Group 3

Final Report

CS 353 - 02

NovaCars

Authors:

Ani Kristo

Burcu Çanakcı

Onatkut Dağtekin

Yasemin Doğancı

Contents

List of Figures

1. Brief Description of the Project

Our project provides a workspace for or an auto-service and it is called Nova Cars. This system is formed in order to keep track of the data which is about the departments, employees, suppliers and customers and their history of car repairs and maintenance. With our system, keeping track of the sales, transactions and operations are done by a database, which also provides reports on various statistics.

Different users can perform different tasks in the database. The users can either be the employees of the auto service and/or customers.

Auto service is made of multiple departments and each department has employees. The service departments provide the operations which can be made with the autos the customers own.

Employees can either be a clerk, a sales manager, a technician or a manager. Each employee works in a specific department. Clerks can complete transactions which are made with the customers with operations. Sales managers can keep track of the sales with the suppliers and they perform transactions by buying spare parts. Technicians can perform operations on customers' autos.

Suppliers supply different spare parts for the auto-service and customers own autos.

As a whole, our system is an integrated tool to deal with every aspect of an auto-service and keeps track of the customer, employee and supplier information during any operation or transaction process.

2. Final E/R Model

Upon receiving feedback on the E/R model presented in the Design Report, we made the following changes to the diagram. The *with_c* relationship is four ways, therefore, we are constrained to use only one arrow head (single cardinality) for disambiguation. Each transaction has a unique id that is why the transaction follows single cardinality. Additionally we added a new relationship called *uses*, which describes which spare part is used for which operation.

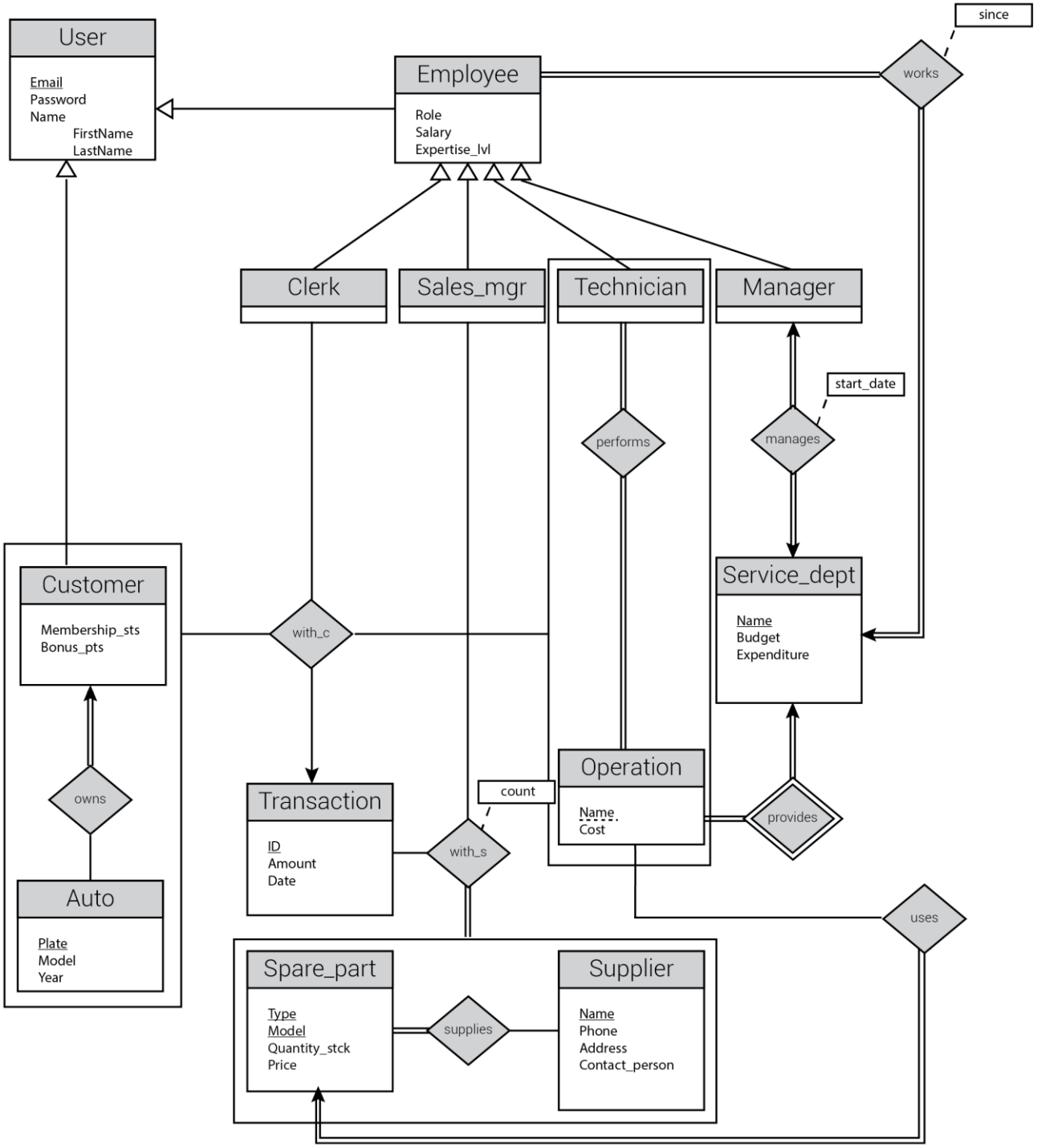


Figure 1 - E/R Diagram for NovaCars system

3. Final List of Tables

3.1. User & Customer

3.1.1.User

Relational Model: User (email, password, first_name, last_name)

Primary Keys: email

3.1.2.Customer

Relational Model: Customer (email, membership_sts, bonus_pts)

Primary Keys: email

Foreign Keys: email references User.email

3.1.3.Auto

Relational Model: Auto (plate, model, year, customer_email)

Primary Keys: plate

Foreign Keys: customer_email references Customer.email

3.2. Department & Employees

3.2.1.Department

Relational Model: Department (name, budget, expenditure)

Primary Keys: name

3.2.2.Employee

Relational Model: Employee (email, salary, role, expertise_lvl, dept_name, since)

Primary Keys: email

Foreign Keys: email references User.email, dept_name references Department.name

3.2.3.Clerk

Relational Model: Clerk (email)

Primary Keys: email

Foreign Keys: email references Employee.email

3.2.4.SalesManager

Relational Model: SalesManager (email)

Primary Keys: email

Foreign Keys: email references Employee.email

3.2.5.Technician

Relational Model: Technician (email)

Primary Keys: email

Foreign Keys: email references Employee.email

3.2.6.Manager

Relational Model: Manager (email, start_date)

Primary Keys: email

Foreign Keys: email references Employee.email

3.3. Suppliers & Spare Parts

3.3.1.Supplier

Relational Model: Supplier (name, phone, address, contact_name)

Primary Keys: name

3.3.2.SparePart

Relational Model: SparePart (type, model, stock_quantity, price)

Primary Keys: type, model

3.3.3.Supply

Relational Model: Supply (type, model, supplier_name)

Primary Keys: type, model, supplier_name

Foreign Keys: type references SparePart.type, model references SparePart.model, supplier_name references Supplier.name

3.4. Employee Operations & Transactions

3.4.1.Operation

Relational Model: Operation (dept_name, op_name, cost, sparepart_type, sparepart_model)

Primary Keys: dept_name, op_name

Foreign Keys: dept_name references Department.name, sparepart_type references SparePart.type, sparepart_model references SparePart.model

3.4.2.Transaction

Relational Model: Transaction (id, amount, date)

Primary Keys: id

3.4.3.TechnicianOperation

Relational Model: TechnicianOperation (dept_name, op_name, tech_email)

Primary Keys: dept_name, op_name, tech_email

Foreign Keys: tech_email references Technician.email, dept_name references Operation.dept_name, op_name references Operation.op_name

3.4.4.SparePartOrder

Relational Model: SparePartOrder (transaction_id, sales_mgr_email, part_type, part_model, supplier_name, count)

Primary Keys: transaction_id

Foreign Keys: transaction_id references Transaction.id, sales_mgr_email references SalesManager.email, part_type references Supply.type, part_mode references Supply.model, supplier_name references Supply.supplier_name

3.4.5.CustomerOperation

Relational Model: CustomerOperation (transaction_id, dept_name, op_name, tech_email, clerk_email, customer_email, auto_plate)

Primary Keys: transaction_id

Foreign Keys: transaction_id references Transaction.id, dept_name references TechnicianOperation.dept_name, op_name references TechnicianOperation.op_name, tech_email references TechnicianOperation.tech_email, clerk_email references Clerk.email, customer_email references Auto.customer_email. auto_plate references Auto.plate

4. Implementation Details

We used MySQL to manage our databases and we used INNODB as our database engine.

For back-end development we used PHP.

For front-end development we used HTML, jQuery JavaScript and Twitter Bootstrap CSS Framework.

5. Advanced Database Features

5.1. Secondary Indexes

In the Department Info page which is displayed to the Manager, the list of Employees working in that particular department and managed by the current logged in user, are displayed. Using the Filter option provided at the top of the list, the user can search for a particular Employee using the first name or last name attributes. Therefore, we need to have a secondary index on these attributes in order to speed up the search process.

The creation of such indexes is done by the following SQL statements:

CREATE INDEX user_first_name_index ON User(first_name(20));

CREATE INDEX user_last_name_index ON User(last_name(20));

Upon the query execution the following output was given by the mysql program:

```
mysql> create index user_first_name_index ON User(first_name(20));
Query OK, 0 rows affected (0.26 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

In order to display the indexes on the User table, the SHOW INDEX FROM User command was executed and the following is the list of all indexes on this table. Note that the Primary index was automatically created upon the table creation.

```
mysql> show index from User;
+-------+------------+-----------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+
| Table | Non_unique | Key_name              | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type |
+-------+------------+-----------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+
| User  |          0 | PRIMARY               |            1 | email       | A         |           1 |     NULL | NULL   |      | BTREE      |
| User  |          1 | user_first_name_index |            1 | first_name  | A         |           1 |     NULL | NULL   |      | BTREE      |
| User  |          1 | user_last_name_index  |            1 | last_name   | A         |           1 |     NULL | NULL   |      | BTREE      |
+-------+------------+-----------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+
3 rows in set (0.00 sec)

Query OK, 0 rows affected (0.18 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

Note that, in other filtering options that we provide in different screens, the search can be done by Operation Name, Department Name and so on. However, since these are Primary Keys, creating a secondary index is redundant and incorrect.

5.2. Advanced Features

5.2.1. Views

List of Customers

CREATE VIEW list-of-customers AS

SELECT u.email, u.first_name, u.last_name

FROM (User u NATURAL JOIN Customer c)

List of Employees

```
CREATE VIEW list-of-employees AS
SELECT u.email, u.first_name, u.last_name, e.dept_name
FROM (User u NATURAL JOIN Employee e)
```

5.2.2. Triggers

Updating Spare Part Count on Transaction with Suppliers

When a transaction between a supplier and sales manager occurs quantity of that certain spare part will be increased.

```
CREATE TRIGGER update-spare-parts AFTER INSERT ON SparePartOrder
REFERENCING NEW ROW AS nrow
FOR EACH ROW
WHEN nrow.count > 0
BEGIN
        UPDATE SparePart SET stock_quantity = stock_quantity + nrow.count
        WHERE type == nrow.part_type AND model == nrow.part_model;
END;
```

Updating Spare Part Count on Transaction with Customers

When a transaction between a customer and clerk occurs quantity of that certain spare part might be decreased depending on operation.

```
CREATE TRIGGER remove-spare-parts AFTER INSERT ON CustomerOperation
REFERENCING NEW ROW AS nrow
FOR EACH ROW
WHEN (SELECT sparepart_type FROM Operation O WHERE O.dept_name ==
nrow.dept_name AND O.op_name == nrow.op_name) IS NOT NULL
BEGIN
        UPDATE SparePart SET stock_quantity = stock_quantity – 1
        WHERE type == (SELECT sparepart_type FROM Operation
                              WHERE  O.dept_name == nrow.dept_name AND
                               O.op_name == nrow.op_name) AND
                  model == (SELECT sparepart_model FROM Operation
                              WHERE  O.dept_name == nrow.dept_name AND
                               O.op_name == nrow.op_name);

END;
```

5.3. Reports

Total Money Spent by Customers

```
SELECT SUM(T.amount)
FROM Transaction T, CustomerOperation CO
WHERE T.id == CO.transaction_id;
GROUP BY CO.customer_email;
```

Number of Staff in Each Department

```
SELECT COUNT(*)
FROM Department D, Employee E
WHERE D.Name == E.dept_name
GROUP BY D.Name;
```

Customers with the Most Expensive Purchases in the Defined Time Interval

```
WITH cust_purchases AS (
    SELECT C.email, SUM(T.amount) AS total_purchases
    FROM Customer C, CustomerOperation CO, Transaction T
    WHERE C.email == CO.customer_email AND CO.transaction_id == T.id
    AND
    T.date BETWEEN @start_date AND @end_date
    GROUP BY C.email
)
SELECT U.first_name, U.last_name
FROM Customer C, User U
WHERE C.email == U.email
AND
C.email IN (
    SELECT CP.email
    FROM cust_purchases CP
    HAVING CP.total_purchases == MAX(CP.total_purchases)
);
```

The Clerk that Has Made Most Sales in the Defined Time Interval

```
WITH clerk_sales AS (
    SELECT C.email, SUM(T.amount) AS total_sales
    FROM Clerk C, CustomerOperation CO, Transaction T
    WHERE C.email == CO.clerk_email AND CO.transaction_id == T.id
    AND
    T.date BETWEEN @start_date AND @end_date
    GROUP BY C.email
)
SELECT U.first_name, U.last_name
FROM User U, Clerk C
WHERE U.email == C.email
AND
C.email IN (
    SELECT CS.email
    FROM clerk_sales CS
    HAVING CS.total_sales == MAX(CS.total_sales)
);
```

Technician that Worked the Most in a Time Interval

```
WITH technician_work AS (
    SELECT  co.tech_email, count(transaction_id) as work_count
    FROM CustomerOperation co, Transaction t
    WHERE t.date BETWEEN @start_date AND @end_date
    AND co.transaction_id  = t.id
    GROUP BY co.tech_email)
SELECT tw.tech_email
FROM technician_work tw
WHERE tw.work_count >= ALL
    (SELECT tw2.work_count
    FROM  technician_work tw2

);
```

Spare Part which Was Sold the Most

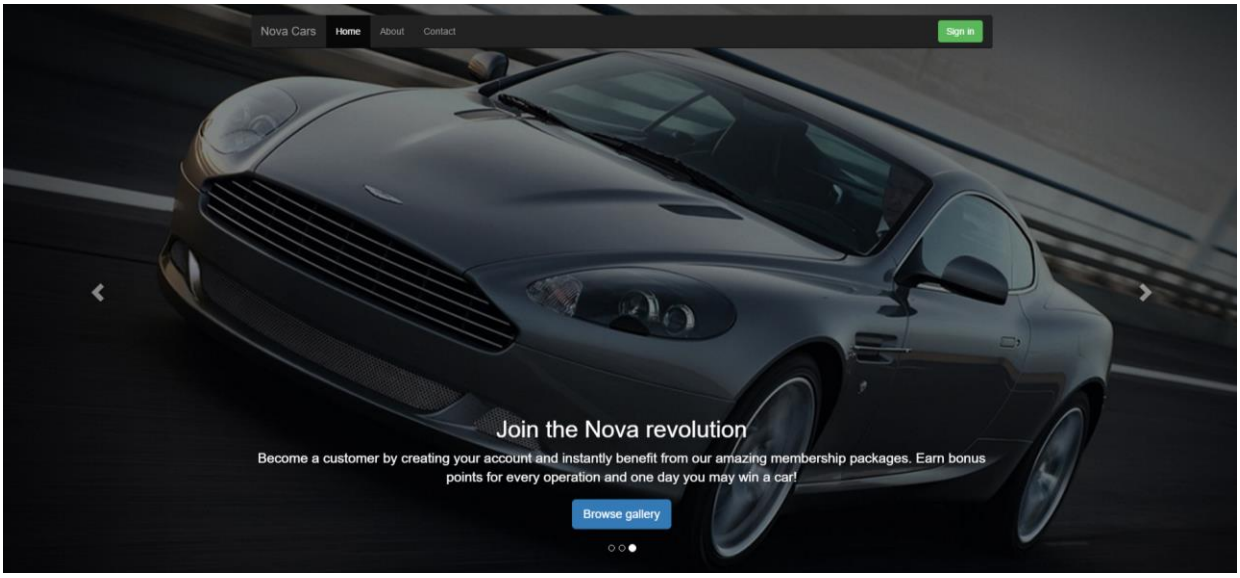
```
WITH sparepart_count AS(
    SELECT SP.type, SP.model, COUNT(DISTINCT SP.type) AS cnt
    FROM CustomerOperation CO, Operation O, SparePart SP,
    WHERE CO.dept_name== O.dept_name AND CO.op_name == O.op_name AND
O.sparepart_type == SP.type AND O.sparepart_model == SP.model
    GROUP BY SP.type, SP.model)
SELECT SP.type, SP.model
FROM SparePart SP, sparepart_count SC
WHERE SP.type == SC.type AND SP.model == SC.model
GROUP BY SP.type, SP.model
HAVING SC.cnt == MAX(SC.cnt);
```

6. User's Manual

6.1. Manual for Customer

Welcome Page:

This is the start of the website. It gives various information about the company and the operations done by the company. It also gives detailed information about the products supplied by the company. There is also a sign in link at the top right of the website.



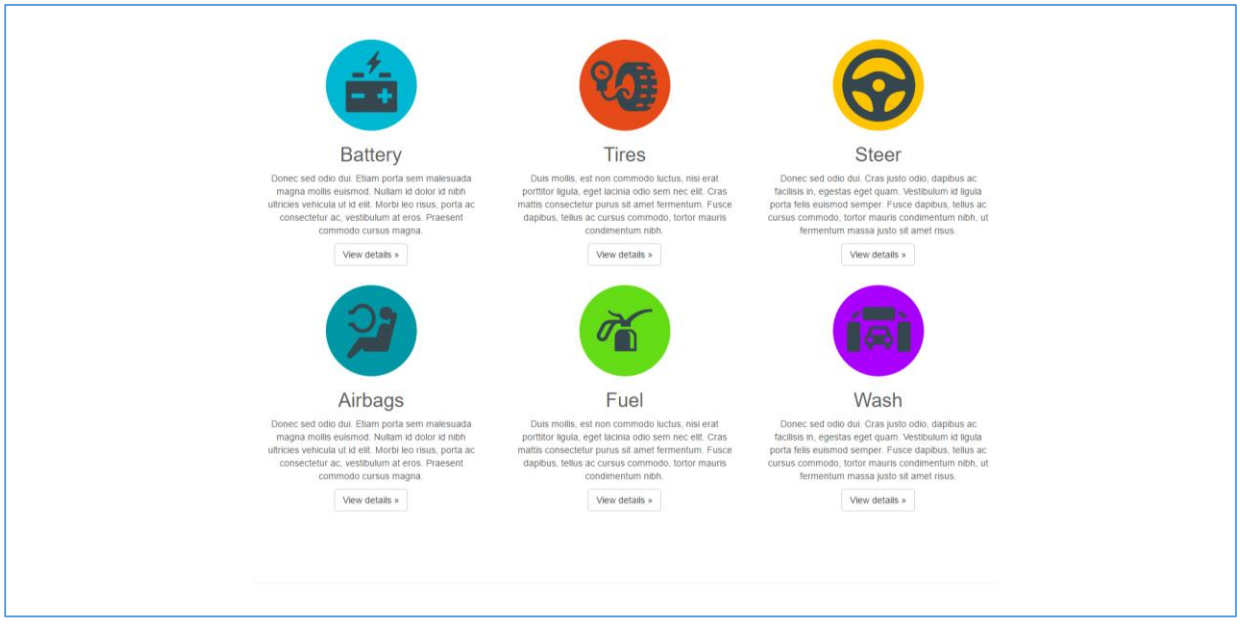


Figure 2 Welcome Page

Sign In Page:

This page asks the user for the credentials for logging into the website which is email and password. If the user does not have an account he/she can click on the create account button and access the register page.

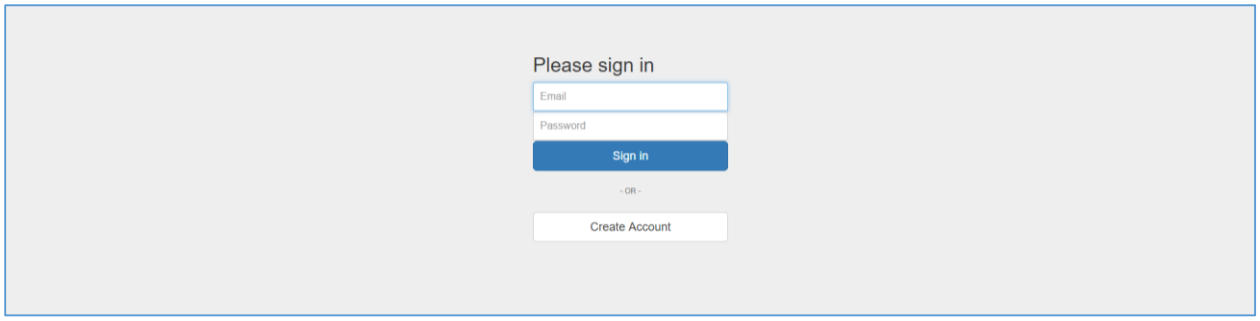


Figure 3 Sign In Page

Register Page:

The user is directed to this page if he/she does not have an account. At this page, the user is asked to enter his/her first and last name, email and a password to create an account.

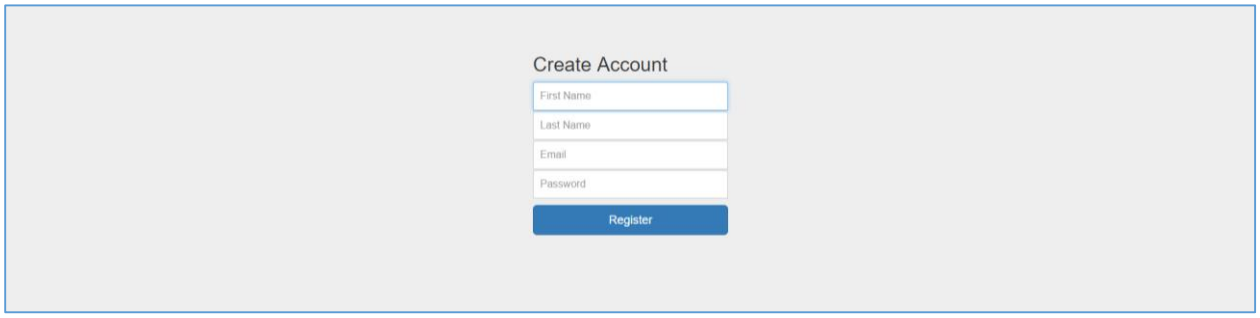


Figure 4 Register Page

Profile Page:

This page gives information about the customer, customer's owned vehicles and operations done on the customer's vehicles. The user can also add more vehicles and change his/her password by clicking the buttons associated with those actions.

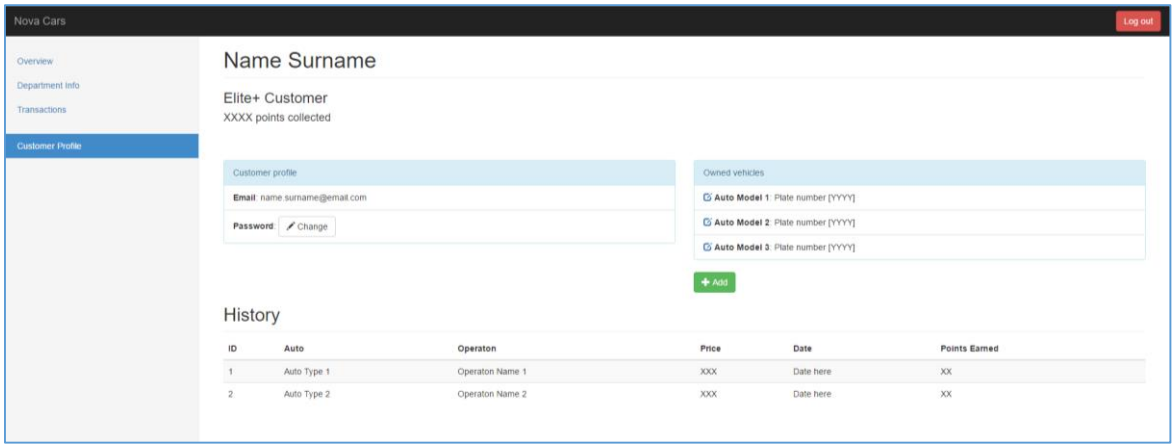


Figure 5 Profile Page

6.2. Manual for Manager

Manager can do what a customer can do except for a different profile page (renamed overview) and have more access than a customer.

Overview Page:

This page shows information about the manager. It also has a button for changing user password.

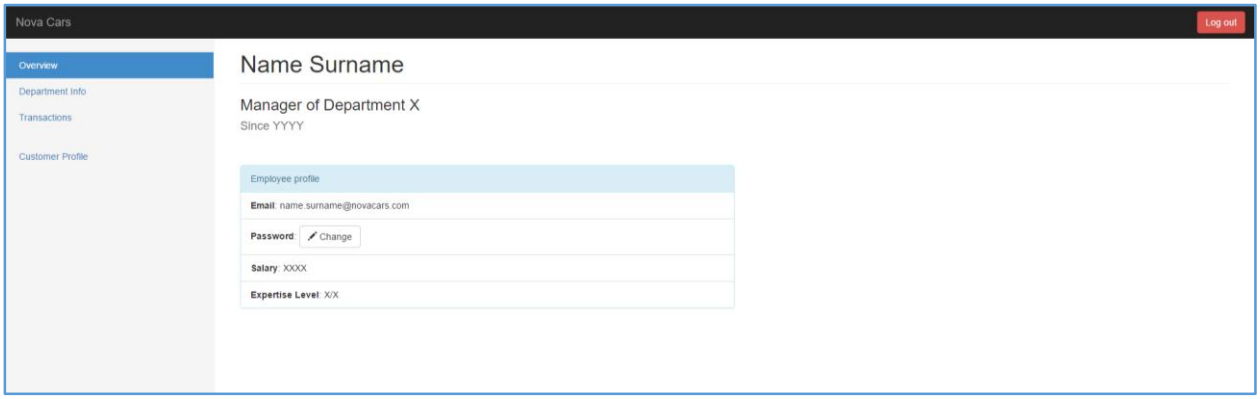


Figure 6 Manager Overview Page

Department Page:

The manager can access this page using the dashboard on the left side of any page. It shows information about the department, the operations it provides and the employees of it. The employees can be filtered by name. New employees can be added to this list.

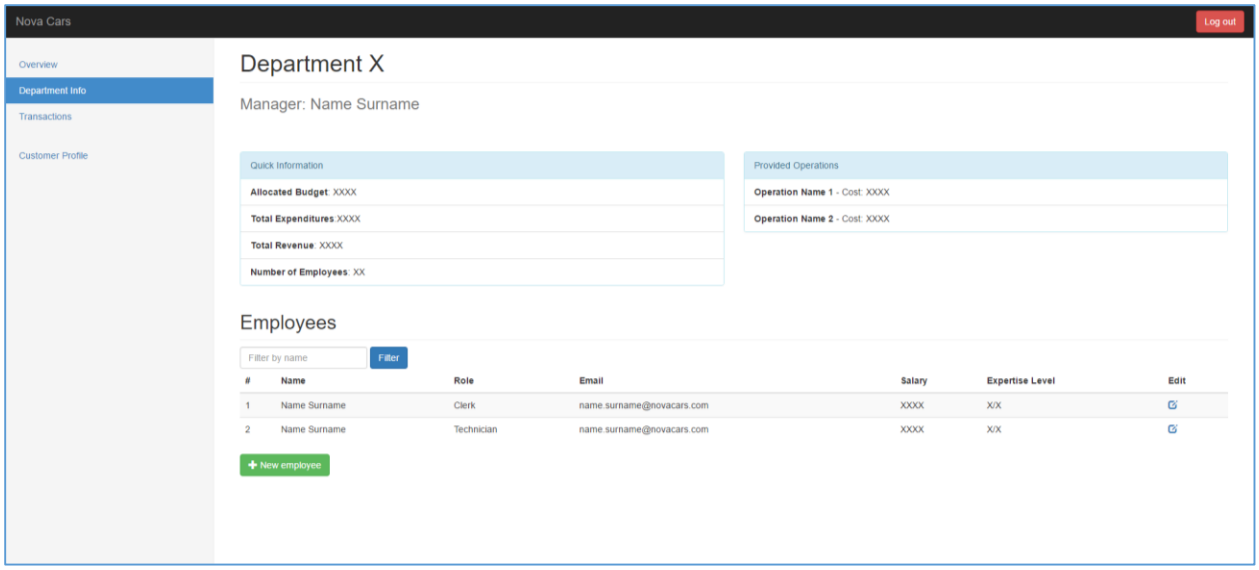


Figure 7 Department Page

Transactions Page:

The manager can access this page using the dashboard on the left side of any page. The page gives information about the customer transaction of the manager's department. The manager can see further information by clicking the button under the details column for each transaction.

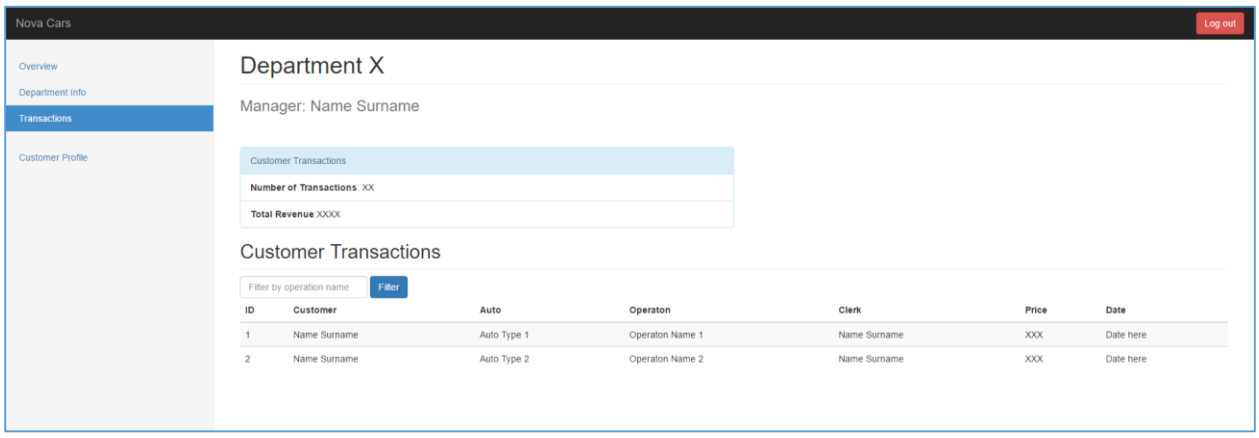


Figure 8 Transactions Page

6.3. Manual for Sales Manager

Sales manager can do what a customer can do except for a different profile page (renamed overview) and have more access than a customer.

Overview Page:

This page shows information about the sales manager. It also has a button for changing user password. This is almost the same as the Manager's overview page.

Suppliers Info Page:

The sales manager can access this page using the dashboard on the left side of any page. This page shows the information about suppliers and spare parts with tables. The sales manager can add new spare parts and/or suppliers to these lists (which redirects the sales manager to new transaction page).

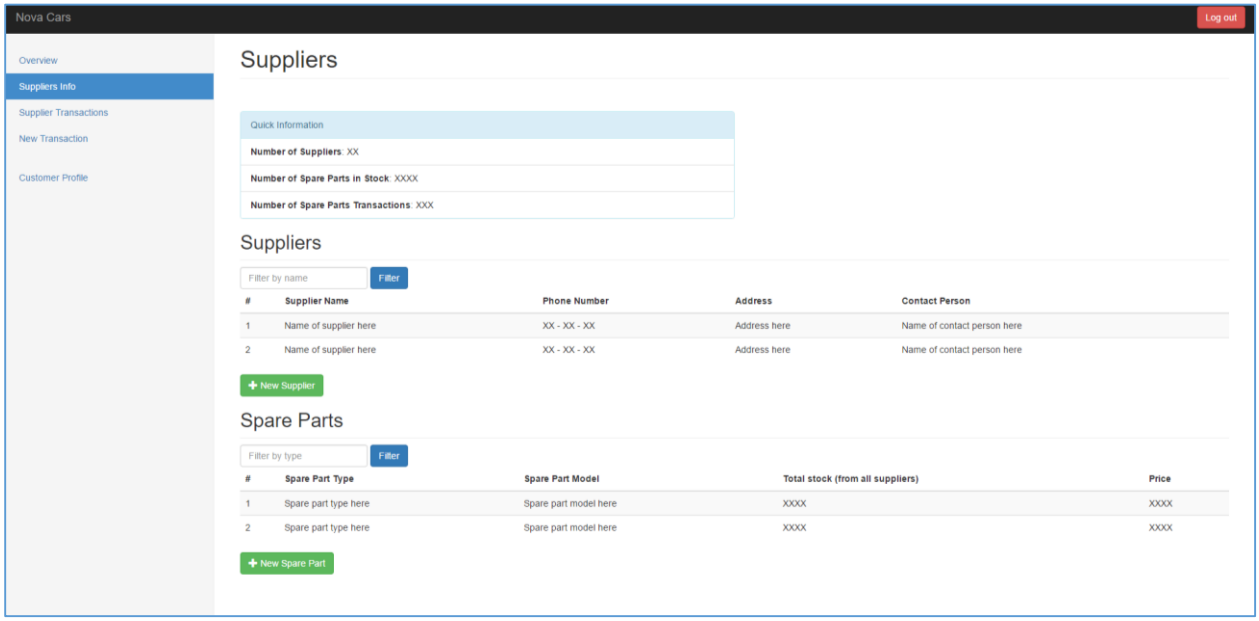
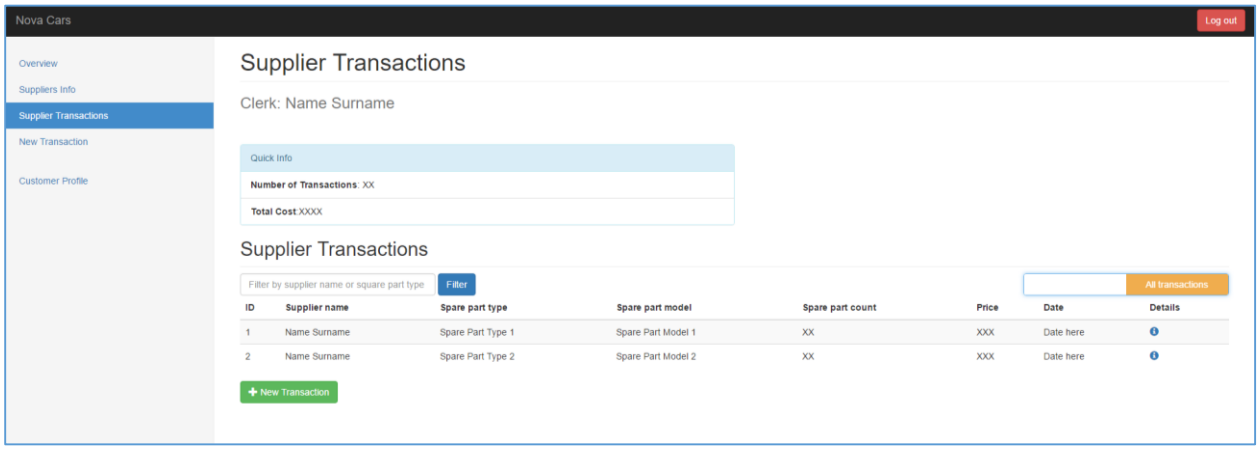


Figure 9 Supplier Info Page

Supplier Transactions Page:

The sales manager can access this page using the dashboard on the left side of any page. This page shows information about transactions done with suppliers and can be filtered to show certain supplier names or spare part types and only showing the transaction done by the sales manager who is logged in.



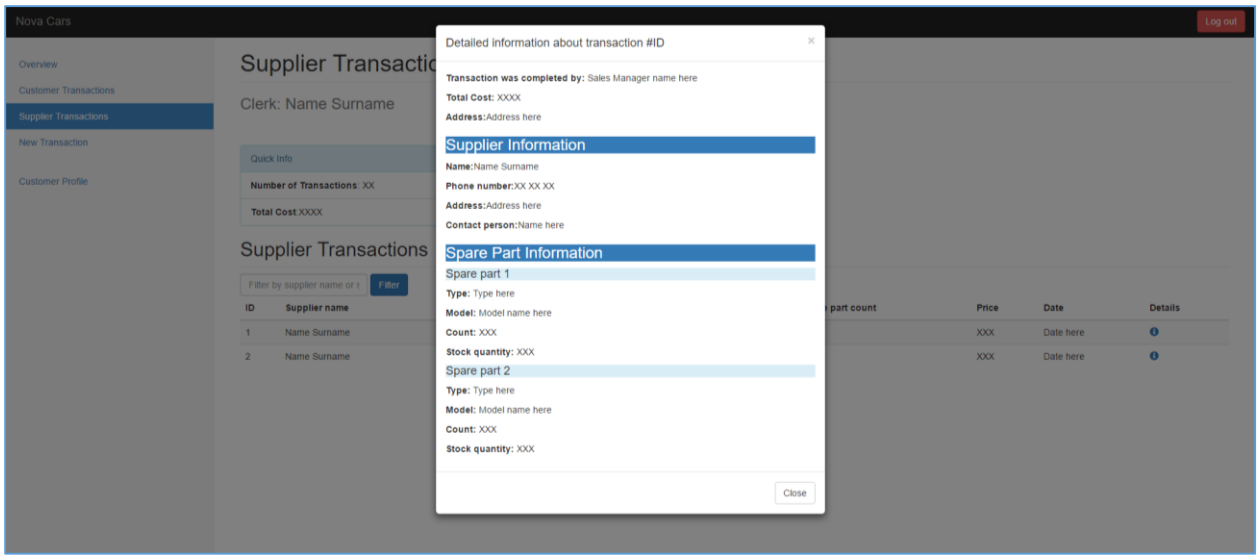


Figure 10 Supplier Transactions Page

New Transaction:

The sales manager can access this page using the dashboard on the left side of any page. By filling the required information and clicking the complete transaction button, the page for transaction will be updated.

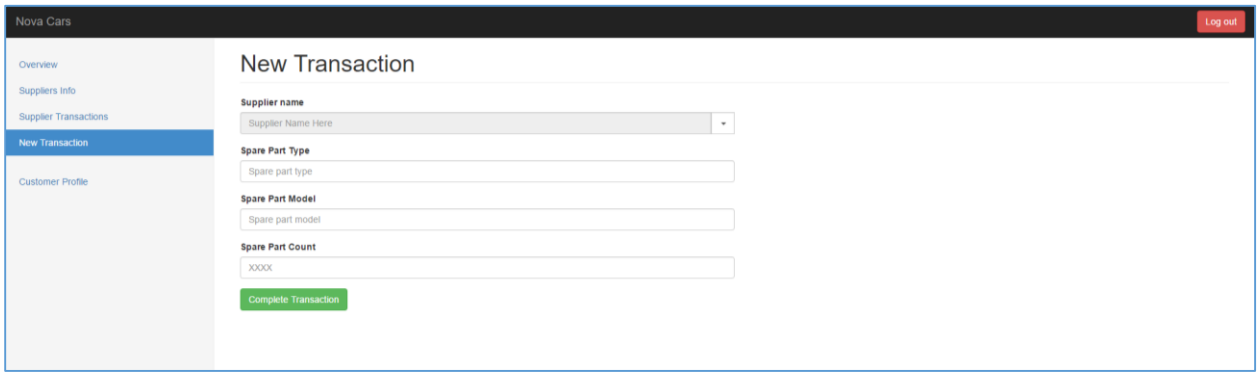


Figure 11 Supplier New Transaction Page

6.4. Manual for Clerk

Clerks can do what a customer can do except for a different profile page (renamed overview) and have more access than a customer.

Overview Page:

At this page the clerk can see the information about himself/herself such as email, salary and expertise level. It also has an option for changing the password. This is almost the same as the Manager's overview page.

Customer Transactions:

The clerk can access this page using the dashboard on the left side of any page. It shows the number of transactions and the total revenue made by them. This page shows information about the transactions all of the clerks have done with customers. The clerk can also obtain further information by clicking the button under details column for each transaction. It can also be filtered by showing only the clerk's transactions and filtering by operation. The clerk can also add new transaction by clicking the button below the table (which redirects the clerk to New Transaction Page).

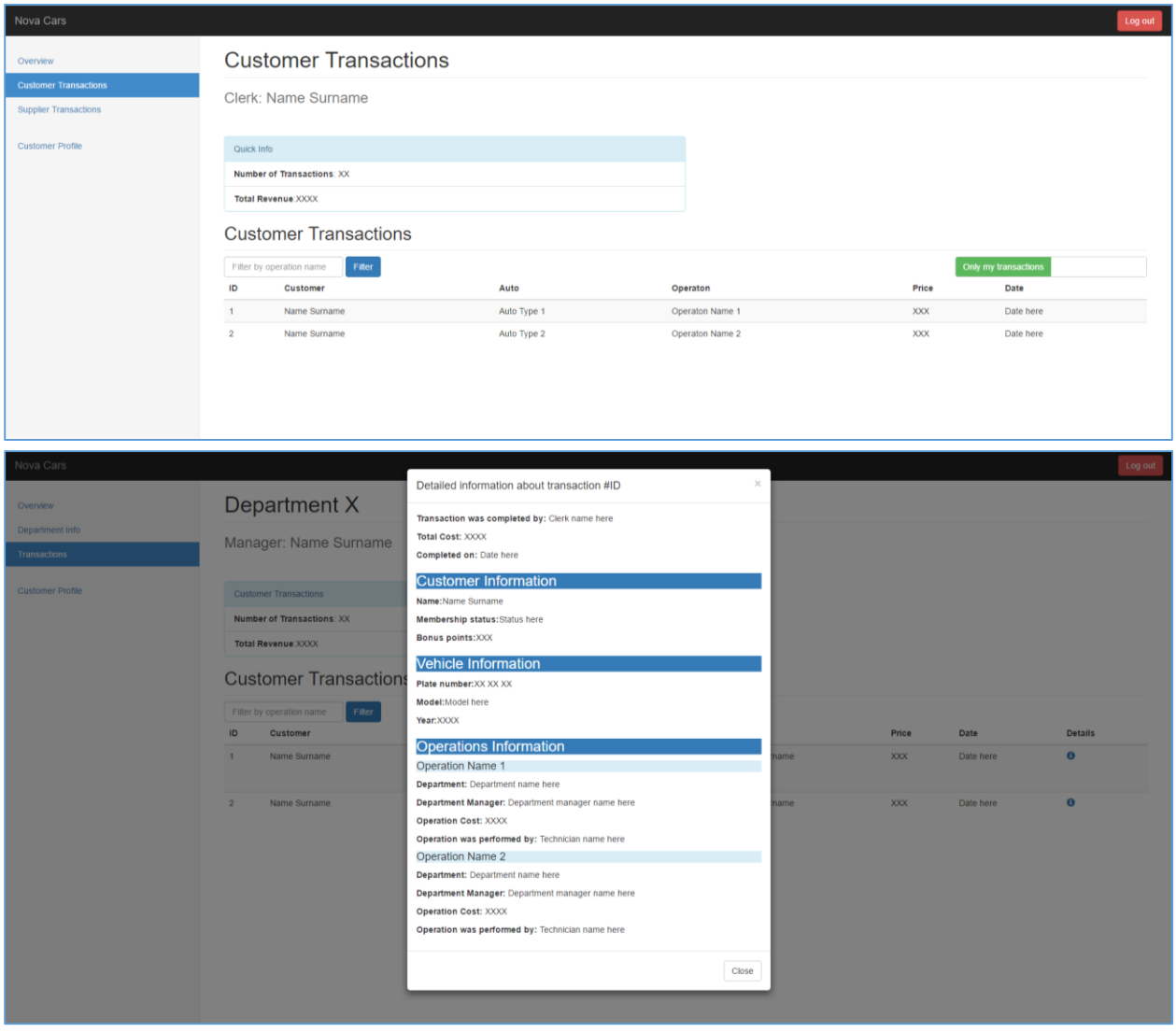


Figure 12 Customer Transactions Pages

Supplier Transactions:

The clerk can access this page using the dashboard on the left side of any page. The page shows the number of transactions and the total cost of transactions done in the table. It shows the information about transactions in the form of a table. The clerk can also obtain further information by clicking the button under details column for each transaction. The clerk can also filter the transactions by supplier name or spare part type.

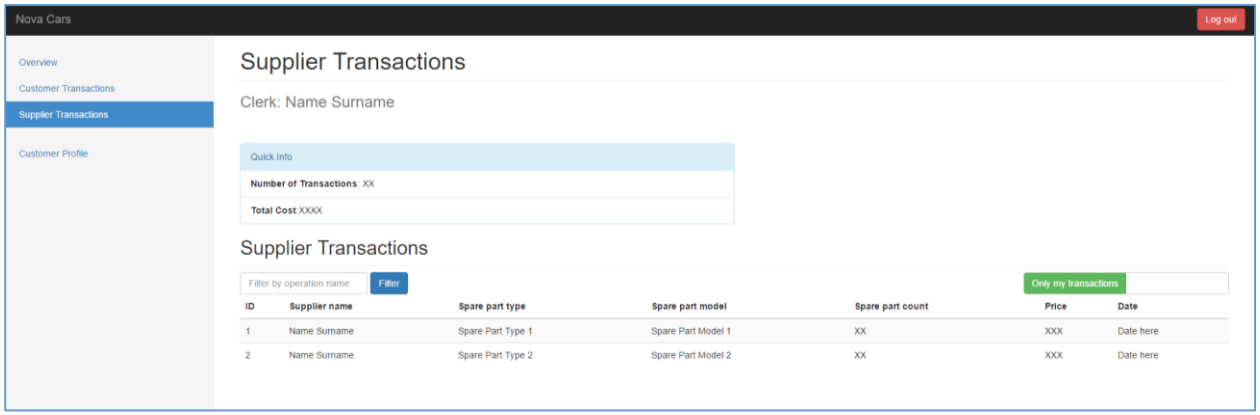


Figure 13 Supplier Transactions Page

New Transaction Page:

The clerk can access this page using the dashboard on the left side of any page or from the customer transactions page. Filling the required information on this page and clicking complete transaction button updates the customer transaction page. The table in this page provides information about operations and technicians and it can be filtered by operation or department name.

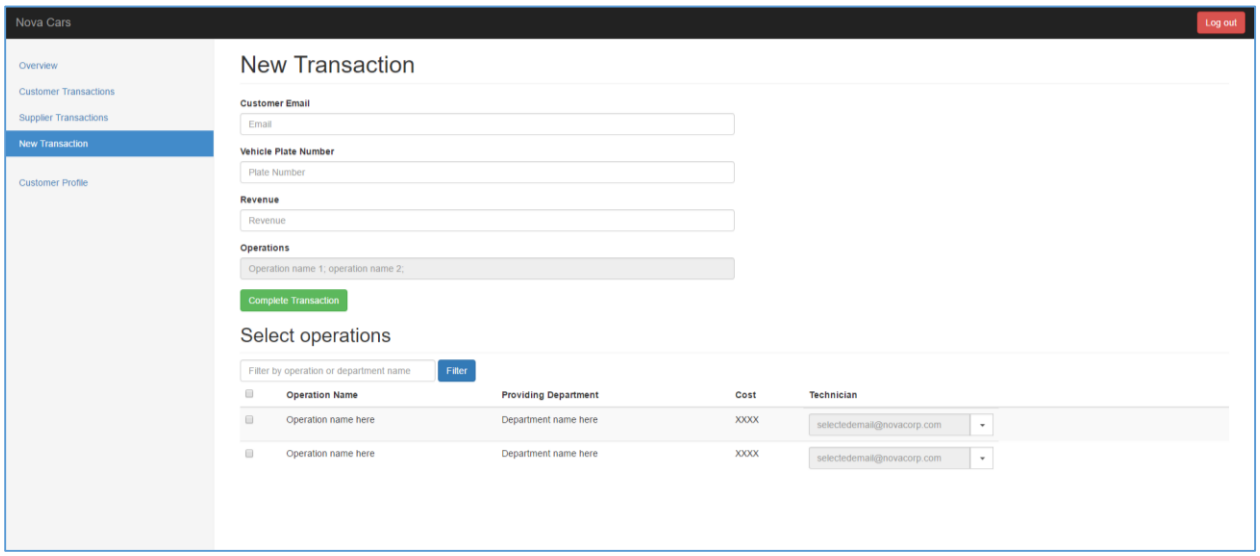


Figure 14 New Transaction for Clerk