

LÜTFİ USLUER

PORTFOLIO

Github: <https://github.com/usluerL/usluerL>

Leetcode: My Profile: <https://leetcode.com/user0250q/>

Linkedin: <https://www.linkedin.com/in/lutfiusluer/>

Medium: <https://luusluer.medium.com>

INTRODUCTION:

In this portfolio, you will find 6 projects in a variety of areas, including RESTful services, microservice solutions, event-driven architectures, cloud computing, containerization, DevOps, CI/CD and so on:

1 CINEPHILL: A RESTful service that allows users to search for movies and view information about them using The Movie Database (TMDB) API

2 IMAGE DETECTION/RECOGNITION SYSTEM: A Spring Boot application that uses various AWS technologies -including NLP/AI services to store, detect and recognize images/videos with funny features.

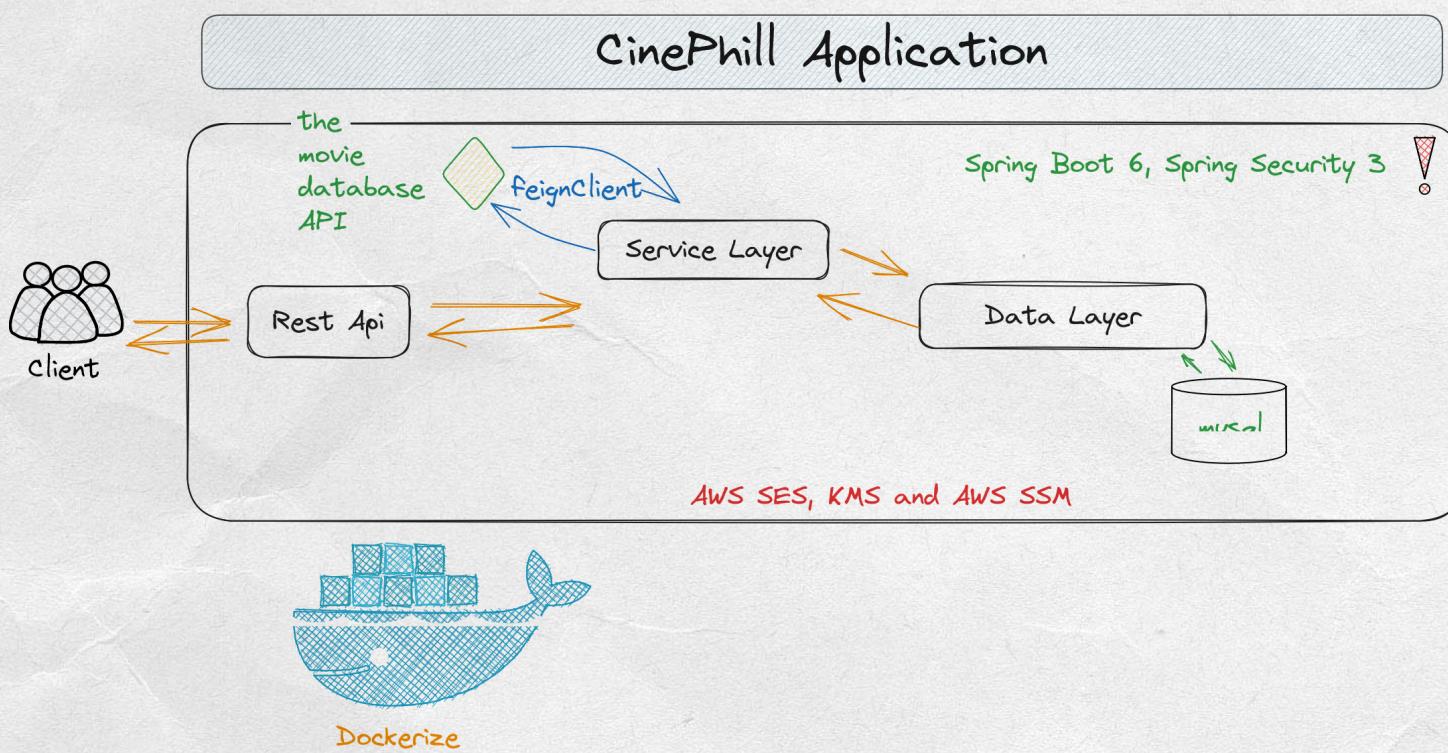
3 IMPLEMENTING CI/CD WITH CODEPIPELINE AND AWS ECS AND FARGATE: A project that demonstrates how to implement continuous integration and continuous delivery (CI/CD) with CodePipeline and AWS Serverless features and containerization technologies.

4 PERFORMANCE, MONITORING, LOGGING, AND DISTRIBUTED TRACING WITH AWS: a project that shows how to use AWS services to monitor and troubleshoot performance issues and implement distributed tracing

5 TWEETSTREAMER: An Event Driven – Microservice Application with Apache Kafka, and ELK Stack using Tweeter API.

6 BANKING APPLICATION: A Containerized microservices application built with Docker and Kubernetes.

CINEPHILL



DESCRIPTION: CinePhill is a movie search application that allows users to search for movies and TV shows by category, time frame, and rating. Users can also search for a movie's credits and cast information by entering the movie ID. Additionally, users can search for a movie based on a query parameter and customize their category settings to receive notifications for new TV show releases based on their criteria.

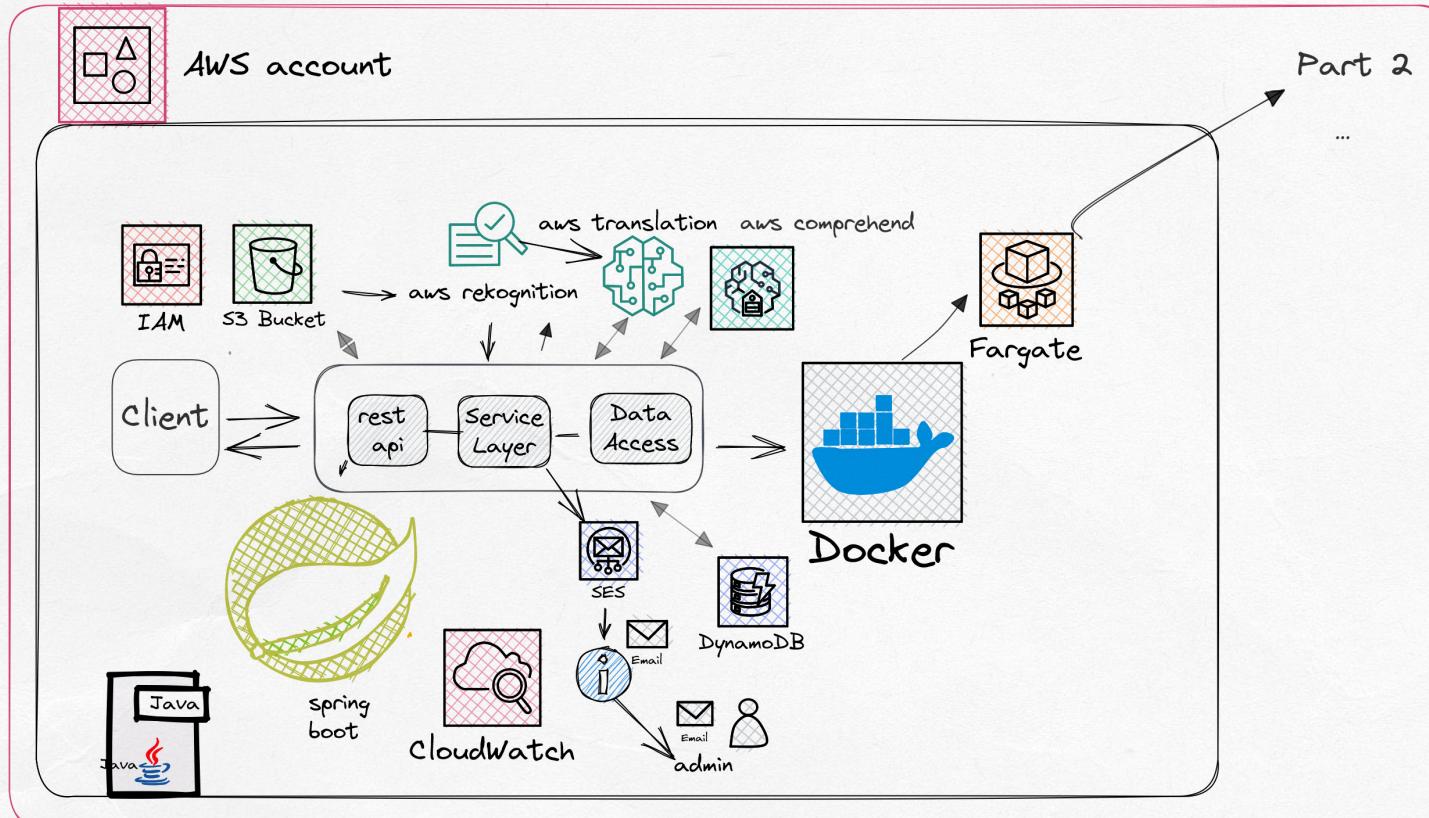
Tech Stack and features: Spring Boot 6 and Spring Security 3, AWS SSM, KMS, SES and The Movie Database API (TMDB).

CODE REPOSITORY: <https://github.com/usluerL/cinephill-public>

DOCUMENTATION: <https://github.com/usluerL/cinephill-public/blob/main/CinePhill-documentation.pdf>

IMAGE/VIDEO DETECTION & RECOGNITION SYSTEM

Building a Scalable Image/Video Detection & Recognition System with Spring Boot / Aws



DESCRIPTION: An image/video storage, detection and recognition application including some fun features like: "identify-me", "find-similarity", "face-details", and "celebrity look-alike". Interacted with various AWS services, including over 20 AWS services covering areas from development, monitoring, and performance to deployment, AI/NLP and some serverless features like Fargate.

TECH STACK AND FEATURES: AWS SDK for Java, AWS Comprehend, AWS Translation, AWS Rekognition, S3, Spring Boot, Docker.

CODE REPOSITORY: <https://github.com/usluerL/image-recognition-system>

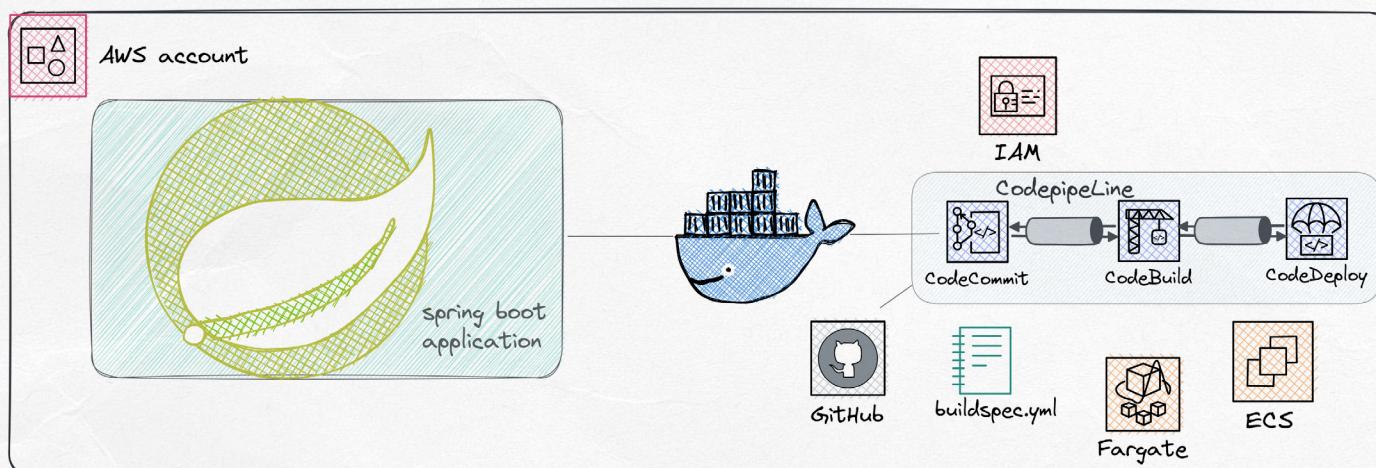
DOCUMENTATION: <https://medium.com/cloud-native-daily/building-an-image-video-detection-recognition-system-with-spring-boot-and-aws-i-991de56aaf5c>

IMPLEMENTING THE CI/CD WITH CODEPIPELINE AND ECS

DESCRIPTION: This is the CI/CD implementation of image/video detection-recognition application using AWS CodePipeline and ECS, Fargate.

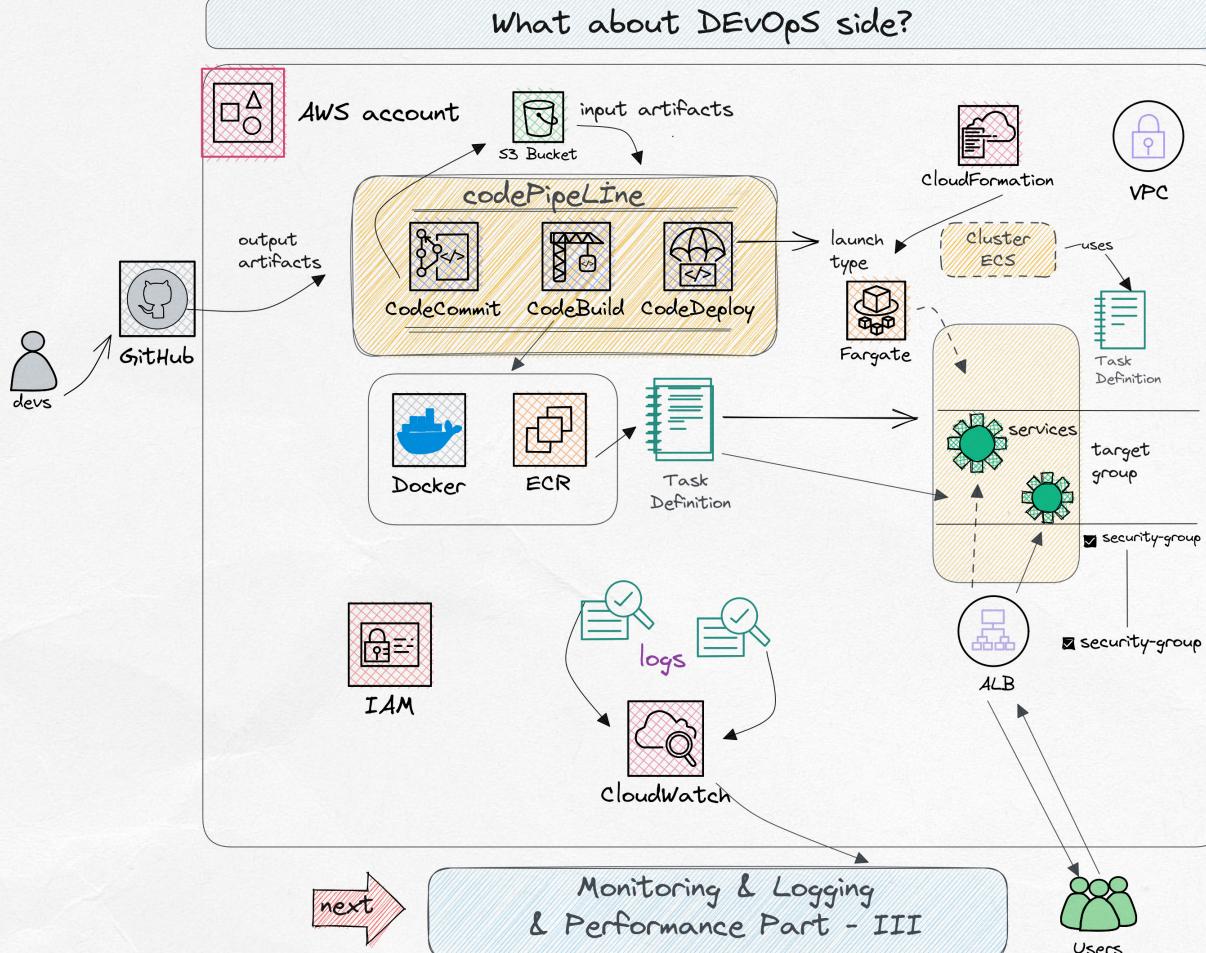
CODE REPOSITORY AND DOCUMENTATION:
<https://medium.com/cloud-native-daily/building-a-scalable-image-video-detection-recognition-system-with-aws-part-2a-implementing-the-763a217e2662>

TECH STACK AND SERVICES: CodePipeline, AWS Serverless, Elastic Container Service, Docker, ECR, AWS ALB.



PERFORMANCE, MONITORING, LOGGING, AND DISTRIBUTED TRACING WITH AWS

What about DEvOps side?



DESCRIPTION: This is the last part of the image/video detection-recognition application project: DevOps side of the application. In this section, performance, Monitoring, Logging, and Distributed Tracing are covered.

CODE REPOSITORY AND DOCUMENTATION:

<https://luusluer.medium.com/part-3-performance-monitoring-logging-and-distributed-tracing-in-aws-cff00b983fc1>

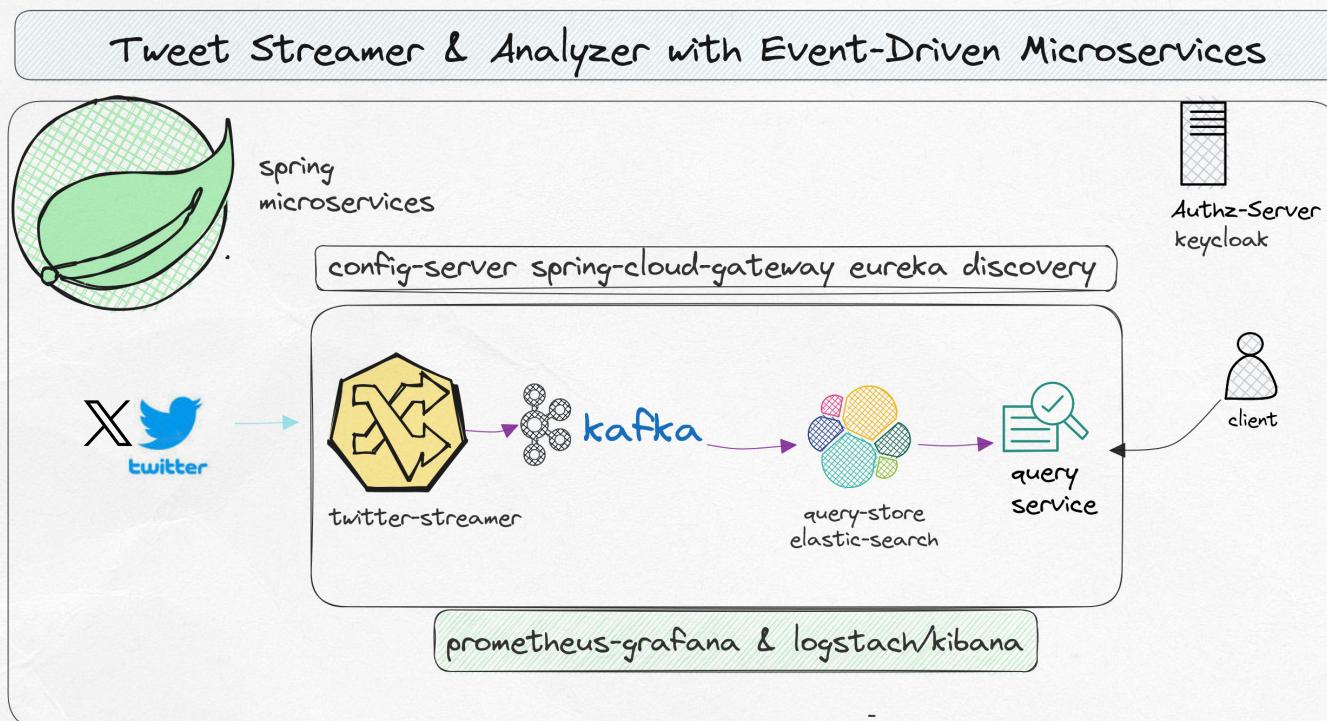
TECH STACK AND SERVICES: CloudWatch, CloudTrail, Xray, EventBridge, Spring AOP.

TWEETER STREAMER APPLICATION

DESCRIPTION: An event-driven microservices solution designed to stream tweets based on specific keywords. Leveraging Apache Kafka and the ELK stack, it efficiently indexes and searches the tweets for data management and analysis.

CODE REPOSITORY: <https://github.com/usluerL/twitter-streaming-project>

TECH STACK AND SERVICES: Spring Cloud, Docker, Kubernetes, Apache Kafka, ELK



BANKING DEMO APPLICATION:

DESCRIPTION: This is a demo banking application consisting of 5 microservices that have been Dockerized.

This project is focused on containerization and includes Docker-compose files for the development, production, and default environments.

CODE REPOSITORY: <https://github.com/usluerL/banking-microservices>

DOCUMENTATION: <https://github.com/usluerL/banking-microservices/blob/main/README.md>

TECH STACK AND SERVICES: Java, Spring Cloud, Docker

THANK YOU...

 <https://github.com/usluerL/usluerL>

 <https://www.linkedin.com/in/lutfiusluer/>

 <https://luusluer.medium.com>

 lu.usluer@gmail.com

 +90 505 093 43 46