# AWS Copilot

# AWS Copilot is a tool for developers to build, release, and operate production-ready containerized applications on AWS App Runner and Amazon ECS on AWS Fargate. From getting started, pushing to staging, and releasing to production, Copilot can help manage the entire lifecycle of your application development.

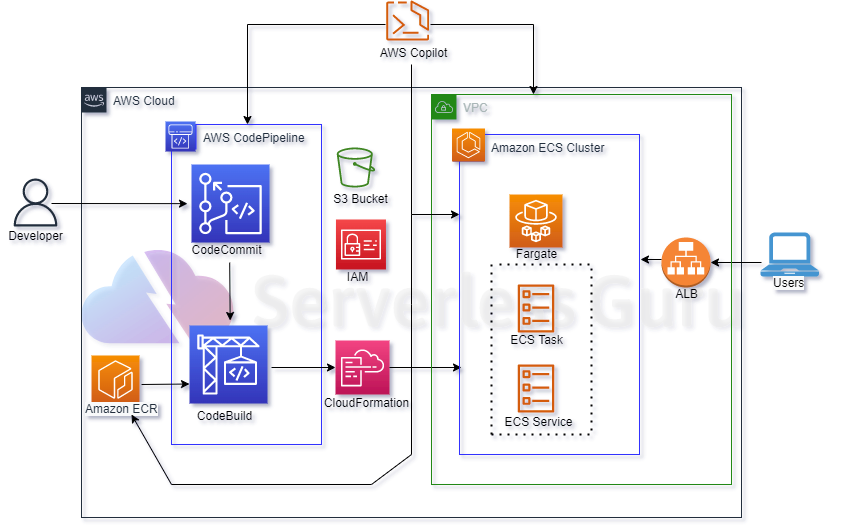
# AWS Copilot CLI aligns with developer workflows that support modern application best practices: from using infrastructure as code to creating a CI/CD pipeline provisioned on behalf of a user. Use the AWS Copilot CLI as part of your everyday development and testing cycle as an alternative to the AWS Management Console.

# It is a Command line Interface that enables customers to quickly launch and easily manage containerized application on AWS.

## AWS Copilot's Features

* Organize all your related micro-services in one application
* Set up CI/CD Pipelines for all of the micro-services
* Set up test and production environments, across regions and accounts
* Set up production-ready, scalable ECS services and infrastructure
* **AWS Copilot’s Commands**
* Copilot app ls: To list the deployed application.
* Copilot app show: Show Information about the environment and service in your application.
* Copilot env ls: Show information about your environment.
* Copilot svc show: Show Information about the service.
* Copilot svc ls: list all the service in the application.
* Copilot svc logs: Show logs of a deployed service.
* Copilot svc status: Show service status
* Copilot app delete.

# *Started with Amazon ECS using AWS Copilot*

****

* Install the AWS Copilot CLI .
* Run aws configure to set up a default profile that the AWS Copilot CLI uses to manage your application and services.
* Clone the application from the github (make sure the Dockerfile will work properly)
* Setup your application by using (Copilot init) Command.
* Setup an ECS Service in your demo application and choose Load balancer web service and select Dockerfile.
* Now we run the command (Copilot pipeline init)
* This command will create the buildspec.yml file
* Now we use following command to add buildspec.yml file into your github repo.
* Git add .
* Git commit –m “ ”
* Git push origin master
* Now final command (Copilot pipeline deploy) and after that go to codepipeline and move to the connections and give permission to attach with your github account.
* And your code deploy on ECS

