# Servian Techchallenge Application – CI/CD pipeline

Pipeline created in circleci for deploy Techchallenge application (golang) in AWS environment. The pipeline will build and deploy the images to the aws ECR (Elastic container registry) and run the applications in ECS (Elastic container service).

**CI/CD Pipeline process:**

Git - > Circleci -> Build and deploy -> AWS ECR -> AWS ECS -> Access the Techchallenge application

Git – Application Repository

CircelCI – This tool is used for continuous integration and delivery

circleci/config.yml - where is the root of your project’s directory. Since most of your work with CircleCI occurs in this file using the following configuration features: A sequential workflow; An orb.

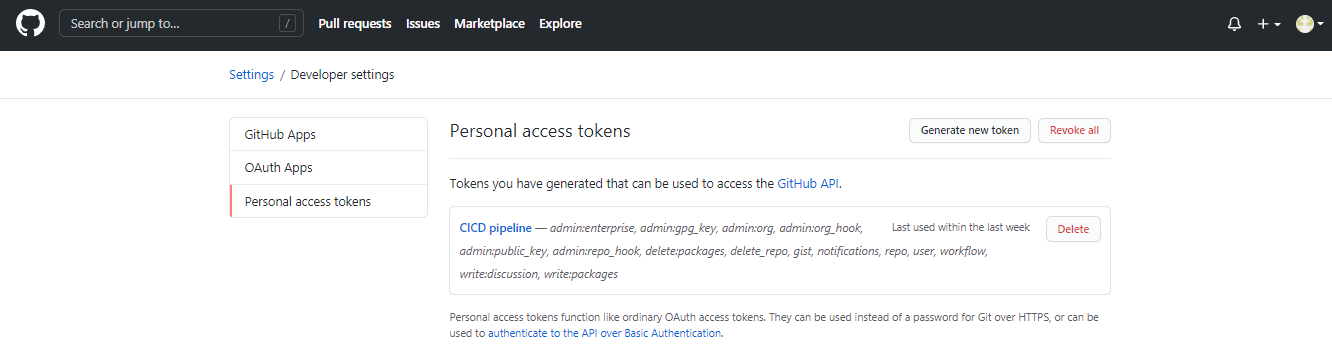
AWS ECR – Amazon Elastic Container Registry (ECR) is a fully managed container registry that makes it easy to store, manage, share, and deploy your container images and artifacts.

AWS ECS - Amazon ECS makes it easy to deploy, manage, and scale Docker containers running applications, services, and batch processes

**GIT Auth token setup**

Integration between git and circleci need to generate the token in github and update the token value in the circleci environment variable.

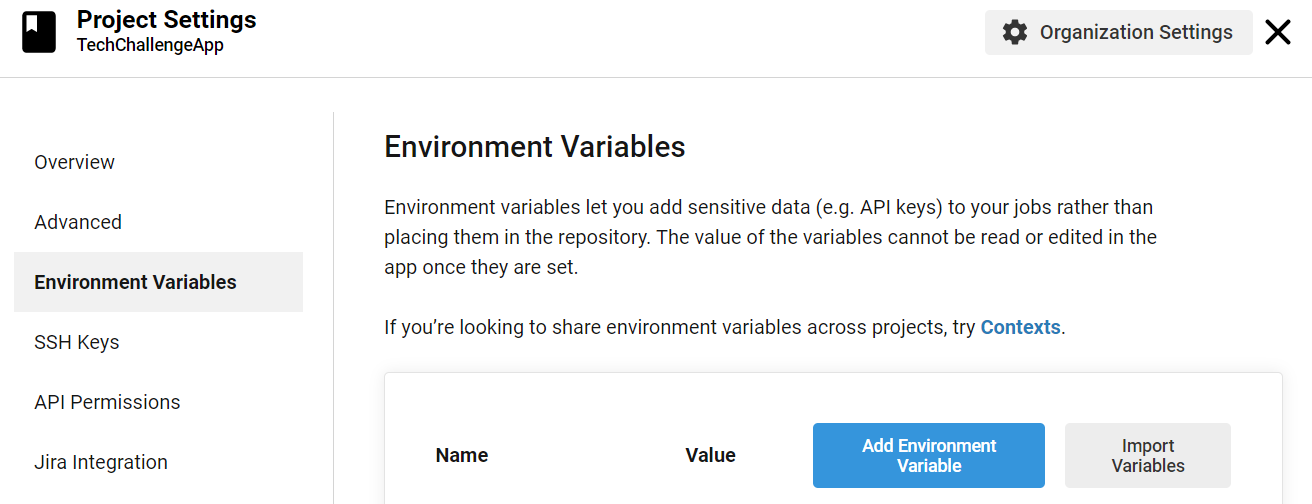
Git -> settings ->Developer settings - > personal access token -> Generate token

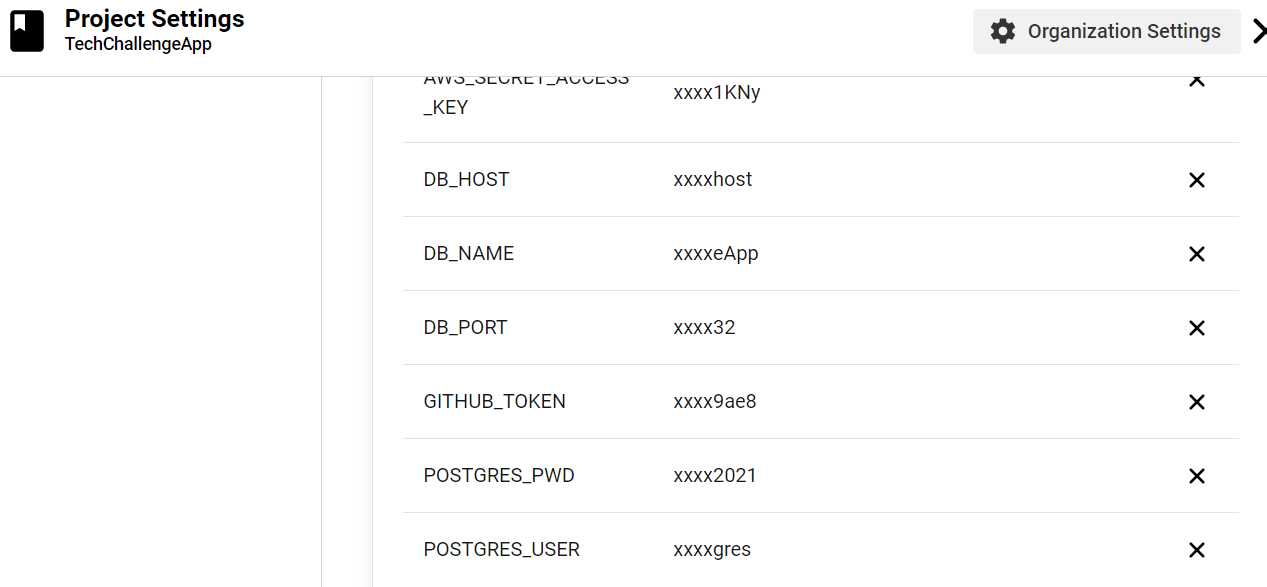


In the environment variable you can create GITHUB\_TOKEN and update this value.

**Environment variable configuration in circleci:**

The entire environment variable configuration created in the CircleCI environment section.





This Section is updated in the **circleci/config.yml file.**

environment:

          POSTGRES\_USER: POSTGRES\_USER

          POSTGRES\_PASSWORD: POSTGRES\_PWD

          POSTGRES\_DB: DB\_NAME

          POSTGRES\_PORT: DB\_PORT

          POSTGRES\_HOST: DB\_HOST

          TEST\_DATABASE\_URL: postgresql://root@localhost/circle\_test

Added the aws access key and secret access key and ECR account URL.

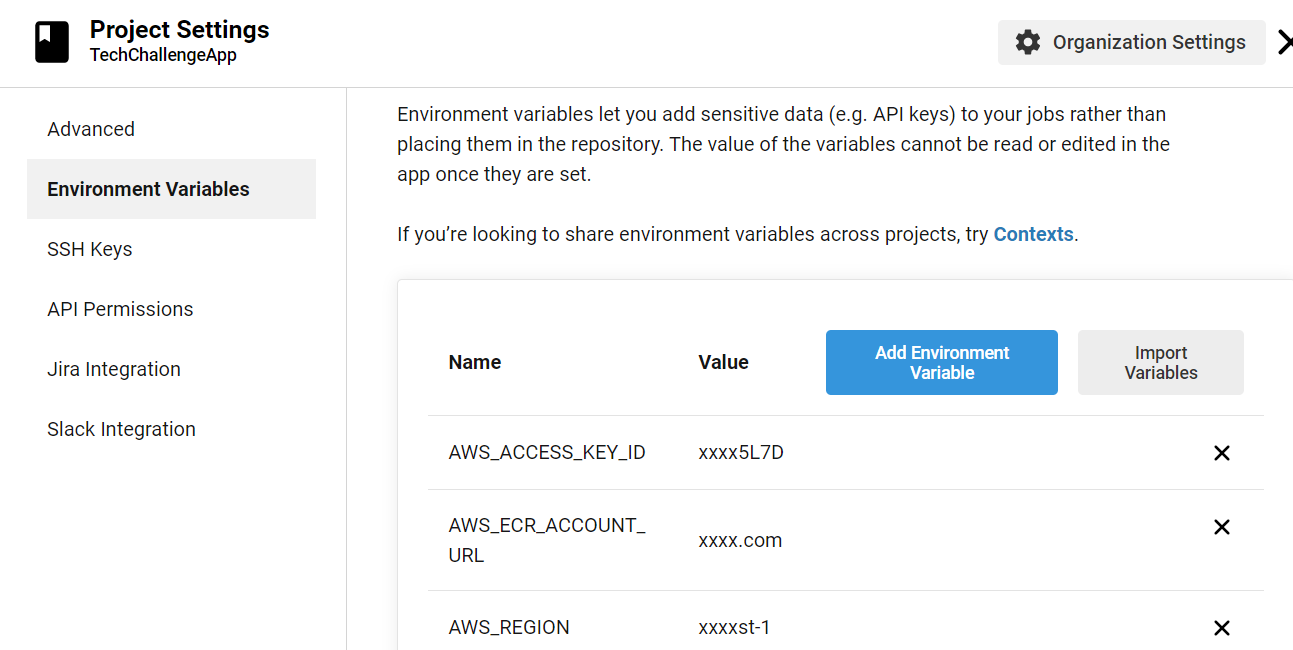
 - aws-ecr/build-and-push-image:

          account-url: AWS\_ECR\_ACCOUNT\_URL

          aws-access-key-id: AWS\_ACCESS\_KEY\_ID

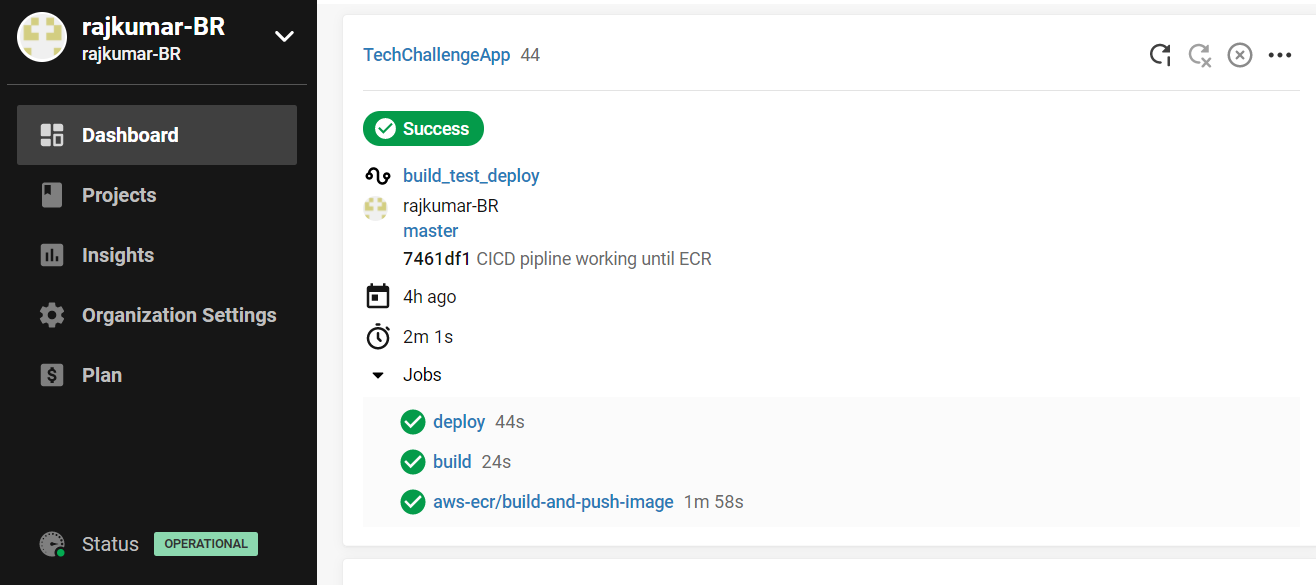
          aws-secret-access-key: AWS\_SECRET\_ACCESS\_KEY

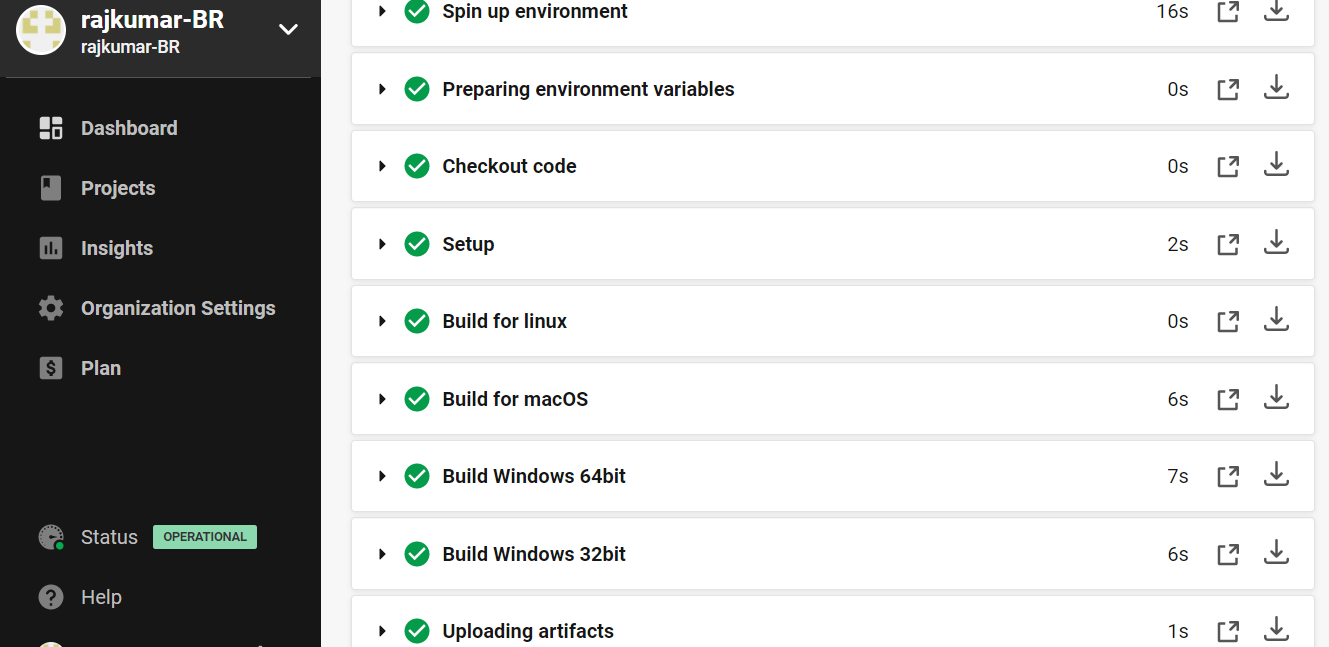
Before adding environment variable you have to create ECR account URL in AWS environment. This can be achieved manually or automation (ex: terraform, awscli,)

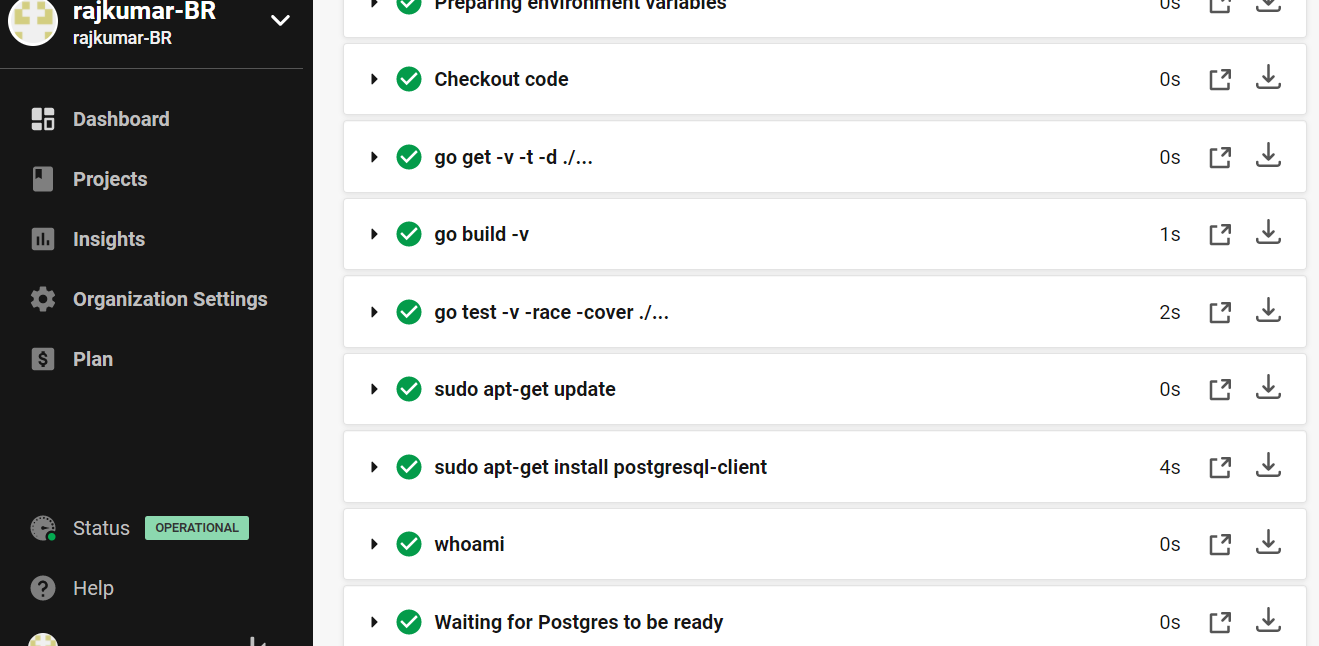


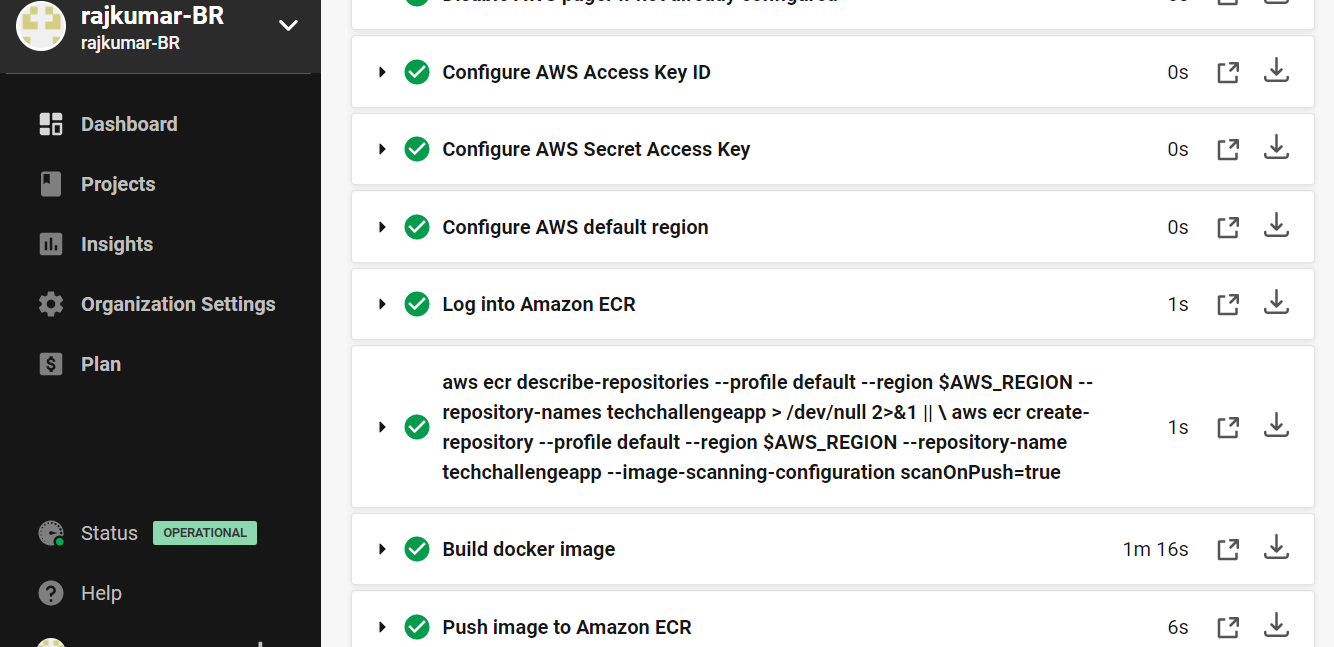
Once the all the configuration has been updated in the circleci/config.yml ready to run the workflow in the circleci tool.

Normally when you commit the new release in the github it will pushes the latest code into circleci and initiate the workflow to deploy the application.









 As of now my pipeline was perfectly working until (Docker Image) ECR. Currently I am facing multiple issues in ECS through circleci. Also, I have tried to create containers in ECS manually but it's still not working.