

```
sudo lsof -i :8080
```

If you want to access it from your browser outside AWS, you need to tell kubectl port-forward to bind to all interfaces, not just localhost.

```
sudo -E kubectl port-forward --address 0.0.0.0 svc/frontend -n chat-app 8080:80
```

Namespace

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat namespace.yml
kind: Namespace
apiVersion: v1
metadata:
  name: chat-app
```

Frontend deployment & service

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat frontend-deployment.yml
kind: Deployment
apiVersion: apps/v1
metadata:
  name: frontend-deployment
  namespace: chat-app

spec:
  replicas: 1
  selector:
    matchLabels:
      app: frontend
  template:
    metadata:
      name: frontend-pod
      namespace: chat-app
      labels:
        app: frontend
    spec:
      containers:
        - name: chatapp-frontend
          image: suraj0221/node-front:latest
          ports:
            - containerPort: 80
          env:
            - name: NODE_ENV
              value: production
```

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat frontend-service.yml
kind: Service
apiVersion: v1
metadata:
  name: frontend
  namespace: chat-app
spec:
  selector:
    app: frontend
  ports:
    - port: 80
      targetPort: 80
```

Backend deployment & service

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat
backend-deployment.yml
kind: Deployment
apiVersion: apps/v1
metadata:
  name: backended-deployment
  namespace: chat-app
spec:
  replicas: 1
  selector:
    matchLabels:
      app: backend
  template:
    metadata:
      name: backend-pod
      namespace: chat-app
      labels:
        app: backend
    spec:
      containers:
        - name: chatapp-backend
          image: suraj0221/node-backend:latest
          ports:
            - containerPort: 5001
          env:
            - name: NODE_ENV
              value: production
            - name: MONGODB_URI
              value: "mongodb://mongoadmin:secret@mongodb:27017/dbname?
authSource=admin"
            - name: JWT_SECRET
              valueFrom:
                secretKeyRef:
                  name: chatapp-secrets
                  key: jwt
            - name: PORT
              value: "5001"
```

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat
backend-service.yml
kind: Service
apiVersion: v1
metadata:
  name: backend
  namespace: chat-app
spec:
  selector:
    app: backend
  ports:
    - port: 5001
      targetPort: 5001
```

Mongodb deployment & service

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat
mongodb-deployment.yml
kind: Deployment
```

```

apiVersion : apps/v1
metadata:
  name: mongodb-deployment
  namespace: chat-app
spec:
  replicas: 1
  selector:
    matchLabels:
      app: mongodb
  template:
    metadata:
      name: mongodb-pod
      namespace: chat-app
      labels:
        app: mongodb
    spec:
      containers:
        - name: chatapp-mongodb
          image: mongo:latest
          ports:
            - containerPort: 27017
          env:
            - name: MONGO_INITDB_ROOT_USERNAME
              value: mongoadmin
            - name: MONGO_INITDB_ROOT_PASSWORD
              value: secret
      volumes:
        - name: mongo-data
          persistentVolumeClaim:
            claimName: mongodb-pvc

```

```

ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat
mongodb-service.yml
kind: Service
apiVersion: v1
metadata:
  name: mongodb
  namespace: chat-app
spec:
  selector:
    app: mongodb
  ports:
    - port: 27017
      targetPort: 27017

```

Mongodb PV & PVC

```

ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat
mongodb-pv.yml
kind: PersistentVolume
apiVersion: v1
metadata:
  name: mongodb-pv
  namespace: chat-app
spec:
  accessModes:
    - ReadWriteOnce
  capacity:
    storage: 5Gi
  hostPath:
    path: /data

```

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat
mongodb-pvc.yml
kind: PersistentVolumeClaim
apiVersion: v1
metadata:
  name: mongodb-pvc
  namespace: chat-app
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 5Gi
```

secrets

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat
secrets.yml
kind: Secret
apiVersion: v1
metadata:
  name: chatapp-secrets
  namespace: chat-app
type: Opaque
data:
  jwt: iaQ5ibWrZHLuzcjkqiCJbbHi/Ob02XNdCfZbshutZ4A=
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$
```

Ingress

```
ubuntu@ip-172-31-16-22:~/kind-cluster/chat-app/full-stack_chatApp/k8s$ cat
ingress.yml
kind: Ingress
apiVersion: networking.k8s.io/v1
metadata:
  name: chatapp-ingress
  namespace: chat-app
  labels:
    name: chatapp-ing
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
    nginx.ingress.kubernetes.io/ssl-redirect: "false"
spec:
  rules:
    - host: chat-suraj.com
      http:
        paths:
          - pathType: Prefix
            path: "/"
            backend:
              service:
                name: frontend
                port:
                  number: 80
          - pathType: Prefix
            path: "/api"
            backend:
              service:
```

```
name: backend
port:
  number: 5001
```