Platform Selection for Backend-as-a-Service

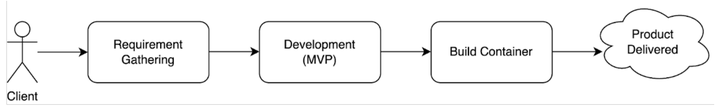
Jingtao Han, Xiaoan Yang and Vrushal Chaudhari
Khoury College of Computer Sciences, Northeastern University

Introduction

Our startup, aiming to attract 1 million users in six months, offers a scalable Backend-as-a-Service solution. We seek a hosting platform that supports our growth and technical needs for seamless, high-quality service delivery.

Among so many available options SPIDERS (Serverless, PaaS, IaaS, DaaS, and SaaS), Containers (CaaS), and Functions (FaaS), We decided to Focus on FaaS and CaaS due to

I. Ease of Implementation
II. Auto-Scale Functions/ Managed for us
III. Cheaper on Cost



Evaluation Methodology

High Availability: Evaluate the reliability and uptime guarantees, crucial for maintaining service continuity and customer satisfaction. We use failure rate to measure the availability of our service

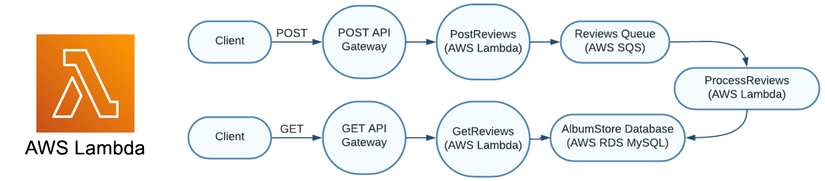
Performance Metrics: Include critical performance indicators such as percentile response times, scalability under load

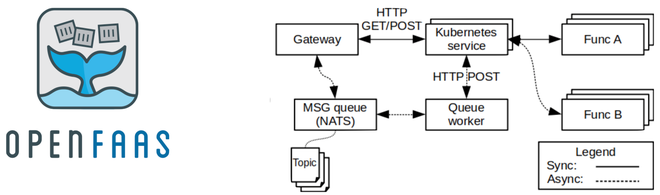
LOCUST

k6

Function-As-A-Service

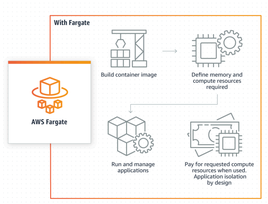
System Architecture



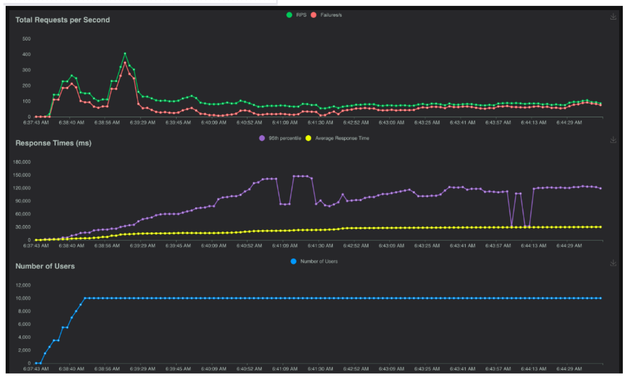


OpenFaas - More Flexibility
OpenFaaS can be deployed on any Kubernetes cluster
More fine-grained control over scaling parameters
OpenFaaS allows more direct control over gateway configuration within the Kubernetes environment.

Container-As-A-Service

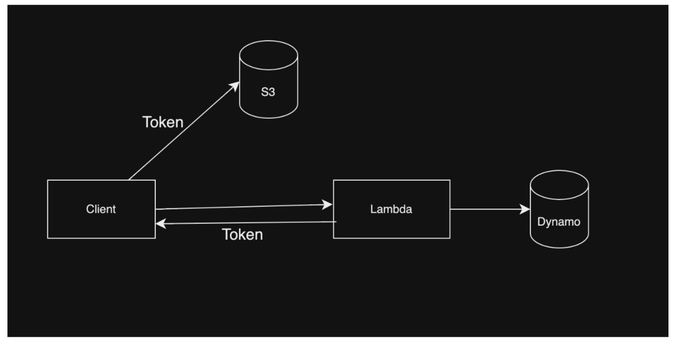


Fargate is Managed CaaS from AWS.
We found it not suitable to our application,
Under heavy workload Fargate is rate limiting requests.
OpenFaaS seems to be a way to go



File Uploading

The client-side computing resources are not as heavily utilized. Therefore, we could consider shifting the file upload operations to the client side.



Future Work

1. Divide the file into segments to facilitate resumption from the last known point.

2. Compute the hash value of a file to enable file sharing and deduplication, Ensure secure file uploading on AWS S3.

3. Compare on following points
I) Ease of Implementation: Assess the complexity involved in deploying and managing our services on each platform.
II) High Availability: Evaluate the reliability and uptime guarantees, crucial for maintaining service continuity and customer satisfaction.
III) Time to Market: Determine the speed with which we can deploy our product on each platform, impacting our responsiveness to market demands.