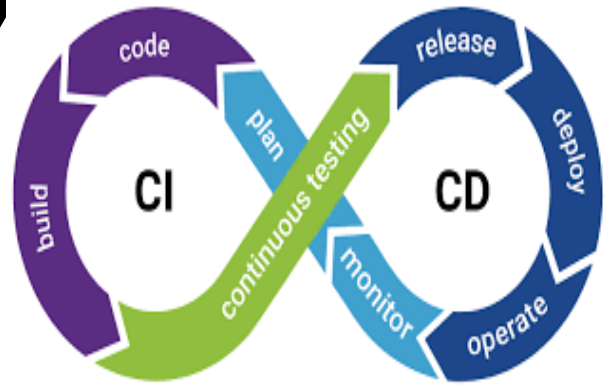
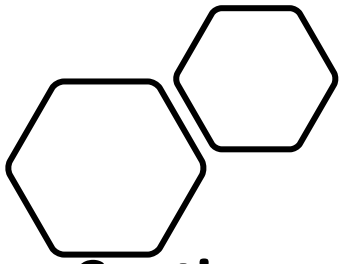
CONTINOUS INTEGRATION, CONTINOUS DELIVERY A PROPOSAL TO ***UDAPEOPLE*** BY LEAD DEVOPS ENGINEER

What is CICD buzzword?



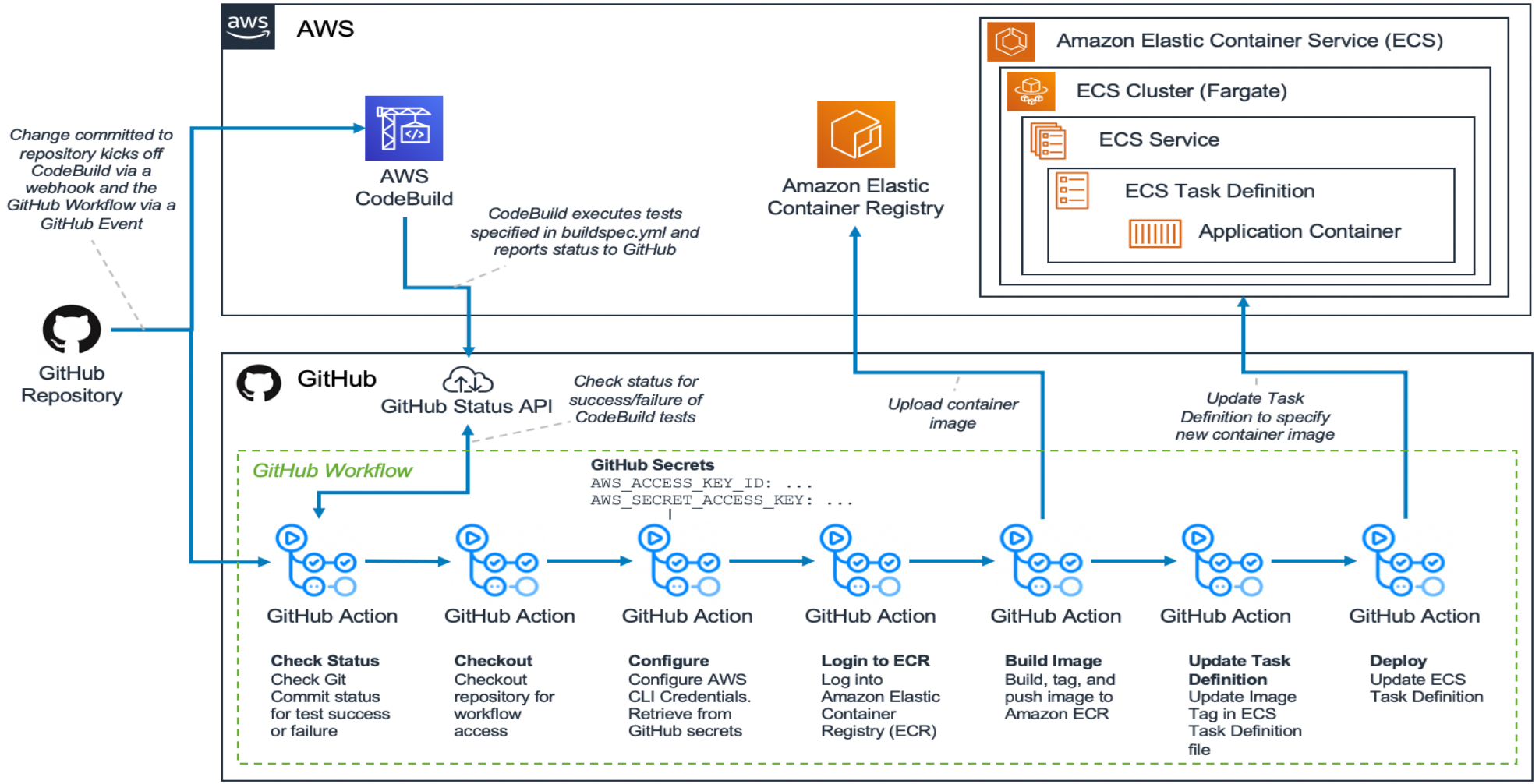
What is CICD



- **Continuous integration (CI)** is the practice of integrating all your code changes into the main branch of a shared source code repository early and often, automatically testing each change when you commit or merge them, and automatically kicking off a build.

- **Continuous delivery (CD)** is a software development practice that works in conjunction with continuous integration to automate the infrastructure provisioning and application release process.

How Does it Work?



CICD PIPELINE

WHY CICD

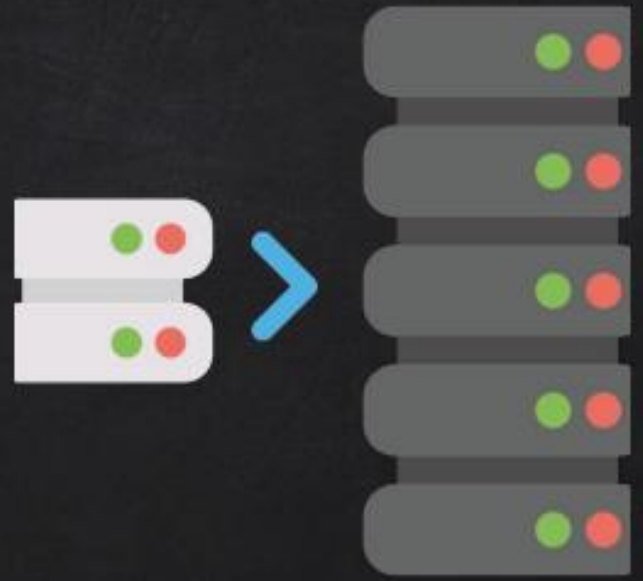
- Smaller code changes are simpler (more atomic) and have fewer unintended consequences.
- Fault isolation is simpler and quicker.
- Mean time to resolution (MTTR) is shorter because of the smaller code changes and quicker fault isolation.
- Testability improves due to smaller, specific changes. These smaller changes allow more accurate positive and negative tests.
- Elapsed time to detect and correct production escapes is shorter with a faster rate of release.
- The backlog of non-critical defects is lower because defects are often fixed before other feature pressures arise.
- The product improves rapidly through fast feature introduction and fast turn-around on feature changes.
- Upgrades introduce smaller units of change and are less disruptive.

What can we do now?

Catch compile and unit test failures after merge

- Less bugs in production, more time for developing new features

Scale servers to our needs



- Avoid extra costs on infrastructure

Automate infrastructure creation and monitoring

- Less human error and faster deployments