**PROJECT TITLE:** **BUILD A PLASMA DONOR APP WITH AWS SERVERLESS COMPUTING**

## INTRODUCTION:

To build a serverless computation using AWS for plasma donation.

## OVERVIEW:

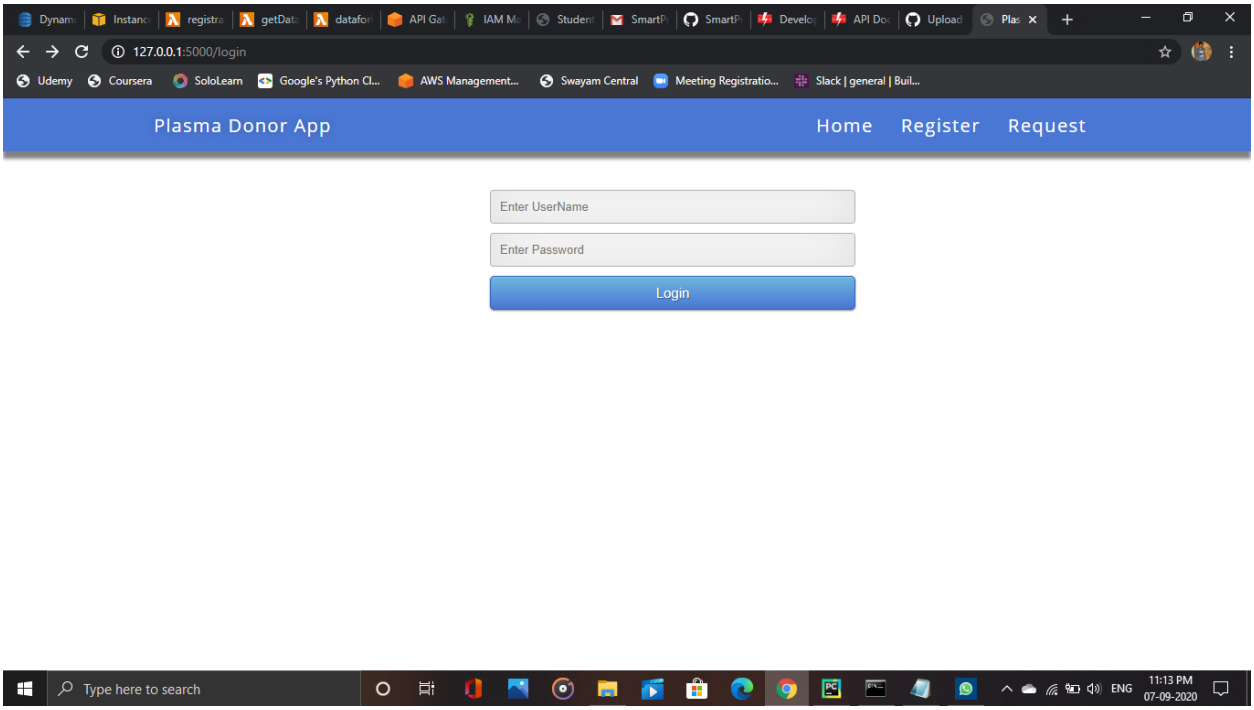
We use Dynamo DB, Lambda, IAM, API Gateway, EC2 in AWS and fast2sms to send the messages. We create tables using Dynamo DB this is embedded with Lambda. In lambda we can write the code to insert the data in the table. The IAM is used for security purposes, identity management. It has set of permission for making AWS service request.The API gateway is between the lambda and user interface, it generates the url which is the frontend. EC2 instance has a resizable compute capacity, it can develop and deploy applications faster.

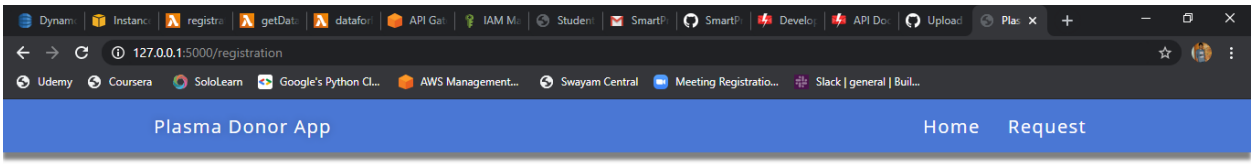
## PURPOSE:

To intimate the person through messages.

## RESULT:

### SCREENSHOTS OF OUTPUT






Enter Your Name

Enter Email

Enter 10-digit mobile number

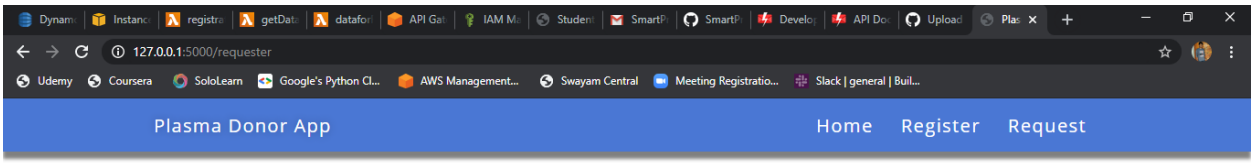
Enter Your City Name

Select COVID infection status

Choose your blood group

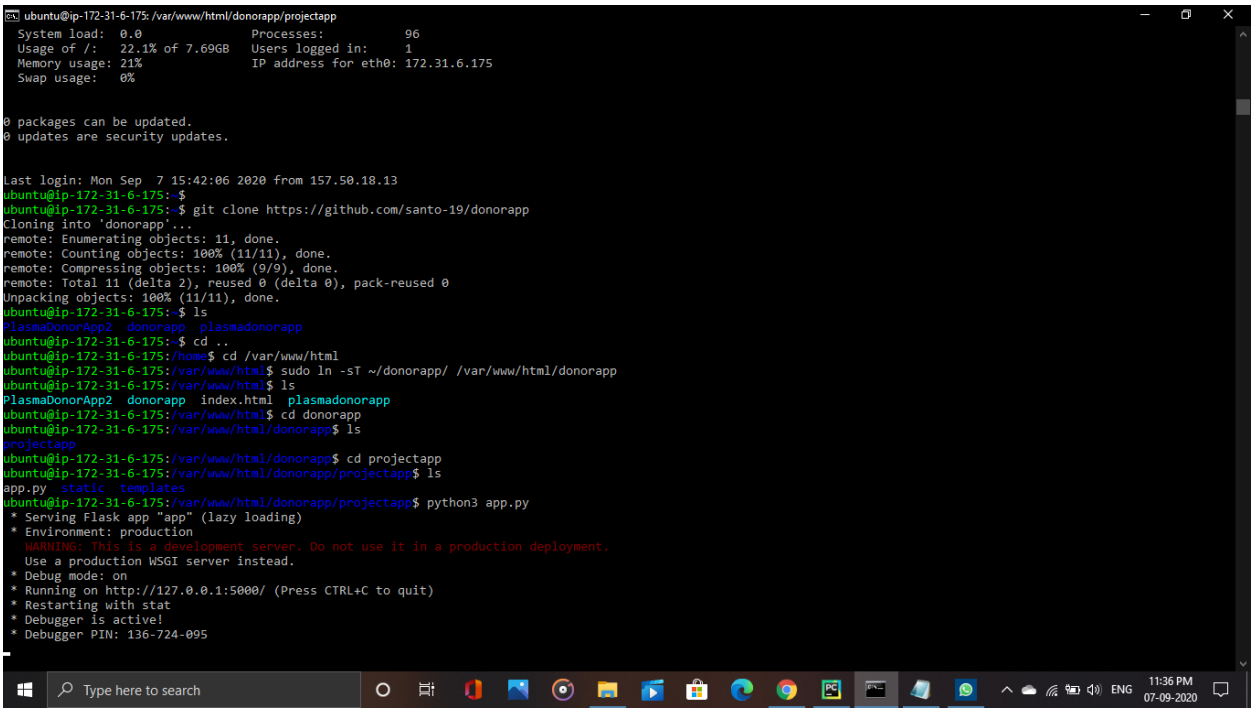
Enter Password

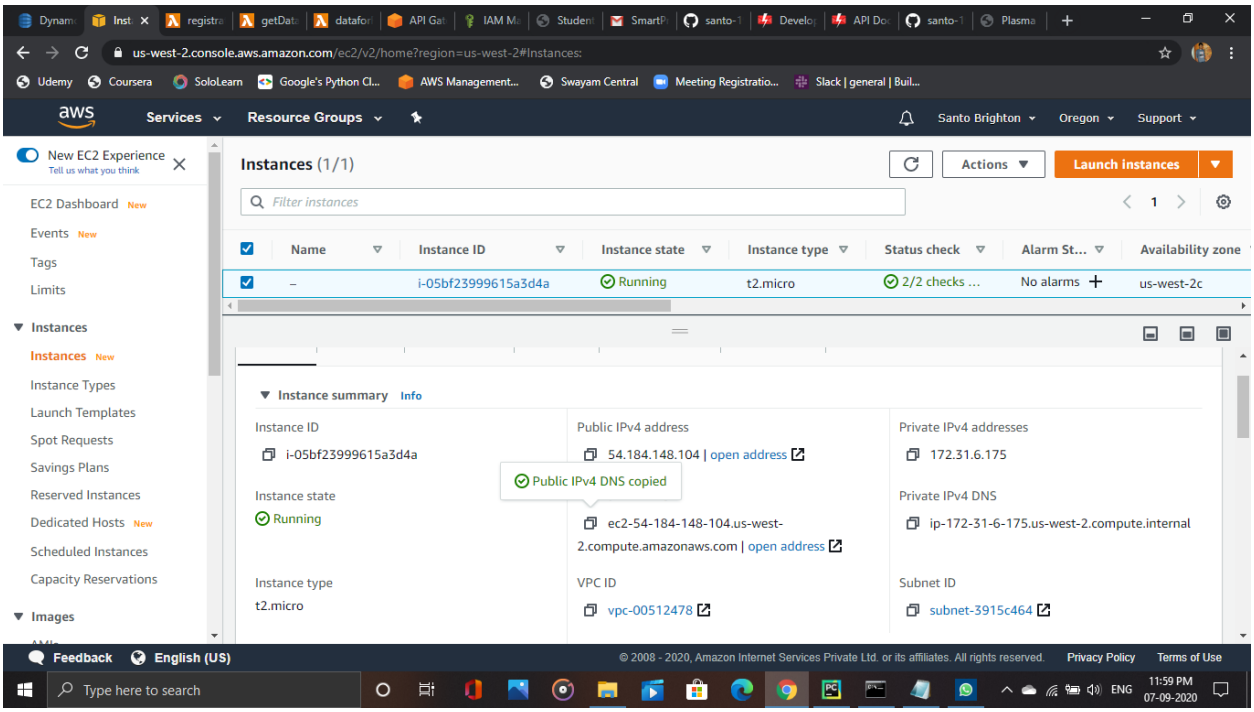
Register

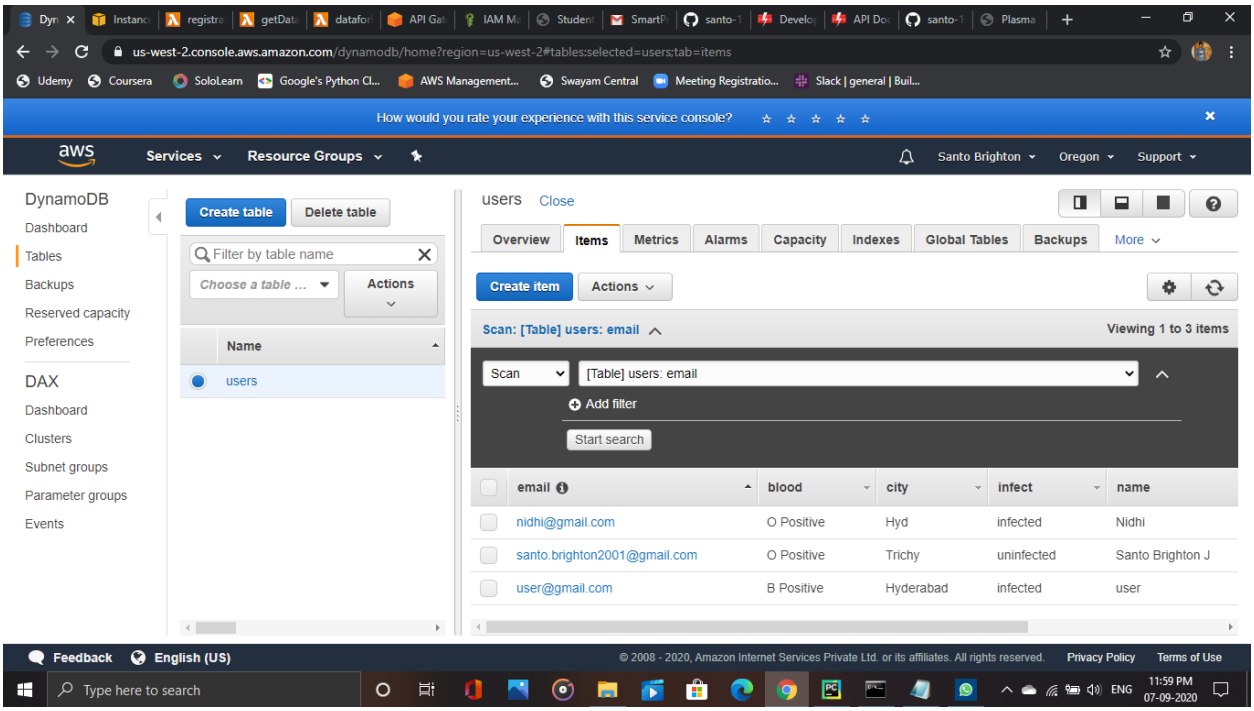



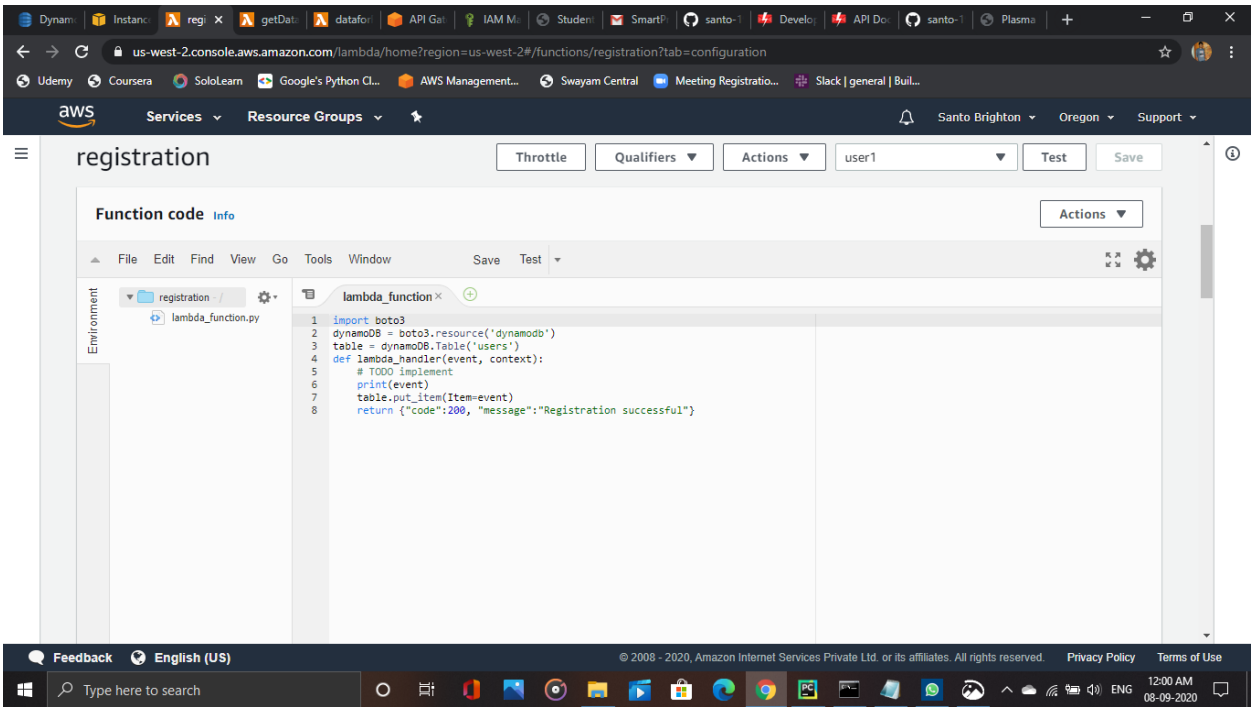

Choose your blood group

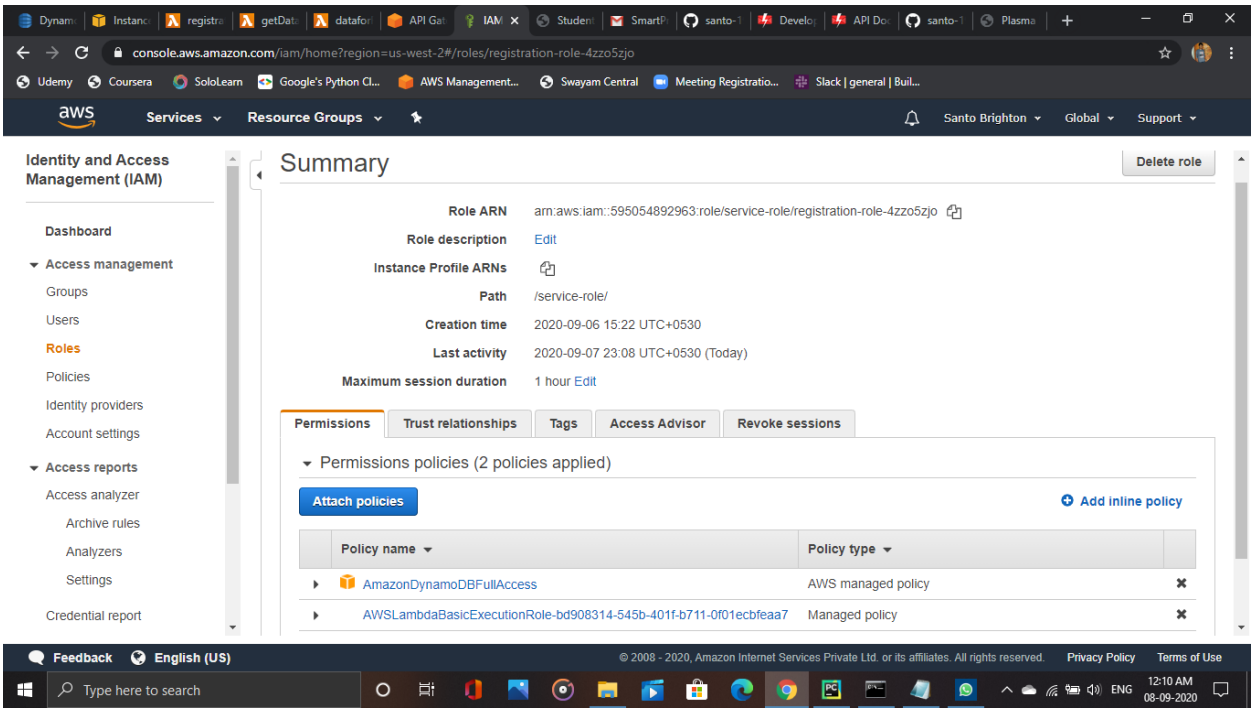
Submit the request

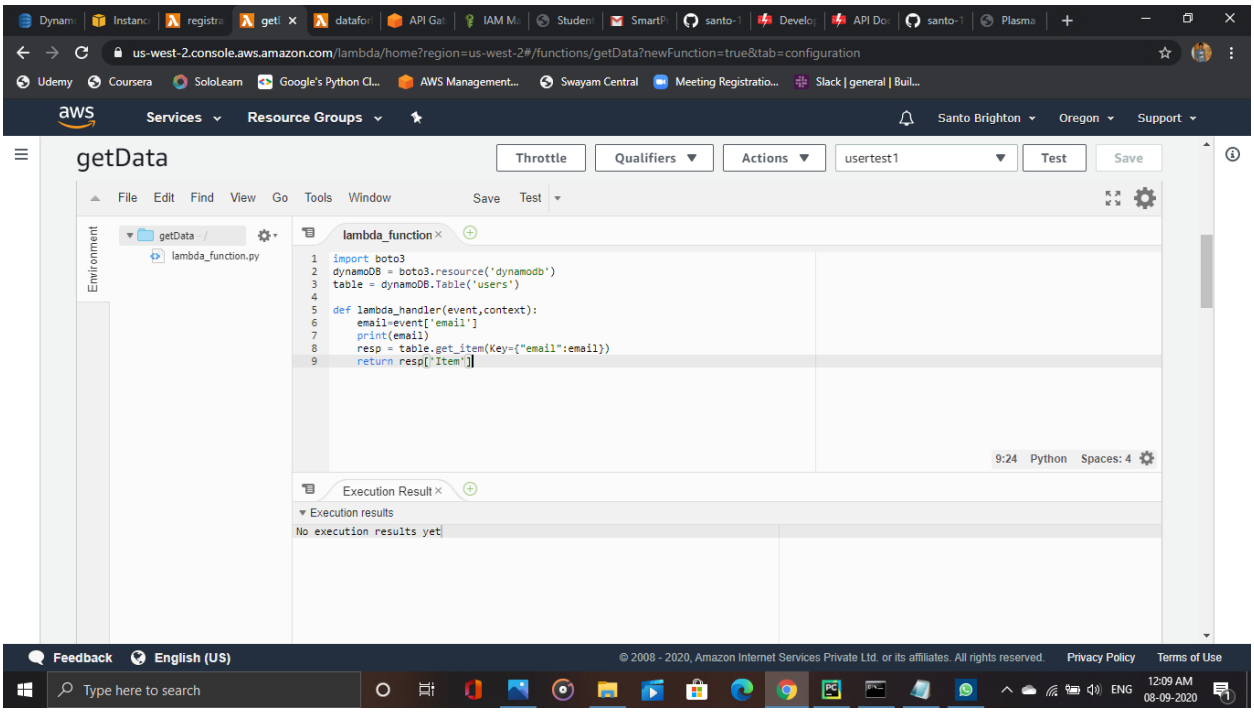

ubuntu@ip-172-31-6-175: /var/www/html/donorapp/projectapp
System load: 0.0 Processes: 96
Usage of /: 22.1% of 7.69GB Users logged in: 1
Memory usage: 21% IP address for eth0: 172.31.6.175
Swap usage: 0%
0 packages can be updated.
0 updates are security updates.
Last login: Mon Sep 7 15:42:06 2020 from 157.50.18.13
ubuntu@ip-172-31-6-175:~$
ubuntu@ip-172-31-6-175:~$ git clone https://github.com/santo-19/donorapp
Cloning into 'donorapp'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 11 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (11/11), done.
ubuntu@ip-172-31-6-175:~$ ls
PlasmaDonorApp2 donorapp plasmadonorapp
ubuntu@ip-172-31-6-175:~$ cd ..
ubuntu@ip-172-31-6-175:/home$ cd /var/www/html
ubuntu@ip-172-31-6-175:/var/www/html$ sudo ln -sT ~/donorapp/ /var/www/html/donorapp
ubuntu@ip-172-31-6-175:/var/www/html$ ls
PlasmaDonorApp2 donorapp index.html plasmadonorapp
ubuntu@ip-172-31-6-175:/var/www/html$ cd donorapp
ubuntu@ip-172-31-6-175:/var/www/html/donorapp$ ls
projectapp
ubuntu@ip-172-31-6-175:/var/www/html/donorapp$ cd projectapp
ubuntu@ip-172-31-6-175:/var/www/html/donorapp/projectapp$ ls
app.py static templates
ubuntu@ip-172-31-6-175:/var/www/html/donorapp/projectapp$ python3 app.py
* Serving Flask app "app" (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: on
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 136-724-095
Type here to search
ENG
11:36 PM
07-09-2020

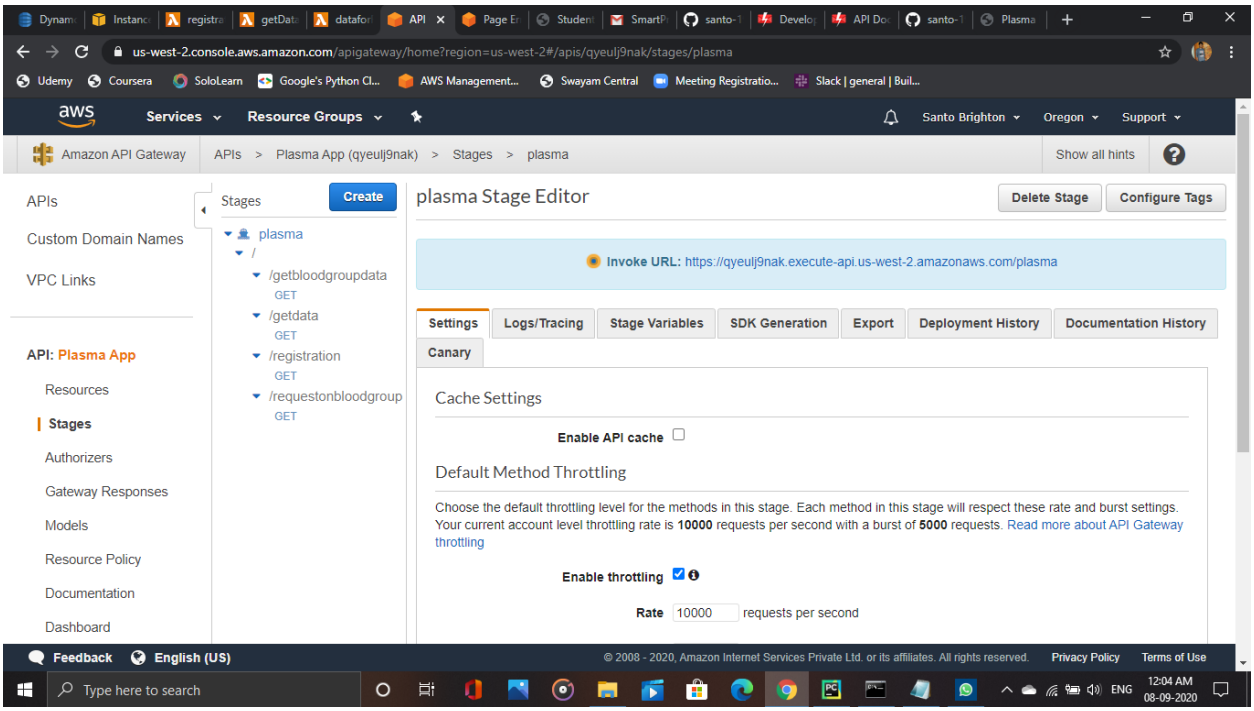

us-west-2.console.aws.amazon.com/ec2/v2/home?region=us-west-2#Instances:
Udemy
Coursera
SoloLearn
Google's Python Cl...
AWS Management...
Swayam Central
Meeting Registratio...
Slack | general | Buil...
aws
Services
Resource Groups
Santo Brighton
Oregon
Support
New EC2 Experience
Tell us what you think
EC2 Dashboard New
Events New
Tags
Limits
Instances
Instances New
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts New
Scheduled Instances
Capacity Reservations
Images
Instances (1/1)
Actions
Launch instances
Filter instances
Name
Instance ID
Instance state
Instance type
Status check
Alarm St...
Availability zone
i-05bf23999615a3d4a
Running
t2.micro
2/2 checks ...
No alarms
us-west-2c
Instance summary Info
Instance ID
i-05bf23999615a3d4a
Instance state
Running
Instance type
t2.micro
Public IPv4 address
54.184.148.104 | open address
Public IPv4 DNS copied
ec2-54-184-148-104.us-west-2.compute.amazonaws.com | open address
VPC ID
vpc-00512478
Private IPv4 addresses
172.31.6.175
Private IPv4 DNS
ip-172-31-6-175.us-west-2.compute.internal
Subnet ID
subnet-3915c464
Feedback
English (US)
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Type here to search
ENG
11:59 PM
07-09-2020

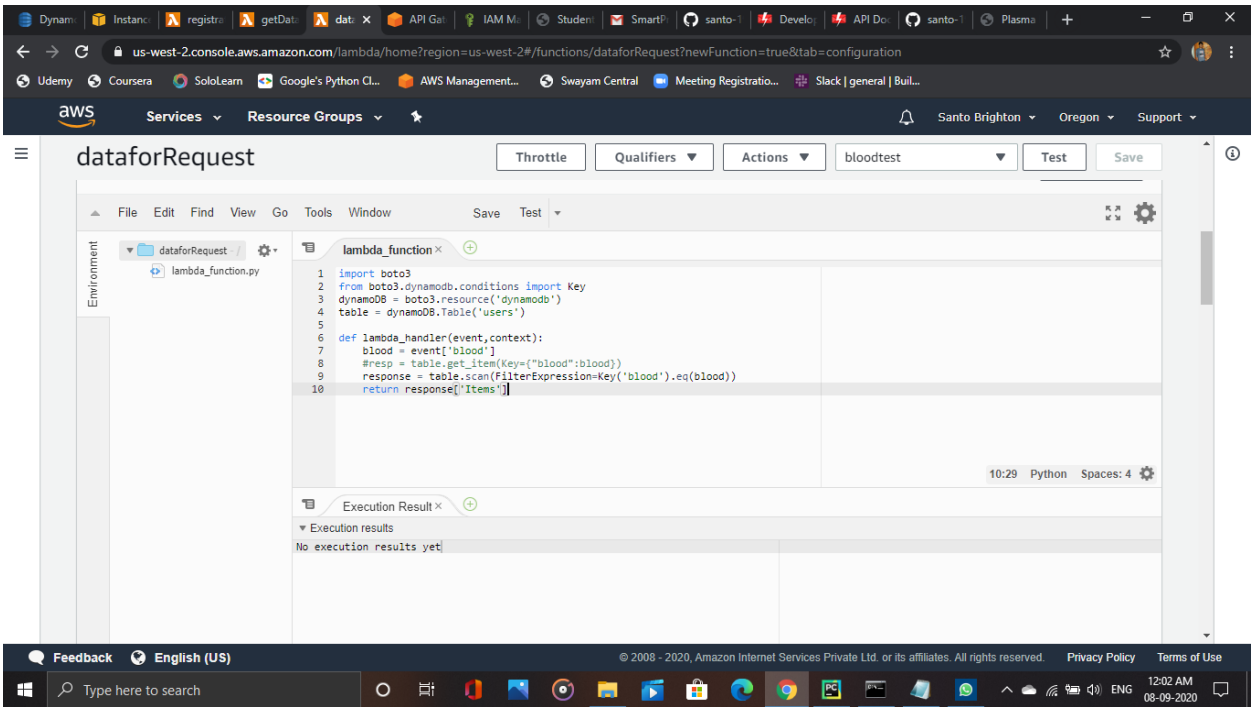

us-west-2.console.aws.amazon.com/dynamodb/home?region=us-west-2#tables:selected=users;tab=items
How would you rate your experience with this service console?
Services
Resource Groups
Santo Brighton
Oregon
Support
DynamoDB
Dashboard
Tables
Backups
Reserved capacity
Preferences
DAX
Dashboard
Clusters
Subnet groups
Parameter groups
Events
Create table
Delete table
Filter by table name
Choose a table ...
Actions
Name
users
users
Close
Overview
Items
Metrics
Alarms
Capacity
Indexes
Global Tables
Backups
More
Create item
Actions
Scan: [Table] users: email
Viewing 1 to 3 items
Scan
[Table] users: email
Add filter
Start search
email
blood
city
infect
name
nidhi@gmail.com
O Positive
Hyd
infected
Nidhi
santo.brighton2001@gmail.com
O Positive
Trichy
uninfected
Santo Brighton J
user@gmail.com
B Positive
Hyderabad
infected
user
Feedback
English (US)
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Type here to search
ENG
11:59 PM
07-09-2020


us-west-2.console.aws.amazon.com/lambda/home?region=us-west-2#/functions/registration?tab=configuration
Services
Resource Groups
Santo Brighton
Oregon
Support
registration
Throttle
Qualifiers
Actions
user1
Test
Save
Function code
Info
Actions
File
Edit
Find
View
Go
Tools
Window
Save
Test
Environment
registration
lambda_function.py
lambda_function
import boto3
dynamoDB = boto3.resource('dynamodb')
table = dynamoDB.Table('users')
def lambda_handler(event, context):
# TODO implement
print(event)
table.put_item(Item=event)
return {"code":200, "message":"Registration successful"}
Feedback
English (US)
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Type here to search
ENG
12:00 AM
08-09-2020


console.aws.amazon.com/iam/home?region=us-west-2#/roles/registration-role-4zzo5zjo
Services
Resource Groups
Santo Brighton
Global
Support
Identity and Access Management (IAM)
Dashboard
Access management
Groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Summary
Delete role
Role ARN
arn:aws:iam::595054892963:role/service-role/registration-role-4zzo5zjo
Role description
Edit
Instance Profile ARNs
Path
/service-role/
Creation time
2020-09-06 15:22 UTC+0530
Last activity
2020-09-07 23:08 UTC+0530 (Today)
Maximum session duration
1 hour Edit
Permissions
Trust relationships
Tags
Access Advisor
Revoke sessions
Permissions policies (2 policies applied)
Attach policies
Add inline policy
Policy name
Policy type
AmazonDynamoDBFullAccess
AWS managed policy
AWSLambdaBasicExecutionRole-bd908314-545b-401f-b711-0f01ecbfeaa7
Managed policy
Feedback
English (US)
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Type here to search
ENG
12:10 AM
08-09-2020


us-west-2.console.aws.amazon.com/lambda/home?region=us-west-2#/functions/getData?newFunction=true&tab=configuration
Services
Resource Groups
Santo Brighton
Oregon
Support
getData
Throttle
Qualifiers
Actions
usertest1
Test
Save
File
Edit
Find
View
Go
Tools
Window
Save
Test
Environment
getData
lambda_function.py
lambda_function
import boto3
dynamoDB = boto3.resource('dynamodb')
table = dynamoDB.Table('users')
def lambda_handler(event,context):
email=event['email']
print(email)
resp = table.get_item(Key={"email":email})
return resp['Item']
9:24
Python
Spaces: 4
Execution Result
Execution results
No execution results yet
Feedback
English (US)
© 2008 - 2020, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use
Type here to search
ENG
12:09 AM
08-09-2020




## APPLICATIONS:

This can be used to send the need for plasma messages directly to the volunteers through phone number which can cure the infected COVID-19 patients regardless of the time.

**CONCLUSION:**

Thus we created a serverless computation using AWS for plasma donation.

**FUTURE SCOPE:**

We can easily access to add, update and delete the data in future since its a serverless computing.