**CS 631 - Data Management System Design**

**Project Phase 3**

## **Woody’s Automotive Application Project**

**PROJECT TEAM 05**

**Team Members**

Krishna vamsi.peddineni(kp864)

Archith Reddy Billa (AB2679)

Vivekananda.peddineni(vp669)

**Woody’s Automotive Application Project**

**Goal of Project Phase 3:**

The objective of the third phase of the project is to build the fully functional application program that will be used for Woody's automotive application project. The program includes functionality for the registration and maintenance of customers, the provision of online services, and the compilation of data.

**Implementation:**

The Woody's Automotive Application Project was developed with the help of the React frontend technology and Node.js for the backend. To design and populate our database tables, we incorporated mysql workbench. Our primary tool during development was Microsoft's Visual Studio Code.

**Problems Faced:**

First, we had trouble integrating both frontend and backend api’s, because of improper declaration of data types of api path variables.  
We fixed it by mapping data types of columns in entities to the path variables of api variables.

Second, because of foreign key constraints deleting a few unnecessary rows inserted improperly.  
We fixed it by properly deleting the rows in both parent and child entities

**2. User Guide**

**Woody Repair Shop Website Guide:**

Woody Repair Shop's webpage! Our website helps clients schedule service appointments and streamlines car repairs. Read the following tutorial to explore the website and use all its features.

**Customer registration and service appointments:**

Our service appointment system requires client registration on our website. Follow these steps:

- Click "Register" on the home page.

- Enter your user details , and vehicle information.

- Click "Submit" to finish your registration.

Log in using your SSN after registering. After logging in, the menu-driven interface lets you order vehicle service. The menu will help you choose a service\_type, vehicle\_type, appointment date, and will display the prices as per serviced opted.

2. Retail Operations:

You can check your order status on our website if you dropped off your vehicle for service. Follow these steps:

Sign in with your SSN.

View your service history by clicking "Service History".

- The option lets you see the list of services you have opted for and their status .

3.Admin operations:  
- Management can change the prices of parts and services at any point of time by logging in as an admin.  
- management can view a list of services and their respective status.

4. Reporting:

To view a report of pending service appointments on a certain day and location, perform these steps:

Click "Admin" in the top right corner of the website.

Select the place and date to view.

Click "Generate Report" to view the report.

Steps to compute revenue by location and service type for a particular time period:

Click "Admin" in the top right corner of the website.

Select the start and end dates for your time period.

Click "Generate Report" to see the revenue calculation.

To examine the top three locations by service revenue for a certain time period, follow these steps:

Click "Admin" in the top right corner of the website.

Select the start and end dates for your time period.

Click "Generate Report" to see the top 3 places.

We hope this user guide simplifies website navigation. Contact us directly with any questions or concerns. We value your business.

**3. SQL Commands of Create Table**

001\_create\_table\_Customer:

CREATE TABLE `customers` (

`cust\_id` varchar(9) NOT NULL,

`name` varchar(255) NOT NULL,

`address` varchar(255) NOT NULL,

`phone\_number` varchar(20) NOT NULL,

`credit\_card` varchar(20) NOT NULL,

PRIMARY KEY (`cust\_id`)

);

002\_create\_table\_Vehicle:

CREATE TABLE `vehicle` (

`vid` varchar(9) NOT NULL,

`manufacturer` varchar(255) NOT NULL,

`model` varchar(255) NOT NULL,

`year` varchar(20) NOT NULL,

`type` varchar(20) NOT NULL,

`color` varchar(20) NOT NULL,

`customer\_id` varchar(9) NOT NULL,

PRIMARY KEY (`vid`),

FOREIGN KEY (`customer\_id`)

REFERENCES `customers` (`cust\_id`)

);

003\_create\_table\_Invoice\_details:

CREATE TABLE `invoice\_details` (

`appoint\_id` int NOT NULL,

`service\_type` varchar(255) NOT NULL,

`vehicle\_type` varchar(255) NOT NULL,

`invoice\_id` varchar(255) NOT NULL,

`price` int NOT NULL,

`appointment\_status` varchar(100) DEFAULT pending ,

PRIMARY KEY (`appoint\_id`,`service\_type`,`vehicle\_type`,`invoice\_id`),

FOREIGN KEY (`appoint\_id`) REFERENCES `appointment` (`appointment\_id`),

FOREIGN KEY (`service\_type`) REFERENCES `services\_offered` (`service\_type`),

FOREIGN KEY (`vehicle\_type`) REFERENCES `services\_offered` (`vehicle\_type`),

FOREIGN KEY (`invoice\_id`) REFERENCES `invoice` (`id`)

);

004\_create\_table\_Parts:

CREATE TABLE `parts` (

`id` int NOT NULL,

`name` varchar(255) NOT NULL,

`price` int NOT NULL,

PRIMARY KEY (`id`)

);

005\_create\_table\_Parts\_required:

CREATE TABLE `parts\_required` (

`parts\_id` int DEFAULT NULL,

`service\_type` varchar(250) DEFAULT NULL,

`vehicle\_type` varchar(255) DEFAULT NULL,

FOREIGN KEY (`parts\_id`) REFERENCES `parts` (`id`),

FOREIGN KEY (`service\_type`) REFERENCES `services\_offered` (`service\_type`),

FOREIGN KEY (`vehicle\_type`) REFERENCES `services\_offered` (`vehicle\_type`)

);

006\_create\_table\_Location:

CREATE TABLE `location` (

`id` int NOT NULL,

`address` varchar(255) NOT NULL,

PRIMARY KEY (`id`)

);

007\_create\_table\_Manager:

CREATE TABLE `manager` (

`ssn` varchar(9) NOT NULL,

`hourly\_rate` int NOT NULL,

PRIMARY KEY (`ssn`),

FOREIGN KEY (`ssn`) REFERENCES `employee` (`ssn`)

);

008\_create\_table\_Technician:

CREATE TABLE `technician` (

`ssn` varchar(9) NOT NULL,

`hourly\_rate` int NOT NULL,

PRIMARY KEY (`ssn`),

FOREIGN KEY (`ssn`) REFERENCES `employee` (`ssn`)

);

009\_create\_table\_Employee:

CREATE TABLE `employee` (

`ssn` varchar(9) NOT NULL,

`name` varchar(255) NOT NULL,

`hired\_date` date NOT NULL,

`role` varchar(255) NOT NULL,

`location\_id` int NOT NULL,

PRIMARY KEY (`ssn`),

FOREIGN KEY (`location\_id`) REFERENCES `location` (`id`)

);

010\_create\_table\_Skills:

CREATE TABLE `skills` (

`id` int NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

PRIMARY KEY (`id`)

);

0011\_create\_table\_Skills\_required:

CREATE TABLE `skills\_required` (

`skills\_id` int NOT NULL,

`technician\_id` int NOT NULL,

FOREIGN KEY (`skills\_id`) REFERENCES `skills` (`id`),

FOREIGN KEY (`technician\_id`) REFERENCES `technician` (`ssn`)

);

**0012\_create\_table\_appointment:**

CREATE TABLE `appointment` (

`appointment\_id` int NOT NULL AUTO\_INCREMENT,

`vid` varchar(9) NOT NULL,

`customer\_id` varchar(9) NOT NULL,

`location\_id` int DEFAULT NULL,

`date` date NOT NULL,

PRIMARY KEY (`appointment\_id`),

FOREIGN KEY (`vid`) REFERENCES `vehicle` (`vid`),

FOREIGN KEY (`customer\_id`) REFERENCES `customers` (`cust\_id`),

FOREIGN KEY (`location\_id`) REFERENCES `location` (`id`)

);

**0013\_create\_Services\_offered:**

**CREATE TABLE `services\_offered` (**

**`service\_type` varchar(255) NOT NULL,**

**`vehicle\_type` varchar(255) NOT NULL,**

**`labour\_price` int DEFAULT NULL,**

**`service\_price` int DEFAULT NULL,**

**`skill\_id` int NOT NULL,**

**PRIMARY KEY (`service\_type`,`vehicle\_type`),**

**FOREIGN KEY (`skills\_id`) REFERENCES `skills` (`id`)**

**)**

**4. SQL Insert Commands Table**

**001\_insert\_table\_location:**

**INSERT INTO `location` (`id`,`address`) VALUES (1,'123 Main Street');**

**INSERT INTO `location` (`id`,`address`) VALUES (2,'456 Park Avenue');**

**INSERT INTO `location` (`id`,`address`) VALUES (3,'789 Oak Street');**

**INSERT INTO `location` (`id`,`address`) VALUES (4,'10 Maple Avenue');**

**INSERT INTO `location` (`id`,`address`) VALUES (5,'555 Elm Street');**

**002\_insert\_table\_employee:**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('012345678','Mike Jones','2022-01-10','oil changer',5);**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('234567890','David Lee','2022-01-12','car mechanic',3);**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('345678901','Lisa Davis','2022-01-13','car mechanic',1);**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('456789012','Steve Wilson','2022-01-14','car mechanic',2);**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('567890123','Michelle Lee','2022-01-15','car mechanic',3);**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('678901234','Alex Chen','2022-01-16','car mechanic',5);**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('789012345','Katie Smith','2022-01-17','car mechanic',1);**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('890123456','Brian Jones','2022-01-18','car mechanic',4);**

**INSERT INTO `employee` (`ssn`,`name`,`hired\_date`,`role`,`location\_id`) VALUES ('901234567','Emily Green','2022-01-09','sedan engineer',4);**

**003\_insert\_table\_manager:**

INSERT INTO `manager` (`ssn`,`hourly\_rate`) VALUES ('998676754',100);

**004\_insert\_table\_parts:**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (1,'Oil Filter',14);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (2,'Brake Pads',50);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (3,'Spark Plugs',8);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (4,'Air Filter',20);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (5,'Radiator Hose',25);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (6,'Serpentine Belt',30);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (7,'Fuel Filter',15);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (8,'Wiper Blades',20);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (9,'Battery',100);**

**INSERT INTO `parts` (`id`,`name`,`price`) VALUES (10,'Headlight Bulbs',8);**

**005\_insert\_table\_parts\_required:**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (8,'Brakes','sports car');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (5,'Brakes','SUV');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (8,'Brakes','truck');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (2,'Engine tune-ups','sedan');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (9,'Engine tune-ups','sports car');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (6,'Engine tune-ups','SUV');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (9,'Engine tune-ups','truck');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (3,'Front-end alignments','sedan');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (10,'Front-end alignments','sports car');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (7,'Front-end alignments','SUV');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (10,'Front-end alignments','truck');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (4,'Oil changes','sedan');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (1,'Oil changes','sports car');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (8,'Oil changes','SUV');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (1,'Oil changes','truck');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (5,'State vehicle inspections','sedan');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (2,'State vehicle inspections','sports car');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (9,'State vehicle inspections','SUV');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (2,'State vehicle inspections','truck');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (6,'Tire repairs and replacements','sedan');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (3,'Tire repairs and replacements','sports car');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (10,'Tire repairs and replacements','SUV');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (3,'Tire repairs and replacements','truck');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (7,'Vehicle computer diagnostics','sedan');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (4,'Vehicle computer diagnostics','sports car');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (1,'Vehicle computer diagnostics','SUV');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (4,'Vehicle computer diagnostics','truck');**

**INSERT INTO `parts\_required` (`parts\_id`,`service\_type`,`vehicle\_type`) VALUES (1,'Brakes','sedan');**

**006\_insert\_table\_service\_offered:**

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Brakes','sedan',95,95,3);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Brakes','sedan',95,95,3);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Brakes','sports car',120,200,3);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Brakes','SUV',100,170,3);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Brakes','truck',140,220,3);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Engine tune-ups','sedan',100,200,5);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Engine tune-ups','sports car',150,250,5);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Engine tune-ups','SUV',120,220,5);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Engine tune-ups','truck',180,300,5);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Front-end alignments','sedan',75,110,2);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Front-end alignments','sports car',95,150,2);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Front-end alignments','SUV',85,130,2);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Front-end alignments','truck',110,170,2);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Oil changes','sedan',50,75,1);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Oil changes','sports car',70,100,1);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Oil changes','SUV',60,85,1);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Oil changes','truck',80,120,1);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('State vehicle inspections','sedan',40,60,7);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('State vehicle inspections','sports car',60,90,7);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('State vehicle inspections','SUV',50,70,7);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('State vehicle inspections','truck',80,120,7);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Tire repairs and replacements','sedan',60,100,4);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Tire repairs and replacements','sports car',90,150,4);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Tire repairs and replacements','SUV',75,120,4);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Tire repairs and replacements','truck',110,180,4);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Vehicle computer diagnostics','sedan',125,25,6);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Vehicle computer diagnostics','sports car',180,320,6);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Vehicle computer diagnostics','SUV',150,280,6);

INSERT INTO `service\_offered` (`service\_type`,`vehicle\_type`,`labour\_price`,`service\_price`,`skill\_id`) VALUES ('Vehicle computer diagnostics','truck',220,380,6);

**007\_insert\_table\_skills\_required:**

**insert into skills\_required values**

**(1,012345678),**

**(2,123456789),**

**(3,234567890),**

**(4,345678901),**

**(5,456789012),**

**(6,567890123),**

**(7,678901234),**

**(1,789012345),**

**(2,890123456),**

**(3,901234567);**

**5. Source Code   
  
Backend code:**

**const express = require("express");**

**const cors = require("cors");**

**const app = express();**

**const axios = require("axios");**

**app.use(cors());**

**app.use(express.json());**

**var mysql = require("mysql");**

**app.post("/login", (req, res) => {**

**const sql = "SELECT \* FROM customers where cust\_id = ?";**

**const { ssn } = req.body;**

**let value = [ssn];**

**con.connect(function (err) {**

**con.query(sql, [value], function (err, result) {**

**// console.log(err)**

**if (result.length !== 0) {**

**res.send("please click vehicle registration");**

**} else {**

**res.send("user not existed");**

**}**

**});**

**});**

**});**

**app.post("/register", (req, res) => {**

**const { name, cust\_id, address, phone\_number, credit\_card } = req.body;**

**const sql = "INSERT INTO customers VALUES (?)";**

**let value = [cust\_id, name, address, phone\_number, credit\_card];**

**con.connect(function (err) {**

**con.query(sql, [value], function (err, result) {**

**console.log(err);**

**console.log(result);**

**});**

**});**

**res.send("response saved");**

**});**

**app.post("/edit\_customer", (req, res) => {**

**const { ssn } = req.body;**

**const sql = "select \* from customers where cust\_id = ?";**

**let value = [ssn];**

**con.connect(function (err) {**

**con.query(sql, [value], function (err, result) {**

**console.log(result);**

**res.send(result);**

**});**

**});**

**});**

**app.post("/vehicleRegistration", (req, res) => {**

**const {**

**vin,**

**manufacturer,**

**model,**

**year,**

**type,**

**color,**

**ssn: customer\_id,**

**} = req.body;**

**console.log(req.body);**

**const sql = "INSERT INTO vehicle VALUES (?)";**

**let value = [vin, manufacturer, model, year, type, color, customer\_id];**

**con.connect(function (err) {**

**con.query(sql, [value], function (err, result) {**

**console.log(err);**

**console.log(result);**

**});**

**});**

**res.send("response saved");**

**});**

**app.post("/appointment", (req, res) => {**

**let lId = "";**

**let totalPrice = 0;**

**const { vid, ssn: customer\_id, date, location, selectedService } = req.body;**

**console.log(req.body, "------------------------------------------");**

**const sql =**

**"INSERT INTO appointment (vid, customer\_id,date, location\_id) VALUES (?)";**

**const location\_id = "SELECT id FROM location where address = ?";**

**const getAppointmentId =**

**"SELECT appointment\_id FROM appointment WHERE customer\_id = ?";**

**const insertInvoiceRequired =**

**"INSERT INTO invoice\_details (appoint\_id, service\_type,vehicle\_type, invoice\_id,price) VALUES (?)";**

**const sum =**

**"SELECT SUM(price) AS price FROM parts WHERE id in (SELECT parts\_id FROM parts\_required WHERE service\_type = ? AND vehicle\_type = ?)";**

**const insertInvoice = "INSERT INTO invoice VALUES(?)";**

**const finalBillDertails =**

**"SELECT id.invoice\_id ,id.service\_type,id.vehicle\_type, id.price AS 'specific price', i.amount AS 'Total Price',i.payment\_date AS 'Date of payment'FROM invoice\_details AS id JOIN invoice AS i ON i.id = id.invoice\_id WHERE id.invoice\_id = ?";**

**con.connect(function (err) {**

**con.query(location\_id, [[location]], function (err, result) {**

**if (err) {**

**console.log(err);**

**} else {**

**lId = result[0].id;**

**let value = [vid, customer\_id, date, lId];**

**con.connect(function (err) {**

**con.query(sql, [value], function (err, result) {**

**if (err) {**

**console.log(err);**

**}**

**});**

**});**

**selectedService.map((element, index) => {**

**let price = 0;**

**price = price + element.labour\_price + element.service\_price;**

**totalPrice = totalPrice + price;**

**con.connect(function (err) {**

**con.query(**

**sum,**

**[[element.service\_type], [element.vehicle\_type]],**

**function (err, result) {**

**if (err) {**

**console.log(err);**

**} else {**

**price = price + result[0].price;**

**totalPrice = totalPrice + result[0].price;**

**con.connect(function (err) {**

**con.query(**

**getAppointmentId,**

**[[customer\_id]],**

**function (err, result) {**

**if (err) {**

**console.log(err);**

**} else {**

**let value = [**

**result[result.length - 1].appointment\_id,**

**element.service\_type,**

**element.vehicle\_type,**

**element.invoice\_id,**

**price,**

**];**

**con.connect(function (err) {**

**con.query(**

**insertInvoiceRequired,**

**[value],**

**function (err, result) {**

**if (err) {**

**console.log(err);**

**} else {**

**value = [**

**element.invoice\_id,**

**totalPrice,**

**date,**

**];**

**if (index === 0) {**

**con.connect(function (err) {**

**con.query(**

**insertInvoice,**

**[value],**

**function (err, result) {**

**if (err) {**

**console.log(err);**

**}**

**con.connect(function (err) {**

**con.query(**

**finalBillDertails,**

**[[selectedService[0].invoice\_id]],**

**function (err, result) {**

**if (err) {**

**console.log(err);**

**} else {**

**console.log(result);**

**res.send({**

**data: result,**

**message: "Response saved",**

**});**

**}**

**}**

**);**

**});**

**}**

**);**

**});**

**}**

**}**

**}**

**);**

**});**

**}**

**}**

**);**

**});**

**}**

**}**

**);**

**});**

**});**

**}**

**});**

**});**

**});**

**app.get("/getPriceDetails", (req, res) => {**

**const sql =**

**"SELECT so.service\_type AS 'SERVICE TYPE', so.vehicle\_type AS 'VEHICLE TYPE',labour\_price AS 'LABOUR PRICE', service\_price AS 'SERVICE PRICE' ,p.name AS 'Part Name',p.price AS 'Part Price' FROM services\_offered AS so join parts\_required AS pr ON pr.service\_type = so.service\_type AND pr.vehicle\_type = so.vehicle\_type join parts AS p ON pr.parts\_id = p.id";**

**con.connect(function (err) {**

**con.query(sql, function (err, result) {**

**if (err) console.log(err);**

**res.send(result);**

**});**

**});**

**});**

**app.post("/invoices", (req, res) => {**

**console.log(req.body.cust\_id);**

**const sql =**

**"select i.id,i.amount , id.appointment\_status from invoice as i join invoice\_details as id on id.invoice\_id = i.id where appoint\_id in (select appointment\_id from appointment where customer\_id = ? )";**

**con.connect(function (err) {**

**con.query(sql, [[req.body.cust\_id]], function (err, result) {**

**if (err) console.log(err);**

**res.send(result);**

**});**

**});**

**});**

**app.get("/getServicesData", (req, res) => {**

**const sql = "SELECT \* FROM services\_offered";**

**con.connect(function (err) {**

**con.query(sql, function (err, result) {**

**if (err) console.log(err);**

**res.send(result);**

**});**

**});**

**});**

**app.get("/getLocation", (req, res) => {**

**const sql = "SELECT \* FROM location;";**

**con.connect(function (err) {**

**con.query(sql, function (err, result) {**

**if (err) console.log(err);**

**res.send(result);**

**});**

**});**

**});**

**app.post("/updatePrices", (req, res) => {**

**const sql =**

**"UPDATE services\_offered set labour\_price = ? , service\_price=? WHERE service\_type = ? and vehicle\_type = ?";**

**const parts =**

**"UPDATE parts SET price = ? WHERE id in (SELECT parts\_id FROM parts\_required WHERE service\_type = ? and vehicle\_type = ? )";**

**req.body.updatedData.forEach((element) => {**

**con.connect(function (err) {**

**con.query(**

**sql,**

**[**

**element["LABOUR PRICE"],**

**element["SERVICE PRICE"],**

**element["SERVICE TYPE"],**

**element["VEHICLE TYPE"],**

**],**

**function (err, result) {**

**if (err) console.log(err);**

**}**

**);**

**});**

**con.connect(function (err) {**

**con.query(**

**parts,**

**[**

**element["Part Price"],**

**element["SERVICE TYPE"],**

**element["VEHICLE TYPE"],**

**],**

**function (err, result) {**

**if (err) console.log(err);**

**}**

**);**

**});**

**res.send("Updated the changes");**

**});**

**});**

**app.post("/getStatusInAdminPage", (req, res) => {**

**const sql =**

**"select id.service\_type, id.vehicle\_type , id.invoice\_id , id.price ,id.appointment\_status from appointment as a inner join invoice\_details as id on a.appointment\_id = id.appoint\_id inner join invoice as i on i.id = id.invoice\_id where a.location\_id in (select id from location where address = ? )and i.payment\_date = ?";**

**con.connect(function (err) {**

**con.query(**

**sql,**

**[[req.body.date], [req.body.address]],**

**function (err, result) {**

**if (err) console.log(err);**

**console.log(result);**

**res.send(result);**

**}**

**);**

**});**

**});**

**app.post("/postSaveChanges", (req, res) => {**

**const sql =**

**"update invoice\_details set appointment\_status = ? where service\_type = ? and vehicle\_type = ? and invoice\_id = ?;";**

**req.body.data.forEach((item) => {**

**con.connect(function (err) {**

**con.query(**

**sql,**

**[**

**[item.appointment\_status],**

**[item.service\_type],**

**[item.vehicle\_type],**

**[item.invoice\_id],**

**],**

**function (err, result) {**

**if (err) console.log(err);**

**else {**

**console.log(result);**

**}**

**}**

**);**

**});**

**});**

**res.send("response saved properly");**

**});**

**app.post("/update", (req, res) => {**

**let { name, cust\_id, address, phone\_number, credit\_card } = req.body;**

**let value = [cust\_id, name, address, phone\_number, credit\_card, cust\_id];**

**const sql =**

**"update customers set cust\_id = ? , name = ? , address = ? , phone\_number =? , credit\_card =? where cust\_id =?";**

**con.connect(function (err) {**

**con.query(**

**sql,**

**[[cust\_id], [name], [address], [phone\_number], [credit\_card], [cust\_id]],**

**function (err, result) {**

**if (err) console.log(err);**

**res.send("successfully edited");**

**}**

**);**

**});**

**});**

**var con = mysql.createConnection({**

**host: "localhost",**

**user: "root",**

**password: "password",**

**database: "woody",**

**multipleStatements: true,**

**});**

**app.listen(8000, () => {**

**console.log(`Server is running on port 8080.`);**

**});**

**app.post("/getReports", (req, res) => {**

**const sql =**

**'select a.appointment\_id ,a.customer\_id, id.appointment\_status from appointment as a join invoice\_details as id on id.appoint\_id = a.appointment\_id where id.appointment\_status = "pending" and a.date = ? and a.location\_id in (select id from location where address = ?)';**

**con.connect(function (err) {**

**con.query(**

**sql,**

**[[req.body.date], [req.body.address]],**

**function (err, result) {**

**if (err) console.log(err);**

**console.log(result);**

**res.send(result);**

**}**

**);**

**});**

**});**

**app.post("/getReports/2", (req, res) => {**

**const sql =**

**`select l.address as 'Location' ,id.service\_type as 'Service Type', sum(i.amount) as 'total amount' from invoice\_details as id join appointment as a on id.appoint\_id = a.appointment\_id join location as l on l.id = a.location\_id join invoice as i on i.id = id.invoice\_id where a.date between ? and ? group by l.address,id.service\_type`;**

**con.connect(function (err) {**

**con.query(**

**sql,**

**[[req.body.date], [req.body.endDate]],**

**function (err, result) {**

**if (err) console.log(err);**

**console.log(result);**

**res.send(result);**

**}**

**);**

**});**

**});**

**app.post("/getReports/3", (req, res) => {**

**const sql =**

**`select l.address as 'Address' , sum(i.amount) as' total revenue'**

**from invoice\_details as id**

**join invoice as i on i.id = id.invoice\_id**

**join appointment as a on a.appointment\_id = id.appoint\_id**

**join location as l on l.id = a.location\_id**

**where a.date between ? and ?**

**group by l.address**

**order by sum(i.amount) desc**

**limit 3`;**

**con.connect(function (err) {**

**con.query(**

**sql,**

**[[req.body.date], [req.body.endDate]],**

**function (err, result) {**

**if (err) console.log(err);**

**console.log(result);**

**res.send(result);**

**}**

**);**

**});**

**});**

***Frontend code: will be uploaded as a folder in the submission.***

**Queries for Manager reporting module  
  
  
  
1.For a given day and location, a report of the pending service**

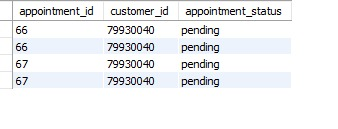
**appointments.**

select a.appointment\_id ,a.customer\_id,a.vid,id.service\_type, id.appointment\_status

from appointment as a

join invoice\_details as id on id.appoint\_id = a.appointment\_id

where id.appointment\_status = "pending" and a.date = "2023-03-28" and a.location\_id in (select id from location where address = "10 Maple Avenue");



**2. For a given time period (begin date and end date) compute revenue by location and service type.**

select l.address as 'Location' ,id.service\_type as 'Service Type', sum(i.amount) as 'total amount'

from invoice\_details as id

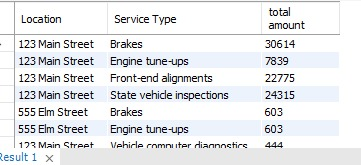
join appointment as a on id.appoint\_id = a.appointment\_id

join location as l on l.id = a.location\_id

join invoice as i on i.id = id.invoice\_id

where a.date between "2023-03-29" and "2023-04-28"

group by l.address,id.service\_type

****

**3. For a given time period (begin date and end date) compute the top 3 locations (in terms of total service revenue) in descending order.**

select l.address as 'Address' , sum(i.amount) as' total revenue'

from invoice\_details as id

join invoice as i on i.id = id.invoice\_id

join appointment as a on a.appointment\_id = id.appoint\_id

join location as l on l.id = a.location\_id

where a.date between "1997-03-29" and "2023-05-30"

group by l.address

order by sum(i.amount) desc

limit 3

****

**Project Screenshots**