AWS Fargate

AWS Fargate is a technology that you can use with Amazon ECS to run containers **without having to manage servers or clusters of Amazon EC2 instances**. We **don't need to provision, configure, or scale clusters of virtual machines** to run containers. This removes the need to choose server types, decide when to scale your clusters, or optimize cluster packing.

When you run your tasks and services with the Fargate launch type, you **package your application in containers, specify the CPU and memory requirements**, define networking and IAM policies, and launch the application. **Each Fargate task has its own isolation boundary and does not share the underlying kernel, CPU resources, memory resources, or elastic network interface with another task**.

You can check all the containers that are running on the following path:
AWS → ECS(us-west-2) → clusters → ImageConvertor-Fargate → tasks

You can check all logs for the previous containers in the cloudwatch.

Task CPU and Memory

Amazon **ECS task definitions** for Fargate require that you specify **CPU and memory** at the task level. Although you can also specify CPU and memory at the container level for Fargate tasks, this is optional. Most use cases are satisfied by only specifying these resources at the task level. The table below shows the valid combinations of task-level CPU and memory.

Simple Queue Service

Amazon Simple Queue Service (SQS) is a fully managed message queuing service that enables you to decouple and scale microservices, distributed systems, and serverless applications. SQS eliminates the complexity and overhead associated with managing and operating message oriented middleware, and empowers developers to focus on differentiating work.

The way we use SQS is, **we send the json data received in the API request to the lambda function which then stores that message in the Queue for container to use**. Container during runtime deletes that message after downloading the corresponding data from the s3.

You can check the Queue messages from the AWS console as well. Our Queue name is **dzi-image-convertor-queue.**

https://sqs.us-west-2.amazonaws.com/774447513285/dzi-image-convetor-queue

API Gateway

Amazon API Gateway is a fully managed service that makes it easy for developers to create, publish, maintain, monitor, and secure APIs at any scale. APIs act as the "front door" for applications to access data, business logic, or functionality from your backend services. Using API Gateway, you can create RESTful APIs and WebSocket APIs that enable real-time two-way communication applications. API Gateway supports containerized and serverless workloads, as well as web applications.

Our API gateway acts as an **interface for the MERN app to trigger lambda**. We are using API gateway in the us-east-1 region. You can send a POST request to the API gateway on the following URL:

https://79mu13jnz6.execute-api.us-east-1.amazonaws.com/Test