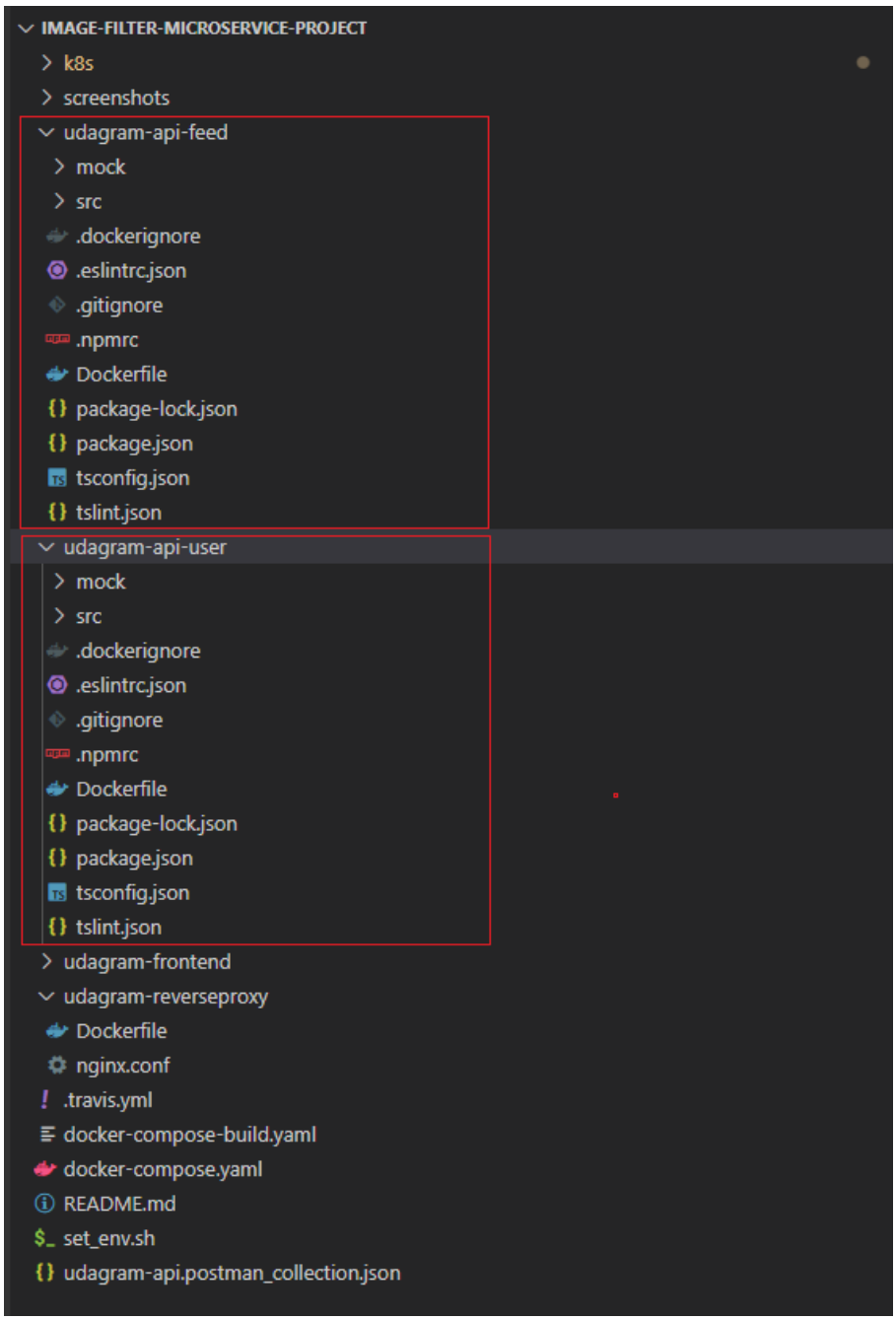
Git Repository

https://github.com/igomezgithub/image-filter-microservice-project.git

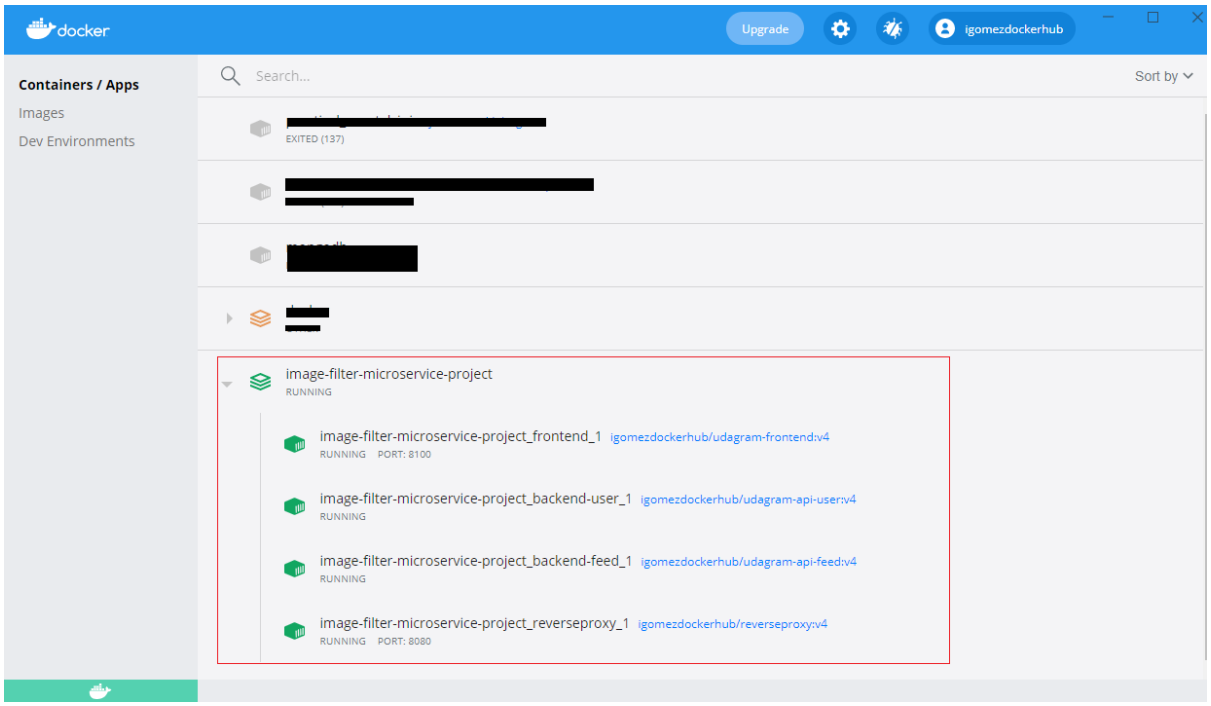
Project Rubric

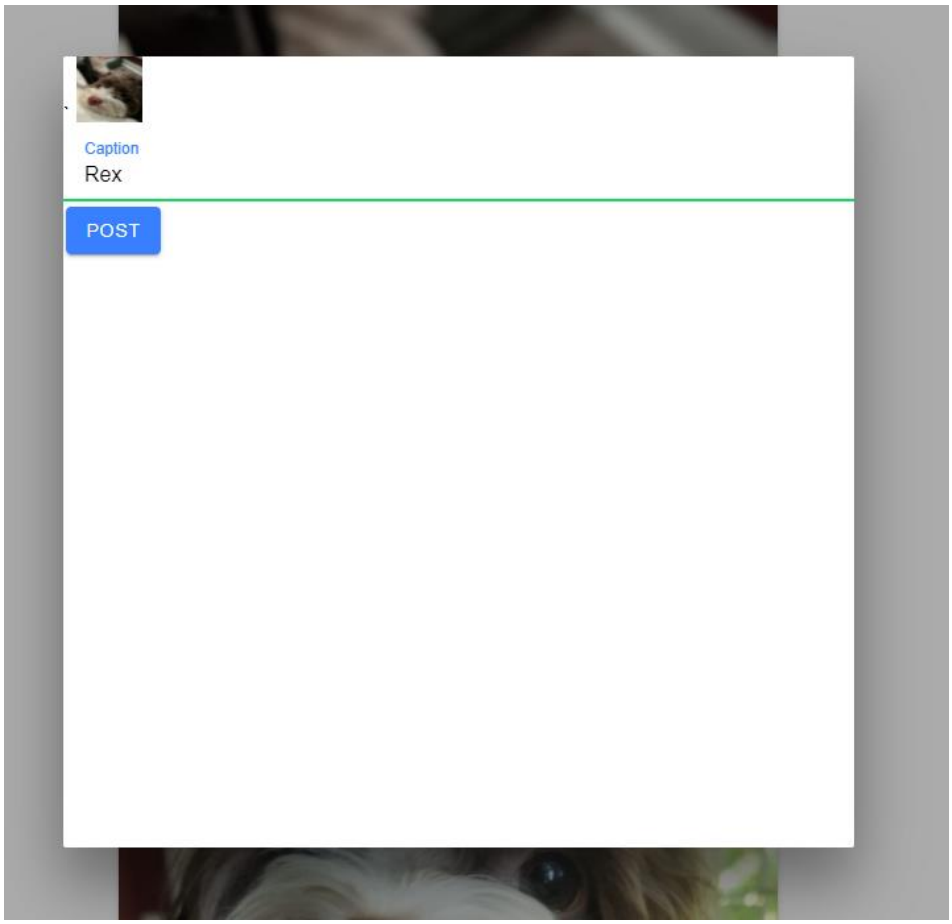
Containers and Microservices

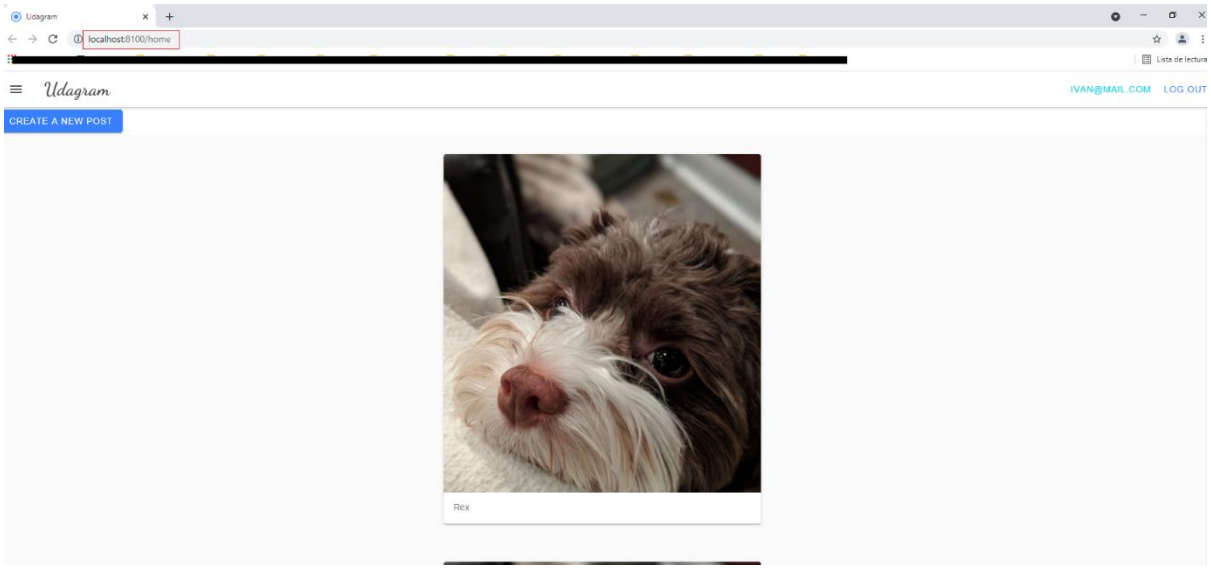
- Divide an application into microservices:



- Build and run a container image using Docker:

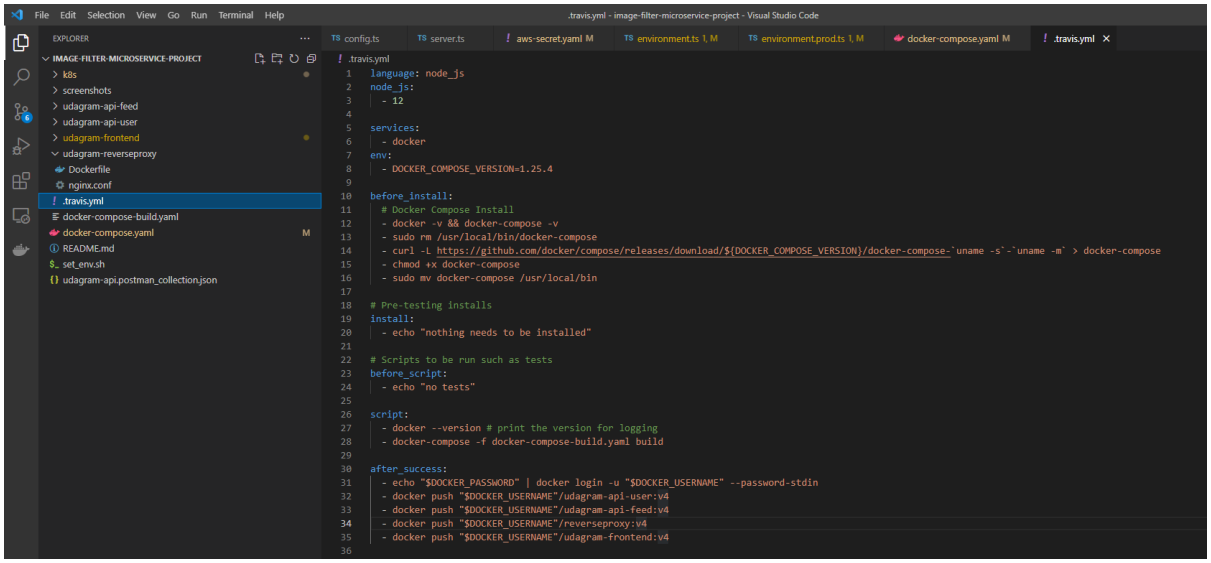


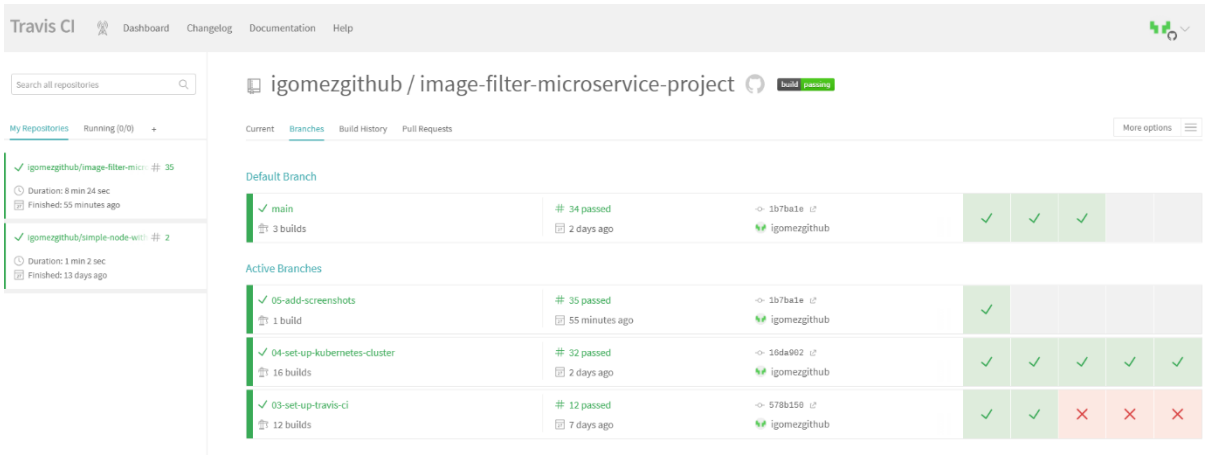


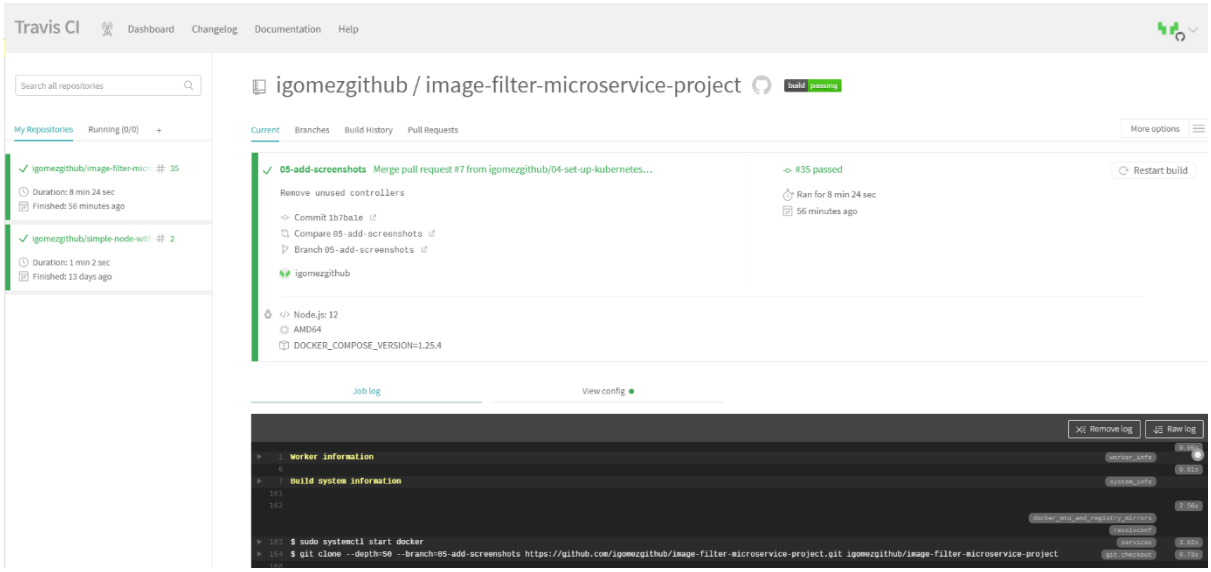


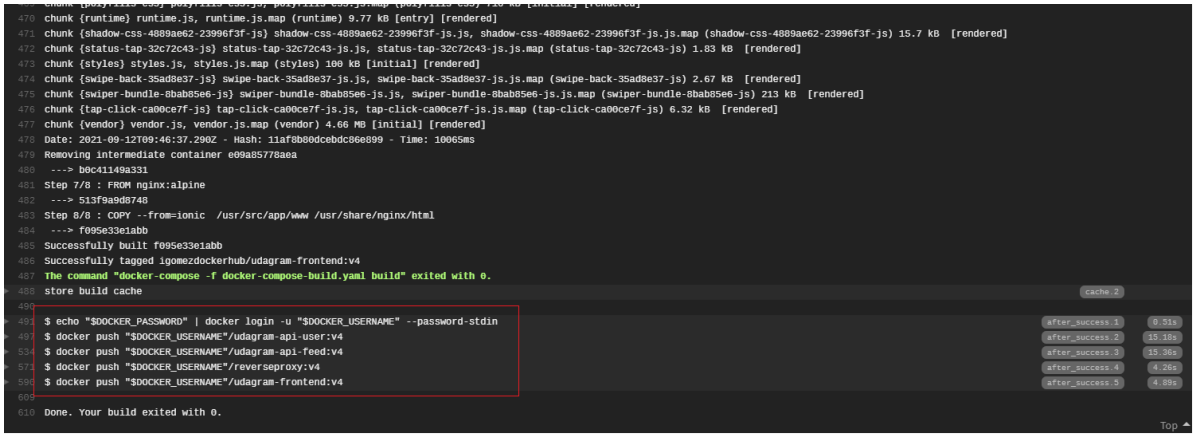
Independent Releases and Deployments

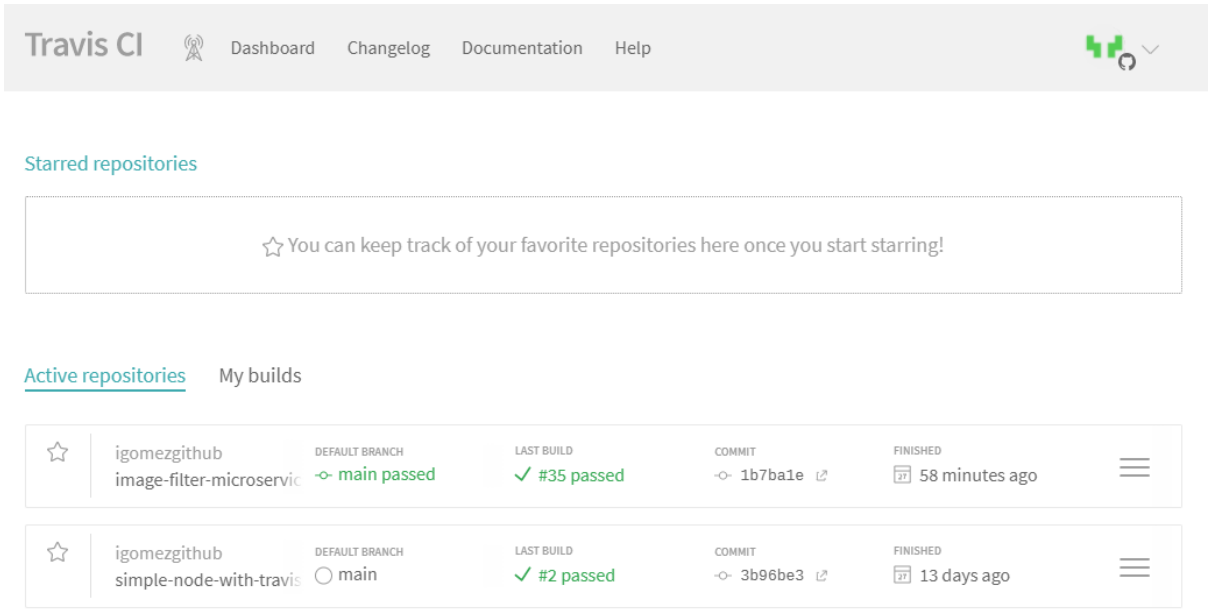
- Use Travis to build a CI/CD pipeline:

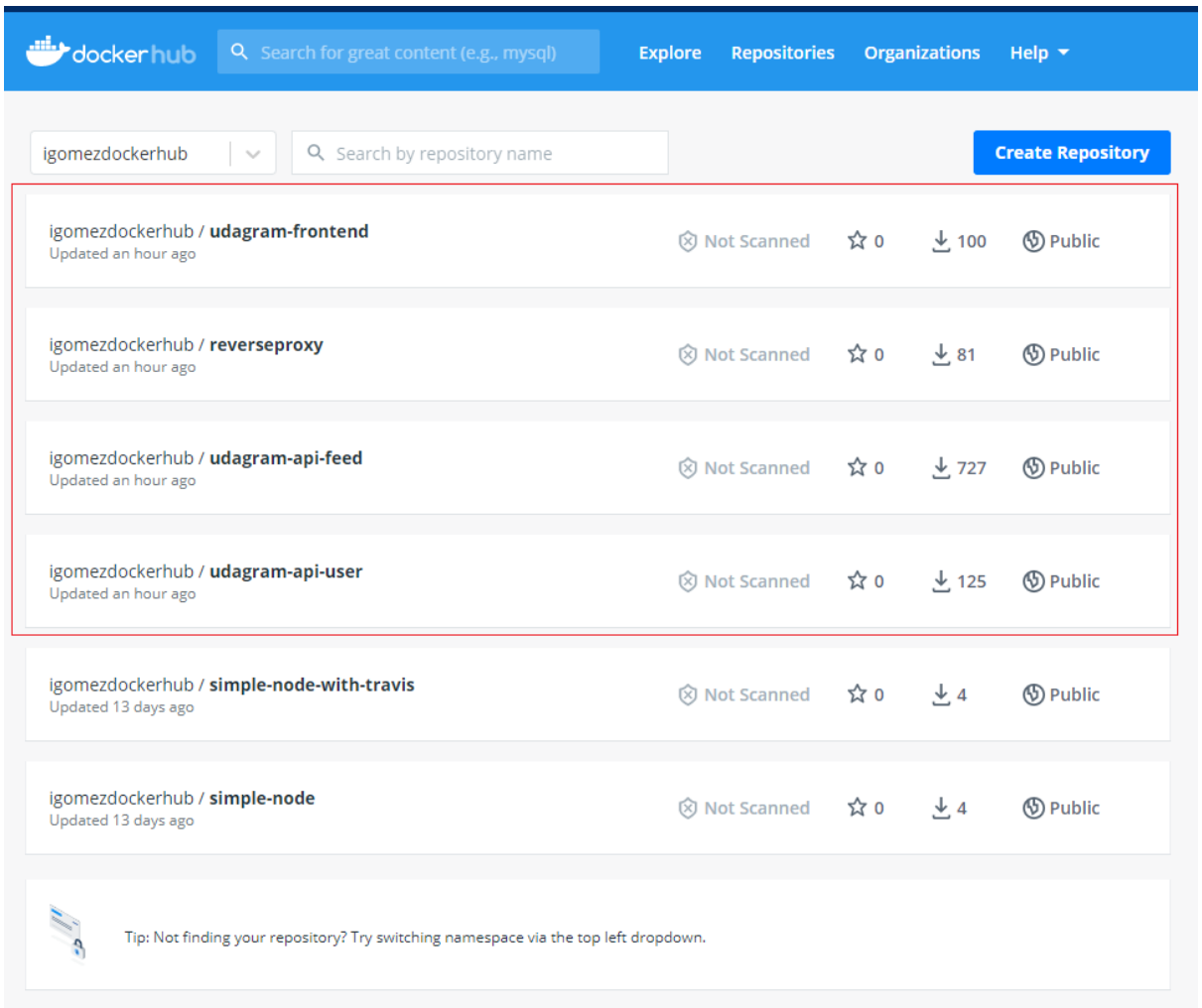



Travis CI
Dashboard
Changelog
Documentation
Help
Search all repositories
My Repositories
Running (0/0)
igomezgithub/image-filter-micr # 35
Duration: 8 min 24 sec
Finished: 55 minutes ago
igomezgithub/simple-node-with # 2
Duration: 1 min 2 sec
Finished: 13 days ago
igomezgithub / image-filter-microservice-project
build passing
Current
Branches
Build History
Pull Requests
More options
Default Branch
main
3 builds
34 passed
2 days ago
1b7ba1e
igomezgithub
Active Branches
05-add-screenshots
1 build
35 passed
55 minutes ago
1b7ba1e
igomezgithub
04-set-up-kubernetes-cluster
16 builds
32 passed
2 days ago
10da902
igomezgithub
03-set-up-travis-ci
12 builds
12 passed
7 days ago
578b150
igomezgithub


Travis CI
Dashboard
Changelog
Documentation
Help
Search all repositories
My Repositories
Running (0/0)
igomezgithub/image-filter-micr # 35
Duration: 8 min 24 sec
Finished: 56 minutes ago
igomezgithub/simple-node-with # 2
Duration: 1 min 2 sec
Finished: 13 days ago
igomezgithub / image-filter-microservice-project
build passing
Current
Branches
Build History
Pull Requests
More options
05-add-screenshots Merge pull request #7 from igomezgithub/04-set-up-kubernetes...
Remove unused controllers
Commit 1b7ba1e
Compare 05-add-screenshots
Branch 05-add-screenshots
igomezgithub
#35 passed
Ran for 8 min 24 sec
56 minutes ago
Restart build
Node.js: 12
AMD64
DOCKER_COMPOSE_VERSION=1.25.4
Job log
View config
Remove log
Raw log
Worker information
Build system information
$ sudo systemctl start docker
$ git clone --depth=50 --branch=05-add-screenshots https://github.com/igomezgithub/image-filter-microservice-project.git igomezgithub/image-filter-microservice-project


chunk {runtime} runtime.js, runtime.js.map (runtime) 9.77 kB [entry] [rendered]
chunk {shadow-css-4889ae62-23996f3f-js} shadow-css-4889ae62-23996f3f-js.js, shadow-css-4889ae62-23996f3f-js.js.map (shadow-css-4889ae62-23996f3f-js) 15.7 kB [rendered]
chunk {status-tap-32c72c43-js} status-tap-32c72c43-js.js, status-tap-32c72c43-js.js.map (status-tap-32c72c43-js) 1.83 kB [rendered]
chunk {styles} styles.js, styles.js.map (styles) 100 kB [initial] [rendered]
chunk {swipe-back-35ad8e37-js} swipe-back-35ad8e37-js.js, swipe-back-35ad8e37-js.js.map (swipe-back-35ad8e37-js) 2.67 kB [rendered]
chunk {swiper-bundle-8bab85e6-js} swiper-bundle-8bab85e6-js.js, swiper-bundle-8bab85e6-js.js.map (swiper-bundle-8bab85e6-js) 213 kB [rendered]
chunk {tap-click-ca00ce7f-js} tap-click-ca00ce7f-js.js, tap-click-ca00ce7f-js.js.map (tap-click-ca00ce7f-js) 6.32 kB [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 4.66 MB [initial] [rendered]
Date: 2021-09-12T09:46:37.290Z - Hash: 11af8b80dcebdc86e899 - Time: 10065ms
Removing intermediate container e09a85778aea
---> b0c41149a331
Step 7/8 : FROM nginx:alpine
---> 513f9a9d8748
Step 8/8 : COPY --from=ionic /usr/src/app/www /usr/share/nginx/html
---> f095e33e1abb
Successfully built f095e33e1abb
Successfully tagged igomezdockerhub/udagram-frontend:v4
The command "docker-compose -f docker-compose-build.yaml build" exited with 0.
store build cache
$ echo "$DOCKER_PASSWORD" | docker login -u "$DOCKER_USERNAME" --password-stdin
$ docker push "$DOCKER_USERNAME"/udagram-api-user:v4
$ docker push "$DOCKER_USERNAME"/udagram-api-feed:v4
$ docker push "$DOCKER_USERNAME"/reverseproxy:v4
$ docker push "$DOCKER_USERNAME"/udagram-frontend:v4
Done. Your build exited with 0.
Top


Travis CI
Dashboard
Changelog
Documentation
Help
Starred repositories
You can keep track of your favorite repositories here once you start starring!
Active repositories
My builds
igomezgithub
image-filter-microservic
DEFAULT BRANCH
main passed
LAST BUILD
#35 passed
COMMIT
1b7ba1e
FINISHED
58 minutes ago
igomezgithub
simple-node-with-travis
DEFAULT BRANCH
main
LAST BUILD
#2 passed
COMMIT
3b96be3
FINISHED
13 days ago


dockerhub
Search for great content (e.g., mysql)
Explore
Repositories
Organizations
Help
igomezdockerhub
Search by repository name
Create Repository
igomezdockerhub / udagram-frontend
Updated an hour ago
Not Scanned
0
100
Public
igomezdockerhub / reverseproxy
Updated an hour ago
Not Scanned
0
81
Public
igomezdockerhub / udagram-api-feed
Updated an hour ago
Not Scanned
0
727
Public
igomezdockerhub / udagram-api-user
Updated an hour ago
Not Scanned
0
125
Public
igomezdockerhub / simple-node-with-travis
Updated 13 days ago
Not Scanned
0
4
Public
igomezdockerhub / simple-node
Updated 13 days ago
Not Scanned
0
4
Public
Tip: Not finding your repository? Try switching namespace via the top left dropdown.

Service Orchestration with Kubernetes

- Deploy microservices using a Kubernetes cluster on AWS:

a) A screenshots of `kubectl` commands show the Frontend and API projects deployed in Kubernetes:

```
ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/
icroservices and Deploy/image-filter-microservice-project (05-add-screenshots)
$ kubectl get deployments
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
backend-feed   2/2     2            2           25h
backend-user   2/2     2            2           25h
frontend       1/1     1            1           25h
reverseproxy   1/1     1            1           25h
```

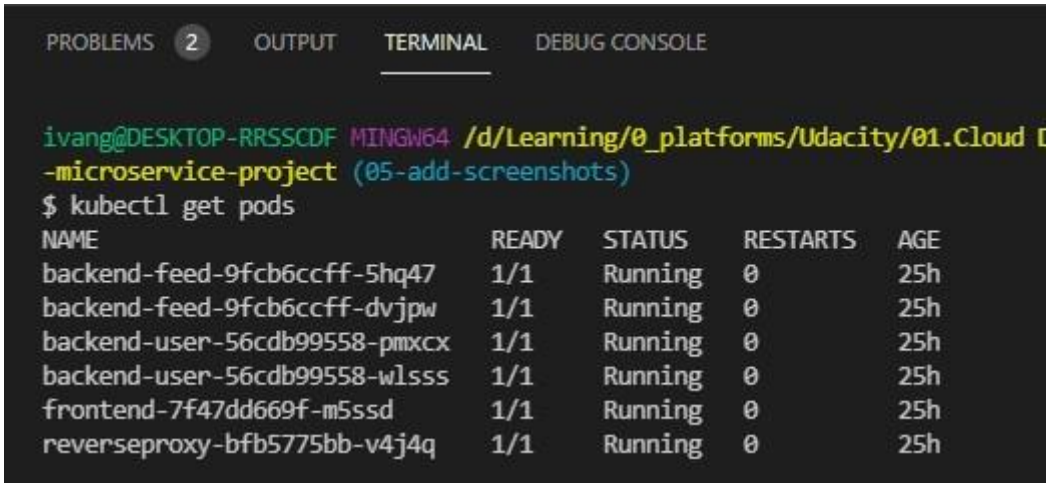
```
ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to Microservices/Project -
icroservices and Deploy/image-filter-microservice-project (05-add-screenshots)
$ kubectl describe deployments
Name:                   backend-feed
Namespace:              default
CreationTimestamp:      Sat, 11 Sep 2021 10:14:13 +0100
Labels:                 app=backend-feed
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=backend-feed
Replicas:               2 desired | 2 updated | 2 total | 2 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=backend-feed
  Containers:
   backend-feed:
    Image:      igomezdockerhub/udagram-api-feed:v4
    Port:       <none>
    Host Port:  <none>
    Environment:
      URL:                 <set to the key 'URL' of config map 'env-config'>                Optional: false
      AWS_MEDIA_BUCKET:    <set to the key 'AWS_MEDIA_BUCKET' of config map 'env-config'>   Optional: false
      AWS_PROFILE:         <set to the key 'AWS_PROFILE' of config map 'env-config'>        Optional: false
      AWS_REGION:          <set to the key 'AWS_REGION' of config map 'env-config'>         Optional: false
      JWT_SECRET:          <set to the key 'JWT_SECRET' in secret 'env-secret'>             Optional: false
      POSTGRESS_DATABASE:  <set to the key 'POSTGRESS_DATABASE' of config map 'env-config'> Optional: false
      POSTGRESS_HOST:      <set to the key 'POSTGRESS_HOST' of config map 'env-config'>     Optional: false
      POSTGRESS_PASSWORD:  <set to the key 'POSTGRESS_PASSWORD' in secret 'env-secret'>     Optional: false
      POSTGRESS_USERNAME:  <set to the key 'POSTGRESS_USERNAME' in secret 'env-secret'>     Optional: false
    Mounts:
      /root/.aws from aws-secret (ro)
  Volumes:
   aws-secret:
    Type:        Secret (a volume populated by a Secret)
    SecretName:  aws-secret
    Optional:    false
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   backend-feed-9fcb6ccff (2/2 replicas created)
Events:          <none>
```

```
Name:                   backend-user
Namespace:              default
CreationTimestamp:      Sat, 11 Sep 2021 10:14:28 +0100
Labels:                 app=backend-user
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=backend-user
Replicas:               2 desired | 2 updated | 2 total | 2 available | 0 unavailable
StrategyType:           RollingUpdate
MinReadySeconds:        0
RollingUpdateStrategy:  25% max unavailable, 25% max surge
Pod Template:
  Labels:  app=backend-user
  Containers:
   backend-user:
    Image:      igomezdockerhub/udagram-api-user:v4
    Port:       <none>
    Host Port:  <none>
    Environment:
      URL:                 <set to the key 'URL' of config map 'env-config'>                Optional: false
      AWS_MEDIA_BUCKET:    <set to the key 'AWS_MEDIA_BUCKET' of config map 'env-config'>    Optional: false
      AWS_PROFILE:         <set to the key 'AWS_PROFILE' of config map 'env-config'>         Optional: false
      AWS_REGION:          <set to the key 'AWS_REGION' of config map 'env-config'>          Optional: false
      JWT_SECRET:          <set to the key 'JWT_SECRET' in secret 'env-secret'>              Optional: false
      POSTGRESS_DATABASE:  <set to the key 'POSTGRESS_DATABASE' of config map 'env-config'>  Optional: false
      POSTGRESS_HOST:      <set to the key 'POSTGRESS_HOST' of config map 'env-config'>      Optional: false
      POSTGRESS_PASSWORD:  <set to the key 'POSTGRESS_PASSWORD' in secret 'env-secret'>      Optional: false
      POSTGRESS_USERNAME:  <set to the key 'POSTGRESS_USERNAME' in secret 'env-secret'>      Optional: false
    Mounts:
      /root/.aws from aws-secret (ro)
  Volumes:
   aws-secret:
    Type:        Secret (a volume populated by a Secret)
    SecretName:  aws-secret
    Optional:    false
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   backend-user-56cdb99558 (2/2 replicas created)
Events:          <none>
```

```
Name:                   frontend
Namespace:              default
CreationTimestamp:      Sat, 11 Sep 2021 10:15:02 +0100
Labels:                 app=frontend
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=frontend
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:  api=external
           app=frontend
  Containers:
   frontend:
    Image:        igomezdockerhub/udagram-frontend:v4
    Port:         80/TCP
    Host Port:    0/TCP
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   frontend-7f47dd669f (1/1 replicas created)
Events:          <none>
```

```
Name:                   reverseproxy
Namespace:              default
CreationTimestamp:      Sat, 11 Sep 2021 10:14:46 +0100
Labels:                 app=reverseproxy
Annotations:            deployment.kubernetes.io/revision: 1
Selector:               app=reverseproxy
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=reverseproxy
  Containers:
   reverseproxy:
    Image:      igomezdockerhub/reverseproxy:v4
    Port:       8080/TCP
    Host Port:  0/TCP
    Limits:
      cpu:     500m
      memory:  1Gi
    Requests:
      cpu:        250m
      memory:     64Mi
    Environment:  <none>
    Mounts:       <none>
  Volumes:        <none>
Conditions:
  Type           Status  Reason
  ----           ------  ------
  Available      True    MinimumReplicasAvailable
  Progressing    True    NewReplicaSetAvailable
OldReplicaSets:  <none>
NewReplicaSet:   reverseproxy-bfb5775bb (1/1 replicas created)
Events:          <none>
```

b) The output of `kubectl get pods` indicates that the pods are running successfully with the `STATUS` value `Running`:



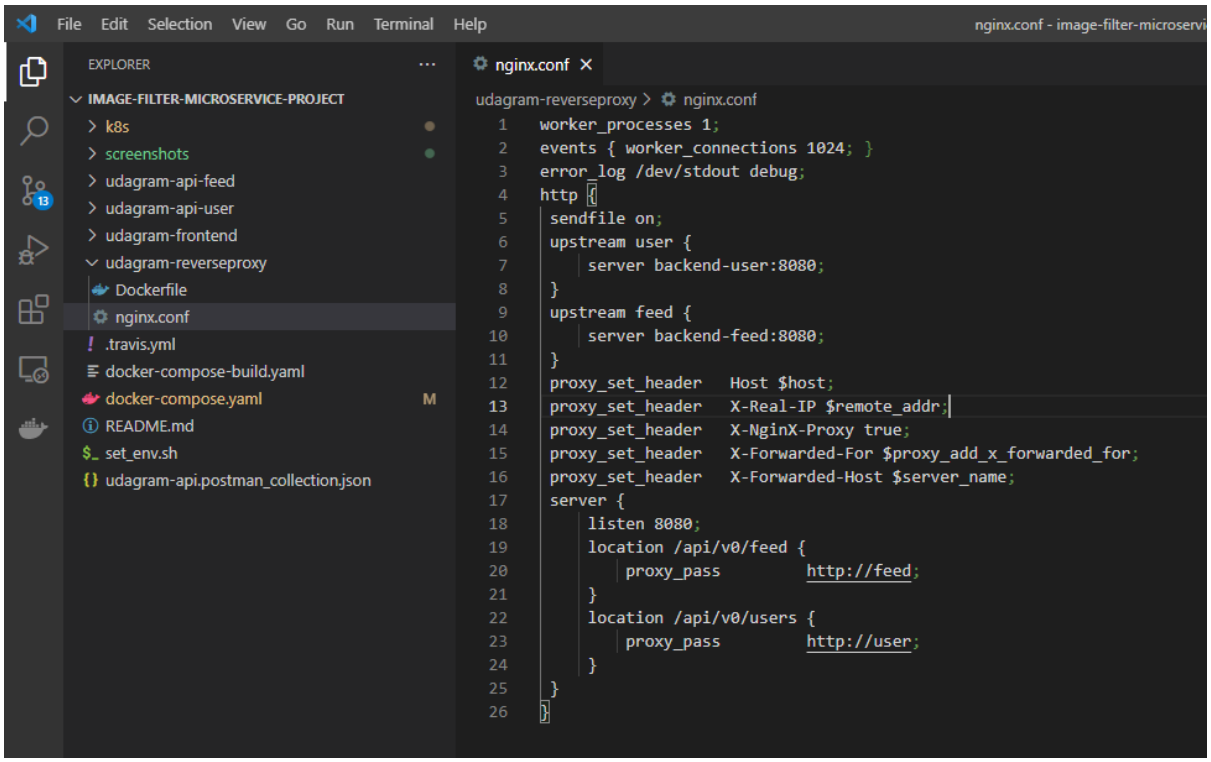
c) The output of `kubectl describe services` does not expose any sensitive strings such as database passwords:

```
ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud D
-microservice-project (05-add-screenshots)
$ kubectl describe services
Name:              backend-feed
Namespace:         default
Labels:            app=backend-feed
Annotations:       <none>
Selector:          app=backend-feed
Type:              ClusterIP
IP:                10.100.149.234
Port:              8080  8080/TCP
TargetPort:        8080/TCP
Endpoints:         172.30.0.159:8080,172.30.1.149:8080
Session Affinity:  None
Events:            <none>


Name:              backend-user
Namespace:         default
Labels:            app=backend-user
Annotations:       <none>
Selector:          app=backend-user
Type:              ClusterIP
IP:                10.100.248.40
Port:              8080  8080/TCP
TargetPort:        8080/TCP
Endpoints:         172.30.0.207:8080,172.30.1.243:8080
Session Affinity:  None
Events:            <none>


Name:              frontend
Namespace:         default
Labels:            app=frontend
Annotations:       <none>
Selector:          app=frontend
Type:              ClusterIP
IP:                10.100.7.88
Port:              8100  8100/TCP
TargetPort:        80/TCP
Endpoints:         172.30.1.115:80
Session Affinity:  None
Events:            <none>
```

```
Name:              kubernetes
Namespace:         default
Labels:            component=apiserver
                   provider=kubernetes
Annotations:       <none>
Selector:          <none>
Type:              ClusterIP
IP:                10.100.0.1
Port:              https  443/TCP
TargetPort:        443/TCP
Endpoints:         172.30.0.228:443,172.30.1.54:443
Session Affinity:  None
Events:            <none>


Name:                     publicfrontend
Namespace:                default
Labels:                   app=frontend
Annotations:              <none>
Selector:                 app=frontend
Type:                     LoadBalancer
IP:                       10.100.77.218
LoadBalancer Ingress:     a2739bf9539104768bc3c56041b3dcea-58075089.us-east-2.elb.amazonaws.com
Port:                     <unset>  80/TCP
TargetPort:               80/TCP
NodePort:                 <unset>  30780/TCP
Endpoints:                172.30.1.115:80
Session Affinity:         None
External Traffic Policy:  Cluster
Events:                   <none>
```

```
Name:                     publicreverseproxy
Namespace:                default
Labels:                   app=reverseproxy
Annotations:              <none>
Selector:                 app=reverseproxy
Type:                     LoadBalancer
IP:                       10.100.88.159
LoadBalancer Ingress:     a85b3be637240486aa7288bbdea0a178-1777650885.us-east-2.elb.amazonaws.com
Port:                     <unset>  8080/TCP
TargetPort:               8080/TCP
NodePort:                 <unset>  30997/TCP
Endpoints:                172.30.0.28:8080
Session Affinity:         None
External Traffic Policy:  Cluster
Events:                   <none>


Name:              reverseproxy
Namespace:         default
Labels:            app=reverseproxy
Annotations:       <none>
Selector:          app=reverseproxy
Type:              ClusterIP
IP:                10.100.129.167
Port:              8080  8080/TCP
TargetPort:        8080/TCP
Endpoints:         172.30.0.28:8080
Session Affinity:  None
Events:            <none>
```

- Use a reverse proxy to direct requests to the appropriate backend:




```
Name:                     publicreverseproxy
Namespace:                default
Labels:                   app=reverseproxy
Annotations:              <none>
Selector:                 app=reverseproxy
Type:                     LoadBalancer
IP:                       10.100.88.159
LoadBalancer Ingress:     a85b3be637240486aa7288bbdea0a178-1777650885.us-east-2.elb.amazonaws.com
Port:                     <unset>  8080/TCP
TargetPort:               8080/TCP
NodePort:                 <unset>  30997/TCP
Endpoints:                172.30.0.28:8080
Session Affinity:         None
External Traffic Policy:  Cluster
Events:                   <none>


Name:              reverseproxy
Namespace:         default
Labels:            app=reverseproxy
Annotations:       <none>
Selector:          app=reverseproxy
Type:              ClusterIP
IP:                10.100.129.167
Port:              8080  8080/TCP
TargetPort:        8080/TCP
Endpoints:         172.30.0.28:8080
Session Affinity:  None
Events:            <none>
```

- Configure scaling and self-healing for each service:

```
ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to M
vice-project (05-add-screenshots)
$ kubectl get pods
NAME                               READY   STATUS    RESTARTS   AGE
backend-feed-9fcb6ccff-5hq47       1/1     Running   0          26h
backend-feed-9fcb6ccff-dvjpw       1/1     Running   0          26h
backend-user-56cdb99558-pmxcx      1/1     Running   0          26h
backend-user-56cdb99558-wlsss      1/1     Running   0          26h
frontend-7f47dd669f-m5ssd          1/1     Running   0          26h
reverseproxy-bfb5775bb-v4j4q       1/1     Running   0          26h

ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to M
vice-project (05-add-screenshots)
$ kubectl autoscale deployment backend-feed --cpu-percent=50 --min=1 --max=2
horizontalpodautoscaler.autoscaling/backend-feed autoscaled

ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to M
vice-project (05-add-screenshots)
$ kubectl autoscale deployment backend-user --cpu-percent=50 --min=1 --max=2
horizontalpodautoscaler.autoscaling/backend-user autoscaled

ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to M
vice-project (05-add-screenshots)
$ kubectl get hpa
NAME           REFERENCE                 TARGETS         MINPODS   MAXPODS   REPLICAS   AGE
backend-feed   Deployment/backend-feed   <unknown>/50%   1         2         2          2m29s
backend-user   Deployment/backend-user   <unknown>/50%   1         2         2          31s
```

Review:

- I have add this settings in the deployment files (e.g. backend-feeed-deployment.yaml):

backend-feed-deployment.yaml

```yaml
  selector:
    matchLabels:
      app: backend-feed
  template:
    metadata:
      labels:
        app: backend-feed
    spec:
      containers:
      - name: backend-feed
        image: igomezdockerhub/udagram-api-feed:v4
        imagePullPolicy: Always
        resources:
          requests:
            memory: "64Mi"
            cpu: "250m"
          limits:
            memory: "2048Mi"
            cpu: "1000m"
        env:
        - name: URL
          valueFrom:
            configMapKeyRef:
              name: env-config
              key: URL
        - name: AWS_MEDIA_BUCKET
          valueFrom:
            configMapKeyRef:
              name: env-config
```

- I have configurated the metrics server and then the CPU is displayed:

```
ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to Microservices/Project - Udagram - Refactor Mo
vice-project/k8s (05-add-screenshots)
$ kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/download/v0.3.7/components.yaml
clusterrole.rbac.authorization.k8s.io/system:aggregated-metrics-reader created
clusterrolebinding.rbac.authorization.k8s.io/metrics-server:system:auth-delegator created
rolebinding.rbac.authorization.k8s.io/metrics-server-auth-reader created
Warning: apiregistration.k8s.io/v1beta1 APIService is deprecated in v1.19+, unavailable in v1.22+; use apiregistration.k8s.io/v1 APIService
apiservice.apiregistration.k8s.io/v1beta1.metrics.k8s.io created
serviceaccount/metrics-server created
deployment.apps/metrics-server created
service/metrics-server created
clusterrole.rbac.authorization.k8s.io/system:metrics-server created
clusterrolebinding.rbac.authorization.k8s.io/system:metrics-server created

ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to Microservices/Project - Udagram - Refactor Mo
vice-project/k8s (05-add-screenshots)
$ kubectl get deployment metrics-server -n kube-system
NAME             READY   UP-TO-DATE   AVAILABLE   AGE
metrics-server   1/1     1            1           12s

ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to Microservices/Project - Udagram - Refactor Mo
vice-project/k8s (05-add-screenshots)
$ kubectl get hpa
NAME           REFERENCE                 TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
backend-feed   Deployment/backend-feed   0%/50%    1         2         1          11m
backend-user   Deployment/backend-user   0%/50%    1         2         1          10m
```

```
ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to Microservices/Project - Udagram - Refa
vice-project/k8s (05-add-screenshots)
$ kubectl describe hpa
Name:                                                  backend-feed
Namespace:                                             default
Labels:                                                <none>
Annotations:                                           <none>
CreationTimestamp:                                     Sun, 12 Sep 2021 20:09:25 +0100
Reference:                                             Deployment/backend-feed
Metrics:                                               ( current / target )
  resource cpu on pods  (as a percentage of request):  0% (0) / 50%
  AbleToScale     True    ReadyForNewScale  recommended size matches current size
  ScalingActive   True    ValidMetricFound  the HPA was able to successfully calculate a replica count from cpu resource utilization (
percentage of request)
  ScalingLimited  True    TooFewReplicas    the desired replica count is less than the minimum replica count
Events:
  Type     Reason                        Age                From                       Message
  ----     ------                        ----               ----                       -------
  Warning  FailedComputeMetricsReplicas  56m (x12 over 59m)  horizontal-pod-autoscaler  invalid metrics (1 invalid out of 1), first er
ror is: failed to get cpu utilization: unable to get metrics for resource cpu: unable to fetch metrics from resource metrics API: the
server could not find the requested resource (get pods.metrics.k8s.io)
  Warning  FailedGetResourceMetric       54m (x21 over 59m)  horizontal-pod-autoscaler  failed to get cpu utilization: unable to get m
etrics for resource cpu: unable to fetch metrics from resource metrics API: the server could not find the requested resource (get pods
.metrics.k8s.io)
```

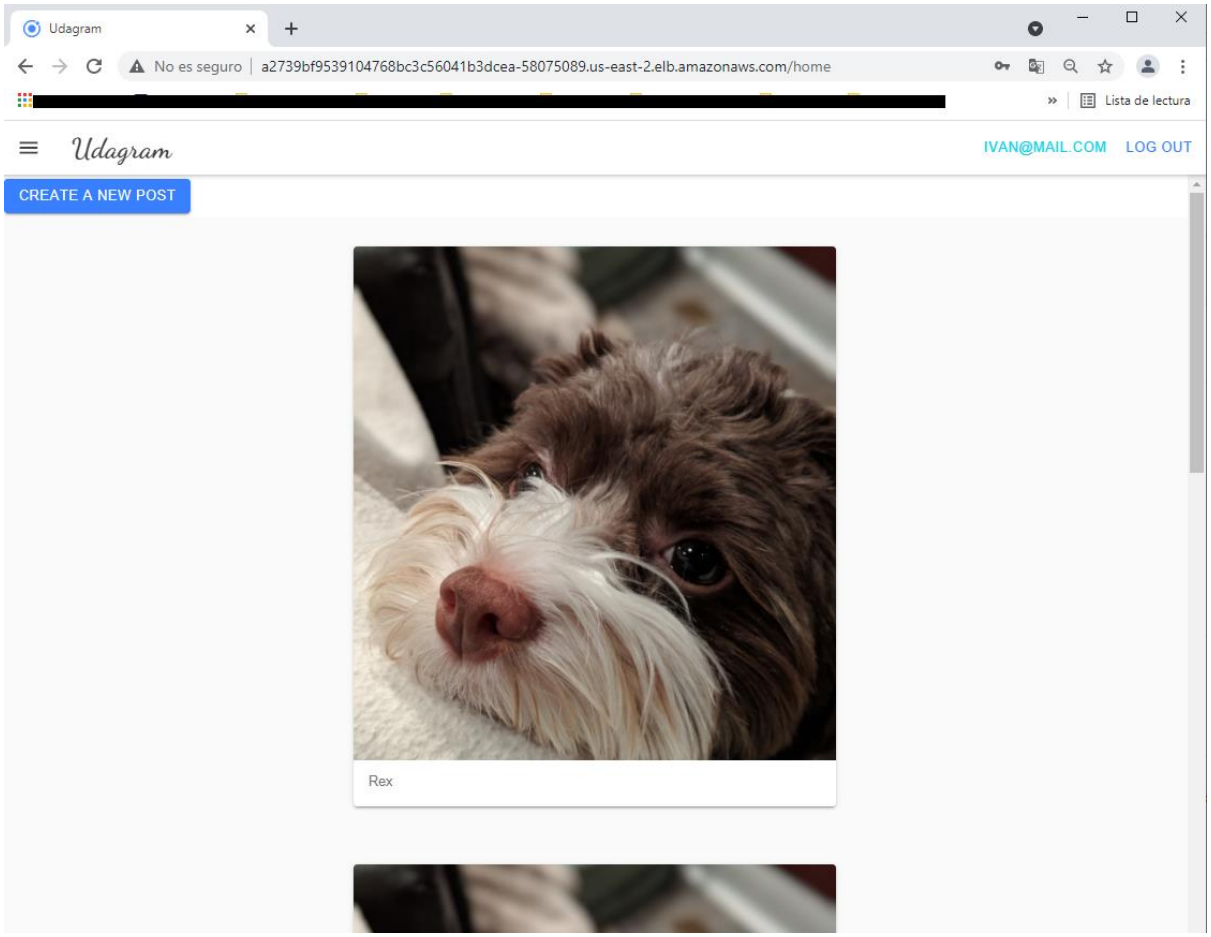
Debugging, Monitoring, and Logging

- Use logs to capture metrics for debugging a microservices deployment:

```
ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to Microservices/Project - Udagram - Ref
actor Monolith to Microservices and Deploy/image-filter-microservice-project (05-add-screenshots)
$ kubectl get pods
NAME                              READY   STATUS    RESTARTS   AGE
backend-feed-9fcb6ccff-5hq47      1/1     Running   0          26h
backend-feed-9fcb6ccff-dvjpw      1/1     Running   0          26h
backend-user-56cdb99558-pmxcx     1/1     Running   0          26h
backend-user-56cdb99558-wlsss     1/1     Running   0          26h
frontend-7f47dd669f-m5ssd         1/1     Running   0          26h
reverseproxy-bfb5775bb-v4j4q      1/1     Running   0          26h

ivang@DESKTOP-RRSSCDF MINGW64 /d/Learning/0_platforms/Udacity/01.Cloud Developer/04.Monolith to Microservices/Project - Udagram - Ref
actor Monolith to Microservices and Deploy/image-filter-microservice-project (05-add-screenshots)
$ kubectl logs backend-user-56cdb99558-pmxcx

> udagram-api@2.0.0 prod /usr/src/app
> tsc && node ./www/server.js

Initialize database connection...
Executing (default): CREATE TABLE IF NOT EXISTS "User" ("email" VARCHAR(255) , "password_hash" VARCHAR(255), "createdAt" TIMESTAMP WI
TH TIME ZONE, "updatedAt" TIMESTAMP WITH TIME ZONE, PRIMARY KEY ("email"));
Executing (default): SELECT i.relname AS name, ix.indisprimary AS primary, ix.indisunique AS unique, ix.indkey AS indkey, array_agg(a
.attnum) as column_indexes, array_agg(a.attname) AS column_names, pg_get_indexdef(ix.indexrelid) AS definition FROM pg_class t, pg_cl
ass i, pg_index ix, pg_attribute a WHERE t.oid = ix.indrelid AND i.oid = ix.indexrelid AND a.attrelid = t.oid AND t.relkind = 'r' and
 t.relname = 'User' GROUP BY i.relname, ix.indexrelid, ix.indisprimary, ix.indisunique, ix.indkey ORDER BY i.relname;
server running http://a2739bf9539104768bc3c56041b3dcea-58075089.us-east-2.elb.amazonaws.com:8100
press CTRL+C to stop server
Executing (default): SELECT "email", "password_hash", "createdAt", "updatedAt" FROM "User" AS "User" WHERE "User"."email" = 'ivan@mai
l.com';
```

Web Page

URL: **http://a2739bf9539104768bc3c56041b3dcea-58075089.us-east-2.elb.amazonaws.com/**


Udagram
No es seguro | a2739bf9539104768bc3c56041b3dcea-58075089.us-east-2.elb.amazonaws.com/home
Lista de lectura
Udagram
IVAN@MAIL.COM
LOG OUT
CREATE A NEW POST
Rex