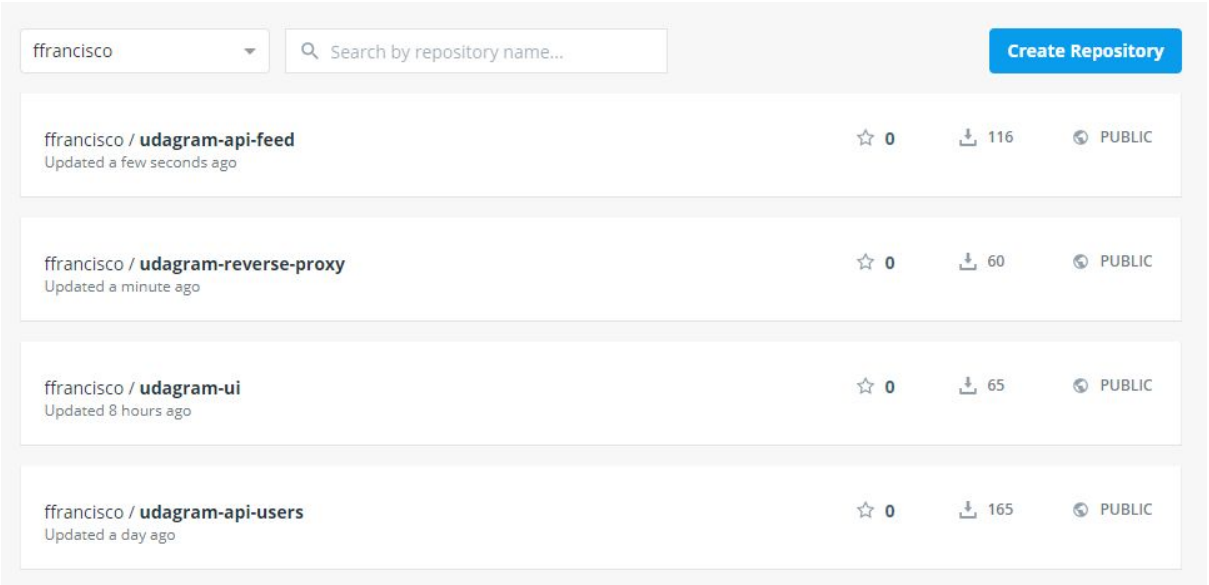
Project Rubric

Build and run a container image using Docker



Use Travis to build a CI/CD pipeline

Deploy microservices using a Kubernetes cluster on AWS

```
PS I:\> kubectl get all
NAME                                        READY   STATUS    RESTARTS   AGE
pod/udagram-api-feed-7fd9c66d66-rdc2m       1/1     Running   0          8h
pod/udagram-api-users-598588c9cd-nj48g      1/1     Running   0          8h
pod/udagram-reverse-proxy-979bcff8d-llxjd   1/1     Running   0          8h
pod/udagram-ui-bc85dbc44-6sn22              1/1     Running   0          7h53m

NAME                            TYPE           CLUSTER-IP       EXTERNAL-IP                                                                  PORT(S)          AGE
service/kubernetes              ClusterIP      10.100.0.1       <none>                                                                       443/TCP          3d1h
service/udagram-api-feed        ClusterIP      10.100.108.1     <none>                                                                       8080/TCP         8h
service/udagram-api-users       ClusterIP      10.100.224.203   <none>                                                                       8080/TCP         8h
service/udagram-reverse-proxy   LoadBalancer   10.100.225.6     a93b2e8de285648bcbc832562fee233f-211218333.eu-central-1.elb.amazonaws.com   8080:31356/TCP   8h
service/udagram-ui              LoadBalancer   10.100.239.246   a1089a60961374bc4acae2e332373f35-102837227.eu-central-1.elb.amazonaws.com   8100:30297/TCP   7h53m

NAME                                    READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/udagram-api-feed        1/1     1            1           8h
deployment.apps/udagram-api-users       1/1     1            1           8h
deployment.apps/udagram-reverse-proxy   1/1     1            1           8h
deployment.apps/udagram-ui              1/1     1            1           7h53m

NAME                                              DESIRED   CURRENT   READY   AGE
replicaset.apps/udagram-api-feed-7fd9c66d66       1         1         1       8h
replicaset.apps/udagram-api-users-598588c9cd      1         1         1       8h
replicaset.apps/udagram-reverse-proxy-979bcff8d   1         1         1       8h
replicaset.apps/udagram-ui-bc85dbc44              1         1         1       7h53m

NAME                                                     REFERENCE                           TARGETS   MINPODS   MAXPODS   REPLICAS   AGE
horizontalpodautoscaler.autoscaling/udagram-api-feed   Deployment/udagram-api-feed   0%/25%    1         3         1          7h26m
horizontalpodautoscaler.autoscaling/udagram-ui         Deployment/udagram-ui         0%/25%    1         3         1          7h26m
```

Get Pods

```
PS C:\> kubectl get pods
NAME                                    READY   STATUS    RESTARTS   AGE
udagram-api-feed-7fd9c66d66-rdc2m       1/1     Running   0          8h
udagram-api-users-598588c9cd-nj48g      1/1     Running   0          8h
udagram-reverse-proxy-979bcff8d-llxjd   1/1     Running   0          8h
udagram-ui-bc85dbc44-6sn22              1/1     Running   0          7h54m
```

The output of kubectl describe services does not expose any sensitive strings such as database passwords.

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: 172.31.15.56:443,172.31.44.147:443
Session Affinity: None
Events: <none>

Name: udagram-api-feed
Namespace: default
Labels: <none>
Annotations: kubectl.kubernetes.io/last-applied-configuration:

{"apiVersion":"v1","kind":"Service","metadata":{"annotations":{},"name":"udagram-api-feed","
namespace":"default"},"spec":{"ports":[{"name"...
Selector: app=udagram-api-feed
Type: ClusterIP
IP: 10.100.108.1
Port: 8080 8080/TCP
TargetPort: 8080/TCP
Endpoints: 172.31.39.44:8080
Session Affinity: None
Events: <none>

Name: udagram-api-users
Namespace: default
Labels: <none>
Annotations: kubectl.kubernetes.io/last-applied-configuration:

{"apiVersion":"v1","kind":"Service","metadata":{"annotations":{},"name":"udagram-api-users",
"namespace":"default"},"spec":{"ports":[{"name...
Selector: app=udagram-api-users
Type: ClusterIP
IP: 10.100.224.203
Port: 8080 8080/TCP
TargetPort: 8080/TCP
Endpoints: 172.31.41.140:8080
Session Affinity: None
Events: <none>

Name: udagram-reverse-proxy
Namespace: default
Labels: <none>
Annotations: kubectl.kubernetes.io/last-applied-configuration:

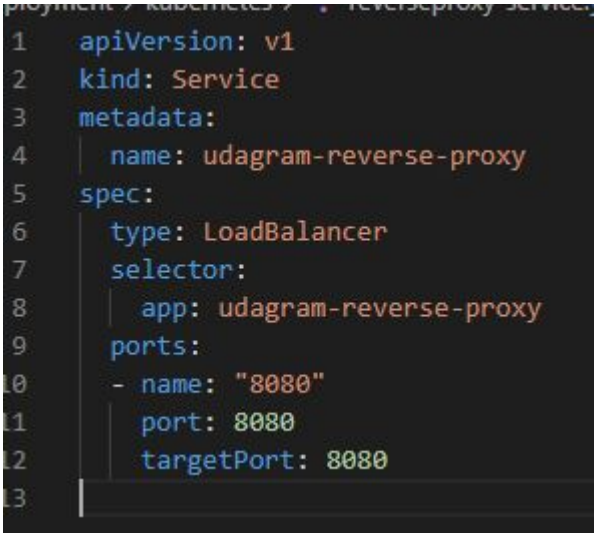
{"apiVersion":"v1","kind":"Service","metadata":{"annotations":{},"name":"udagram-reverse-pr
oxy","namespace":"default"},"spec":{"ports":[{"...
Selector: app=udagram-reverse-proxy
Type: LoadBalancer
IP: 10.100.225.6
LoadBalancer Ingress:
a93b2e8de285648bcbc832562fee233f-211218333.eu-central-1.elb.amazonaws.com
Port: 8080 8080/TCP
TargetPort: 8080/TCP
NodePort: 8080 31356/TCP
Endpoints: 172.31.15.223:8080

Session Affinity: None
External Traffic Policy: Cluster
Events: <none>

Name: udagram-ui
Namespace: default
Labels: <none>
Annotations: kubectl.kubernetes.io/last-applied-configuration:

{"apiVersion":"v1","kind":"Service","metadata":{"annotations":{},"name":"udagram-ui","namespace":"default"},"spec":{"ports":[{"name":"8100...
Selector: app=udagram-ui
Type: LoadBalancer
IP: 10.100.239.246
LoadBalancer Ingress: a1089a60961374bc4acae2e332373f35-102837227.eu-central-1.elb.amazonaws.com
Port: 8100 8100/TCP
TargetPort: 8100/TCP
NodePort: 8100 30297/TCP
Endpoints: 172.31.39.107:8100
Session Affinity: None
External Traffic Policy: Cluster
Events: <none>

Screenshot of Kubernetes services shows a reverse proxy



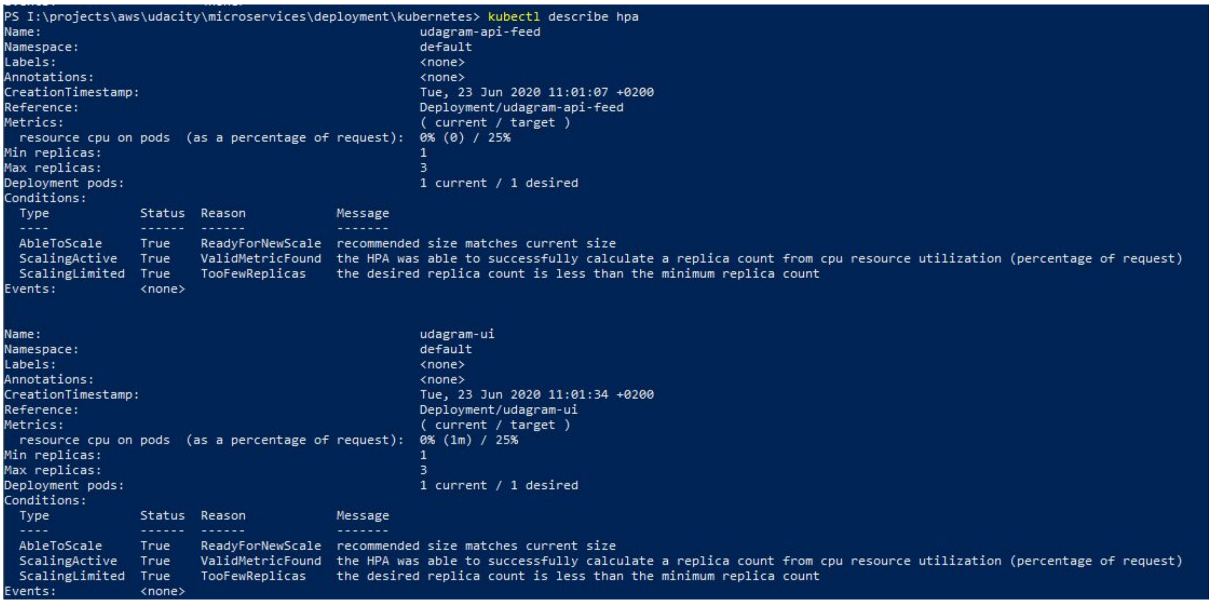
Configure scaling and self-healing for each service

deployment > kubernetes > ! udagram-api-feed.deployment.yaml > {} spec > # replicas

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: udagram-api-feed
  name: udagram-api-feed
spec:
  replicas: 2
  selector:
    matchLabels:
      app: udagram-api-feed
  template:
    metadata:
      labels:
        app: udagram-api-feed
    spec:
      containers:
      - name: udagram-api-feed
        image: ffrancisco/udagram-api-feed
        envFrom:
        - configMapRef:
            name: app-settings
        resources:
          requests:
            memory: "64Mi"
            cpu: "250m"
          limits:
            memory: "1024Mi"
            cpu: "500m"
        ports:
        - containerPort: 8080
        volumeMounts:
        - name: aws-secret
          mountPath: /usr/src/app/secret/
          readOnly: true
      volumes:
      - name: aws-secret
        secret:
          secretName: aws-secret-file

```

Screenshot of Kubernetes cluster of command kubectl describe hpa has autoscaling configured with CPU metrics.



Screenshot of one of the backend API pod logs indicates user activity that is logged when an API call is made.