

Leveraging Continuous Integration and Continuous Delivery (CI/CD) Using GKE



Janani Ravi

CO-FOUNDER, LOONYCORN

www.loonycorn.com

Overview

Jenkins open-source automation server

Orchestrates CI/CD build, test, deploy pipelines

Several advantages to using Jenkins with the GKE

Uses ephemeral build executors to run jobs in a clean environment

Jenkins for CI/CD on the GKE

Continuous Integration

Merging development features into the main code branch regularly, perhaps several times a day. Changes are validated using automated tests. Prevents integration headaches that arise with large code merges.

Continuous Delivery

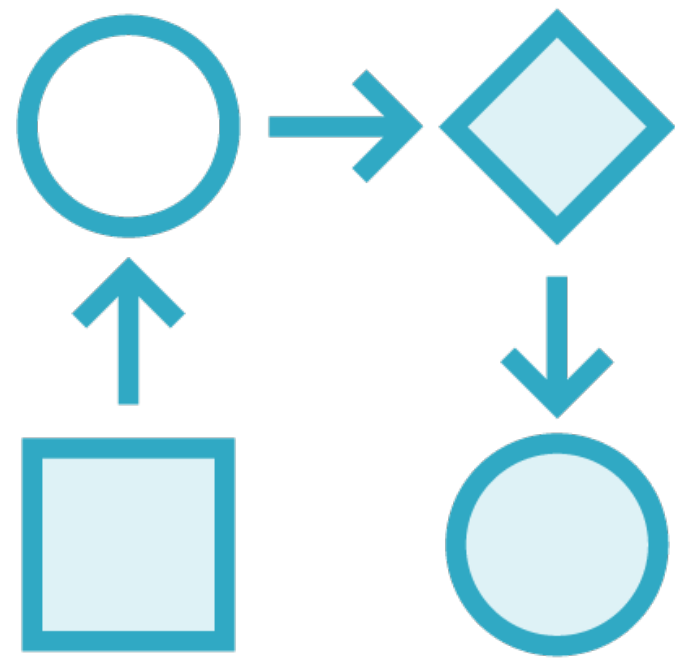
Extension of continuous integration where development occurs in short cycles, allowing features to be reliably released at any time. Automated processes allow releases to occur at greater speed and frequency.

CI/CD

Combining the processes of integration and delivery to quickly release stable and well-tested features

Jenkins

Open source automation server that lets you flexibly orchestrate your build, test and deployment pipelines



Jenkins

Open source automation tool used for CI/CD

Easy to install and configure

Extensible via a plugin architecture

Well integrated with other tools using pre-built plugins

Can run in a distributed cluster of machines



Jenkins on the GKE

Several advantages over running on a cluster of VMs

Use containers to run jobs on different operating systems

GKE uses the Google global load balancer to route web traffic

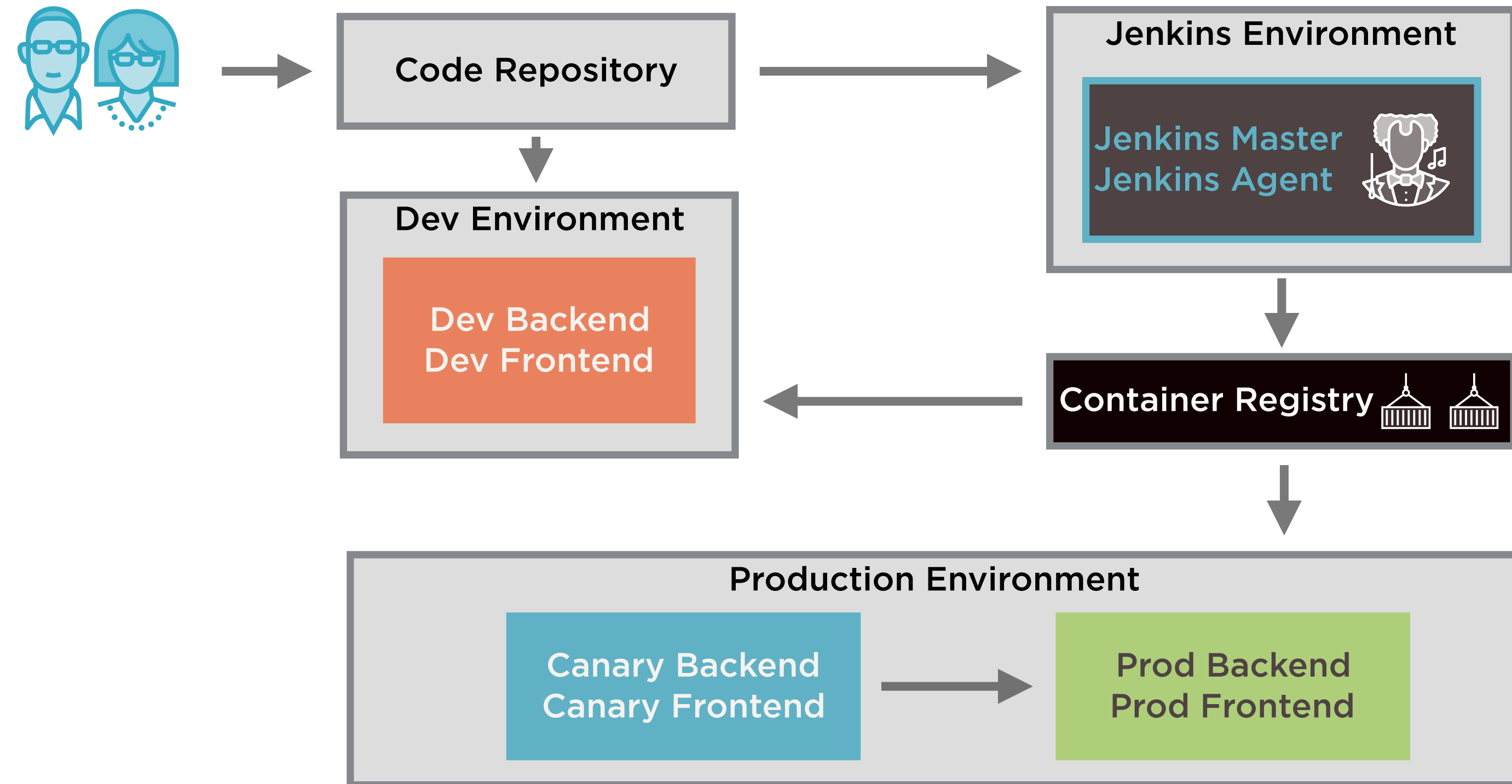


Jenkins on the GKE

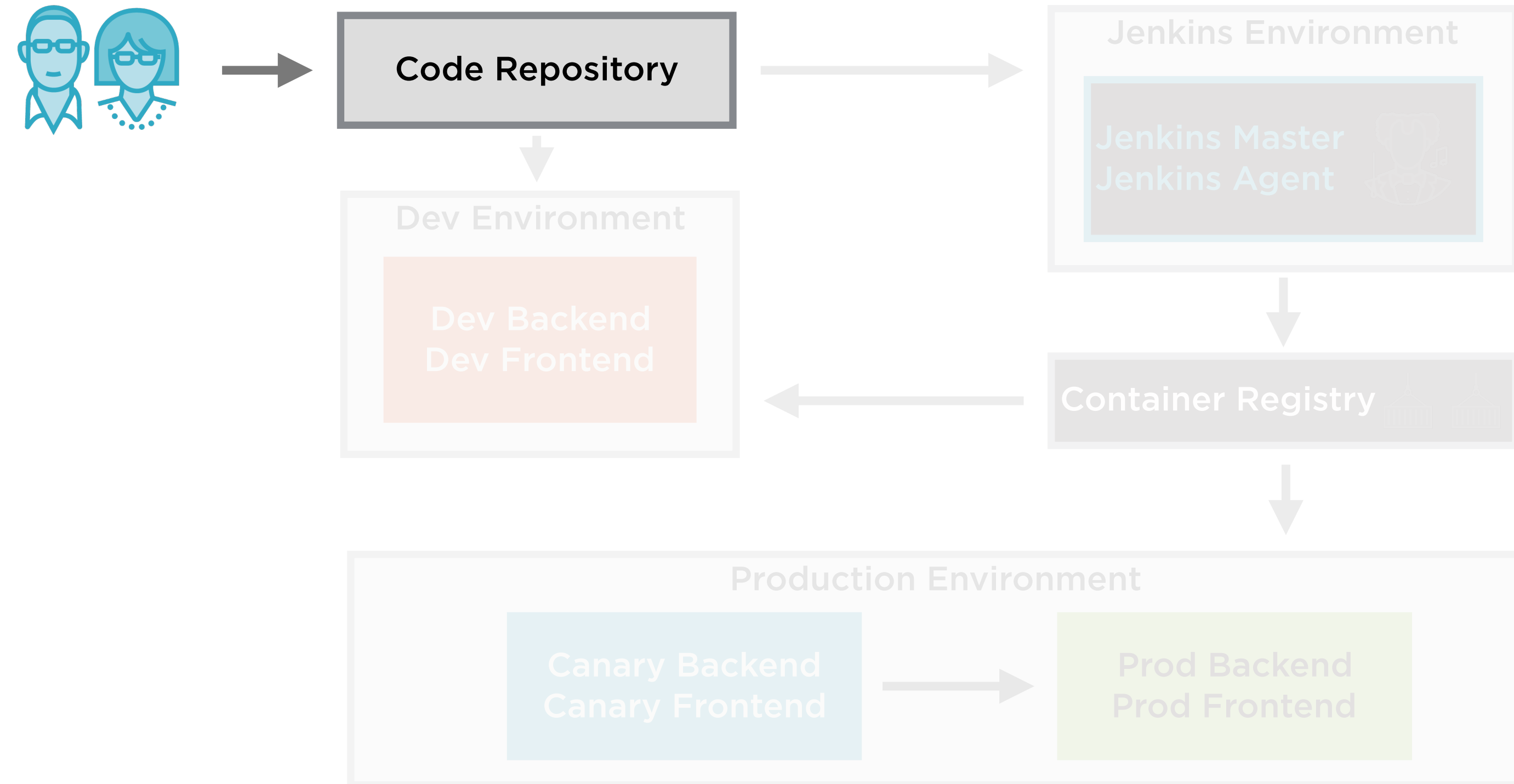
Builds run on ephemeral **build executors**

- Fast launch of builds, in seconds
- Each build runs in a clean environment
- Use resources only when builds are running

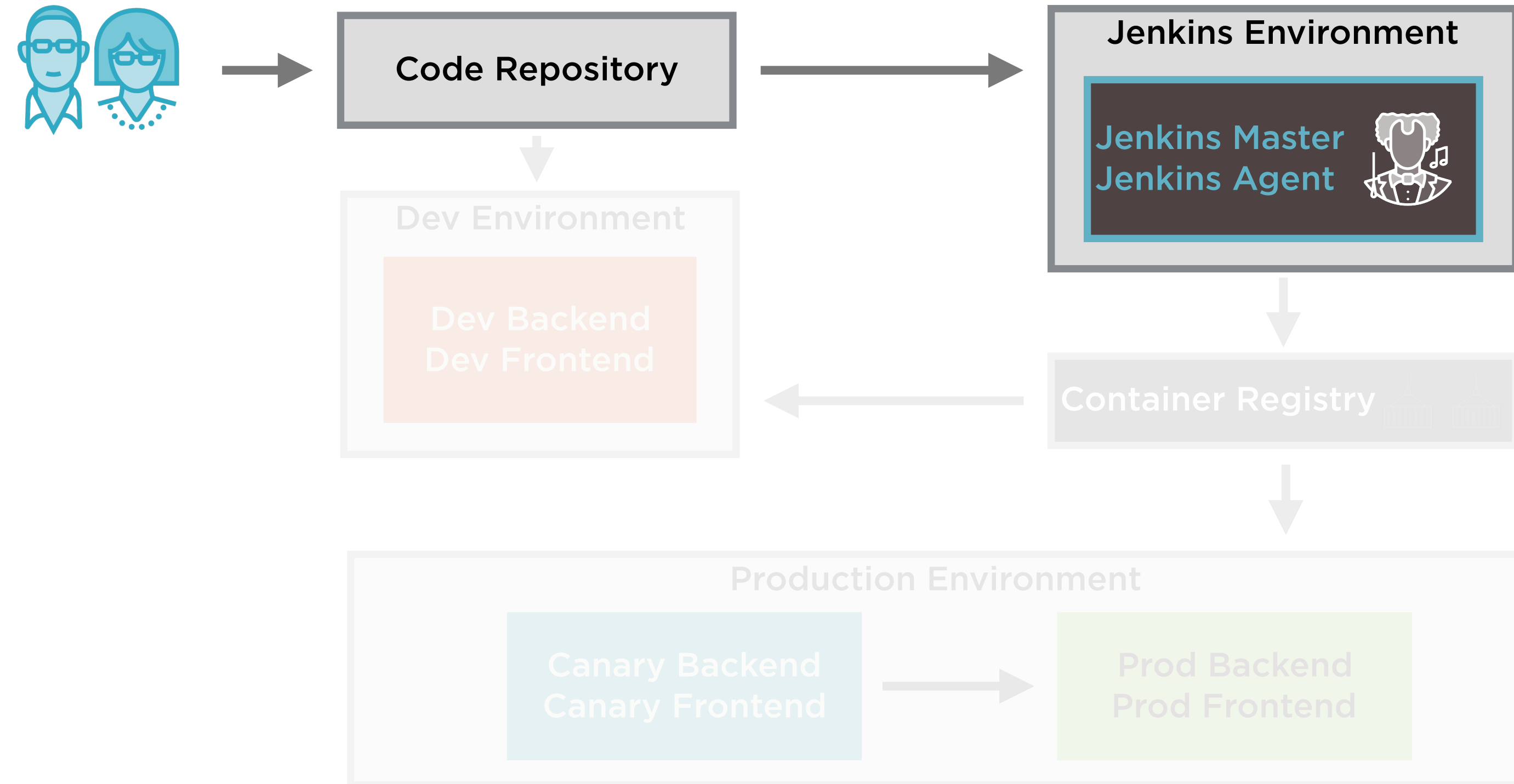
CI/CD with Jenkins on the GKE



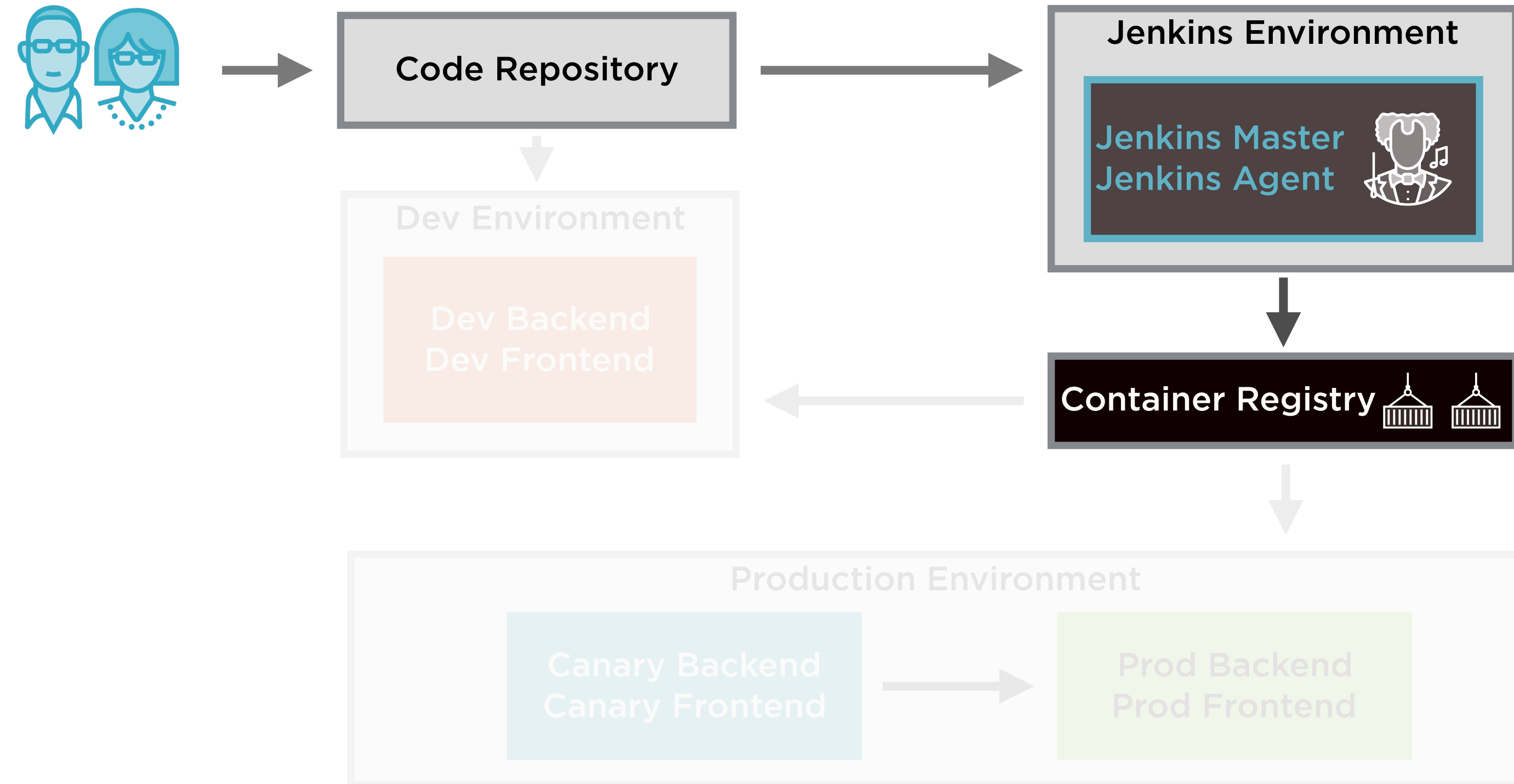
Developer Commits Code



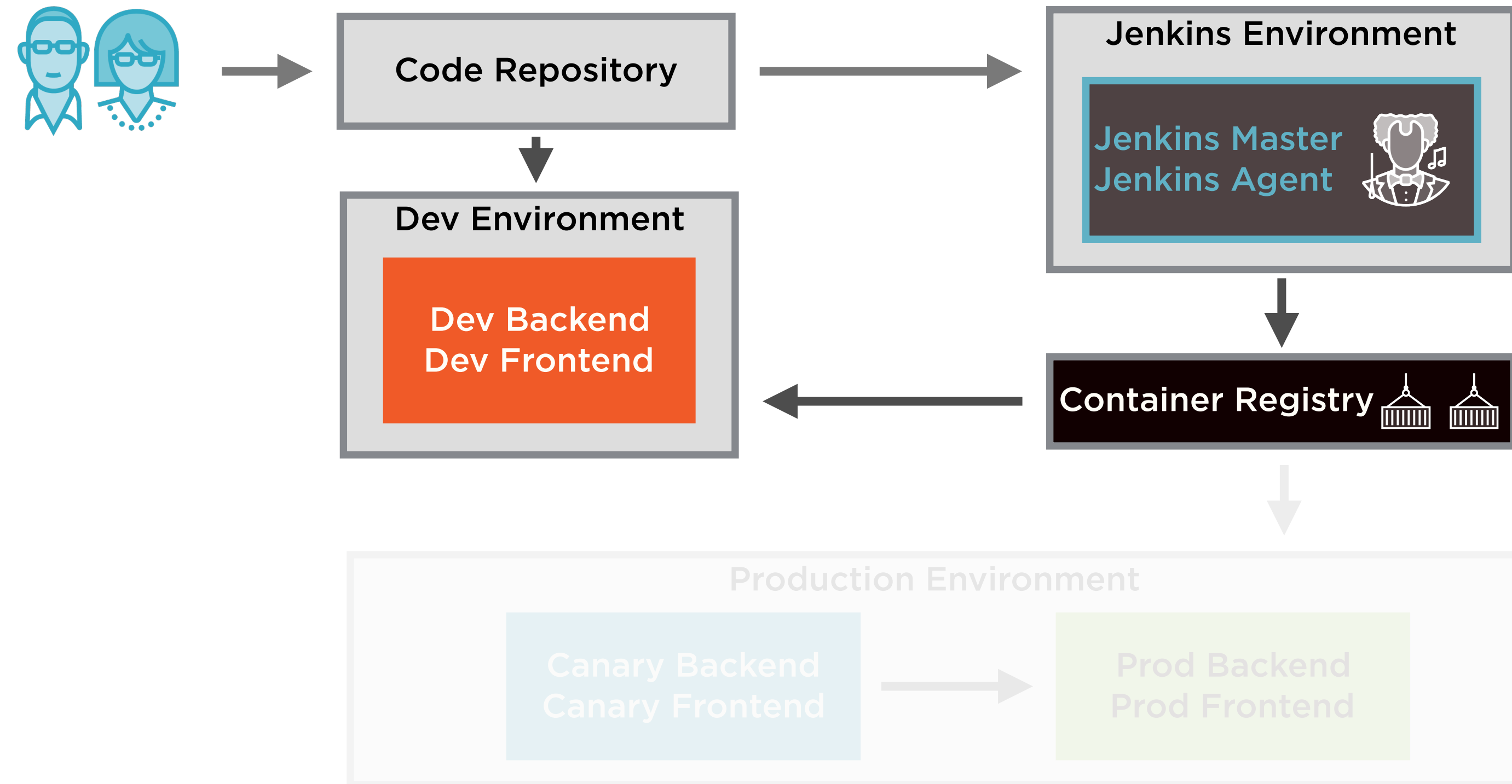
Jenkins on the GKE Builds, then Tests



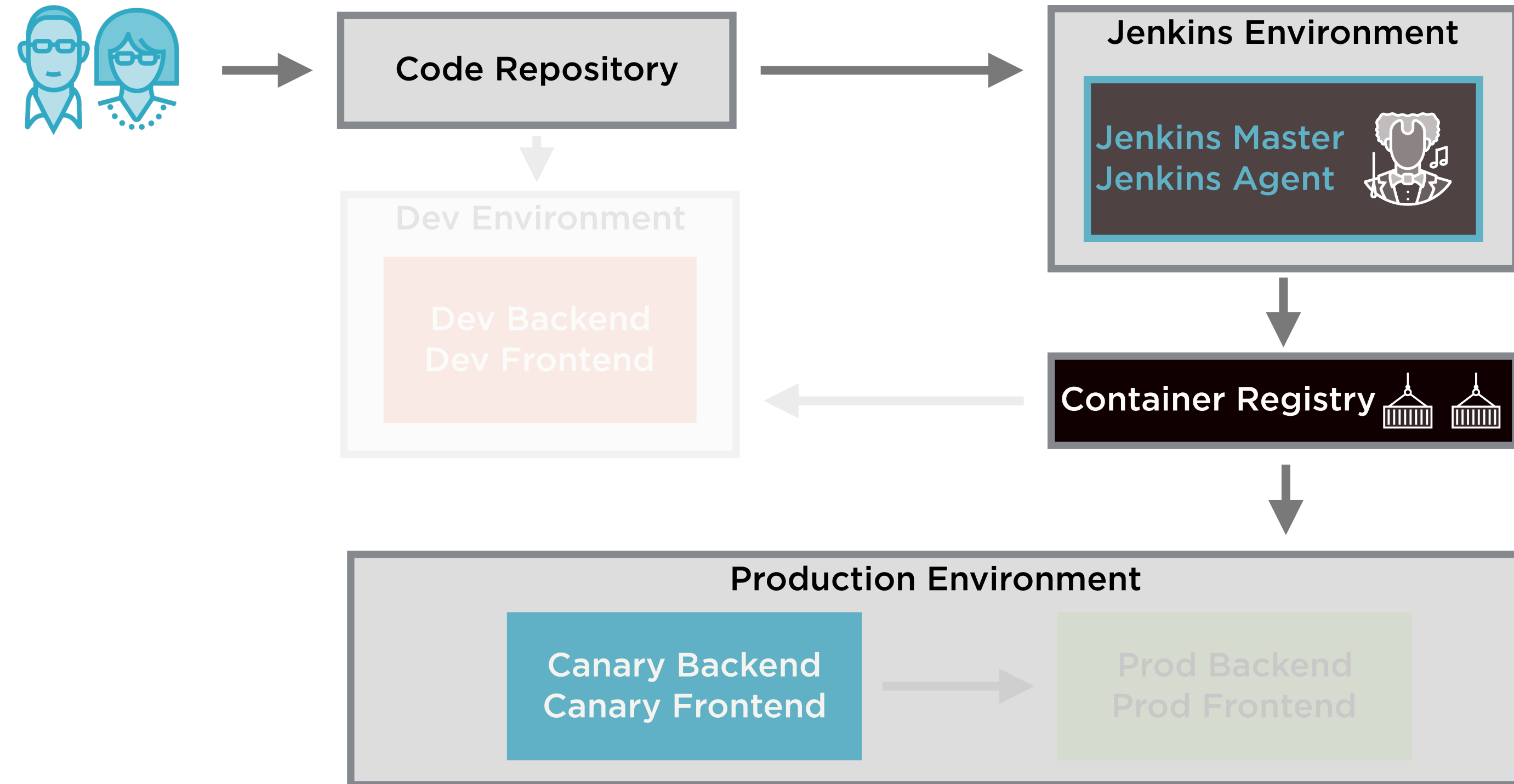
Pushes Build to Container Registry



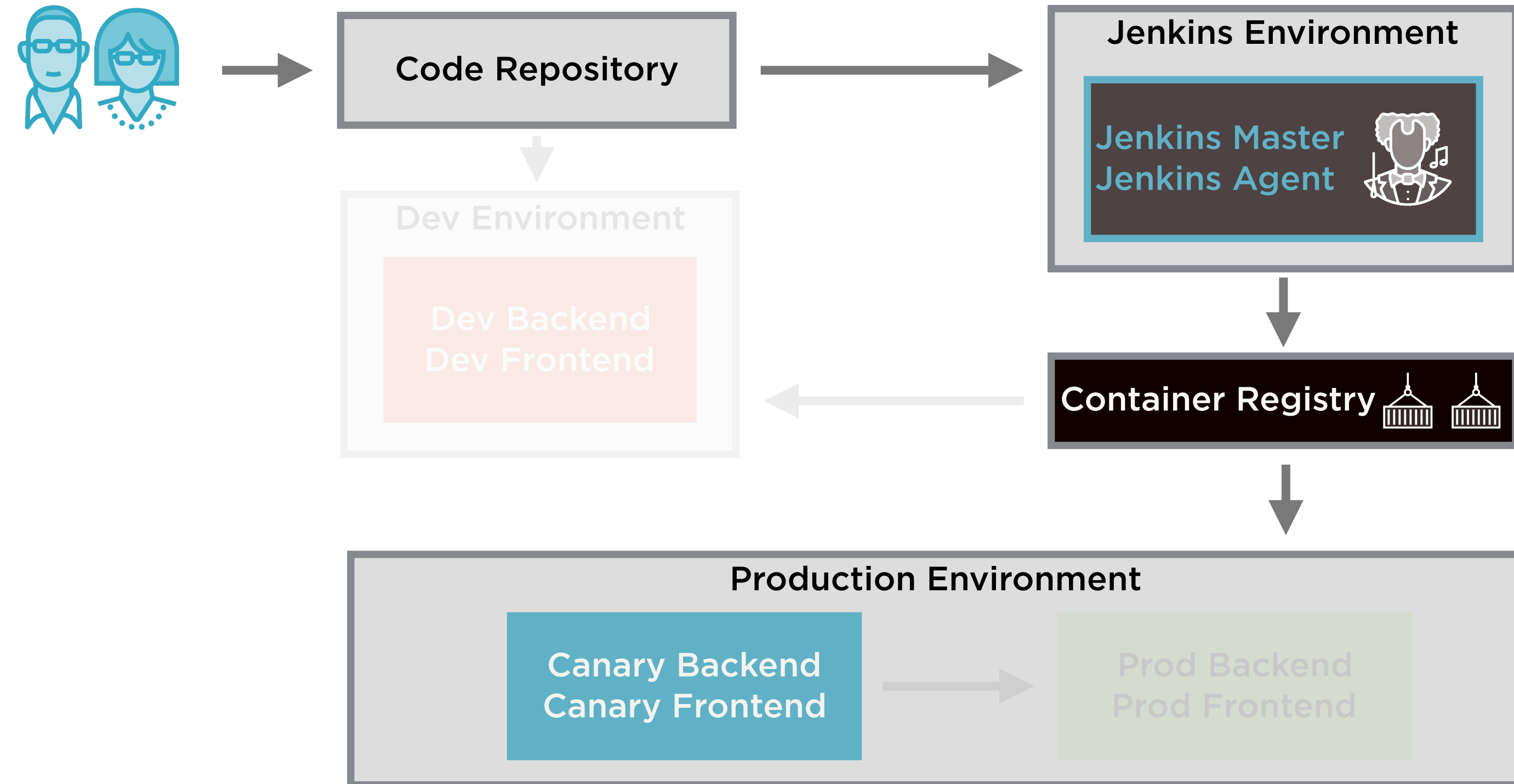
Development Environment: Test and Iterate



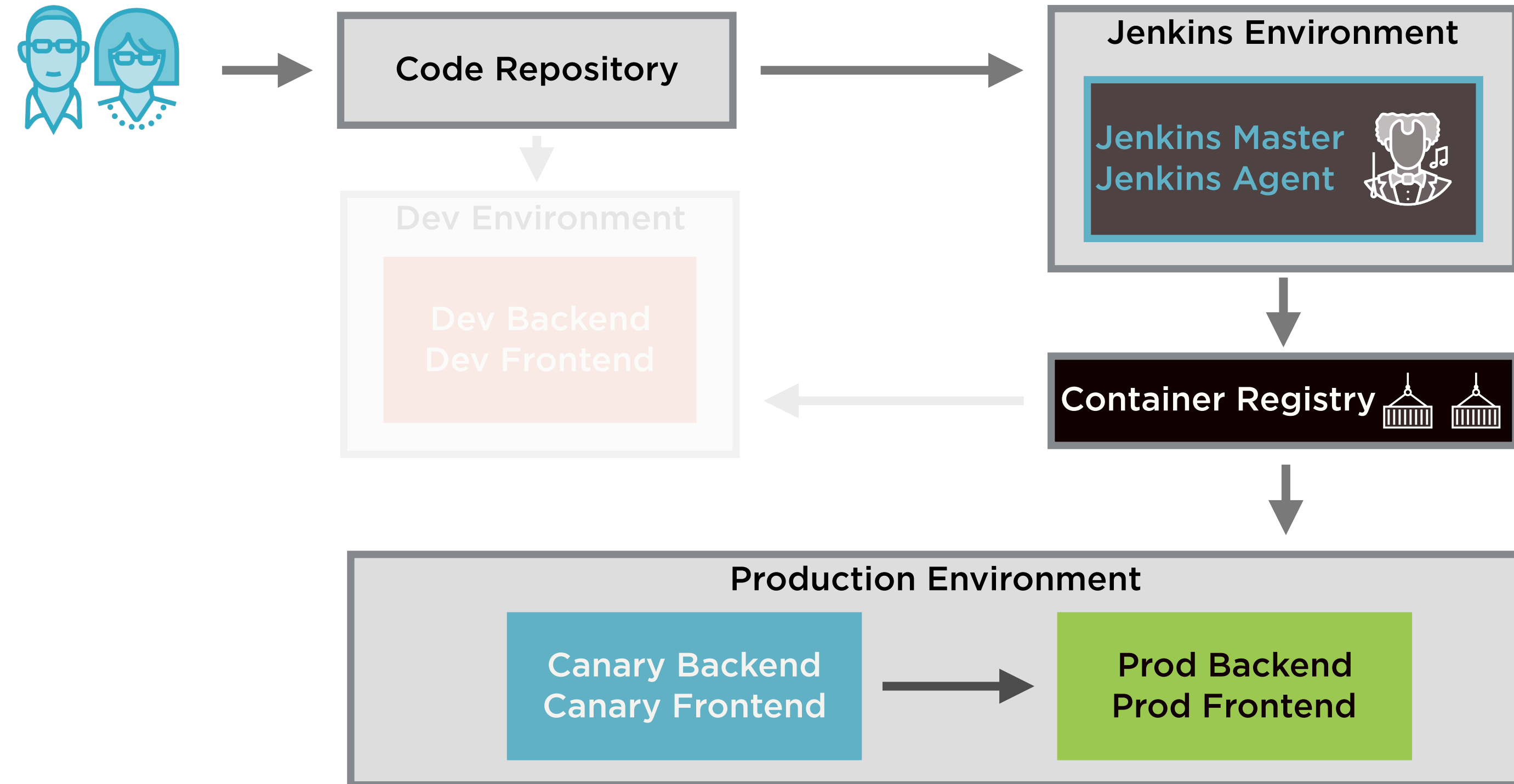
Merge Dev Changes with Canary



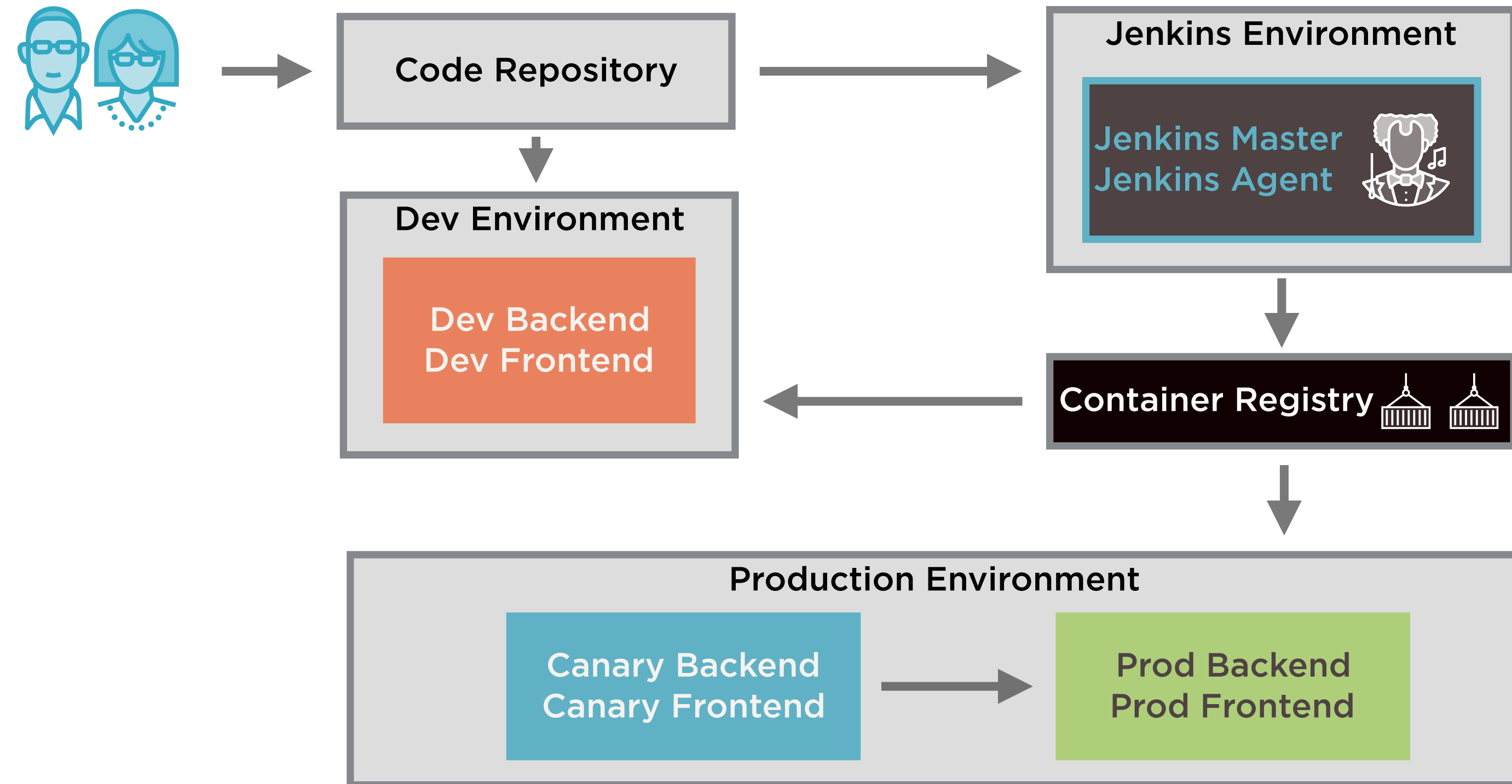
Push to Canary, Test with Subset of Users



Deploy on Production for All Users



Jenkins on the GKE



The gceme Sample Application



gceme

Sample application to run CI/CD pipeline

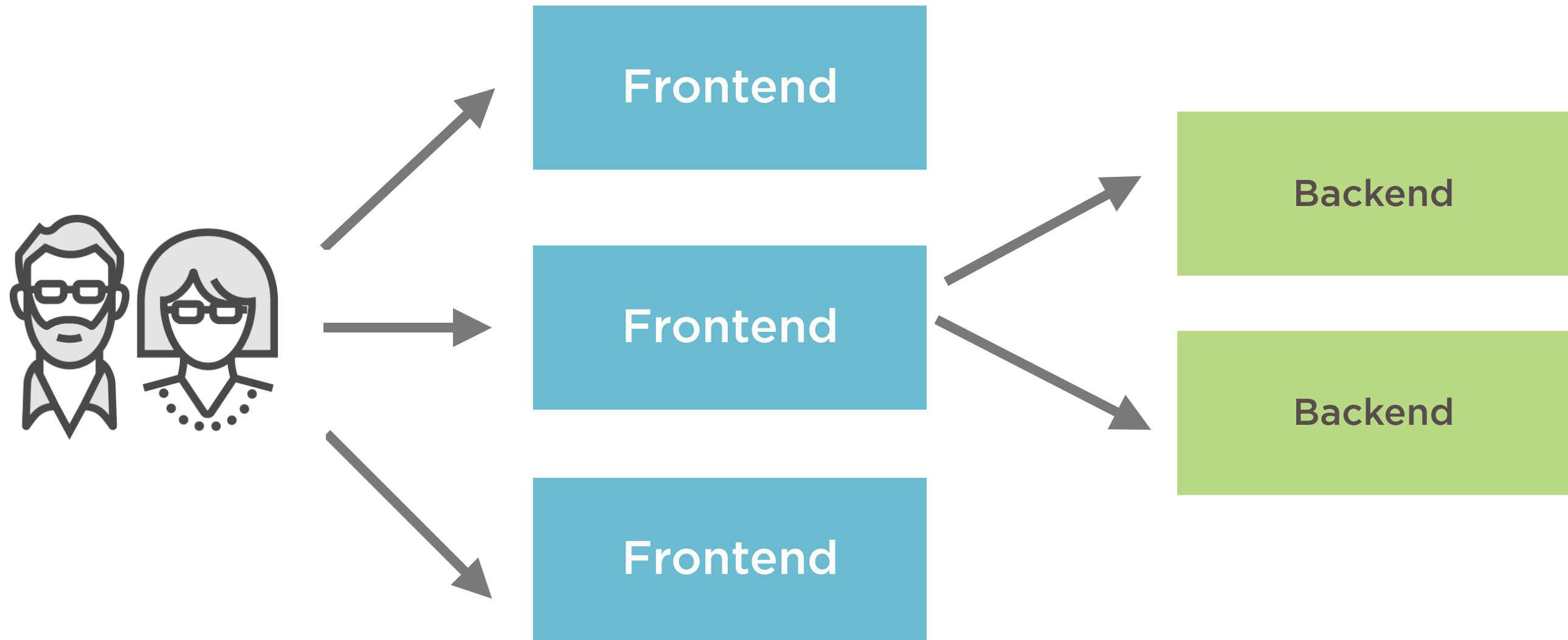
Contains frontend and backend

Code available on GCP's github page

gceme



Frontend and Backend Pods Can Be Scaled



gceme Frontend

Backend details
Name
ID
Zone
Project
etc.

Proxy details
Address
Request
tc.

A simple HTML web UI

Contains 2 cards with details of the VM instance on which the app runs

Backend and proxy information

Connects to the backend on port 8080 to get details

gceme Backend

Backend details

Name

ID

Zone

Project

etc.

Proxy details

Address

Request

etc.

Queries the instance metadata

Returns as a JSON object

Exposes port 8080 for requests

Demo

**Create and build a CI/CD pipeline with
Jenkins on the GKE**

CI/CD with Jenkins on the GKE

Cluster creation

Create a GKE cluster

Configure and install Jenkins

Create Jenkins master and agent running on GKE

Install and set up Helm

Package manager to configure and deploy Kubernetes apps

Connect to Jenkins

Use the admin username and password generated

CI/CD with Jenkins on the GKE

Deploy sample app to cluster

Deploy the production and canary frontend and backend

Create a Jenkins pipeline

Connect Jenkins with source repo; deploy to production

Merge canary with master

Changes will now be rolled out to all of production

Push code to Cloud Source Repositories

Create a new repo and push the sample application to this repo

Create canary branch

Jenkins will push changes to the canary i.e. subset of users

Summary

Use Jenkins to orchestrate CI/CD build, test, deploy pipelines on the GKE

GKE uses ephemeral build executors to run jobs in a clean environment

Integrate Jenkins with Cloud Source Repositories to push to canary and production



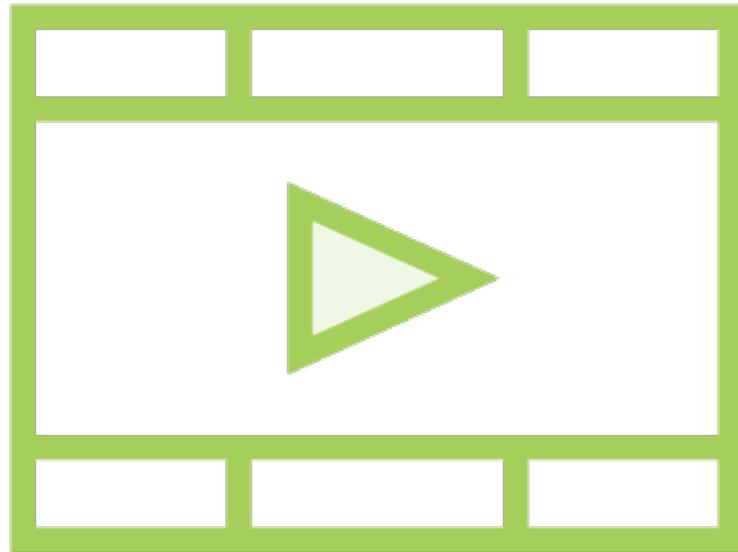
Delete and Clean Up Resources

Delete GKE clusters

Delete VM instances

Delete repo in Cloud Source Repositories

Related Courses



Azure Kubernetes Service (AKS) - The Big Picture

Getting Started With Jenkins Continuous Integration