**High level Architecture Design Diagram for the Location based Taxi Aggregator and Selector (Cloud)**

A diagram of a computer

Description automatically generated

**Demo Presentation:**

1. Delete existing DB if present

A screenshot of a computer

Description automatically generated

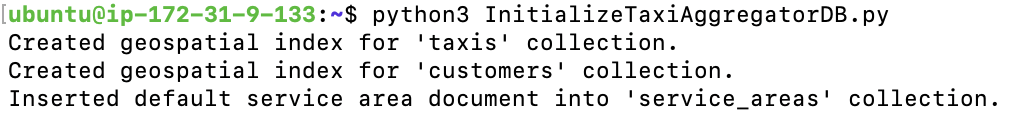
1. Connect to Ec2 instance using below command from terminal

ssh -i "ec2cp2.pem" [ubuntu@ec2-3-143-22-60.us-east-2.compute.amazonaws.com](mailto:ubuntu@ec2-3-143-22-60.us-east-2.compute.amazonaws.com)

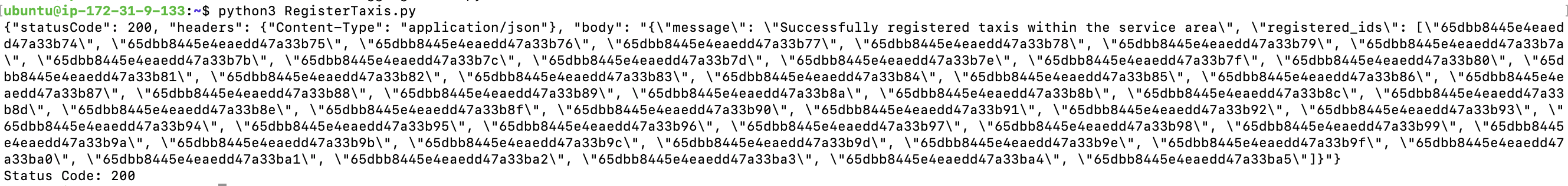
A screenshot of a computer error

Description automatically generated

1. Run InitializeTaxiAggregatorDB.py to create DB, Collections and 2dsphere index and then insert default service area location.



1. Run RegisterTaxis.py to create 50 sample taxi data.



A screenshot of a computer

Description automatically generated

1. Run RegisterCustomers.py to create 5 sample Customer data.

A screenshot of a computer code

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Run TaxiLocationUpdateSimulator.py to send location data for different taxis at regular interval.

A screenshot of a computer code

Description automatically generated

A screenshot of a computer code

Description automatically generated

1. Post below request in postman to find rides based on customer location proximity search. API - <https://udmqfurfie.execute-api.us-east-2.amazonaws.com/dev/ride/find>

{

"location": {

"type": "Point",

"coordinates": [77.23149, 28.61123]

},

"type": "Luxury"

}

A screenshot of a computer

Description automatically generated

1. Post below json request to create a taxi data using API - <https://udmqfurfie.execute-api.us-east-2.amazonaws.com/dev/taxi>

{

"name": "Taxi\_052",

"type": "Luxury",

"location": {

"type": "Point",

"coordinates": [

78.3368142118244,

27.66077927262116

]

}

}

A screenshot of a computer

Description automatically generated

{"name": "Taxi\_052"} find result:

A screenshot of a web page

Description automatically generated

9. Post below json request to create a customer data using API - <https://udmqfurfie.execute-api.us-east-2.amazonaws.com/dev/customer>

{

"name": "Manju",

"location": {

"type": "Point",

"coordinates": [

77.24098,

28.65454

]

}

}

A screenshot of a computer

Description automatically generated

{"name": "Manju"} find result:

A screenshot of a computer

Description automatically generated

1. Service area validation for customer. Post below request to API (<https://udmqfurfie.execute-api.us-east-2.amazonaws.com/dev/customer> ) to test this.

{

"name": "Manju",

"location": {

"type": "Point",

"coordinates": [

87.24098,

28.65454

]

}

}

A screenshot of a computer

Description automatically generated

1. Service area validation for taxi. Post below request to API - https://udmqfurfie.execute-api.us-east-2.amazonaws.com/dev/taxi to test this.

{

"name": "Taxi\_052",

"type": "Luxury",

"location": {

"type": "Point",

"coordinates": [

78.3368142118244,

37.66077927262116

]

}

}

A screenshot of a computer

Description automatically generated

**EC2 instance:**

A screenshot of a computer

Description automatically generated

**API Gateway:**

**Lambda Functions:**

**Taxi Registration Lambda:**

**A screenshot of a computer

Description automatically generated**

**Customer/User registration Lambda:**

**A screenshot of a computer

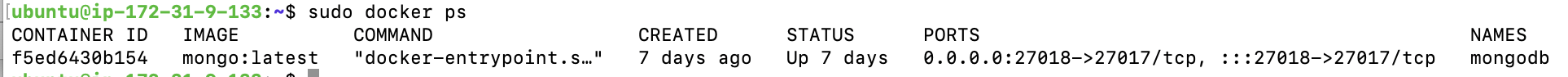
Description automatically generated**

**Ride Find Lambda:**

**A screenshot of a computer

Description automatically generated**

**Docker instance running in EC2:**

****