Vehicle Management System

Harshan Ganugula

Firdose Shaik

Jenvith Manduva

overview

- The purpose of our vehicle management system is to automate the existing manual system by computerizing the data with easy accessing and manipulating it. Thus this will help organizations in better utilization of resources.
- This will assists the vehicle dealer to reduce the manual work and deliver the high level of service and support to the customers.
- Developed a web interface for the users to buy vehicles, book appointments from different branches, as well as for the employees to update the data.

Objective

The objective of our system is to arrange a platform to enable showroom dealers to maintain the records of the employees, their salaries, and sales of the vehicles, which helps in forming tighter relations with the customers and providing a high level of service to the customers after the sales.

Data Sources

- Vehicle data is populated by scraping data from third-party vehicle websites
- The user's data will be populated using User Fake API endpoints.
- We scrape the most common services provided using websites for the services table.
- For the remaining tables we populated data manually.

Libraries Used

- Flask (Web Development)
- Selenium(Web Scraping)
- Axiom (API Endpoint)
- TextBlob, Profanity (NLP)
- Plotly (Data Visualisation)
- Mockaroo (Data Population)

Data Pre-Processing

- In order to fulfill the requirements of our system design, the information we obtain from scraping websites and calling API endpoints will be preprocessed.
- In order to clean up the data, we will initially remove any entries that are null or duplicate, and then we will convert the data into the required format that was specified when the tables were being created.
- For the data in the reviews entity, we have filtered negative reviews and applied **sentiment analysis using machine learning** to filter the inappropriate words from the sentences.
- The remaining data for the other tables will be populated manually following the data collected from the APIs and scraped data, respectively.

- In addition, we have used stored procedures and triggers to check that the data is being added correctly and make sure there were no errors.
- To foster the query reusability, we have included multiple functions.

Entities Information

Vehicle: The data we have collected for this entity is thorough **web scraping** from multiple websites.In this vehicle entity, we will be storing vehicle names, model, engine details, no.of seats and the price of the vehicle. To retrieve all this information, we will be assigning a primary key vehicle_id to this entity.

Employees : This entity has all the information of the employee designation, salary, name, address, number and the branch which they belong to.

customers : This entity consists of all the names, email address,gender type and mobile number of each customers.

services requested: Service requested entity has details of where the service is being done, details of the employee assigned to the service and requested and completed dates of the service.

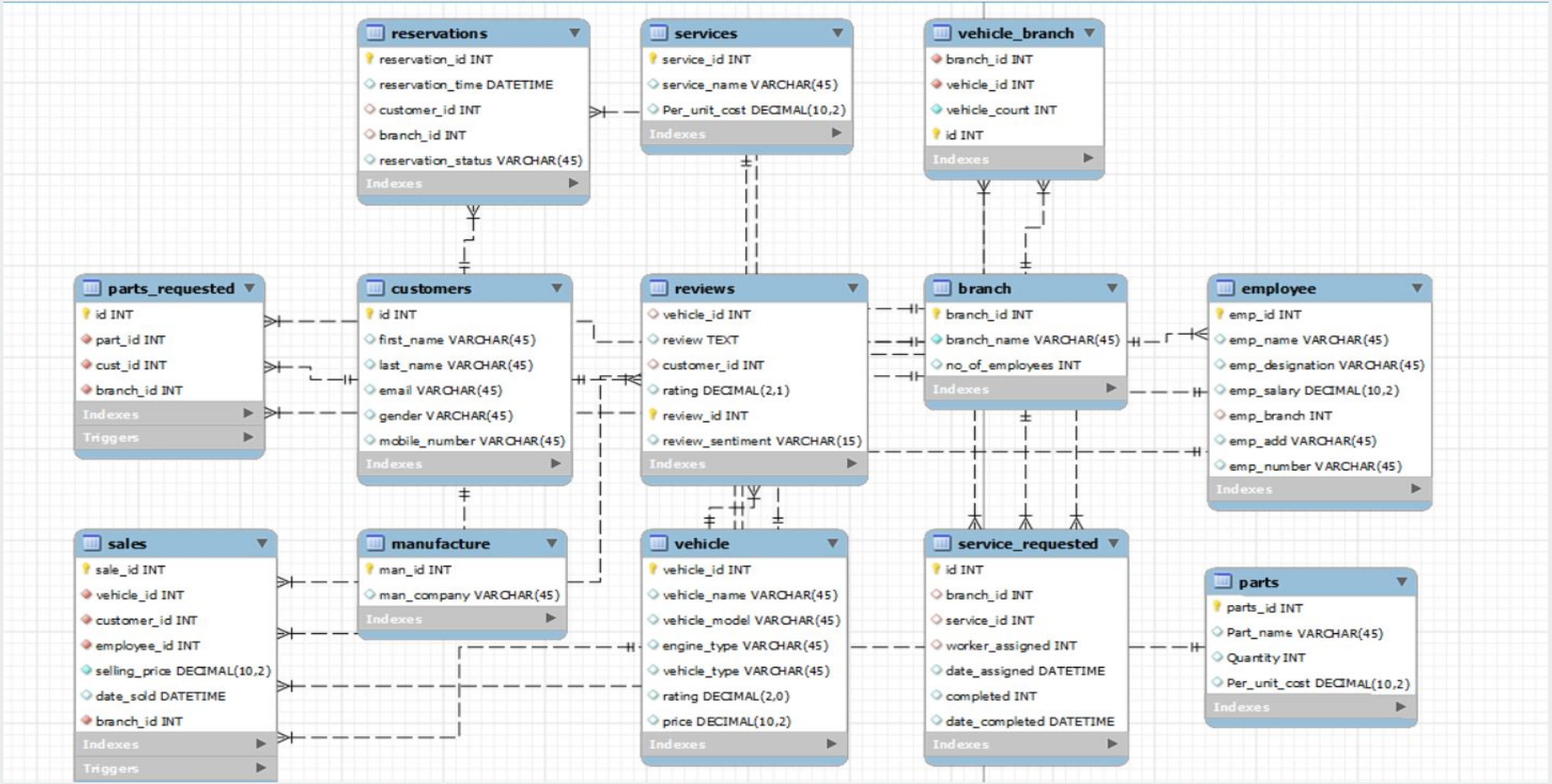
Vehicle Branch: It has the details of the vehicles that belongs to a particular branch and count of total ,.,,..number of the vehicles in a specific branch.

Branch: It will store the branch name and number of employees in a particular branch.

Reviews: This entity stores the data of the reviews reported by the customers and by whom it was recorded. It also has the decimal rating given by the customer on a scale of 1 to 5 , 1 being being worst and 5 being excellent. Here we have used sentiment analysis to predict the sentiment of the customer.

Sales : This table will include the sale id, vehicle id, customer id, employee id, selling price, date of sale, and branch id for all vehicle sales.

ER - Diagram


reservations
reservation_id INT
reservation_time DATETIME
customer_id INT
branch_id INT
reservation_status VARCHAR(45)
Indexes
services
service_id INT
service_name VARCHAR(45)
Per_unit_cost DECIMAL(10,2)
Indexes
vehicle_branch
branch_id INT
vehicle_id INT
vehicle_count INT
id INT
Indexes
parts_requested
id INT
part_id INT
cust_id INT
branch_id INT
Indexes
Triggers
customers
id INT
first_name VARCHAR(45)
last_name VARCHAR(45)
email VARCHAR(45)
gender VARCHAR(45)
mobile_number VARCHAR(45)
Indexes
reviews
vehicle_id INT
review TEXT
customer_id INT
rating DECIMAL(2,1)
review_id INT
review_sentiment VARCHAR(15)
Indexes
branch
branch_id INT
branch_name VARCHAR(45)
no_of_employees INT
Indexes
employee
emp_id INT
emp_name VARCHAR(45)
emp_designation VARCHAR(45)
emp_salary DECIMAL(10,2)
emp_branch INT
emp_add VARCHAR(45)
emp_number VARCHAR(45)
Indexes
sales
sale_id INT
vehicle_id INT
customer_id INT
employee_id INT
selling_price DECIMAL(10,2)
date_sold DATETIME
branch_id INT
Indexes
Triggers
manufacture
man_id INT
man_company VARCHAR(45)
Indexes
vehicle
vehicle_id INT
vehicle_name VARCHAR(45)
vehicle_model VARCHAR(45)
engine_type VARCHAR(45)
vehicle_type VARCHAR(45)
rating DECIMAL(2,0)
price DECIMAL(10,2)
Indexes
service_requested
id INT
branch_id INT
service_id INT
worker_assigned INT
date_assigned DATETIME
completed INT
date_completed DATETIME
Indexes
parts
parts_id INT
Part_name VARCHAR(45)
Quantity INT
Per_unit_cost DECIMAL(10,2)
Indexes

Triggers

- When a user purchases a particular vehicle the count of that vehicle in that branch decreases by 1 unit.
- When a particular part of vehicle is bought by the system the system automatically reduces the count of that part in the warehouse.

Stored procedures

In total we have implemented 5 stored procedures.

1. Given employee id, it returns all the details of that employee performance. (Like the name, branch, salary, Services done, revenue generated, profit generated)
2. A stored procedure to assign a service by the manager to an employee working in that particular branch.
3. A stored procedure to update employee salary by only the manager, used multiple functions inside this stored procedure to update salary of the employee.
4. A stored procedure to update vehicle count in a particular branch.
5. A stored procedure to create a reservation.

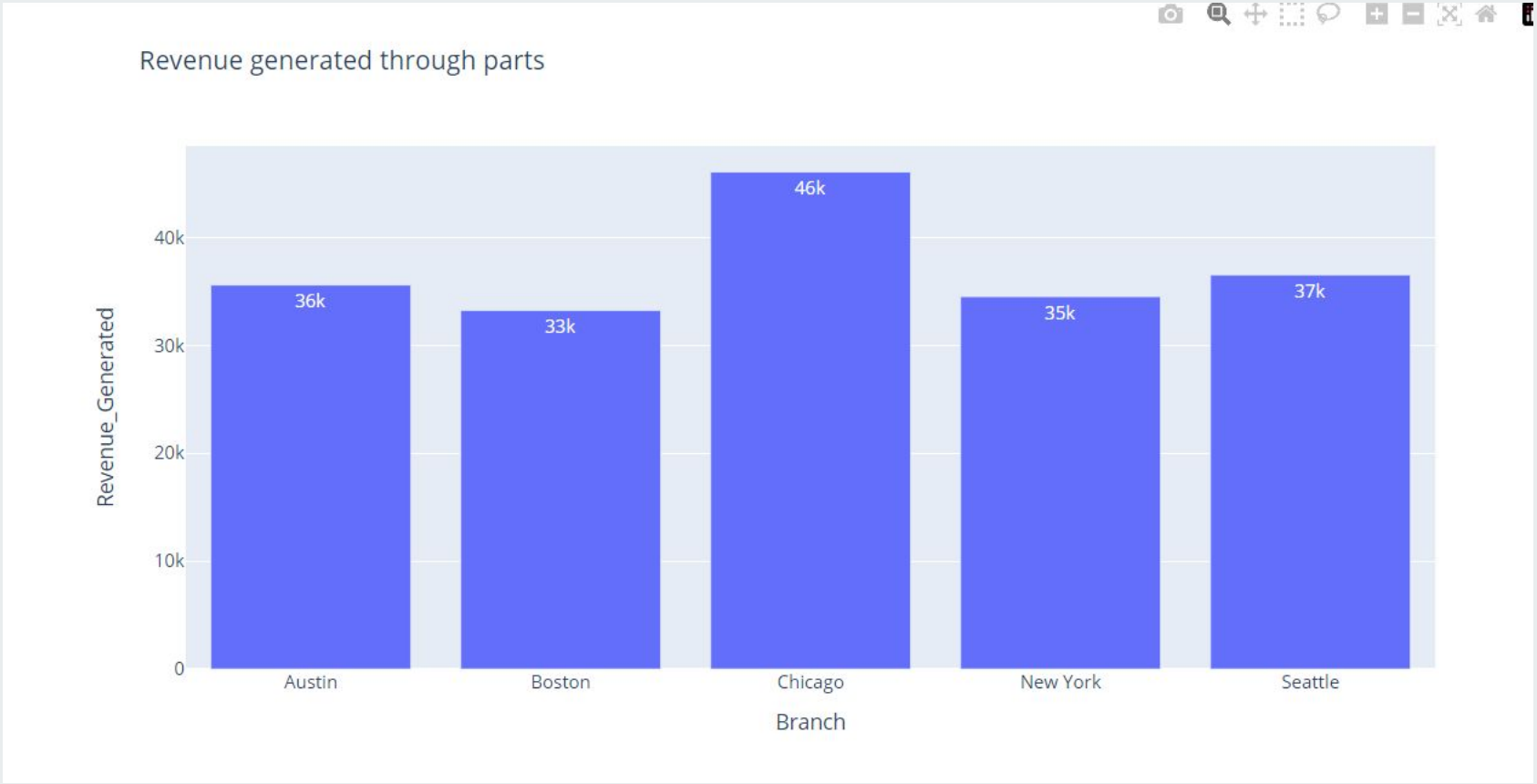
Views

1. Created a view with all the sales information of all the branches until now.

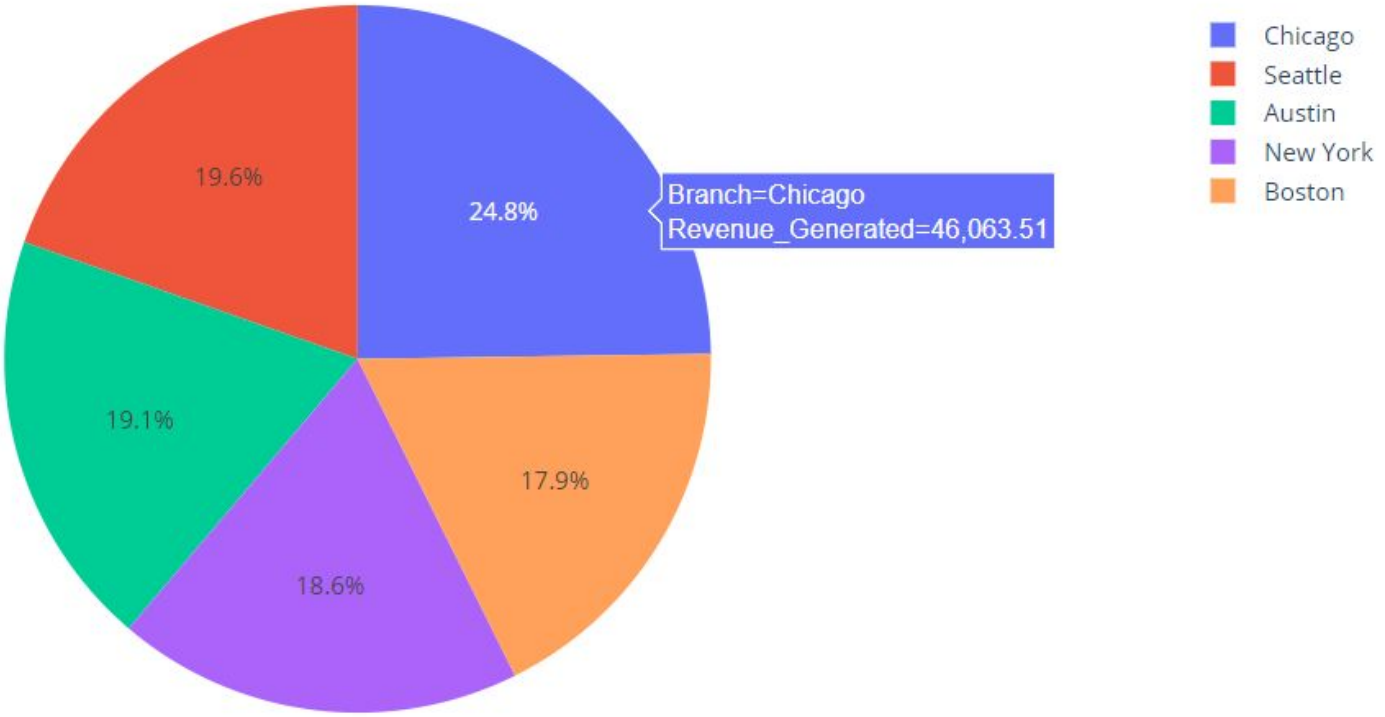
2. Using this view created multiple plot to visualise the revenue generated through Vehicles, Parts and Services.

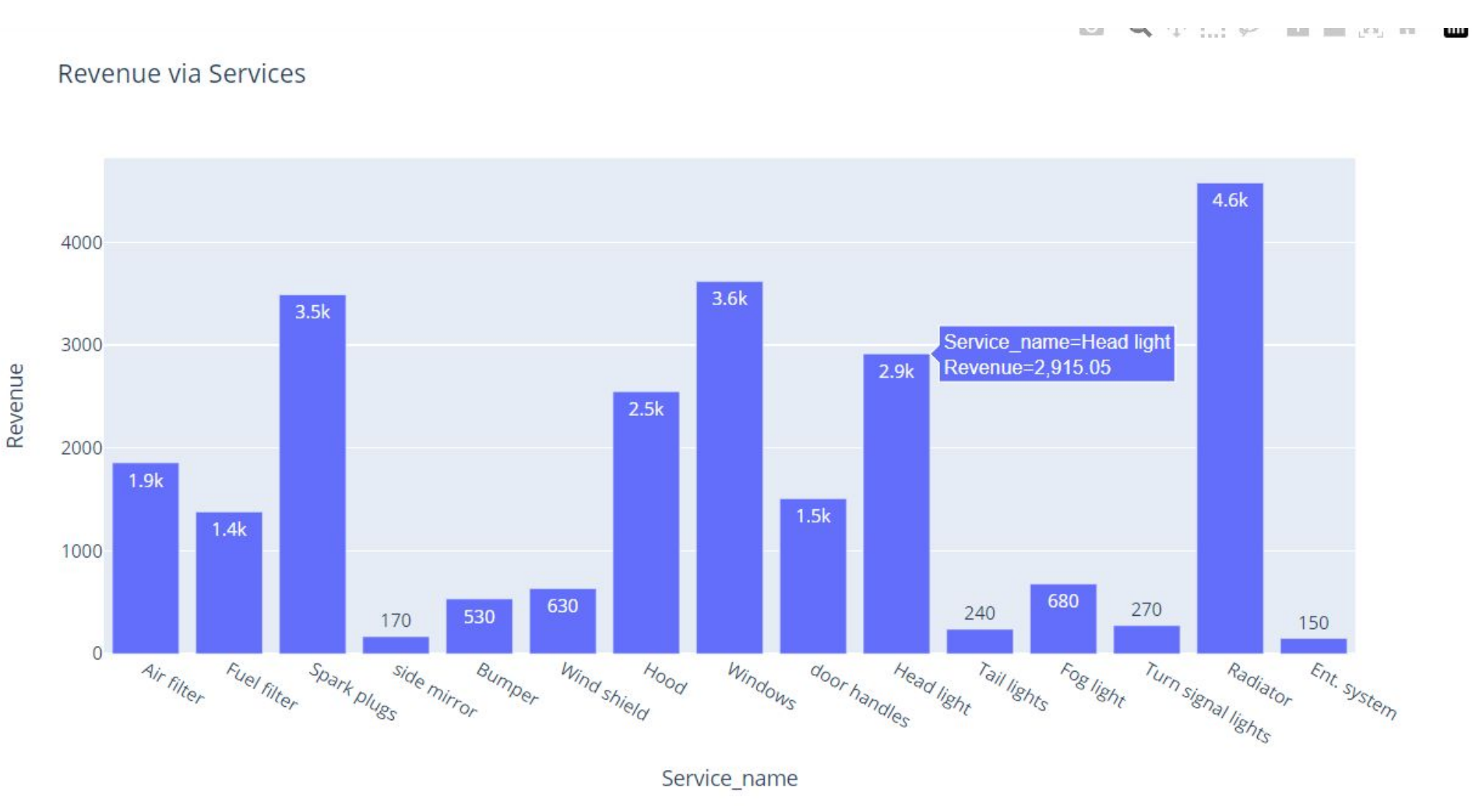
3. And from this we have even interpreted the sales made by each employee.

Results

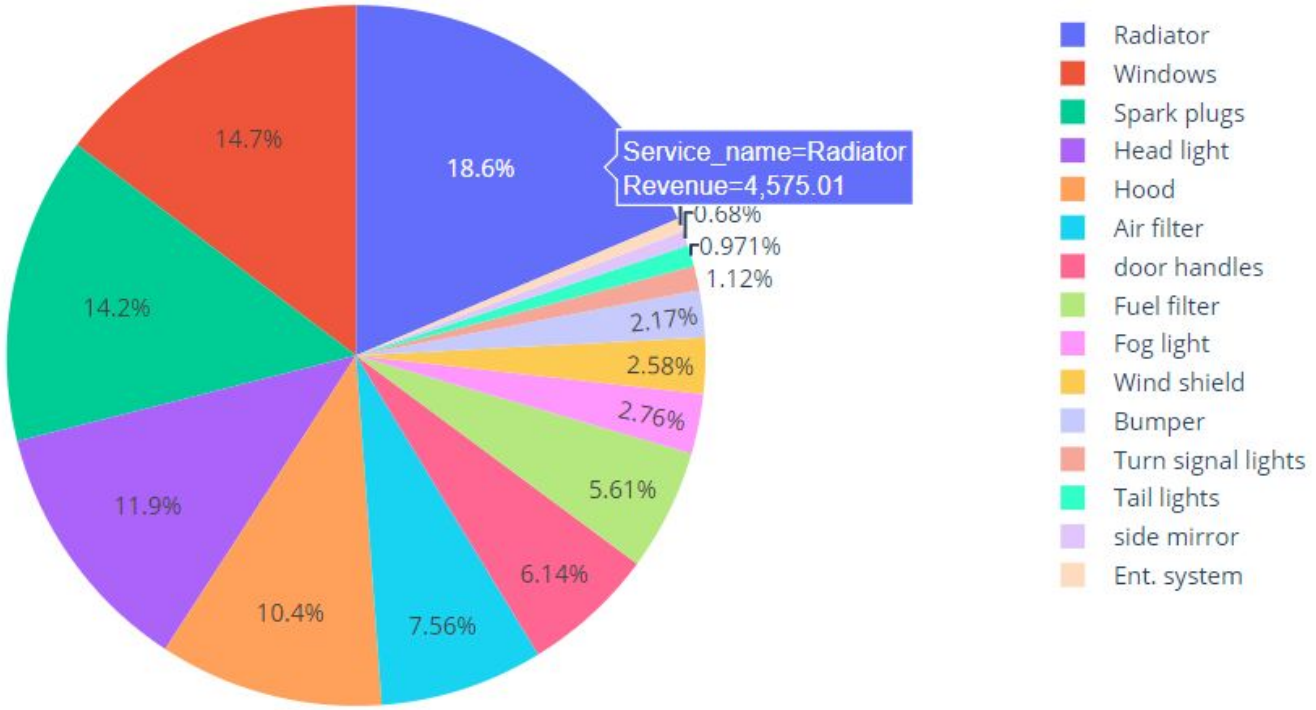

Revenue generated through parts
Revenue_Generated
0
10k
20k
30k
40k
36k
33k
46k
35k
37k
Austin
Boston
Chicago
New York
Seattle
Branch

Revenue generated through parts




Revenue via Services
Revenue
4000
3000
2000
1000
0
1.9k
1.4k
3.5k
170
530
630
2.5k
3.6k
1.5k
2.9k
240
680
270
4.6k
150
Service_name=Head light
Revenue=2,915.05
Air filter
Fuel filter
Spark plugs
side mirror
Bumper
Wind shield
Hood
Windows
door handles
Head light
Tail lights
Fog light
Turn signal lights
Radiator
Ent. system
Service_name

Services Distribution



Conclusion

We have developed this project vehicle management system to digitalize the entire records to accessing it better which improves the business efficacy alongside commencing with the clients in a better way.

This has effectively includes accessing data and manipulating it in the comfort of managing it more efficiently.

Thank You