

# Business card website development CI/CD

---

By Serhii Borovyk

# Goal

The goal of the project is to create a CI/CD Pipeline that automates the software development life cycle - the creation of a web application - a business card website, to ensure fast, efficient, and high-quality code delivery.

This is achieved through a series of automated steps that include getting the code from GitHub, building it, testing it, and delivering it to the deployment servers - the stage server and the production server.

The ultimate goal of this process is to make the process of developing and deploying web applications more streamlined, reduce errors, and ensure fast deployment with minimal manual intervention.

# Tools and tech

Used in project

- Terraform
- Ansible
- Docker
- AWS and OCI



# Key benefits

1. Developers work together on the code base, ensuring smooth and continuous integration of changes. Reduces the time and effort required to resolve integration conflicts. Helps speed up release cycles.
2. Different environments such as automation, stage and production servers ensures that code is tested, meets customer requirements reducing runtime issues.
3. Automation creates the pipeline and automates tasks such as code building, testing and deployment. Reduces manual effort required.
4. Scalable and secure infrastructure as AWS and OCI allows to scale resources and deploy a web application. Provides a secure and reliable infrastructure for the pipeline. Ensures the application is available to end users with minimal downtime.

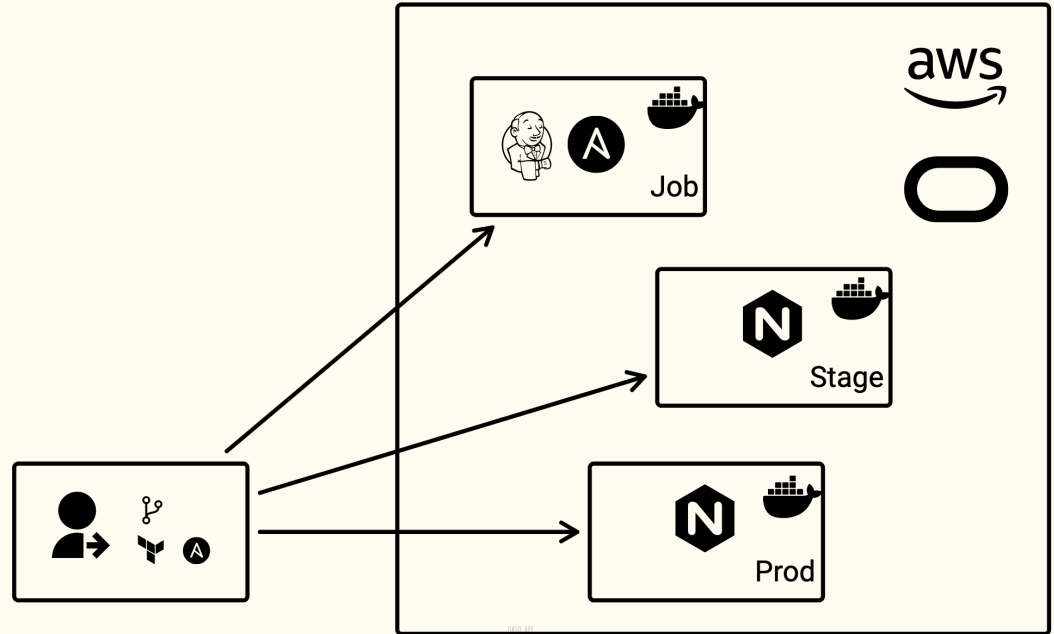
# Infrastructure

---

Distributed infrastructure across servers.

### Local tools

- Git for project code development
- Terraform for building infrastructure
- Ansible to configure it



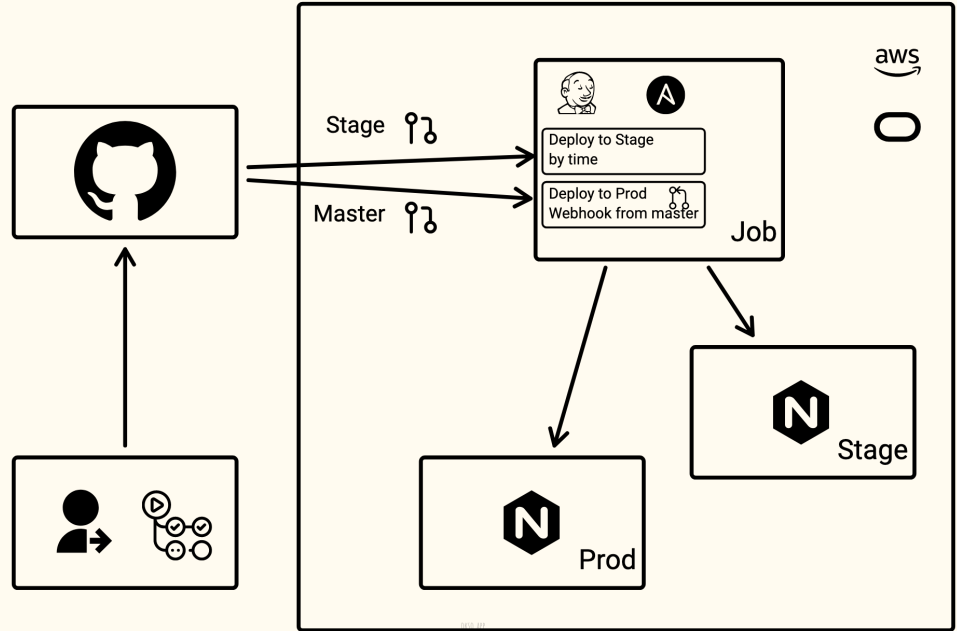


# Pipeline

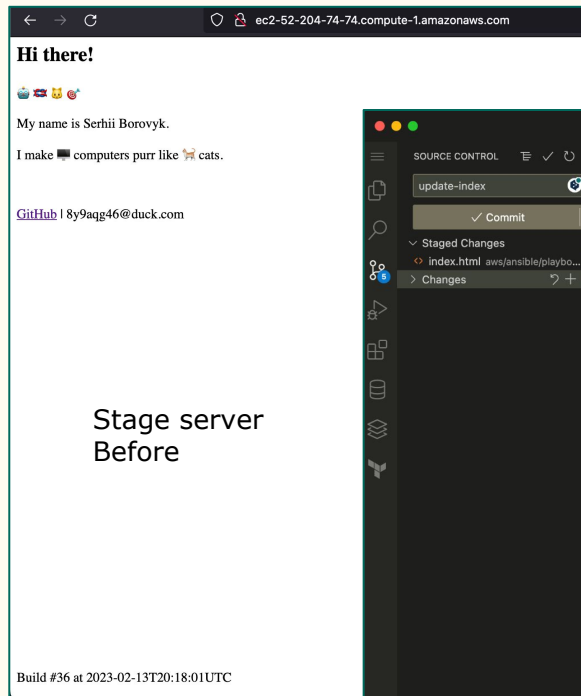
---



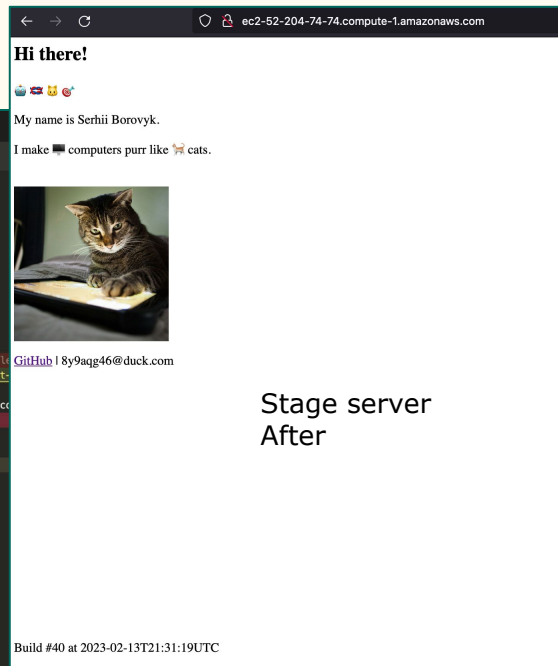
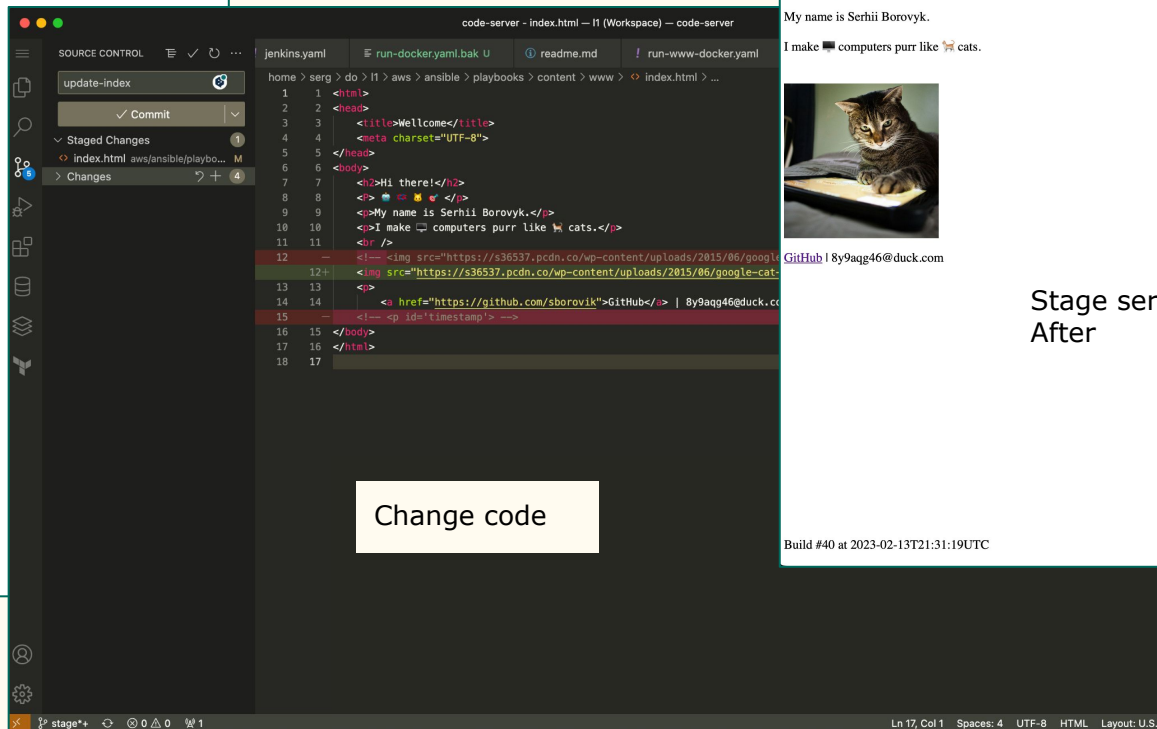
1. Developers push local code to the stage branch on GitHub
2. A Jenkins node periodically clones code from the stage branch on GitHub, tests it, and deploys it to the stage server for review.
3. If the web application meets the requirements, the developers merge the stage branch into the master branch.
4. Changes to the master branch trigger the job on the Jenkins node, which automatically receives and builds the code, tests it, and deploys the changes to the production server.



# Stage pipeline

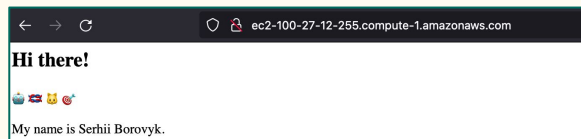


Stage server  
Before



Stage server  
After

# Production pipeline



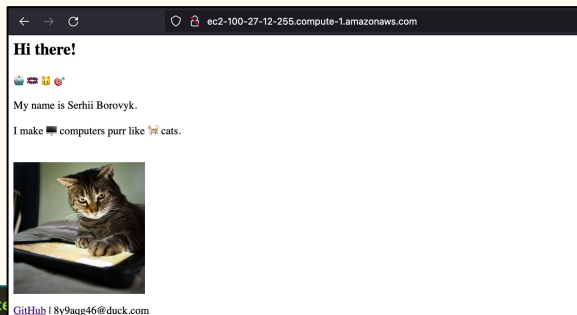
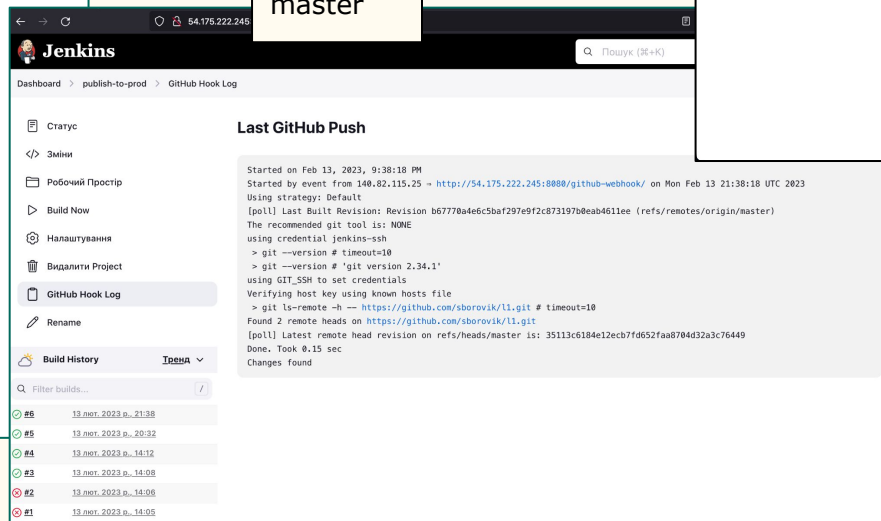
I make 🖥️ computers purr like 🐱 cats.

[GitHub](#) | 8y9ag46@duck.com

```
Sergey@poo: ~/lib/11$ git log
commit 35113c6184e12ecb7fd652faa8704d32a3c76449 (HEAD -> stage, origin/stage, origin/master, origin/HEAD, master)
Author: Serhii Boorvyk <sb@avtolidier-ua.com>
Date: Mon Feb 13 23:36:19 2023 +0200

    merge stage master
```

Merge to  
master



[GitHub](#) | 8y9ag46@duck.com

Prod server  
after

Prod server  
before

# Summary

In summary, my CI/CD Pipeline is an effective tool for automating the web application development process.

Of course, the process can always be improved.

In this case, for example, create a Docker image, publish the image to the Docker Hub, and deploy it to the job server.

Add additional nodes to perform tasks and test the code.

# Thank You

Thank you for your time and attention.

I remain immersed in continuous improvement and study of innovative solutions, implementation of new technologies in my work.