

Compilation Steps:

Initially start the backend server elastic search

- Create a docker container for the postgresql
 - `docker run -d --name postgres_docker -v dbdata:/var/lib/postgresql/data -p 54320:5432 -e POSTGRES_PASSWORD=HaveFun postgres:14` (my pgadmin password is HaveFun)
- Install and deploy minikube package
 - `brew install minikube`(for mac)
 - `minikube start`
 - `minikube status`
 - `eval $(minikube -p minikube docker-env)`
- Create a docker container for the go package module.
 - Create a docker file

```
FROM golang:1.16-alpine
WORKDIR /app
COPY go.mod ./
COPY go.sum ./
RUN go mod download
COPY . /app
RUN go build -o /app/covid-19-daily-stat
EXPOSE 8080
CMD [ "/app/covid-19-daily-stat " ]
```

- Build the module
 - `docker build . -t ms/covid-19-daily-stat`
 - This will fetch the data from Chicago data portal and will be saved in postgres
- Create a Kubernetes cluster for the dockerized modules
 - Create a microservice-fetch-data.yaml file

```
apiVersion: apps/v1

kind: Deployment
metadata:
  name: covid-19-daily-stat
  labels:
    app: covid-19-daily-stat
spec:
  replicas: 1
  selector:
    matchLabels:
      app: covid-19-daily-stat
  template:
    metadata:
      labels:
        app: covid-19-daily-stat
    spec:
      containers:
```

```
- name: app
  image: ms/ covid-19-daily-stat
  imagePullPolicy: Never
```

- Delete any existing kubernetes

```
kubectl delete -f microservice-fetch-data.yaml
```

- deploy the cluster using the following command

```
kubectl create -f microservice-fetch-data.yaml
```

- Check the status and logs from the cluster using the following command

```
kubectl get pods
kubectl logs <pod name>. // Our pod name covid-19-daily-stat-b57899c65-
kj4db
```

- Go to the backend folder and start the backend server using Node.js

In the backend folder we need to give cd node server then its starts my node.js server.

Start the react using npm start in my frontend_react folder

My path: cd

/Users/vidyas/Downloads/IIT_Courses/Fall_2021/CSP_584_EWA/Assignments/
Bonus_Assignment/ChicagoSocialHub_Student_Angular_11/frontend_react

Using the npm start in the folder, start the react server.

ChicagoSocialHub application can run through react server

localhost:3000/find/

Ensure the PostgreSQL server is running, to ensure the application and run all the python files.