VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI-590014, KARNATAKA

A Mini Project Report

On

"WHEELS-SERVICE CENTER PACKAGE"

Submitted in Partial Fulfillment of the Requirement for

"DBMS Laboratory with Mini Project -V Semester"

For the Award of Degree

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE & ENGINEERING

Submitted By:

ASHWIN RAMESH P (1SG16CS016)
ASHISH GOUROJI (1SG16CS015)

Under the Guidance of:

Mrs. Kavya N L
Assistant Professor

Mrs. Nanda M B
Assistant Professor

Department of Computer Science and Engineering

SAPTHAGIRI COLLEGE OF ENGINEERING

14/5, Chikkasandra, Hesarghatta Main Road

Bengaluru - 560057

2018-2019

SAPTHAGIRI COLLEGE OF ENGINEERING

14/5, Chikkasandra, Hesaraghatta Main Road, Bengaluru – 560057.

Department of Computer Science and Engineering

Certificate

Certified that the Mini Project Work entitled **"WHEELS-SERVICE CENTER PACKAGE"** carried out by **ASHISH GOUROJI (1SG16CS015) & ASHWIN RAMESH P (1SG16CS016),** bonafide students of **Sapthagiri College of Engineering**, in partial fulfillment for the award of **Bachelor of Engineering** degree in **Computer Science and Engineering** of **Visvesvaraya Technological University, Belagavi** during the academic year 2018-2019. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The Mini project report has been approved as it satisfies the academic requirements in respect of DBMS Laboratory with Mini Project (15CSL58) prescribed for the said Degree.

Signature of the Guide	**Signature of the Guide**	**Signature of the HOD**
Mrs. Kavya N L	**Mrs. Nanda M B**	**Dr. Yogish H K**
Assistant Professor	**Assistant Professor**	**Professor & Head**

EXTERNAL EXAMINATION

Name of the Examiners	**Signature with Date**
1._____	_____
2._____	_____

ACKNOWLEDGEMENT

Any achievement does not depend solely on the individual efforts but on the guidance, encouragement and co-operation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped us in carrying out this mini project work. We would like to take this opportunity to thank them all.

We would like to express my profound thanks to **Sri. G Dayanand,** Chairman, Sapthagiri College of Engineering Bangalore, for his continuous support in providing amenities to carry out this Mini Project.

Special Thanks to **Manoj G D,** Executive Director, Sapthagiri College of Engineering Bangalore, for his valuable suggestion.

Also we would like to express our immense gratitude to **Dr. K L Shivabasappa**, Principal, Sapthagiri College of Engineering Bangalore, for his help and inspiration during the tenure of the course.

We also extend our sincere thanks to **Dr. Yogish H K**, Professor and Head, Department of Computer Science and Engineering, Sapthagiri College of Engineering, for his constant support.

We would like to express our heartful gratitude to **Mrs. Kavya N L,** Assistant professor and **Mrs. Nanda M B,** Assistant professor, Department of Computer Science and Engineering, Sapthagiri College of Engineering, for their timely advice on the mini project and regular assistance throughout the work.

We also extend our sincere thanks to all the **Faculty members** and **supporting staff** Department of Computer Science and Engineering, Sapthagiri College of Engineering, for their constant support and encouragement.

Finally, we thank our parents and friends for their moral support.

ASHISH GOUROJI

ASHWIN RAMESH P

ABSTRACT (SYNOPSIS)

The objective of Wheels-Service centre package is to provide necessary spares for a convenient service of the vehicle and to maintain a record for the same. It includes details of the vehicle that are arrived at the service centre such as the vehicles unique registration number and the distance covered on the odometer. Once the registration of the vehicle is completed the required spares parts is provided for servicing of the vehicle. If the vehicle has any history of service at the service centre it will get updated with the same else a new vehicle will be registered. The job card number is a unique number that is provided by the service centre to every vehicle that gets serviced in the service centre. Once the service is completed the total billing is done as per the spares used for the vehicle.

TABLE OF CONTENTS

CHAPTER 1

INTRDUCTION

1.1ARCHITECTURE

Database is a collection of related data. DBMS came into existence in 1960 by Charles. Again in 1960 IBM brought IMS-Information management system. In 1970 Edgor Codd at IBM came with new database called RDBMS. In 1980 then came SQL Architecture-Structure Query Language. In 1980 to 1990 there were advances in DBMS e.g. DB2, ORACLE. A database has the following implicit properties:

- A database represents some aspect of the real world, sometimes called the miniworld or the universe of discourse (UOD). Changes to the miniworld are reflected in the database.
- A database is a logically coherent collection of data with some inherent meaning. A random assortment of data cannot correctly be referred to as a database.
- A database is designed, built, and populated with data for a specific purpose. It has an intended group of users and some preconceived applications in which these users are interested.

In other words, a database has some source from which data is derived, some degree of interaction with events in the real world, and an audience that is actively interested in its contents.

Metadata (meta data, or sometimes meta information) is "data about data", of any sort in any media. An item of metadata may describe a collection of data including multiple content items and hierarchical levels, for example a database schema. In data processing, metadata is definitional data that provides information about or documentation of other data managed within an application or environment. The term should be used with caution as all data is about something, and is therefore metadata.

A database management system (DBMS) is a collection of programs that enables users to create and maintain database. The DBMS is a general purpose software system that facilitates the process of defining, constructing, manipulating and sharing databases among various users and applications.

Defining a database specifying the database involves specifying the data types, constraints and structures of the data to be stored in the database. The descriptive information is also stored in the database in the form database catalogue or dictionary; it is called meta-data. Manipulating the data includes the querying the database to retrieve the specific data. An application program accesses the database by sending the queries or requests for data to DBMS. The important function provided by the DBMS includes protecting the database and maintain the database.

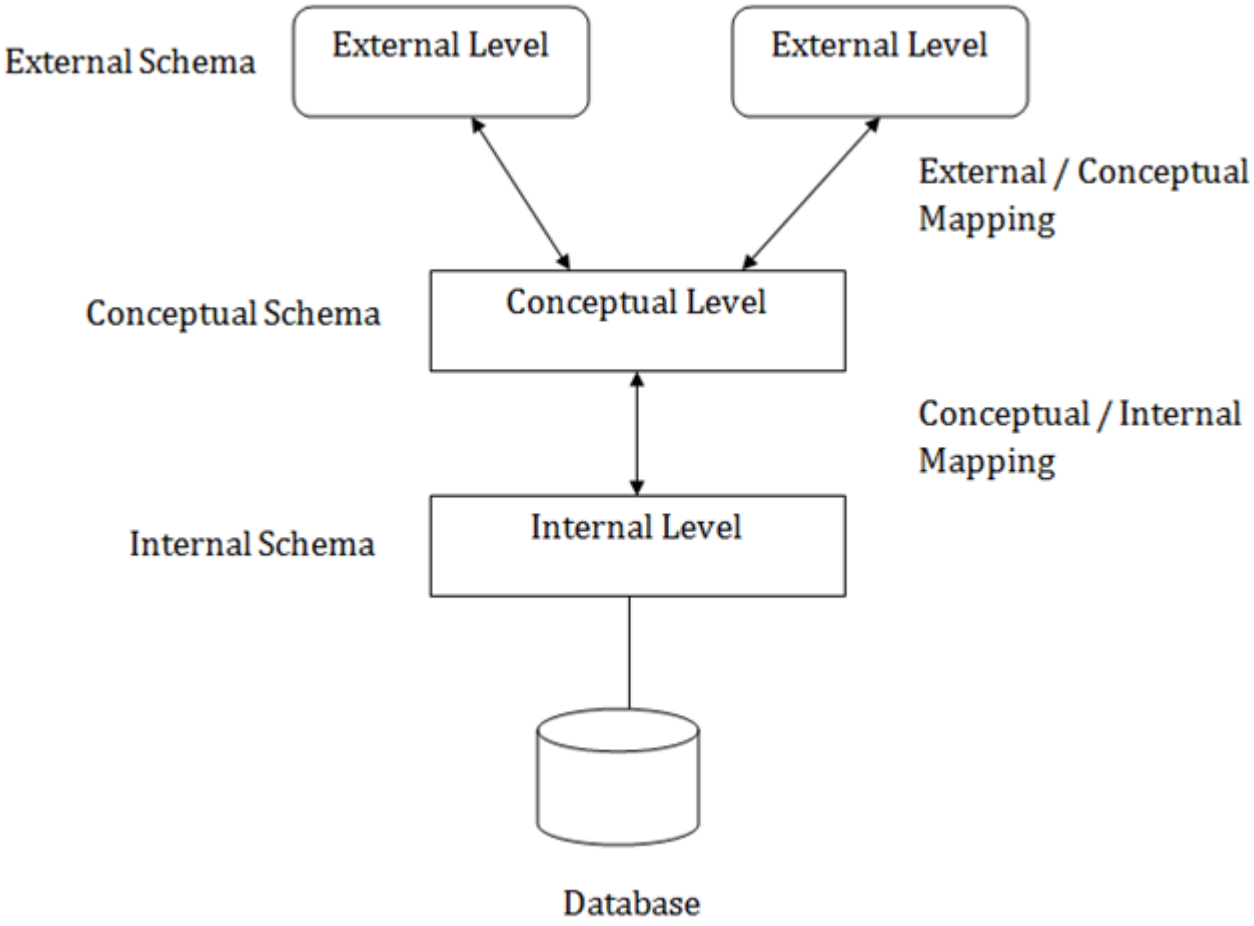

Figure 1.1: Three schema architecture

The figure 1.1 shows the Three schema architecture of Database Management System, The Three schema architecture consists of three levels of the architecture**:**

- **External Level:**

  The external level is the view that the individual user of the database has. This view is often a restricted view of the database and the same database may provide a number of different views for different classes of users. In general, the end users and even the application programmers are only interested in a subset of the database.

- **Conceptual Level:**

  The conceptual view is the information model of the enterprise and contains the view of the whole enterprise without any concern for the physical implementation. The conceptual view is the overall community view of the database and it includes all the information that is going to be represented in the database.

- **Internal Level:**

  The internal view is the view about the actual physical storage of data. It describes what data is stored in database and how.

1.2 OVERVIEW OF THE PROJECT

This is a Wheels-Service centre package project report which gives a brief description on the tasks worked out, development of ERD from the given scenario whereby we had four entities Spares, Vehicle, Job Card and Billing with their attributes. Afterward we were able to build the database using MySQL whereby we had five tables that we developed from the ERD which had four entities thus by breaking many to many relationships we were able to come up with five tables. Then on developing the front end application we used JAVA SWINGS. Thus the other task was to write and apply the SQL queries that could answer the provided questions.

1.2.1 Problem statement:

The purpose of the Wheels-Service centre package is to design a database system for effective management of spares inventory and maintain a record of vehicles being serviced at the service centre through the front-end application together with the required queries.

1.2.2 Objectives of the project:

- The main objective behind wheels service center package is to maintain a record of all vehicles being serviced at the Service centre.
- The Wheels-Service centre package gives the information about Spares inventory, vehicles being serviced, billing, history of service, etc.
- The wheels-service center package is built at administrative end for authorized personals of the service centre, hence only the administrator is guaranteed the access .

- The main purpose of wheels-service center package is to reduce manual work for maintaining the vehicle database.

CHAPTER 2

SYSTEM DESIGN AND METHODOLOGY

2.1 SYSTEM ARCHITECTURE

The main software used are java and MySQL.

Java at the core is an Object Oriented Programming Language. This java follows 3 OOPS programming language principles that are encapsulation, inheritance and polymorphism. Encapsulation is an mechanism by which both the data and code that acts on it are encapsulated into a single unit and thus increases safety and outside interference. Inheritance is the process by which one object can be derived from another object and have its properties passed down to it. Thus it supports the idea of hierarchical classification. Polymorphism concept allows overloading of the function with the same name. this allows one interface to be used for general class of actions. A few buzzwords of java are that java is simple: It is easy to use and easy to understand, secure: it is secure due to data abstraction, portable: this means that the same program can run on any computer that has a JVM regardless of the system requirements, object-oriented: It follows OOPS concepts, robust: It is robust as it not only checks your code at compile time but also at run time, multithreaded: We can write programs that do many things simultaneously, architecture-neutral: Java follows the policy of “write once, run anywhere, anytime, forever”, interpreted: Using byte code it allows the creation of cross-platform programs, high performance: Using just-in-time compiler it is possible to convert the code into its native machine code directly allowing for high performance, distributed: Java is designed for distributed environments. This means that it can run on various platforms, dynamic: Java contains a lot of run-time type information that is used to verify and resolve the access at run time.

MySQL is an open source relational database management system. The main advantages of DBMS system are controlling redundancy, restricting unauthorized access, providing persistent storage, providing storage structure and search. Techniques of Efficient Query Processing are providing backup and recovery, providing multiple user interfaces, representing complex relationships among data.

2.2 PROJECT REQUIREMENTS

Hardware Requirements		
Processor	**RAM**	**Disk Space**
Pentium i3, i5 or more	2 GB or Higher	10 GB
Software Requirements		
Operating System		**Language/Tools/Database**
Windows 7, 8, 10		JDK1.7/Eclipse/SQL

2.3 ENTITY RELATIONSHIP DIAGRAM

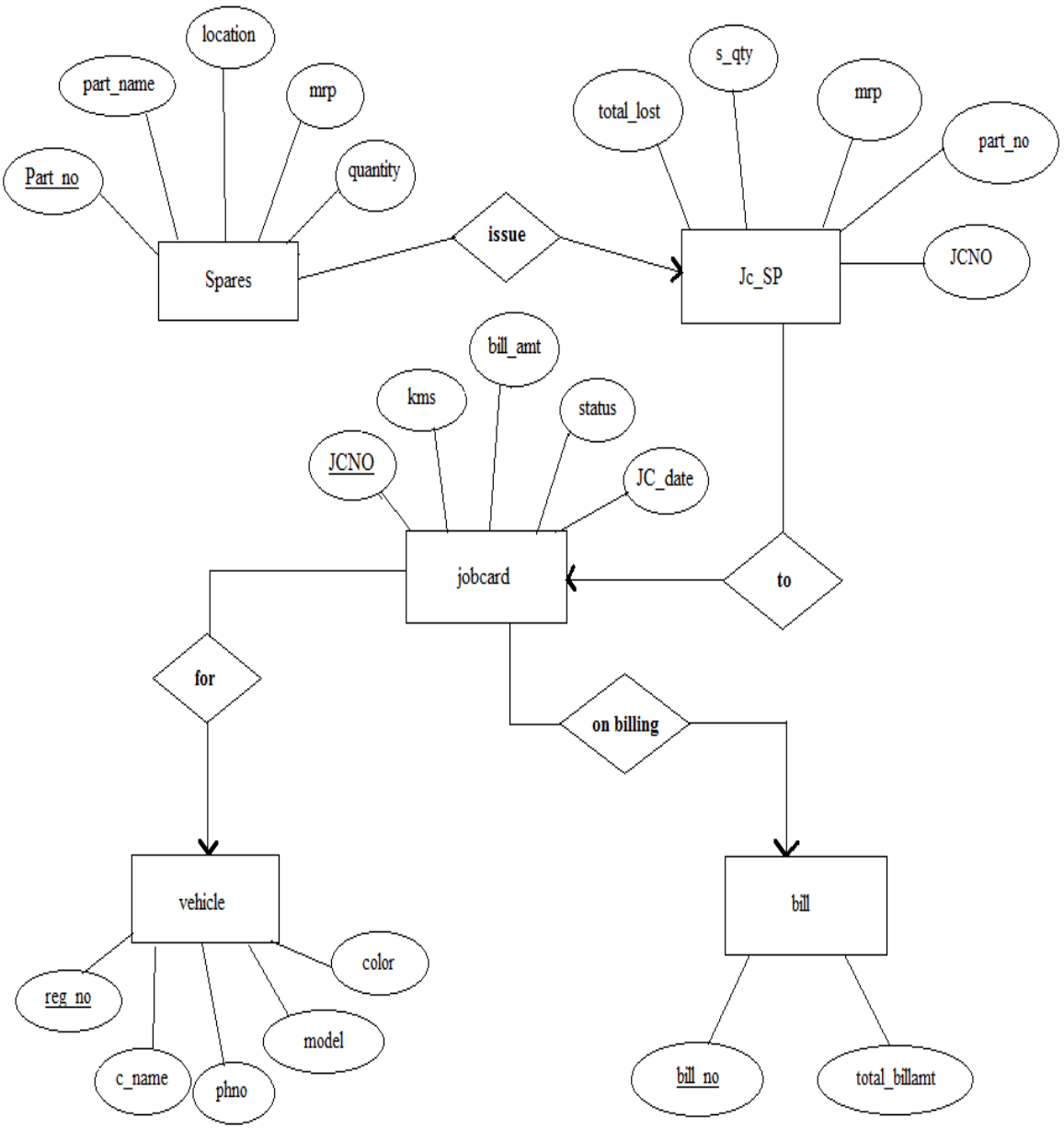

Figure 2.1: ER diagram for wheels-service center package

An Entity-Relationship Diagram (ERD) is a data modelling technique that graphically illustrates an information system's entities and the relationships between those entities. An ERD is a conceptual and representational model of data used to represent the entity framework infrastructure.
The ER Diagram shown in fig 2.2 consists of 5 attributes namely

1. Spares
2. Vehicle
3. Jobcard
4. JC_SP
5. Bill

The **SPARES** consist of the following attributes: part_no, part_name, location, mrp and quantity where part_no is primary key.

The **VEHICLE** consist of the following attributes: reg_no, c_name, phno, model and color where reg_no is primary key.

The **JOBCARD** consist of the following attributes: jcno, kms, bill_amt, status and JC_date where jcno is primary key.

The **JC_SP** consist of the following attributes: total_lost, s_qty and mrp.

The **BILL** consists of the following attributes: bill_no and total_billamt where bill_no is primary key.

2.3 Schema Diagram

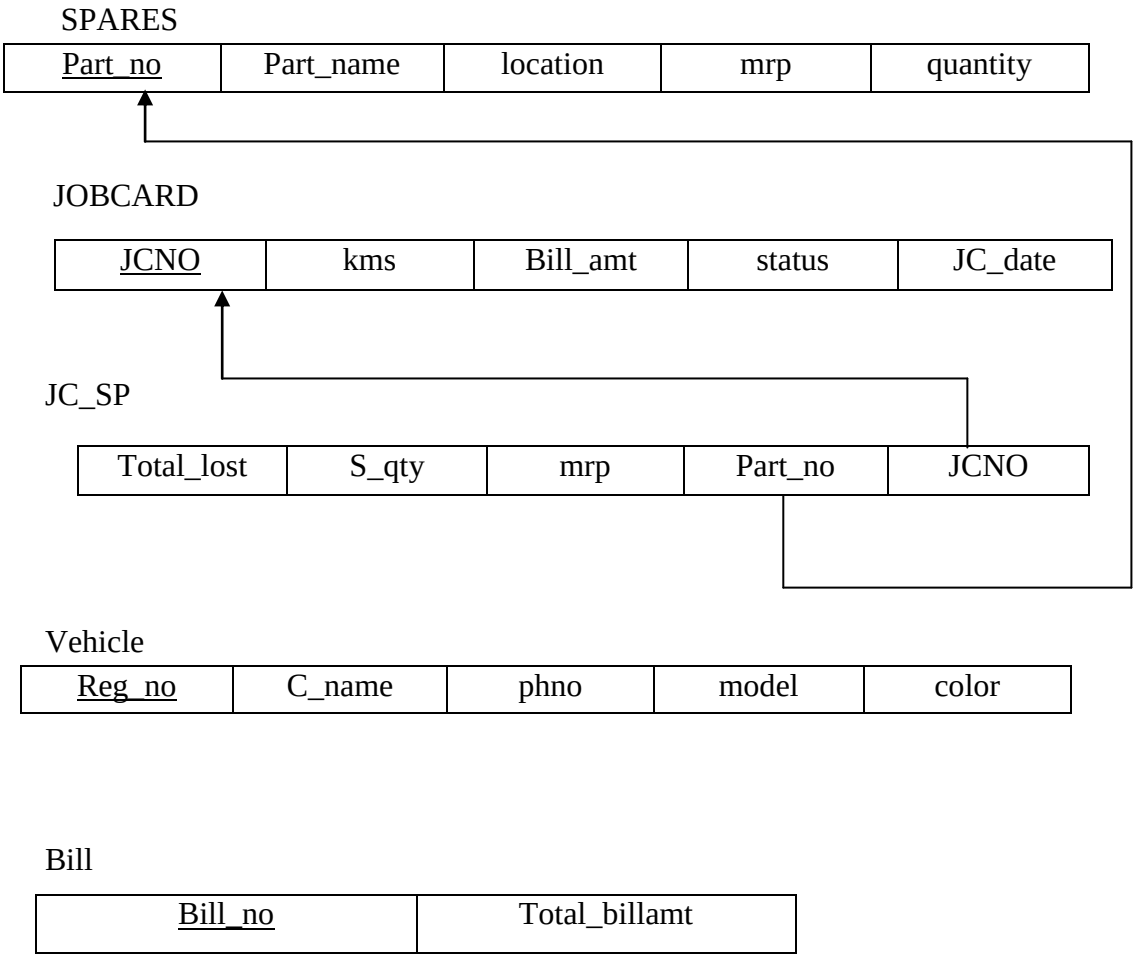

Figure 2.2: Schema Diagram of wheels-service center package

2.5 ALGORITHM

Store procedure

Step 1: BEGIN

Step 2: DELIMITER //

Step 3: CREATEPROCEDUREtotal_amt(INj_noVARCHAR(11))
DECLAREamtINT ;
SETamt=(SELECTSUM(total_cost) FROM `js_sp` WHERE jcno=j_no);
UPDATE `jobcard` SET `Bill_amt`=amt WHERE `JC_no`=j_no;

END //
DELIMITER ;
Step 4: END

Description: the jobcard number is passed as a parmeter to the stored procedure 'total_amt', which calculates the sum of all the costs of spares issued to the particular jobcard and updates it into the 'bill_amt' of corresponding jobcard.

Trigger

To Decerease quantity after issuing spares to jobcard

```
BEGIN
DELIMITER //
CREATE TRIGGER decqty
AFTERINSERT ON js_sp
FOR EACH ROW
BEGIN
UPDATE spares s set s.Qty=s.Qty- NEW.sqty WHERE s.PARTNO=NEW.partno ;
END //
DELIMITER ;
```

Description: to descrease the quantity of spare parts from the service center when it has been issued by the customer.

To Update quantity issued to jobcard

```
DELIMITER //
CREATE TRIGGER updqty
AFTERUPDATE ON js_sp
FOR EACH ROW
BEGIN
UPDATE spares s set s.Qty=s.Qty+(OLD.sqty-NEW.sqty) WHERE
S.PARTNO=NEW.partno ;
END //
DELIMITER ;
```

Description: to update the quantity of the spares in the jobcard by making changes in the already existing data.

On Delete quantity from jobcard stock needs to be added back

```
DELIMITER //
CREATE TRIGGER delqty
AFTERDELETE ON js_sp
FOR EACH ROW
BEGIN
UPDATE spares s set s.Qty=s.Qty+OLD.sqty WHERE s.PARTNO=OLD.partno ;
END //
DELIMITER ;
```

Description: to delete the quantity of stock from the jobcard.

To update status after billing

```
DELIMITER //
CREATE TRIGGER update_status
AFTERINSERT ON bill
FOR EACH ROW
BEGIN
UPDATE `jobcard` SET `Status`="Delivered" WHERE `JC_no`=new.`JC_no` ;
END //
DELIMITER ;
```

Description: updating the status of bill in the jobcard from pending to delivered after the final calculation is completed.

CHAPTER 3

Module Implementation

To implement this project, MYSQL is used for backend and java is used for frontend (GUI) creation.

Some of the features of java are:

- **Simplicity:** Java was designed with a small number of language constructs so that programmers could learn it quickly.

- **Object-Oriented:** Java supports the construction of programs that consists of collection of objects.

- **Robust:** Java is designed to eliminate certain types of programming errors. Java is strongly typed, which allows extensive compile-time error checking. It does not support memory pointers, which eliminates the possibility of overwriting memory and corrupting data.

- **Secure:** Java is designed to be secure in a networked environment. The Java run-time environment uses a bytecode verification process to ensure that code loaded over the network does not violate Java security constraints.

- **Portable:** In addition to supporting architecture neutrality, Java ensures that other implementation-dependent aspects of language specification are eliminated

- **Multithreaded:** Java supports multiple threads of execution. This makes programming with threads much easier.

- **Dynamic Language:** Java supports dynamic loading of classes, dynamic compilation, and automatic memory management.

3.1. Module Description

The modules included in this project are:

1. LOGIN.

INPUT : username and the password.

OUTPUT:A successful login, shall take the administrative to the home page, else if the login attempt fails, it will ask to enter once again login details.

DESCRIPTION: Front end is designed using Java Swings.
Login page provides 2 textboxes to enter username and password, on entering the administrative shall click on login button, on clicking login button, an Event Handler will call the associated method.
If the entered username and password matches with registered details, the administrator will be taken to login page, otherwise error message will be displayed in the same page, user will be asked to enter username and password again.

2.QUERIES.

INPUT: The input is specified by type of the query being executed.

OUTPUT: It displays the output of selected query.

DESCRIPTION: Front end is designed using Java Swings.
This query page consists of a list containing 23 queries,it displays the output of selected query iff proper input is provided,otherwise it results NO DATA with improper inputs.

3.INSERT.

INPUT: The input is specified by type of the query being executed.

OUTPUT:' Successfully stored' message is displayed when data is inserted.

DESCRIPTION: It inserts the provided data into the selected table.

4.DELETE:

INPUT: The input is specified by type of the query being executed.

OUTPUT:‘ Successfully deleted’ message is displayed when data is deleted.

DESCRIPTION: It deletes the data from the selected table.

5.UPDATE

INPUT: The input is specified by type of the query being executed.

OUTPUT:‘ Successfully updated’ message is displayed when data is updated.

DESCRIPTION: It updates the data from the selected table.

CHAPTER 4

Results and Screenshots

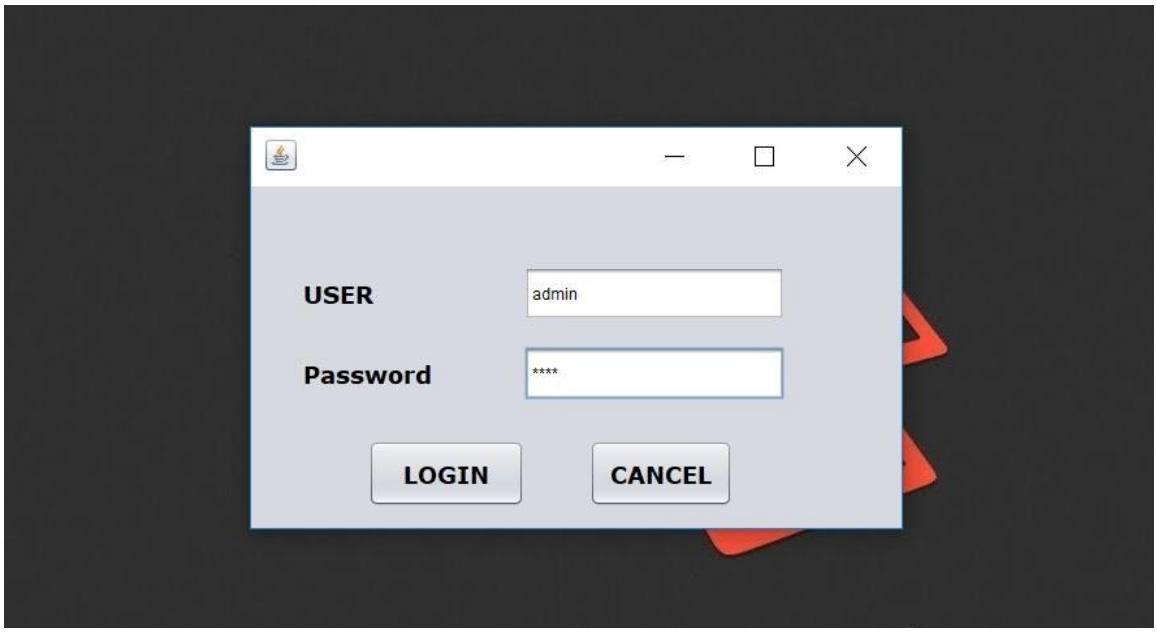

Figure 4.1: login page

The above figure 4.1 login page represents the users login where the registered user can log in the system.

Figure 4.2: homepage

The above figure 4.2 homepage gives the description about the system as the navigation to the different section involved.

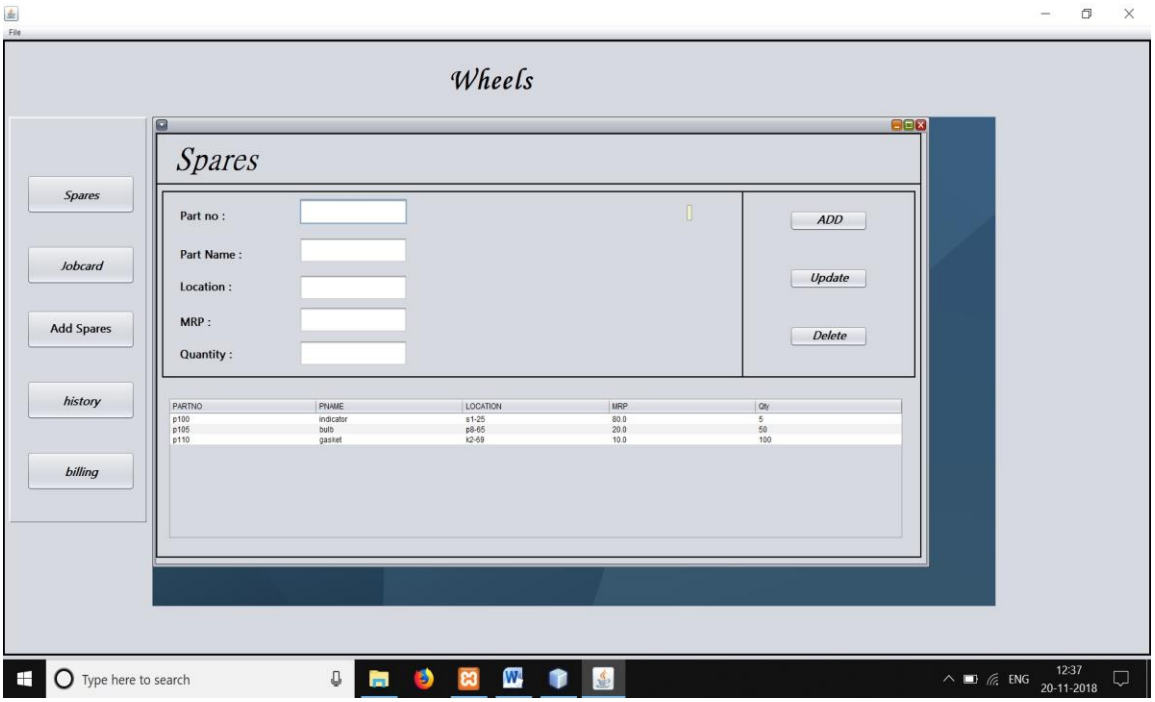

Figure 4.3: spares page

The above figure 4.3 the spares page represents the information of the spares available in the service center.

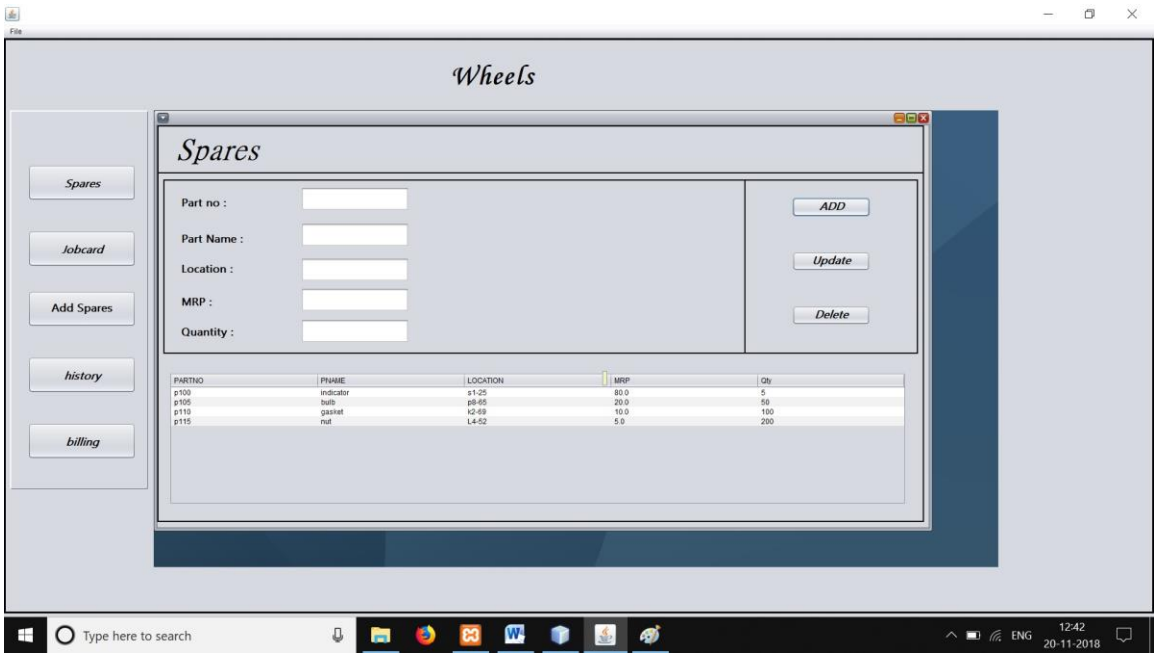

Figure 4.5: spare parts added

The above figure 4.5 represents the spare parts that gets added and stored in the database.

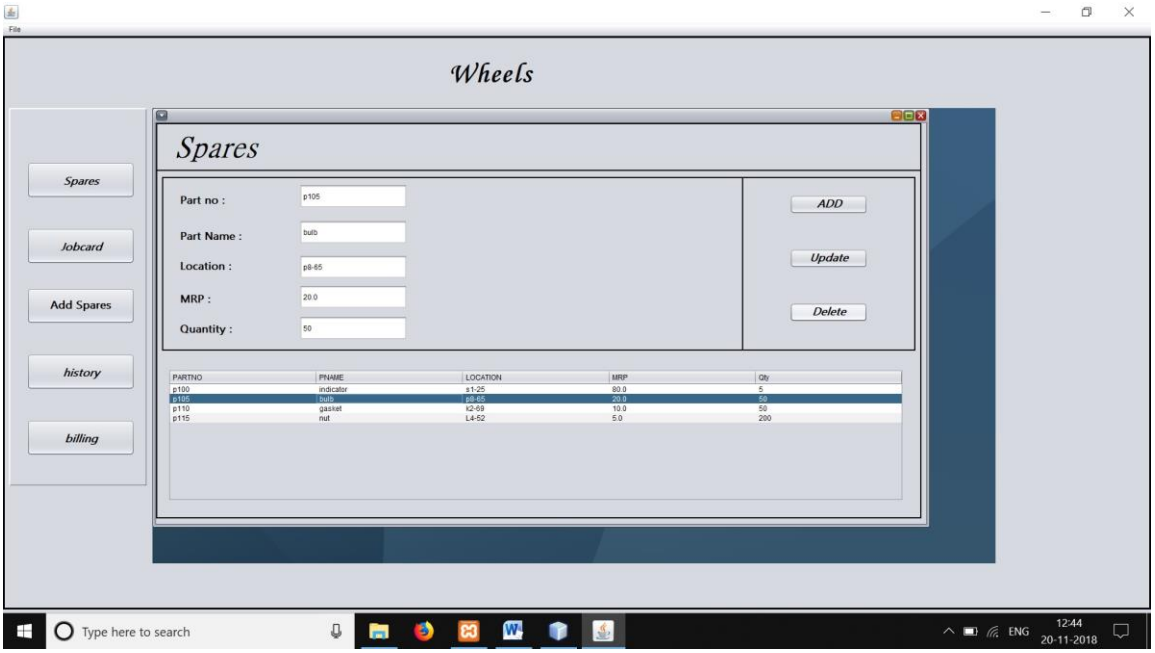

Figure 4.6: deleting spare parts

The above figure 4.6 represents the deleting of information of the spare.

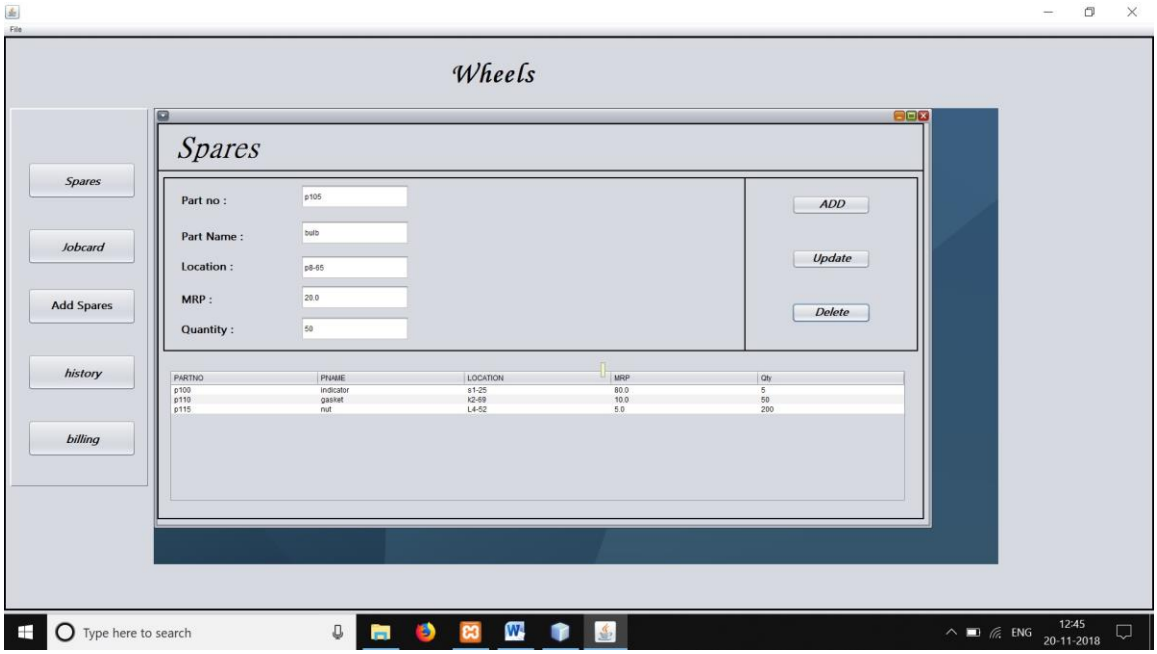

Figure 4.7: spare part deleted

The above figure 4.7 represents the spare parts that got deleted from the database.

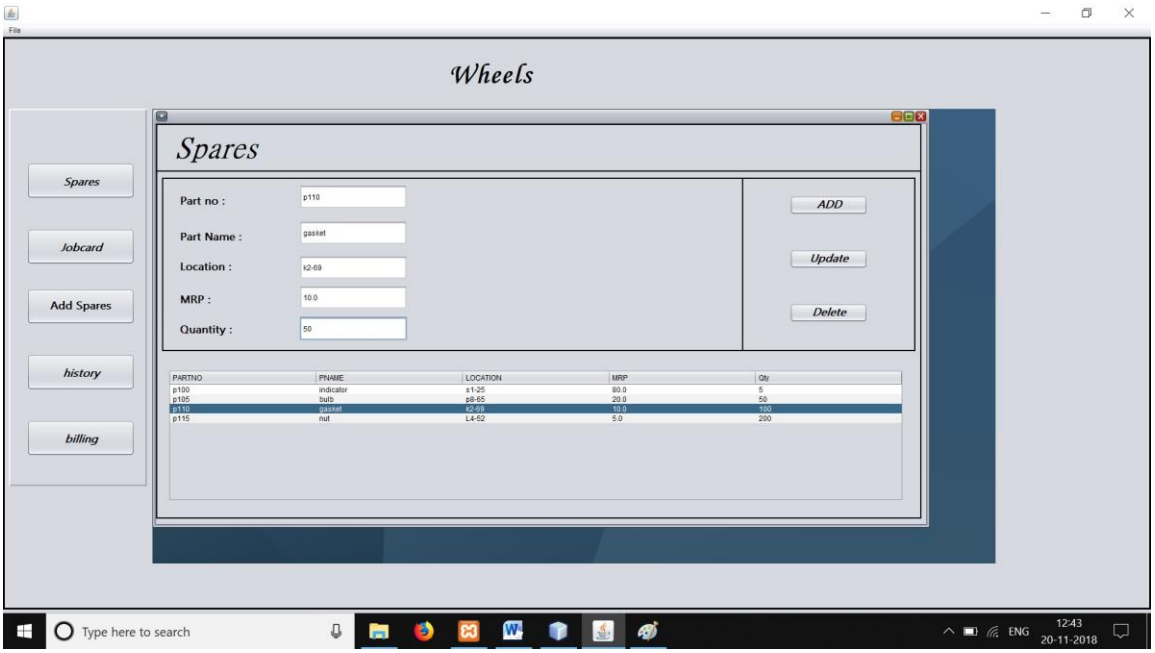

Figure 4.8: updating the list

The above figure 4.8 helps in updating the existing spares information.

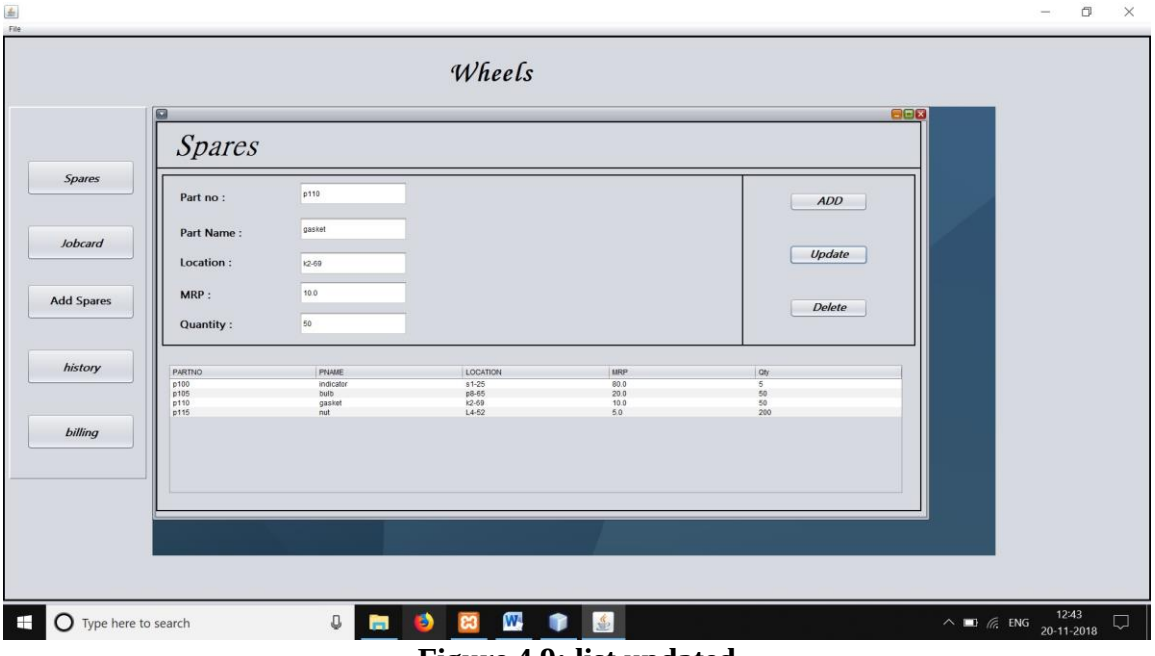

Figure 4.9: list updated

The above figure 4.9 represents the updated information of the spares and stores it to the database.

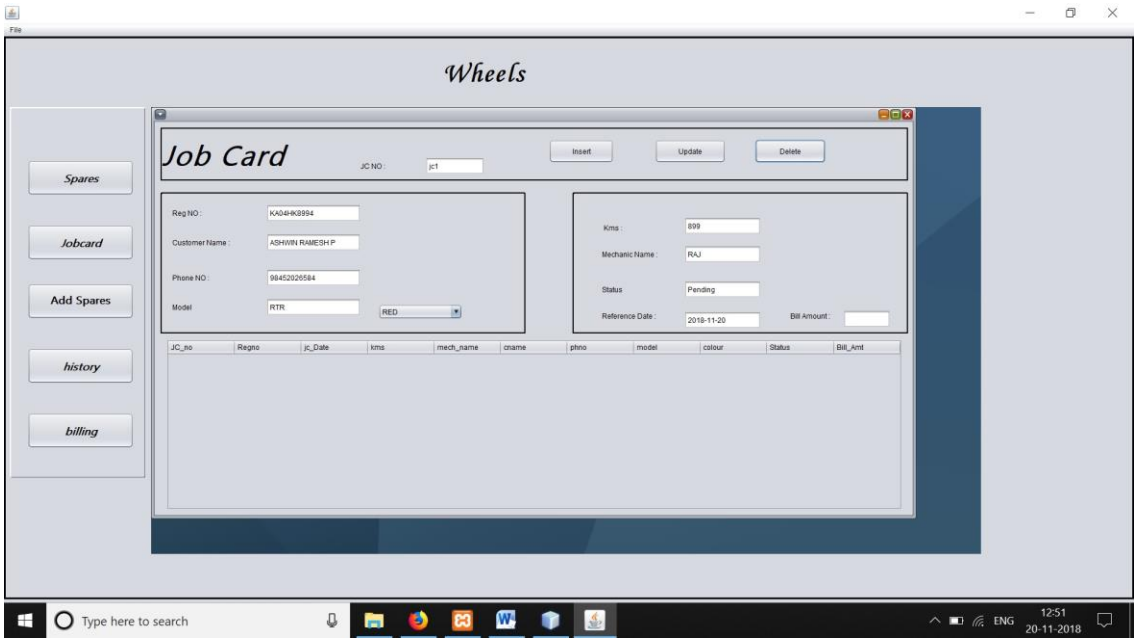

Figure 4.10:adding jobcard

The above figure 4.10 represents the jobcard that is to be added to the database.

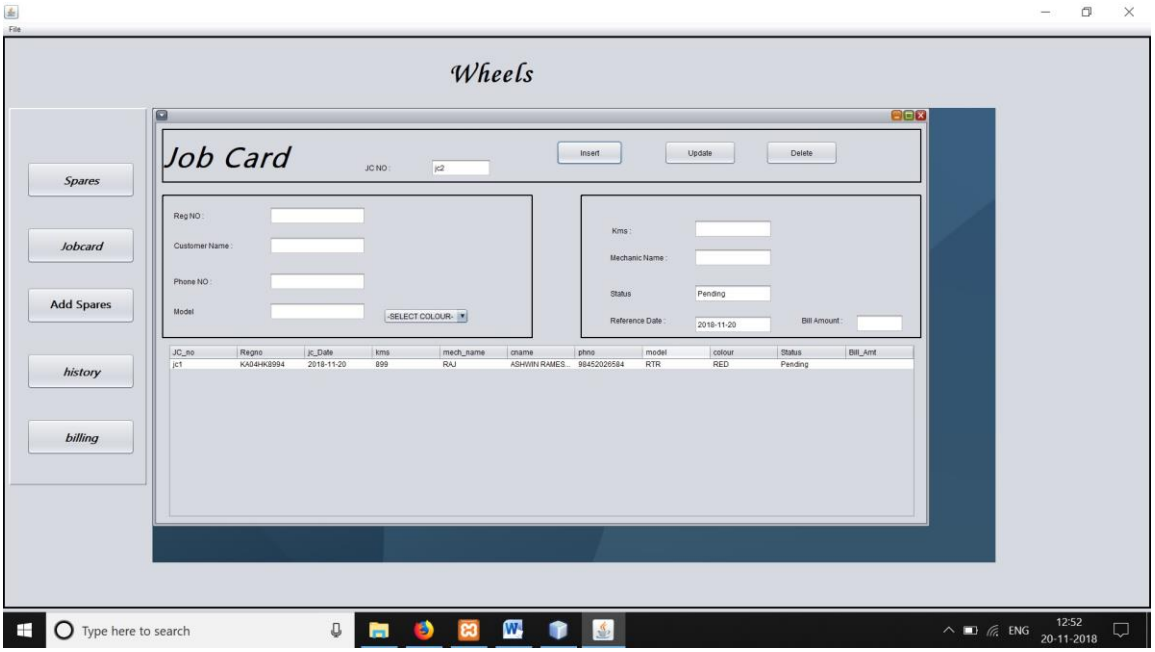

Figure 4.11:jobcard added

The above figure 4.11 represents the jbcard that has been added to the database.

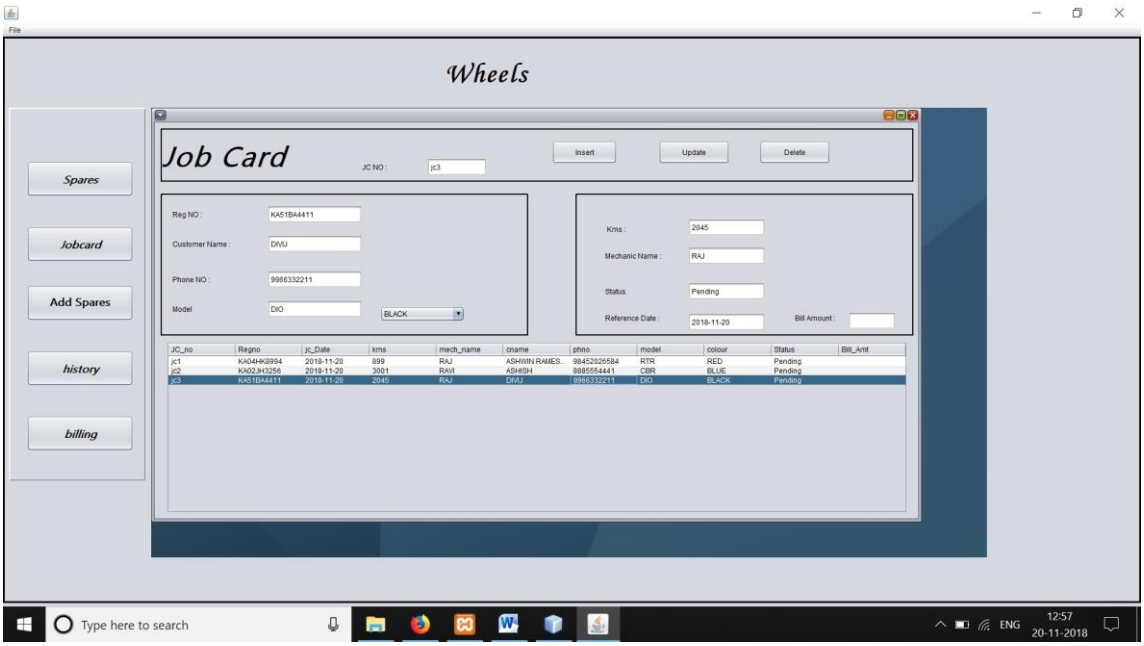

Figure 4.12: deleting jobcard

The above figure 4.12 represents the deletion of an already existing jobcard,

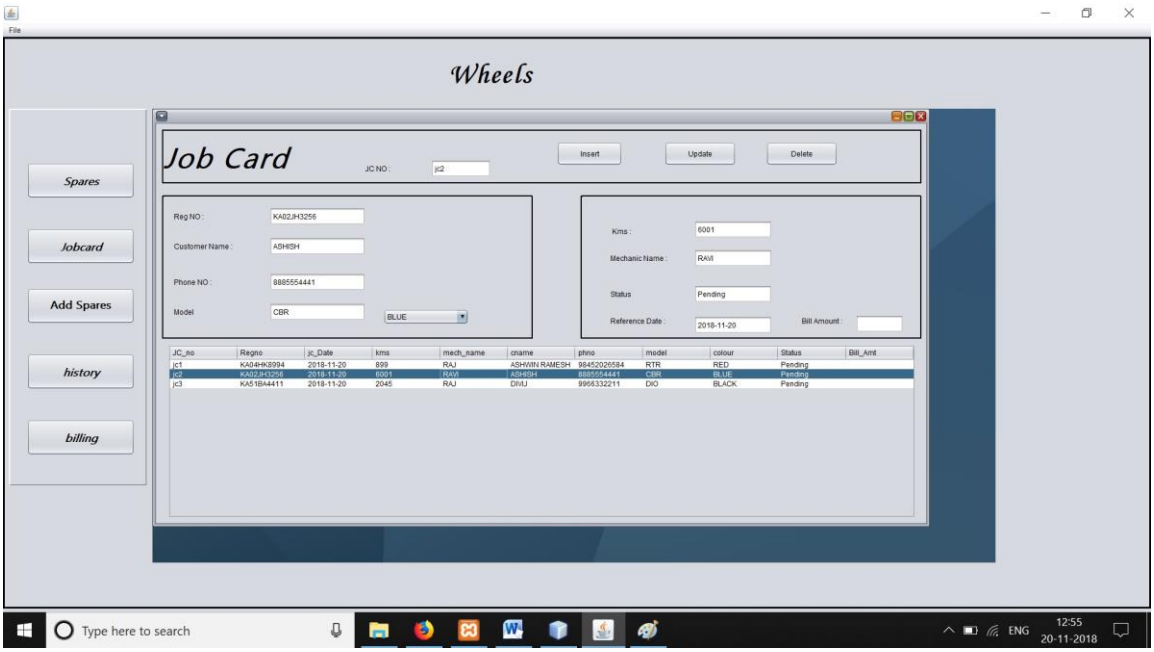

Figure 4.13: updating jobcard

The above figure 4.13 represents the updation of the existing jobcard.

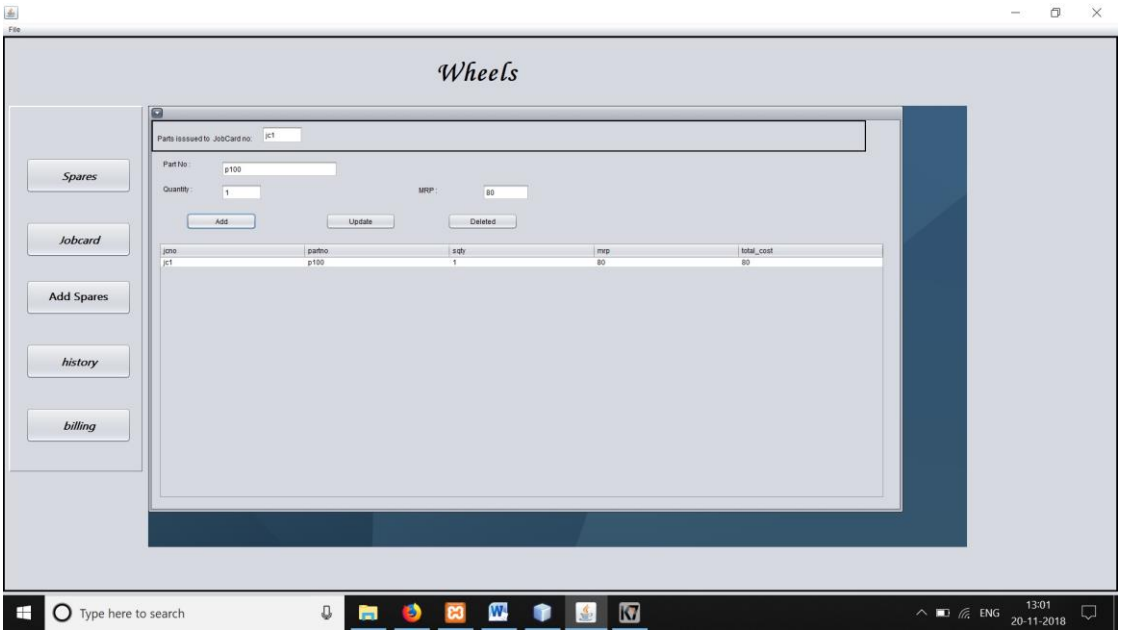

Figure 4.14: adding spare parts to jobcard

The above figure 4.14 represents adding of spare parts to the jobcard.

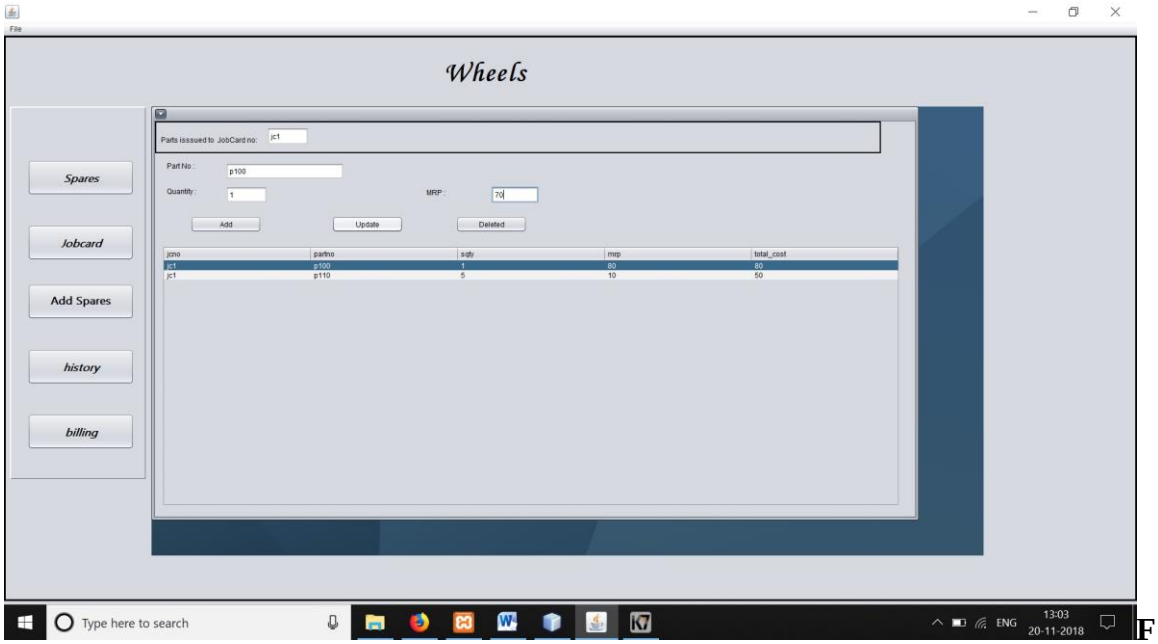

Figure 4.15: updating spare parts in jobcard

The above figure 4.15 represents the updation of already added spare parts in the jobcard

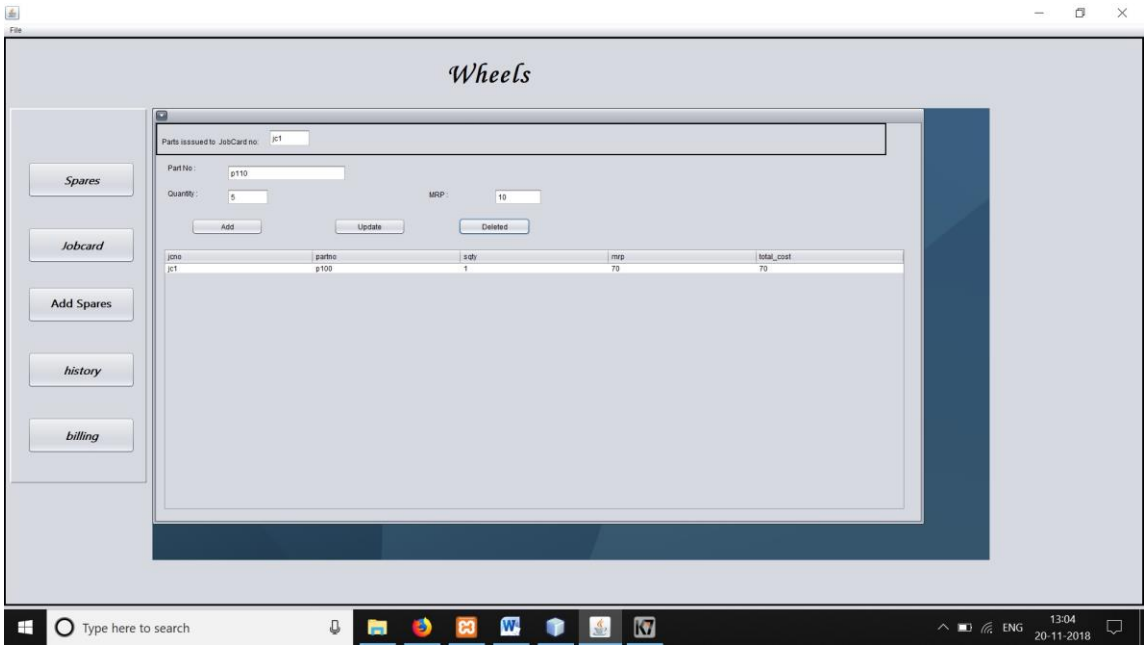

Figure 4.16: deleting spare parts from jobcard

The above figure 4.16 represents the deletion of the already existing spare parts from the jobcard.

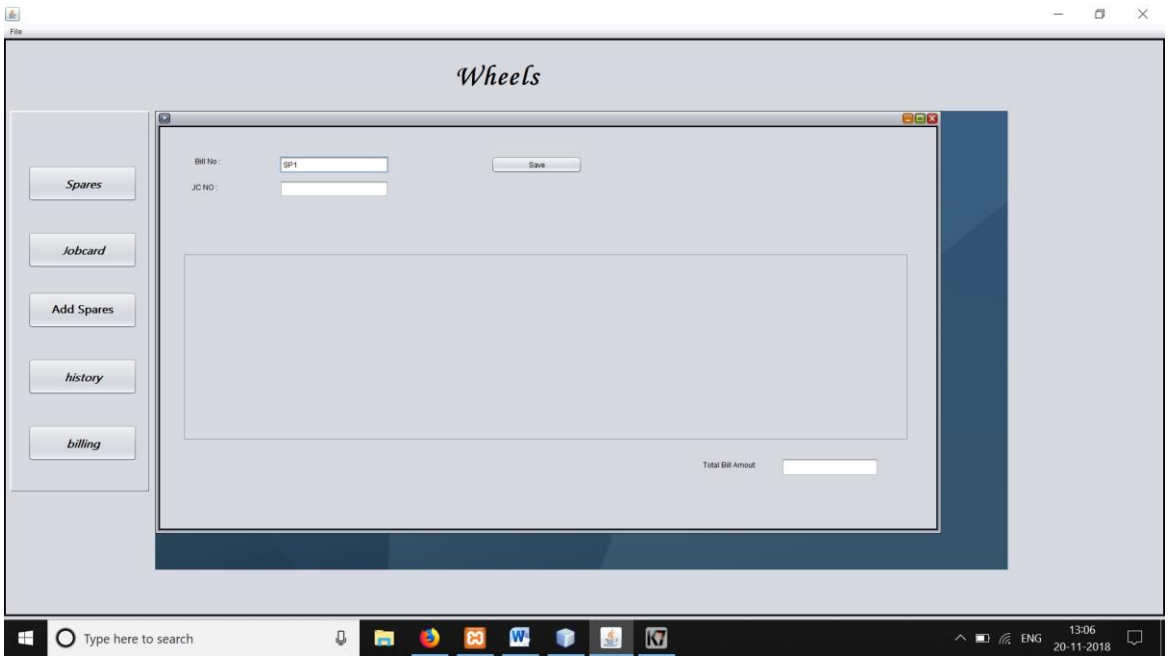

Figure 4.17: billing the total

The above figure 4.17 represents the total billing as per the spares used.

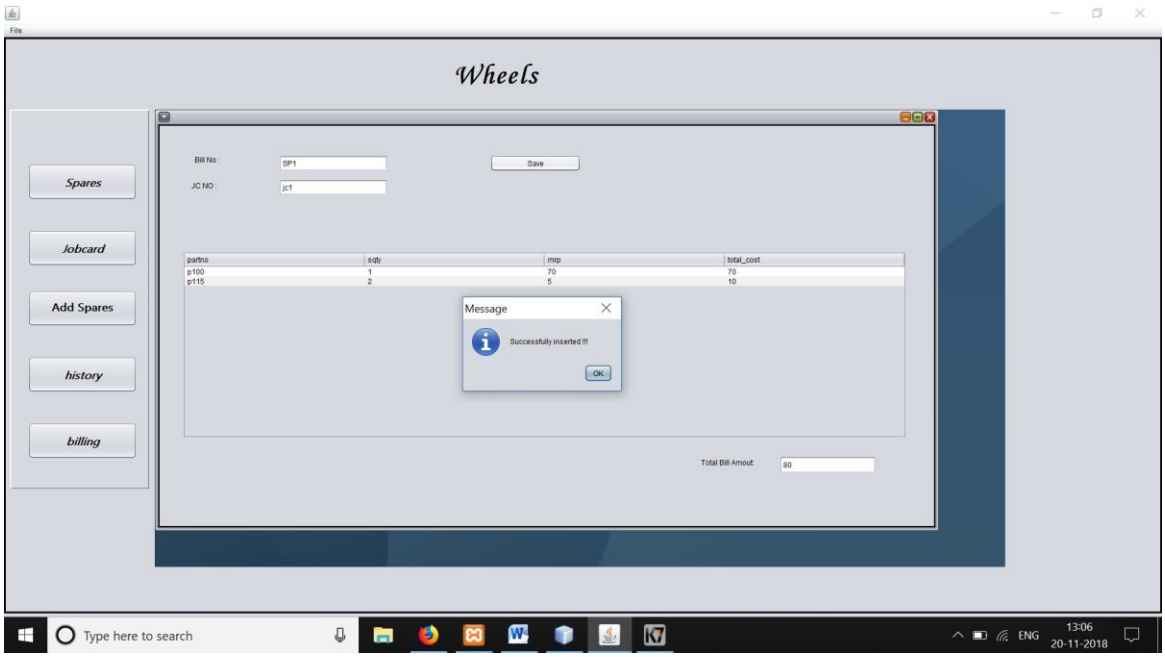

Figure 4.18: final bill

The above figure 4.18 represents the final bill calculated by the service center.

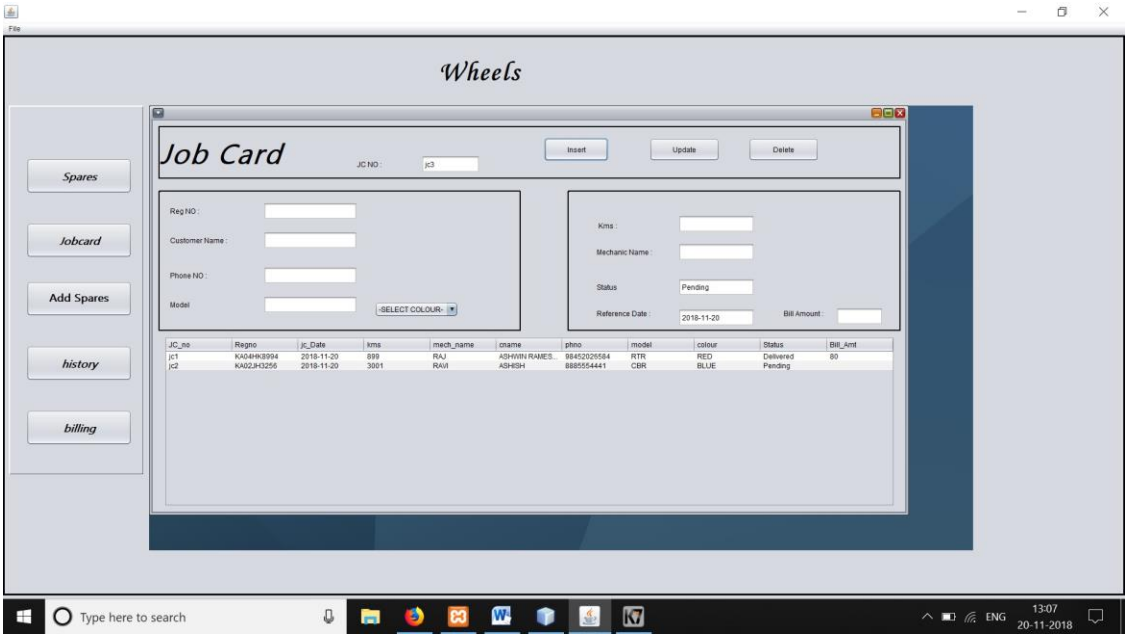

Figure 4.19: billed and triggered to set status as delivered

The above figure 4.19 represents the status of the bill which gets delivered from pending by a trigger.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

As a Database Administrator through the design and development Wheels-Service Centre package, we are able to organize all the information necessary for the spares inventory as well as to maintain a record for each service done at the Service Centre. The complete and proper implementation of all the required details will lead to the effective management a vehicle service and simplify the process of checking the history of service done.

Future works:

- The project shall host the platform on online servers to make is accessible worldwide.
- The project shall integrate multiple load balancers to distribute loads on system.
- The project shall include a master-slave database structure to reduce overload on databases on regular basis on different servers.

BIBLIOGRAPHY

[1] Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson

[2] www.stackoverflow.com

[3] www.github.com

[4] www.scribd.com

[5] www.tutorialpoint.com

[6] Database management systems, Ramakrishnan, and Gehrke, 3rd Edition 2014, McGraw Hill

[7] Jim Keogh: J2EE-TheCompleteReference, McGraw Hill, 2007