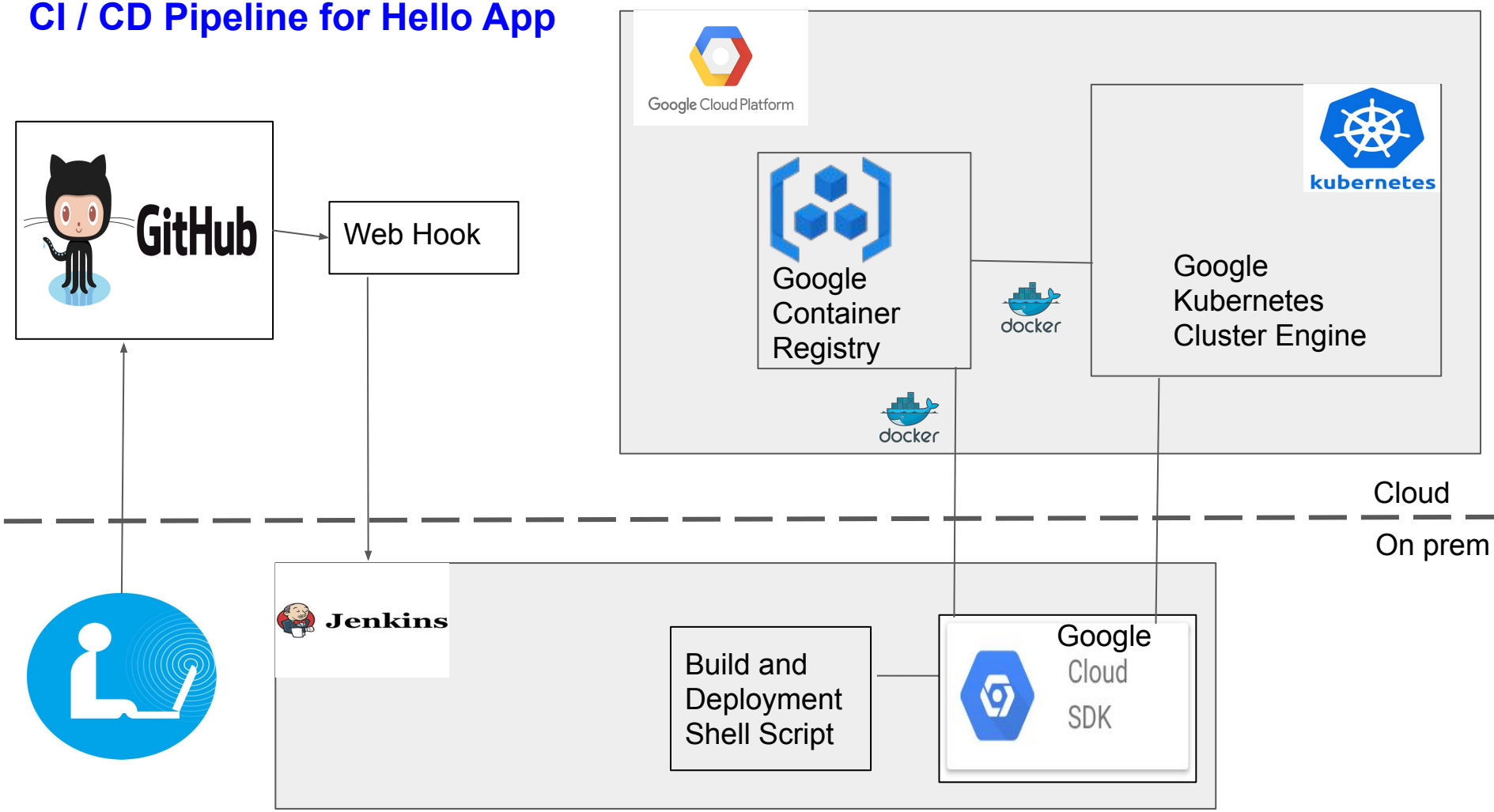
Jignesh Gujarati

Cloud CI / CD Project

Section 1: Strategy and Tactics

- Create fully automated CI / CD pipeline
- Dockerised the application
- Deployment on Google Kubernetes Engine, probably the most robust , automated and scalable container engine/cluster service out there.
- All steps, 100% of it , is Automated including the testing

CI / CD Pipeline for Hello App



CI / CD Steps

All the CI / CD Steps are 100% automated including testing.

CI Steps

1. Code change to Github
2. Github Webhook for jenkins triggers the MyHelloApp jenkins job
3. Jenkins job runs the build_n_deploy_on_gke.sh script

   build_n_deploy_on_gke.sh does following

4. Builds Docker image
5. Runs Docker container locally
6. Tests the app on local container using curl command

CI / CD Steps

7. Stops the local container

8. Pushes the tested Docker image to Google Container Registery (GCR)

9. Deletes the local container

CD Steps

10. Creates Google Container Cluster on Google Kuberenetes Engine (GKE)

11. Runs GKE deployment for Docker image just pushed to GCR

CI / CD Steps

12. Creates GKE Service resource

13. Creates external IP and Load Balancer using GKE

14. Tests App on external IP using curl

15. Configures Autoscaling

Terminate the deployment

16. Auto destroys the Service, Deployment and Cluster

Section 2: Deployment Instructions

Requirements for executing deployment

Github

Jenkins

GCP Account

GCP Project

Google Cloud SDK installed

GCP credentials locally stored

Docker Installed