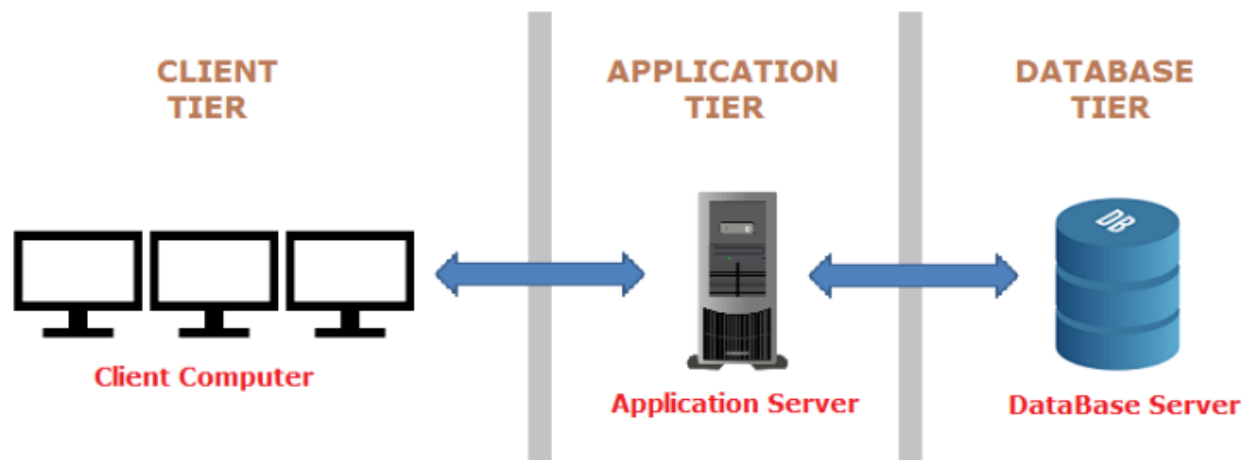**PROJECT 5: CLIENT-SERVER ARCHITECTURE USING MYSQL DATABASE**

A client-server architecture can be referred to as a system that hosts, provides, and maintains a majority of the resources and services that the client requests. These requests and services are supplied across a network under this architecture. The client-server has gradually decreased the time needed for programme development by dividing information exchange responsibilities into client and server roles. The client which can also be referred to as the computer or device submits the request, while the server which is a physical or virtual device is the one who provides the service.

Majorly, in client-server environments, the server processes the data and returns the results to the clients, hence accelerating the pace of performance. For instance, on a workstation, a printer may be connected to one computer, which acts as the client, while other computers sharing the workstation serve as the server. Application servers, database servers, and PCs are the typical components of 3tier client-server architecture as illustrated below.


However, this project aims to implement a client-server architecture with the MySQL Database Management System. In this case, the client-server must establish a user connection with the application server, which then fetches or returns data from the MySQL database.

Project 5 - Server

The first step will be to launch two instances. This can also be done directly from the remote

server, all you need to do is open up 2 different terminals. In this project, I will launch two EC2 Instances on my AWS Virtual Servers. To help differentiate the instances, I will use Project5-Server and Project5-Client as the instance names.

Launching Instances
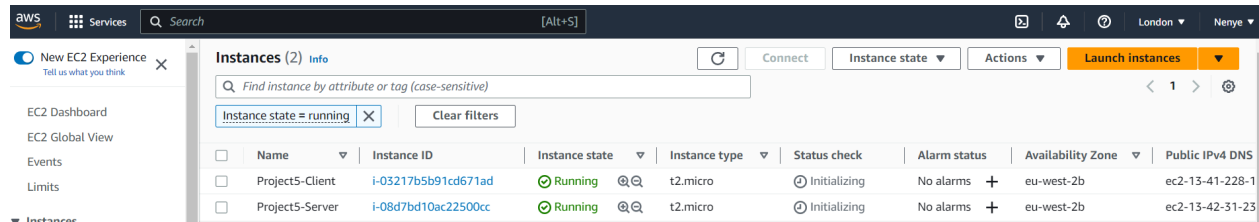
Once this has been launched, I had to connect using their respective SSH Client. As always, it is important to run the below command to check for new releases and updates on Ubuntu.

Installing MySQL Server

Now we have both instances running, time to set up the DBServer environment by installing our SQL server.

Creating User and Databases

Following the completion of this, I executed the command below to switch over to the MySQL database environment to create a user and assign some privileges.

```
ubuntu@ip-172-31-44-92:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.33-0ubuntu0.22.04.2 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

I created the remote_user and assigned a password to it before the creation of the DevOps and Project5 Databases.

```
mysql> CREATE USER 'remote_user'@'%' IDENTIFIED WITH mysql_native_password BY '        ';
Query OK, 0 rows affected (0.02 sec)

mysql> CREATE DATABASE Devops_db;
Query OK, 1 row affected (0.01 sec)

mysql> GRANT ALL ON Devops_db.* TO 'remote_user'@'%' WITH GRANT OPTION;
Query OK, 0 rows affected (0.00 sec)

mysql> CREATE DATABASE Project5_db;
Query OK, 1 row affected (0.01 sec)

mysql> GRANT ALL ON Project5_db.* TO 'remote_user'@'%' WITH GRANT OPTION;
Query OK, 0 rows affected (0.01 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)

mysql> exit
```

To validate the database password credentials, you must run the below command.

```
ubuntu@ip-172-31-44-92:~$  sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: d
```

Oops! I must have forgotten a step when I created the user since I received the following error.

```
New password:

Re-enter new password:
    ... Failed! Error: SET PASSWORD has no significance for user 'root'@'localhost' as the authentication method used doesn't store authentication data in the
MySQL server. Please consider using ALTER USER instead if you want to change authentication parameters.
```

I had to go back and execute the SQL query below, which allowed me to progress to the password validation.

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY '          ';
Query OK, 0 rows affected (0.01 sec)

mysql> exit
Bye
ubuntu@ip-172-31-44-92:~$ sudo mysql_secure_installation
```

Remember that to execute this operation, you must permit connections from remote hosts. To enable users to connect to the MySQL database from other networks and IP addresses, update the bind address to 0.0.0.0.

```
ubuntu@ip-172-31-44-92:~$ sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf
```

```
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 127.0.0.1
mysqlx-bind-address     = 127.0.0.1
#
# * Fine Tuning
```

It is important to restart the service after changes like this have been made.

```
ubuntu@ip-172-31-44-92:~$ sudo systemctl restart mysql
```

Enabling TCP Port

The Project5-Server instance's TCP port 3306 must now be enabled as the final step. Click on the security group and then select "inbound rules" to do this. Ensure to use the Project5-Client Private IP which in this case is 172.31.40.119 and then click on add rules. If unsure of the IP this can easily be gotten from the EC2 instance as a Private IP address or by using the hostname command on the terminal.

Instance: i-08d7bd10ac22500cc (Project5-Server)

Details	Security	Networking	Storage	Status checks	Monitoring	Tags

▼ **Security details**

IAM Role
–

Owner ID
▢ 476363385873

Launch time
Sun May 28 2023 21:55:39 GMT+(

Security groups
▢ sg-07f94d791c99ae0ac (launch-wizard-10)

Inbound rules Info

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
sgr-0b17dccb102d03e7f	MYSQL/Aurora ▼	TCP	3306	Custom ▼ 🔍		Delete
				172.31.40.119/32 ✕		

Project 5 - Client

As always, please run the sudo apt update command to check for new releases and updates on Ubuntu.

Installing MySQL Client

The next step is to get the Client ready to accept connection from MySQL by running the below command to install the MySQL client.

```
ubuntu@ip-172-31-40-119:~$ sudo apt install mysql-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

Connecting to MySQL Server

Type in the following syntax to access your database:
$ sudo mysql -u {username} -h {hostIP} -u -p
Password: {your password}

Where the **host IP**: the IP of the MySQL server that you are assigned to, e.g., 172.31.44.92

username: the name of your MySQL database user

password: the password you created to use to access your MySQL database

```
ubuntu@ip-172-31-40-119:~$ sudo mysql -u remote_user -h 172.31.44.92 -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.33-0ubuntu0.22.04.2 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Now you are successfully connected to the MySQL Database from the Client. To view the Databases that I had created on the Project5-Server, run the Show databases; command and there you have it!

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| Devops_db          |
| Project5_db        |
| information_schema |
| performance_schema |
+--------------------+
4 rows in set (0.01 sec)
```

Conclusion

Imagine a world without a client-server system, even the most basic activities would take a very long time to accomplish, wasting resources and manpower. Due to its structural features, the client-server system covers a wide range of networking and database administration topics. It is crucial to recognise that technological advancement is a breakthrough in both the business and IT worlds.

REFERENCES

3-tier Client Server Architecture in DBMS PDF, Doc, images (2023) 3-tier client-server architecture in dbms. Available at:
https://www.pdfprof.com/PDF_Image.php?idt=69458&t=27 (Accessed: 29 May 2023).

How do I access my MySQL database? (2023) HostMySite. Available at:
https://www.hostmysite.com/support/linux/mysql/access/ (Accessed: 28 May 2023).

Oluwatosin, H.S., 2014. Client-server model. IOSR Journal of Computer Engineering, 16(1), pp.67-71.

Terra, J. (2023) What is client-server architecture? everything you should know: Simplilearn, Simplilearn.com. Available at:
https://www.simplilearn.com/what-is-client-server-architecture-article (Accessed: 29 May 2023).