

**DBMS Lab Project Report**

**(Inventory Management System)**

**Submitted By:**

Muhammad Haseeb Akbar 03-134202-052

Shahroz Ashraf 03-134202-080

Ameer Humza 03-134202-043

**Submitted to:**

Mam Sadia Aslam

**Section:**

BS (CS) 4B

**Abstract**

The inventory management system allows

user to have control on the products and

brands to fulfill customer requirements and

make them easy to handle all information and

records.

**Table of Contents**

[**Abstract** 1](file:///E:\HARIS%20DATA\New%20folder\DBMS\project\final%20report.docx#_Toc75111544)

[**Introduction** 2](file:///E:\HARIS%20DATA\New%20folder\DBMS\project\final%20report.docx#_Toc75111545)

[**Project Scope** 3](file:///E:\HARIS%20DATA\New%20folder\DBMS\project\final%20report.docx#_Toc75111546)

[**Background Study** 4](file:///E:\HARIS%20DATA\New%20folder\DBMS\project\final%20report.docx#_Toc75111552)

[**Methodology** 5](file:///E:\HARIS%20DATA\New%20folder\DBMS\project\final%20report.docx#_Toc75111552)

[**ERD** 6](file:///E:\HARIS%20DATA\New%20folder\DBMS\project\final%20report.docx#_Toc75111572)

[**Schema** 7](file:///E:\HARIS%20DATA\New%20folder\DBMS\project\final%20report.docx#_Toc75111573)

[**Conclusion** 8](file:///E:\HARIS%20DATA\New%20folder\DBMS\project\final%20report.docx#_Toc75111574)

**Inventory Management System**  **Project**

**Database Management System**

**Introduction**

Project is related to Inventory Management System

The project maintains three levels of users:-

* Billing Counter Level
* Manager Level
* Owner Level

Main facilities available in this project are:-

* We can forecast the sales by analyzing the previous sales statistics.
* We can get an idea that when we need to order new inventory.
* We can reduce the chances of any kind of frauds done by the staff members

in the inventory.

* Customer details can be added.
* Invoice generation.

We can keep a track of transactions received through different payment methods.

An inventory management system is the combination of technology (hardware and software) and processes and procedures that oversee the monitoring and maintenance of stocked products, whether those products are company assets, raw materials and supplies, or finished products ready to be sent to vendors or end consumers. This system can widely be used by normal shops, departmental stores or MNCs for keeping a proper track of the stock. It also consists of information like manager details, customer details etc. With the help of this system we can fix a minimum quantity of any inventory below which we need to place an order for that inventory. This will help us in good sales results and never the out of stock stage for any inventory.

**SCOPE:**

This will help us in maintain the exact count of any product. ¬ Can help us to set minimum quantity of any product below which we can order the product from manufacturer. ¬ Can reduce duplicate entries

**Background:**

This application is nowadays a basic use of any company, firm, shop, or departmental store because stock maintenance, stock forecasting are some things which are very essential these days for earning great profits.

In ancient times we need to maintain the complete inventory in paper pen method.

The ancient method is quite un-easy, uncomfortable, and sometimes inaccurate. For overcoming this problem, we came with a solution of inventory management system. From this system we can generate invoice for every purchase. In addition to this we can have the employee details, customer details in this system.

In short, we can call this as all-in-one system...!!

**Purpose:**

INVENTORY MANAGEMENT must tie together the following objectives ,to ensure that there is continuity between functions :

• Company’s Strategic Goals • Sales Forecasting • Sales & Operations Planning • Production & Materials Requirement Planning.

Inventory Management must be designed to meet the dictates of market place and support the company’s Strategic Plan . The many changes in the market demand , new opportunities due to worldwide marketing , global sourcing of materials and new manufacturing technology means many companies need to change their Inventory Management approach and change the process for Inventory Control . Inventory Management system provides information to efficiently manage the flow of materials , effectively utilize people and equipment , coordinate internal activities and communicate with customers . Inventory Management does not make decisions or manage operations, they provide the information to managers who make more accurate and timely decisions to manage their operations. INVENTORY is defined as the blocked Working Capital of an organization in the form of materials . As this is the blocked Working Capital of organization, ideally it should be zero. But we are maintaining Inventory . This Inventory is maintained to take care of fluctuations in demand and lead time. In some cases it is maintained to take care of increasing price tendency of commodities or rebate in bulk buying.

**Goals of proposed system:**

1. Planned approach towards working: - The working in the organization will be well planned and organized. The data will be stored properly in data stores, which will help in retrieval of information as well as its storage.

2. Accuracy: - The level of accuracy in the proposed system will be higher. All operation would be done correctly and it ensures that whatever information is coming from the center is accurate.

3. Reliability: - The reliability of the proposed system will be high due to the above stated reasons. The reason for the increased reliability of the system is that now there would be proper storage of information.

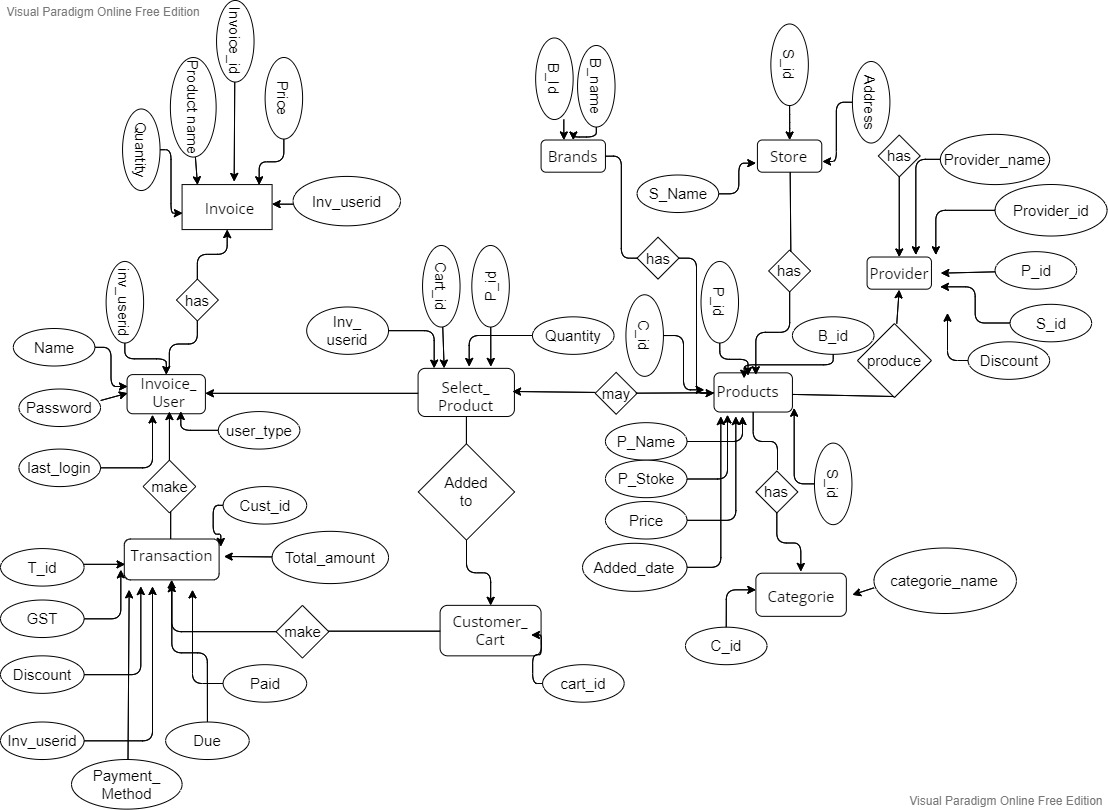
4. No Redundancy: - In the proposed system utmost care would be that no information is repeated anywhere, in storage or otherwise. This would assure economic use of storage space and consistency in the data stored.

5. Immediate retrieval of information: - The main objective of proposed system is to provide for a quick and efficient retrieval of information.

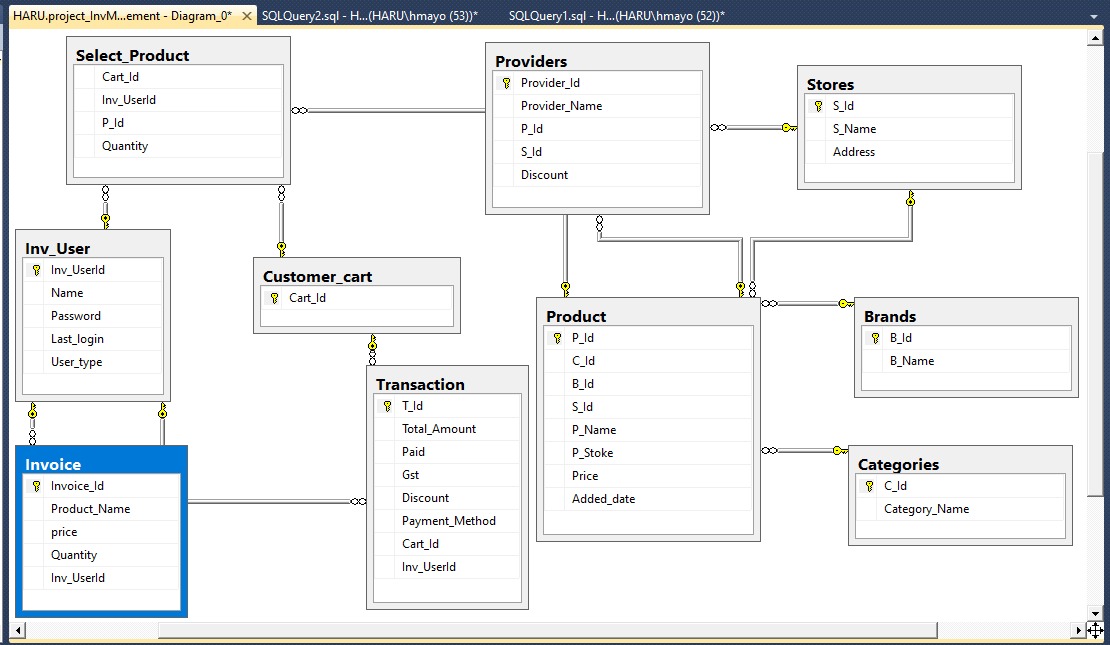
6. Immediate storage of information: - In manual system there are many problems to store the largest amount of information.

7. Easy to Operate: - The system should be easy to operate and should be such that it can be developed within a short period of time and fit in the limited budget of the user

**ERD:**





****

In this project we’ve only implemented the back end of the system which is designed

on “SQL Plus”

On this sequence query language, we created 10 tables named:

1. Brands

1. inv\_user

1. Categories

1. Products

1. Stores

1. Providers

1. Customer\_cart

1. Select\_product

1. Transaction

1. Invoice

**SQL CODE IMPLEMENTATION:**

create database project\_InvManagement;

use project\_InvManagement;

create table Brands

(B\_Id int primary key,

B\_Name nvarchar(50)

);

insert into Brands values(1,'Dell');

insert into Brands values(2,'Lenove');

insert into Brands values(3,'IBM');

insert into Brands values(4,'Apple');

insert into Brands values(5,'Bata');

insert into Brands values(6,'Stylo');

insert into Brands values(7,'Adidas');

insert into Brands values(8,'Nike');

insert into Brands values(9,'Khaadi');

insert into Brands values(10,'Generation');

select \*from Brands;

create table Inv\_User

(Inv\_UserId int primary key,

Name nvarchar(50),

Password nvarchar(50),

Last\_login datetime default (getdate()),

User\_type nvarchar(50)

);

insert into Inv\_User (Inv\_UserId,Name,Password,User\_type ) values(11,'Shehroz','7856', 'Buyer');

insert into Inv\_User (Inv\_UserId,Name,Password,User\_type ) values(12,'Hamza','1122', 'Buyer');

insert into Inv\_User (Inv\_UserId,Name,Password,User\_type ) values(13,'Haseeb','2233', 'Buyer');

insert into Inv\_User (Inv\_UserId,Name,Password,User\_type )values(14,'Ali Arif','3344','Buyer');

select \*from Inv\_User;

create table Categories

(C\_Id int primary key,

Category\_Name nvarchar(50)

);

insert into Categories values(21,'Electronic');

insert into Categories values(22,'Machinery');

insert into Categories values(23,'Accessories');

insert into Categories values(24,'Cables');

insert into Categories values(25,'Shoes');

insert into Categories values(26,'Clothes');

select \*from Categories;

Create table Stores

(S\_Id int primary key,

S\_Name nvarchar(50),

Address nvarchar(50),

);

insert into Stores values(31,'Amazone','Lahore');

insert into Stores values(32,'Daraz','karachi');

insert into Stores values(33,'OLX','Multan');

insert into Stores values(34,'Ali baba','Quetta');

insert into Stores values(35,'Big Bazar','Raiwind');

select \*from Stores;

create table Product

(P\_Id int primary key,

C\_Id int foreign key references Categories (C\_Id) on delete set null on update set null,

B\_Id int foreign key references Brands (B\_Id)on delete set null on update cascade,

S\_Id int foreign key references Stores (S\_Id) on delete set null on update cascade,

P\_Name nvarchar(50),

P\_Stoke int,

Price int,

Added\_date datetime default (getdate())

);

insert into Product (P\_Id,C\_Id,B\_Id,S\_Id,P\_Name,P\_Stoke,Price) values(41,23,1,31,'Keyboard',76,299);

insert into Product (P\_Id,C\_Id,B\_Id,S\_Id,P\_Name,P\_Stoke,Price) values(42,23,3,32,'Mouse',59,99);

insert into Product (P\_Id,C\_Id,B\_Id,S\_Id,P\_Name,P\_Stoke,Price) values(43,22,2,35,'PC desktop',42,5999);

insert into Product (P\_Id,C\_Id,B\_Id,S\_Id,P\_Name,P\_Stoke,Price) values(44,24,1,34,'Power Supply',80,90);

insert into Product (P\_Id,C\_Id,B\_Id,S\_Id,P\_Name,P\_Stoke,Price) values(45,25,7,33,'RACER TR21 SHOES',89,299);

insert into Product (P\_Id,C\_Id,B\_Id,S\_Id,P\_Name,P\_Stoke,Price) values(46,25,6,32,'Golden Color Fancy Sandals FN4864',19,199);

insert into Product (P\_Id,C\_Id,B\_Id,S\_Id,P\_Name,P\_Stoke,Price) values(47,26,9,31,'Classic Kameez Printed Embroidered',11,999);

insert into Product (P\_Id,C\_Id,B\_Id,S\_Id,P\_Name,P\_Stoke,Price) values(48,26,10,34,'FABRIC Shirt: Lawn (100% Cotton)',25,1000);

select \*from Product;

create table Providers

(Provider\_Id int primary key,

Provider\_Name nvarchar(50),

P\_Id int foreign key references Product(P\_Id) on delete cascade on update cascade,

S\_Id int foreign key references Stores(S\_Id) on delete set null on update cascade,

Discount int

);

insert into Providers values(51,'Bilal',48,34,25);

insert into Providers values(52,'Huzaifa',43,32,600);

insert into Providers values(53,'Asad',45,35,250);

insert into Providers values(54,'Chand',41,33,0);

insert into Providers values(55,'Asis',47,31,100);

select \*from Providers;

create table Customer\_cart

(Cart\_Id int primary key,

);

insert into Customer\_cart values(61);

insert into Customer\_cart values(62);

insert into Customer\_cart values(63);

insert into Customer\_cart values(64);

insert into Customer\_cart values(65);

select \*from Customer\_cart;

create table Select\_Product

(Cart\_Id int foreign key references Customer\_cart(Cart\_Id) on delete set null on update cascade,

Inv\_UserId int foreign key references Inv\_User(Inv\_UserId) on delete cascade on update cascade,

P\_Id int foreign key references Product (P\_Id) on delete cascade on update cascade,

Quantity int

);

insert into Select\_Product values(62,11,43,1);

insert into Select\_Product values(61,14,45,2);

insert into Select\_Product values(63,12,44,5);

insert into Select\_Product values(65,13,47,3);

select \*from Select\_product;

create table [Transaction]

(T\_Id int primary key,

Total\_Amount int,

Paid int,

Gst int,

Discount int,

Payment\_Method nvarchar(50),

Cart\_Id int foreign key references Customer\_cart(Cart\_Id) on delete set null on update cascade,

Inv\_UserId int foreign key references Inv\_User(Inv\_UserId) on delete cascade on update cascade

);

insert into [Transaction] values(71,5999,5000,999,999,'cash',62,11);

insert into [Transaction] values(72,598,550,48,0,'bank Transaction',61,14);

insert into [Transaction] values(73,450,400,50,50,'cash',63,12);

insert into [Transaction] values(74,2997,2950,47,47,'cash',65,13);

insert into [Transaction] values(75,5999,5000,999,999,'bank Transaction',62,11);

select \*from [Transaction];

create table Invoice

(Invoice\_Id int primary key,

Product\_Name nvarchar(50),

price int,

Quantity int,

Inv\_UserId int foreign key references Inv\_User(Inv\_UserId) on delete cascade on update cascade

);

insert into Invoice values(81,'PC desktop',5000,1,11);

insert into Invoice values(82,'RACER TR21 SHOES',550,2,14);

insert into Invoice values(83,'Classic Kameez Printed Embroidered',2950,3,13);

insert into Invoice values(84,'Power Supply',400,5,12);

select \*from Invoice;

drop table Brands;

drop table Inv\_User;

drop table Categories;

drop table Product;

drop table Stores;

drop table Providers;

drop table Customer\_cart;

drop table Select\_Product;

drop table [Transaction];

drop table Invoice;

***Conclusion***

In this project we developed a complete back end software in which we can update the stock, modify stock, we can forecast the stock, generate invoice. From this application we can get an update that if a particular inventory or stock is less than the some pre-fixed quantity then it’ll be easy for the manager/owner to reorder the product from supplier to overcome the “Out of Stock” stage. In addition to this it can also help us to manage the warehouses, add warehouses which can be proved as very useful feature.

We can have complete customer details which can help us to retrieve the order details of regular customers.

From this program we can also keep a track of transactions performed by different customers/clients. We can also get an idea that how much fund we received from different payment methodologies.