

```
@startuml
```

```
title Sequence Diagram - DevOps Workflow Automation Tool
```

```
actor Developer
```

```
participant "GitHub" as GH
```

```
participant "CI Server (Jenkins/GitHub Actions)" as CI
```

```
participant "Docker Engine" as DE
```

```
participant "Cloud Provider (AWS/Azure)" as CP
```

```
Developer -> GH : Push Code
```

```
GH -> CI : Trigger Build Pipeline
```

```
CI -> CI : Run Unit Tests
```

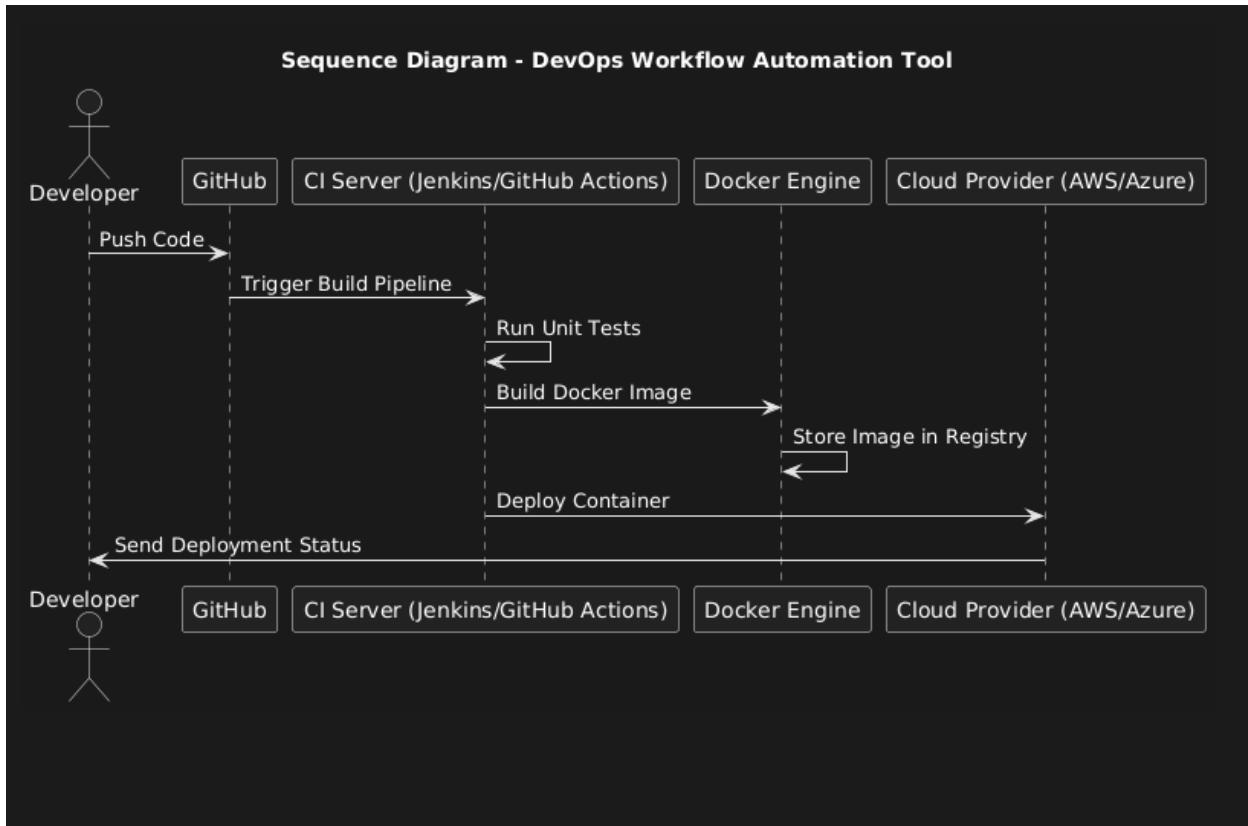
```
CI -> DE : Build Docker Image
```

```
DE -> DE : Store Image in Registry
```

```
CI -> CP : Deploy Container
```

```
CP -> Developer : Send Deployment Status
```

```
@enduml
```



@startuml

title Deployment Diagram - DevOps Workflow Automation Tool

```

node "Developer Machine" {
    artifact "Source Code" as src
}

```

```

node "GitHub Repository" {
    artifact "Code Repo" as repo
}

```

```

node "CI Server (Jenkins/GitHub Actions)" {
    artifact "Build Scripts" as build
    artifact "Pipeline Config" as pipe
}

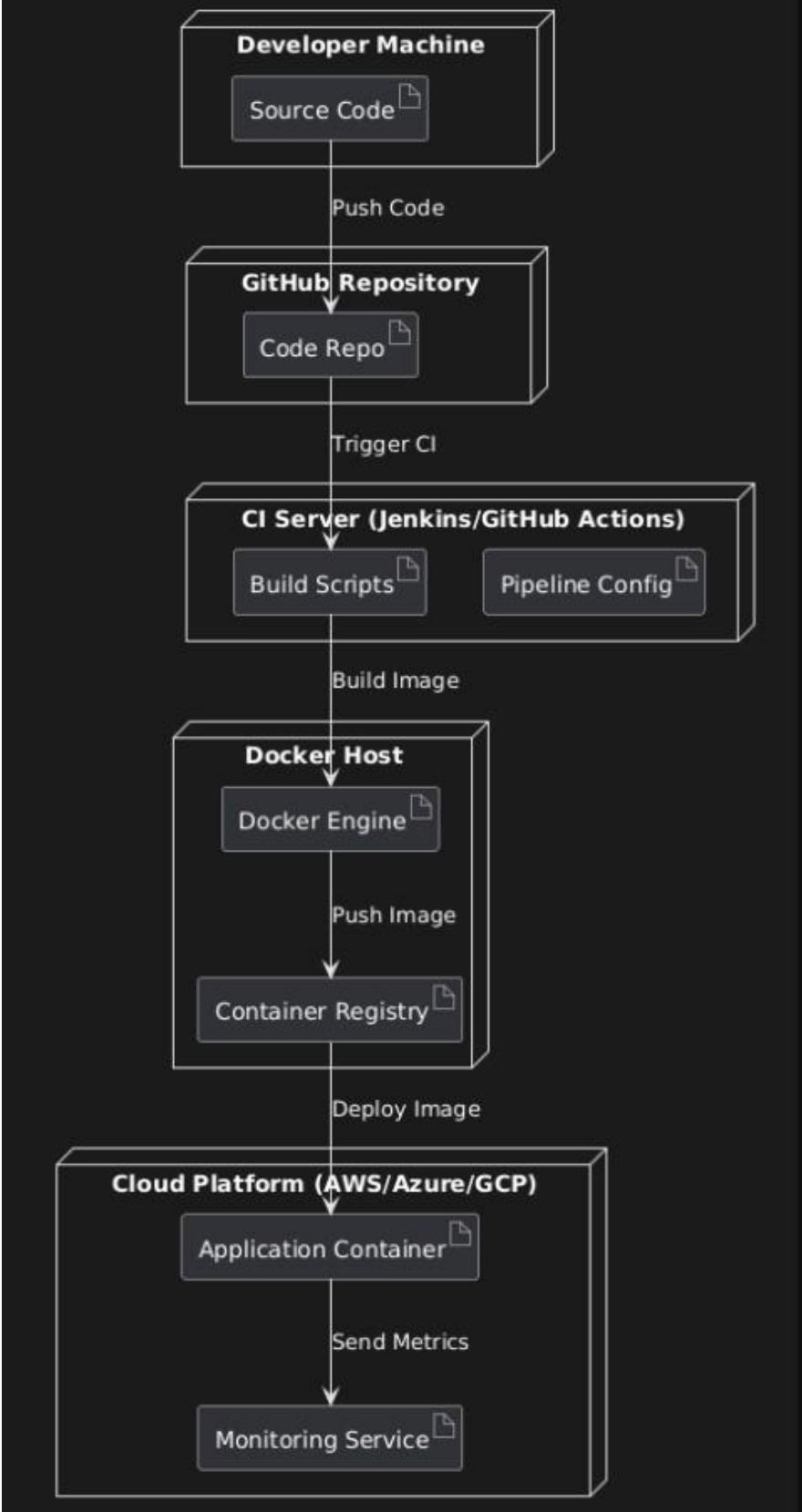
```

```
node "Docker Host" {
    artifact "Docker Engine" as docker
    artifact "Container Registry" as registry
}
```

```
node "Cloud Platform (AWS/Azure/GCP)" {
    artifact "Application Container" as app
    artifact "Monitoring Service" as monitor
}
```

```
src --> repo : Push Code
repo --> build : Trigger CI
build --> docker : Build Image
docker --> registry : Push Image
registry --> app : Deploy Image
app --> monitor : Send Metrics
@enduml
```

### Deployment Diagram - DevOps Workflow Automation Tool



```
@startuml
```

```
title Class Diagram - DevOps Workflow Automation Tool
```

```
class "Repository" {
```

```
    +url: String
```

```
    +branch: String
```

```
    +commitCode(): void
```

```
}
```

```
class "Pipeline" {
```

```
    +id: int
```

```
    +triggerEvent(): void
```

```
    +runBuild(): void
```

```
    +runTests(): void
```

```
    +deploy(): void
```

```
}
```

```
class "DockerManager" {
```

```
    +buildImage(): void
```

```
    +pushImage(): void
```

```
    +deployContainer(): void
```

```
}
```

```
class "CloudManager" {
```

```
    +deployToCloud(): void
```

```
    +monitorService(): void
```

```
}
```

```
class "NotificationService" {
```

```
+sendStatus(): void  
+alertFailure(): void  
}
```

Repository --> Pipeline : triggers >

Pipeline --> DockerManager : uses >

DockerManager --> CloudManager : deploys to >

Pipeline --> NotificationService : notifies >

### Class Diagram - DevOps Workflow Automation Tool

