

**Build a CI/CD Pipeline In The  
Cloud With Less Than 10  
Dollars Per Month**

# Dante Faña Badia

SR. SOFTWARE DEVELOPER

TWITTER / GITHUB: @dantefanabadia

# EXPECTATIONS

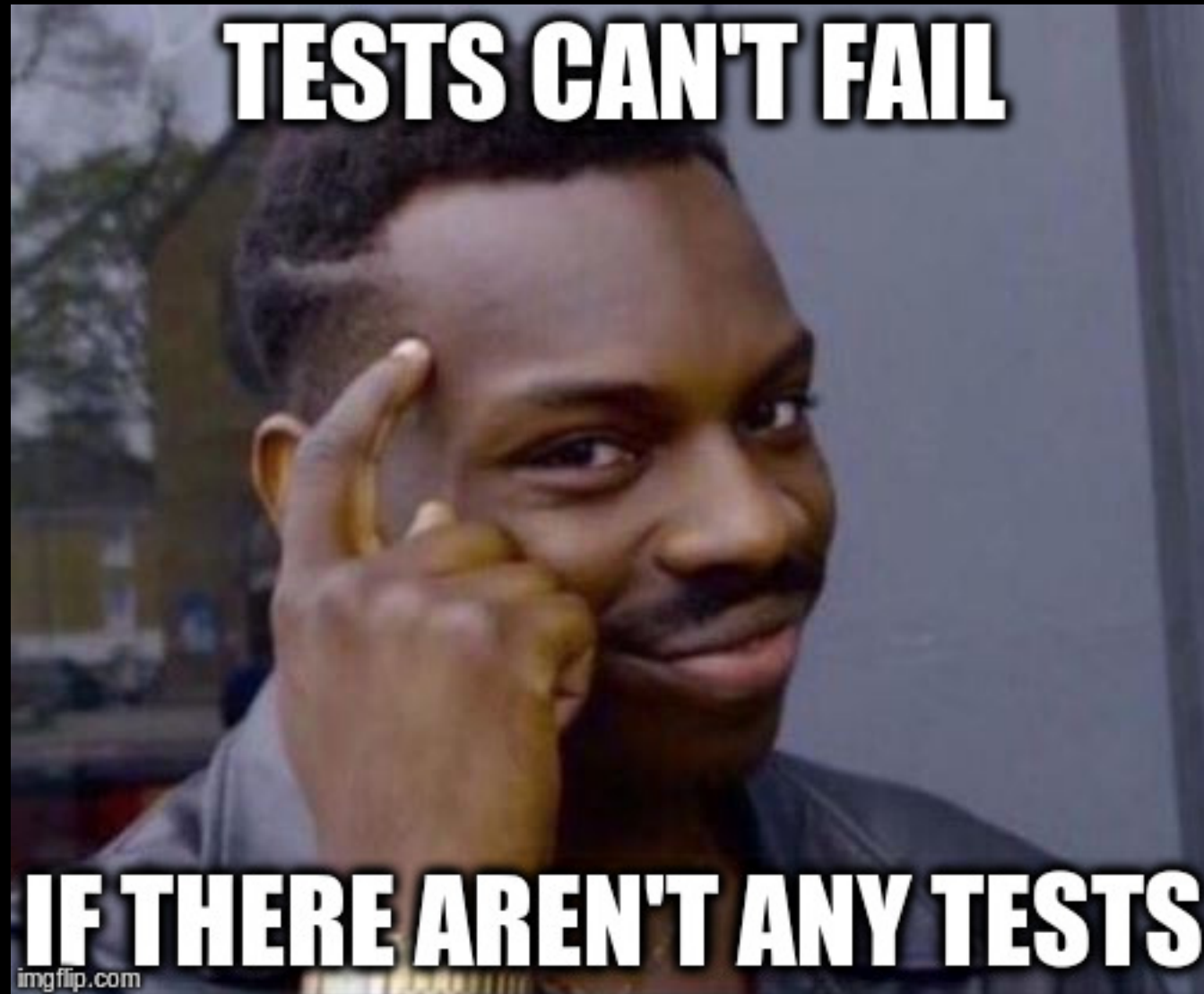
WHY, HOW IMPLEMENT AND USE CI/CD

**ANALOGY**

# THE PROBLEM

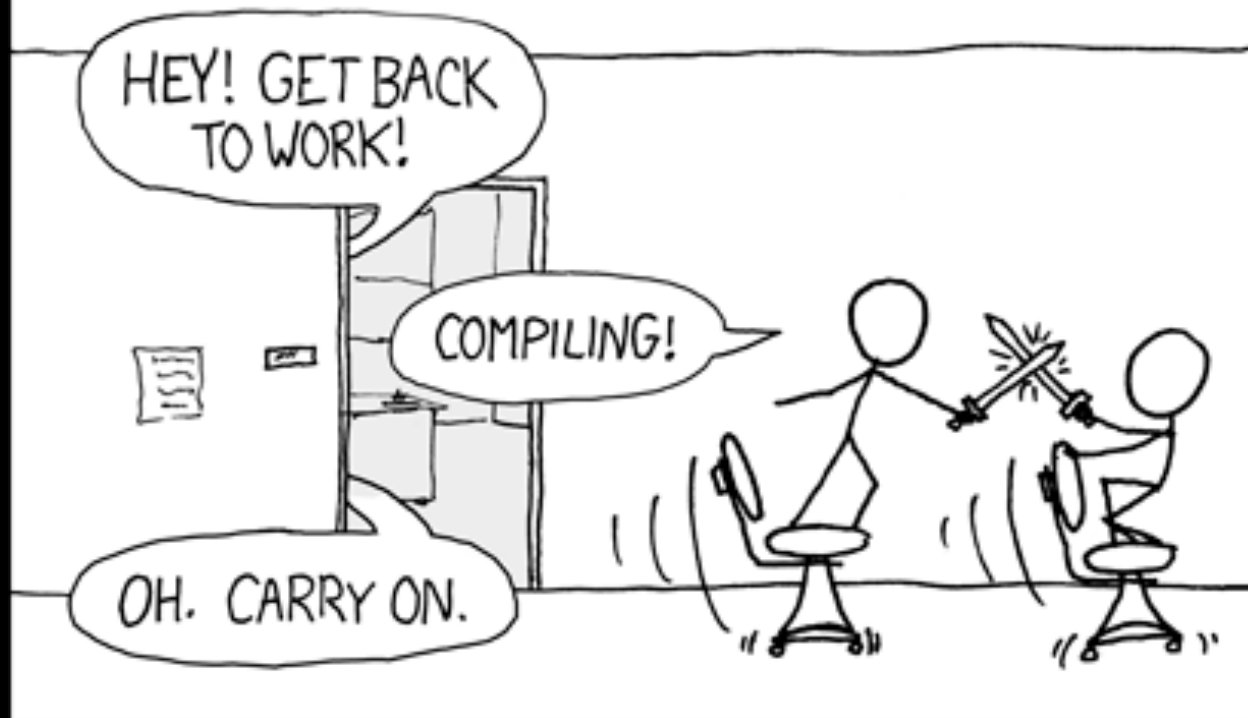


*“Debemos hacer el pase a producción en la madrugada”*



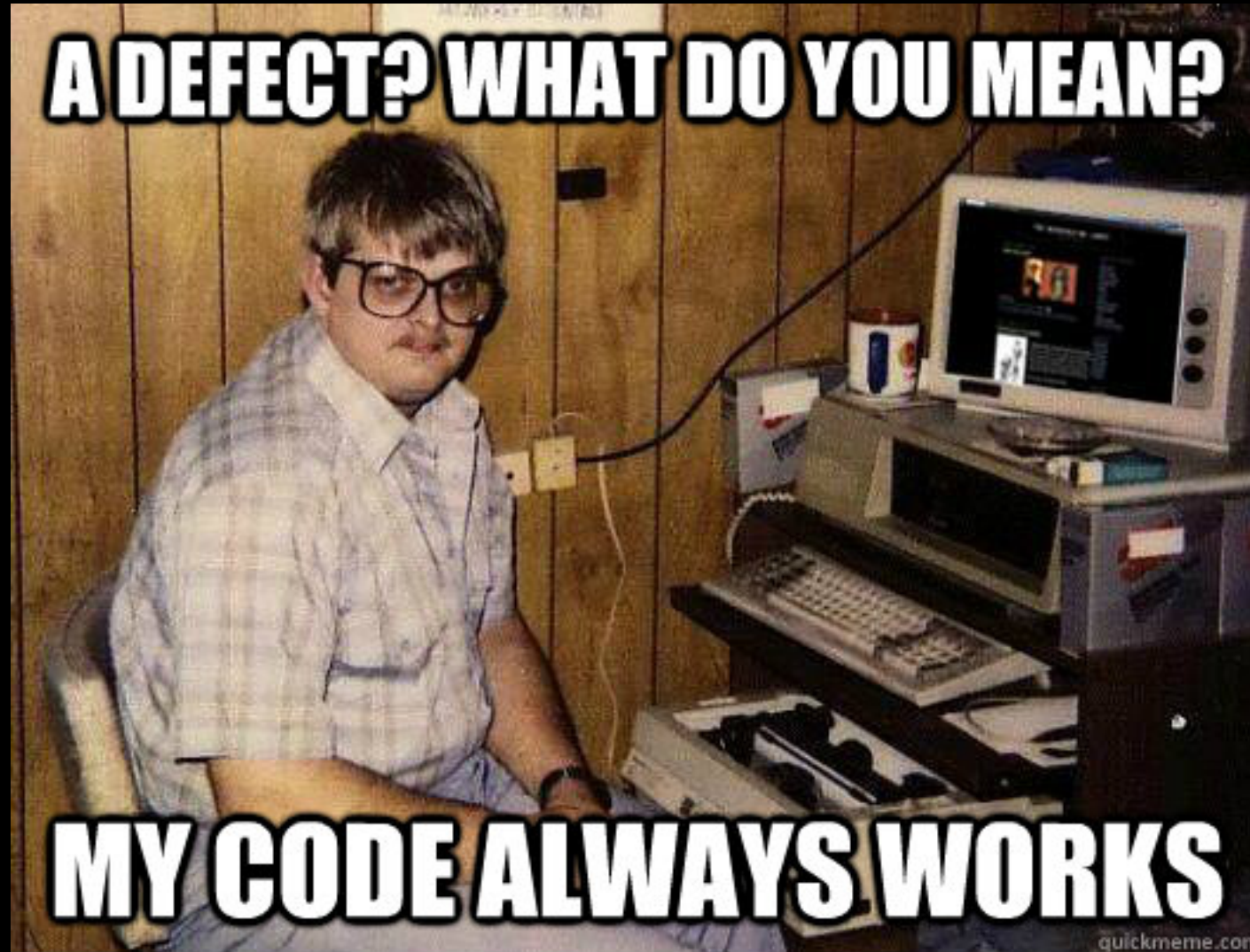
*“Actualice el código del repositorio pero ya no funciona”*

THE #1 PROGRAMMER EXCUSE  
FOR LEGITIMATELY SLACKING OFF:  
"MY CODE'S COMPILING."



***"Solo falta compilar"***





*"Hey, pero funciona en mi maquina"*

**THE SOLUTION**

**CD/CI**

**THE SOLUTIONS**

# WHAT IS CI?

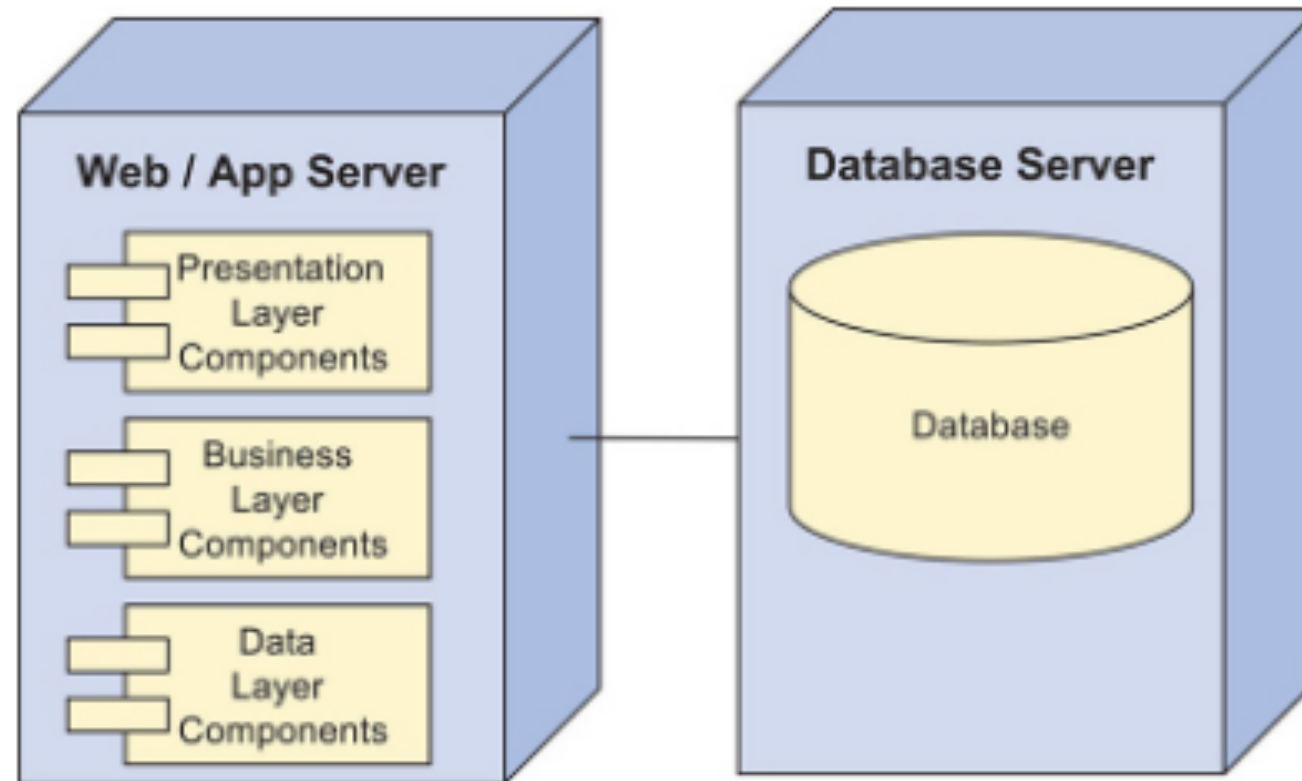
- El desarrollador debe integrar y comitiar código en el repositorio cada pocas horas, siempre que sea posible.
- Nunca retener los cambios mas de un día
- Evitar cambios de desarrollo fragmentado donde los desarrolladores no se comunican
- Se debe trabajar en la ultima versión pues cambios en el código obsoleto traer dolores de cabeza

# WHAT IS CD?

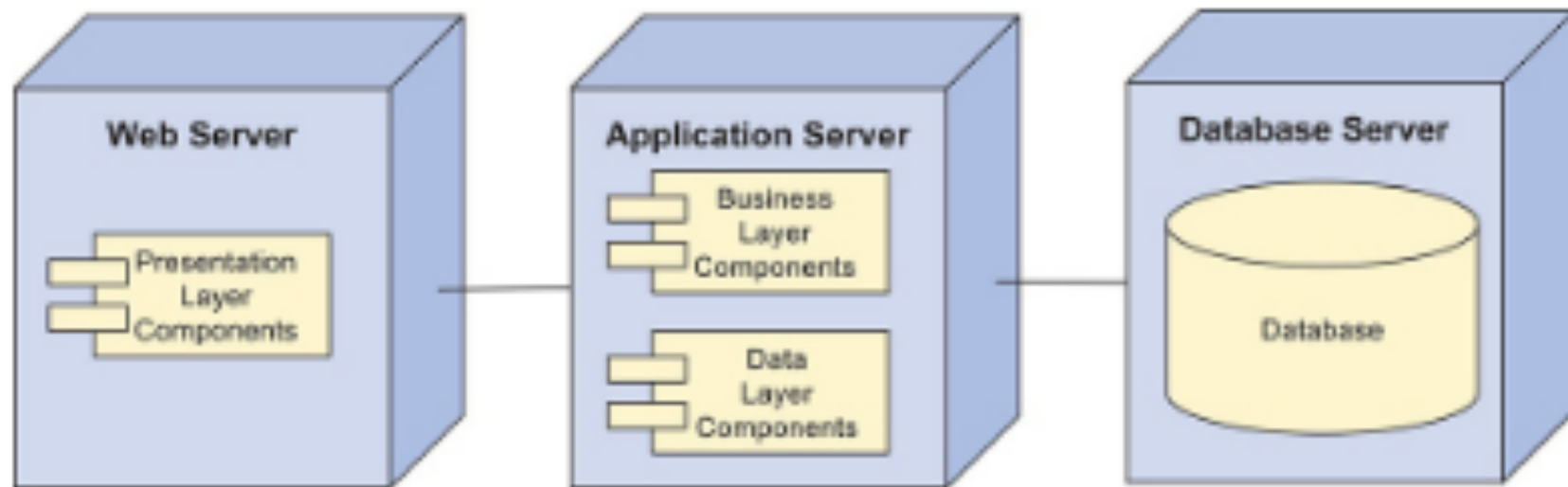
- Es la capacidad de desplegar todo tipo de cambios (configuración, nueva funcionalidad, correcciones) de forma segura y rápida
- Lograr que nuestro código siempre este en estado de para publicar a producción
- Eliminar complejidad en la fase de integración, prueba y producción.
- Eliminar el código congelado
- Mejor documentación para los pases a producción es automatizar el mismo

# Deployability

Se refiere a cómo un ejecutable, ¿dónde se actualizan a los usuarios, dónde necesitan solicitar actualizaciones explícitamente? ¿Cómo se integra en un sistema existente? ¿Se puede hacer esto mientras se ejecuta el sistema existente?



**Nondistributed deployment example**

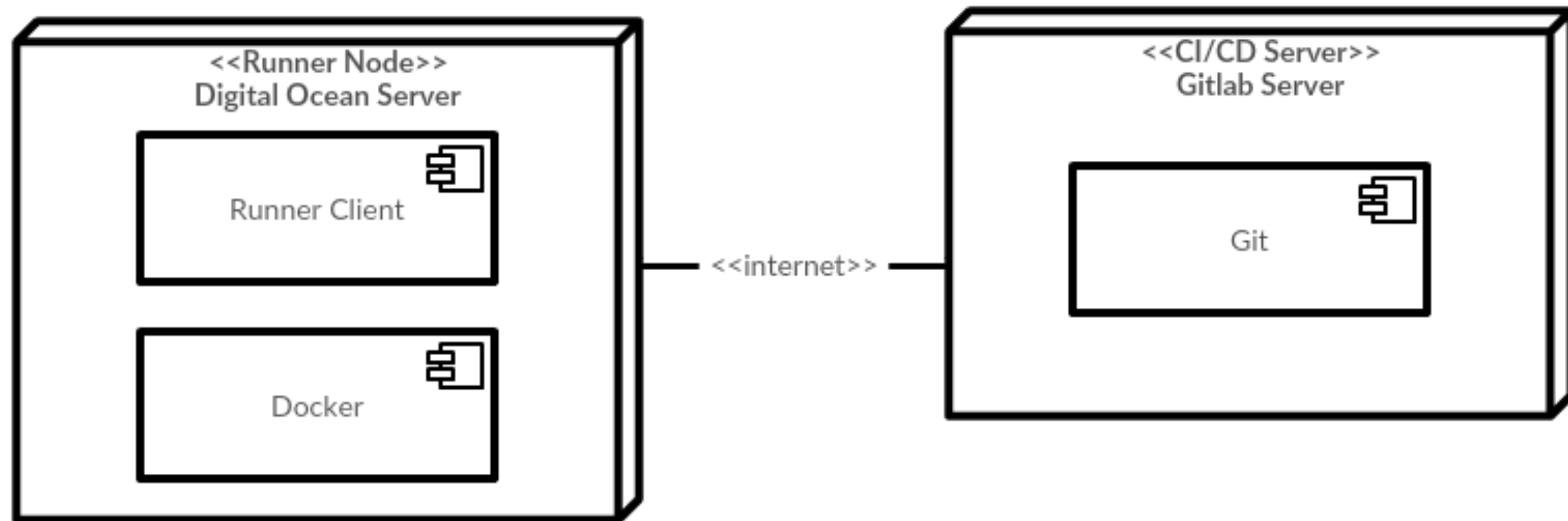


**Distributed deployment example**



# TOOLS

Gitlab CI/CD, Git, Docker, Digital Ocean





DigitalOcean



GitLab



docker



git

\$5.00/mo

\$0.00/mo

\$0.00/mo

\$0.00/mo

---

\$5.00/mo

# LIVE DEMO

STEP BY STEP

A close-up of a man with brown hair and black-rimmed glasses, wearing a red jacket. He has a slight, smug smile on his face. The background is blurred, showing some indistinct shapes and colors.

**LIVE DEMO**

**I ALSO LIKE TO LIVE  
DANGEROUSLY**

memegenerator.net

## **STEP 1: Setting up my server droplet**

<https://cloud.digitalocean.com/projects/>

## **STEP 2: Setting up my gitlab project**

<https://gitlab.com/>

## STEP 3: Setting up the runner client

```
sudo curl -L  
https://packages.gitlab.com/install/repositories/runner/gitlab-  
runner/script.deb.sh | sudo bash
```

```
sudo apt-get install gitlab-runner
```



## STEP 4: Setting up the ssh runner

```
sudo gitlab-runner register -n \  
  --url https://gitlab.com/ \  
  --registration-token $REGISTRATION_TOKEN \  
  --executor shell \  
  --description ${runner-desc}
```

## STEP 5: Setting up the docker runner

```
sudo gitlab-runner register -n \  
  --url https://gitlab.com/\  
  --registration-token $REGISTRATION_TOKEN \  
  --executor docker \  
  --description ${runner-desc}\  
  --docker-image "docker:stable" \  
  --docker-privileged
```

# STEP 6: Creating my .gitlab-ci.yml file

```
# archivo que se encarga de definir el pipeline del CI/CD de gitlab
# .gitlab-ci.yml
# stage name of the pipeline to execute. they are in order
stages:
  - ${stage-n}
# folder to cache example: .m2, node_module, etc.
cache:
  key: "$CI_BUILD_REF_NAME"
  paths:
    - ${path-to-cache-n}
# job name
${name}:
  images: ${images-name}
  stage: ${stage-name}
  tags:
    - ${runner-tag}
  services:
    - name: docker:dind
  variables:
    ${KEY_N}: ${VALUE_N}
  before_script:
    - ${command-n}
  script:
    - ${command-n}
```

## **STEP 7: Make a Commit**

<https://gitlab.com/dantefanabadia/my-awesome-app>

## STEP 8: **Monitoring the pipeline**

<https://gitlab.com/dantefanabadia/my-awesome-app/pipelines>



# REFERENCES

- <http://www.extremeprogramming.org/rules/integrateoften.html>
- <https://www.infoq.com/minibooks/scrum-xp-from-the-trenches-2>
- <https://pragprog.com/book/tpp/the-pragmatic-programmer>
- <https://www.digitalocean.com/community/tutorials/how-to-install-and-use-docker-on-ubuntu-16-04>
- <https://www.digitalocean.com/community/tutorials/initial-server-setup-with-ubuntu-18-04>
- <https://docs.gitlab.com/runner/install/>
- <https://docs.gitlab.com/runner/register/index.html>
- [https://docs.gitlab.com/ee/ci/docker/using\\_docker\\_build.html](https://docs.gitlab.com/ee/ci/docker/using_docker_build.html)

**Thank you!**

**@dantefanabadia**