PHARMACY MANAGEMENT SYSTEM

Mini Project Report -Database Lab (DSE 2260)

Department of Data Science & Computer Applications



B. Tech Data Science

4th Semester – Batch: B3

Submitted By

|  |  |
| --- | --- |
| Namrata Dutta | 200968064 |
| Sahaj Jaggi | 200968076 |
| Gaddam Harsha Vardhan Reddy | 200968078 |
| K Krishna Chaitanya Reddy | 200968084 |

**Mentored By**

|  |  |
| --- | --- |
| Vinayak M | Archana H/ Shameem |
| Assistant Professor-Senior | Assistant Professor-Senior |
| DSCA, MIT | DSCA, MIT |

Text

Description automatically generated

Date:

**CERTIFICATE**

This is to certify that the Namrata Dutta (200968064) , Sahaj Jaggi (200968076), Gaddam Harsha Vardhan Reddy (200968078), K Krishna Chaitanya Reddy (200968084), have successfully executed a mini project titled “Pharmacy Management System” rightly brining forth the competencies and skill sets they have gained during the course- Database Lab (DSE 2262 & DSE ), thereby resulting in the culmination of this project.

|  |  |
| --- | --- |
| **Vinayak M** | **Archana H / Shameem** |
| **Assistant Professor-Senior** | **Assistant Professor-Senior** |
| **DSCA, MIT** | **DSCA, MIT** |

**ABSTRACT**

Most pharmacies faced problems such as insufficient service promotions, lack of coherence of pharmacy services in hospitals, poor drug information systems, and the inconsistency of the pharmacy information management due to its manual processes. Now, these are the problems that must be solved with this Pharmacy Management System Project Proposal. This program can be used in any pharmaceutical shops having a database to maintain. The software used can generate reports, as per the user’s requirements. The software can print invoices, bills, receipts etc.

Pharmacists can use the Pharmacy Management System program to help them methodically manage their pharmacies. When a medicine’s name is input, the Pharmacy Management System can help by providing details about the medicine. In large medical stores, manually handling the specifics of all the drugs becomes very tough. We can keep track of all the medicines by using this pharmacy management system. It is updated with new information as new medicines are introduced. It also displays user information and medicines they bought using a username/email and password which makes it efficient and easy to store and retrieve user data.

One of the most important responsibilities of pharmacy management is to supervise and manage the pharmacy employees to ensure healthy working relationships and outcomes. Each of these functions is critical to the pharmacy’s operation and should be improved.

Contents

**1. Introduction 5**

**2. Synopsis 6**

**2.1 Proposed System 6**

**2.2 Objectives 6**

**3. Functional Requirements 8**

**4. Detailed Design 10**

**4.1 ER Diagram 10**

**4.2 Schema Diagram 11**

**4.3 Data Dictionary 11**

**4.4 Relational Model Implementation 13**

**4.5 Queries 19**

**4.7 Triggers 21**

**4.8 Stored Procedures 23**

**4.9 Stored Functions 25**

**5. Result 26**

**6. Conclusion and Future Work 27**

1. **INTRODUCTION**

The pharmacy management system is a system that stores data and enables functionality that organizes and maintains the medication use process within pharmacies. Pharmacy Management is a challenging task in the field of Medicines. This challenge is mainly because of the huge flow of data, drugs and number of users on a daily basis making it difficult to maintain and extract information efficiently. Maintaining a proper database of drugs and users is also important to have a proper healthcare resource.

Pharmacy management system deals with the maintenance of drugs and consumables in the pharmacy unit. The set-up of this pharmacy management system will ensure availability of sufficient quantity of drugs and consumable materials for the patient. This will enhance the efficiency of clinical work and ease patient’s convenience. In addition, Pharmacy management system will be able to process drug prescription with ease.

Pharmacist patient care process consists of five steps: *collect*, *assess*, *plan*, *implement*, and *follow-up.*  Ideally, the pharmacy management system assists with each of these practices. The pharmacy system should **collect** data at intake and continue to store and organize information as the pharmacist learns more about the patient's medications, their history, and other factors that may affect their health. The technology within the pharmacy information system should allow the pharmacists to **assess**the collected information to form a **plan** and **implement**creative strategies that address the patient's issues. After implementing a plan, the pharmacist should routinely **follow-up** with the patient and make adjustments as needed to further progress.

Pharmacists spend most of their working hours dispensing drugs. This task requires lots of concentration, a great deal of verification, drug interaction checking, not to mention making sense of the doctor’s handwriting. With seamless computer-computer communication in place, prescriptions can be easily handled by software, freeing more time for pharmacists to interact with patients. Pharmacies play a pivotal role in helping manage the distribution of controlled dangerous substances (CDSs) by entering all prescription information in the database and checking it when dispensing drugs. Such a pharmacy management system allows one to cut down logging time and effort to just a few clicks as information is automatically added to the patient’s history.

In general, the Pharmacy management system is based on computer technology that gives service for users, managed by the pharmacist who give implementation of function relatively in effective times as well as will design for removing time wasting, saving resources, easy data access of the medicine, security on data input and data access by removing almost manual based system.

1. **SYNOPSIS**

**Proposed System :**

Most pharmacies faced problems such as insufficient service promotions, lack of coherence of pharmacy services in hospitals, poor drug information systems, and the inconsistency of the pharmacy information management due to its manual processes. Through this Pharmacy Management System, we intend to solve these problems. The aim of the project is to create an effective software to help the pharmacist to maintain the records of the medicines, handle user details, generate invoice, check, and renew validity and provide a scope of communication between users by using inbuilt messaging system.

**Objectives:**

* To gain practical experience by modelling a software based on real world problem.
* To develop an application that deals with the day-to-day requirement of any pharmacy.
* To develop the easy management of the medicines (drugs).
* To handle the inventory details like sales details, purchase details
* To provide competitive advantage to the pharmacy.
* To make it easy to extract a user’s information using their id and password.

To manage medicines, modules will assess the need for and use of medication, the patient’s response to medication, and the patient’s level of understanding of the drug and how to take it with the patient.

The modules aid in the reduction of medication errors, the improvement of patient safety, the reporting of drug usage, and the tracking of expenses.

1. **FUNCTIONAL REQUIREMENTS**

Functional requirement helps define the functionality of a system or one of its subsystems. They help clearly define the expected system service and behaviour.

For the pharmacy management system the different functionalities provided are – different login portals for pharmacists, new customer and pharmacist registration and an option to generate a new password if the user has forgotten the password.

**New User Registration**

The user (pharmacist or customer) must be able to create user id and password by supplying appropriate details.

|  |  |
| --- | --- |
| INPUT | New username, Password, phone |
| Processing | The system must check availability of entered valid user name.  Password must follow criteria- minimum 9 char, at least one capital, one number and one special character.  Check for validity of phone number by prompting to enter OTP |
| OUTPUT | User created Successfully message / highlight the information entered which is wrong and allow to renter. |

**Login**

The existing user must be able to login upon entering proper user name and password.

|  |  |
| --- | --- |
| INPUT | username, Password |
| Processing | Check the user name and password against information stored in data storage |
| OUTPUT | If user entered correct user name & Password  Login successful and open main application menu  Else  Display Login not successful, retry logging in |

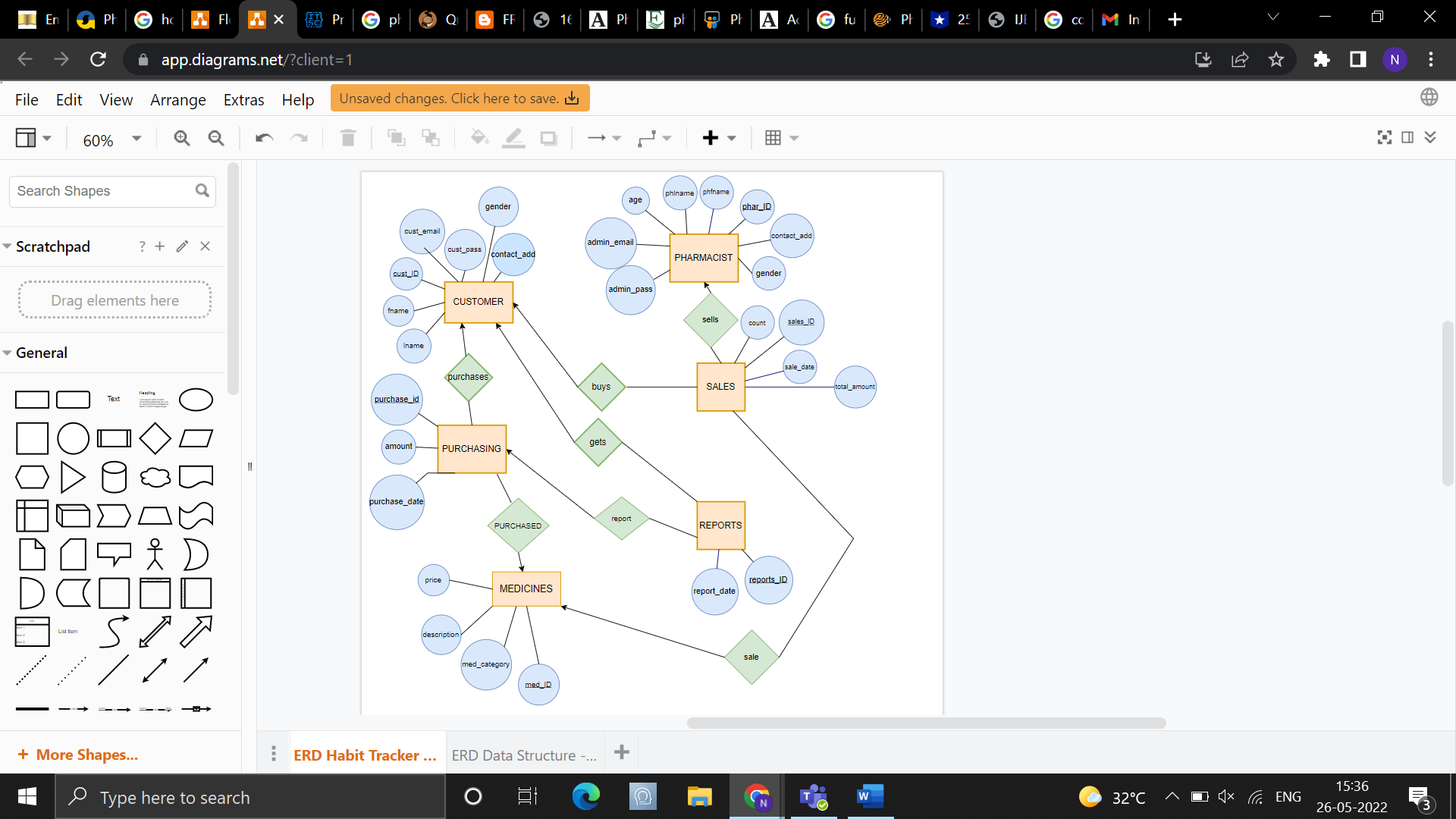
**Forgot password**

If existing user name is not bale to login, forgot password can be used to reset password.

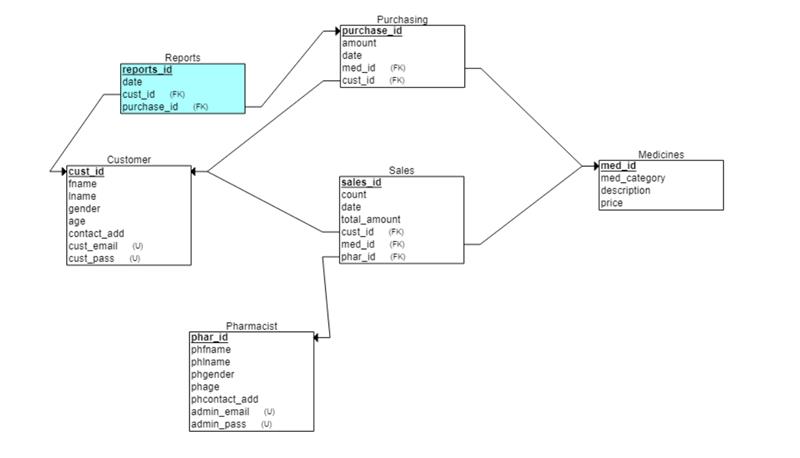
|  |  |
| --- | --- |
| INPUT | Prompt user to enter username, Phone |
| Processing | If username and corresponding phone exist in the data storage  Send OTP to Phone.  Prompt the user to enter OTP  If OTP matching  Prompt user to change password according to criteria.  Else  OTP not matching.  Else  User name and corresponding Phone not existing in the storage |
| OUTPUT | Password successfully changed / User name, phone not matching |

1. **DETAILED DESIGN**

**ER DIAGRAM**



**RELATIONAL SCHEMA**



**DATA DICTIONARY**

**CUSTOMER**

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data type (size) | Constraint | Constraint Name |
| Cust\_id | Varchar2(11) | Primary Key ,  Check like ‘C%’ | Cust\_id\_pkey, starts\_with\_C |
| Fname | Varchar2(90) |  |  |
| lname | Varchar2(90) | >0 |  |
| gender | Char(1) | M or F | Gender\_const |
| Age | Number(3) | >0 | Check\_age |
| Contact\_add | Number(10) |  |  |
| Cust\_email | Varchar2(90) | Check @ and . | Check\_email |
| Cust\_pass | Varchar2(90) | >9 | Check\_pass |

**PHARMACIST**

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data type (size) | Constraint | Constraint Name |
| Phar\_ID | Varchar2(11) | Primary Key  Check like ‘PH%’ | phar\_id\_pkey,  starts\_with\_PH |
| phfname | Varchar2(90) |  |  |
| Phlname | Varchar2(90) |  |  |
| Phgender | Char(1) | M or F | phGender\_const |
| Phage | Number(3) | >0 | phCheck\_age |
| phContact\_add | Number(10) | Unique |  |
| admin\_email | Varchar2(90) | Check @ and . | phCheck\_email |
| admin\_pass | Varchar2(90) | Unique notnull  >9 | phCheck\_pass |

**MEDICINES**

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data type (size) | Constraint | Constraint Name |
| Med\_ID | Varchar2(11) | Primary Key ,  Check like ‘M%’ | phar\_id\_pkey  starts\_with\_M |
| Med\_category | Varchar2(90) |  |  |
| Description | Varchar2(90) |  |  |
| Price | Number(10,2) | >0 | Check\_price |

**PURCHASING**

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data type (size) | Constraint | Constraint Name |
| Purchase\_id | Varchar2(11) | Primary Key  Check like ‘P%’ | purchase\_id\_pkey  starts\_with\_P |
| Cust\_id | Varchar2(90) | References customer | Fk\_cust\_id |
| Med\_id | Varchar2(90) | References medicines | Fk\_med\_id |
| Amount | Number(10,2) | >0 | Check\_amount |
| purchase\_date | Date |  |  |

**SALES**

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data type (size) | Constraint | Constraint Name |
| Sales\_id | Varchar2(11) | Primary Key  Check like ‘S%’ | sales\_id\_pkey  starts\_with\_S |
| Phar\_id | Varchar2(11) | References pharmacist | Fk\_phar\_id |
| cust\_id | Varchar2(90) | References customer | Fk\_cust\_id |
| Med\_id | Varchar2(90) | References medicines | Fk\_med\_id |
| Count | Number(10) |  |  |
| sale\_date | Date |  |  |
| Total\_amount | Number(10,2) | Count\*amount | Tot\_amount |

**REPORTS**

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data type (size) | Constraint | Constraint Name |
| reports\_id | Varchar2(11) | Primary Key  Check like ‘R%’ | purchase\_id\_pkey  starts\_with\_R |
| purchase\_id | Varchar2(11) | References purchasing | Fk\_purchase\_id |
| Cust\_id | Varchar2(11) | References customer | Fk\_cust\_id |
| Date | date |  |  |

**RELATIONAL MODEL IMPLEMENTATION**

**TABLE CREATION :**

CREATE TABLE CUSTOMER (

cust\_id varchar2(11) constraint cust\_id\_pkey primary key, constraint starts\_with\_C CHECK(cust\_id LIKE ‘C%’),

fname varchar2(90),

lname varchar2(90),

gender char(1) constraint gender\_const CHECK(gender in ('M','F')),

age number(3) constraint check\_age CHECK(age > 0),

contact\_add number(10),

cust\_email varchar2(90) constraint check\_email CHECK(cust\_email like '%@%.%') not null,

cust\_pass varchar2(90) unique not null);

CREATE TABLE Pharmacist(

phar\_id varchar2(11) constraint phar\_id\_pkey primary key, constraint starts\_with\_p CHECK(phar\_id LIKE ‘P%’)

phfname varchar2(90),

phlname varchar2(90),

phgender char(1) check phgender\_const check(phgender in ('M','F')),

phage number(3) constraint phcheck\_age check(phage > 0),

phcontact\_add number(10) unique,

admin\_email varchar2(90) constraint phcheck\_email CHECK(admin\_email like '%@%.%') not null,

admin\_pass varchar2(90) unique not null);

CREATE TABLE medicines(

med\_id varchar2(11) constraint med\_id\_pkey primary key, constraint starts\_with\_M CHECK(med\_id LIKE ‘M%’)

med\_category varchar2(90) not null,

description varchar2(90),

price number(10,2) constraint check\_price check(price >0 ));

CREATE TABLE purchasing(

purchase\_id varchar2(11) constraint purchase\_id\_pkey primary key,

cust\_id varchar2 (11) constraint fk\_cust\_id references customer,

med\_id varchar2(11) constraint fk\_med\_id references medicines,

amount number(10,2) constraint check\_amount check(amount > 0),

purchase\_date date);

CREATE TABLE sales(

sales\_id varchar2 (11) constraint sales\_id\_pkey primary key,

phar\_id varchar2 (11) constraint fk\_phar\_id references pharmacist,

cust\_id varchar2 (11) constraint fk\_cust references customer,

med\_id varchar2 (11) constraint fk\_med references medicines,

sales\_count number(10),

sale\_date Date,

total\_amount number(10,2));

CREATE TABLE reports(

reports\_id varchar2 (11) constraint reports\_pk primary key,

purchase\_id varchar2(11) constraint fk\_purchase\_idr references purchasing,

cust\_id varchar2 (11) constraint fk\_cust\_idr references customer,

report\_date date);

**INSERTING THE VALUES:**

*CUSTOMER TABLE*

INSERT INTO CUSTOMER VALUES(‘C0091234545’,'ABHISHEK','REDDY','M',25,7894561230,'ABCD@GMAIL.COM', ‘Abhi@1234’);

INSERT INTO CUSTOMER VALUES(‘C2361357136’ , 'KRISHNA','CHAITANYA','M',19,9894561230,'EFGH@GMAIL.COM', ‘Krishna@1597’);

INSERT INTO CUSTOMER VALUES(‘C6871313581’,'HARSHA','VARDHAN','M',19,6784561230,'MNOP@GMAIL.COM',’Harsha@3497’);

INSERT INTO CUSTOMER VALUES(‘C3157761383’,'VAMSI','BHARGAV','M',25,9635561230,'VAMSHI@GMAIL.COM', ‘Vamsi@2358’);

INSERT INTO CUSTOMER VALUES(‘C6871368973’,'SWAPNA','KUMARI','F',23,6982361230,'SWAPNA@GMAIL.COM', ‘Swapna@3456’);

INSERT INTO CUSTOMER VALUES(‘C2698575573’ , 'NAMRATA','DUTTA','F,19,8981561230,'NAMRATA@GMAIL.COM', ‘Namrata@99’);

INSERT INTO CUSTOMER VALUES(C2154434545,'MOHAN','BABU','M',23,8894561230,'MOHAN@GMAIL.COM', Mohan@9999);

INSERT INTO CUSTOMER VALUES(C3651436716,'ANANYA','PANDEY','F',28,6985461230,'ANANYA@GMAIL.COM',ANNU@1234);

INSERT INTO CUSTOMER VALUES(C3681369171,'SHRADDHA','KAPOOR','F',35,9658561230,'SHRADDHA@GMAIL.COM',Shraddha@5555);

INSERT INTO CUSTOMER VALUES(C0023545547,'SAMANTHA','RUTHPRABHU','F',35,7894561230,'SAMANTHA@GMAIL.COM',Sam@6666);

*PHARMACIST TABLE*

INSERT INTO PHARMACIST VALUES(P3645645612,'ALLU','ARJUN','M',39,7412589630,'ALLU@gmail.com','Allu@3694');

INSERT INTO PHARMACIST VALUES(P9945645612,'PRABHAS','VARMA','M',38,9653589630,'VARMA@gmail.com','Prabh@2984');

INSERT INTO PHARMACIST VALUES(P1235645612,'TARAKA','RAMARAO','M',26,6954589630,'TARAK@gmail.com','Tarak@2365');

INSERT INTO PHARMACIST VALUES(P3698645612,'NITHIN','CHOWDARY','M',35,9652589630,'NITHIN@gmail.com','Nithin@1257');

INSERT INTO PHARMACIST VALUES(P2548645612,'ANUSHKA','SHARMA','F',34,6999589630,'ANU@gmail.com','Anu@1236');

INSERT INTO PHARMACIST VALUES(P5784645612,'KAJAL','AGARWAL','F',35,9657489630,'KAJAL@gmail.com','Kajal@4597');

INSERT INTO PHARMACIST VALUES(P1596645612,'KATRINA','KAIF','F',49,6634589630,'KATRINA@gmail.com','Katrina@1256');

INSERT INTO PHARMACIST VALUES(P1528645612,'DISHA','PATANI','F',31,9644449630,'DISHA@gmail.com','Disha@1598');

INSERT INTO PHARMACIST VALUES(P3255645612,'SHRUTI','HASSAN','F',41,8652989630,'SHRUTI@gmail.com','Shruti@1478');

INSERT INTO PHARMACIST VALUES(P2998645612,'MAHESH','BABU','M',26,8965189630,'MAHESH@gmail.com','Mahesh@1234');

*MEDICINES TABLE*

INSERT INTO MEDICINES VALUES('M3245697851','BRODIL','USED FOR COUGH',199);

INSERT INTO MEDICINES VALUES('M1234567895','AJITHROMYCIN','USED FOR THROAT INFECTION' , 325);

INSERT INTO MEDICINES VALUES('M3598846123','PARACETAMOL','USED FOR MILD FEVER',099);

INSERT INTO MEDICINES VALUES('M3598764123','NAPROXEN','USED FOR HEADACHE',399);

INSERT INTO MEDICINES VALUES('M1234976512','ACETAMINOPHEN','USED FOR COLD',359);

INSERT INTO MEDICINES VALUES('M3688751365','AMOXICILLIN','USED FOR SKIN INFECTION',156);

INSERT INTO MEDICINES VALUES('M3613675367','PERIDEX','USED FOR ANTICEPTIC',450);

INSERT INTO MEDICINES VALUES('M1357369839','IBUPROFEN','USED FOR INJURIES ',330);

INSERT INTO MEDICINES VALUES('M9865423435','ASPIRIN','USED FOR PAIN KILLER',550);

INSERT INTO MEDICINES VALUES('M5649845848','PEPTO BISMOL','USED FOR VOMTING',440);

*PURCHASING TABLE*

INSERT INTO PURCHASING VALUES('PU561223563','C0091234545','M3245697851',199, ‘12-DEC-2022’);

INSERT INTO PURCHASING VALUES('PU996123566','C2361357136','M1234567895',325, ‘15-OCT-2021’);

INSERT INTO PURCHASING VALUES('PU556132563','C6871313581','M3598846123',099, ‘26-APR-2020’);

INSERT INTO PURCHASING VALUES('PU126198563','C3157761383','M3598764123',399, ‘22-MAR-2020’);

INSERT INTO PURCHASING VALUES('PU896125256','C6871368973','M1234976512',359, ‘14-DEC-2019’);

INSERT INTO PURCHASING VALUES('PU666198563','C2698575573','M3688751365',156, ‘10-JUL-2018’);

INSERT INTO PURCHASING VALUES('PU556168563','C2154434545','M3613675367',450, ‘01-NOV-2019’);

INSERT INTO PURCHASING VALUES('PU446148563','C3651436716','M1357369839',330, ‘29-AUG-2017’);

INSERT INTO PURCHASING VALUES('PU336134563','C3681369171','M9865423435',550, ‘06-SEP-2016’);

INSERT INTO PURCHASING VALUES('PU226159563','C0023545547','M5649845848',440, ‘30-MAR-2022’);

*SALES TABLE*

INSERT INTO SALES VALUES('S59894555', ' C0091234545',' M3245697851',2,' PU561223563', ‘12-DEC-2022’,398);

INSERT INTO SALES VALUES('S25698745',' C2361357136',' M1234567895',3,' PU996123566', ‘15-OCT-2021’,975);

INSERT INTO SALES VALUES('S69855215', ' C6871313581',' M3598846123',5,' PU556132563', ‘26-APR-2020’,495);

INSERT INTO SALES VALUES('S12345678',' C3157761383',' M3598764123',1,' PU126198563', ‘22-MAR-2020’,399);

INSERT INTO SALES VALUES('S36987412', ' C6871368973',' M1234976512',10,' PU896125256', ‘14-DEC-2019’,3590);

INSERT INTO SALES VALUES('S36987456', ‘C2698575573',' M3688751365',4,' PU666198563', ‘10-JUL-2018’,624);

INSERT INTO SALES VALUES('S25698746', ' C2154434545',' M3613675367',10,' PU556168563', ‘01-NOV-2019’,4500);

INSERT INTO SALES VALUES('S15236987', ' C3651436716',' M1357369839',7,' PU446148563', ‘29-AUG-2017’,2310);

INSERT INTO SALES VALUES('S36559821', ' C3681369171',' M9865423435',8,' PU336134563', ‘06-SEPT-2016’,4400);

INSERT INTO SALES VALUES('S12365986', ' C0023545547',' M5649845848',3,' PU226159563', ‘30-MAR-2022’,1320);

*REPORTS TABLE*

INSERT INTO REPORTS VALUES('R458796587','P3645645612',' C0091234545',12-12-2022);

INSERT INTO REPORTS VALUES('R852369741','P2548645612','S36987412' ,14-12-2019);

INSERT INTO REPORTS VALUES('R485933274','P2998645612',' C0023545547',30-03-2022);

**QUERIES**

**1. Retrieve all customers information**

select \* from customer;

2. Retrieve all pharmacists information

select \* from pharmacist;

**3. Retrieve name and age of all male customers**

select fname, lname, age

from customer

where gender = 'M';

**4. retrieve the name and med\_id of all the medicines whose price is less then average cost of medicines**

select med\_id, med\_category, price

from medicines

where price < (select avg(price) from medicines);

**5. Retrieve the name of customers who spent more than 1000RS and the pharmacist who sold it to them**

select phfname, fname

from sales natural join customer natural join pharmacist

where total\_amount>1000;

**6. Retrieve the reports\_id filed in december of any year**

select reports\_id

from reports

where extract(month from report\_date) = 12;

**7. Find the month during which the most sales where made**

select extract(month from sale\_date)

from sales

where total\_amount = (select max(total\_amount) from sales)

group by extract(month from sale\_date);

**8. Display the purchase\_id,cust\_id and name of those customers who purchased medicine in 2019 and name of the medicine**

select purchase\_id, cust\_id, fname, lname, med\_category

from customer natural join purchasing natural join medicines

where extract(year from purchase\_date) = 2019;

**9. display the phar\_id and names of pharmacists whose number starts with 9**

select phar\_id, phfname, phlname

from pharmacist

where phcontact\_add LIKE '9%';

**10. display the details of the medicines for infection**

select med\_id, med\_category, description, price

from medicines

where description LIKE '%INFECTION%';

**TRIGGERS**

1. **Design a trigger to be fired when customer’s age is below 18**

create or replace trigger customer\_age before insert or update on customer

for each row

begin

if :new.age<18 then

raise\_application\_error(-20111, 'Customer should be older than 18 year');

else

dbms\_output.put\_line('Successful');

end if;

end;

/

INSERT INTO CUSTOMER VALUES('C6871368000','SWANA','KMARI','F',12,6982361230,'SWAPNA@GMAIL.COM', 'Swapna@3456');

1. **Design a trigger to ensure password length is 9 or more**

create or replace trigger pass\_length before insert on customer

for each row

begin

if length(:new.cust\_pass) <8 then

dbms\_output.put\_line('Incorrect password enter again');

raise\_application\_error(-20111,'Password must be atleast 8 length characters');

end if;

end;

/

INSERT INTO CUSTOMER VALUES('C6871368000','SWANA','KMARI','F',12,6982361230,'SWAPNA@GMAIL.COM', 'Swap@');

create or replace trigger pass\_length\_admin before insert on pharmacist

for each row

begin

if length(:new.admin\_pass) <8 then

dbms\_output.put\_line('Incorrect password enter again');

raise\_application\_error(-20111,'Password must be atleast 8 length characters');

end if;

end;

**STORED PROCEDURES**

1. **Procedure to find name of pharmacist who made highest sale in input month**

set serveroutput on;

create or replace procedure max\_pharmacist(input\_month in number, ph\_name out varchar) IS

begin

select phfname into ph\_name from pharmacist natural join sales where total\_amount = (select max(total\_amount) from sales where extract(month from sale\_date) = input\_month);

exception

when no\_data\_found then

dbms\_output.put\_line('pharmacist not found');

end max\_pharmacist;

/

set serveroutput on;

declare

v\_month number(2) := 4;

ph\_name pharmacist.phfname%TYPE;

begin

v\_month = '&v\_month';

max\_pharmacist(v\_month,ph\_name);

dbms\_output.put\_line(ph\_name || ' made the most sales in ' || v\_month);

end;

/

1. **Procedure to print all the names of customer who bought a particular medicine**

set serveroutput on;

create or replace procedure name\_med(medicine in varchar2) IS

cursor c\_cur is select fname from sales natural join customer natural join medicines where med\_category = medicine;

begin

dbms\_output.put\_line('People who bought ' || medicine || ' are ');

for i in c\_cur

loop

dbms\_output.put\_line(i.fname);

end loop;

end name\_med;

/

set serveroutput on;

declare

med medicines.med\_category%type := 'PARACETAMOL';

begin

name\_med(med);

end;

/

**STORED FUNCTIONS**

1. **Function to calculate total price of sales done by a particular pharmacist**

CREATE OR REPLACE function Total\_Price(pharid IN Varchar2) return number as price number(6):=0;

Cursor c\_ph is Select total\_amount from sales where phar\_id = pharid;

begin

for i in c\_ph

loop

price := price + i.total\_amount;

end loop;

return price;

end Total\_Price;

/

--INSERT INTO SALES VALUES('S15236007','P1528645612','C3651436716','M1357369839',7,'PU446148563', '29-AUG-2017',2400);

set serveroutput on;

declare

id pharmacist.phar\_id%type := 'P1528645612';

price number(6);

begin

price := Total\_Price(id);

dbms\_output.put\_line('Total amount of medicines sold by pharmacist are : ' || price);

end;

/

1. **Function to Login to the portal using username and password**

CREATE OR REPLACE function Login(email IN Varchar2,pass in varchar2) return number as access := 1;

Cursor c\_ct is Select cust\_id,fname,lname from customer where cust\_email = email and cust\_pass = pass;

Begin

Open c\_ct;

Loop

Fetch into i;

If i%notfound then

If i%ROWCOUNT=0 then

Dbms\_output.put\_line(‘Enter valid id and password’);

End if; exit;

Else dbms\_output.put\_line(‘Logged in successfully’);

End if;

End loop; close c\_ct;

End;

/

**RESULT**

The role of pharmacy management is to supervise and manage the pharmacy employees to preserve excellent working relationships and outcomes.

Pharmacy management system was developed to ensure the security of information and reliability of pharmacy records when accessing and providing services to the customers. The information gathered during the data collection was properly analysed and the results provided the basis for the new system. The system was tested and found to be functional and the outputs produced by this system were encouraging. The application will hence reduce the loss of information and also information will be processed faster than before.

**CONCLUSION AND FUTURE WORK**

This project has been a venture to satisfy the growing needs of a pharmacy. The objective of this project has been to provide a user friendly frame work that enables both the pharmacists as well as the customer to interact with the database. Effective implementation of this software will take care of the basic requirements of the pharmacy management system because it is capable of providing easy and effective storage of information related to activities happening in the stipulated area.

In order to allow future expansion, the system has been designed in such a way that will allow possible modification as it may deem necessary by the pharmacy management whenever the need arises.

In the future more functionalities can be added –

1. Customers will be able to check the dosage, pharmaceutical manufacturing companies and other such information
2. Pill imaging can also be added which will help prevent incorrect medication dispensing. Quality assurance check takes place and pills are compared and displayed with bottle contents thus ensuring that correct drugs are available.
3. Electronic Prescriptions: Using this the pharmacy will receive new and refill prescriptions directly into the pharmacy management system for review