client-management-system-using-php-mysql

Submitted by

Chennaboina Naveen

About Project

Client Management System in PHP MySQL is a simple web application develop in PHP MySQL Database, Using HTML, CSS JavaScript, Bootstrap, Modal and Ajax, the system contains of adding clients, services, invoices, and reports. It is simple to understand and friendly user, the admin can manage all data on the system, the clients is only can manage by own, functioning to cater all services to provide to clients.

- **Name of Project:** Client Management System Project
- **Language**: PHP
- **Databases used**: MySQL
- **Design used**: HTML JavaScript, Ajax, JQuery, Bootstrap
- **Browser used:** IE8, Google Chrome, Opera Mozilla
- **Software used:** WAMP/ XAMPP/ LAMP/MAMP

1.Introduction to project

In this project, we use PHP and MYSQL database.

It has two module

1. Admin
2. User(clients)

Admin is the superuser of the website who can manage everything on the website. Admin can log in through the login page

3 code formation and Process

Using some smile attributes we design the code for the creation of RDS, Auto Scaling group, load balancer, Ec2 server Machine, cloud matrix for alarms

```
AWSTemplateFormatVersion:
 2010-09-09
                        Metadata:
                          'AWS::CloudFormation::Designer':
                            5a49fac4-991e-4070-916c-5e2a0b1311bd:
                              size:
                                width: 60
                                height: 60
                              position:
                                x: 198
                                'y': 99
                              z: 0
                              embeds: []
                            fef8a509-ced8-430b-81d1-a353eb3cbb59:
                              size:
                                width: 60
                                height: 60
                              position:
                                x: 315
                                'y': 97
```

```
      z: 0
      embeds: []
    b943b41c-78a1-4c80-b3ab-ea3c57fca06f:
      size:
        width: 60
        height: 60
      position:
        x: 461
        'y': 104
      z: 0
      embeds: []
    f135fd8f-ac4f-48ab-8ad3-5625d0c5d2ad:
      size:
        width: 60
        height: 60
      position:
        x: 586
        'y': 105
      z: 0
      embeds: []
      dependson:
        - fef8a509-ced8-430b-81d1-a353eb3cbb59
        - b943b41c-78a1-4c80-b3ab-ea3c57fca06f
    355e45e0-9d36-49a7-8d8d-fc3b7f760b9c:
      size:
        width: 60
        height: 60
      position:
        x: 708
        'y': 99
      z: 0
      embeds: []
    1453d6ad-0801-4957-a2f6-c9cc5887b06b:
      size:
        width: 60
        height: 60
      position:
        x: 843
        'y': 100
      z: 0
      embeds: []
      dependson:
        - b943b41c-78a1-4c80-b3ab-ea3c57fca06f
Resources:
  RDSDBI32QQE:
```

```yaml
    Type: 'AWS::RDS::DBInstance'
    Properties:
      AllocatedStorage: 10
      Engine: MySQL
      MasterUsername: admin
      MasterUserPassword: Naveen1999
      DBInstanceClass: db.t2.micro
      DBName: clientmsdb
      DBInstanceIdentifier: rds24
      VPCSecurityGroups:
        - sg-0b947349537e69ed2
      EngineVersion: 5.7
      PubliclyAccessible: true
    Metadata:
      'AWS::CloudFormation::Designer':
        id: 5a49fac4-991e-4070-916c-5e2a0b1311bd
  ELBV2LB1HUDC:
    Type:
'AWS::ElasticLoadBalancingV2::LoadBalancer'
    Properties:
      Name: my-load-balancer-new1
      Subnets:
        - subnet-010656811f3381d92
        - subnet-026d1c34b85e10605
      IpAddressType: ipv4
    Metadata:
      'AWS::CloudFormation::Designer':
        id: fef8a509-ced8-430b-81d1-a353eb3cbb59
  ELBV2TG4A6L0:
    Type: 'AWS::ElasticLoadBalancingV2::TargetGroup'
    Properties:
      Name: my-target-group-new1
      Port: 80
      Protocol: HTTP
      VpcId: vpc-01ac3cdf74d6afae6
    Metadata:
      'AWS::CloudFormation::Designer':
        id: b943b41c-78a1-4c80-b3ab-ea3c57fca06f
  ELBV2L42QE:
    Type: 'AWS::ElasticLoadBalancingV2::Listener'
    Properties:
      LoadBalancerArn: !Ref ELBV2LB1HUDC
      Port: 80
      Protocol: HTTP
      DefaultActions:
```

```yaml
        - Type: forward
          TargetGroupArn: !Ref ELBV2TG4A6L0
    Metadata:
      'AWS::CloudFormation::Designer':
        id: f135fd8f-ac4f-48ab-8ad3-5625d0c5d2ad
    DependsOn:
      - ELBV2LB1HUDC
      - ELBV2TG4A6L0
  ASLC55LJA:
    Type: 'AWS::AutoScaling::LaunchConfiguration'
    Properties:
      LaunchConfigurationName: my-lc-new1
      ImageId: ami-04505e74c0741db8d
      InstanceType: t2.micro
      IamInstanceProfile: demo-Role
      KeyName: Naveen06training
      SecurityGroups:
        - sg-0b947349537e69ed2
      UserData:
        'Fn::Base64': !Sub
        - |
          #!/bin/bash
          sudo apt-get update -y
          sudo apt-get upgrade -y
          sudo wget
https://www.apachefriends.org/xampp-
files/8.1.2/xampp-linux-x64-8.1.2-0-installer.run
          sudo chmod 755 xampp-linux-x64-8.1.2-0-
installer.run
          sudo ./xampp-linux-x64-8.1.2-0-
installer.run
          Y
          Y
          ENTER
          Y
          sudo apt install net-tools
          sudo /opt/lampp/lampp start
          sudo rm -rf /opt/lampp/htdocs/*
          sudo chmod 777 /opt/lampp/htdocs
          sudo git clone
https://github.com/naveen101999/aws.git
          sudo cp -r aws/cms/clientms/*
/opt/lampp/htdocs
          sudo chmod 777
/opt/lampp/htdocs/includes/dbconnection.php
```

```
        sudo sed -i.bak
's/localhost/${endpoint}/g'
/opt/lampp/htdocs/includes/dbconnection.php
        sudo sed -i.bak
's/localhost/${endpoint}/g'
/opt/lampp/htdocs/admin/includes/dbconnection.php
        sudo chmod 777 /opt/lampp/etc/extra/httpd-
xampp.conf
        sudo sed -i.bak 's/local/all granted/g'
/opt/lampp/etc/extra/httpd-xampp.conf
        sudo chmod 755 /opt/lampp/etc/extra/httpd-
xampp.conf
        sudo /opt/lampp/lampp restart
        sudo chmod 777
/opt/lampp/phpmyadmin/config.inc.php
        sudo echo '$cfg["Servers"][$i]["verbose"]
= "Amazon RDS";' >>
/opt/lampp/phpmyadmin/config.inc.php
        sudo echo '$cfg["Servers"][$i]["host"] =
"${endpoint}";' >>
/opt/lampp/phpmyadmin/config.inc.php
        sudo echo '$cfg["Servers"][$i]["user"] =
"admin";' >> /opt/lampp/phpmyadmin/config.inc.php
        sudo echo '$cfg["Servers"][$i]["password"]
= "Naveen1999";' >>
/opt/lampp/phpmyadmin/config.inc.php
        sudo echo '$cfg["Servers"][$i]["port"] =
"3306";' >> /opt/lampp/phpmyadmin/config.inc.php
        sudo echo
'$cfg["Servers"][$i]["auth_type"] = "config";' >>
/opt/lampp/phpmyadmin/config.inc.php
        sudo echo
'$cfg["Servers"][$i]["AllowNoPassword"] = true;' >>
/opt/lampp/phpmyadmin/config.inc.php
        sudo chmod 400
/opt/lampp/phpmyadmin/config.inc.php
        sudo /opt/lampp/lampp restart
        sudo apt install mysql-client-core-8.0
        sudo mysql -h ${endpoint} -u admin --
password=Naveen1999 clientmsdb < /aws/cms/SQL\
File/clientmsdb.sql
      - endpoint: !GetAtt
RDSDBI32QQE.Endpoint.Address
    Metadata:
      'AWS::CloudFormation::Designer':
```

```yaml
                          id: 355e45e0-9d36-49a7-8d8d-fc3b7f760b9c
                    DependsOn:
                      - RDSDBI32QQE
                  ASASG3CEB8:
                    Type: 'AWS::AutoScaling::AutoScalingGroup'
                    Properties:
                      AutoScalingGroupName: my-asg-new1
                      LaunchConfigurationName: my-lc-new1
                      AvailabilityZones:
                        - us-east-1d
                        - us-east-1c
                      DesiredCapacity: 2
                      MaxSize: 3
                      MinSize: 2
                      TargetGroupARNs:
                        - !Ref ELBV2TG4A6L0
                    Metadata:
                      'AWS::CloudFormation::Designer':
                        id: 1453d6ad-0801-4957-a2f6-c9cc5887b06b
                    DependsOn:
                      - ELBV2TG4A6L0
                      - ASLC55LJA
```

creation of stack code

```
pipeline{
          agent any
            stages{
              stage('hosting application'){
               steps{
                  sh "ls"
                    sh "aws cloudformation create-stack --stack-name
          mystack21 --template-body file://awscli/stack.yml --region us-east-
          1"
                }
              }
            }
          }
```
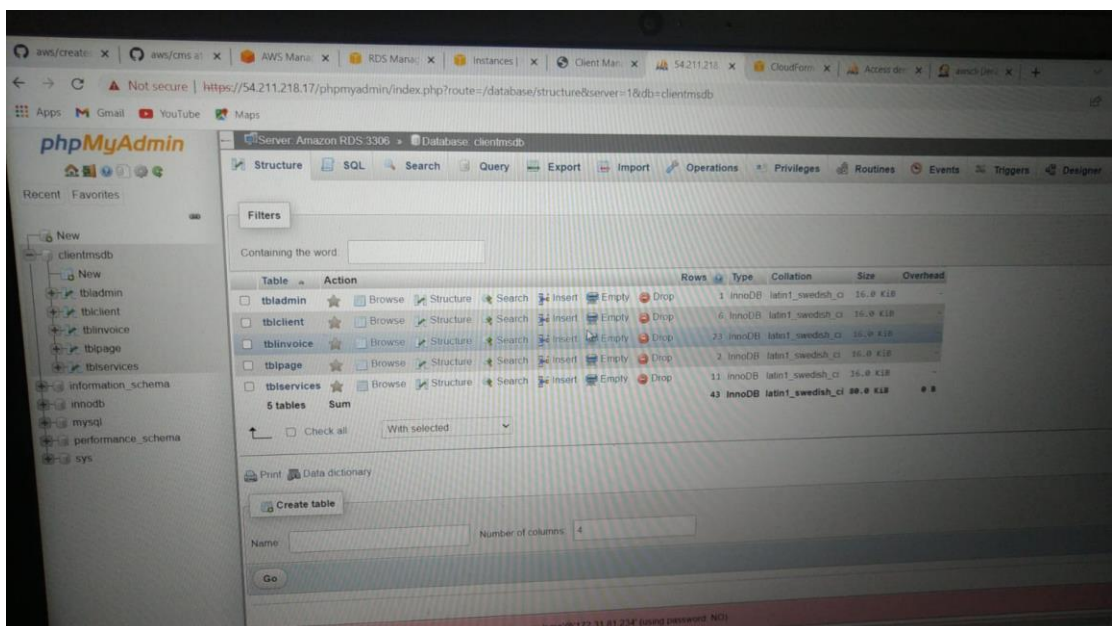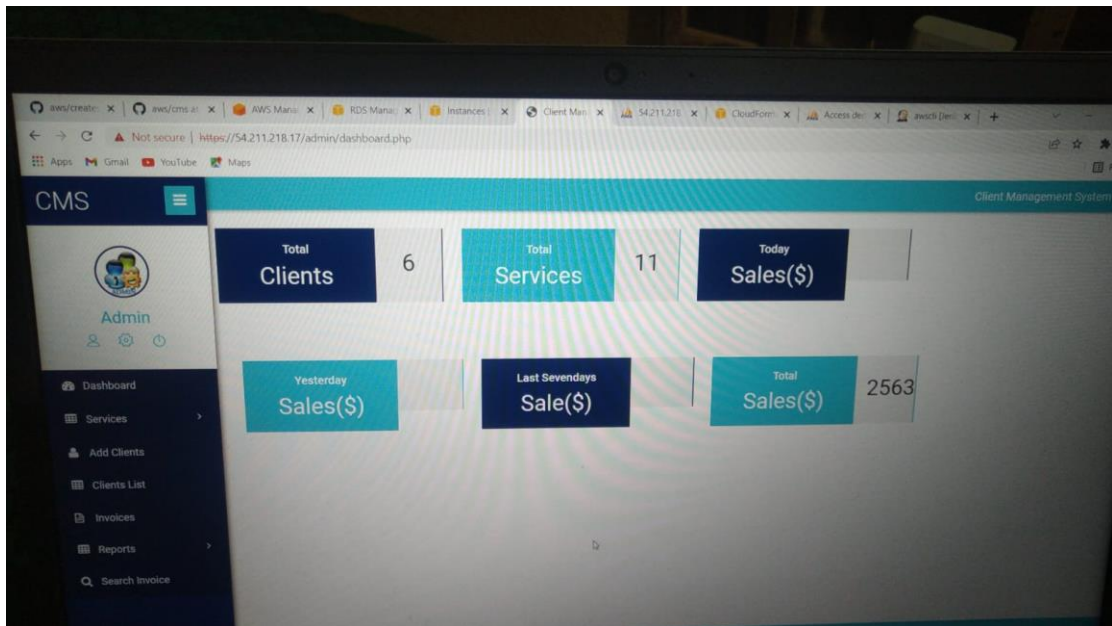
After the Jenkin job success all the required AWS services are created and the project is deployed for using

4 Final out put of the project

5 Referrals

For awscli code

https://github.com/naveen101999/aws/blob/main/awscli/stack.yml

project base code

https://phpgurukul.com/client-management-system-using-php-mysql/