

# SIT323 Task 6.1P

## Step 1: Build and push docker images

```
docker build -t starwindsurf/backend-image:latest .
docker push starwindsurf/backend-image:latest
docker build -t starwindsurf/frontend-image:latest .
docker push starwindsurf/frontend-image:latest
```

## Step 2: Setup ingress controller

### Download Helm and install ingress

1. Download from here [Releases · helm/helm \(github.com\)](https://github.com/helm/helm/releases)
2. Add the Helm repository of NGINX Ingress Controller

```
helm repo add ingress-nginx https://kubernetes.github.io/ingress-nginx
helm repo update
```

3. Install NGINX Ingress Controller using Helm

```
helm install my-nginx-ingress ingress-nginx/ingress-nginx --set
controller.publishService.enabled=true
```

4. Configure ingress rules

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: my-app-ingress
spec:
  ingressClassName: "nginx"
  rules:
  - host: localhost
    http:
      paths:
      - path: /
        pathType: Prefix
        backend:
          service:
            name: frontend
            port:
              number: 80
      - path: /api
        pathType: Prefix
        backend:
          service:
            name: backend
            port:
              number: 8080
```

## Step 3: Configure front-end and back-end services

---

frontend-deployment.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: frontend-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: frontend
  template:
    metadata:
      labels:
        app: frontend
    spec:
      containers:
        - name: frontend
          image: starwindsurf/frontend-image:latest
          ports:
            - containerPort: 80
```

frontend-service.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: frontend
spec:
  selector:
    app: frontend
  ports:
    - protocol: TCP
      port: 80
      targetPort: 80
```

backend-deployment.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: backend-deployment
spec:
  replicas: 1
  selector:
    matchLabels:
      app: backend
  template:
    metadata:
      labels:
        app: backend
    spec:
```

```
containers:
  - name: backend
    image: starwindsurf/backend-image:latest
    imagePullPolicy: Always
    ports:
      - containerPort: 8080
```

backend-service.yaml

```
apiVersion: v1
kind: Service
metadata:
  name: backend
spec:
  selector:
    app: backend
  ports:
    - protocol: TCP
      port: 8080
      targetPort: 8080
```

## Step 4: Deploy services

---

```
kubectl apply -f backend-deployment.yaml
kubectl apply -f backend-service.yaml
kubectl apply -f frontend-deployment.yaml
kubectl apply -f frontend-service.yaml
kubectl apply -f my-app-ingress.yaml
```