

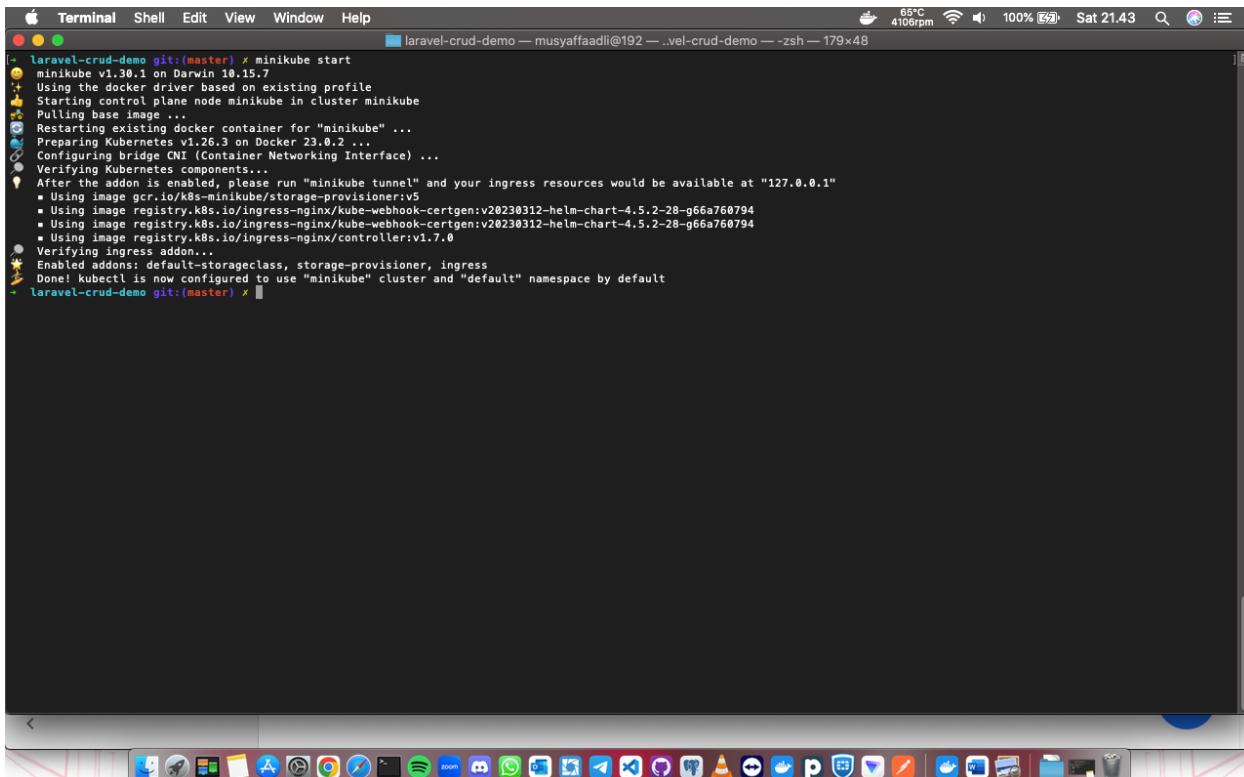
DOKUMENTASI PENGERJAAN ASSESSMENT TECHNICAL TEST

SooltanPay by Telkom Indonesia

Musyaffa Adli Syauqi

Local Kubernetes Cluster (Minikube)

1. Lakukan instalasi MiniKube pada host os berdasarkan referensi pada minikube.sigs.k8s.io.
2. Setelah instalasi pada host os dilakukan, kita dapat melakukan startup cluster dengan command "minikube start".



```
[+] laravel-crud-demo git:(master) ✘ minikube start
minikube v1.30.1 on Darwin 10.15.7
Using the docker driver based on existing profile
Starting control plane node minikube in cluster minikube
Pulling base image ...
Restarting existing docker container for "minikube" ...
Preparing Kubernetes v1.26.3 on Docker 23.0.2 ...
Configuring bridge CNI (Container Networking Interface) ...
Verifying Kubernetes components...
After the addon is enabled, please run "minikube tunnel" and your ingress resources would be available at "127.0.0.1"
  • Using image gcr.io/k8s-minikube/storage-provisioner:v5
  • Using image registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20230312-helm-chart-4.5.2-28-g66a760794
  • Using image registry.k8s.io/ingress-nginx/kube-webhook-certgen:v20230312-helm-chart-4.5.2-28-g66a760794
  • Using image registry.k8s.io/ingress-nginx/controller:v1.7.0
Verifying ingress addons...
Enabled addons: default-storageclass, storage-provisioner, ingress
Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
laravel-crud-demo git:(master) ✘
```

3. Dapat dilihat bahwa node group sudah berhasil dibuat yang running diatas CRI docker container.
4. Karena kita membutuhkan service tambahan yaitu ingress maka kita perlu melakukan enable addon pada MiniKube menggunakan command "minikube addons enable ingress".
5. Lalu kita nyalakan addons tersebut dengan command "minikube tunnel".



```
[+] laravel-crud-demo git:(master) ✘ minikube tunnel
Tunnel successfully started

NOTE: Please do not close this terminal as this process must stay alive for the tunnel to be accessible ...

Starting tunnel for service keel.
! The service/ingress crud-ingress requires privileged ports to be exposed: [80 443]
⚠ sudo permission will be asked for it.
Starting tunnel for service crud-ingress.
Password:
```

6. Instalasi cluster Kubernetes sudah selesai, kita dapat mengecek system yang ada menggunakan command “kubectl get all -A” atau command kubectl lainnya.

```

Terminal Shell Edit View Window Help
manifest — musyaffaadi@192 — zsh — 179x48
minikube ..demo/manifest ..demo/manifest
+-----+
default  pod/mysql-55dd4c6c84-rpvkz   1/1   Running  4 (4h45m ago)  29h
ingress-nginx pod/ingress-nginx-admission-create-h92jt  0/1   Completed  0      29h
ingress-nginx pod/ingress-nginx-admission-patch-mpxx  0/1   Completed  0      29h
ingress-nginx pod/ingress-nginx-controller-6cc5ccb977-86fq5 1/1   Running  4 (4h45m ago)  29h
keel    pod/keel-6b75747444-54n8w  2/2   Running  4      10m
kube-system pod/coredns-787d4945fb-56v2c  1/1   Running  4 (4h45m ago)  29h
kube-system pod/kube-apiserver-minikube  1/1   Running  4 (4h45m ago)  29h
kube-system pod/kube-controller-manager-minikube 1/1   Running  4 (4h45m ago)  29h
kube-system pod/kube-proxy-grptd  1/1   Running  4 (4h45m ago)  29h
kube-system pod/kube-scheduler-minikube 1/1   Running  4 (4h45m ago)  29h
kube-system pod/storage-provisioner 1/1   Running  9 (3h11m ago)  29h

NAMESPACE   NAME          TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)           AGE
default     service/crud-laravel  ClusterIP  10.98.62.113  <none>           80/TCP            7h52m
default     service/kubernetes  ClusterIP  10.96.0.1       <none>           443/TCP           29h
default     service/mysql      ClusterIP  10.103.255.97  <none>           3306/TCP          29h
ingress-nginx service/ingress-nginx-controller  NodePort  10.98.178.93   <none>           80:31356/TCP,443:31326/TCP  29h
ingress-nginx service/ingress-nginx-controller-admission  ClusterIP  10.104.136.15  <none>           443/TCP           29h
keel      service/keel       LoadBalancer  10.96.70.248  127.0.0.1       9300:31909/TCP   6h12m
kube-system service/kube-dns  ClusterIP  10.96.0.10     <none>           53/UDP,53/TCP,9153/TCP  29h

NAMESPACE   NAME          DESIRED  CURRENT  READY  UP-TO-DATE  AVAILABLE  NODE SELECTOR           AGE
kube-system daemonset.apps/kube-proxy  1       1       1       1       1       kubernetes.io/os=linux  29h

NAMESPACE   NAME          READY  UP-TO-DATE  AVAILABLE  AGE
default     deployment.apps/crud-laravel  1/1   1       1       7h52m
default     deployment.apps/mysql      1/1   1       1       29h
ingress-nginx deployment.apps/ingress-nginx-controller 1/1   1       1       29h
keel      deployment.apps/keel       1/1   1       1       6h12m
kube-system deployment.apps/coredns  1/1   1       1       29h

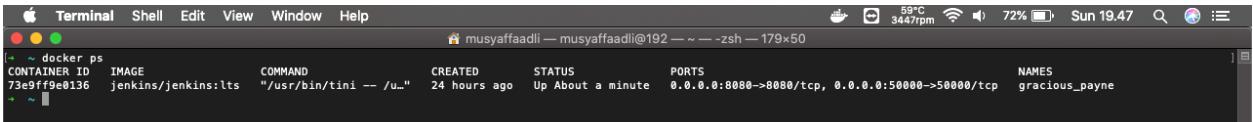
NAMESPACE   NAME          DESIRED  CURRENT  READY  AGE
default     replicaset.apps/crud-laravel-578d44fd6c  0       0       0       7h52m
default     replicaset.apps/crud-laravel-5d7f76f4859  1       1       1       3m34s
default     replicaset.apps/crud-laravel-649c5b5f9  0       0       0       38m
default     replicaset.apps/crud-laravel-669c758c98  0       0       0       96m
default     replicaset.apps/crud-laravel-7845b4d7db  0       0       0       7h20m
default     replicaset.apps/mysql-55dd4c6c84  1       1       1       29h
ingress-nginx replicaset.apps/ingress-nginx-controller-6cc5ccb977 1       1       1       29h
keel      replicaset.apps/keel-5cddb67cd7  0       0       0       6h12m
keel      replicaset.apps/keel-6b75747444  1       1       1       10m
kube-system replicaset.apps/coredns-787d4945fb  1       1       1       29h

NAMESPACE   NAME          COMPLETIONS  DURATION  AGE
ingress-nginx job.batch/ingress-nginx-admission-create 1/1   12s      29h
ingress-nginx job.batch/ingress-nginx-admission-patch  1/1   12s      29h
+ manifest git:(master) ✘

```

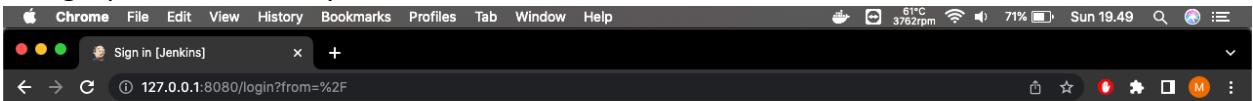
CI (Continuous Integration)

1. Pada tahap ini kita akan melakukan instalasi tools CI yaitu Jenkins.
2. Instalasi Jenkins dilakukan diatas CRI Docker Container, kita dapat melakukan instalasi tersebut dengan menjalankan command “`docker run -p 8080:8080 -p 50000:50000 -d -v /var/run/docker.sock:/var/run/docker.sock -v jenkins_home:/var/jenkins_home jenkins/jenkins:lts --network minikube`”.



```
[+ ~ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
73e9ff9e0136 jenkins/jenkins:lts "/usr/bin/tini -- /u..." 24 hours ago Up About a minute 0.0.0.0:8080->8080/tcp, 0.0.0.0:50000->50000/tcp gracious_payne]
```

3. Akses dashboard Jenkins dengan mengetikan localhost:8080 karena Jenkins menggunakan port 8080 sebagai port dashboardnya.



Welcome to Jenkins!



Username
Password
<input type="checkbox"/> Keep me signed in
Sign in



4. Di tahap ini jika Jenkins pertama kali dilakukan maka akan diminta memasukan unlock password yang dapat kita fetch dengan command “`docker exec -it <container id> bash`”, lalu setelah masuk ke dalam container gunakan command “`cat var/jenkins_home/secrets/initialAdminPassword`”.

5. Karena di tahap ini saya udah melakukan initial set up pada Jenkins maka setelah akses ke dashboard akan diarahkan untuk login dan masuk ke halaman utama.

6. Lalu lakukan set up pipeline baru jika belum, di sini saya sudah melakukan inisialisasi pipeline yang mengarah ke repo source code di github saya. Pada tahap ini Jenkins sudah berhasil diinstall

CD (Continuous Delivery/Deployment)

1. Pada tahap ini kita akan melakukan instalasi tools CD yaitu Keel.
2. Keel sendiri adalah salah satu tools CD open source yang digunakan untuk melakukan deploy pada cluster Kubernetes menggunakan metode gitops atau gitflow sama halnya seperti ArgoCD atau FluxCD.
3. Buat akun pada situs webhookrelay.com untuk mendapatkan free tunnelling relay gratis yang akan kita gunakan pada keel.

The screenshot shows the 'Webhook Relay' application interface. On the left is a sidebar with 'Configuration Management' sections: Dashboard, Request Forward..., Bidirectional Tunnels (selected), and Domain Reservations. The main content area is titled 'Tunnels' and shows a table with one row:

Name	Host	Region	Status	Destination	Crypto	Settings	Auth	Create
keel	http://s6yfob249fhxdgxhr7jzu.webrelay.io	eu	●	http://localhost:9300	OFF			Ma 13t 202 3:14:4 pr

At the bottom right of the main area is a blue speech bubble icon. The browser's address bar shows 'my.webhookrelay.com/tunnels'.

4. Lakukan instalasi keel di dalam cluster mengikuti referensi pada keel.sh dan lakukan konfigurasi tunnelling ke webhookrelay dengan cara memasukan access token dan secret yang sudah kita dapat.

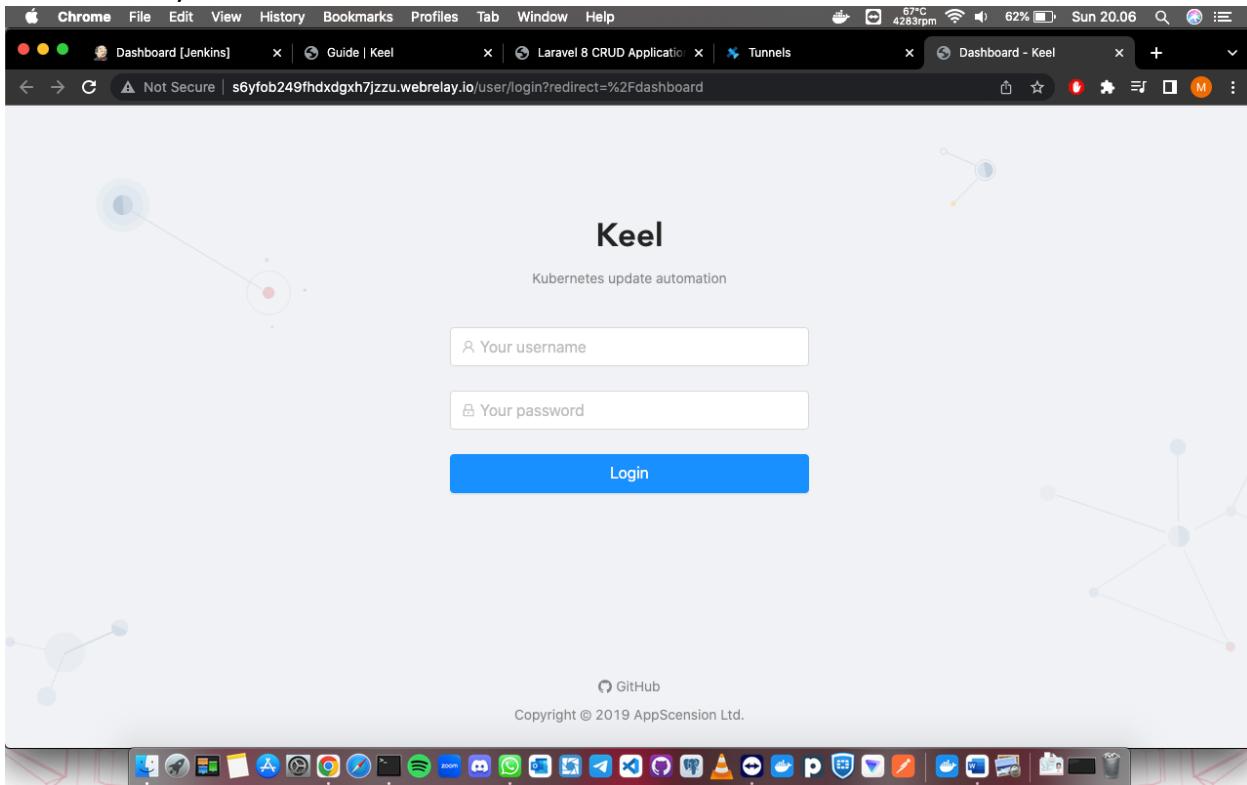
```
Last login: Sun May 14 19:46:02 on ttys000
+ sooltanpay-demo kga -n keel
NAME          READY   STATUS    RESTARTS   AGE
pod/keel-6b75747444-84n8w  0/2     ImagePullBackOff  2 (22h ago)  22h

NAME        TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
service/keel  LoadBalancer  10.96.70.248  127.0.0.1      9300:31909/TCP  28h

NAME          READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/keel  0/1       1           0           28h

NAME          DESIRED  CURRENT   READY   AGE
replicaset.apps/keel-5cddb67cd7  0        0        0      28h
replicaset.apps/keel-6b75747444  1        1        0      22h
+ sooltanpay-demo
```

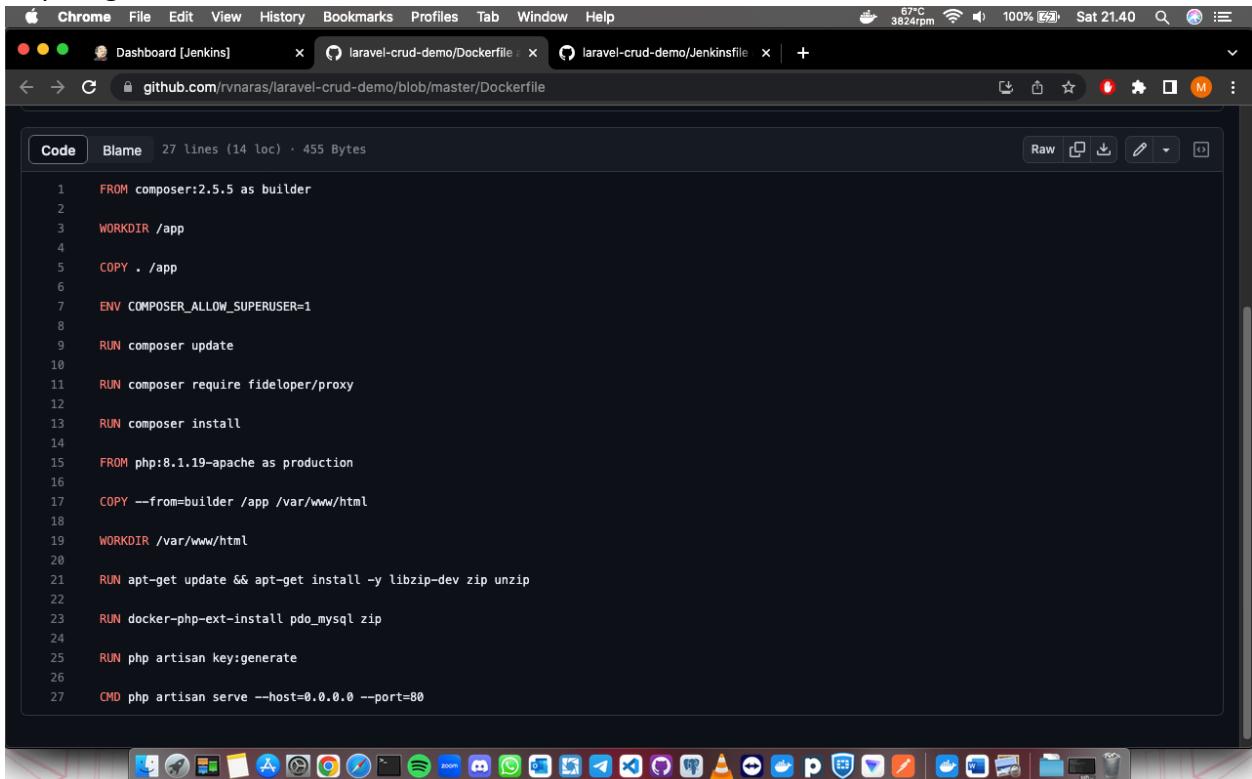
- Setelah kita melakukan instalasi keel, kita dapat mengakses dashboardnya dengan url host pada webhookrelay.



- Instalasi keel sudah berhasil kita lakukan.

Service Deployment (Basic Laravel 8 CRUD)

1. Pada Langkah ini kita akan melakukan deploy aplikasi laravel yang menjadi ketentuan pada assessment ini.
2. Lakukan fork pada repository github pada aplikasi yang sudah ditentukan yaitu <https://github.com/savanihd/Laravel-8-CRUD>.
3. Karena ada beberapa asset yang tidak ada seperti Dockerfile dan Jenkinsfile, maka kita perlu melakukan konfigurasi pada repository yang sudah kita fork.
4. Untuk Dockerfile dapat dilihat pada repository saya pada link berikut
<https://github.com/rvnaras/laravel-crud-demo/blob/master/Dockerfile>



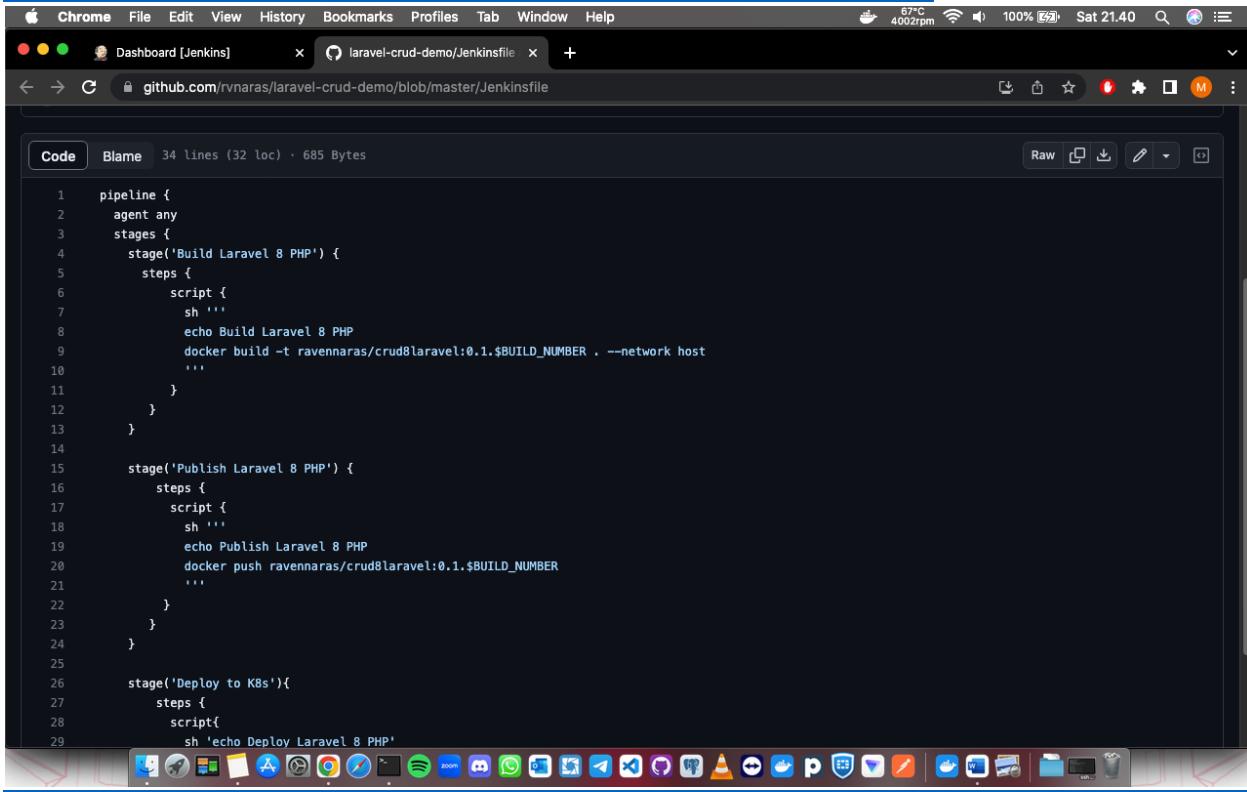
The screenshot shows a Mac OS X desktop with a Chrome browser window open. The address bar shows github.com/rvnaras/laravel-crud-demo/blob/master/Dockerfile. The browser content displays the Dockerfile code:

```
1  FROM composer:2.5.5 as builder
2
3  WORKDIR /app
4
5  COPY . /app
6
7  ENV COMPOSER_ALLOW_SUPERUSER=1
8
9  RUN composer update
10
11 RUN composer require fideloper/proxy
12
13 RUN composer install
14
15 FROM php:8.1.19-apache as production
16
17 COPY --from=builder /app /var/www/html
18
19 WORKDIR /var/www/html
20
21 RUN apt-get update && apt-get install -y libzip-dev zip unzip
22
23 RUN docker-php-ext-install pdo_mysql zip
24
25 RUN php artisan key:generate
26
27 CMD php artisan serve --host=0.0.0.0 --port=80
```

The browser has tabs for 'Dashboard [Jenkins]', 'laravel-crud-demo/Dockerfile', and 'laravel-crud-demo/Jenkinsfile'. The status bar at the top shows system information like battery level (67%), temperature (38.24°C), and date (Sat 21.40). The dock at the bottom contains various application icons.

5. Pada Dockerfile tersebut saya melakukan multistage docker build dengan tujuan untuk mengefisiensikan asset file didalam docker image production dan meminimalisir ukuran docker image.

6. Untuk Jenkinsfile dapat dilihat pada repository saya pada link berikut
<https://github.com/rvnaras/laravel-crud-demo/blob/master/Jenkinsfile>



The screenshot shows a Chrome browser window with the address bar displaying "github.com/rvnaras/laravel-crud-demo/blob/master/Jenkinsfile". The main content area shows the Jenkinsfile code:

```
1 pipeline {
2   agent any
3   stages {
4     stage('Build Laravel 8 PHP') {
5       steps {
6         script {
7           sh ''
8           echo Build Laravel 8 PHP
9           docker build -t ravennaras/crud8laravel:0.1.$BUILD_NUMBER . --network host
10          ''
11        }
12      }
13    }
14    stage('Publish Laravel 8 PHP') {
15      steps {
16        script {
17          sh ''
18          echo Publish Laravel 8 PHP
19          docker push ravennaras/crud8laravel:0.1.$BUILD_NUMBER
20          ''
21        }
22      }
23    }
24    stage('Deploy to K8s'){
25      steps {
26        script{
27          sh 'echo Deploy Laravel 8 PHP'
28        }
29      }
30    }
31  }
32 }
```

7. Pada Jenkinsfile saya membuat 2 stage utama dan 1 stage informasi, stage pertama adalah building docker image menggunakan semver tagging minor, stage kedua adalah pushing docker image yang sudah dibuild pada stage sebelumnya ke repository, dan pada stage terakhir adalah stage informasi yang prosesnya akan dilakukan oleh tools bernama keel.
8. Pada langkah ini kita akan melakukan deployment pada initial cluster Kubernetes yang masih kosong, manifest deployment dapat dilihat di repository saya pada link berikut
<https://github.com/rvnaras/laravel-crud-demo/tree/master/manifest>.
9. Pada repository tersebut terdapat tiga manifest yaitu manifest aplikasi (crud-laravel.yaml), manifest database (mysql.yaml) dan manifest konfigurasi ingress nginx (ingress.yaml).

10. Lakukan deployment ketiga manifest tersebut di dalam cluster.

```

Terminal Shell Edit View Window Help
minikube
Last login: Sat May 13 20:51:21 on ttys001
+ laravel-crud-demo git:(master) ✘ kubectl get
NAME          READY   STATUS    RESTARTS   AGE
pod/crud-laravel-5df76f4859-xp7c   1/1     Running   0          6m38s
pod/mysql-55dd4c6c84-rpvkz        1/1     Running   4 (4h48m ago) 29h

NAME            TYPE      CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/crud-laravel   ClusterIP  10.98.62.113 <none>       80/TCP    7h55m
service/kubernetes   ClusterIP  10.98.0.1    <none>       443/TCP   29h
service/mysql        ClusterIP  10.103.255.97 <none>       3306/TCP  29h

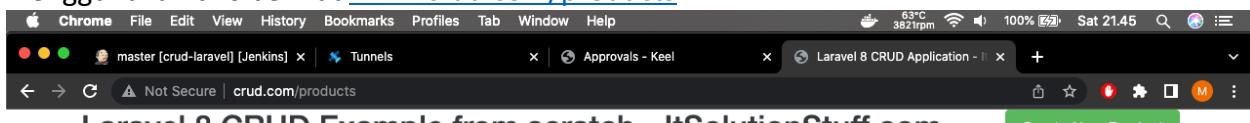
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/crud-laravel  1/1     1           1           7h55m
deployment.apps/mysql        1/1     1           1           29h

NAME           DESIRED  CURRENT  READY   AGE
replicaset.apps/crud-laravel-578d44fd6c  0        0        0       7h55m
replicaset.apps/crud-laravel-5df76f4859  1        1        1       6m38s
replicaset.apps/crud-laravel-649c5b5f9  0        0        0       41m
replicaset.apps/crud-laravel-669f758c98  0        0        0       99m
replicaset.apps/crud-laravel-7845bd79db  0        0        0       7h23m
replicaset.apps/mysql-55dd4c6c84        1        1        1       29h
+ laravel-crud-demo git:(master) ✘

```

11. Pastikan semua service running dengan normal.

12. Layanan database menggunakan service type ClusterIP karena aplikasi dapat melakukan query data sebelum mengirimkan data pada client, selain itu juga untuk meminimalisir security issue dimana jika database di expose ke public maka akan menjadi salah satu jalur masuknya peretas.
13. Karena kita sudah melakukan konfigurasi pada ingress, maka kita dapat melakukan akses pada aplikasi menggunakan dns berikut www.crud.com/products



Laravel 8 CRUD Example from scratch - ItSolutionStuff.com

No	Name	Details	Action
1	Applewatch SE 2022	garansi DigiMap, semuanya ilang, sisa kardusnya aja	Show Edit Delete
2	Iphone 14 Pro	garansi IBOX, sisa kabelnya aja, hpnya ilang	Show Edit Delete
3	MacBook Pro M2 2023	garansi IBOX, kelengkapan sisa laptopnya aja, sisanya ilang	Show Edit Delete

14. Di tahap ini kita sudah berhasil melakukan deployment aplikasi pada cluster.

15. Langkah selanjutnya adalah kita akan mencoba pipeline Jenkins CI/CD yang sudah kita provision.

16. Lakukan build pada pipeline Jenkins untuk melakukan build, push dan deploy pada cluster.

The screenshot shows the Jenkins interface for the 'crud-laravel' project's 'master' branch. On the left, there's a sidebar with links like 'Status', 'Changes', 'Build Now', 'View Configuration', 'Full Stage View', 'GitHub', 'Pipeline Syntax', and 'Build History'. The 'Build History' section shows three builds: #8 (May 13, 2023, 2:44 PM), #7 (May 13, 2023, 21:30), and #6 (May 13, 2023, 2:30 PM). The main area is titled 'Branch master' and shows a 'Stage View' grid. The grid has four columns: 'Declarative: Checkout SCM', 'Build Laravel 8 PHP', 'Publish Laravel 8 PHP', and 'Deploy to K8s'. Each column contains a horizontal bar indicating the average stage time. Below the grid, specific times are listed: 3s, 49s, 11s, 377ms for the first row; 2s, 5s, 9s, 573ms for the second; and 2s, 1min 27s, 24s, 626ms for the third. The Jenkins logo is at the top right, and the system tray shows various application icons.

17. Pastikan build job sudah berhasil dieksekusi.

The screenshot shows the Jenkins console output for build #8. The output is as follows:

```
[Pipeline] 
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Deploy to K8s)
[Pipeline] script
[Pipeline] {
[Pipeline] sh
+ echo Deploy Laravel 8 PHP
Deploy Laravel 8 PHP
[Pipeline] 
[Pipeline] // script
[Pipeline] 
[Pipeline] // stage
[Pipeline] 
[Pipeline] // withEnv
[Pipeline] 
[Pipeline] // node
[Pipeline] End of Pipeline
```

GitHub has been notified of this commit's build result

Finished: SUCCESS

In the bottom right corner, it says 'REST API Jenkins 2.387.3'. The system tray at the bottom of the screen shows various application icons.

18. Untuk memastikan apakah docker image sudah berhasil dipush, cek di repository dockerhub.
<https://hub.docker.com/repository/docker/ravennaras/crud8laravel/general>

ravennaras / crud8laravel

Description

This repository does not have a description.

Last pushed: a minute ago

Tags

This repository contains 3 tag(s).

Tag	OS	Type	Pulled	Pushed
0.1.7	🐧	Image	2 minutes ago	2 minutes ago
0.1.6	🐧	Image	37 minutes ago	37 minutes ago
0.1.5	🐧	Image	34 minutes ago	2 hours ago

[See all](#) [Go to Advanced Image Management](#)

Automated Builds

Manually pushing images to Hub? Connect your account to GitHub or Bitbucket to automatically build and tag new images whenever your code is updated, so you can focus your time on creating.

Available with Pro, Team and Business subscriptions. [Read more about automated builds](#).

[Upgrade](#)

19. Karena tahap continuous deployment dilakukan oleh tools bernama keel, maka kita dapat melakukan pengecekan apakah image terbaru sudah terdeteksi.

Keel

Pending: 1 | Approved: 0 | Rejected: 0

Approvals

Last Activity	Provider	Identifier	Votes	Delta	Status	Expires In	Action
May 13, 2023 at 09:44:36 PM	kubernetes	deployment/default/crud-laravel:0.1.8	0/1	0.1.7 -> 0.1.8	Collecting...	23:59:49	

< 1 >

Copyright © 2019 Keel

20. Karena image terbaru sudah dideteksi oleh keel, tugas kita adalah melakukan approval untuk mengijinkan deployment pada cluster diperbarui.

The screenshot shows a browser window with the Keel application. The title bar says "Approvals - Keel". The main interface has tabs for "Dashboard", "Tracked Images", "Approvals" (which is selected), and "Audit". On the left, there are two counts: "Pending" (1) and "Rejected" (0). The central area is titled "Approvals" and lists a single item:

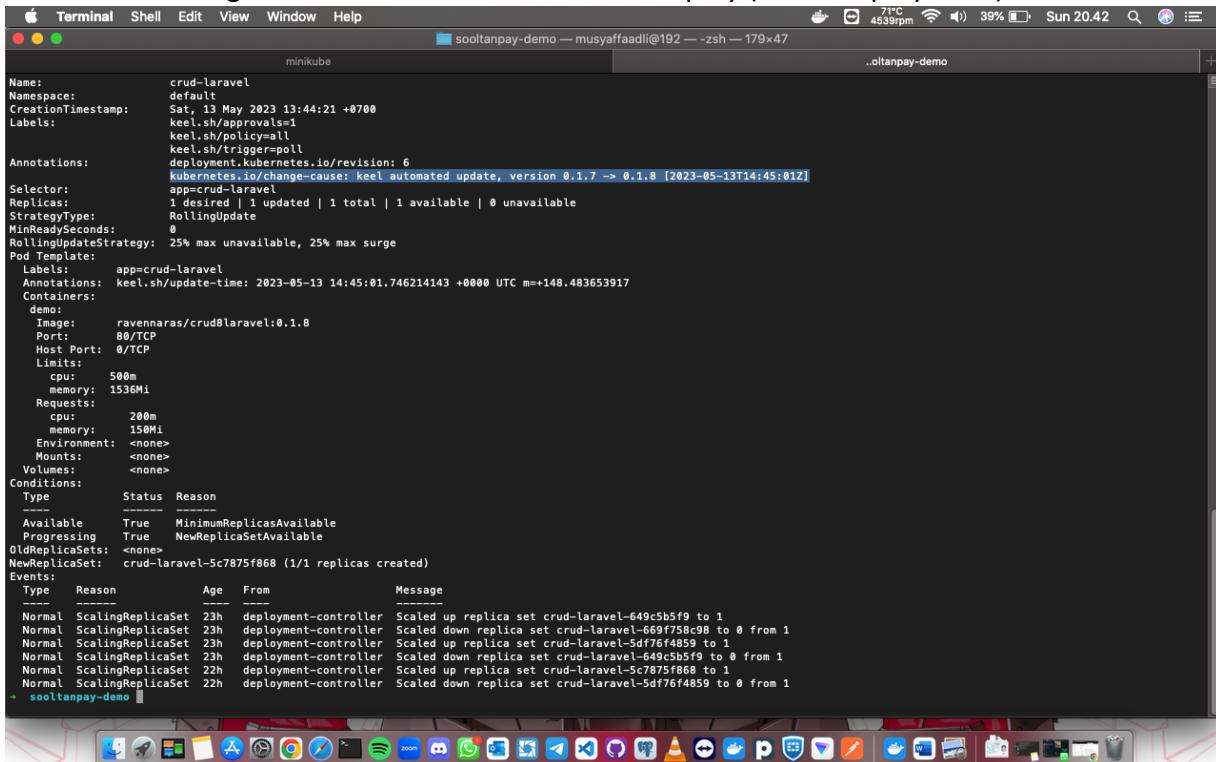
Last Activity	Provider	Identifier	Votes	Delta	Status	Expires In	Action
May 13, 2023 at 09:44:36 PM	kubernetes	deployment/default/crud-laravel:0.1.8	0/1	0.1.7 -> 0.1.8	Collecting...	23:59:42	

A modal dialog box is open in the center, titled "Confirm update", asking "are you sure want to approve update for deployment/default/crud-laravel:0.1.8?". It has "Cancel" and "OK" buttons. Below the table, there are navigation arrows and a search bar.

21. Maka image pada cluster akan diupdate mengikuti latest version pada image registry yaitu dockerhub.

The screenshot shows the same Keel interface as before, but now the "Pending" count is 0 and the "Approved" count is 1. A success message box is displayed in the top right corner: "Approved! deployment/default/crud-laravel:0.1.8 approved successfully!". The approvals table shows the same item as before, but its status is now "Archived" with a green progress bar indicating completion. The "Expires In" field shows a long duration of "-". The bottom of the screen shows the Mac OS X dock with various application icons.

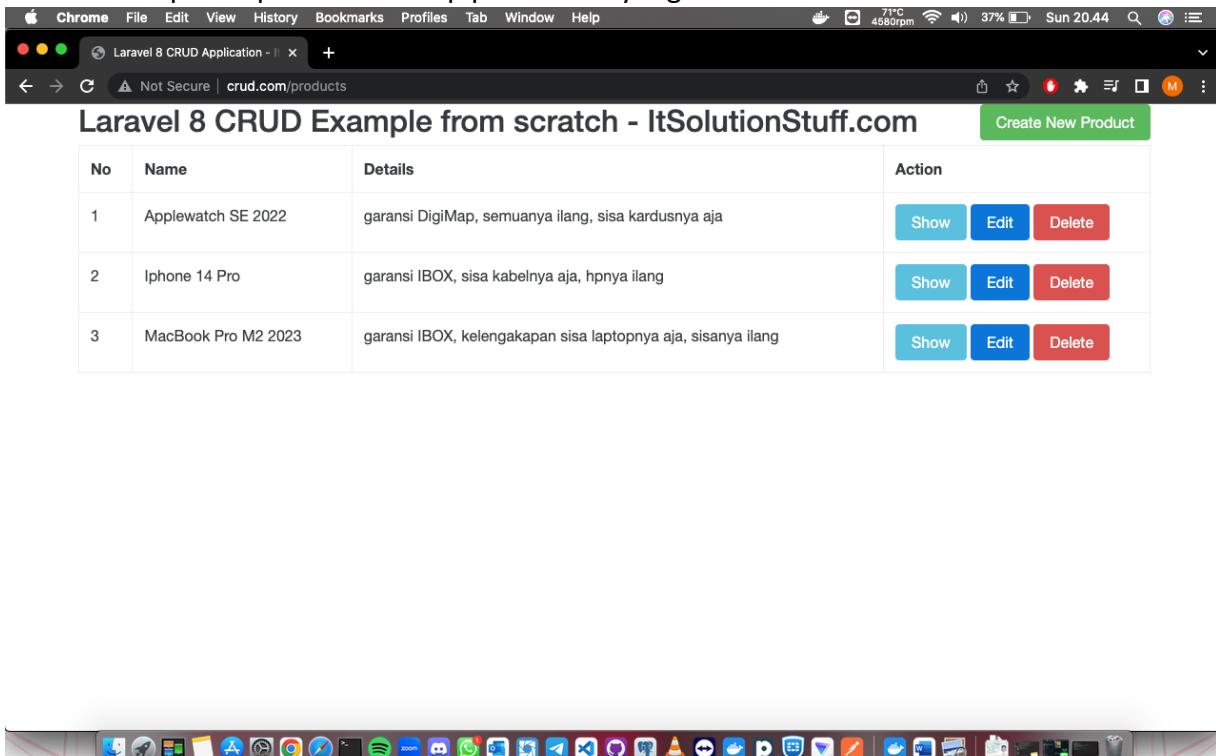
22. Lalu langkah terakhir adalah memastikan apakah image version di cluster sudah terupdate dengan versi terbaru dengan command “kubectl describe deploy (nama deployment)”.



```
Terminal Shell Edit View Window Help
minikube
sooltanpay-demo — musyaffaadi@192 — -zsh — 179x47
..sooltanpay-demo

Name: crud-laravel
Namespace: default
CreationTimestamp: Sat, 13 May 2023 13:44:21 +0700
Labels: keel.sh/approvals=1
keel.sh/policy=all
keel.sh/trigger=poll
Annotations: deployment.kubernetes.io/revision: 6
kubernetes.io/change-cause: keel automated update, version 0.1.7 -> 0.1.8 [2023-05-13T14:45:01Z]
Selector: app=crud-laravel
Replicas: 1 desired | 1 updated | 1 total | 1 available | 0 unavailable
StrategyType: RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels: app=crud-laravel
  Annotations: keel.sh/update-time: 2023-05-13 14:45:01.746214143 +0000 UTC m=+148.483653917
  Containers:
    demo:
      Image: ravennaras/crud8laravel:0.1.8
      Port: 80/TCP
      Host Port: 0/TCP
      Limits:
        cpu: 500m
        memory: 1536Mi
      Requests:
        cpu: 200m
        memory: 150Mi
      Environment: <none>
      Mounts: <none>
      Volumes: <none>
  Conditions:
    Type Status Reason
    ----
    Available True   MinimumReplicasAvailable
    Progressing True  NewReplicaSetAvailable
OldReplicaSets: <none>
NewReplicaSet: crud-laravel-5c7875f868 (1/1 replicas created)
Events:
  Type Reason Age From Message
  Normal ScalingReplicaSet 23h deployment-controller Scaled up replica set crud-laravel-649c5h5f9 to 1
  Normal ScalingReplicaSet 23h deployment-controller Scaled down replica set crud-laravel-669f758c98 to 0 from 1
  Normal ScalingReplicaSet 23h deployment-controller Scaled up replica set crud-laravel-5df76f4859 to 1
  Normal ScalingReplicaSet 23h deployment-controller Scaled down replica set crud-laravel-649c5b5f9 to 0 from 1
  Normal ScalingReplicaSet 22h deployment-controller Scaled up replica set crud-laravel-5c7875f868 to 1
  Normal ScalingReplicaSet 22h deployment-controller Scaled down replica set crud-laravel-5df76f4859 to 0 from 1
+ sooltanpay-demo
```

23. Lalu kita cek pada aplikasi terhadap perubahan yang kita lakukan.



No	Name	Details	Action
1	Applewatch SE 2022	garansi DigiMap, semuanya ilang, sisa kardusnya aja	Show Edit Delete
2	Iphone 14 Pro	garansi IBOX, sisa kabelnya aja, hpnya ilang	Show Edit Delete
3	MacBook Pro M2 2023	garansi IBOX, kelengkapan sisa laptopnya aja, sisanya ilang	Show Edit Delete

24. Langkah CI/CD dan deployment aplikasi pada cluster sudah berhasil.