Google Cloud

DevOps Automation

Learning objectives

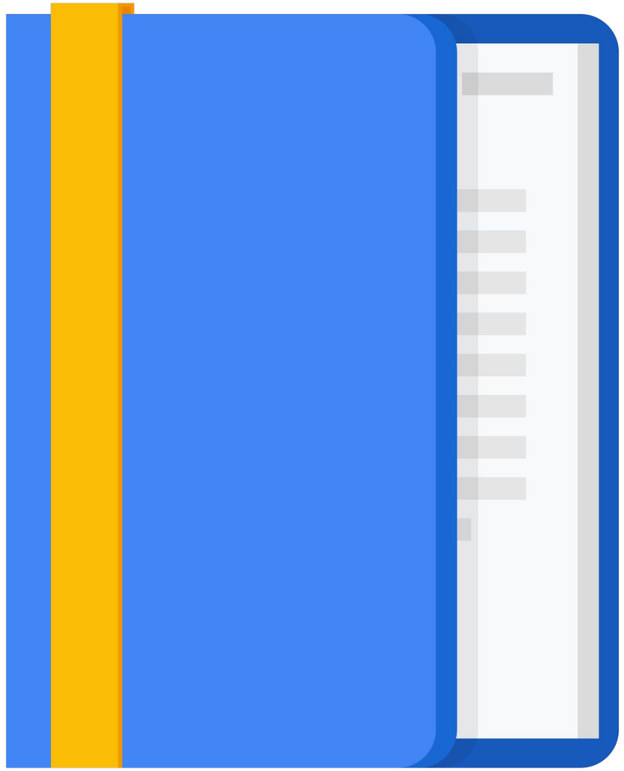
- Automate service deployment using CI/CD pipelines.
- Leverage Cloud Source Repositories for source and version control.
- Automate builds with Cloud Build and build triggers.
- Manage container images with Container Registry.
- Investigate infrastructure with code using Cloud Deployment Manager and Terraform.

Agenda

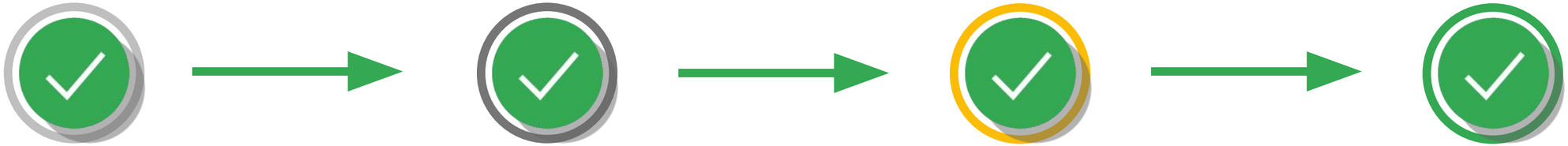
Continuous Integration Pipelines

Infrastructure as Code

Lab



Continuous integration pipelines automate building applications



Developers check-in code

Use a Git repo for each microservice and branches for versions.

Run unit tests

If the tests don’t pass, stop.

Build deployment package

Create a Docker image.

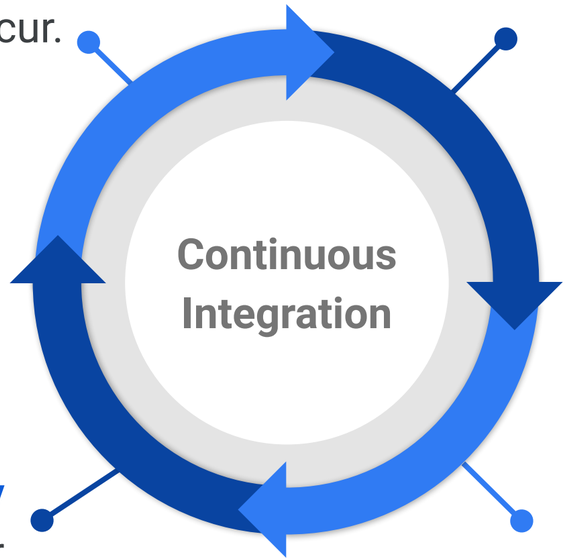
Deploy

Save your new Docker image in a container registry.

Google provides the components required for a continuous integration pipeline

Cloud Source Repositories
Developers push to a central repository when they want a build to occur.

Cloud Build
Build system executes the steps required to make a deployment package or Docker image.

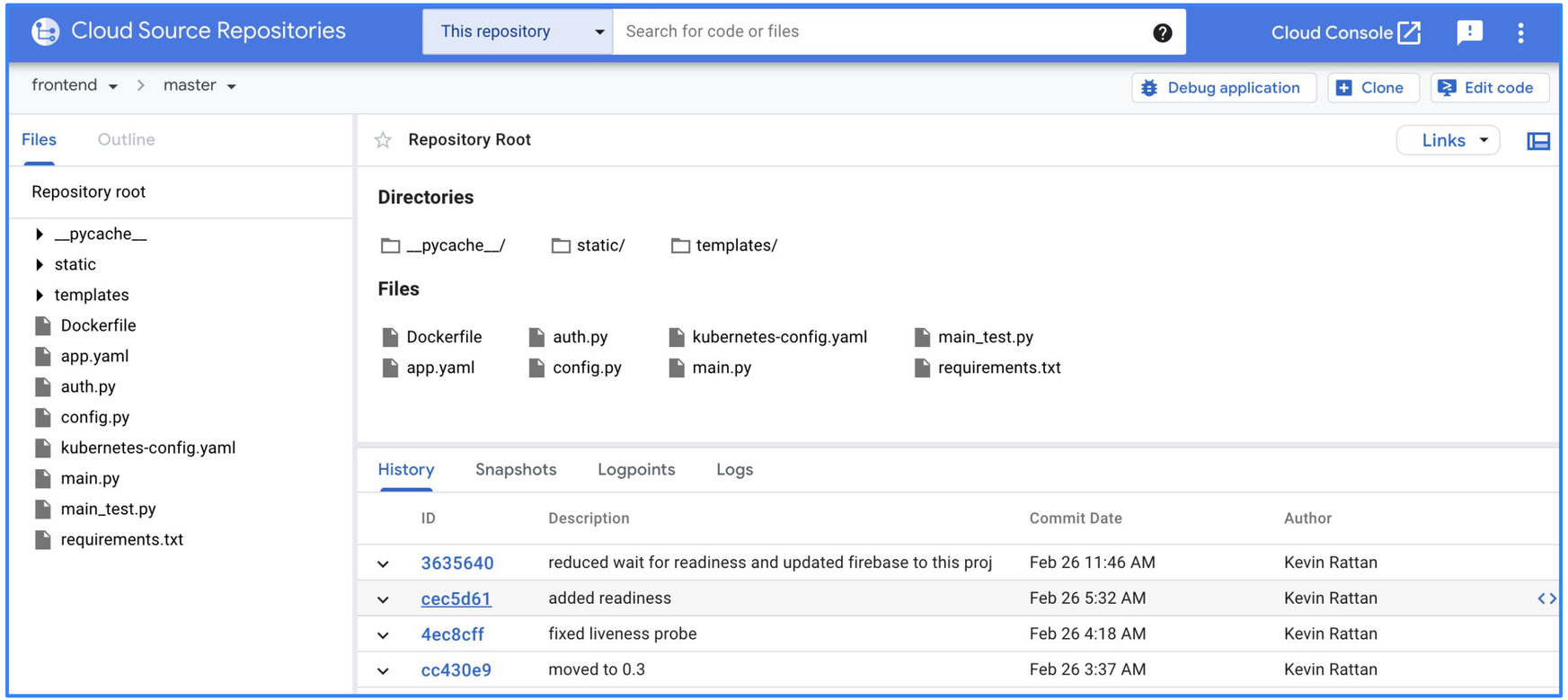


Container Registry
Store your Docker images or deployment packages in a central location for deployment.

Build triggers
Watches for changes in the Git repo and starts the build.

Cloud Source Repositories provides managed Git repositories

Control access to your repos using IAM within your Google Cloud projects.

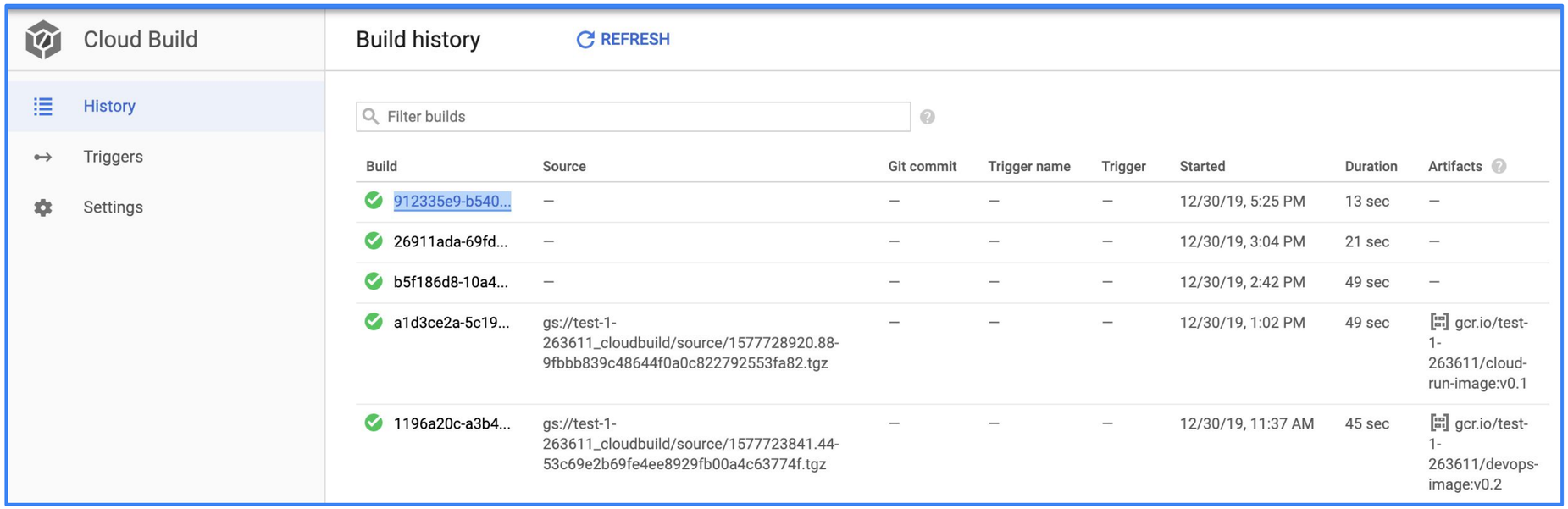


Google Cloud

Cloud Build lets you build software quickly across all languages

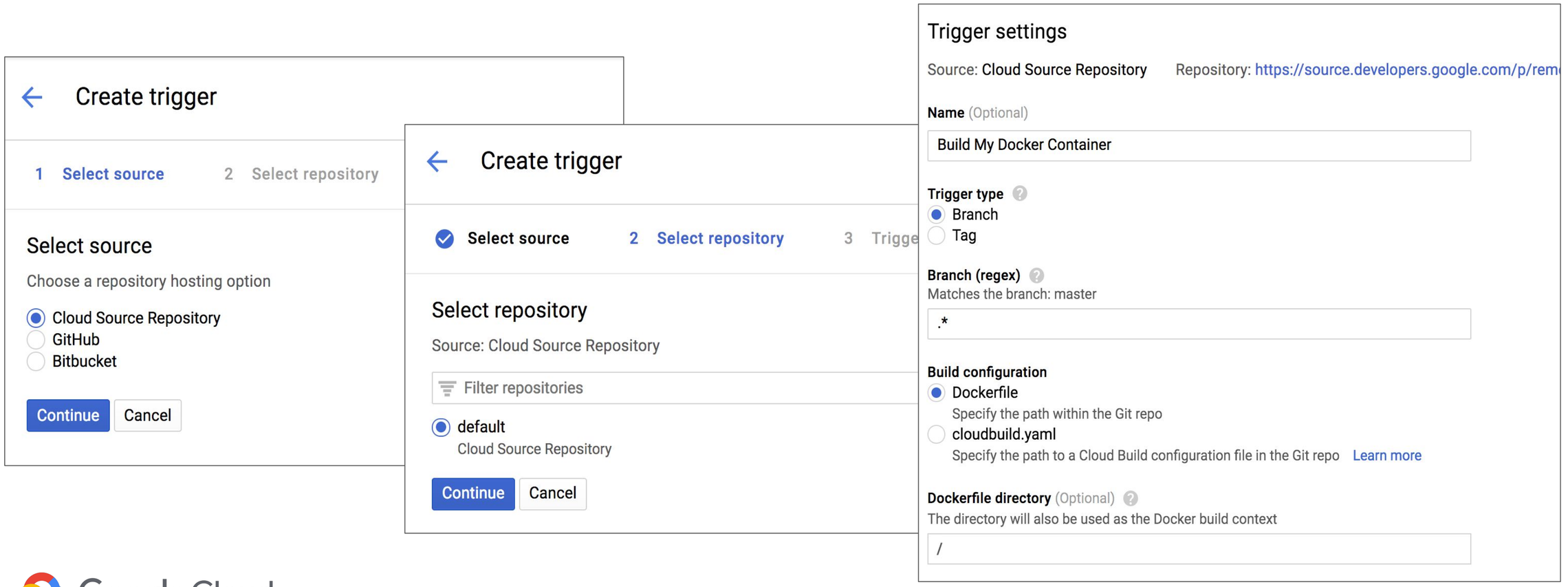
- Google-hosted Docker build service
  - Alternative to using Docker build command
- Use the CLI to submit a build

```
gcloud builds submit --tag gcr.io/your-project-id/image-name .
```



Build triggers watch a repository and build a container whenever code is pushed

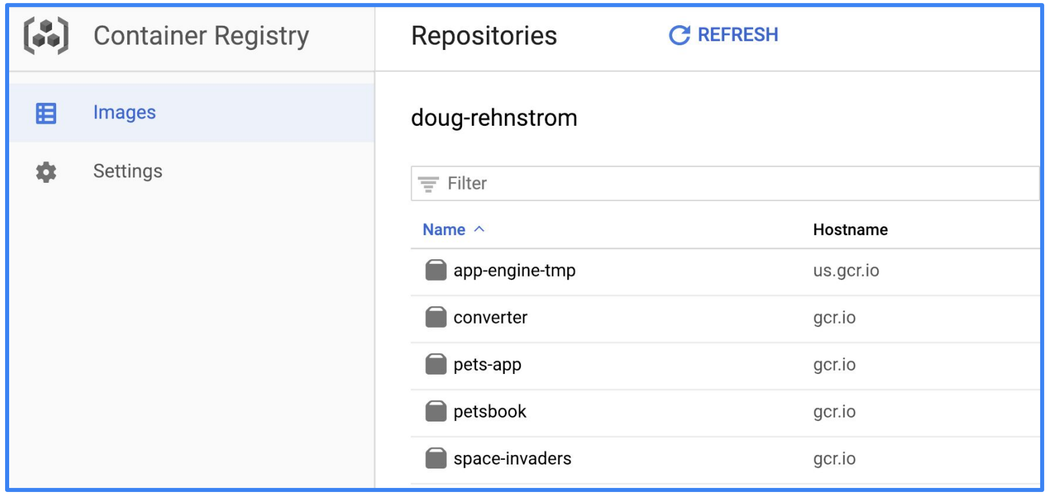
Supports Maven, custom builds, and Docker



Google Cloud

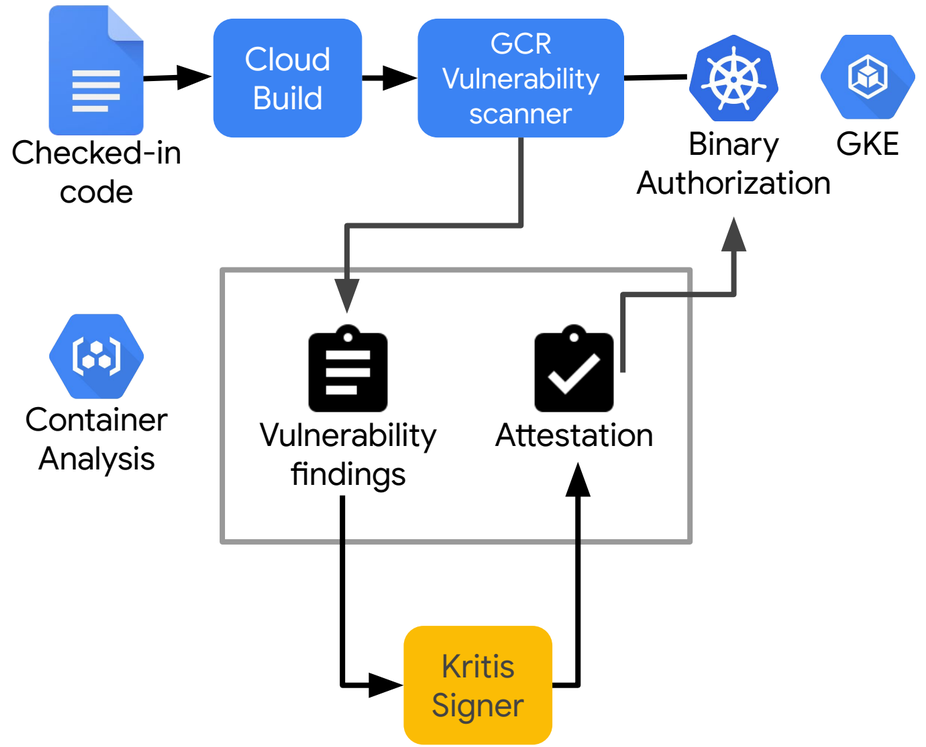
Container Registry is a Google Cloud–hosted Docker repository

- Images built using Cloud Build are automatically save in Container Registry.
  - Tag images with the prefix **`gcr.io/your-project-id`**`/image-name`
- Can use Docker push and pull commands with Container Registry.
  - `docker push gcr.io/your-project-id/image-name`
  - `docker pull gcr.io/your-project-id/image-name`



Binary authorization allows you to enforce deploying only trusted containers into GKE

- Enable binary authorization on GKE cluster.
- Add a policy that requires signed images.
- When an image is built by Cloud Build an “attestor” verifies that it was from a trusted repository (Source Repositories, for example).
- Container Registry includes a vulnerability scanner that scans containers.

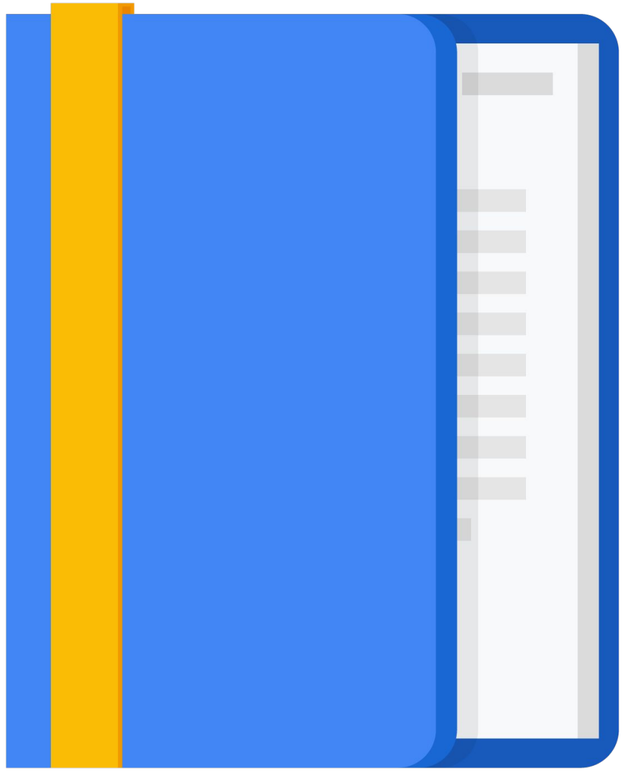


Agenda

Continuous Integration Pipelines

Infrastructure as Code

Lab



Moving to the cloud requires a mindset change

On-Premises	Cloud
Buy machines.	Rent machines.
Keep machines running for years.	Turn machines off as soon as possible.
Prefer fewer big machines.	Prefer lots of small machines.
Machines are capital expenditures.	Machines are monthly expenses.

In the cloud, all infrastructure needs to be disposable

- Don’t fix broken machines.
- Don’t install patches.
- Don’t upgrade machines.
- If you need to fix a machine, delete it and re-create a new one.

- To make infrastructure disposable, automate everything with code:
  - Can automate using scripts.
  - Can use declarative tools to define infrastructure.

Infrastructure as code (IaC) allows for the quick provisioning and removing of infrastructures

- Build an infrastructure when needed.
- Destroy the infrastructure when not in use.
- Create identical infrastructures for dev, test, and prod.
- Can be part of a CI/CD pipeline.
- Templates are the building blocks for disaster recovery procedures.
- Manage resource dependencies and complexity.

- Google Cloud supports many IaC tools.

Cloud Deployment Manager is Google Cloud's native IaC tool

- Define infrastructure using YAML syntax.
- Can create dynamic templates using Python or Jinja.
- Use gcloud to create, update, and delete deployments.

```
resources:
# Configure a VM
- name: devops-vm
  type: compute.v1.instance
  properties:
    zone: us-central1-a
    machineType: zones/us-central1-a/machineTypes/f1-micro
    disks:
    - deviceName: boot
      type: PERSISTENT
      boot: true
      autoDelete: true
      initializeParams:
        sourceImage: projects/debian-cloud/global/images...
    # Add VM to default network and give it an external IP
    networkInterfaces:
    - network: global/networks/default
      accessConfigs:
        - name: External NAT
          type: ONE_TO_ONE_NAT
```

Terraform is similar to Deployment Manager but can be used on multiple public and private clouds

- Considered a first-class tool in Google Cloud.
- Already installed in Cloud Shell.

```
provider “google” {
    credentials = “”
    project     = “project name”
    region      = “us-central1”
}

resource “google_compute_instance” {
    name         = “instance name”
    machine_type = “n1-standard-1”
    zone         = “us-central1-f”

    disk {
        image = “image to build instance”
     }

    }

output “instance_ip” {
    value = “${google_compute.ip_address}”
}
```

60 minutes

Lab

Building a DevOps Pipeline

Objectives

- Create a Git Repository
- Create a Simple Python Application
- Test Your Web Application in Cloud Shell
- Define a Docker Build
- Manage Docker Images with Cloud Build and Container Registry
- Automate Builds with Triggers
- Test Your Build Changes

Google Cloud

Quiz

Which Google Cloud tools can be used to build a continuous integration pipeline?

A. Cloud Source Repositories

B. Cloud Build

C. Container Registry

D. All of the above

Quiz

Which Google Cloud tools can be used to build a continuous integration pipeline?

A. Cloud Source Repositories

B. Cloud Build

C. Container Registry

D. All of the above

Quiz

List some reasons to automate infrastructure creation using code tools like Deployment Manager and Terraform.

Quiz

List some reasons to automate infrastructure creation use code tools like Deployment Manager and Terraform.

Easier to make dev, test, and prod environments the same
Easier to change and fix infrastructure over time
Simplify administration
Automate provisioning and decommissioning
Save money

Quiz

What Google Cloud feature would be easiest to use to automate a build in response to code being checked into your source code repository?

A. Build triggers

B. Cloud Functions

C. App Engine

D. Cloud Scheduler

Quiz

What Google Cloud feature would be easiest to use to automate a build in response to code being checked into your source code repository?

A. Build triggers

B. Cloud Functions

C. App Engine

D. Cloud Scheduler

Review

DevOps Automation

More resources

Google Cloud DevOps solutions

https://cloud.google.com/devops/

Deployment Manager

https://cloud.google.com/deployment-manager/

Terraform on Google Cloud

https://cloud.google.com/community/tutorials/getting-started-on-gcp-with-terraform


Google Cloud