PROJECT REPORT ON

PHARMACY MANAGEMENT SYSTEM

Submitted in partial fulfillment for the requirement

for the award of the Degree of

Bachelor of Technology

Session:2018-2022

Even semester (VIth)

Submitted to:

Mr. Vijay Bijlwan

Submitted by:

Archana Bharti &

B.Tech.(C.S.E. IVth Sem)

Department of Computer Science and Engineering

School of Engineering and Technology

HEMWATI NANDAN BAHUGUNA GARHWAL UNIVERSITY

ACKNOWLEDGEMENT

Before getting into the thickest of things, we would like to thank the personalities who were parts of our project in numerous ways, those who gave us outstanding support from the birth of the project and solely recognize our indebtedness for guidance and assistance of the project adviser and other member of the faculty.

Courtesy demands that we also recognize specific contribution by other person or institutions such as libraries and research foundation and internet services too. We acknowledge our project guide Mr. Vijay Bijlwan sir, without his support and guidance, we couldn't complete this project.

We are highly indebted to Dr. Y.P. Raiwani, Head of Department, Department of Computer Science and Engineering, for this support during the tenure of this project.

CERTIFICATE

This is to certify that this report entitled “Pharmacy Management System” submitted in partial fulfillment of the degree of Bachelor of Technology (B.Tech.) in Computer Science Engineering to the department of computer science and engineering, school of engineering and technology H.N.B. Garhwal University, Srinagar, Uttarakhand done by Archana Bharti, is an authentic work carried out by them under my guidance. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

SELF CERTIFICATE

This is to certify that the report entitled **Pharmacy Management System** is done by us is an authentic work carried out for the partial fulfillment of the requirement for the award of the degree of Bachelor of Technology (B.Tech.) in Computer Science Engineering under the guidance of Mr. Vijay Bijlwan sir. The matter embodied in the project has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

Archana Bharti

(2018-2022)

PREFACE

For efficient and quality improvement of an educational institute, feedback plays a key role in it. With a proper feedback mechanism in place, the growth rate of an institute shows an upward movement. The feedback mechanism helps in catering the weaknesses and further strengthening the strengths. The SWOT analysis of an institute can be done by ensuring a strong feedback system is in place. For a large institute, it is impossible to manually manage a proper feedback system.

CONTENTS

Abstract

The purpose of this project is in partial fulfilment of the requirements of customer using the online mode of ordering medicines. The Design and development of this Pharmacy Management system provides a more secured approach in managing pharmacy customer's information which strengthens the relationships between pharmacies and their customers by providing the right solutions that uses a multi- level security to improve customer satisfaction. The programming language used to develop this project is C++.

The Domain "Pharmacy Management System" keeps the manual system of the medical shop into the computerized system, by which an end user will be able to manage the medical records very well and accurate. It can keep the information of list of medicines available, delete and modify medicine orders, print receipt, make payments, summary of total sales.

Introduction

Pharmacy Management System is based on a concept of taking medicine orders with payments. In this system, there are no login features. The user can use the features of it which include taking new medicine order, deleting, modifying medicine orders, printing the receipt of payment and viewing the summary of total sales within a day. This mini project contains limited features, but the essential one.

Talking about the features of the Pharmacy Management System, the user can take a new medicine order. For this, the user has to provide the order number, name, date, number of medicine and select medicine. The user can delete the latest medicine order by providing the order number. In order to modify the order, he/she has to enter each and every detail. After taking orders, the payment is to be done from the receipt section. Here in the receipt section, the system asks for a receipt number. Then the system displays total receipt with medicine type, name, quantity, and total price.

Project Objective:

The main objective of this package is:

- Take medicine orders
- Delete and Modify medicine orders
- Print Receipt
- Make Payments
- Summary of Total Sales

Problem in existing Pharmacy System:

At present, manual system is being utilized in the pharmacy. It requires the pharmacist to manually monitor each drug that is available in the pharmacy. Pharmacy management has kept paper record in filing cabinets. Managing a very large pharmacy with records on papers will be tedious and difficult to keep track of inventories with regards to the drugs in the store, expiry date, quantity of drugs available based on the categories and their functions. This implies that these services will be manually completed by the pharmacist. This usually leads to mistakes as the workload of the pharmacist increases. This system also ensures that there exists a level of restricted access based on functionality and role. This system also provides optimal drug

inventory management by monitoring the drug movement in the pharmacy. Significant amount of time is allocated for writing the order as the pharmacist needs to go through the stock balance and make rough estimate of the amount to order based on Figures. The system will not be able to handle drug prescription, drug to drug interaction.

The pharmacist has to order drugs to replenish the already diminishing stock. In addition, ordering of drugs is being carried out manually. Drugs are not supposed to be used after they have expired. The set-up of this pharmacy management system will ensure availability of sufficient quantity of drugs and consumable materials for the patient. This project work will prompt the pharmacist about drugs that are close to expiry, preventing those drugs from being sold and also providing solution to the earlier stated problems. Pharmacy management system helps in storing the daily transactions of medicines. The project is small and supports ample operations only.

Project Benefits:

Some benefits are:

- Access to privilege pharmacy shops
- Reduces clerical work
- Customers can buy medicines anywhere with higher limits
- Helps to keep record of daily medicines sales
- Helps customer to escape the long in-person queues
- Provide greater speed & reduced time consumption.

Methodology-

Concepts of C++ used in the project are:

1. Header files
2. Class
3. Member function
4. Function

   Declaration, Definition and Calling
5. Switch statement
6. File management

System Requirements

Hardware Requirements:

Hardware is a set of physical components, which performs the functions of applying appropriate, predefined instructions. In other words, one can say that electronic and mechanical parts of computer constitute hardware. This project is built on a one of the basic programing languages, C++. It can run on all popular microcomputers.

The following is the minimum hardware specification to use this package:

- Computer with either Intel Pentium processor or AMD processor.
- 128MB DDR RAM
- 40GB hard disk drive

Software Requirements:

The software is a set of procedures of coded information or a program which when fed into the computer hardware, enables the computer to perform the various tasks. Software is like a current inside the wire, which cannot be seen but its effect can be felt.

The following is the minimum software requirement for this package:

- Windows NT and all later release.
- Microsoft Office package.
- Dev C++ or any other C++ compiler

Source Code

```
#include <iostream>

#include <stdlib.h>

#include <string>

#include <cctype>

#include <cmath>

#include <cstdio>

#include <fstream>

#include <iomanip>

#define max 10


using namespace std;


class medicineType

{

public:


    void take_order();

    void delete_order();

    void modify();

            void order_list();
```

```
        void daily_summary();
        void exit();
        medicineType();

};

 medicineType::medicineType ()
{

}

struct node
{
        int reciept_number;
        string customerName;
        string date;
        int quantity[10];
        string type = {"OTC"};
        int x, menu2[10];
    double amount[10];
```

```
    string medicineName[10]={"Probiotics","Vitamin C(500mg)","Acid Free C(500mg)","Women'S Multivate","Marino Tablet","Maxi Cal Tablet",
        "Amino Zinc Tablet","Burnex","Fabuloss 5","Royal Propollen"};
        double Medicine[10] = {2.00,3.00,1.00,4.00,1.00,5.00,7.00,4.00,3.00,5.00};
        double total;

        node *prev;
        node *next;
        node *link;

}*q, *temp;


node *start_ptr = NULL;
node *head = NULL;
node *last = NULL;

int main()
{

        system("COLOR 0");
        medicineType medicine;
```

```
int menu;
do
{
	system("cls");
	cout<<"\t\t\t    Pharmacy Management System \n";

cout<<"\t\t==========================================================\n\n"
;
	cout<<"\t\t-----------------------------------------------\n";
	cout<<"\t\t||\t1. Take new Medicine order \t\t ||\n";
	cout<<"\t\t||\t2. Delete latest Medicine order\t\t ||\n";
	cout<<"\t\t||\t3. Modify Order List \t\t\t ||\n";
	cout<<"\t\t||\t4. Print the Reciept and Make Payment \t ||\n";
	cout<<"\t\t||\t5. Daily Summary of total Sale \t\t ||\n";
	cout<<"\t\t||\t6. Exit\t\t\t\t\t ||\n";
	cout<<"\t\t-----------------------------------------------\n";
	cout<<"Enter choice: ";

	cin>>menu;

	switch (menu)
	{
```

```
case 1:
        {
                medicine.take_order();
                break;
        }

case 2:
        {
                medicine.delete_order();
                system("PAUSE");
                break;
        }

case 3:
        {
                medicine.modify();
                system("PAUSE");
                break;
        }
```

```
            case 4:
                {
                    medicine.order_list();
                    system("PAUSE");
                    break;
                }
            case 5:
                {
                    medicine.daily_summary();
                    system("PAUSE");
                    break;
                }
        case 6:
                {
                    medicine.exit();
                    goto a;
                    break;
                }


            default:
```

```
                {
                        cout<<"You enter invalid input\nre-enter the input\n"<<endl;
                        break;
                }
        }
    }while(menu!=6);
    a:
    cout<<"thank you"<<endl;
    system ("PAUSE");
    return 0;
}



void medicineType::take_order()
{
    system("cls");
    int i;
   int choice, quantity, price,None;

    cout <<"\nAdd Order Details\n";
    cout <<"_____ \n\n";
```

```
        node *temp;
        temp=new node;


                                cout
<<"**********************************************************************
**\n";
                                cout<<"DRUGS  ID"<<"\tDRUGS  TYPE"<<"      \t\tDRUGS
NAME"<<"         DRUGS PRICE(RM)"<<endl;
                                cout
<<"**********************************************************************
**\n";
        cout<<"0001"<<"\t"<<"\tOTC"<<"\t\t"<<"   Probiotics"<<"                RM
2.00"<<endl;
        cout<<"0002"<<"\t"<<"\tOTC"<<"\t\t"<<"   Vitamin C(500mg)"<<"          RM
3.00"<<endl;
        cout<<"0003"<<"\t"<<"\tOTC"<<"\t\t"<<"   Acid Free C(500mg)"<<"       RM
1.00"<<endl;
        cout<<"0004"<<"\t"<<"\tOTC"<<"\t\t"<<"   Women'S Multivate"<<"        RM
4.00"<<endl;
        cout<<"0005"<<"\t"<<"\tOTC"<<"\t\t"<<"   Marino Tablet"<<"          RM
1.00"<<endl;
        cout<<"0006"<<"\t"<<"\tOTC"<<"\t\t"<<"   Maxi Cal Tablet"<<"           RM
5.00"<<endl;
        cout<<"0007"<<"\t"<<"\tOTC"<<"\t\t"<<"   Amino Zinc Tablet"<<"        RM
7.00"<<endl;
```

```
            cout<<"0008"<<"\t"<<"\tOTC"<<"\t\t"<<"    Burnex"<<"                    RM
4.00"<<endl;//1353fn

            cout<<"0009"<<"\t"<<"\tOTC"<<"\t\t"<<"    Fabuloss 5"<<"                RM
3.00"<<endl;

            cout<<"0010"<<"\t"<<"\tOTC"<<"\t\t"<<"    Royal Propollen"<<"          RM
5.00"<<endl;

            cout<<" "<<endl;



        temp = new node;

        cout << "Type Order no: ";

   cin >> temp->reciept_number;

        cout<< "Enter Customer Name: ";

        cin>> temp->customerName;

        cout<<"Enter Date : ";

        cin>>temp->date;

        cout << "How many Medicine would you like to order:"<< endl;

        cout<<"( Maximum is 10 order for each transaction ) \n";

        cout << "   " ;

        cin >> temp->x;

        if (temp->x >10)

        {

                cout << "The Medicine you order is exceed the maximum amount of order !";

                system("pause");
```

```
        }
        else{
        for (i=0; i<temp->x; i++)
        {

                cout << "Please enter your selection : "<<endl;
                cin>> temp->menu2[i];
      cout<< "Medicine Name: " <<temp->medicineName[temp->menu2[i]-1]<<endl;
      cout << "How many medicine do you want: ";
      cin >> temp->quantity[i];
      temp->amount[i] = temp->quantity[i] * temp->Medicine[temp->menu2[i]-1];
      cout << "The amount You need to pay is: " << temp->amount[i]<<" RM"<<endl;
      system("PAUSE");

        }
        cout<<"=========================================================
====================="<<endl;
   cout << "Order Taken Successfully"<<endl;

cout<<"=========================================================
================"<<endl;
   cout << "Go to Reciept Menu to Pay The Bill"<<endl;
```

```
cout<<"================================================================
================"<<endl;
        system ("PAUSE");


   temp->next=NULL;
        if(start_ptr!=NULL)
        {
                temp->next=start_ptr;
        }
        start_ptr=temp;
        system("cls");
}
}


void medicineType::order_list()
{
        int i, num, num2;
        bool found;
        system("cls");
        node *temp;
```

```
	temp=start_ptr;
	found = false;

	cout<<" Enter the Reciept Number To Print The Reciept\n";
	cin>>num2;
	cout<<"\n";
	cout<<"=============================================================
====================="<<endl;
	cout <<"\t\tHere is the Order list\n";
	cout<<"=============================================================
====================="<<endl;


	if(temp == NULL)
	{
		cout << "\tThere is no Order to show\n\t\t\tSo The List is Empty\n\n\n";
	}
	while(temp !=NULL && !found)
	{
		if (temp->reciept_number==num2)
		{
```

```
                found = true;
            }
            else
            {
                temp = temp -> next;
            }
    if (found)
    {
            cout <<"Reciept Number : "<<temp->reciept_number;
            cout <<"\n";
            cout<<"Customer Name: "<<temp->customerName<<endl;

            cout<<"Order Date : "<<temp->date<<endl;

        cout<<"_____
_____"<<endl;

            cout                                                    <<
"=============================================================================
==============" << endl;
            cout << "|  Medicine Type |     Medicine Name    |    Quantity     |    Total Price
|" << endl;
```

```
                cout                                                                                    <<
"=======++==================++================++===============++=
==============" << endl;

                for (i=0;i<temp->x;i++)

                {

                        cout << temp->type <<"  \t\t";

                        cout<<temp->medicineName[temp->menu2[i]-1]<<"\t\t\t  ";

                        cout<<temp->quantity[i] <<"\t\t";

                        cout<< temp->amount[i]<<" RM"<<endl;


        cout<<"_____________________________________________________________
_____________________"<<endl;

                }


                temp->total  =  temp->amount[0]+temp->amount[1]+temp->amount[2]+temp-
>amount[3]+temp->amount[4]+temp->amount[5]+temp->amount[6]+temp->amount[7]

                                                +temp->amount[8]+temp->amount[9];

                cout<<"Total Bill is : "<<temp->total;

                cout<<"\n";

                cout << "Type the exact amount You need to pay: ";

      cin >> num;


                cout <<"\n";
```

```
                cout <<"\n";
                cout<<"Payment Done\nThank You\n";
                cout
<<"\n_____
_____\n";
                }



}
}


void medicineType::delete_order()
{
        system("cls");
        int i, num, count;
    cout<<"Enter the data you want to delete \n";
    cin>>num;
    node *q;
        node *temp;
        bool found;
```

```
if(start_ptr == NULL)
        cerr<<"Can not delete from an empty list.\n";
else
{
        if(start_ptr->reciept_number == num)
        {
                q = start_ptr;
                start_ptr = start_ptr->next;
                count--;
                if(start_ptr == NULL)
                last = NULL;
                delete q;
                cout<<"The Reciept is Deleted Successfully"<<endl;
        }
        else
        {
                found = false;
                temp = start_ptr;
                q = start_ptr->next;

        while((!found) && (q != NULL))
```

```
{
        if(q->reciept_number != num)
        {
                temp = q;
                q = q-> next;
        }
        else
                found = true;
}

        if(found)
        {
                temp->next = q->next;
                count--;

                if(last == q)
                last = temp;
                delete q;
                cout<<"The Reciept is Deleted Successfully"<<endl;
        }
        else
```

```
                                cout<<"Item to be deleted is not in the list."<<endl;
                            }
                        }
}
```

```
void medicineType::modify()
{
 system("cls");
 int i, ch, sid;
 bool found;
 found = false;
 temp = start_ptr;
 cout<<"Enter Receipt Number To Modify: ";
 cin>>sid;
 if (temp==NULL && sid==0)
 {
   cout<<"NO RECORD TO MODIFY..!"<<endl;
 }
```

```
else
{
        while(temp !=NULL && !found)
        {
                if (temp->reciept_number==sid)
                {
                        found = true;
                }
                else
                {
                        temp = temp -> next;
                }
  if (found)
  {
        cout << "Change  Order Number: ";
  cin >> temp->reciept_number;
        cout<< "Change Customer Name: ";
        cin>> temp->customerName;
        cout<<"Change Date : ";
        cin>>temp->date;
        cout << "How many New Medicine would you like to Change:"<< endl;
```

```
cout<<"( Maximum is 10 order for each transaction ) \n";
cout << "  " ;
cin >> temp->x;
if (temp->x >10)
{
        cout << "The Medicine you order is exceed the maximum amount of order !";
        system("pause");
}
else{
for (i=0; i<temp->x; i++)
{

        cout << "Please enter your selection to Change: "<<endl;
        cin>> temp->menu2[i];
cout<< "Change Medicine Name: " <<temp->medicineName[temp->menu2[i]-1]<<endl;
cout << "How many New medicine do you want: ";
cin >> temp->quantity[i];
temp->amount[i] = temp->quantity[i] * temp->Medicine[temp->menu2[i]-1];
cout << "The amount You need to pay After Modify  is: " << temp->amount[i]<<"
RM"<<endl;
system("PAUSE");
}
```

```
        temp = temp->next;
        system("cls");

        }

cout<<"RECORD MODIFIED....!"<<endl;
}
else
{
        if(temp != NULL && temp->reciept_number != sid)
        {
        cout<<"Invalid Reciept Number...!"<<endl;
   }
}
}
}
}

void medicineType::daily_summary()
```

```
{
	int i,num;
	system("cls");
	node *temp ;

	temp=start_ptr;


	if(temp == NULL)
	{
		cout << "\t\t\tThere is no Order to show\n\t\t\tSo The List is Empty\n\n\n";
	}
	else
	{
		cout<<"\n";

	cout<<"=============================================================
====================="<<endl;
		cout <<" \t\tHere is the Daily Summary of All Orders \n"; //print all receipt

	cout<<"=============================================================
====================="<<endl;
```

```
while(temp!=NULL)
{

cout <<"Reciept Number : "<<temp->reciept_number;
cout <<"\n";
cout<<"Customer Name: "<<temp->customerName<<endl;

cout<<"Order Date : "<<temp->date<<endl;

cout<<"_____
_____"<<endl;

cout <<
"=================================================================
=========" << endl;
cout << "|  Medicine Type |   Medicine Name    |    Quantity    |
Total Price |" << endl;
cout <<
"=======++==================++================++===============++=
=========" << endl;
for (i=0;i<temp->x;i++)
{
cout << temp->type <<"  \t\t";
```

```
                cout<<temp->medicineName[temp->menu2[i]-1]<<"\t\t";

                cout<<temp->quantity[i] <<"\t\t";

                cout<< temp->amount[i]<<" RM"<<endl;


        cout<<"_____
_____"<<endl;

                }


                temp->total = temp->amount[0]+temp->amount[1]+temp-
>amount[2]+temp->amount[3]+temp->amount[4]+temp->amount[5]+temp-
>amount[6]+temp->amount[7]

                +temp->amount[8]+temp->amount[9];

                cout<<"Total Bill is : "<<temp->total;


                cout <<"\n";

                cout <<"\n";

                cout
<<"\n_____
_____\n";


                temp=temp->next;

        }

    }

}
```

```
void medicineType::exit()

{

	cout<<"\nYou choose to exit.\n"<<endl;

}
```

Source Code Description

The source code includes four header files namely **iostream, stdlib.h, string, cctype, cmath, cstdio, fstream, iomanip**

It has a public class named **medicineType** with six member functions mentioned below:

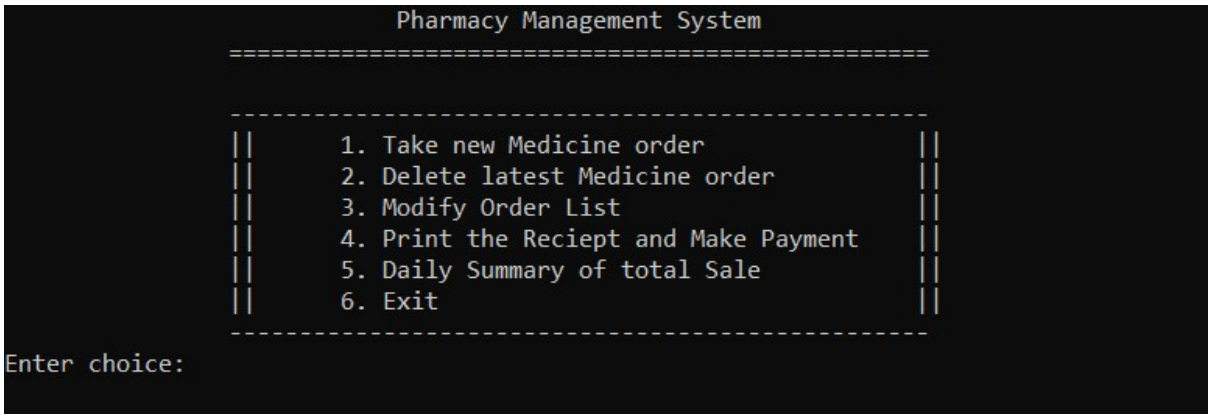
- **take_order()**- This member function gets data from the user and take a new order for the user.
- **delete_order()-** This member function deletes the detail of existing order by its order number. This is a const member function.
- **modify()-** This member function is used to modify details of existing order.
- **order_list()-** This member function is used to show details of an existing order. This is a const member function.
- **daily_summary()**- This member function is used to show the daily summary of transactions. This is a const member function.
- **exit()**- This member function is used to the main menu. This is a const member function.

The member functions are defined outside the class definition using the scope resolution :: operator. **medicineType** const member function and will never modifies data members in an object.

There are seven functions in the code as listed below:

- **take_order()-** This function is used to write record in binary file.
- **delete_order()-** This function is used to delete a record from file.
- **modify()-** This function is used to modify a record of file.
- **order_list()-** This function is used to display details of all the orders in a tabular format.
- **daily_summary()-** This function is used to display details of all the daily transactions in pharmacy in a tabular format.
- **exit()-** This function is used to get back to the main menu interface of the project.

A switch statement has been used to provide Main Menu selection control.


Pharmacy Management System
1. Take new Medicine order
2. Delete latest Medicine order
3. Modify Order List
4. Print the Reciept and Make Payment
5. Daily Summary of total Sale
6. Exit
Enter choice:

Main Menu

Project Limitations

- Can generate a better graphic based interface for the package.
- Link with database for better data management.
- The implementation and maintenance costs run very high
- Implementing the software requires change in the business practices.
- It leads to streaming of business processes.
- Detailed information gathering has to be done. Without that the purpose for using the software won't be satisfied properly.

Future Scope of the Project

This project can be handled in future by doing various modifications like:

- We can go further for Online Pharmacy system.
- We can also deal through internet by creating web pages and a website for internet dealing.
- We can establish one platform and link other various shops and available help centers for customer's queries.
- In this project we can also include BAR CODE facility using the bar code reader, which will detect the expiry date and the other information about the related medicines.
- Company using this software will always be able to plan in future and always be aware of their financial position in the market.
- To attract new customers, we can offer various offers during festivals months.
- To have more and more customer satisfaction we will emphasize more and more on our dealings.

References

- Object Oriented Programming with E.Balagurusamy Fourth edition
- https://nptel.ac.in/noc/courses/noc20/SEM2/noc20-cs57/
- https://www.geeksforgeeks.org/cpp-tutorial/
- https://www.w3schools.com/cpp/
- https://www.tutorialspoint.com/cplusplus/index.htm
- https://www.youtube.com/watch?v=vLnPwxZdW4Y