

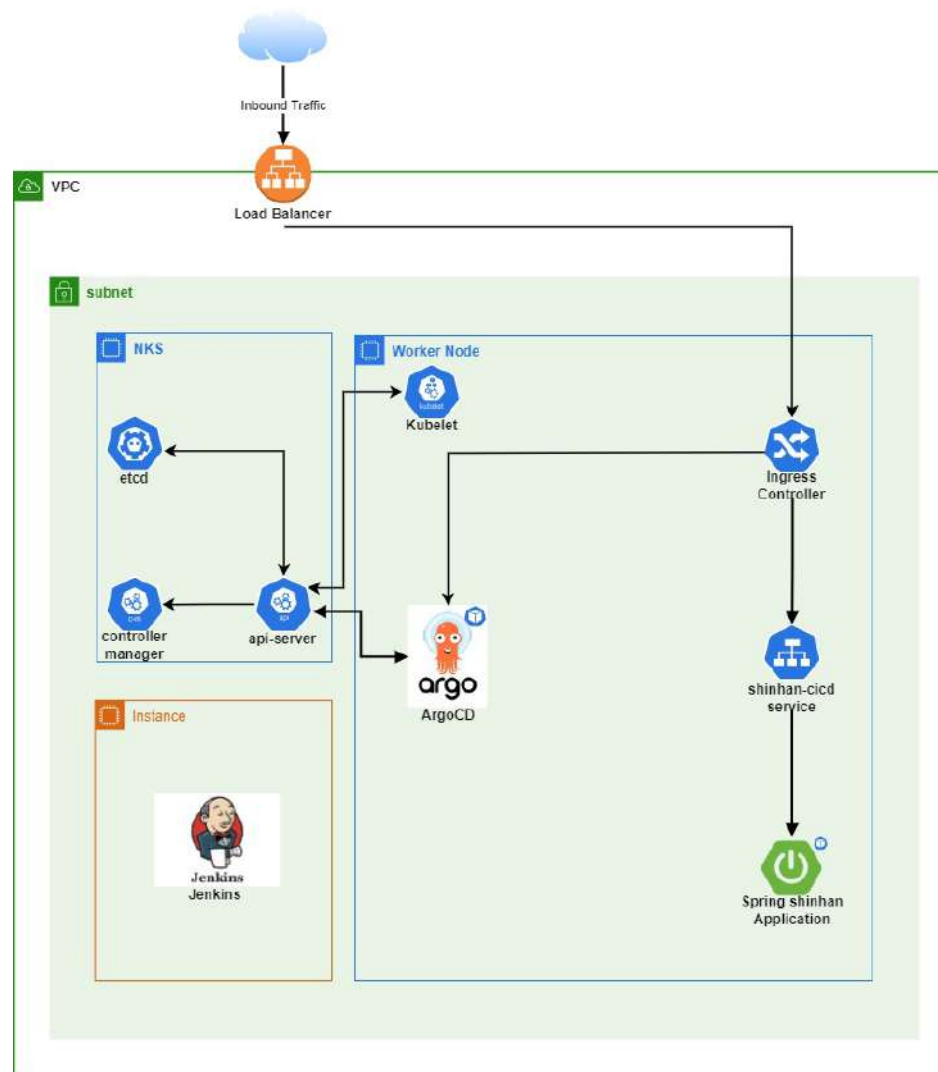
금융업 적합 CI/CD 파이프라인 구축

2023.11.17

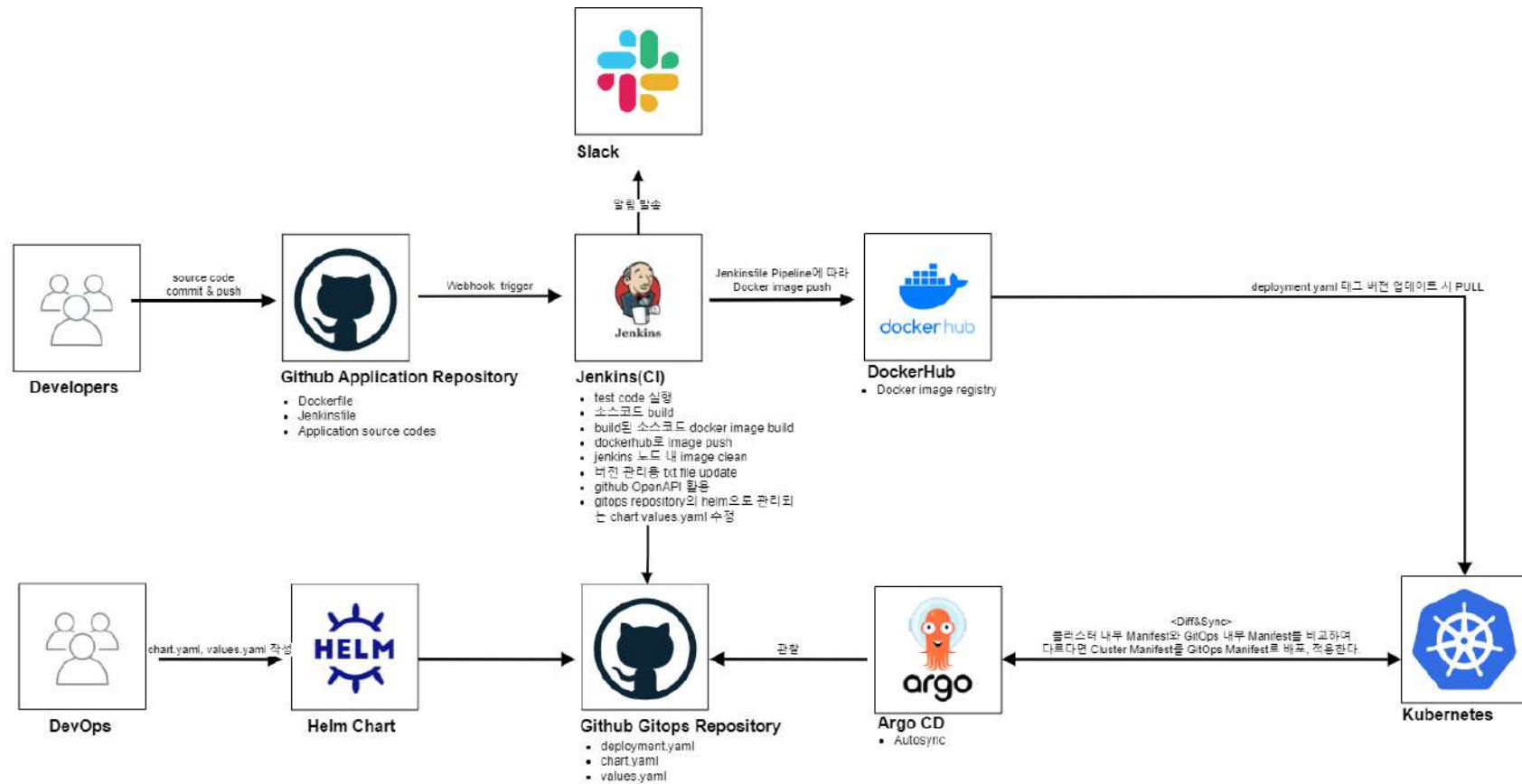
인턴사원 이수빈

- I. 설계 및 사용 툴
- II. Application Repository Push
- III. Jenkins CI
- IV. ArgoCD
- V. Gitops Repository Push
- VI. Dashboard
- VII. 교훈

클라우드 아키텍처 설계안



CICD 파이프라인 설계안



Jenkins vs Others



1. 유연성과 맞춤화

Jenkins는 Groovy 및 Jenkinsfile을 통해 무한한 맞춤화와 유연한 파이프라인 구성을 제공합니다.

이는 GitLab CI, CircleCI와 같은 다른 CI 도구들이 제공하는 것보다 훨씬 더 복잡하고 세밀한 워크플로우를 가능하게 합니다.

2. 플러그인 생태계

Jenkins의 방대한 플러그인 생태계는 다양한 개발, 테스트, 배포 도구들과의 통합을 지원합니다.

이는 Travis CI나 Bamboo와 같은 도구들보다 훨씬 넓은 범위의 확장성을 제공합니다.

ArgoCD vs Others



1. GitOps 중심

ArgoCD는 GitOps 철학에 따라 Git의 장점을 모두 가져올 수 있습니다.

- 선언적 인프라
- 롤백과 복구
- 코드의 추적 용이
- 협업 용이

2. 시각화

Jenkins나 Bamboo같은 전통적인 CI/CD 도구들이 제공하지 않는 기능입니다.

Datadog vs Others



DATADOG

1. Prometheus와 Grafana 기능을 통합 제공

Prometheus의 메트릭 수집 기능과 쿼리 언어를 사용 가능 -> Prometheus 대체

Grafana보다 더 많은 통합 서비스와 로깅을 제공

I. 설계 및 사용 툴

II. Application Repository Push

III. Jenkins CI

IV. ArgoCD












V. Gitops Repository Push


VI. Dashboard

VII. 교훈

2. Github Application Repository Push

Github Application Repository 구성

 bin-pro	Update Jenkinsfile	7a9eb4f 19 minutes ago	26 commits
	gradle/wrapper	Init: empty spring project	3 weeks ago
	src	Update TestController.java	18 hours ago
	.gitignore	Init: empty spring project	3 weeks ago
	Dockerfile	Update Dockerfile	3 days ago
	Jenkinsfile	Update Jenkinsfile	19 minutes ago
	README.md	Update README.md	3 days ago
	build.gradle	Init: empty spring project	3 weeks ago
	gradlew	Init: empty spring project	3 weeks ago
	gradlew.bat	Init: empty spring project	3 weeks ago
	settings.gradle	Init: empty spring project	3 weeks ago

README.md	
shinhan ICT 기획운영부 개인과제	
<ul style="list-style-type: none"> NKS를 활용한 CI/CD 파이프라인 구축 	



2. Github Application Repository Push

Github Application Repository 구성

Commit

Update TestController.java

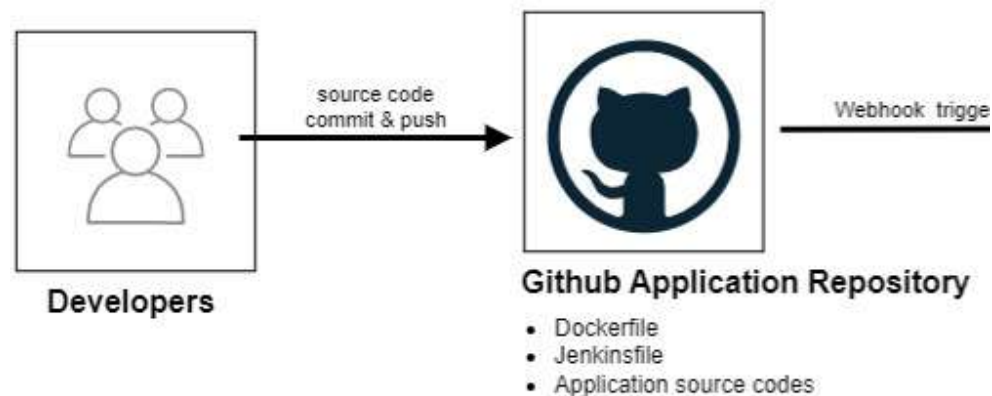
🔗 deployment

bin-pro committed now Verified

Showing 1 changed file with 1 addition and 1 deletion.

```

src/main/java/blackshoe/estheteuserservice/controller/TestController.java
@@ -9,6 +9,6 @@
 9      9      public class TestController {
10     10         @GetMapping("/cicd")
11     11         String test() {
12     -      return "Welcome to shinhan securities!";
13     +      return "신한투자증권 ICT 기획운영부 개인과제 제출 시연 - 인턴사원 이수빈";
14     14     }
  
```



- I. 설계 및 사용 툴
- II. Application Repository Push
- III. Jenkins CI**
- IV. ArgoCD
- V. Gitops Repository Push
- VI. Dashboard
- VII. 교훈**

```

graph LR
    Repository[Repository] -- "Webhook trigger" --> Jenkins[Jenkins]
    Jenkins -- "알림 발송" --> Slack[Slack]
    Jenkins -- "Jenkinsfile Pipeline에 따라 Docker image push" --> DockerHub[DockerHub]
    DockerHub --- Registry["Docker image registry"]
    ArgoCD[Argo CD] -- "관찰" --> GitopsRepo[Github Gitops Repository]
    GitopsRepo --> Repository
  
```

Repository

Jenkins

- test code 실행
- 소스코드 build
- build된 소스코드 docker image build
- dockerhub로 image push
- jenkins 노드 내 image clean
- 버전 관리용 txt file update
- github OpenAPI 활용
- gitops repository의 helm으로 관리되는 chart values.yaml 수정

Slack

DockerHub

- Docker image registry

Argo CD

- Autosync

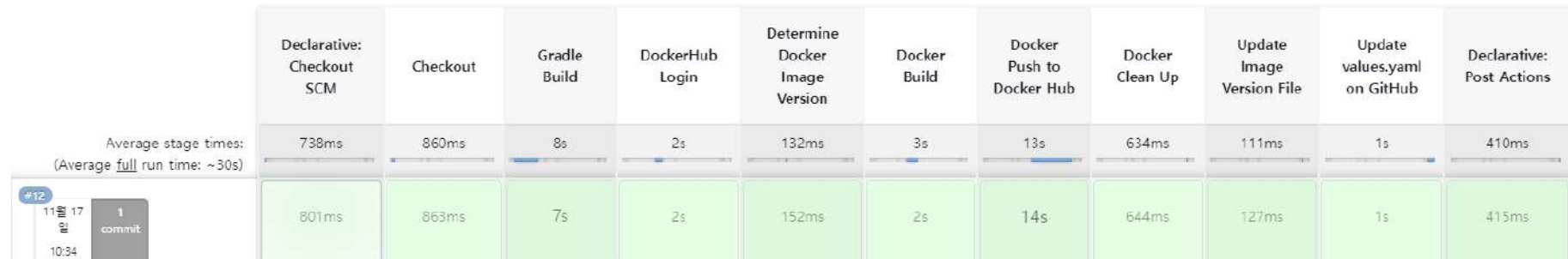
Github Gitops Repository

- deployment.yaml
- chart.yaml
- values.yaml

3. Jenkins CI

Webhook 이후 Pipeline

Stage View



1. Gradle Build
2. DockerHub Login
3. Docker Image Version
4. Docker Image Build
5. Docker Image Push to DockerHub
6. Docker Image Clean up
7. Update Image version tag
8. Update values.yaml on Github(Github API)
9. Post to slack

Chore: Update values.yaml by Jenkins

main

bin-pro committed 11 minutes ago

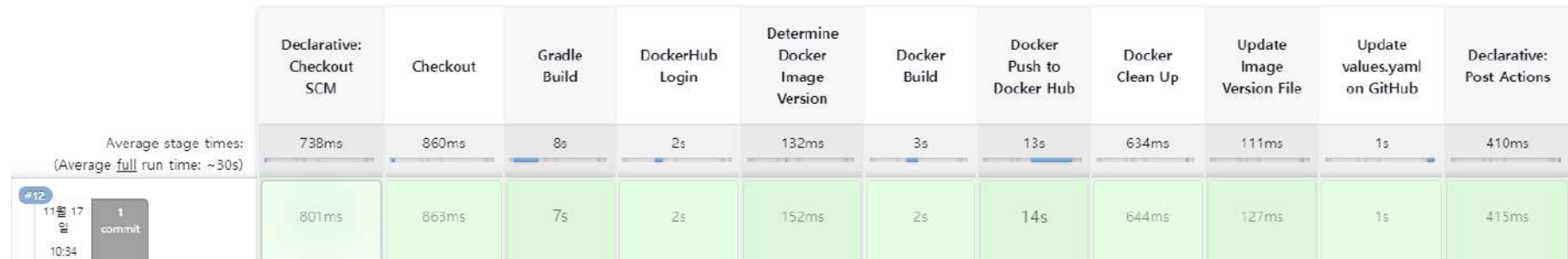
Showing 1 changed file with 1 addition and 1 deletion.

```
shinhan-charts/shinhan-chart/values.yaml
@@ -9,7 +9,7 @@ replicaCount: 1
9      9
10     10     image:
11     11         repository: lsb8375/shinhan-cicd
12     - tag: 1.8
12     + tag: 1.9
13     13
14     14     containerPort: 8080
15     15
```

3. Jenkins CI


Webhook 이후 Pipeline


Stage View





1. Gradle Build
2. DockerHub Login
3. Docker Image Version
4. Docker Image Build
5. Docker Image Push to DockerHub
6. Docker Image Clean up
7. Update Image version tag
8. Update values.yaml on Github(Github API)
9. Post to slack


 Subin Lee 오전 9:40
#test에 참여했습니다.

 Subin Lee 오전 9:43
이 채널에 통합 추가됨: jenkins

 Subin Lee 오전 9:49
이 채널에 통합 추가됨: jenkins

 jenkins 오픈 오전 9:55
Slack/Jenkins plugin: you're all set on <http://133.186.152.248:8888/>

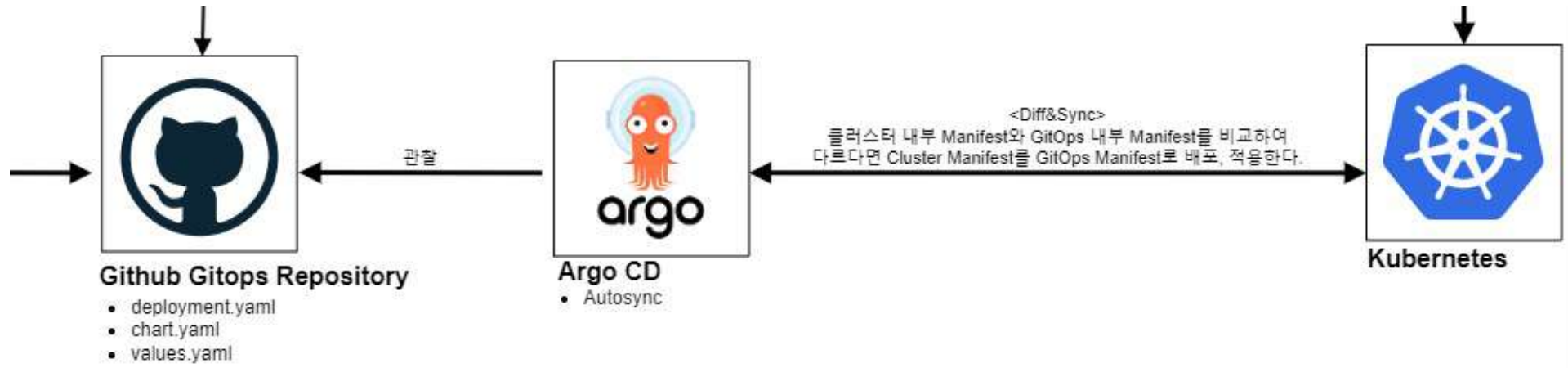
 jenkins 오픈 오전 10:13
SUCCESS: Job shinhan-cicd [1.8]

 jenkins 오픈 오전 10:34
SUCCESS: Job shinhan-cicd [1.9]

- I. 설계 및 사용 툴
- II. Application Repository Push
- III. Jenkins CI
- IV. ArgoCD**
- V. Gitops Repository Push
- VI. Dashboard
- VII. 교훈

4. ArgoCD

Webhook 이후 Pipeline



APP HEALTH ● **Healthy**

SYNC STATUS ● **Synced to HEAD (e97fdd5)**

LAST SYNC ● **Sync OK to e97fdd5**

Auto sync is enabled.
Author: SUBIN <70842920+bin-pro@users.noreply.github.com> -
Comment: Chore: Update values.yaml by Jenkins

Succeeded 39 minutes ago (Fri Nov 17 2023 10:36:31 GMT+0900)
Author: SUBIN <70842920+bin-pro@users.noreply.github.com> -
Comment: Chore: Update values.yaml by Jenkins



4. ArgoCD



CICD 결과

Commit

Update TestController.java

🔑 deployment

🌐 bin-pro committed now Verified

Showing 1 changed file with 1 addition and 1 deletion.

```
src/main/java/blackshoe/estheteuserservice/controller/TestController.java
@@ -9,6 +9,6 @@
9      9      public class TestController {
10     10         @GetMapping("/cicd")
11     11         String test() {
12     -      return "Welcome to shinhan securities!";
12     +      return "신한투자증권 ICT 기획운영부 개인과제 제출 시연 - 인턴사원 이수빈";
13     13     }
14     14     }
```

← → ↻ ⚠ 주의 요함 | 133.186.228.235/shinhan/cicd

신한투자증권 ICT 기획운영부 개인과제 제출 시연 - 인턴사원 이수빈

- I. 설계 및 사용 툴
- II. Application Repository Push
- III. Jenkins CI
- IV. ArgoCD
- V. Gitops Repository Push**
- VI. Dashboard
- VII. 교훈

5. GitOps Repository Push



Deployment.yaml 수정

Commit

Update deployment.yaml

main

bin-pro committed now Verified

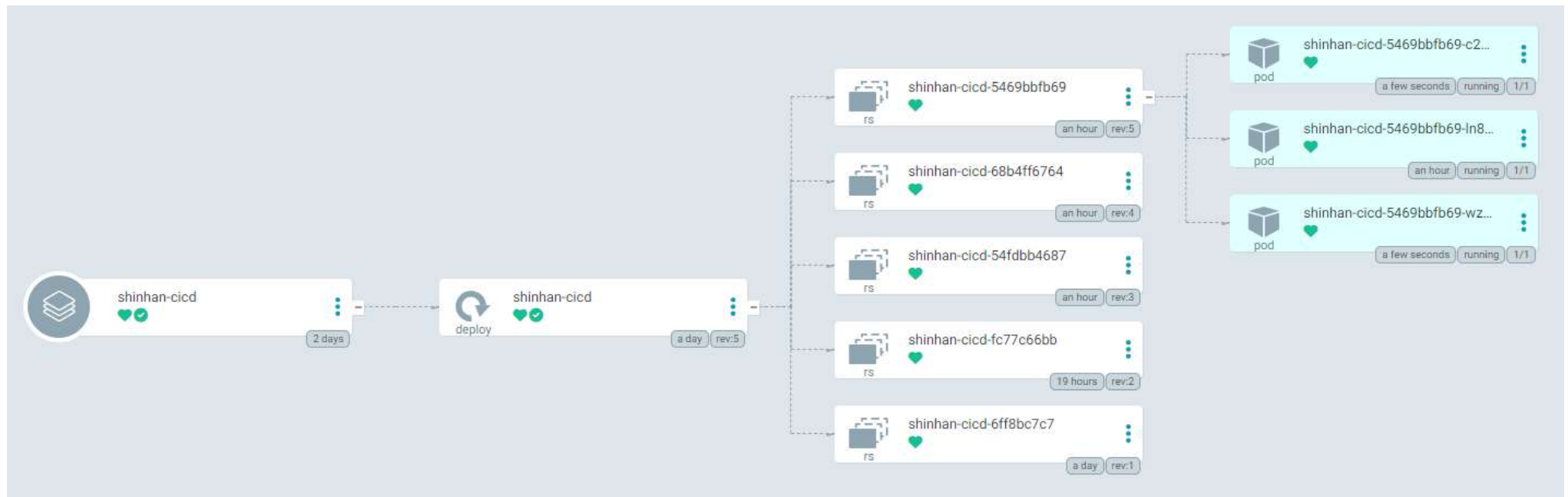
Showing 1 changed file with 1 addition and 0 deletions.

shinhan-charts/shinhan-chart/templates/deployment.yaml		
	@@ -6,6 +6,7 @@ metadata:	
6	6	name: shinhan-cicd
7	7	namespace: shinhan
8	8	spec:
9	+	replicas: 3
9	10	selector:
10	11	matchLabels:
11	12	app: shinhan-cicd

5. GitOps Repository Push



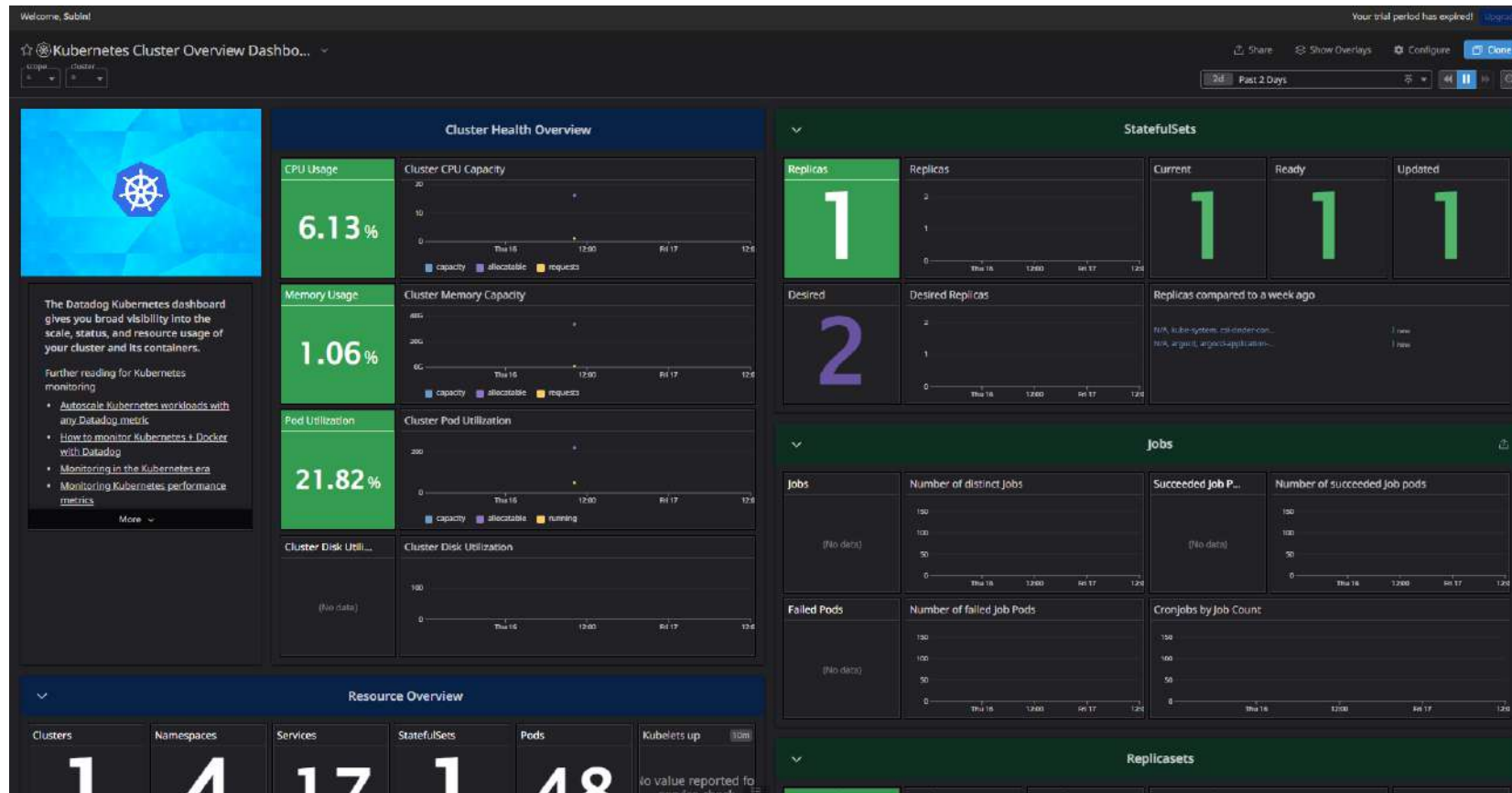
Deployment.yaml 수정



- I. 설계 및 사용 툴
- II. Application Repository Push
- III. Jenkins CI
- IV. ArgoCD
- V. GitOps Repository Push
- VI. Dashboard**
- VII. 교훈

6. Dashboard

Datadog 활용 K8s Cluster 대시보드



- I. 설계 및 사용 툴
- II. Application Repository Push
- III. Jenkins CI
- IV. ArgoCD
- V. GitOps Repository Push
- VI. Dashboard
- VII. 교훈**

1. 국내 클라우드 서비스 활용

국내 클라우드 업체인 NHN Cloud를 활용하여 NKS 클러스터를 직접 구축해보는 유의미한 시간이었습니다. NKS를 활용하며 국내 클라우드 업체들이 빠른 속도로 발전하고 있음을 체감했습니다.

2. 본격적인 CI/CD 파이프라인 구축

Helm Chart, ArgoCD, Jenkins, K8s, Datadog을 활용하고 공부해볼 수 있었고 아직 부족한 점이 많다는 것을 깨닫는 유의미한 시간이었습니다.

감사합니다