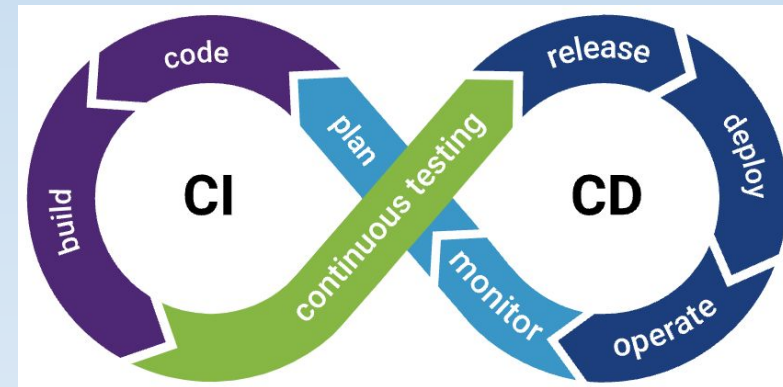




DevOps Atölyesi



Hakan BAYRAKTAR

Jenkins ile Blue/Green Deployment (Java App - AWS EKS)

- Kubernetes ve AWS EKS Temel Bilgiler
- Deployment Stratejileri & Blue/Green Mimarisi
- Maven, Sonarqube, Trivy Hakkında Kısa Bilgiler
- Bitirme Ödevi Tanıtımı
- **Uygulamalı Proje:** Java Uygulamasının Jenkins'de Blue/Green Metodoloji Kullanılarak AWS EKS Kubernetes Cluster Ortamına Deploy Edilmesi

Kubernetes ve AWS EKS Temel Bilgiler





Kubernetes Nedir?

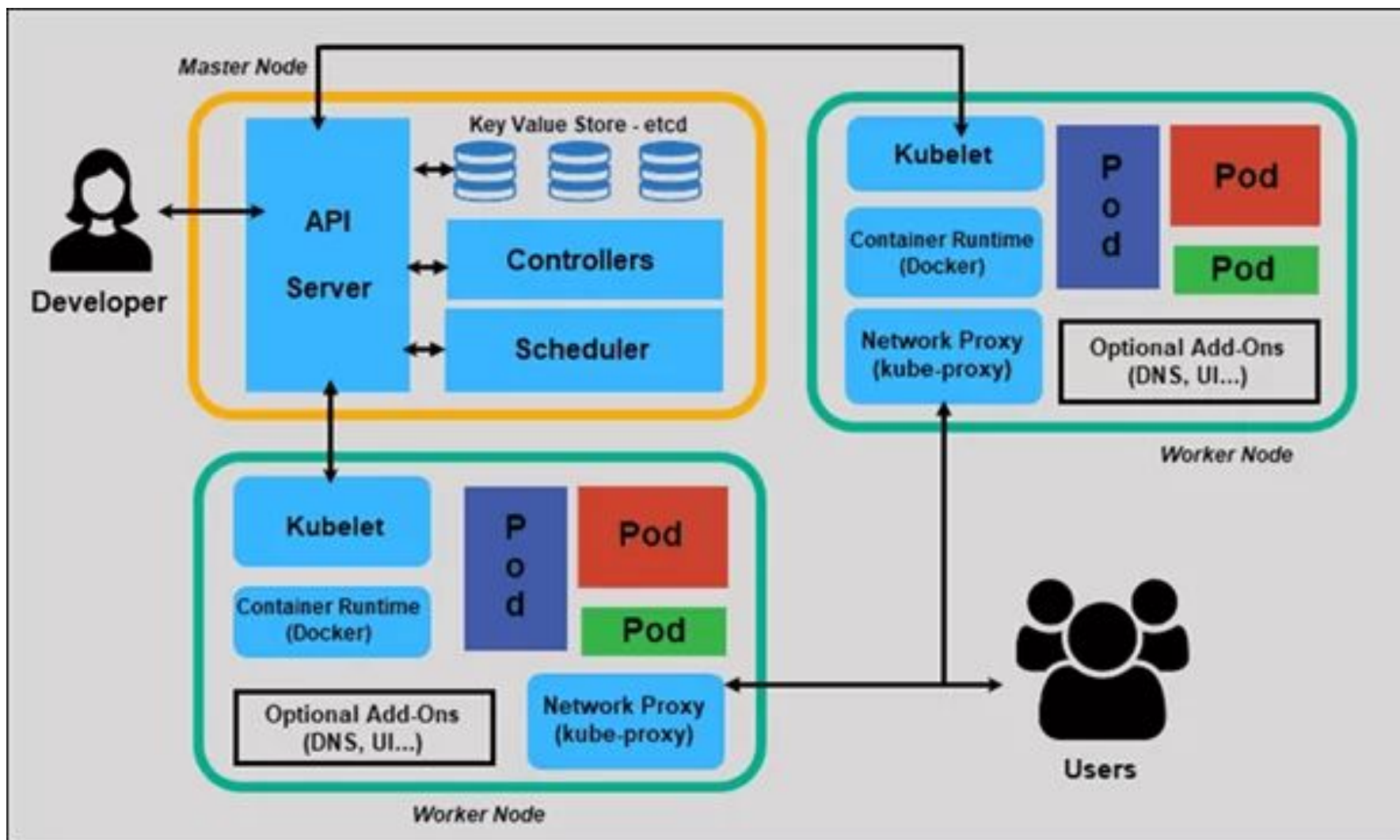
Kubernetes (K8s), **container orchestration** platformudur. Uygulamaları **ölçeklenebilir, taşınabilir ve otomatik** hale getirir.

Temel Özellikleri

- **Otomatik Ölçeklendirme**
- **Self-Healing**
- **Load Balancing**
- **Rolling Updates**
- **Service Discovery**



Kubernetes Mimarisi





Kubernetes Temel Kavramları

Pod: Kubernetes'te en küçük dağıtılabilir birim

Service: Pod'lara erişim sağlar

Deployment: Pod'ları ve ReplicaSet'leri yönetir

ConfigMap: Hassas olmayan konfigürasyon veriler

Secret: Hassas veriler (şifreler, token'lar)

Namespace: Cluster'ı mantıksal olarak böler



AWS EKS Nedir?

Amazon Elastic Kubernetes Service (EKS), yönetilen Kubernetes servisiidir.

Master node'lar AWS tarafından yönetilir, sen sadece Worker node'larla ilgilenirsin.

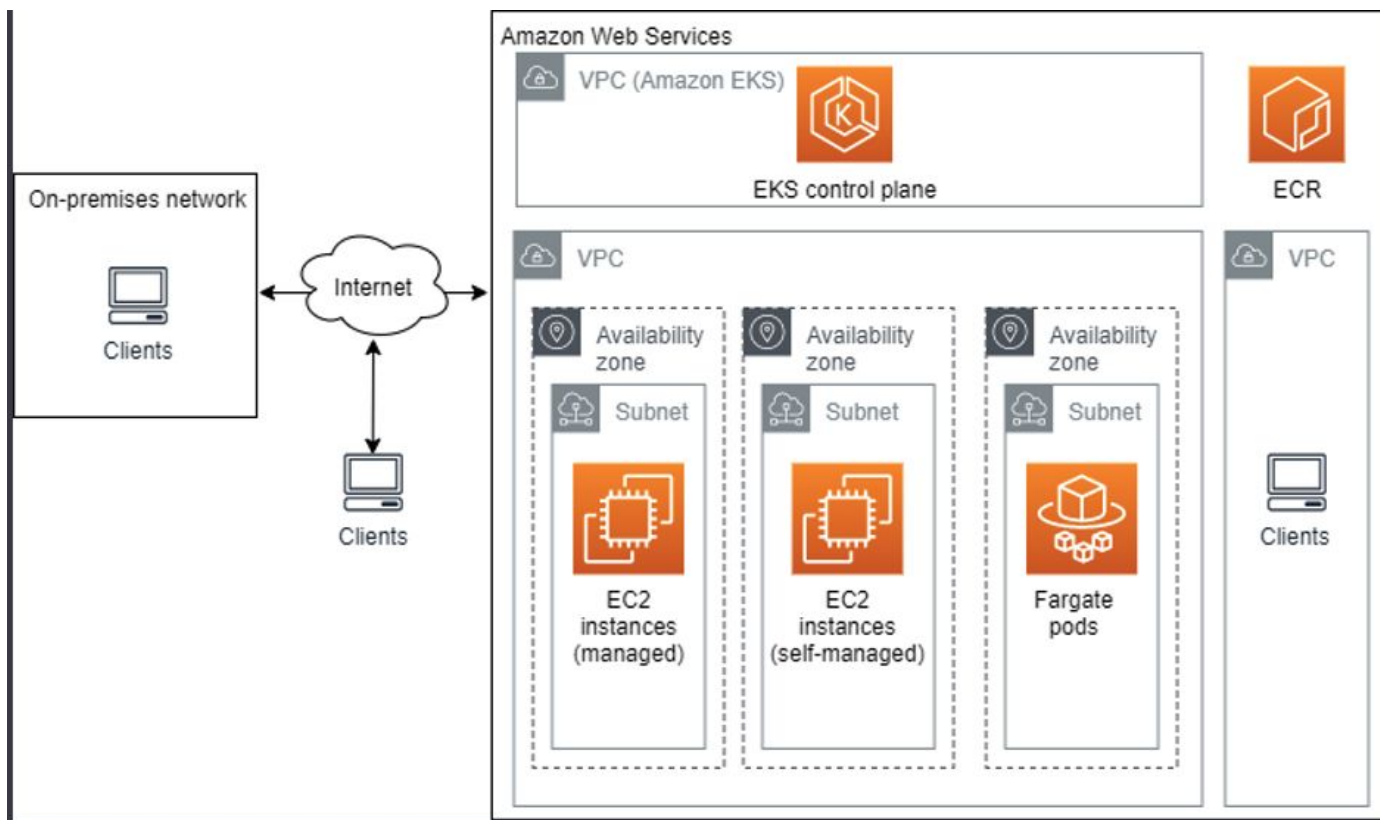
Özellikler:

- Otomatik scaling
- IAM ile güvenli erişim
- VPC entegrasyonu
- Fargate ile serverless pod desteği



EKS Mimarisi Örneği

`kubectl`, Jenkins gibi araçlarla EKS cluster'a deployment yapılır.
Her pod bir uygulama parçasıdır.
EKS, altyapıyı otomatik yönetir ve güvenliği sağlar.





EKS Kurulum ve Yapılandırma



Gereksinimler

- AWS CLI kubectl eksctl (opsiyonel)
- Uygun IAM izinleri

Kurulum Adımları

1. AWS CLI Yapılandırması

```
aws configure
```

2. kubectl Kurulumu

```
curl -LO "https://dl.k8s.io/release/$(curl -L -s  
https://dl.k8s.io/release/stable.txt)/bin/linux/a  
md64/kubectl"
```

```
chmod +x kubectl
```

```
sudo mv kubectl /usr/local/bin/
```

3. EKS Cluster Oluşturma

eksctl ile

```
eksctl create cluster --name my-cluster  
--region us-west-2 --nodegroup-name  
my-nodes --node-type t3.medium --nodes  
3
```

veya AWS Console/CLI ile

```
aws eks create-cluster --name my-cluster  
--version 1.24 --role-arn  
arn:aws:iam::123456789012:role/eks-servi  
ce-role
```

4. kubeconfig Güncelleme

```
aws eks update-kubeconfig --region  
us-west-2 --name my-cluster
```

Deployment Stratejileri & Blue/Green Mimarisi







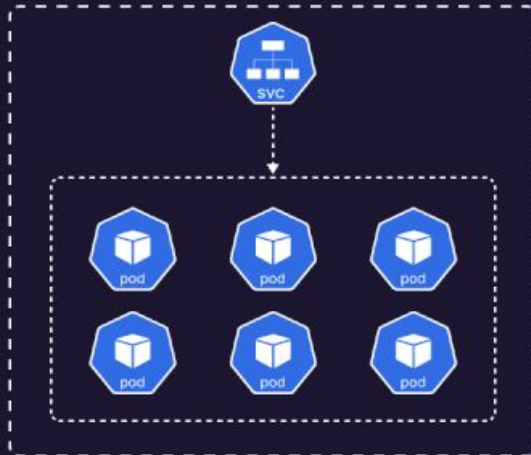
Yaygın Deployment Stratejileri

Strateji	Açıklama	Risk	Geri Dönüş
Recreate	Eski podları kapatıp yenilerini başlatır.	Yüksek	Zor
Rolling Update	Pod'ları sırayla günceller. Sürekli canlılık sağlar.	Orta	Orta
Blue/Green	Yeni sürüm paralel deploy edilir. Trafik yönlendirmesiyle geçiş yapılır.	Düşük	Kolay
Canary	Kullanıcıların küçük kısmına yeni sürüm verilir.	Çok Düşük	Kolay

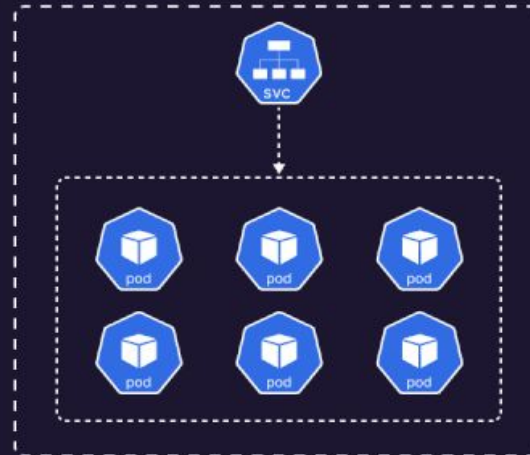


KUBERNETES DEPLOYMENT STRATEGIES

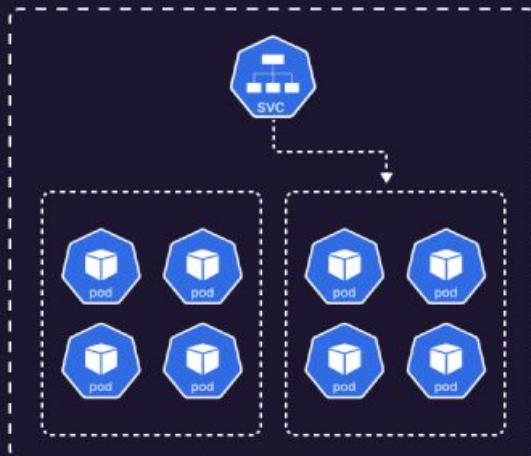
1. RECREATE



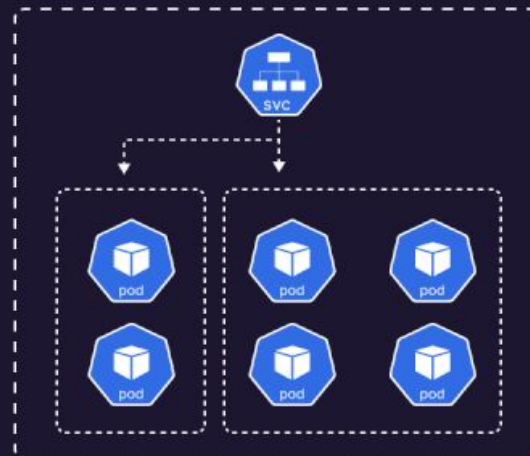
2. ROLLING



3. BLUE/GREEN



3. CANARY





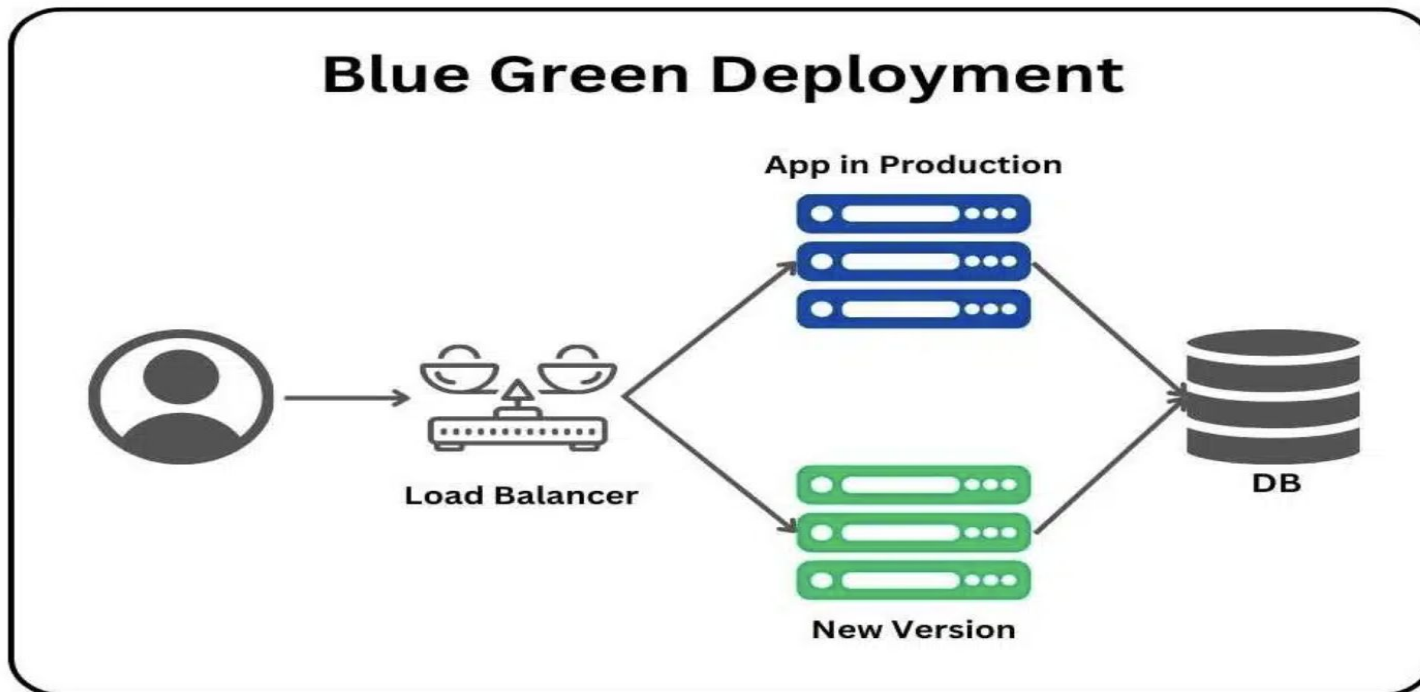
Blue/Green Deployment Nedir?

- Blue/Green Deployment, uygulamanın yeni versiyonunu canlıya kesintisiz şekilde geçirmek için kullanılan bir **deployment** stratejisidir.
- **İki ayrı ortam:** Aynı uygulamanın iki versiyonu (Blue ve Green) aynı anda ama izole çalışır.
- **Geçiş nasıl olur?** Trafik yönlendirmesi Load Balancer üzerinden yapılır.
- **Rollback kolaylığı:** Yeni sürümde hata varsa sadece yönü değiştirerek eskiye dönülebilir.



Mimaride Blue/Green Nasıl Konumlanır?

Başlangıçta tüm trafik **Blue** ortama yönlendirilir.
Green ortama yeni sürüm deploy edilir, test edilir.
Test başarılıysa trafik **Green** ortama kaydırılır.





Neden Blue/Green?

- ✓ **Canlı sistem etkilenmeden** test yapılabilir.
- ✓ **0 downtime**, yani kesintisiz deploy.
- ✓ Geri dönüş süresi **saniyelerle sınırlı**.
- ✓ Uygulama **ayrı ortamda** izlenebilir ve izole edilir.

Maven, SonarQube, Trivy Hakkında Kısa Bilgiler





Maven Nedir?

Maven, Java projeleri için kullanılan bir **build otomasyon** aracıdır.

- Proje bağımlılıklarını, derlemeyi, testleri ve paketlemeyi yönetir.





Öne Çıkan Özellikleri:

- `pom.xml` dosyası ile proje yapılandırması yapılır.
- Otomatik bağımlılık yönetimi sağlar.
- Proje yaşam döngüsü (Lifecycle) aşamalarını kontrol eder:
`clean` → `compile` → `test` → `package` → `install` → `deploy`



SonarQube Nedir?

SonarQube, kod kalitesi ve güvenliği için kullanılan bir **statik analiz** aracıdır.

- Kodunuzu analiz eder ve şunları raporlar:
 - a.  Hatalar (Bugs)
 - b.  Güvenlik açıkları (Vulnerabilities)
 - c.  Kod kokuları (Code Smells)
 - d.  Test kapsamı (Coverage)

Kullanımı:

- Maven ile entegre edilebilir:
`mvn sonar:sonar` komutu ile analiz tetiklenir.
- Jenkins pipeline içinde otomatik analiz yapılabilir.



Trivy Nedir?

Trivy, container imajları ve kodda güvenlik açıklarını bulmak için kullanılan hafif bir **vulnerability scanner** aracıdır.

Taradığı şeyler:

- Docker image'larındaki CVE'ler
- Uygulama bağımlılıkları
- Misconfigurations (Kubernetes YAML, Terraform dosyaları vs.)

Kullanımı:

```
trivy image my-app:latest
```

CI/CD'de Kullanımı:

- Jenkins pipeline'a kolayca entegre edilir.
- Güvensiz imajlar prod ortamına geçmeden önce engellenebilir

Bitirme Ödevi Tanıtımı





Bitirme Ödevi – DevOps Projesi

✓ Ödev Seçenekleri

1. DevOps Atölyesinde Yaptığınız Projelerden Biri

- Daha önce birlikte yaptığımız projelerden herhangi birini seçip yeniden kurabilir, pipeline oluşturabilirsiniz.

2. Yeni Bir CI/CD Projesi Geliştirme

- GitHub Actions **veya** Jenkins kullanarak, herhangi bir uygulamayı
- Aşağıdaki ortamlardan birine deploy edilecek şekilde yapılandırılmalı:
 - Dilerseniz projenizi tamamen **kendi bilgisayarınızda** geliştirebilirsiniz.
 - Docker
 - Kubernetes (Kind, Minikube, veya AWS EKS, GKE Digital Ocean..)
 - Cloud sunucu(EC2..)



Bitirme Ödevi – DevOps Projesi

✓ Bitirme Ödevi Teslimi:

Bitirme ödevinizi tamamladıktan sonra, şu bilgileri içeren bir PDF dosyası oluşturmalsınız:

- Adınız Soyadınız
- Telefon Numaranız
- E-posta Adresiniz
- Projenizin GitHub Linki
- Projenizin çalıştığını gösteren ekran görüntüleri

Hazırladığınız PDF dosyasını, size vereceğim Google Drive adresine yükleyerek teslim edebilirsiniz.



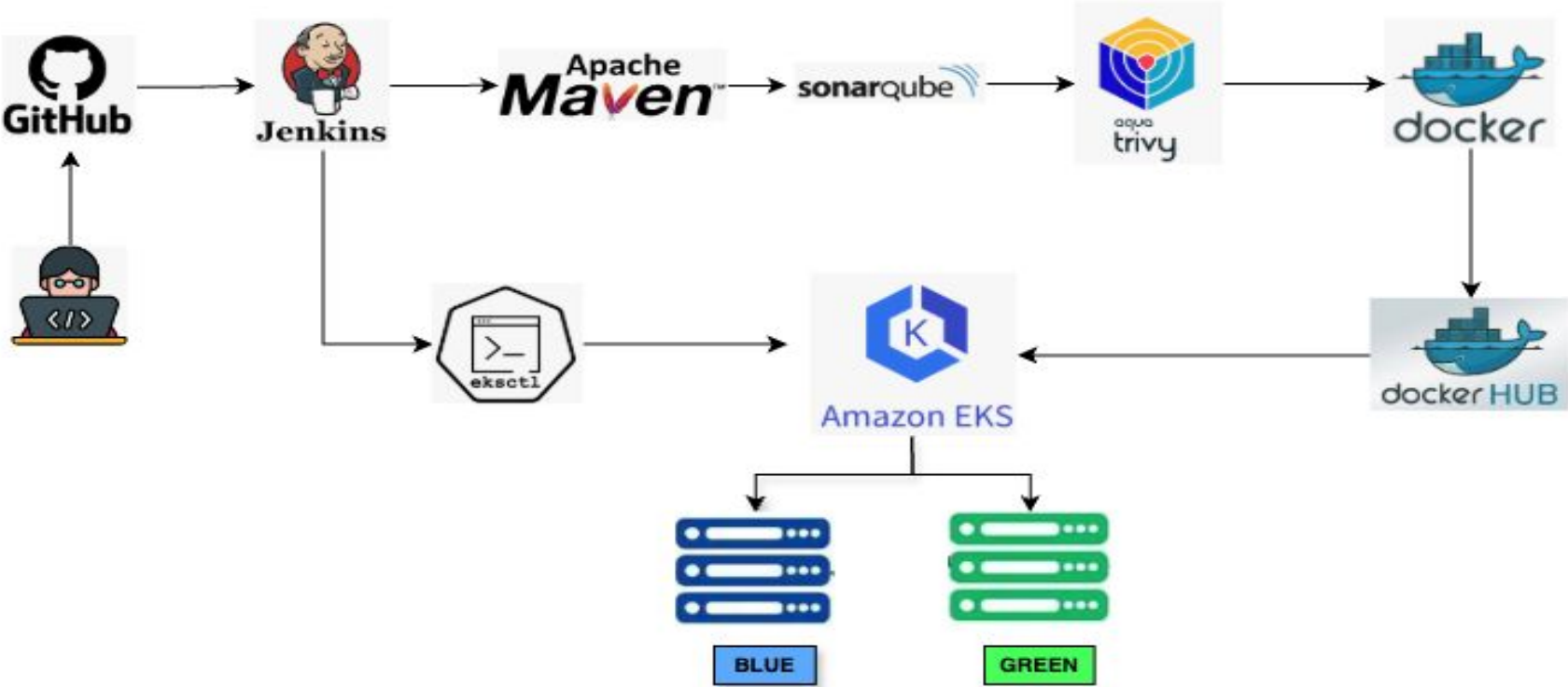
Jenkins ile Blue/Green Deployment (Java App - AWS EKS)





Jenkins ile Blue/Green Deployment

Blue-Green Deployment



Jenkins ile Blue/Green Deployment – Sıfır Kesintili Java Uygulaması Yayını (AWS EKS)




Proje Amacı

- Downtime olmadan yeni sürüm geçişi (Blue → Green)
- Jenkins ile otomatik CI/CD süreci
- Kod kalitesi ve güvenlik taraması
- AWS EKS üzerinde canlı ortam

Bileşenler

- Java (Spring Boot), Maven, Docker, SonarQube, Trivy
- Jenkins Pipelines, Docker Hub, AWS EKS, ALB
- Yayın stratejisi: **Blue/Green Deployment**



Eğitime ait tüm sunum, döküman ve materyaller için:

<https://github.com/hakanbayraktar/ibb-tech>

Uygulamalı proje kodlarına ulaşmak için:

<https://github.com/hakanbayraktar/blue-green>
adresinden ulaşabilirsiniz



Kurulum Adımları

- **SonarQube kurulumu** → Token alınır, Jenkins'e tanımlanır
- **Jenkins kurulumu** → Giriş ve plugin kurulumu
- **Gerekli Jenkins Credentials** eklenir (AWS, DockerHub, GitHub, Sonar)

CI/CD Süreci (Jenkins Pipelines)

- ♦ **infra-create:** eksctl ile EKS cluster kurulumu
- ♦ **blue-green**
 - Maven build → Docker image
 - SonarQube kalite analizi
 - Trivy güvenlik taraması
 - DockerHub push
 - EKS'e Blue veya Green versiyon deploy
 - ALB ile trafik yönlendirme
- ♦ **infra-destroy:** EKS cluster silme job'ı (onay mekanizmalı)

manuel Temizlik Scriptleri

- **destroy-jenkins.sh**
- **destroy-sonarqube.sh**