LSDE Coursework Part 2 –
Scaling the WordFreq Application

Rohan Anthony (2704500)

Task A: Installing and Verifying the WordFreq Application

To begin the WordFreq application setup, we first accessed the AWS Learner Lab (142821) and ensured that the account had the specified $50 credit. Task A involves deploying the required resources for the application and verifying the end-to-end flow for an uploaded text file. This includes creating the EC2 worker instance, configuring S3 buckets, setting up SQS queues, and a DynamoDB table.

The deployment consisted of an Ubuntu EC2 instance that runs the worker service, two S3 buckets used for storing uploaded and processed files, and two SQS queues used for holding jobs and the top 10 results of processed jobs. The setup script in the application also creates a DynamoDB table. Figures 2-4 show these resources created in the AWS account.

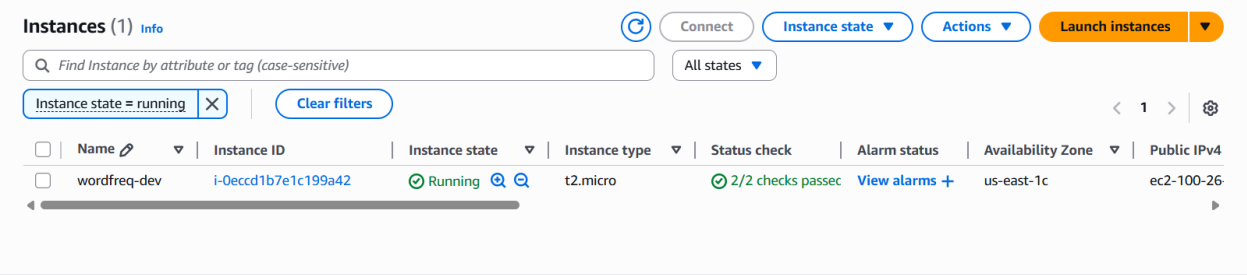

Figure 2: EC2 Instance

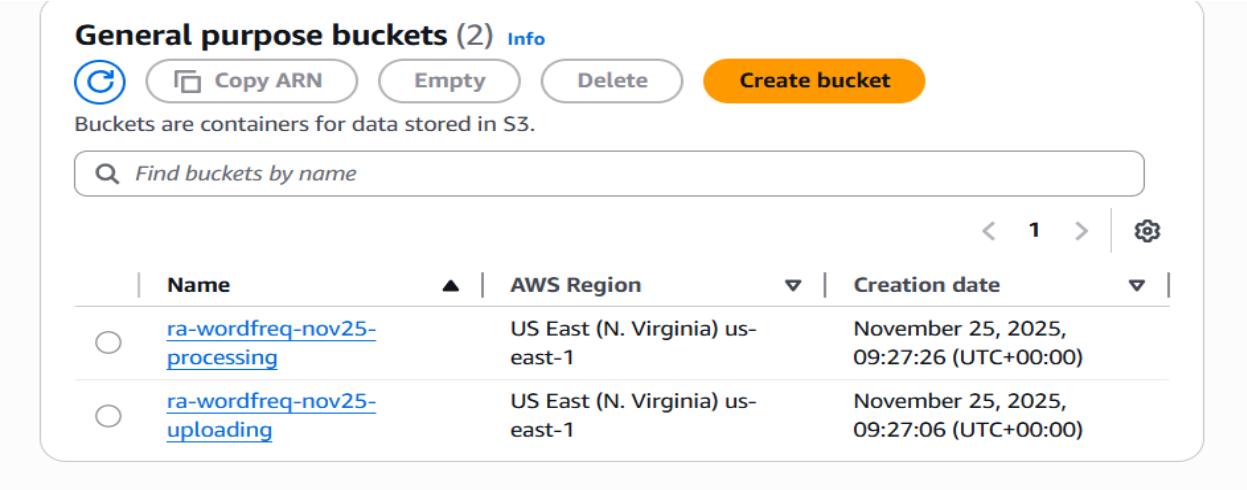

Figure 3: S3 Buckets

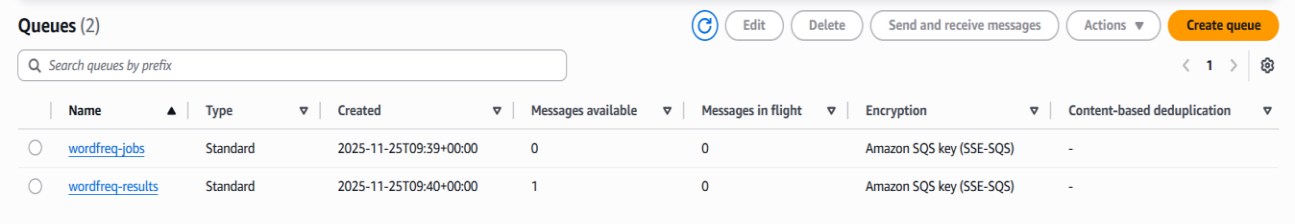

Figure 4: SQS Queues

The application operates through an event-driven workflow across these services. When a text file is uploaded to the processing S3 bucket, the bucket's configured S3 event notification automatically sends a message to the jobs SQS queue containing the file location. The worker service running on EC2 continuously polls this queue for new messages. Upon receiving a new message, the worker service retrieves the S3 file and processes it to compute the top 10 most frequent words in the file. After processing the worker publishes the result to the results queue and stores the output in the DynamoDB wordfreq table.

To verify the installation, we uploaded a text file to the processing S3 bucket. This triggered an S3 event and pushed a new message in the jobs queue. The EC2 worker detected the job, processed the file, and successfully wrote the output to the DynamoDB table. This confirms the end-to-end working of the application. Figures 5-6 show the EC2 worker logs and the new record in the DynamoDB table.

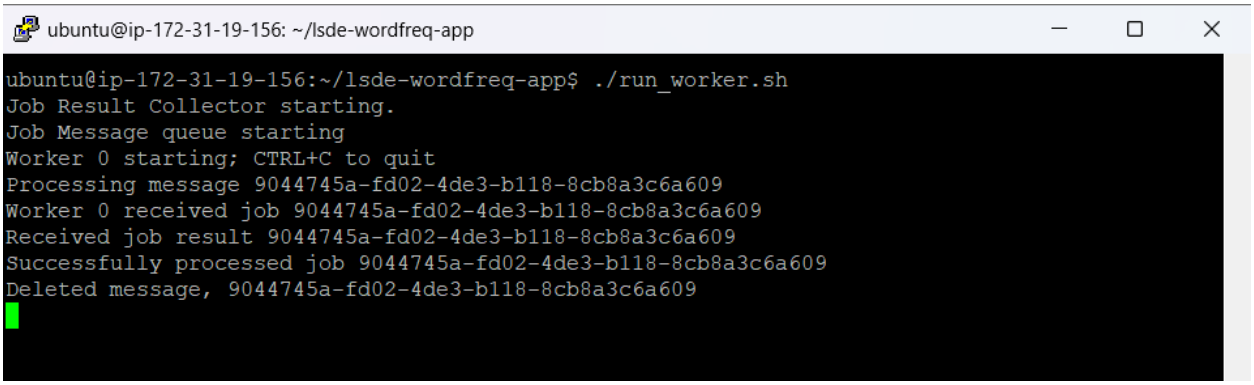

Figure 5: EC2 Worker

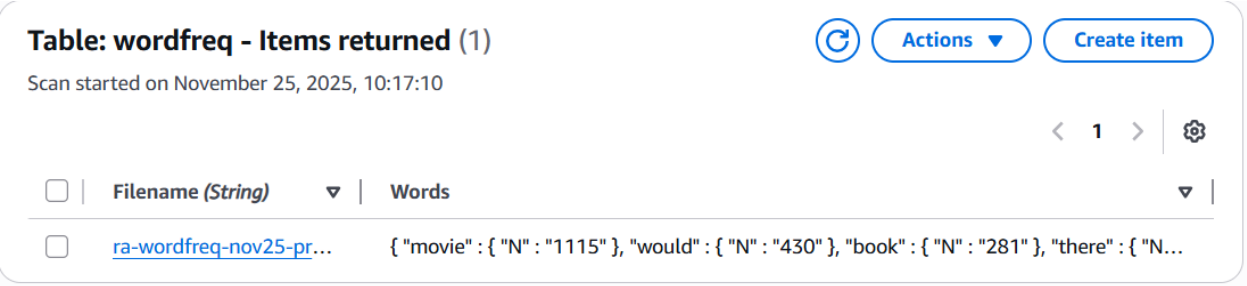

Figure 6: DynamoDB Table