USE CASE STUDY REPORT

Group No.: Group 20

Student Names: Ankita Shukla and Soham Palnitkar

Executive Summary:

The main purpose of this project was to design and implement a relational database that is designed for supply chain industry and can be used by any painting service provider and potential customers. This database is designed for all the people who own a painting solutions company and customers who wish to take painting services. This Database was designed for the company owners who is always having a complaint about the log keeping and maintenance of the record of the workers, customers and suppliers. This database has eliminated a significant issue by not having any duplicate data. This database produced significant cost-saving benefits to the company for workforce management. The database also provides extensive features for managing and looking over the workforce.

The database was created and developed with input from experts in the field while also considering all the criteria and the potential effects of each field on the others. The first stage was to gather the requirements from the experts, after which the EER and UML diagrams were created. The conceptual model was then mapped to a relational model, along with the requisite foreign keys and primary keys for the database. The database was then fully constructed in MySQL with all of the relations and tables, and a small prototype of the original database was implemented in the MongoDB NoSQL database with three tables and two relations to test its viability and usability there.

A remarkable accomplishment is the creation and implementation of the database. In order to gain some analytical capabilities and insights from the data entered in the database, we first implemented it in the MYSQL and NoSQL databases and then connected the database to Python. We were able to develop some really important insights after installing the database in the Jupyter notebook, such as understanding about the various customer types that are seeking painting services, and we were also able to offer a few insights for the suppliers providing the necessary paints and tools. We were able to present a bar graph that illustrated the many customer categories and services the business provides.

I. Introduction

The Splash Painters Company offers painting services in the Boston metropolitan area. It has amassed a sizable customer base in the city because to its high-quality work and effective maintenance crew. It collaborates with a number of suppliers to meet the need for paints. As the amount of data grows, the company now wants to create a database for its distributors and customers. The objective is to create and keep track of supply chain information and customer requests.

Platinum, Gold, and Silver class service packages are offered by Splash Painters. Painting, maintenance, and damage recovery are among the premium services offered by Platinum, while painting and damage recovery are offered by Gold. Only the painting service is offered under the silver tier.

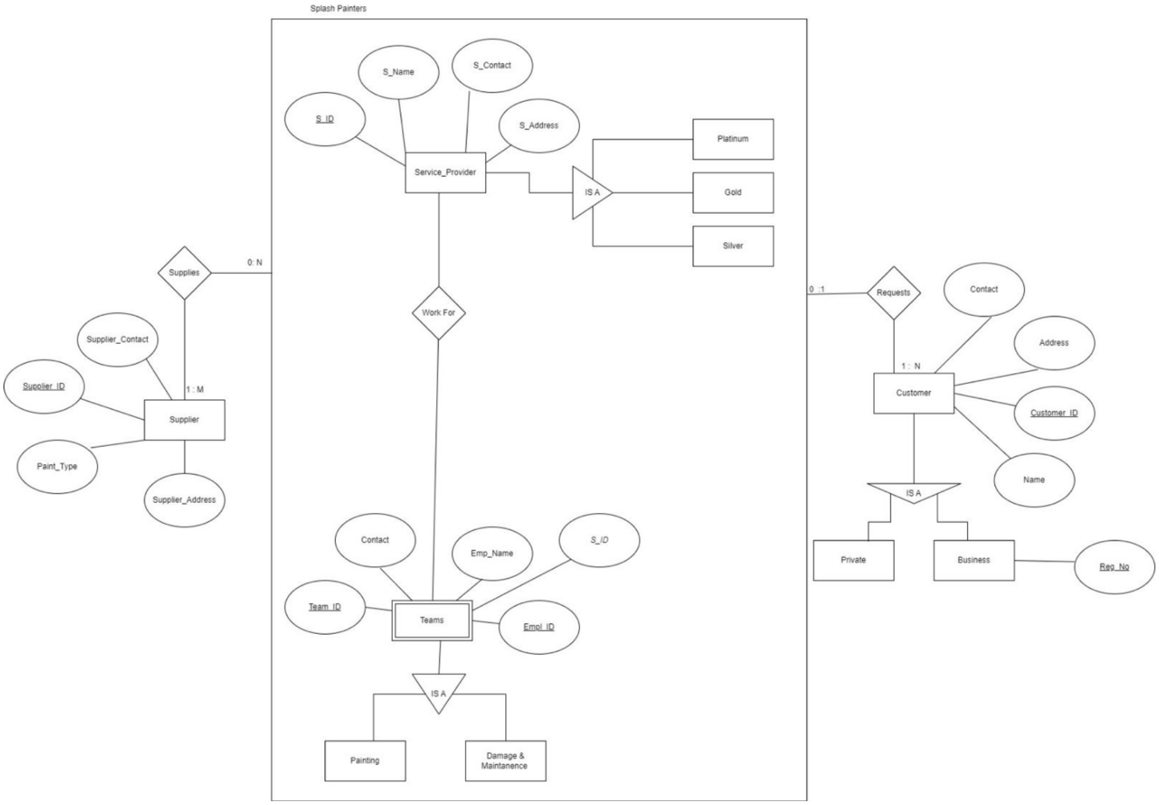
Only one service can be requested by a client at a time. Private and corporate clients are the two main customer groups that the company deals with. Customers register online with their basic information, such as name, address, and contact information, and an individual ID is created for each customer.

The personnel at Splash are separated into two teams: the painting team and the damage and maintenance team. Each team member is given an ID, and their personal information including name, phone number, and address is recorded in the business database.

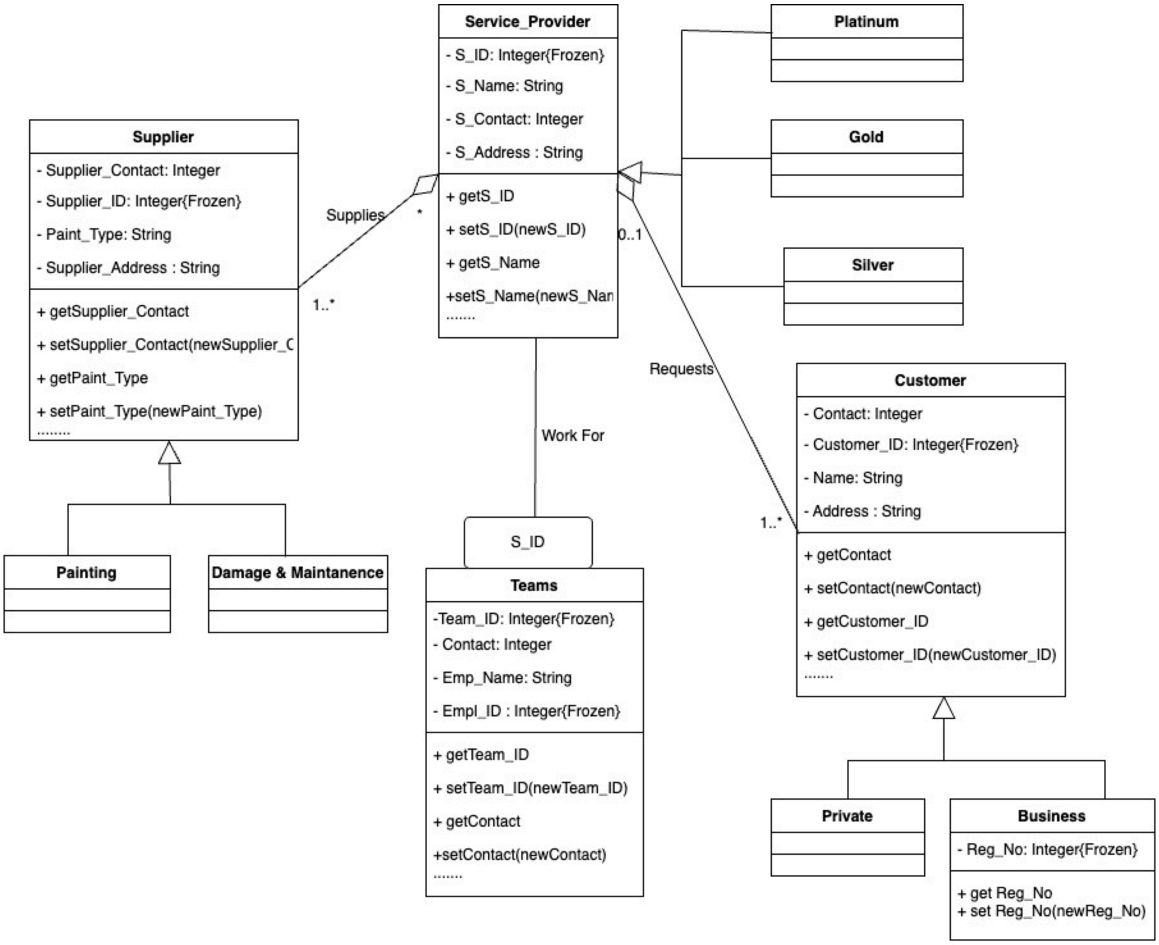
The company orders the paint kind and tools from the distributors in accordance with the service that the customer has asked. Company keeps records of distributors' IDs, names, addresses, and contact information. The distributor offers various types of paint for the various service tiers: Royal paint is offered for the platinum tier, Premium paint for the gold tier, and Standard paint for the silver tier. Only if the customer's complete package includes those services will they be able to access utility services.

II. Conceptual Data Modeling

1. EER Diagram



2. UML Diagram



III. Mapping Conceptual Model to Relational Model

Service_Provider(S_ID, S_Name, S_Contact, S_Address, Service_Type)

Platinum(*S_ID*)

Gold(*S_ID*)

Silver(*S_ID*)

Supplier(Sup_ID, Paint_Type, Sup_Contact, Sup_Address)

Supplies(*Sup_ID, S_ID*)

Teams(T_ID, T_Contact, Emp_Name, Emp_ID, *S_ID*, T_Type)

Painting_Team(*T_ID*)

Damage_Maintenance_Team*(T_ID*)

Customer(Cust_ID *, S_ID*, Cust_Contact, Cust_Address, Cust_Name, Cust_Type)

Business(*Cust_ID*, REG_NO)

Private(*Cust_ID*)

(Primary Keys are Underlined; Foreign Keys are in Italics)

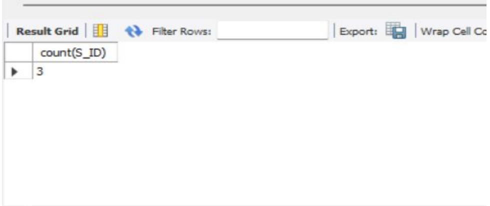
IV. Implementation of Relation Model via MySQL and NoSQL

MySQL Implementation:

The database was created in MySQL and the following queries were performed:

<u>Query 1:</u> Get Count of service ids from service provider providing service type 'Gold'

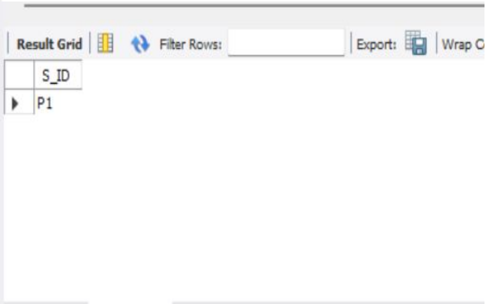
```
select count(S_ID) from service_provider where
Service_Type='Gold';
```



count(S_ID)
3

<u>Query 2:</u> Get service id which offers service to a customer with registration number '3434'

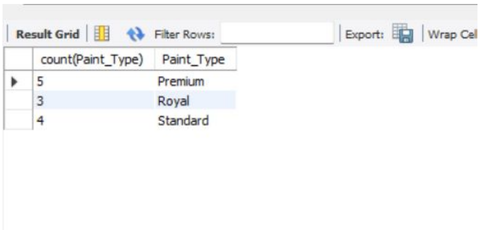
```
select s.S_ID from service_provider s
inner join Customer c
on s.S_ID=c.S_ID
inner join Business b
on c.Cust_ID=b.Cust_ID
where b.Reg_No='3434';
```



S_ID
P1

<u>Query 3:</u> Get the count of number of customers obtaining different paint types offered by all suppliers

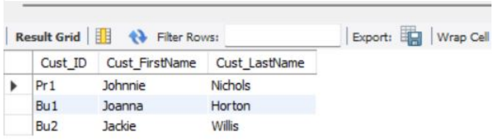
```
select count(Paint_Type),Paint_Type from Supplier
group by Paint_Type;
```



count(Paint_Type)	Paint_Type
5	Premium
3	Royal
4	Standard

<u>Query 4:</u> Get customer first name starting with 'J' from customer details table

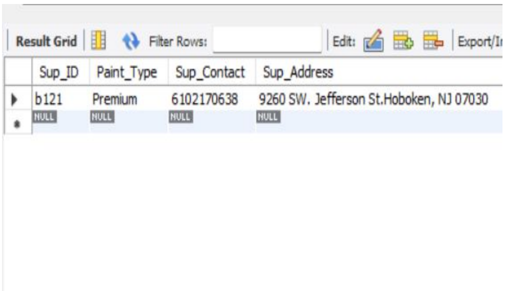
```
select * from customer_details
where Cust_FirstName like 'J%';
```



Cust_ID	Cust_FirstName	Cust_LastName
Pr1	Johnnie	Nichols
Bu1	Joanna	Horton
Bu2	Jackie	Willis

<u>Query 5:</u> Get details of the supplier who supplies to the service provider with id 'P1'

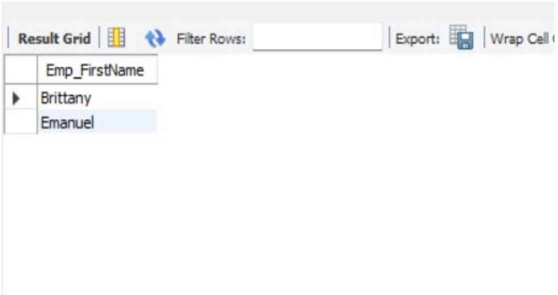
```
select * from supplier
where Sup_ID = ANY
(select Sup_ID from supplies
where S_ID='P1');
```



Sup_ID	Paint_Type	Sup_Contact	Sup_Address
b121	Premium	6102170638	9260 SW. Jefferson St.Hoboken, NJ 07030
NULL	NULL	NULL	NULL

Query 6: **Get fisrt name for employees who work in a team with team Id 'A4' and employees who work for damage and maintenance team**

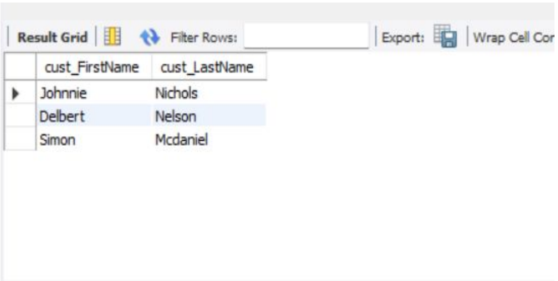
select Emp_FirstName from employee_details
where Emp_ID in (select Emp_ID from teams where T_ID='A4')
union
select Emp_FirstName
from employee_details
where Emp_ID = (select Emp_ID from Teams where T_Type='DT');



Emp_FirstName
Brittany
Emanuel

Query 7: **Get first name and last name for customers who requested 'Premium' paint type**

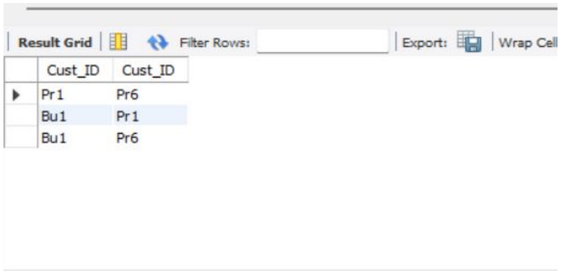
select cd.cust_FirstName, cd.cust_LastName
from Customer_Details cd
where exists (select * from Customer c, Supplies s, Supplier sup where cd.Cust_ID = c.Cust_ID and c.S_ID = s.S_ID and s.Sup_ID=sup.Sup_ID and sup.Paint_Type ='Premium');



cust_FirstName	cust_LastName
Johnnie	Nichols
Delbert	Nelson
Simon	Mcdaniel

Query 8: **Get the pair of customer ids which are located in New Jersey city ('NJ')**

select c1.Cust_ID, c2.Cust_ID
from customer c1, customer c2
where c1.Cust_Address like '%NJ%' and c2.Cust_Address like '%NJ%' and c1.Cust_ID < c2.Cust_ID;

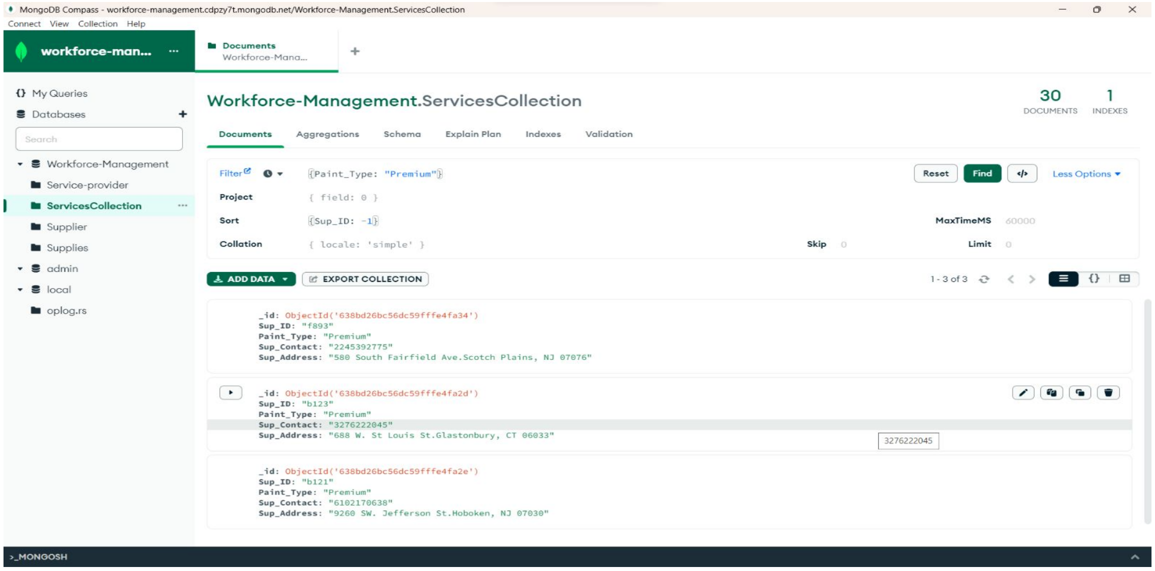


Cust_ID	Cust_ID
Pr1	Pr6
Bu1	Pr1
Bu1	Pr6

NoSQL Implementation:

Two tables (Service Provider, Supplier) and a relation(Supplies) have been created in MongoDB. The following Cypher queries were done:

Query 1: **Find the supplier details who supplies 'Premium' paint sorted in descending order**



Workforce-Management.ServicesCollection

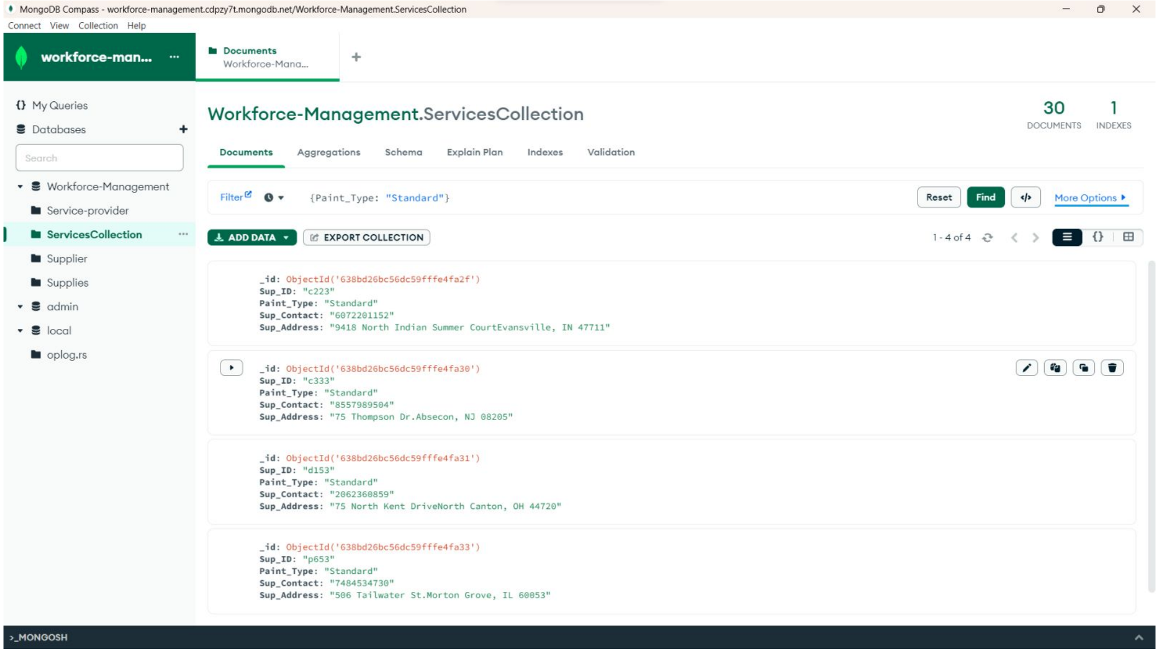
30 DOCUMENTS 1 INDEXES

```
Filter      {Paint_Type: "Premium"}
Project     { field: 0 }
Sort        {Sup_ID: -1}                                   MaxTimeMS  60000
Collation   { locale: 'simple' }          Skip  0          Limit      0
```

1 - 3 of 3

```
_id: ObjectId('638bd26bc56dc59fffe4fa34')
Sup_ID: "f893"
Paint_Type: "Premium"
Sup_Contact: "2245392775"
Sup_Address: "580 South Fairfield Ave.Scotch Plains, NJ 07076"

_id: ObjectId('638bd26bc56dc59fffe4fa2d')
Sup_ID: "b123"
Paint_Type: "Premium"
Sup_Contact: "3276222045"
Sup_Address: "688 W. St Louis St.Glastonbury, CT 06033"
                                                        3276222045

_id: ObjectId('638bd26bc56dc59fffe4fa2e')
Sup_ID: "b121"
Paint_Type: "Premium"
Sup_Contact: "6102170638"
Sup_Address: "9260 SW. Jefferson St.Hoboken, NJ 07030"
```

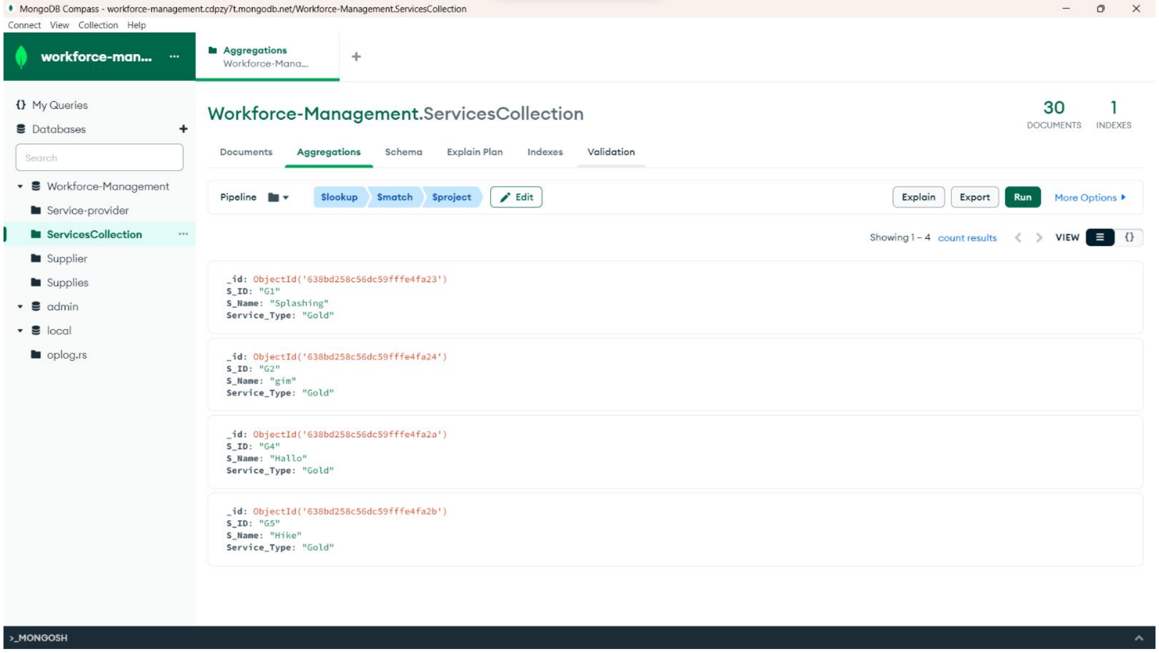
Query 2: Find the supplier details who supplies 'Standard' paint



```
{Paint_Type: "Standard"}
```

```
_id: ObjectId('638bd26bc56dc59fffe4fa2f')
Sup_ID: "c223"
Paint_Type: "Standard"
Sup_Contact: "6072201152"
Sup_Address: "9418 North Indian Summer CourtEvansville, IN 47711"

_id: ObjectId('638bd26bc56dc59fffe4fa30')
Sup_ID: "c333"
Paint_Type: "Standard"
Sup_Contact: "8557989504"
Sup_Address: "75 Thompson Dr.Absecon, NJ 08205"

_id: ObjectId('638bd26bc56dc59fffe4fa31')
Sup_ID: "d153"
Paint_Type: "Standard"
Sup_Contact: "2062360859"
Sup_Address: "75 North Kent DriveNorth Canton, OH 44720"

_id: ObjectId('638bd26bc56dc59fffe4fa33')
Sup_ID: "p653"
Paint_Type: "Standard"
Sup_Contact: "7484534730"
Sup_Address: "506 Tailwater St.Morton Grove, IL 60053"
```

Query 3: Find the supplier Id, supplier name for the service providers who provides service type of 'Gold' category using aggregation

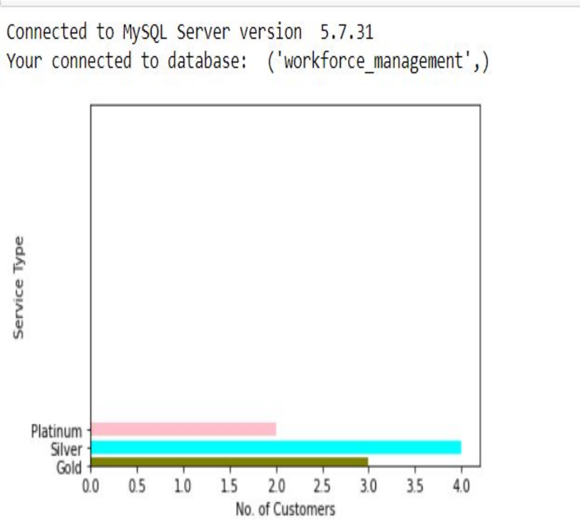


```
_id: ObjectId('638bd258c56dc59fffe4fa23')
S_ID: "G1"
S_Name: "Splashing"
Service_Type: "Gold"

_id: ObjectId('638bd258c56dc59fffe4fa24')
S_ID: "G2"
S_Name: "gim"
Service_Type: "Gold"

_id: ObjectId('638bd258c56dc59fffe4fa2a')
S_ID: "G4"
S_Name: "Hallo"
Service_Type: "Gold"

_id: ObjectId('638bd258c56dc59fffe4fa2b')
S_ID: "G5"
S_Name: "Hike"
Service_Type: "Gold"
```

V. Database Access via R or Python

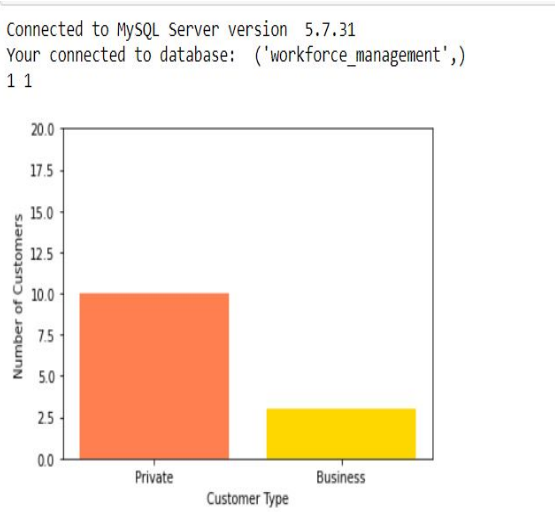
The database is accessed using Python and visualization of analyzed data is shown below. The connection of MySQL to Python is done using mysql.connector, followed by cursor.excecute to run and fetch all from query, followed by using matplotlib to plot the graphs for the analytics.

Graph 1. No. of customers in different service categories



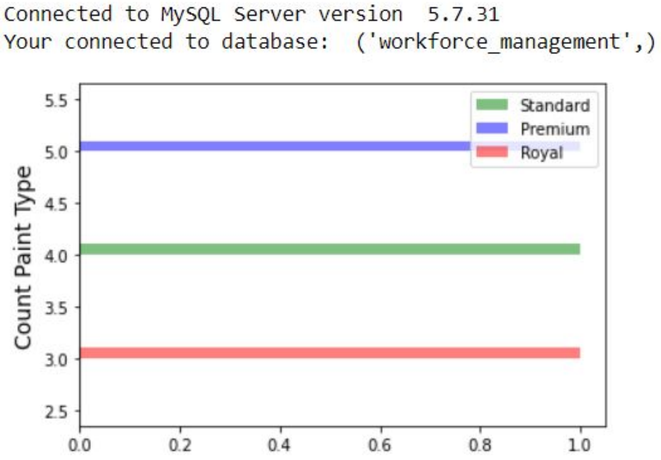
```
Connected to MySQL Server version  5.7.31
Your connected to database:  ('workforce_management',)
```

Graph 2. No. of customers in different customer categories



```
Connected to MySQL Server version  5.7.31
Your connected to database:  ('workforce_management',)
1 1
```

Graph 3. Count of suppliers suppling paint of different categories



```
Connected to MySQL Server version  5.7.31
Your connected to database:  ('workforce_management',)
```

VI. Summary and recommendation

The Database is absolutely ready to be implemented by the organization, fully built, and capable of handling enormous amounts of data. The database will make it simple for the service providers to keep track of everyone's information, including that of clients and employees. Both the service provider and the client can access this database, which is mobile phone application ready. The database will provide information on each and every client, vendor, and worker at the company. The generation of insights from the data was the next part that was completed. The data was fetched into the python notebook using the MySQL connector after being connected, and various data retrieval queries were run. With the aid of the matplotlib library, bar graphs and histograms were generated for the data retrieved using queries from the MySQL database.

The database could be improved further by adding necessary fields and deleting unnecessary fields and numerous keys from the tables. For instance, storing login id and password in the customer table would ease up the process. Additionally, the database must be able to spot duplicate data and not contain graphs.

This database's NoSQL implementation by MongoDB would be vulnerable. It is important to conduct more research on how to build a distinctive relational database, similar to the SQL database. There should be research done on how to import the data from a single CSV file and detect the relation beforehand, rather than writing the data to input for each of the fields. Even though using a relational database has more advantages than using a non-relational database does, the advantages of the table outweigh those of the NoSQL database for the same purpose.