

# Multistack App on Kubernetes

Deploying a Full-Stack Voting  
Application to AWS EKS

**Technologies:** Python • Node.js • .NET •  
Redis • PostgreSQL

**Platform:** Kubernetes on AWS EKS



# Project Overview

## The Challenge

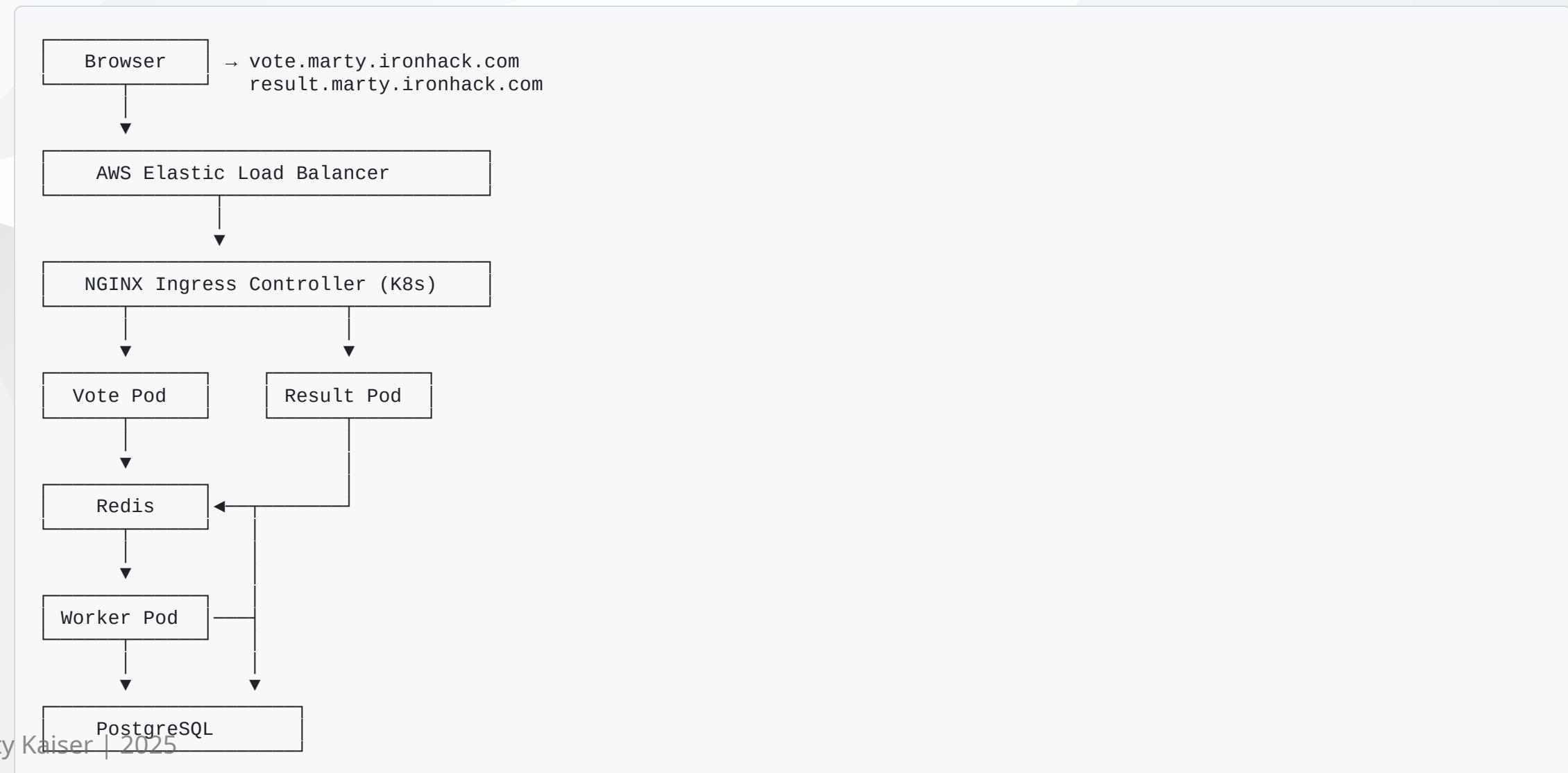
Deploy a **multi-language microservices application** to Kubernetes with:

- Real-time voting system
- Message queue processing
- Database persistence
- Production-ready infrastructure

## Tech Stack

- **Vote Service:** Python/Flask
- **Worker Service:** .NET 7
- **Result Service:** Node.js/Express

# Architecture & Infrastructure



# CI/CD Workflow

## GitHub Actions Pipeline

1. **Trigger:** Push to `main` branch

2. **Build Phase:**

- Build Docker images for `vote`, `worker`, `result`
- Push to Docker Hub ( `kaiserdatal/*:latest` )

3. **Deploy Phase:**

- Configure AWS credentials
- Connect to EKS cluster ( `ironhack-main-2` )
- Create Kubernetes secrets from GitHub Secrets
- Apply manifests ( `kubectl apply -f K8s/` )

4. **Result:** Automatic deployment to production

# Problem 1: DNS Resolution Chaos

## The Issue

Browser: "DNS\_PROBE\_FINISHED\_NXDOMAIN"

## Root Causes

1. **Outdated /etc/hosts entry** pointing to deleted ELB

```
# ❌ Old entry  
ac4ad35e3a4964c368b127366d7be51a...elb.amazonaws.com marty.ironhack.com
```

2. **Hostname mismatch:** Ingress configured for `marty-v1.ironhack.com`
3. **Path rewriting issues:** Flask/Node apps expect `/` not `/vote`

## Solution

# **Problem 2: Taylor Swift vs Lady Gaga**

**Or: The Wildcard Ingress War**



# The Wildcard Ingress Battle

## What Happened?

Accessing `vote.marty.ironhack.com` showed **Jakob's voting app** (Taylor Swift vs Lady Gaga) instead of mine (Cats vs Dogs)!

## The Culprit

```
# Jakob's Ingress
spec:
  rules:
    - http: # ← No "host:" = matches EVERYTHING!
      paths:
        - path: /vote
```

## The Fix

Marty Kaiser | 2025  
# My Ingress - Specific hostnames

# Problem 3: Hardcoded Connections Everywhere

## Vote App (Flask)

```
# ❌ Before  
Redis(host="redis")  
  
# ✅ After  
Redis(host=os.getenv('REDIS_HOST', 'redis'))
```

## Worker App (.NET)

```
# ❌ Wrong env vars  
POSTGRES_HOST, POSTGRES_USER...  
  
# ✅ Correct vars  
DB_HOST, DB_USERNAME, DB_PASSWORD...
```

# Solutions Applied

## Security

- GitHub Secrets → Kubernetes Secrets
- No credentials in code/manifests
- Automated secret injection via CI/CD

## Networking

- Subdomain-based routing
- Proper service discovery ( `marty-svc-*` )
- NGINX Ingress with explicit hostnames

## Configuration

# Summary & Key Learnings

## What We Accomplished

- ✓ Deployed multi-language microservices to Kubernetes
- ✓ Configured AWS ELB + NGINX Ingress routing
- ✓ Implemented secure secret management
- ✓ Automated CI/CD pipeline with GitHub Actions
- ✓ Real-time voting system with WebSocket updates

## Technical Skills Demonstrated

- **Kubernetes:** Deployments, Services, Ingress, Secrets
- **AWS:** EKS, ELB, IAM permissions
- **Docker:** Multi-stage builds, image management

# Is Kubernetes easy?

Questions?



# Thank You!

## Project Repository

<https://github.com/kaiser-data/marty-voting-app>

## Live Application

-  **Vote:** <http://vote.marty.ironhack.com>
-  **Result:** <http://result.marty.ironhack.com>

## Documentation

- Complete troubleshooting guide
- GitHub Secrets setup
- CI/CD workflow documentation