# **Cloud DevOps Engineer Assignment**

## **Overview**

This repository contains the CI/CD pipeline configuration for automating the deployment of a simple web application to AWS ECS (Elastic Container Service) using GitHub Actions. The pipeline includes steps for building a Docker image, deploying it to ECS, performing integration tests, and handling rollbacks in case of failures.

## **Problem Statement**

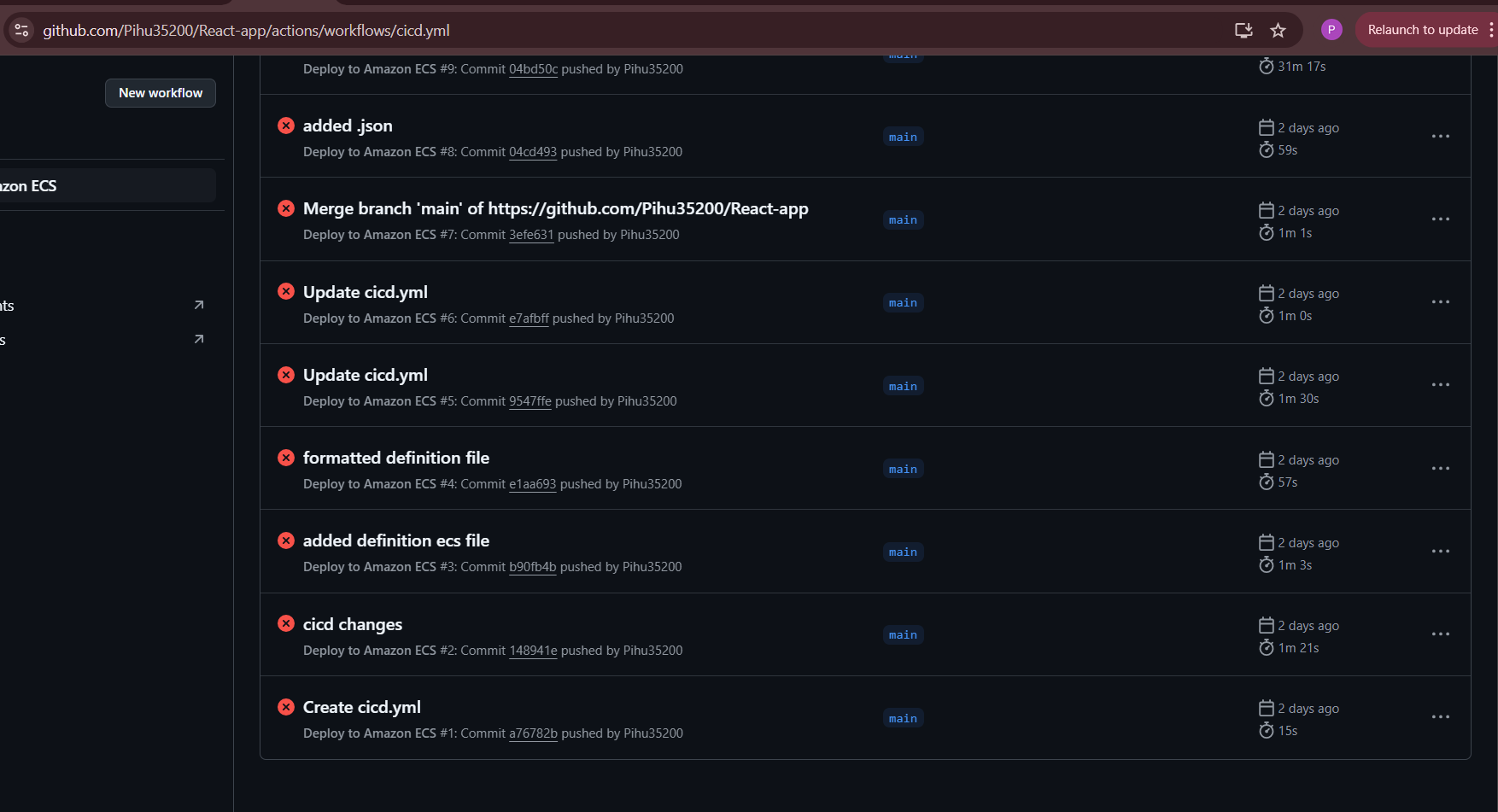
The task is to configure a GitHub Actions workflow to automate the following steps:

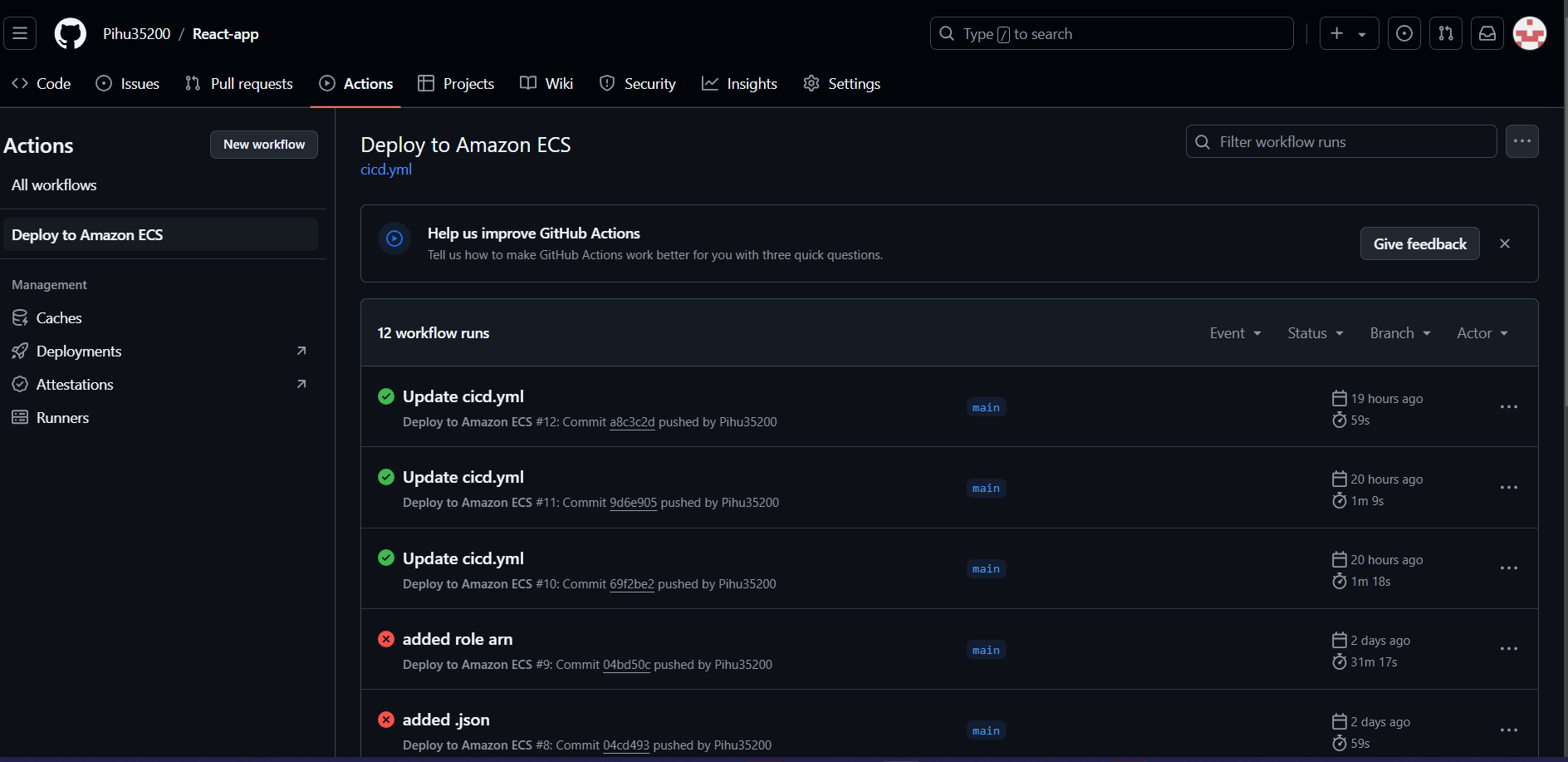
1. Checkout code from the GitHub repository.
2. Build an optimized Docker image with a multistage build to serve with NGINX as a reverse proxy.
3. Push the Docker image to Amazon ECR (Elastic Container Registry).
4. Deploy the Docker image to AWS ECS.
5. Perform integration tests and implement rollback functionality in case of failure.

## **Pipeline Workflow**

The GitHub Actions workflow for this assignment is defined in .https://github.com/Pihu35200/React-app/blob/main/.github/workflows/cicd.yml

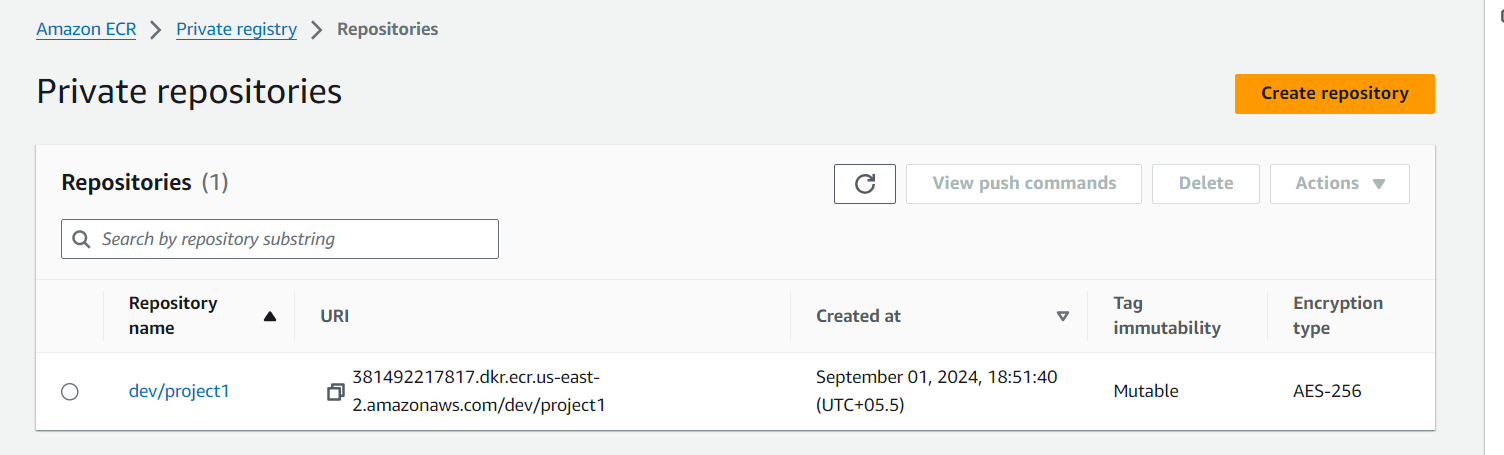
OUTPUTs:



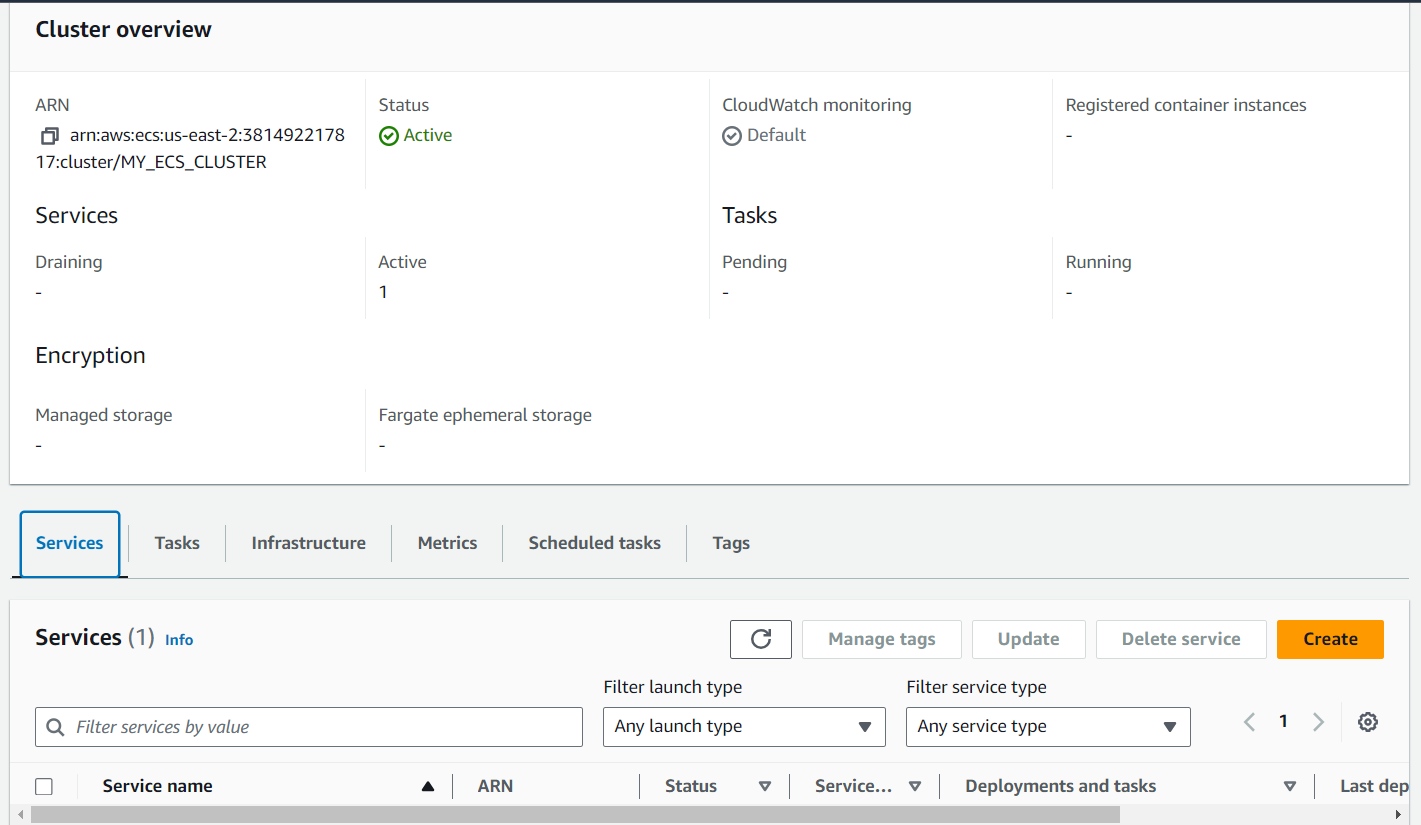


The workflow includes the following jobs:

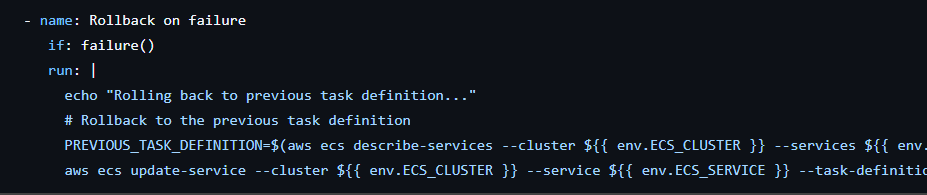
1. **Checkout Code**: Checks out the code from the GitHub repository.
2. **Build Docker Image**: Uses a multistage Docker build to create an optimized image served by NGINX.
3. **Push Image to ECR**: Pushes the built Docker image to Amazon ECR.



1. **Deploy to ECS**: Deploys the Docker image to AWS ECS.

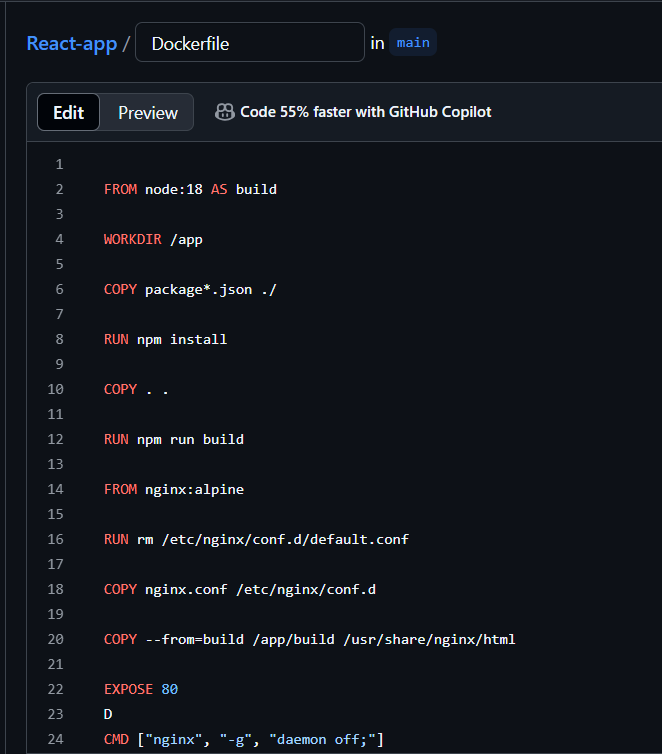


1. **Integration Tests**: Runs integration tests on the deployed application.
2. **Rollback**: Rolls back the deployment if the integration tests fail.



## **Setup Instructions**

1. **Repository Setup**: Fork or clone this repository to your local machine.
2. **Configure GitHub Secrets**: Set up the required secrets in GitHub repository:
   * AWS\_REGION: us-east-2
   * ECR\_REPOSITORY: dev/project1
   * ECS\_SERVICE: dev
   * ECS\_CLUSTER: MY\_ECS\_CLUSTER
   * ECS\_TASK\_DEFINITION: MY\_ECS\_TASK\_DEFINITION.json
   * CONTAINER\_NAME: reactapp
3. **Secrets and variables:** The AWS\_ACCESS\_KEY\_ID: AWS\_SECRET\_ACCESS\_KEY are saved here.
4. **Dockerfile**: Dockerfile in the root of repository that defines the multistage build process.



1. **ECS Configuration**: Ensure that ECS is properly configured with the necessary task definitions and service settings.

## **Usage**

1. **Push Code Changes**: Push your code changes to the GitHub repository. This will trigger the GitHub Actions workflow.
2. **Monitor Pipeline**: Navigate to the "Actions" tab in your GitHub repository to monitor the progress of the pipeline.
3. **Review Logs**: Check the logs for each job in the workflow to verify successful execution or diagnose issues.
4. **Integration Tests**: Review the results of the integration tests to ensure that the deployment is functioning correctly.

## **Snapshots**

Please include snapshots of each job and step from the GitHub Actions pipeline. These snapshots will demonstrate the successful execution of the pipeline.

## **Repository URL**

Provide the URL to your GitHub repository here:

https://github.com/Pihu35200/React-app/tree/main

