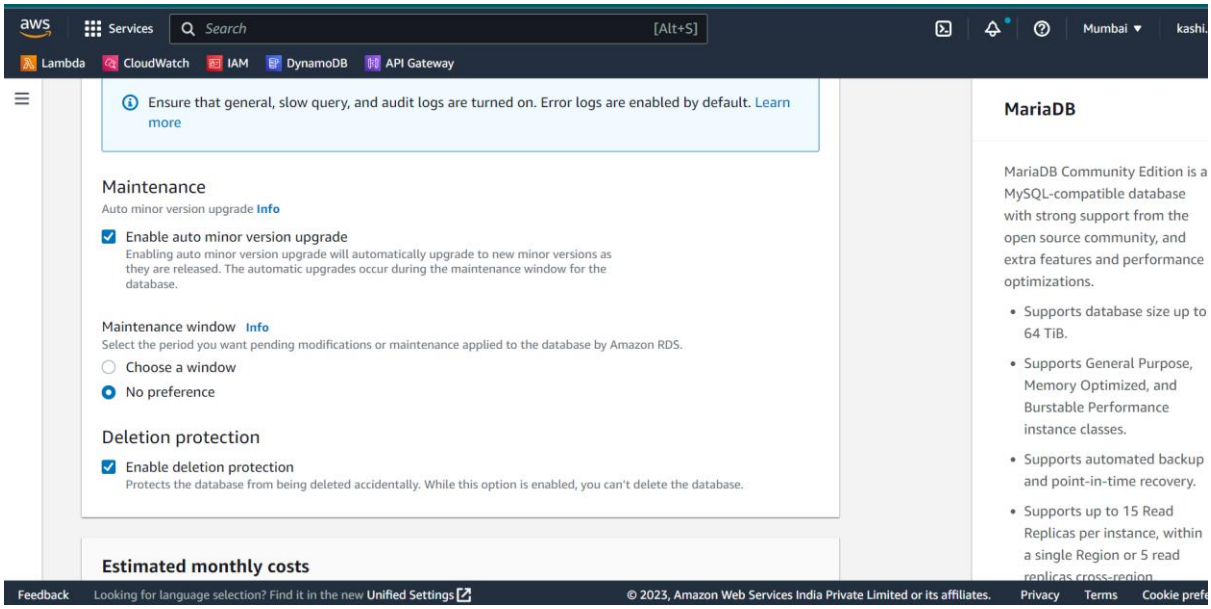


➔

➔ Always available when the traffic increases the traffic is served

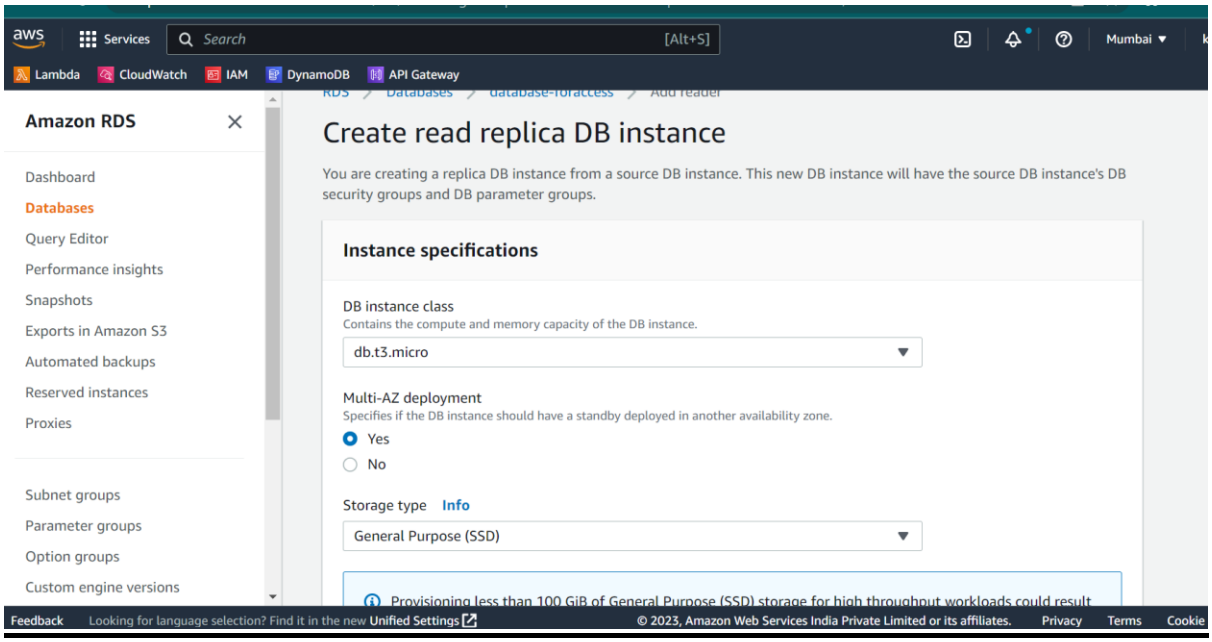


➔ The backup Retention period is set to 2 and backup is also enabled which will serve the purpose and avoid the DB from accidental deletion.

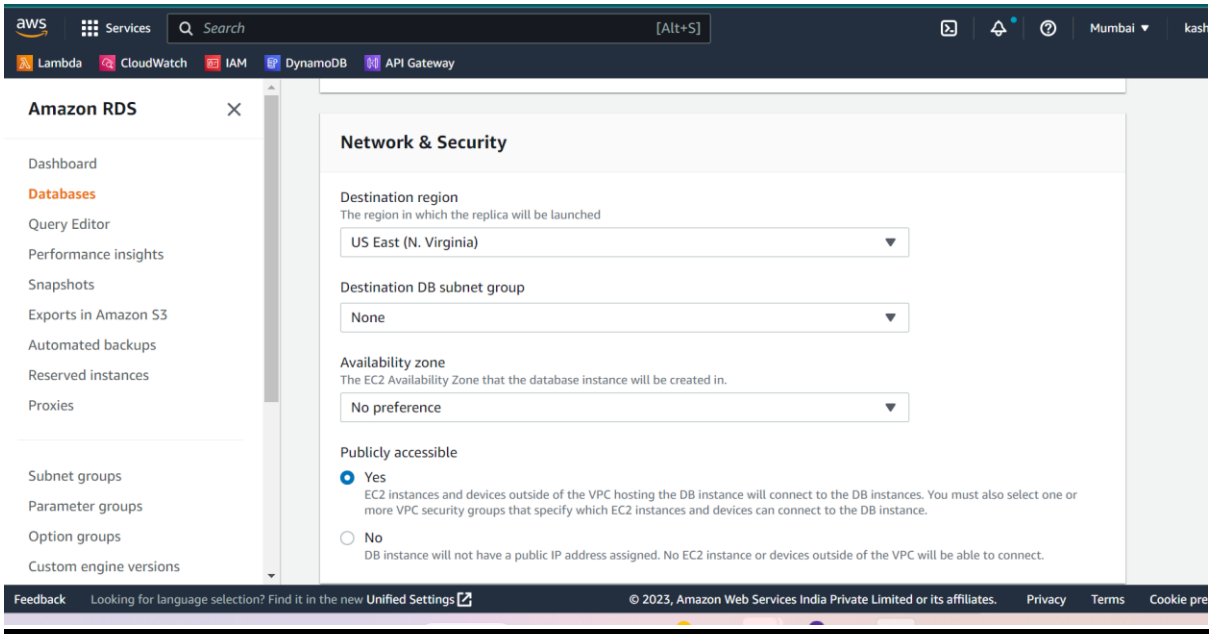


➔Even the window for maintenance can be set for modifications of the database. Deletion protection and upgrade is enabled

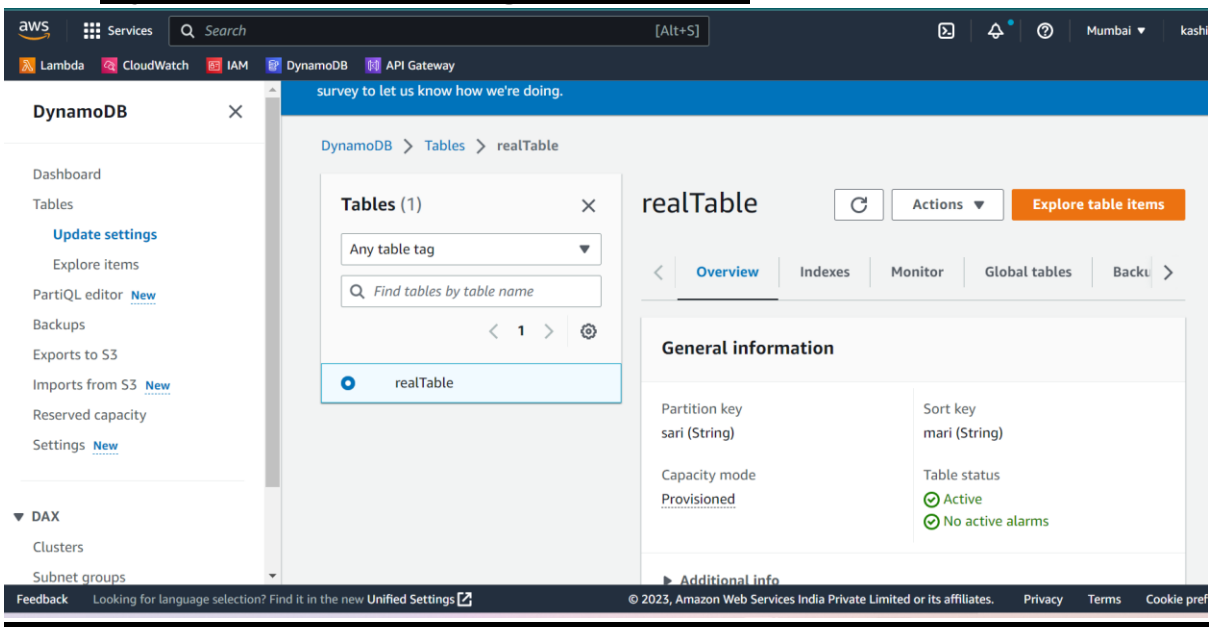
2. Read Replica



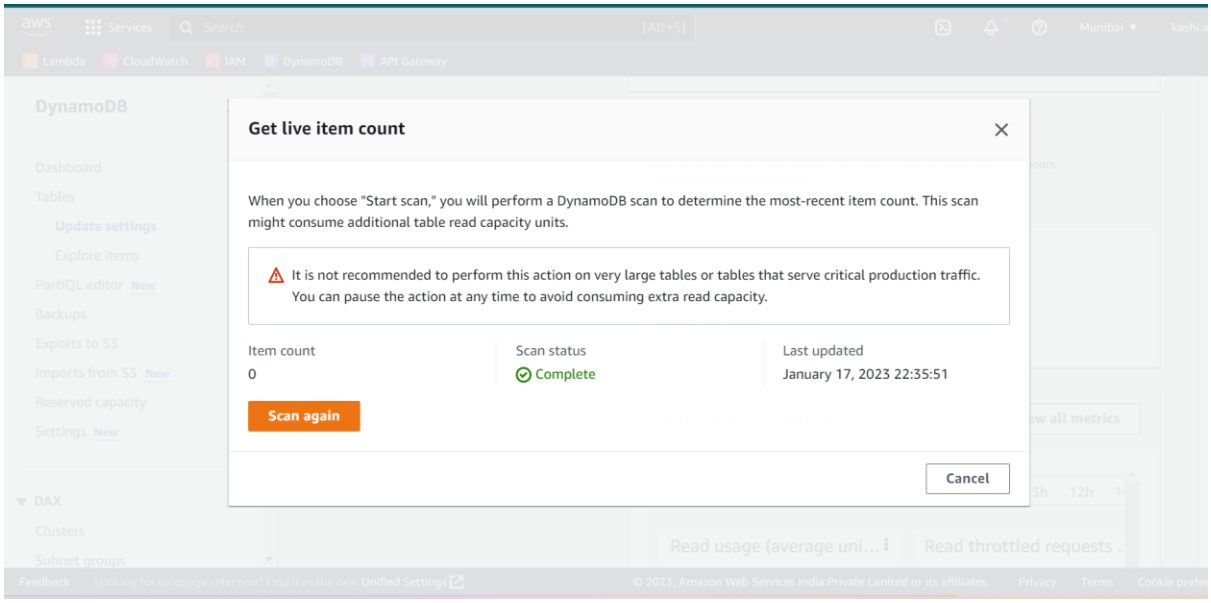
➔This will allow to read data from different region



3. DynamoDB would be great solution



➔ For real time data scan



- **Latency Issues**

- Network connectivity
- ELB configuration
- Memory utilization – One of the most common causes of web application latency is when most or all available physical memory (RAM) has been consumed on the host EC2 instance.
- CPU utilization – High CPU utilization on the host EC2 instance can significantly degrade web application performance and in some cases cause a server crash.

Solution:

1. CPU and storage-Checking CPU utilization. If it is constantly high or frequently it goes to full capacity it might be a good idea to look at your instance type and upgrade.

2. Storage-If the database makes a lot of IO requests, then we want to upgrade the storage of our DB instance from general-purpose SSD to provisioned IOPS SSD.

3. Debugging Slow queries