Mysql.yaml:

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: mysqldb
  labels:
    app: mysqldb
spec:
  replicas: 1
  selector:
    matchLabels:
      app: mysqldb
  template:
    metadata:
      labels:
        app: mysqldb
    spec:
      containers:
      - name: mysqldb
        image: hitendramhatre/mysql:8.0.30
        ports:
        - containerPort: 3306
        env:
        - name: MYSQL_DATABASE
          value: etour
        - name: MYSQL_PASSWORD
          value: "1234"
        - name: MYSQL_ROOT_PASSWORD
          value: "1234"
---
apiVersion: v1
kind: Service
metadata:
  name: mysqldb-service
spec:
  selector:
    app: mysqldb
  ports:
    - name: mysql
      port: 3306
      targetPort: 3306
---
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: mysqldb-route
spec:
  to:
    kind: Service
    name: mysqldb-service
  port:
    targetPort: mysql
  tls:
    termination: edge
```

spring.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: server
  labels:
    app: server
spec:
  replicas: 1
  selector:
    matchLabels:
      app: server
  template:
    metadata:
      labels:
        app: server
    spec:
      containers:
      - name: server
        image: hitendramhatre/springbootmysql
        ports:
        - containerPort: 8080
        env:
        - name: SPRING_DATASOURCE_URL
          value: jdbc:mysql://mysqldb-service:3306/etour
        - name: SPRING_DATASOURCE_USERNAME
          value: root
        - name: SPRING_DATASOURCE_PASSWORD
        value: "1234"
      - name: SERVER_PORT
        value: "8080"
---
apiVersion: v1
kind: Service
metadata:
  name: server
spec:
  selector:
    app: server
  ports:
  - name: http
    port: 8080
    targetPort: 8080
  type: NodePort
---
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: server-route
spec:
  to:
    kind: Service
    name: server
  port:
    targetPort: http
  tls:
    termination: edge
```

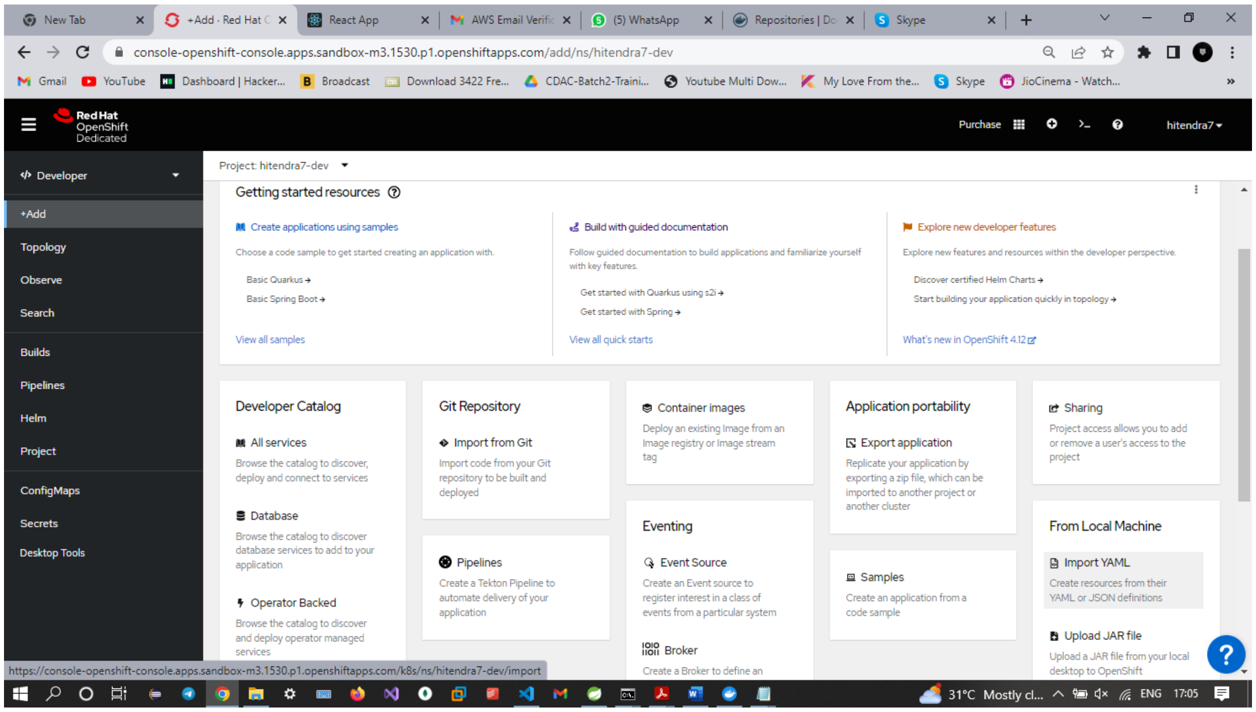
react.yaml:

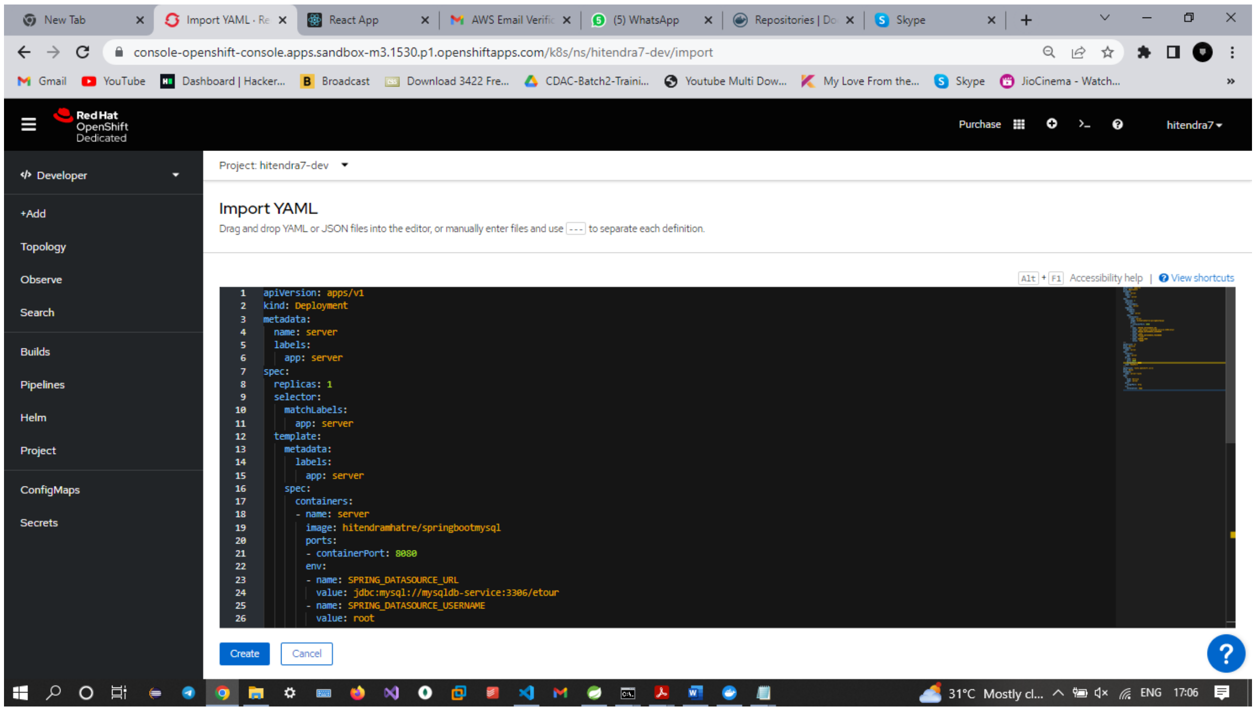
```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: react-app
  labels:
    app: react-app
spec:
  replicas: 1
  selector:
    matchLabels:
      app: react-app
  template:
    metadata:
      labels:
        app: react-app
    spec:
      containers:
      - name: react-app
        image: hitendramhatre/my-react-app2
        ports:
        - containerPort: 3000
---
apiVersion: v1
kind: Service
metadata:
  name: react-app-service
```

```
spec:
  selector:
    app: react-app
  ports:
  - name: http
    port: 80
    targetPort: 3000
  type: NodePort
---
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: react-app-route
spec:
  to:
    kind: Service
    name: react-app-service
  port:
    targetPort: http
  tls:
    termination: edge
```

Process to add .yaml deploment file in readhat

//Add>import YAML

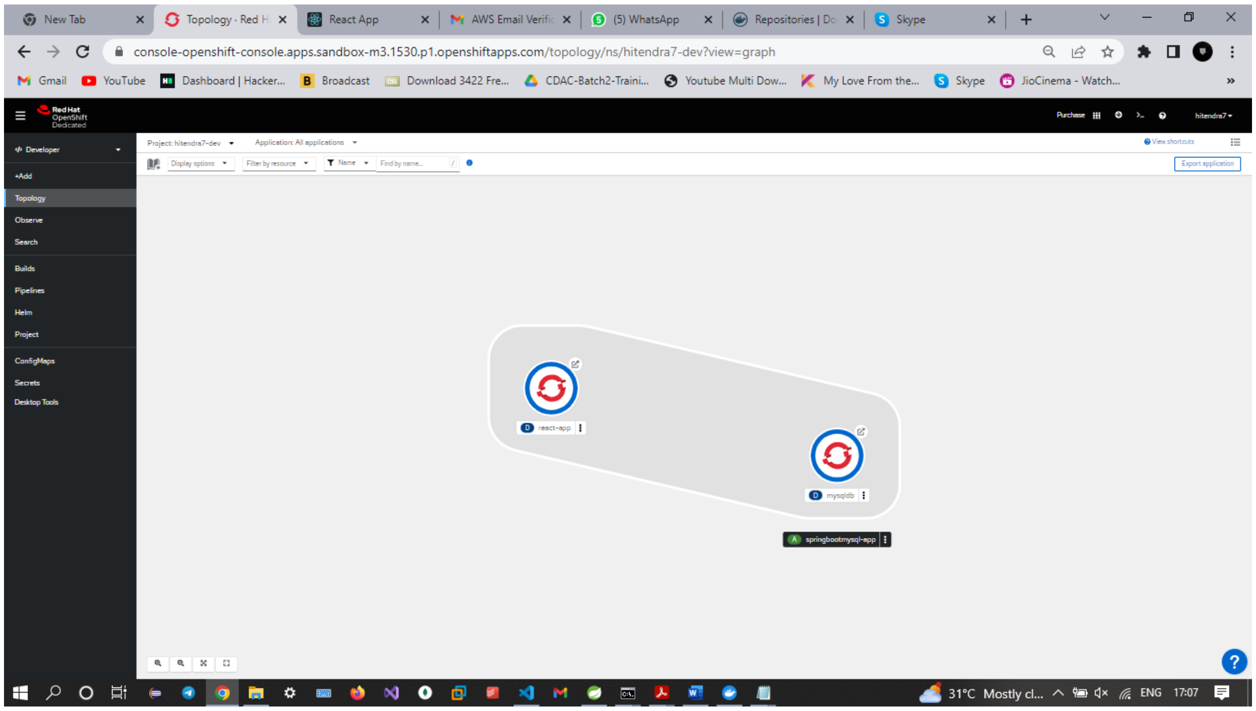

console-openshift-console.apps.sandbox-m3.1530.p1.openshiftapps.com/add/ns/hitendra7-dev
Red Hat OpenShift Dedicated
Purchase
hitendra7
Developer
+Add
Topology
Observe
Search
Builds
Pipelines
Helm
Project
ConfigMaps
Secrets
Desktop Tools
Project: hitendra7-dev
Getting started resources
Create applications using samples
Build with guided documentation
Explore new developer features
Developer Catalog
All services
Database
Operator Backed
Git Repository
Import from Git
Pipelines
Container images
Eventing
Event Source
Broker
Application portability
Export application
Samples
Sharing
From Local Machine
Import YAML
Create resources from their YAML or JSON definitions
Upload JAR file
https://console-openshift-console.apps.sandbox-m3.1530.p1.openshiftapps.com/k8s/ns/hitendra7-dev/import


console-openshift-console.apps.sandbox-m3.1530.p1.openshiftapps.com/k8s/ns/hitendra7-dev/import
Project: hitendra7-dev
Import YAML
Create
Cancel

Drag and drop YAML or JSON files into the editor, or manually enter files and use --- to separate each definition.

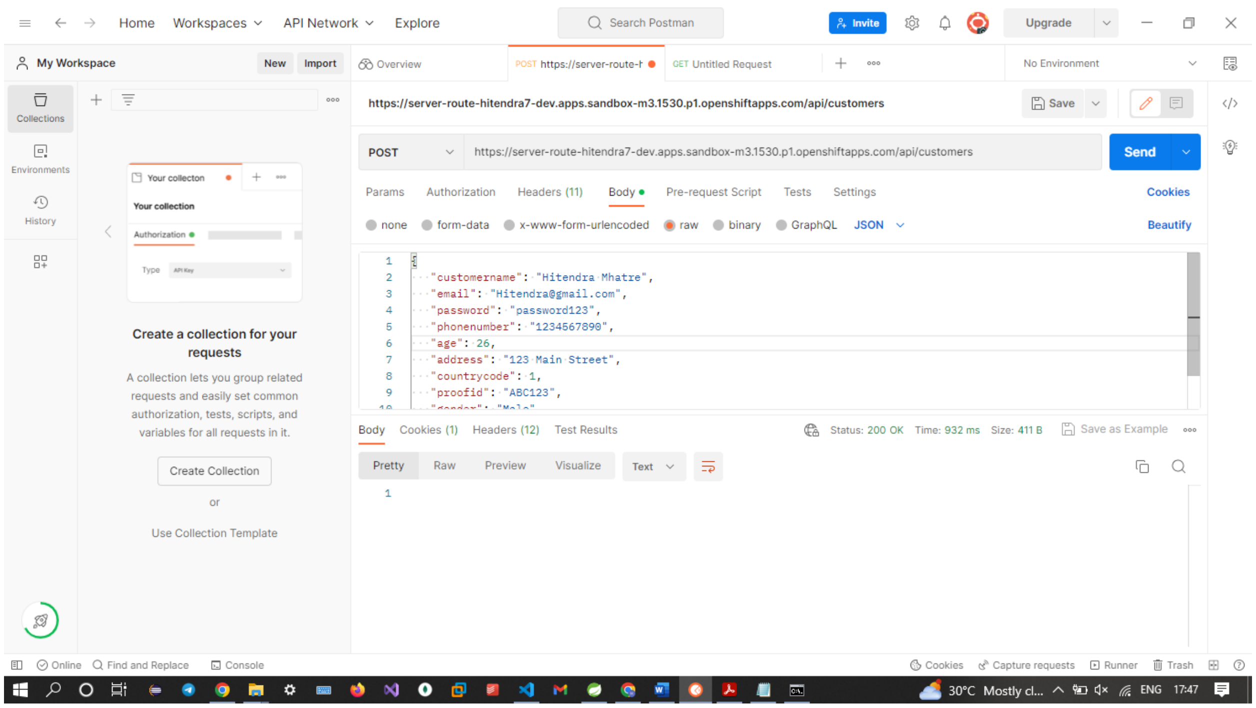
```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: server
  labels:
    app: server
spec:
  replicas: 1
  selector:
    matchLabels:
      app: server
  template:
    metadata:
      labels:
        app: server
    spec:
      containers:
      - name: server
        image: hitendramhatre/springbootmysql
        ports:
        - containerPort: 8080
        env:
        - name: SPRING_DATASOURCE_URL
          value: jdbc:mysql://mysqldb-service:3306/etour
        - name: SPRING_DATASOURCE_USERNAME
          value: root
```

//After importing yaml we will get pods in Toplogy



//To check working of springboot and database we are using postman:

//Registering a user using post



https://server-route-hitendra7-dev.apps.sandbox-m3.1530.p1.openshiftapps.com/api/customers

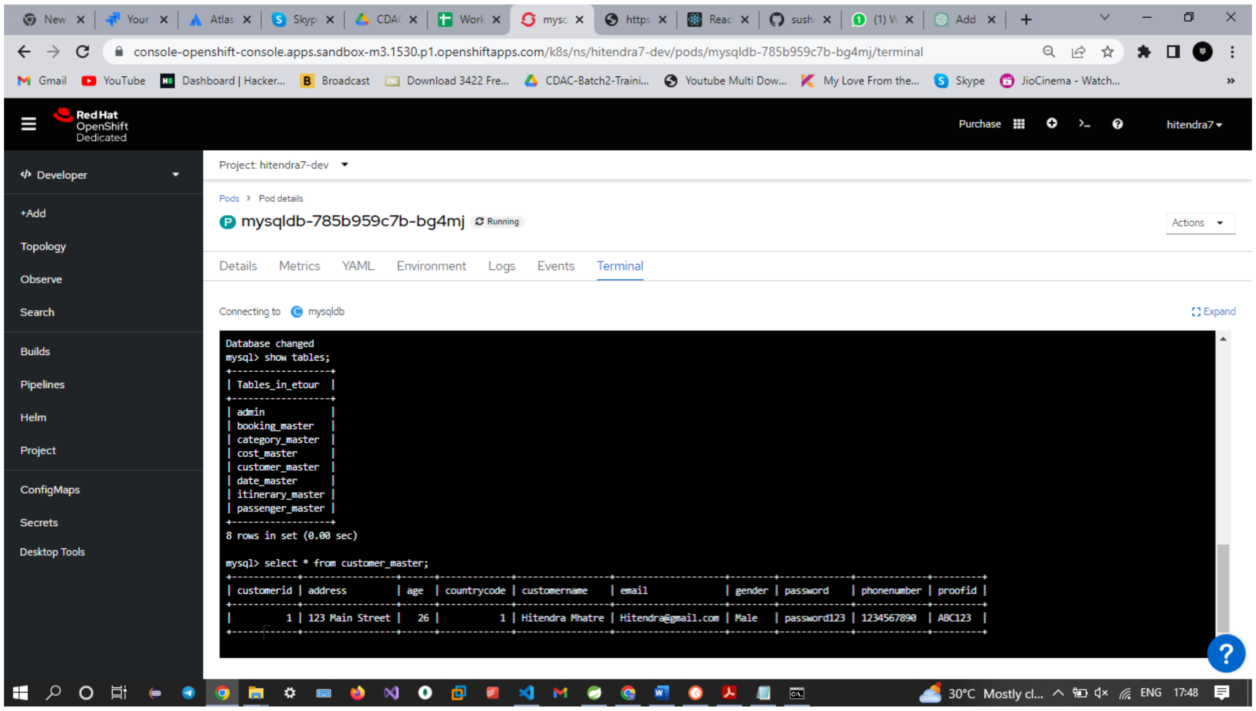
POST https://server-route-hitendra7-dev.apps.sandbox-m3.1530.p1.openshiftapps.com/api/customers

none form-data x-www-form-urlencoded [✓ raw] binary GraphQL JSON

```
{
  "customername": "Hitendra Mhatre",
  "email": "Hitendra@gmail.com",
  "password": "password123",
  "phonenumber": "1234567890",
  "age": 26,
  "address": "123 Main Street",
  "countrycode": 1,
  "proofid": "ABC123",
```

Status: 200 OK Time: 932 ms Size: 411 B

//checking in database that user is added



console-openshift-console.apps.sandbox-m3.1530.p1.openshiftapps.com/k8s/ns/hitendra7-dev/pods/mysqldb-785b959c7b-bg4mj/terminal

Project: hitendra7-dev

Pods > Pod details

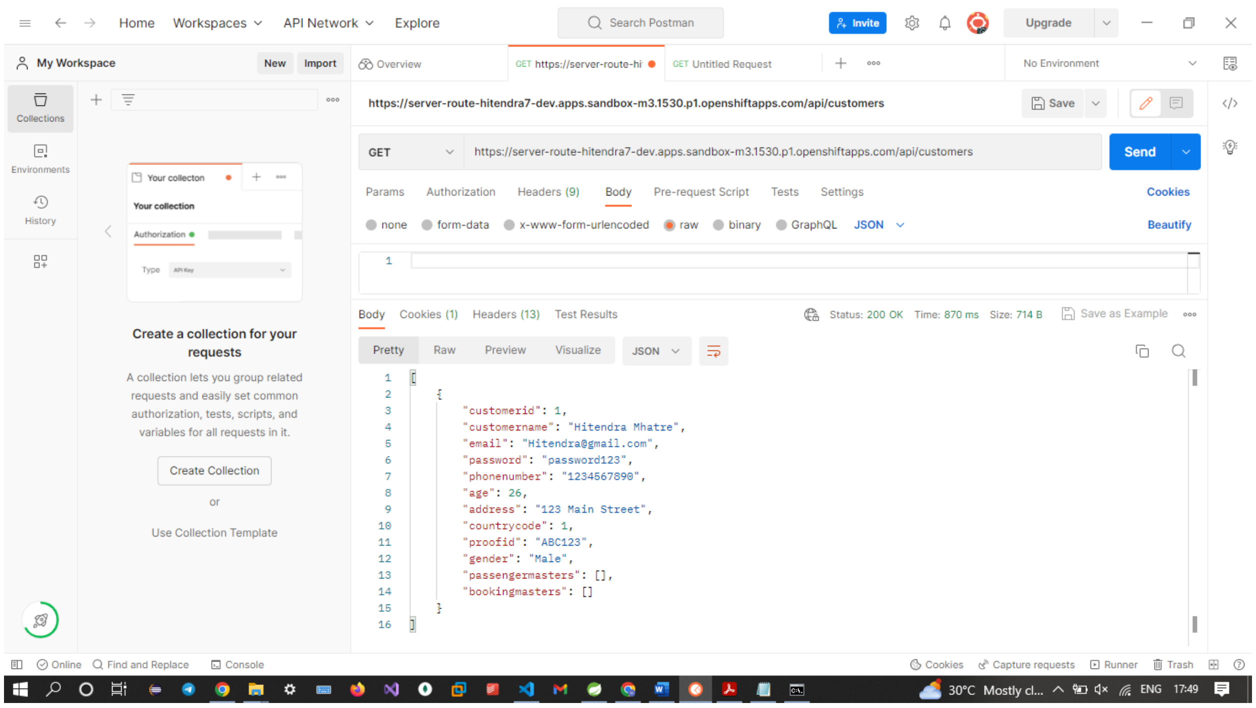
mysqldb-785b959c7b-bg4mj Running

Connecting to mysqldb

```
Database changed
mysql> show tables;
+------------------+
| Tables_in_etour  |
+------------------+
| admin            |
| booking_master   |
| category_master  |
| cost_master      |
| customer_master  |
| date_master      |
| itinerary_master |
| passenger_master |
+------------------+
8 rows in set (0.00 sec)

mysql> select * from customer_master;
+------------+-----------------+-----+-------------+-----------------+--------------------+--------+-------------+-------------+---------+
| customerid | address         | age | countrycode | customername    | email              | gender | password    | phonenumber | proofid |
+------------+-----------------+-----+-------------+-----------------+--------------------+--------+-------------+-------------+---------+
|          1 | 123 Main Street |  26 |           1 | Hitendra Mhatre | Hitendra@gmail.com | Male   | password123 | 1234567890  | ABC123  |
+------------+-----------------+-----+-------------+-----------------+--------------------+--------+-------------+-------------+---------+
```

//getting the user list by using get commands

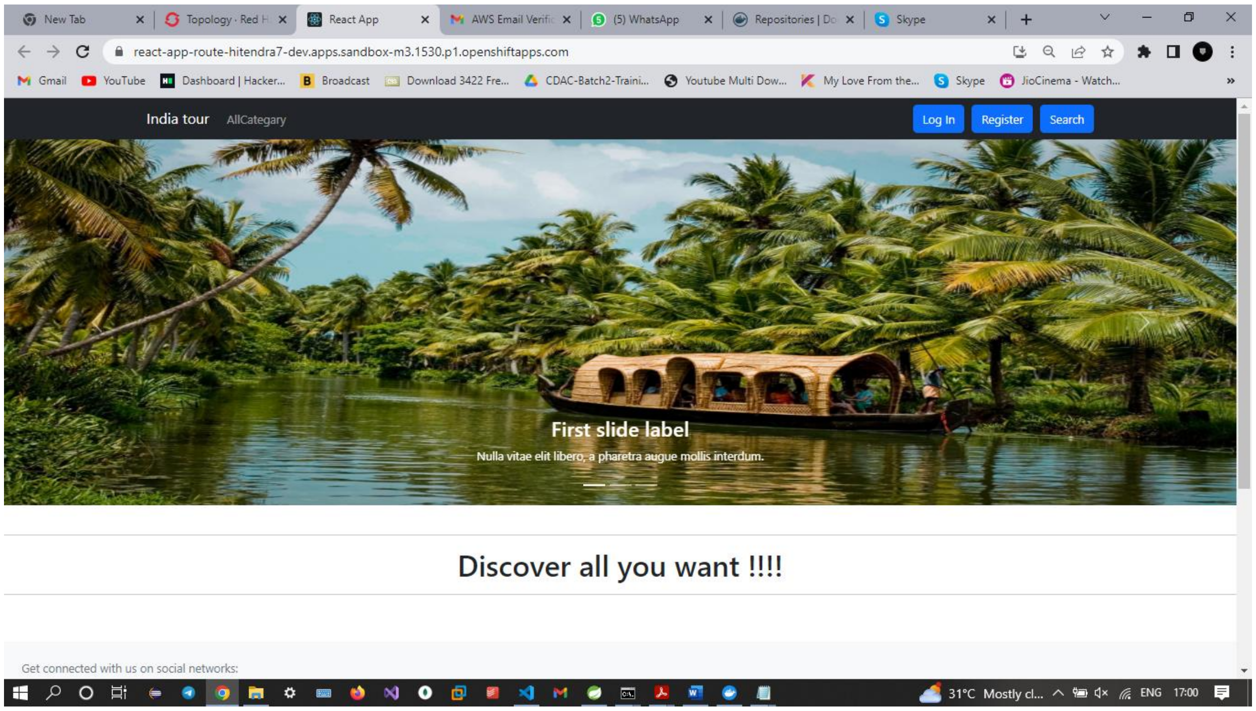


GET https://server-route-hitendra7-dev.apps.sandbox-m3.1530.p1.openshiftapps.com/api/customers

Status: 200 OK Time: 870 ms Size: 714 B

```
[
    {
        "customerid": 1,
        "customername": "Hitendra Mhatre",
        "email": "Hitendra@gmail.com",
        "password": "password123",
        "phonenumber": "1234567890",
        "age": 26,
        "address": "123 Main Street",
        "countrycode": 1,
        "proofid": "ABC123",
        "gender": "Male",
        "passengermasters": [],
        "bookingmasters": []
    }
]
```

//React app is working and can be assessed by provided route by openshift:


react-app-route-hitendra7-dev.apps.sandbox-m3.1530.p1.openshiftapps.com
India tour
AllCategary
Log In
Register
Search
First slide label
Nulla vitae elit libero, a pharetra augue mollis interdum.
Discover all you want !!!!
Get connected with us on social networks: