**ABSTRACT**

The Shopping Centre Inventory Management system is a stand-alone application. It provides a user-friendly, interactive Graphical User Interface (GUI) based on Java’s Swing components. All data is stored in an Oracle database 11g. The application uses a thin Oracle JDBC driver to communicate with the database.

It is used by the Shopping Centre Database Manager to manage the stocks that has been purchased, in active years.

It is used by the Database Administrators to store and organize the data regarding the monthly stock of products that has been purchased for Shopping Centre. It is also used by them to generate graphs which gives insight into trending products by Data Analysis, whether and when product is in demand.

It is used by the Admin to configure and maintain all data, relevant and critical, for the application to run.

**ACKNOWLEDGMENT**

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without the specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

I would like to profoundly thank **Management** of **RNS Institute of Technology** for providing such a healthy environment to carry out this Project work.

I would like to thank our beloved Director **Dr. H N Shivashankar** for his confidence filling words and support for providing facilities throughout the course.

I would like to express my thanks to our Principal **Dr. M K Venkatesha** for his support and for inspiring me towards the attainment of knowledge.

I wish to place on record my words of gratitude to **Dr. M V Sudhamani,** Professor and Head of the Department, Information Science and Engineering, for being the enzyme and master mind behind my Project work.

I would like to express my profound and cordial gratitude to my Faculty in charge and guide **Mr. R Rajkumar**, Assistant Professor, Department of Information Science and Engineering for his valuable guidance, constructive comments and continuous encouragement throughout the Project work and also for his valuable guidance in preparing the project report.

I would like to thank all other teaching and non-teaching staff of Information Science & Engineering who have directly or indirectly helped me to carry out the project work specially my guide **Mr. T S Bhagavath Singh**.

And lastly, I would hereby acknowledge and thank my parents who have been a source of inspiration and also instrumental in carrying out this Project work.

**ANURAG VERMA**

**USN: 1RN15IS014**

**TABLE OF CONTENTS**

**CERTIFICATE**

**ABSTRACT i**

**ACKNOWLEDGMENT ii**

**TABLE OF CONTENTS iii**

**LIST OF FIGURES iv**

**ABBREVIATIONS v**

**1. INTRODUCTION 1**

1.1 Background 1

1.1.1 Purpose 1

1.1.2 Scope 1

1.1.3 Motivation 1

1.2 Introduction about the project 2

**2. E-R AND RELATIONAL SCHEMA DESIGN 3**

**3. SYSTEM DESIGN 7**

3.1 Table Descriptions 7

**4. IMPLEMENTATION 10**

4.1 Front end and back end used 10

4.1.1 Front end 10

4.1.2 Back end 10

4.2 Discussion of code segments 11

4.3 Applications of project work 13

4.4 Discussion of the results 14

**5. CONCLUSION AND FUTURE ENHANCEMENTS 18**

**REFERENCES 19**

**LIST OF FIGURES**

**Fig. No. Descriptions Page**

Fig. 2.1 E-R diagram of SCIMS 05

Fig. 2.2 Schema diagram of SCIMS 06

Fig 4.1 Snapshot of Login Window 14

Fig 4.2 New Account Signup form 14

Fig 4.3 Admin Options Window 15

Fig 4.4 Categories Window 15

Fig 4.5 Product View Window 16

Fig 4.6 Customers Details Entry Window 16

Fig 4.7 Suppliers Details Entry Window 17

Fig 4.8 Staffs Details Entry Window 17

**ABBREVIATIONS**

SCIMS - Shopping Centre Inventory Management System

ER - Entity Relationship

SQL - Structured Query Language

PL/SQL - Programming Language SQL

ID - Identifier

RDBMS - Relational Database management system

GUI - Graphical User Interface

SDK - Standard Development Kit

JDBC - Java Database Connectivity

API - Application Program Interface

OOP - Object Oriented Programming

JVM - Java Virtual Machine

IDE - Integrated Development Environment

PC - Personal Computer

****

**INTRODUCTION**

**1.1 Background**

**1.1.1 Purpose**

The Shopping Centre Inventory Management system is a stand-alone application. It provides a user-friendly, interactive Graphical User Interface (GUI) based on Java’s Swing components. All data is stored in an Oracle database. The application uses a thin Oracle JDBC driver to communicate with the database. It is used by the Shopping Centre Database Manager to manage the stocks that has been purchased, in active years.

It is used by the Database Administrators to store and organize the data regarding the monthly stock of products that has been purchased for Shopping Centre. It is also used by them to generate graphs which gives insight into trending products by Data Analysis, whether and when product is in demand. It is used by the Admin to configure and maintain all data, relevant and critical, for the application to run.

* + 1. **Scope**

The Shopping Centre Inventory Management System has a scope of three levels of users. At the first level, the ADMIN is allowed to maintain data about staff, customers, suppliers and user authentication, relevant and critical, for the application to run He may also backup the Stocks to an Excel spreadsheet. At the second level, the STAFF are allowed to deal with Customers and other Stuffs for each product category allocated to them, in the respective shops assigned to them. A detailed Insight into trends of products report can be obtained at any point of time.

At the third level, the products in stocks can be viewed by the staff as part of the self-service. This application restricts its scope to being a stand-alone application It can further be enhanced by making it Web-based.

**1.1.3 Motivation**

The main motivation behind the selection of this project was to implement something which will be user-friendly and challenging to design and develop in core Java.

**1.2 Introduction to project title**

The Standalone application can be adapted by any Shopping Centre to keep their stocks organized. The project is aimed at providing the Shopping Centre Manager with a platform to manage stocks. It also provides the manager with a platform to be informed about the ongoing trends. The Shopping Centre Inventory management system will also provide the Manager with a platform to choose Salesman of the year.

**Chapter 2**

**E-R AND RELATIONAL SCHEMA DIAGRAM**

**2.1 Entity-relationship Diagram**

An entity relationship diagram, usually referred to as an e-r diagram represents the attributes, entities and relationships in a relational schema design.

* Entity types like Products and Staffs are in rectangular boxes. Relationships like Bought\_by and Supplies are in diamond boxes, attached to entity types with straight lines.
* Attributes are shown in ovals, each attached by a straight line to entity or relationship type.
* Multivalued attributes (like Administrator) are in double ovals.
* Key attributes (like Staff\_Id) are underlined.
* Component attributes of a composite attribute are attached to oval representing it.
* Derived attributes (like New\_Sell\_Price) are in dotted ovals.
* The cardinality ratios are as follows:

1. Products: Stocks is of cardinality 1:N as one stock can have many same products and 1 product can have more than 1 stock. There is total participation of both entities as :
   * A Product cannot exist without belonging to a Stock.
   * A Stock cannot exist without Products that belong to it.
2. Products: Customers is of cardinality N:M as N products can have M customers to buy it at a time.

There is partial participation of both entities as:

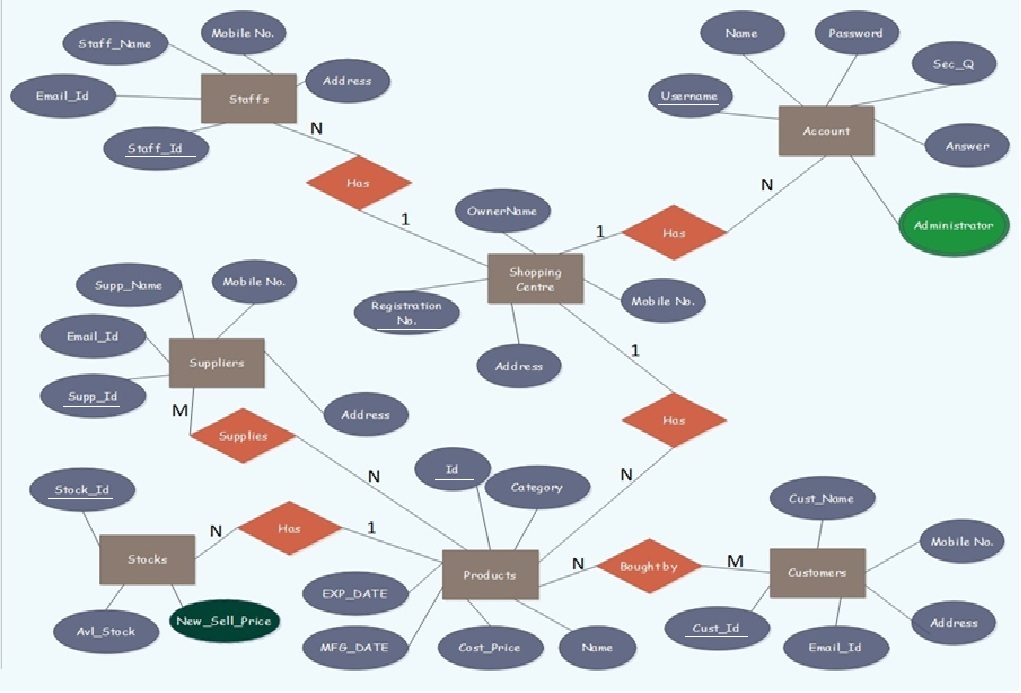
* + A Product can exist without a Customer Buying it.
  + A Customer cannot exist without Products in the shopping centre.

1. Products: Suppliers is of cardinality N:M as a Product can have more than 1 Supplier for more than 1 product. There is total participation of both entities as
   * A Product cannot exist without Supplier supplying products.
   * Supplier cannot exist without products being supplied by him.
2. Shopping Centre: Staffs is of cardinality 1:N as a Shopping Centre can have more than 1 Staff.

There is partial participation of both entities as:

* + A Shopping Centre can exist without Staffs.
  + Staffs cannot exist without anything like shopping centre.

1. Shopping Centre**:** Products is of cardinality 1:N as a Shopping centre can have more than 1 Products and a Products are made up for buyer in Shopping Centre. There is total participation of both entities as:
   * A Shopping Centre cannot exist without any Products in it.
   * A Products cannot exist without any Shopping Centre for it.
2. Shopping Centre**:** Account is of cardinality 1:N as a Shopping Centre can have more than 1 Account and a Accounts can be handled by more than 1 staff. There is total participation of both entities as:
   * A Shopping Centre needs to have Database Accounts to handle its Stocks.
   * A Account is made for something like Shopping Centre Database Management.

Fig 2.1 E-R diagram of Shopping Centre Inventory Management System

**2.2 Schema Diagram**

The term database schema refers to the description of the database that includes the database structure and various constraints on the database.

The schema diagram is in turn an illustrative display of the database schema.

The primary keys are underlined and the referential integrity constraints are depicted by arrows pointing to the keys they reference.

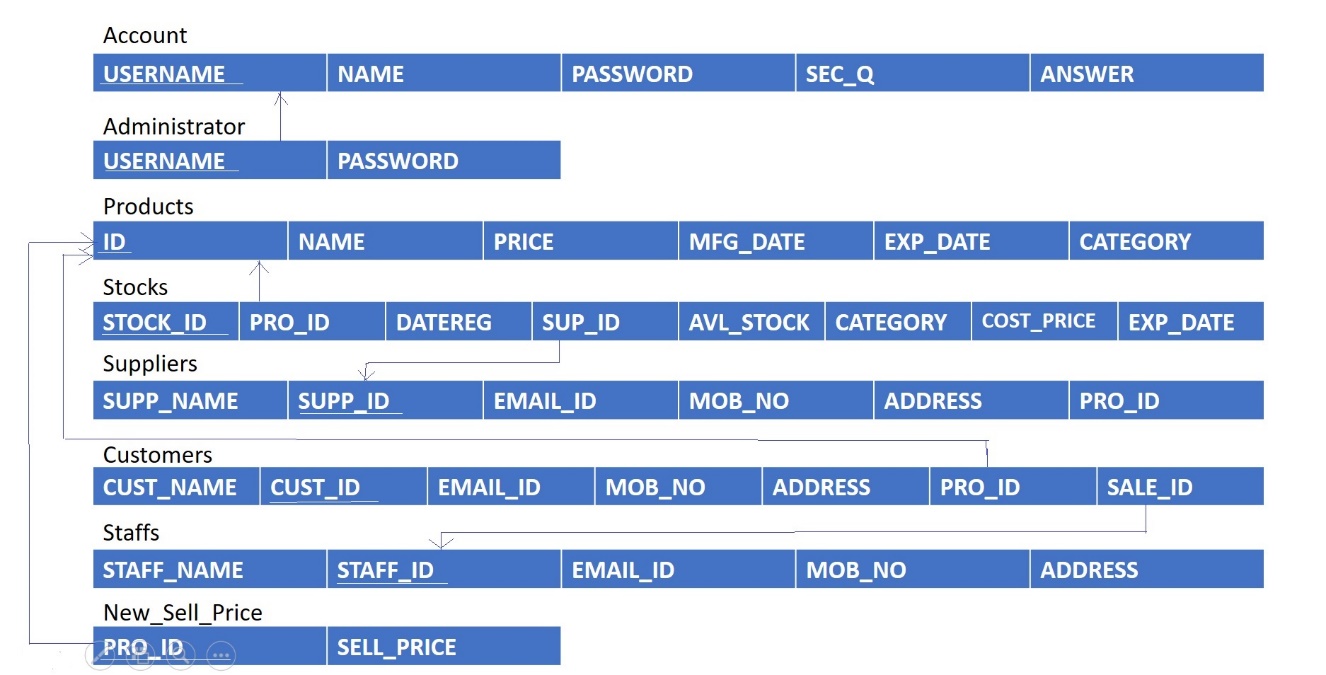


Fig 2.2 Schema diagram of Shopping Centre Inventory management system

**CHAPTER 3**

**SYSTEM DESIGN**

**3.1 Tables description**

In this chapter we describe the different tables used to store all the information regarding the Shopping Centre. A relational database is made up of several components, of which the table is most significant. In relational database terms, a *table* is responsible for storing data in the database. A database consists of one or more tables. Database tables consist of *rows* and *columns*. The SQL script used to create the tables required for the Shopping Centre Inventory Management system is presented in this section.

The parent Tables "ACCOUNT", "SHOPCENTRE", "Products", "Suppliers" and "Customers" must be created first in that order and data must be configured by ADMIN. Later the, the child tables like "NEW\_SELL\_PRICE" must be created and data must be configured by ADMIN. Lastly the "Staffs" must be created by ADMIN so that staff can login and generate and capture products detail. The "ADMINISTRATOR" is created to capture the audit trail of password resets by each user. The PRODUCTS AND STOCKS data is dynamic in nature and gets populated based on the weekly demand of a particular product.

**CREATE TABLE "ACCOUNT"**

( "USERNAME" VARCHAR2(32) NOT NULL ENABLE,

"NAME" VARCHAR2(32),

"PASSWORD" NUMBER(16,0),

"SEC\_Q" VARCHAR2(128),

"ANSWER" VARCHAR2(32),

CONSTRAINT "ACCOUNT\_PK" PRIMARY KEY ("USERNAME") ENABLE

) ;

**CREATE TABLE "ADMINISTRATOR"**

( "USERNAME" VARCHAR2(32) NOT NULL ENABLE,

"PASSWORD" NUMBER(32,0) NOT NULL ENABLE,

CONSTRAINT "ADMINISTRATOR\_PK" PRIMARY KEY ("USERNAME") ENABLE

) ;ALTER TABLE "ADMINISTRATOR" ADD CONSTRAINT "ADMINISTRATOR\_FK2" FOREIGN KEY ("USERNAME")

REFERENCES "ACCOUNT" ("USERNAME") ENABLE;

**CREATE TABLE "CUSTOMERS"**

( "CUST\_NAME" VARCHAR2(32),

"CUST\_ID" NUMBER(5,0) NOT NULL ENABLE,

"EMAIL\_ID" VARCHAR2(32),

"MOB\_NO" NUMBER(10,0),

"ADDRESS" VARCHAR2(128),

"CUST\_IMG" BLOB,

"PRO\_ID" NUMBER(10,0),

"SALE\_ID" NUMBER(10,0),

CONSTRAINT "CUSTOMERS\_PK" PRIMARY KEY ("CUST\_ID") ENABLE

) ;ALTER TABLE "CUSTOMERS" ADD CONSTRAINT "CUSTOMERS\_PID" FOREIGN KEY ("PRO\_ID")

REFERENCES "PRODUCTS" ("ID") ON DELETE CASCADE ENABLE;ALTER TABLE "CUSTOMERS" ADD CONSTRAINT "CUSTOMERS\_SID" FOREIGN KEY ("SALE\_ID")

REFERENCES "STAFFS" ("STAFF\_ID") ON DELETE CASCADE ENABLE;

**CREATE TABLE "PRODUCTS"**

( "ID" NUMBER(8,0) NOT NULL ENABLE,

"NAME" VARCHAR2(32),

"PRICE" NUMBER(12,0),

"MFG\_DATE" DATE,

"EXP\_DATE" DATE,

"IMAGE" BLOB,

"CATEGORY" VARCHAR2(24),

CONSTRAINT "PRODUCTS\_PK" PRIMARY KEY ("ID") ENABLE

) ;

**CREATE TABLE "SHOPCENTRE"**

( "OWNERNAME" VARCHAR2(32) NOT NULL ENABLE,

"MOBILE\_NO" NUMBER(10,0) NOT NULL ENABLE,

"ADDRESS" VARCHAR2(128),

"REGISTRATION\_NO" NUMBER(10,0),

CONSTRAINT "SHOPCENTRE\_PK" PRIMARY KEY ("MOBILE\_NO") ENABLE

) ;

**CREATE TABLE "STAFFS"**

( "STAFF\_NAME" VARCHAR2(32),

"STAFF\_ID" NUMBER(5,0) NOT NULL ENABLE,

"EMAIL\_ID" VARCHAR2(32),

"MOB\_NO" NUMBER(10,0),

"ADDRESS" VARCHAR2(128),

"STF\_IMG" BLOB,

CONSTRAINT "STAFFS\_PK" PRIMARY KEY ("STAFF\_ID") ENABLE

) ;

**CREATE TABLE "STOCKS"**

( "STOCK\_ID" NUMBER(5,0) NOT NULL ENABLE,

"PRO\_ID" NUMBER(5,0) NOT NULL ENABLE,

"DATEREG" DATE,

"SUP\_ID" NUMBER(5,0),

"AVL\_STOCK" NUMBER(5,0),

"CATEGORY" VARCHAR2(32),

"SELL\_PRICE" NUMBER,

"EXP\_DATE" DATE,

CONSTRAINT "STOCKS\_PK" PRIMARY KEY ("STOCK\_ID") ENABLE

) ;ALTER TABLE "STOCKS" ADD CONSTRAINT "STOCKS\_FK" FOREIGN KEY ("PRO\_ID")

REFERENCES "PRODUCTS" ("ID") ON DELETE CASCADE ENABLE;ALTER TABLE "STOCKS" ADD CONSTRAINT "STOCKS\_FK2" FOREIGN KEY ("SUP\_ID")

REFERENCES "SUPPLIERS" ("SUPP\_ID") ON DELETE CASCADE ENABLE;

**CREATE TABLE "SUPPLIERS"**

( "SUPP\_NAME" VARCHAR2(32),

"SUPP\_ID" NUMBER(5,0) NOT NULL ENABLE,

"EMAIL\_ID" VARCHAR2(32),

"MOB\_NO" NUMBER(10,0),

"ADDRESS" VARCHAR2(128),

"SUPP\_IMG" BLOB,

CONSTRAINT "SUPPLIERS\_PK" PRIMARY KEY ("SUPP\_ID") ENABLE

) ;

**CREATE TABLE "NEW\_SELL\_PRICE"**

( "PRO\_ID" NUMBER(32,0) NOT NULL ENABLE,

"SELL\_PRICE" NUMBER(32,0)

) ;

**CHAPTER 4**

**IMPLEMENTATION**

**4.1 Front end and back end used**

**4.1.1 Front-end: JAVA**

The programming language used for the development work is Java. The reason for selection of this language includes among many others the following few.

* In the development of complex, interactive applications, Java can be used to embed rich GUI using swing components.
* Java enables designers to reuse, the portable code.
* Java enables designer to use rich form elements. There will be different types of text inputs and different Swing components implemented for different purposes. Some of the form elements used in this SAS project are JTextField, JComboBox, JButtons, JPasswordField, JLabel, JTable, JFileChooser, JPanel, JFrame, JScrollPane, JOptionPane. The Layouts used are panel and frames are GridLayout and BorderLayout.

**4.1.2 Backend: PL/SQL Oracle**

Oracle is a relational data base management system (RDBMS) that runs as a server providing multi-user access to a number of databases.

* Oracle 11G Express Edition is an open source tool.
* Oracle is a popular choice of database for use in web and stand-alone applications.
* Oracle 11G Express Edition provides web-based tools to administer databases and manage data contained within.
* Oracle 11G Express Edition provides an ease in creating tables by a graphical as well as query based interface.
* Oracle PL/SQL Function and Database Triggers are executed based on the functionality.
* SQL statements are executed against one or more database tables based on the front end GUI functionality.

Oracle implements the following feature, which some other RDBMS systems may not:

* Commit grouping, gathering multiple transactions from multiple connections together to increase the number of commits per second.

**4.2 Discussion of Code Segments**

**4.2.1 Code to establish connection with the Database**

import java.sql.\*;

import javax.swing.JOptionPane;

public class javaconnect {

Connection conn;

public static Connection ConnecrDb(){

try{

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection conn=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","ShopNew","#rockstar08");

return conn;

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e);

return null; }}}

The above code establishes connection with the database by taking into account the username and password for the Oracle SQL account and also the name of the database it is trying to establish the connection with

**4.2.2 Stored Procedure**

A stored procedure is a set of Structured Query Language (SQL) statements with an assigned name, which are stored in a relational database management system as a group, so it can be reused and shared by multiple programs.

CREATE OR REPLACE PROCEDURE GIVE\_DISCOUNT  
IS  
Begin  
UPDATE products SET price=price\*.95;  
END;​

The stored procedure used in the Standalone application program will operate on the products table. Every time a Stock table is Discounted it will update the products Sell\_Price, the procedure will be called and will access the Sell\_Price in products table and will update its value after discount calculation.

**4.2.3 Trigger**

A **database trigger** is procedural code that is automatically executed in response to certain events on a particular table or view in a database.

CREATE OR REPLACE TRIGGER "STOCK\_T1"

after update of Sell\_Price on Stocks

for each row

BEGIN

insert into New\_Sell\_Price(Pro\_Id,Sell\_Price)

values(:old.Pro\_Id,:new.Sell\_Price);

END;

/

ALTER TRIGGER "STOCK\_T1" ENABLE;

The trigger used in the application will perform its operations every time an update is carried out on the Sell\_Price of Stocks table.

**4.2 Applications**

* This project can be implemented by Shopping Centre to provide an organized management of products stock in inventory.
* This Standalone Application can help in getting insight into the trends of market demand.
* This application also lets the Salesman search for products easily.
* This application also aims at providing logs of price changes of product.
* The application provides the Manager to find its best Salesman and reward him.
* The computerized maintenance of stocks can help the problem of overstocking.
* In the future an android app can be launched too to allow the Manager to stay updated with the activities inside Inventory.
* Upon future enhancement the Manager can also get report generated about products to be purchase or not.

**4.4 Discussion of results**

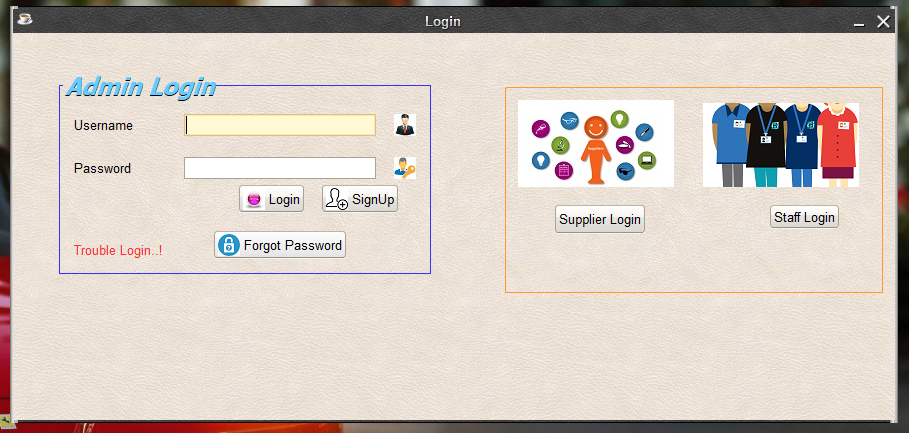
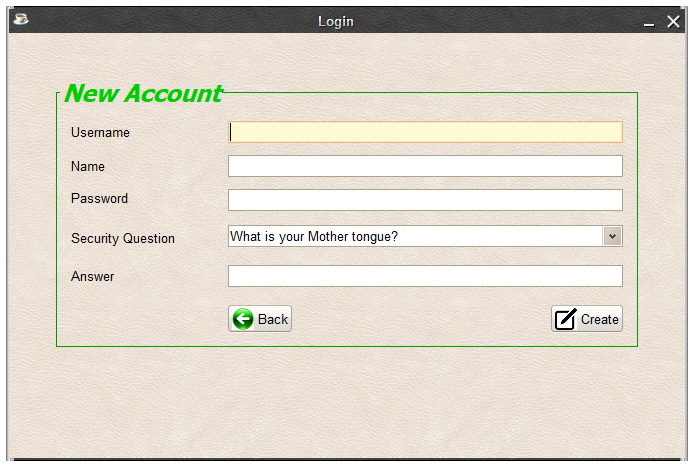


Fig 4.1 Snapshot of Login Window

The login form authenticates the user based on his login name and password and prints “Successfully logged in” if the entered details tally with the details in the database.

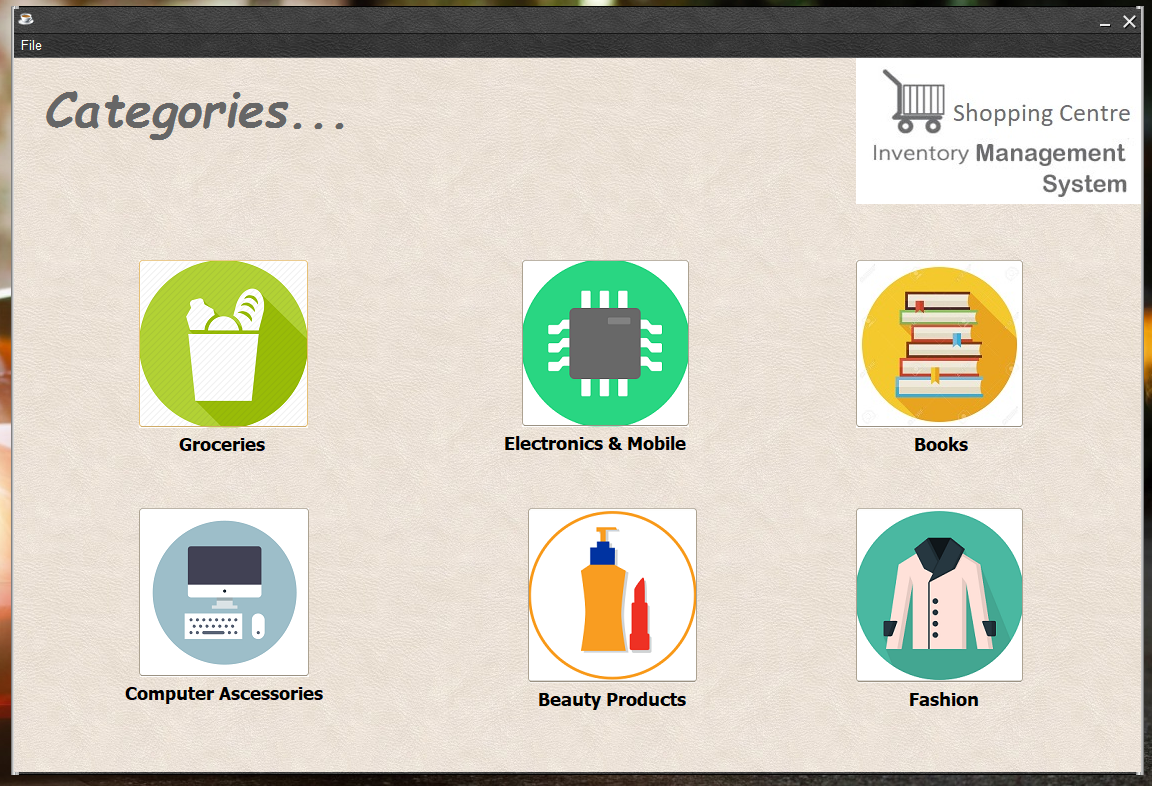
Fig 4.2 New Account Signup form

The New Account Form enables the user based to create a new account to signup him.



Fig 4.3 Admin Options Window

Admin Options Window enables the Admin to view, insert, update and delete various options related to Shopping Centre.

Fig 4.4 Categories Window

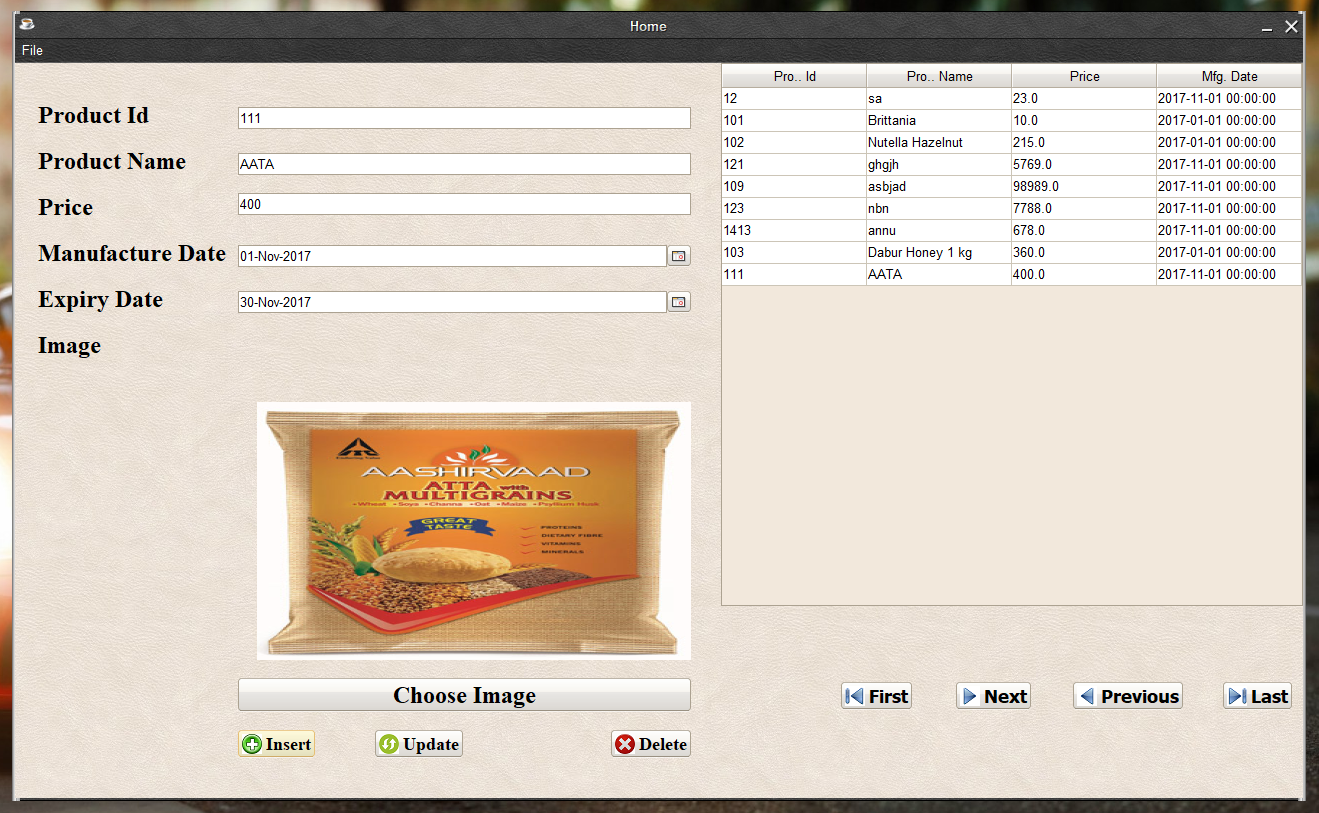
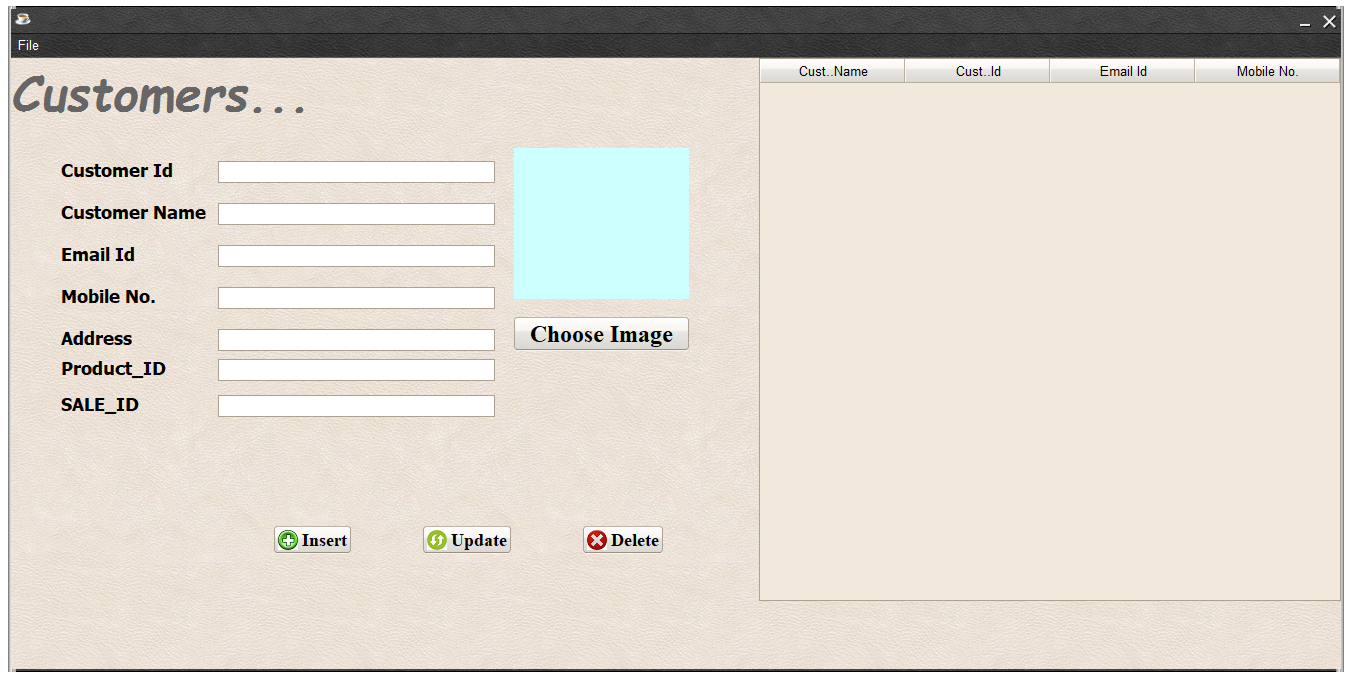


Fig 4.5 Products View Window

Products View window enables one to view, update, insert and delete products currently in stock.

Fig 4.6 Customer Details Entry Window

Customer entry window allows salesman to enter Customer details and product bought.

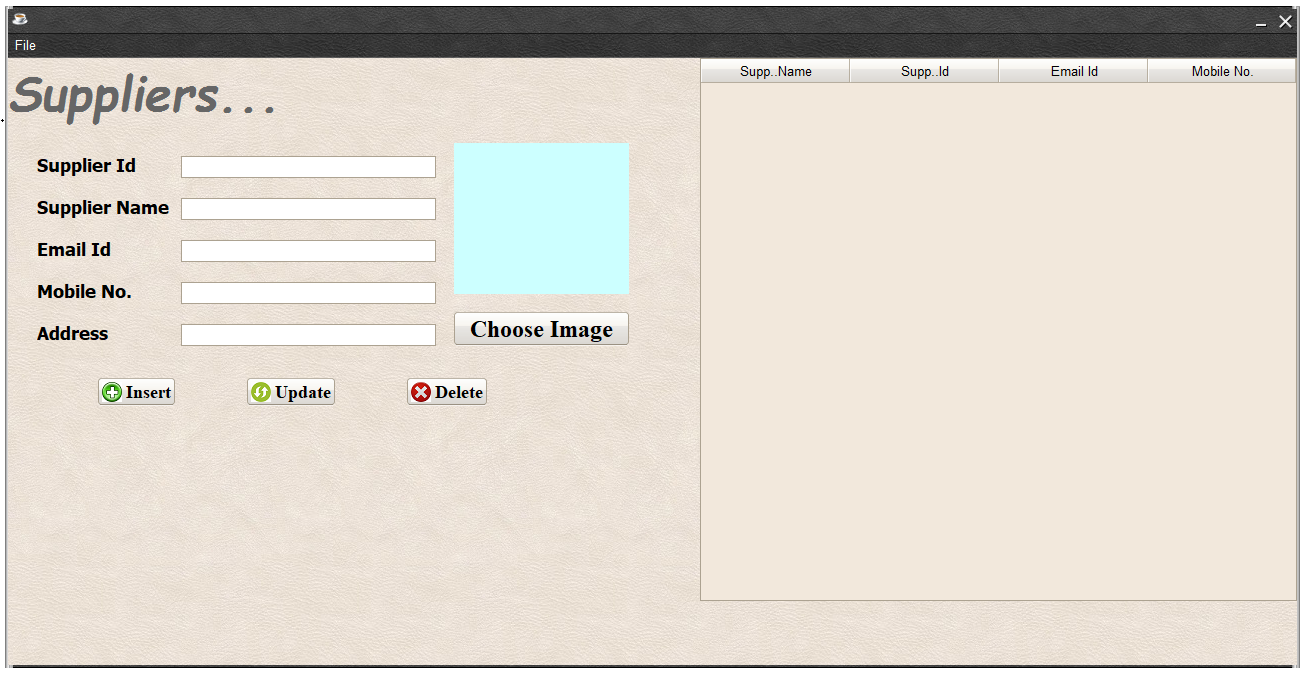


Fig 4.7 Supplier Details Entry Window

Supplier detail entry window allows admin to enter the details of supplier from whom particular product is bought.

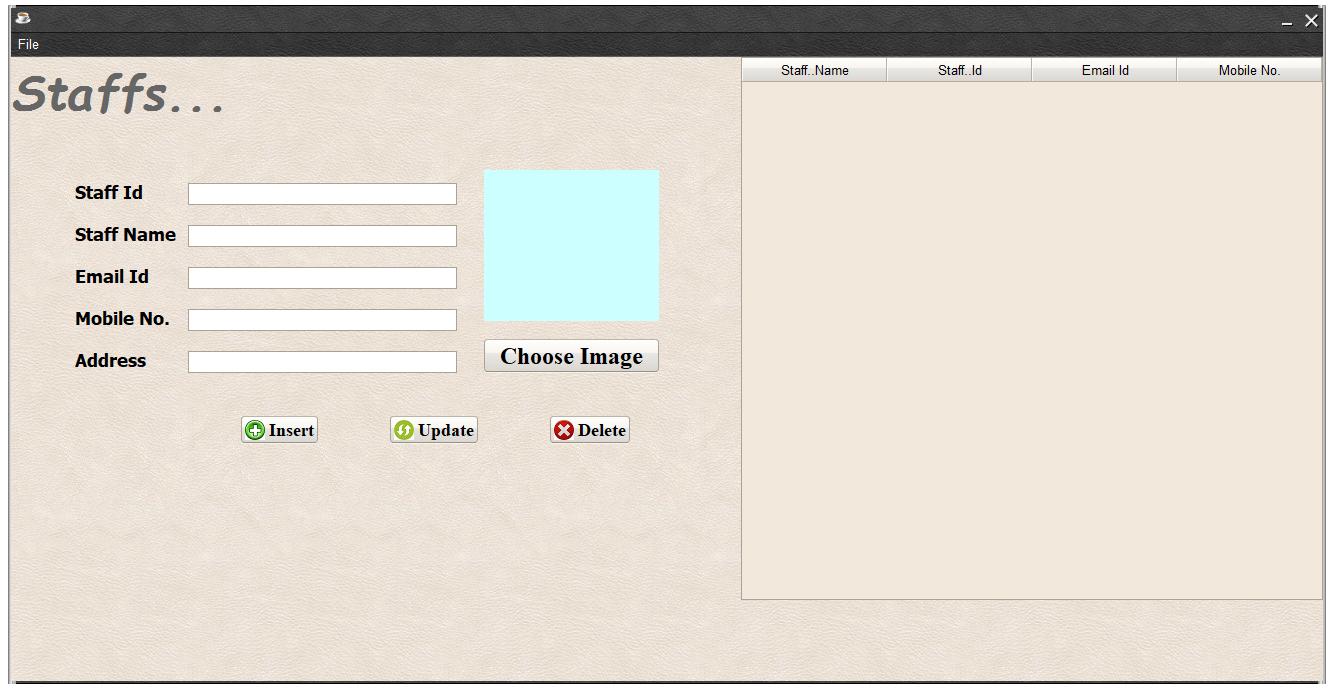


Fig 4.8 Staffs Details Entry Window

Staffs details entry window allow admin to enter the details of Staffs working in the Shopping Centre.

**CHAPTER 5**

**CONCLUSION AND FUTURE ENHANCEMENTS**

A Good inventory strategy sophisticates the administration to take better inventory control decisions. An inventory control decides and manages about when to replenish the items and how much it should be replenished. A good inventory let us know how much in quantity products should be ordered and when to order them and whether those products are in demand. It helps Shopping Centre from overstocking.

**Future enhancements**

There is always a room for improvement in any software package, however good and efficient it may be. But the improvement thing is that the system should be flexible enough for further modifications.

Considering this important factor, the system is designed in such a way that provisions can be given for further enhancement without affecting the system presently developed.

* It can be tightly integrated with cloud for:-
* Tracking inventory in real time
* Fast Deployment
* Easy Integration
* Enhanced Efficiency
* Improved Coordination

**REFERENCES**

**BOOKS**

1. Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, McGRAW HILL, 3rd Edition.
2. Ramez Elmasri and Shamkant B. Navathe , Fundamentals of Database Systems , Pearson , 7th Edition
3. Herbert Schildt, Java:The Complete Reference , McGRAW HILL , 7th Edition.

**WebPage Links**

1. For Database and Java SDK: https://www.oracle.com/index.html
2. For NetBeans IDE 8.2 : https://netbeans.org/downloads/
3. SQL Essentials: https://www.essentialsql.com/
4. Introduction to SQL :

https://lagunita.stanford.edu/courses/DB/SQL/SelfPaced/courseware/ch-sql/seq-vid- introduction\_to\_sql/

1. Intro to SQL: Querying and managing data: https://www.khanacademy.org/computing/computer-programming/sql