

GitHub Actions

ISC 629: Computing Ecosystems
School of Computing
University of South Alabama
September 20, 2022

Conducted by Harith Warnakulasooriya

What is Developer Workflow?

“The developer workflow typically involves writing code, executing automated tests, building the application, and running the app locally. In most cases, developers repeat these steps throughout the day, creating a development cycle. “

(John Harris, 2022)

What is Github?

GitHub is a web-based interface that uses Git, the open-source version control software that lets multiple people make separate changes to web pages at the same time. As Carpenter notes, because it allows for real-time collaboration, GitHub encourages teams to work together to build and edit their site content

(Digital.gov, 2022)

What is “Github Actions”?

GitHub Actions is a continuous integration and continuous delivery (CI/CD) platform that allows you to automate your build, test, and deployment pipeline

(GitHub.inc, 2022)

Terminologies

1. Events

An event is a specific activity in a repository that triggers a workflow run

2. Jobs

A job is a set of steps in a workflow that executes on the same runner

3. Actions

An action is a custom application for the GitHub Actions platform that performs a complex but frequently repeated task

4. Runner

A runner is a server that runs your workflows when they're triggered.

John Harris. (2022). VMware.inc. <https://tanzu.vmware.com/developer/guides/dev-workflow/>

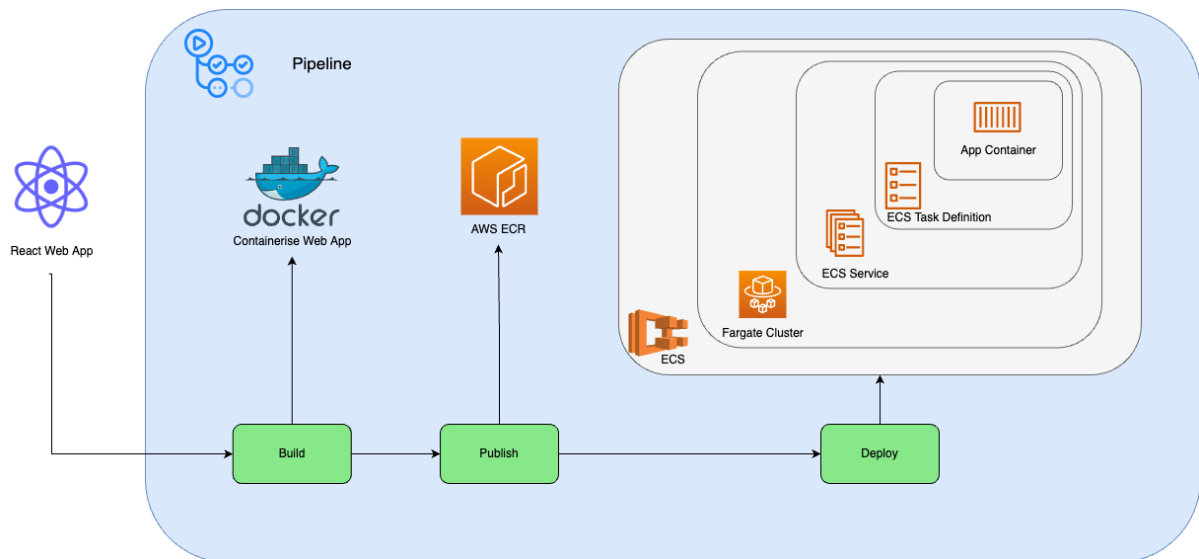
Nick-mon-1, (2022). Digital.gov. <https://digital.gov/resources/an-introduction-github/>

GitHub.inc, (2022). <https://docs.github.com/en/actions/learn-github-actions/understanding-github-actions>

Requirements for this workshop

1. You need to have a GitHub account
2. You need to have a AWS account

Our Pipeline



Steps

1. Create GitHub repository called “Github-Actions”
2. Import the code from
<https://github.com/kolithawarnakulasooriya/GitHub-Workshop.git>
1. Goto your AWS console
2. Create an ECR repository
 - a. Goto repositories tab
<https://us-east-2.console.aws.amazon.com/ecr/get-started>
 - b. Click on the “create repository” tab
 - c. Set repository name as “ecr-github-actions-web-app”
 - d. Click on “Create repository” button
3. Create ECS Cluster
 - a. Goto “Amazon Elastic Container Service”
<https://us-east-2.console.aws.amazon.com/ecs/v2/clusters>
 - b. Click on “Clusters” tab
 - c. Click on the “Create cluster” button

- d. Set your cluster name as “ecs-github-actions-web-app”
 - e. Configure your VPC and Subnet
 - f. Click on “Create” button
 - g. Wait few seconds till creating the cluster
4. Create task definition
 - a. Click on the “Task definitions” tab
 - b. Click on the “Create new task definition” button to create task definition
 - c. Set the task definition name as “task-definition-github-actions-web-app”
 - d. Set container name as “react-web-app”
 - e. Goto ECR repository and copy the ecr image uri of “ecr-github-actions-web-app” to the clipboard
 - f. Place the image uri by above copied image uri
 - g. Set the port mapping as TCP “80”
 - h. Click on “Next” button and create the task definition
 - i. Wait few seconds till creating task definition
5. Create the service
 - a. Click on your cluster “ecs-github-actions-web-app”
 - b. Click on the “Services” Tab
 - c. Click on the “Deploy” button to deploy a service
 - d. Select “task-definition-github-actions-web-app” as task definition family
 - e. Set the name as “ecs-service-github-actions-web-app”
 - f. Wait few seconds till creating the service
6. Create deployment script
 - a. Goto task definition
 - b. Copy the JSON script
 - c. Goto your “Github-Actions” github repository
 - d. Create new file called “.aws/task-definition.json”
 - e. Paste your task definition json content here and commit
7. Create Github Action
 1. Configure your github action
 - a. Goto “Actions” Tab
 - b. Search “Deploy to Amazon ECS” template
 - c. Set the configurations

AWS_REGION: us-east-2
ECR_REPOSITORY: ecr-github-actions-web-app
ECS_SERVICE: ecs-service-github-actions-web-app
ECS_CLUSTER: ecs-github-actions-web-app
ECS_TASK_DEFINITION: .aws/task-definition.json
CONTAINER_NAME: react-web-app

- d. Commit the changes
2. Set your credentials
 - a. Goto your AWS console
 - b. Goto IAM console
<https://us-east-1.console.aws.amazon.com/iamv2/home#/home>
 - c. Click on “Manage access key” button
 - d. Create new access key
 - e. Goto your github repository
 - f. Click on settings tab
 - g. Goto Secrets/actions view
 - h. Create new two repository secrets called
AWS_ACCESS_KEY_ID
AWS_SECRET_ACCESS_KEY
3. Rerun your deployment scripts
 - a. Goto Actions tab
 - b. Click on the last run action
 - c. Rerun the action
 - d. Wait few minutes till the service get deployed
4. Run the web app
 - a. Goro the task view
 - b. Goto the networking tab
 - c. Open the web app using public IP