

Twitter-like app

Fall 2020, Presentation

By Yang Liu, Nuo Xu, Weibo Dai(Group 6)

Run and Deploy the application

Functionality introductions

Run, Deploy and monitor the application

- 1. Run the application on local machine**
- 2. Deploy and run the app on minikube**
- 3. Deploy and run the app on EKS**

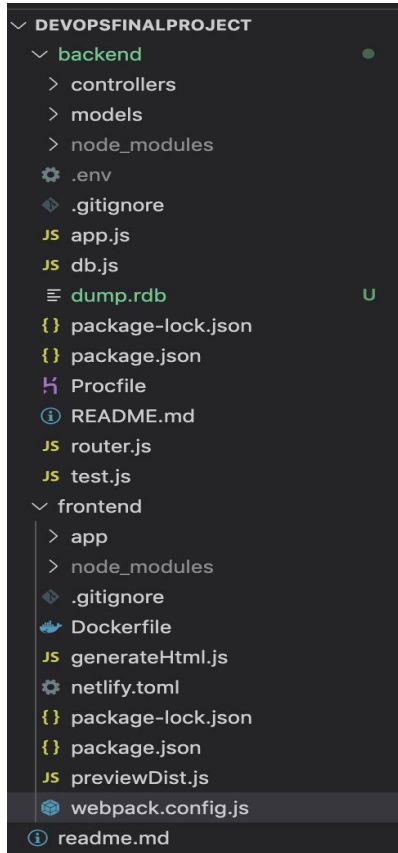
Services Implementations

Front-end: React

Back-end: Node.js, Redis

Database: Mongo Atlas

Deployment File



Run and Deploy the application

Run the application on local machine

1. Go to the front end app folder and run npm install
2. Run npm run dev
3. Add .env file under the back end folder
4. Go to back end app folder and run npm start
5. Go to <http://localhost:3000/>

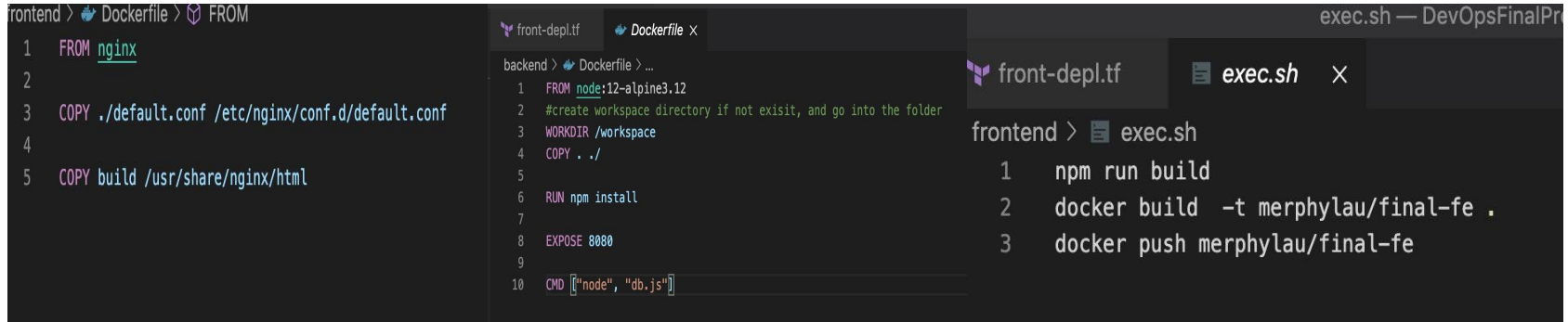
Run and Deploy the application

Run the application on local machine

1. Go to the front end app folder and run npm install
2. Run npm run dev
3. Add .env file under the back end folder
4. Go to back end app folder and run npm start
5. Go to <http://localhost:3000/>

Deploy the app on EKS

First, dockerize the apps. Then build and push the app to docker hub by running the exec.sh file.



```

frontend > Dockerfile > FROM
1 FROM nginx
2
3 COPY ./default.conf /etc/nginx/conf.d/default.conf
4
5 COPY build /usr/share/nginx/html

front-depl.tf Dockerfile x
backend > Dockerfile > ...
1 FROM node:12-alpine3.12
2 #create workspace directory if not exist, and go into the folder
3 WORKDIR /workspace
4 COPY . ./
5
6 RUN npm install
7
8 EXPOSE 8080
9
10 CMD ["node", "db.js"]

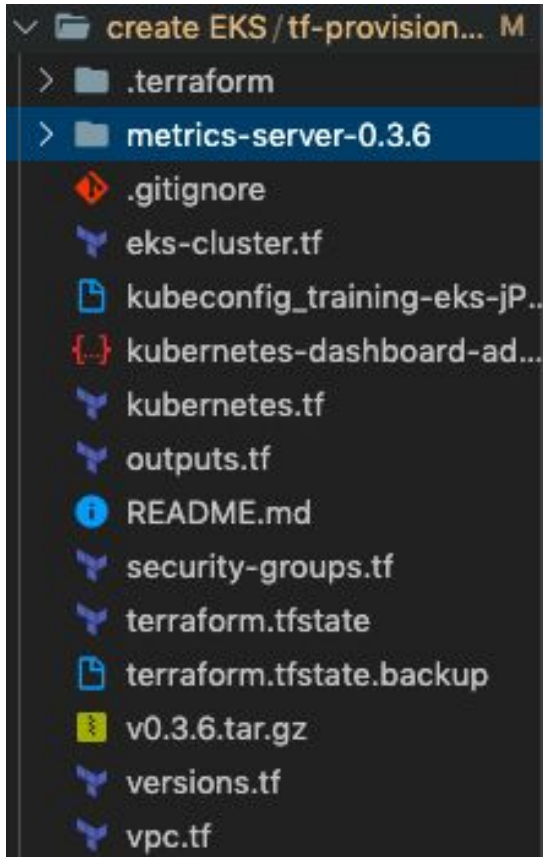
exec.sh — DevOpsFinalPr
front-depl.tf exec.sh x
frontend > exec.sh
1 npm run build
2 docker build -t merphylau/final-fe .
3 docker push merphylau/final-fe
  
```

Deploy the app on minikube

Second, run the yaml file of each services to deploy the app on the minikube.

```
kubectl delete deployment --all  
kubectl delete service --all  
  
kubectl apply -f front-svc.yaml  
kubectl apply -f redis.yaml  
kubectl apply -f back-svc.yaml
```


Deploy the app on EKS



After deployed on the minikube, run terraform init, terraform plan, terraform apply to create the eks.

Deploy the app on EKS

Example of tf files.

```
resource "aws_security_group" "worker_group_mgmt_one" {
  name_prefix = "worker_group_mgmt_one"
  vpc_id      = module.vpc.vpc_id

  ingress {
    from_port = 22
    to_port   = 22
    protocol  = "tcp"

    cidr_blocks = [
      "10.0.0.0/8",
    ]
  }
}

resource "aws_security_group" "worker_group_mgmt_two" {
  name_prefix = "worker_group_mgmt_two"
  vpc_id      = module.vpc.vpc_id

  ingress {
    from_port = 22
    to_port   = 22
    protocol  = "tcp"

    cidr_blocks = [
      "192.168.0.0/16",
    ]
  }
}

resource "aws_security_group" "all_worker_mgmt" {
  name_prefix = "all_worker_management"
  vpc_id      = module.vpc.vpc_id
```

```
variable "region" {
  default     = "us-east-2"
  description = "AWS region"
}

provider "aws" {
  version = ">= 2.28.1"
  region  = "us-east-2"
}

data "aws_availability_zones" "available" {}

locals {
  cluster_name = "training-eks-${random_string.suffix.result}"
}

resource "random_string" "suffix" {
  length  = 8
  special = false
}

module "vpc" {
  source = "terraform-aws-modules/vpc/aws"
  version = "2.6.0"

  name             = "training-vpc"
  cidr             = "10.0.0.0/16"
  azs              = data.aws_availability_zones.available.names
  private_subnets = ["10.0.1.0/24", "10.0.2.0/24", "10.0.3.0/24"]
  public_subnets  = ["10.0.4.0/24", "10.0.5.0/24", "10.0.6.0/24"]
  enable_nat_gateway = true
  single_nat_gateway = true
  enable_dns_hostnames = true
```

```
module "eks" {
  source      = "terraform-aws-modules/eks/aws"
  cluster_name = local.cluster_name
  cluster_version = "1.17"
  subnets    = module.vpc.private_subnets

  tags = {
    Environment = "training"
    GithubRepo  = "terraform-aws-eks"
    GithubOrg   = "terraform-aws-modules"
  }

  vpc_id = module.vpc.vpc_id

  worker_groups = [
    {
      name             = "worker-group-1"
      instance_type    = "t2.small"
      additional_userdata = "echo foo bar"
      asg_desired_capacity = 2
      additional_security_group_ids = [aws_security_group.worker_group_mgmt_one.id]
    },
    {
      name             = "worker-group-2"
      instance_type    = "t2.medium"
      additional_userdata = "echo foo bar"
      additional_security_group_ids = [aws_security_group.worker_group_mgmt_two.id]
      asg_desired_capacity = 1
    },
  ]

  data "aws_eks_cluster" "cluster" {
    name = module.eks.cluster_id
  }
```

Deploy the app on EKS

Example of yaml file(front-end service)

```
apiVersion: v1
kind: Service
metadata:
  name: final-front-svc
spec:
  type: LoadBalancer
  ports:
    - port: 80
      protocol: TCP
      targetPort: 80
  selector:
    app: final-front
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: final-front
spec:
  replicas: 3
  selector:
    matchLabels:
      app: final-front
  minReadySeconds: 15
  strategy:
    type: RollingUpdate
    rollingUpdate:
      maxUnavailable: 1
      maxSurge: 1
  template:
    metadata:
      labels:
        app: final-front
    spec:
      containers:
        - name: final-front
          image: merphylau/final-fe:latest
          imagePullPolicy: Always
          ports:
            - containerPort: 80
          resources:
            limits:
              cpu: '1'
            requests:
              cpu: '0.2'
```

Deploy the app on EKS

Finally deploy the services and the deployments.

```
kubectl delete deployment --all  
kubectl delete service --all  
  
kubectl apply -f front-svc.yaml  
kubectl apply -f redis.yaml  
kubectl apply -f back-svc.yaml  
kubectl apply -f autoScale.yaml
```

Functionality introductions

Basic:

1. Create, find, edit, delete posts
2. Auto delete offensive posts
3. Ability to use on phone

additional:

1. user login logout, sign up
2. user follow other users and see who follows the user
3. user can search tweets posts by users they following
4. Use metric server to monitor k8s
5. Auto scale pods

Basic Functions

1. Create, find, edit, delete posts

create

Twitter navigation sidebar:

- Home
- Explore
- Notifications
- Messages
- Bookmarks
- Lists
- Profile
- More

Buttons: Tweet, Sign Out

Form fields:

Title: DevOps

Body Content: DevOps is so cool!

Buttons: Save New Post

edit

Twitter navigation sidebar:

- Home
- Explore
- Notifications
- Messages
- Bookmarks
- Lists
- Profile
- More

Buttons: Tweet, Sign Out

Form fields:

Title: DevOps

Body Content: DevOps is so cool!
I like DevOps!!!

Buttons: Save Updates

find

Twitter navigation sidebar:

- Home
- Explore
- Notifications
- Messages
- Bookmarks
- Lists
- Profile
- More

Buttons: Tweet, Sign Out

Post was updated!

The latest from those you follow

amazon	by rezy on 12/14/2020
testing3	by yang on 12/13/2020
testing1	by yang on 12/13/2020

delete

Twitter navigation sidebar:

- Home
- Explore
- Notifications
- Messages
- Bookmarks
- Lists
- Profile
- More

Buttons: Tweet, Sign Out

...30122-1803852003-us-east-2.elb.amazonaws.com says

Do you really want to delete this post?

Cancel OK

posted by annew on 12/14/2020

testing3

Delete

2. Auto delete offensive posts

Browser address bar: a3b7c3fac94f049cfa65cb5a42130122-1803852003.us-east-2.elb.amazonaws.com/create-post

Twitter interface showing a post creation form with a warning message: "Your post contains offensive words, it has been deleted."

The form contains the text "shit" in the title field and "shit shit" in the body content field.

Buttons: Tweet, Sign Out, Save New Post

Additional features

1. sign up

Validation Check



Remember Writing?

Q Follow your interests.

Hear what people are talking about.

Join the conversation.

Username

Pick a username

You must provide a valid email address.

aaaa

Password

Create a password

Sign up for Twitter



Remember Writing?

Q Follow your interests.

Hear what people are talking about.

Join the conversation.

Username

Pick a username

Email

you@example.com

Password must be at least 12 characters.

Sign up for Twitter



Remember Writing?

Q Follow your interests.

Hear what people are talking about.

Join the conversation.

Username can only contain letters and numbers

aa

Email

you@example.com

Password

Create a password

Sign up for Twitter

2. user sign in and sing out

Main Page

The screenshot shows a web browser window with the address bar displaying 'localhost:3000'. The page title is 'Your Feed | ComplexApp'. The interface is divided into a left sidebar and a main content area.

Left Sidebar:

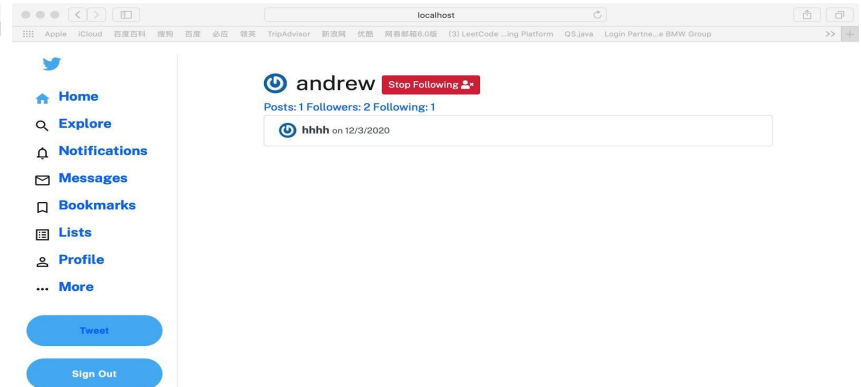
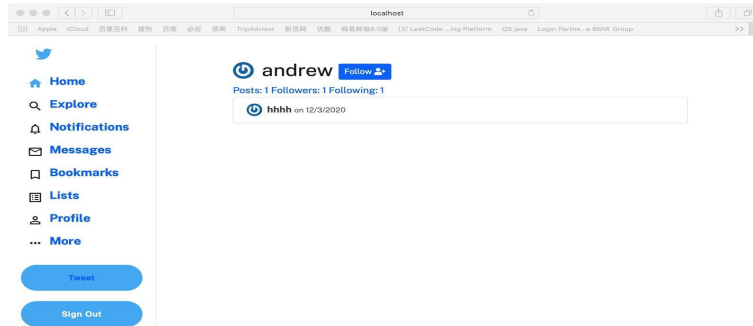
- Twitter logo
- Home (with house icon)
- Explore (with magnifying glass icon)
- Notifications (with bell icon)
- Messages (with envelope icon)
- Bookmarks (with bookmark icon)
- Lists (with list icon)
- Profile (with person icon)
- More (with three dots icon)
- Tweet button
- Sign Out button

Main Content Area:

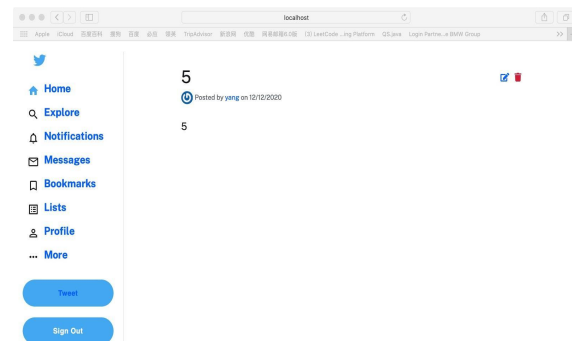
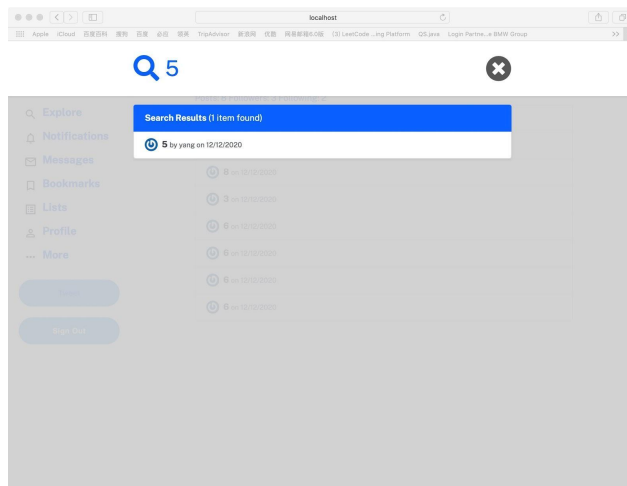
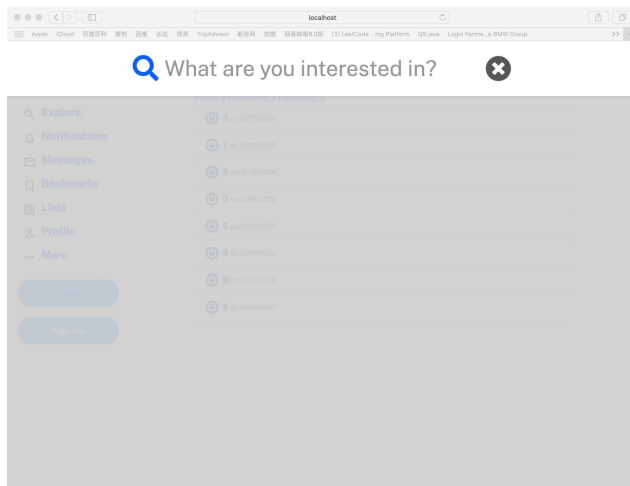
The latest from those you follow

5	by yang on 12/12/2020
7	by yang on 12/12/2020
8	by yang on 12/12/2020
3	by yang on 12/12/2020
6	by yang on 12/12/2020
6	by yang on 12/12/2020
6	by yang on 12/12/2020
6	by yang on 12/12/2020

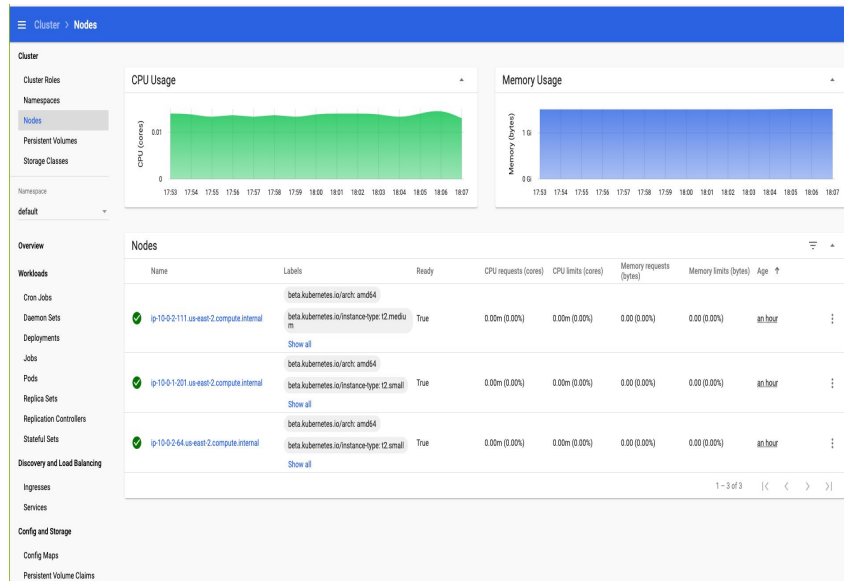
3. user follow other users and see who follows the user



4. user search for favourite tweets in following users



5. Monitoring the app using metrics server.



Overview

Discovery and Load Balancing

Cluster Roles

Namespaces

Nodes

Persistent Volumes

Storage Classes

Namespace

default

Overview

Workloads

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Discovery and Load Balancing

Ingresses

Services

Config and Storage

Config Maps

Persistent Volume Claims

Name	Namespace	Labels	Cluster IP	Internal Endpoints	External Endpoints	Age
final-back-svc	default	-	172.20.201.106	final-back-svc:8080 TCP final-back-svc:31743 TCP	alb-2cd411823a443c825f-1178027169.us-east-2.elb.amazonaws.com:8080	1h
redis	default	app: node:redis	172.20.7.241	redis:6379 TCP redis:32846 TCP	elastic-7c0e49c454ab4ed-1286304999.us-east-2.elb.amazonaws.com:6379	1h
final-front-svc	default	-	172.20.171.76	final-front-svc:80 TCP final-front-svc:30050 TCP	elastic-7c0e49c454ab4ed-1286304999.us-east-2.elb.amazonaws.com:80	1h
kubernetes	default	component: apiserver provider: kubernetes	172.20.0.1	kubernetes:443 TCP kubernetes:0 TCP	-	1h

6. Good Sign up mobile UI Interface on mobile phones.

AA Not Secure — a3b7c3fac94f049cfa65cl 🔗

Twitter-clone

Username

Password

Sign In

Remember Writing?

🔍 Follow your interests.

Hear what people are talking about.

Join the conversation.

Username

Pick a username

Email

you@example.com

Password

Create a password

Sign up for Twitter

< > 📄 📖 📄

7. auto scaling

We set the number of minimum pods to be 2 and maximum pods to be 4. When the CPU of a pod is consumed by more than 50 percent, we would use a new pod

```
apiVersion: autoscaling/v1
kind: HorizontalPodAutoscaler
metadata:
  name: front-hpa
spec:
  scaleTargetRef:
    apiVersion: apps/v1beta2
    kind: Deployment
    name: final-front
  minReplicas: 2
  maxReplicas: 4
  targetCPUUtilizationPercentage: 50
```

Demo Time

<http://a3b7c3fac94f049cfa65cb5a42130122-1803852003.us-east-2.elb.amazonaws.com/?webapp=http://a19189527b342477ead75aa685d1c68c-1836042969.us-east-2.elb.amazonaws.com>