SRM MEDICAL STORE MANAGEMENT

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology/Master of Technology

In

Computer Science and Engineering

School of Engineering and Sciences

Submitted by

A.Sai Vivek - AP23110010268

M.Sandeep - AP23110010271

N .Tej Karthik - AP23110010305

R.Venkat Sai - AP23110010334

Under the Guidance of

Kavitha Rani Karnena

SRM University–AP

Neerukonda, Mangalagiri, Guntur

Andhra Pradesh – 522 240

[11, 2024]

Certificate

Date: 16-Nov-22

This is to certify that the work present in this Project entitled "**SRM MEDICAL STORE MANAGEMENT**" has been carried out by **A.Sai Vivek, M.Sandeep, N .Tej Karthik, R.Venkat Sai** under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology/Master of Technology in **School of Engineering and Sciences**.

Supervisor

Kavitha Rani Karnena

Acknowledgements

We express our profound gratitude to our project guide, **Ms. Kavitha Rani Karnena**, for her unwavering support, invaluable insights, and dedicated guidance throughout the development of this project, **SRM Medical Store Management System**. Her expertise and encouragement were instrumental in shaping our understanding and approach to solving complex problems.

We also extend our thanks to the faculty and staff of **SRM University – AP** for providing the necessary resources and a conducive environment for learning and innovation. Finally, we acknowledge the support and collaboration of our team members, without whom this project would not have been possible.

Table of Contents

Abstract

The **SRM Medical Store Management System (MSMS)** is an innovative software solution designed to streamline operations in medical stores by automating inventory and billing processes. This project specifically addresses the requirements of the SRM Medical Store, catering to students and faculty.

Key functionalities include the management of purchase and sales stock by company and customer categories, real-time inventory updates, and automated bill generation. The system significantly improves operational efficiency, accuracy in stock tracking, and customer experience through intuitive features such as expiration monitoring and sales reporting. This report presents the methodology, implementation, and future prospects of this software, highlighting its adaptability for use in other medical stores and pharmacies.

Abbreviations

MSMS: Medical Store Management System

MIS: Medical Inventory System

CMS: Chemist Management System

HIMS: Healthcare Inventory Management System

PMS: Pharmacy Management System

SIMS: Stock and Inventory Management System

MSS: Medicine Stock System

MAS: Medical Accounting System

POS: Point of Sale (for billing and sales tracking)

IPMS: Integrated Pharmacy Management System

List of Figures

1. Introduction

The **SRM Medical Store Management System** aims to alleviate the operational challenges of manual inventory and billing management in medical stores. This software provides an all-in-one solution for handling inventory control, customer management, transaction tracking, and report generation.

Project Objectives:

1. **Customer Management**: Simplify customer interactions by maintaining detailed profiles and transaction histories.
2. **Transaction and Sales Management**: Facilitate seamless sales processing with automated billing and receipt generation.
3. **Inventory Control**: Enable real-time tracking of stock levels, expiration dates, and restocking needs.
4. **Reporting**: Provide actionable insights through detailed reports on sales, purchases, and stock levels.

The system's intuitive design ensures that both retailers and wholesalers can easily transition from traditional methods to this efficient automated solution. With features like **expiration date tracking** and **flexible inventory controls**, the software not only ensures regulatory compliance but also enhances customer satisfaction by reducing errors and improving service speed.

2. Methodology

The **SRM Medical Store Management System (MSMS)** was developed using **C++ programming** to provide an efficient and user-friendly platform for medical store management. The methodology adopted ensures a structured and scalable approach to automate the traditionally manual processes of inventory and billing management.

Problem Definition:

Manual inventory systems in medical stores are prone to human errors, inefficiency, and difficulties in tracking stock expiration and sales data. This project aims to resolve these challenges by automating key operations such as:

- Managing stock inventory.
- Generating automated bills.
- Providing real-time insights on stock and sales.

Design and Development:

The project was implemented with the following structured approach:

- **Data Modeling**: The system models key entities such as medicines, customers, and sales. Each medicine has attributes like ID, name, company, price, manufacturing and expiry dates, quantity, and additional information.
- **System Features**:
  - **Purchase Management**: The system allows customers (students and faculty) to purchase medicines, calculating the total price based on quantity and available stock.
  - **Stock Management**: Tracks available stock, monitors expiration dates, and generates reports to avoid losses due to expired medicines.
  - **Medicine Information**: Displays detailed information, including reviews and usage instructions, for each medicine.
  - **Add/Delete Medicine**: Enables administrators to add new medicines to the inventory or remove outdated ones.
  - **Dynamic Updates**: Supports changes in quantity, price, or other attributes of medicines as needed.

Workflow Implementation:

The workflow is represented by diagrams such as the **Use-Case Diagram** and **Flow Chart** included in the report. The system's core workflow includes:

1. **Input Handling**:
    - User inputs such as IDs or names are validated to locate medicines in the inventory.
    - Functions like PurchaseMedicine() and AddMedicineinStore() handle these inputs efficiently.
2. **Processing**:
    - Conditional checks and calculations are performed to ensure correct pricing and availability.
    - Inventory is updated dynamically with each transaction.
3. **Output Generation**:
    - Generates outputs such as automated bills, updated stock reports, and detailed medicine information.

Code Features:

The software leverages a class-based structure in **C++**, ensuring modularity and scalability. Key components of the code include:

- **Class Medicine**: Defines attributes for medicines and provides a blueprint for managing the inventory.
- **Functionality**:
    - PurchaseMedicine() processes transactions, checks stock, and calculates costs.
    - AddMedicineinStore() adds new medicines with complete details, while DeleteMedicineStore() removes them.
    - StockOfMedicine() provides a detailed overview of all available medicines in the store.
- **Input and Output**:
    - Inputs are taken via user-friendly prompts, while outputs are displayed as detailed information or transaction results.

Tools and Techniques:

- **Programming Language**: C++ for its efficiency, reliability, and ease of handling data structures.
- **Development Environment**: Standard IDEs supporting C++ were used for development and debugging.

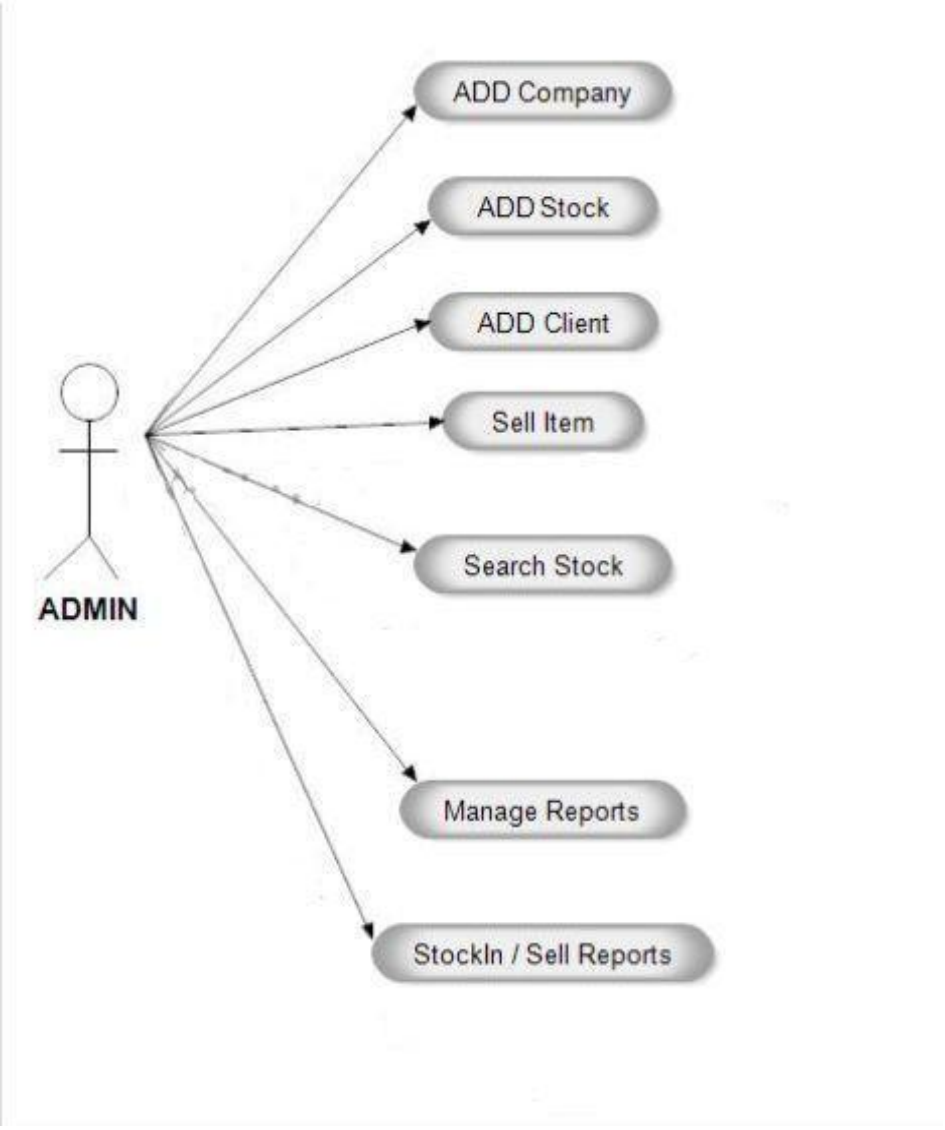
- **Testing**: Each feature was rigorously tested to ensure reliability and correctness.

Benefits and Features:

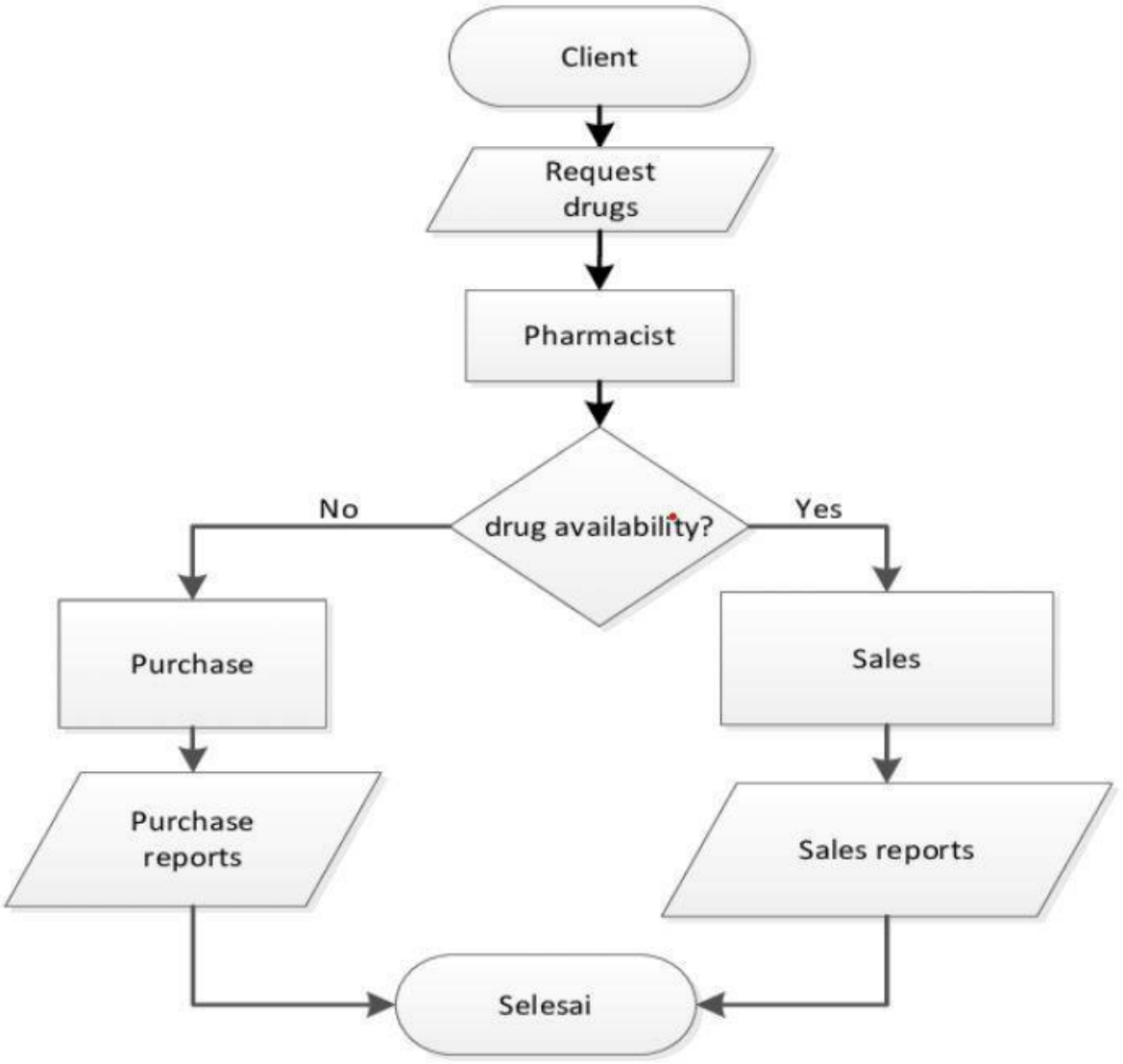
- The software reduces human errors, saves time, and ensures accurate inventory management.
- Provides pharmacists with tools to manage sales, track expiration dates, and generate reports.
- Offers a seamless interface for users, enhancing the overall experience for both customers and administrators.

3.Implementation

Use-Case Diagram :-



Flow Chart


Client
Request drugs
Pharmacist
drug availability?
No
Yes
Purchase
Purchase reports
Sales
Sales reports
Selesai

Code:-

```
#include <bits/stdc++.h>
using namespace std;



class Medicine
{
public:
  int id, price, quantity;
  char medicneName[100], Company[100], Mfg_Date[11], Exp_Date[11], info[5000];
};



class Medicine m[100];


void PurchaseMedicine(int number);
void EnterInfoAboutMedicine(int number);
void StockOfMedicine(int number);
void KnowInfoAboutMedicine(int number);
void AddMedicineinStore(int number, struct Medicine m[]);
void DeleteMedicineStore(int number);
void ChangeMedicineDetails(int number);



int main()
{
  int i, j, choice, number = 0, c;
  for (i = 0; i < 100; i++)
  {
    m[i].id = 0;
    m[i].price = 0;
m[i].quantity = 0;
    strcpy(m[i].Mfg_Date, "");
    strcpy(m[i].Exp_Date, "");
strcpy(m[i].medicneName, "");
    strcpy(m[i].Company, "");
    strcpy(m[i].info, "");
}
  m[0].id = 1;
m[0].price = 120;
  m[0].quantity = 30;
  strcpy(m[0].Mfg_Date, "23-04-2016");
strcpy(m[0].Exp_Date, "24-04-2020");
  strcpy(m[0].medicneName, "Paracetmol");
  strcpy(m[0].Company, "ABCD");
  strcpy(m[0].info, "Good Medicine for Fever");
  do
{
    cout<<endl<<endl<<"---- SRM AP MEDICAL STORE ----"<<endl<<endl;
    cout<<"Enter\n1 - Purchase Medicine\n2 - Enter Information about a Medicine\n3 - Stock
of Medicine in Store\n4 - Medicine Information\n5 - Add Medicine\n6 - Delete a Medicine\n7
- Change Quantity of Medicine\n";
```

```
    cin>>choice;



    switch (choice)
{
    case 1:
```

```
    {
      PurchaseMedicine(number + 1);
break;
    }
    case 2:
{
      EnterInfoAboutMedicine(number + 1);
break;
    }
    case 3:
{
      StockOfMedicine(number + 1);
      break;
}
    case 4:
{
      KnowInfoAboutMedicine(number + 1);
      break;
}
    case 5:
    {
      ++number;
      AddMedicineinStore(number, m);
break;
    }
    case 6:
{
      DeleteMedicineStore(number + 1);
      break;
}
    case 7:
{
      ChangeMedicineDetails(number + 1);
      break;
}
    }
    cout<<"To Continue with other Options Enter 1 Else any other number\n";
cin>>c;
  } while (c == 1);
}



void PurchaseMedicine(int number)
{
  int id, check, i, quantity, flag = 0;
  char name[100];
  cout<<"Enter 1 if you know ID else any other number to enter Name of Medicine"<<endl;
  fflush(stdin);
```

```
  cin>>check;



  if (check == 1)
{
    cout<<"Enter Id to purchase Medicine"<<endl;
    fflush(stdin);
cin>>id;


    for (i = 0; i < number; i++)
    {
```

```
      if (m[i].id == id)
      {
flag = 1;
        int c;
        cout<<"These are the details of Medicine"<<endl;
        cout<<"Name"<<m[i].medicneName<<endl<<"Price="<< m[i].price<<endl<<"Available
Quantity="<< m[i].quantity<<endl<<"Company="<< m[i].Company<<endl<<"Mfg
Date="<<m[i].Mfg_Date<<endl<<"Exp Date="<< m[i].Exp_Date<<endl;
        if (strcmp(m[i].info, "") == 0)
        {
          cout<<"Medicine Review/Info=Not Available"<<endl;
        }
        else
{
          cout<<"Medicine Review/Info="<<m[i].info<<endl;
}
        cout<<"Do you want to purchase"<<m[i].medicneName<<endl<<"If Yes Enter 1 else any
other number"<<endl;
fflush(stdin);
        cin>>c;
        if (c == 1)
{
          cout<<"Enter Quantity to Purchase"<<endl;
cin>>quantity;
          if (m[i].quantity > quantity)
          {
            cout<<"Total Price to be paid="<<quantity * m[i].price<<endl;
          }
          else
{
            cout<<"Please Enter quantity below Available Quantity"<<endl;
}
        }
        break;
}
    }
    if (flag == 0)
{
      cout<<"Entered Id Not Found"<<endl;
}
  }
  else
{
    cout<<"Enter Name to search and Purchase"<<endl;
```

```
    fflush(stdin);
gets(name);
    for (i = 0; i < number; i++)
    {
      if (strcmp(m[i].medicneName, name) == 0)
      {
flag = 1;
        int c;
        cout<<"These are the details of Medicine"<<endl;
        cout<<"Name="<<m[i].medicneName<<endl<<"Price="<<m[i].price<<endl<<"Available
Quantity="<<m[i].quantity<<endl<<"Company="<<m[i].Company<<endl<<"Mfg
Date="<<m[i].Mfg_Date<<endl<<"Exp Date="<<m[i].Exp_Date<<endl;
if (strcmp(m[i].info, "") == 0)
        {
          cout<<"Medicine Review/Info=Not Available"<<endl;
        }
        else
```

```
        {
          cout<<"Medicine Review/Info="<< m[i].info<<endl;
}
        cout<<"Do you want to purchase"<<m[i].medicneName<<endl<<"If Yes Enter 1 else any
other number"<<endl;
fflush(stdin);
        cin>>c;
if (c == 1)
        {
          cout<<"Enter Quantity to Purchase"<<endl;
cin>>quantity;
          if (m[i].quantity > quantity)
          {
            cout<<"Total Price to be paid="<<quantity * m[i].price<<endl;
          }
else
          {
            cout<<"Please Enter quantity below Available Quantity"<<endl;
}
        }
        break;
}
    }
    if (flag == 0)
    {
      cout<<"Entered Name Not Found"<<endl;
}
  }
}

void EnterInfoAboutMedicine(int number)
{
  int i, flag = 0, c;
  char name[100], info[100];
  cout<<"Enter Name of the medicine you want to Review or include Info"<<endl;
  fflush(stdin);
  gets(name);
  for (i = 0; i < number; i++)
  {
    if (strcmp(m[i].medicneName, name) == 0)
    {
```

```
      flag = 1;
      cout<<"These are the details of Medicine"<<endl;
      cout<<"Name="<<m[i].medicneName<< endl <<"Price="<<m[i].price<<endl<<"Available
Quantity="<<m[i].quantity<< endl <<"Company="<<m[i].Company<< endl <<"Mfg
Date="<<m[i].Mfg_Date<< endl <<"Exp Date="<< m[i].Exp_Date<< endl;
      if (strcmp(m[i].info, "") != 0)
{
        cout<<"Review Already Available!\nIf you want to Add a Review Enter 1 else Any
other number"<<endl;
fflush(stdin);
        cin>>c;
      }
else
      {
        cout<<"Enter review(less than 100 Characters)"<<endl;
fflush(stdin);
        gets(m[i].info);
}
      if (c == 1)
      {
```

```
        cout<<"Enter review(less than 100 Characters)"<<endl;
        fflush(stdin);
gets(info);
        strcat(m[i].info, "; ");
        strcat(m[i].info, info);
}
    }
  }
  if (flag == 0)
  {
    cout<<"Entered Name Not Found"<<endl;
  }
}

void KnowInfoAboutMedicine(int number)
{
  int i, flag = 0;
  char name[100];
  cout<<"Enter Name of the medicine you want to see Review and Info"<<endl;
  fflush(stdin);
  gets(name);
  for (i = 0; i < number; i++)
  {
    if (strcmp(m[i].medicneName, name) == 0)
    {
      flag = 1;
      cout<<"These are the details of Medicine"<<endl;
      cout<<"Name="<<m[i].medicneName<<endl<<"Price="<<m[i].price<<endl<<"Available
Quantity="<<m[i].quantity<<endl<<"Company="<<m[i].Company<<endl<<"Mfg
Date="<<m[i].Mfg_Date<<endl<<"Exp Date="<<m[i].Exp_Date<<endl;
      if (strcmp(m[i].info, "") != 0)
{
        cout<<"Review or Info="<<m[i].info<<endl;
      }
else
      {
        cout<<"Review or Info=Not Available"<<endl;
}
```

```
    }
  }
  if (flag == 0)
  {
    cout<<"Entered Name Not Found"<<endl;
  }
}
void StockOfMedicine(int number)
{   int
i;
  if (number != 0)
  {
    cout<<"All Available Items are"<<endl;
    for (i = 0; i < number; i++)
    {
      if (m[i].id != 0)
      {
        cout<<"Id="<<m[i].id<<endl<<"Name="<<m[i].medicneName<<"\t\tPrice="<<m[i].price<<"\
tAvailable Quantity="<<m[i].quantity<<endl<<"Company="<<m[i].Company<<"\t\tMfg
Date="<<m[i].Mfg_Date<<"\tExp Date="<<m[i].Exp_Date<<endl;
if (strcmp(m[i].info, "") != 0)
        {
          cout<<"Review or Info="<<m[i].info<<endl;
```

```
        }
        else
{
          cout<<"Review or Info=Not Available"<<endl;
        }
      }
    }
  }
  else
  {
    cout<<"No Items or Medicines Available"<<endl;
  }
}
void AddMedicineinStore(int number, struct Medicine m[])
{
  char name[100];
  cout<<"Enter Medicine Id"<<endl;
  cin>>m[number].id;
fflush(stdin);
  cout<<"Enter Medicine Name"<<endl;
  fflush(stdin);
gets(name);
  strcpy(m[number].medicneName, name);
cout<<"Enter Company Name"<<endl;
  fflush(stdin);
  gets(m[number].Company);
  cout<<"Enter Manufactured Date"<<endl;
  fflush(stdin);
  gets(m[number].Mfg_Date);
cout<<"Enter Expiry Date"<<endl;
  fflush(stdin);
  gets(m[number].Exp_Date);
  cout<<"Enter Quantity"<<endl;
  fflush(stdin);
cin>>m[number].quantity;
```

```
  cout<<"Enter Price"<<endl;
  fflush(stdin);
cin>>m[number].price;
  strcpy(m[number].info, "");
  cout<<"Medicine with id"<<m[number].id<<"Added Successfully"<<endl;
}


void DeleteMedicineStore(int number)
{
  int id, i, flag = 0, num;
  cout<<"Enter Id to be deleted"<<endl;
fflush(stdin);
  cin>>id;
  for (i = 0; i < number; i++)
  {
    if (m[i].id == id)
    {
flag = 1;
      m[i].id = 0;
      m[i].price = 0;
m[i].quantity = 0;
      strcpy(m[i].medicneName, "");
strcpy(m[i].Company, "");
      strcpy(m[i].Mfg_Date, "");
      strcpy(m[i].Exp_Date, "");
```

```
      strcpy(m[i].info, "");
      num = i;
break;
    }
  }   if (flag
== 1)
  {
    cout<<"Medicine with "<<id<<" is Deleted Successfully"<<endl;
  }
}



void ChangeMedicineDetails(int number)
{
  int id, quantity, choice, c, i;
cout<<"Enter id to change Details"<<endl;
  cin>>id;
  for (i = 0; i < number; i++)
  {
    if (m[i].id == id && m[i].id != 0)
    {
do
      {
        cout<<"Enter\n1 - Change Quantity\n2 - Change Price\n3 - Change Name\n4 - Change
Company\n5 - Change Manufaturing Date\n6 - Change Expiry Date\n7 - Change Info\nAny other
number to exit";
cin>>choice;
        if (choice == 1)
        {
          int quantity;
```

```
          cout<<"Enter Quantity to be changed"<<endl;
fflush(stdin);
          cin>>quantity;
          m[i].quantity = quantity;
          cout<<"Quantity changed Successfully"<<endl;
        }
        if (choice == 2)
{
          int price;
          cout<<"Enter Price to be changed"<<endl;
          fflush(stdin);
          cin>>price;
m[i].price = price;
          cout<<"Price changed Successfully"<<endl;
        }
        if (choice == 3)
        {
          char name[100];
          cout<<"Enter Name to be changed"<<endl;
          fflush(stdin);
gets(name);
          strcpy(m[i].medicneName, name);
          cout<<"Medicine Name changed Successfully"<<endl;
}
        if (choice == 4)
        {
          char company[100];
          cout<<"Enter company to be changed"<<endl;
fflush(stdin);
          gets(company);
          strcpy(m[i].Company, company);
```

```
          cout<<"Company changed Successfully"<<endl;
        }
        if (choice == 5)
        {
          char mfg[11];
          cout<<"Enter Manufacturing date to be changed"<<endl;
          fflush(stdin);
gets(mfg);
          strcpy(m[i].Mfg_Date, mfg);
          cout<<"Manufacturing Date changed Successfully"<<endl;
}
        if (choice == 6)
        {
          char exp[11];
          cout<<"Enter Expiry date to be changed"<<endl;
fflush(stdin);
          gets(exp);
          strcpy(m[i].Exp_Date, exp);
          cout<<"Expiry Date changed Successfully"<<endl;
        }
        if (choice == 7)
{
          char info[100];
          cout<<"Enter Info to be changed(Less than 100 Characters)"<<endl;
          fflush(stdin);
          gets(info);
          strcpy(m[i].info, info);
```

```
          cout<<"Info changed Successfully"<<endl;
        }
        if (choice <= 0 && choice > 7)
        {
          cout<<"Enter valid Choice"<<endl;
        }
        cout<<"Enter 1 to Change other Details Else any other number"<<endl;
fflush(stdin);
         cin>>c;
      } while (c == 1);




      break;
    }
  }
```

Output:

```
---- SRM AP MEDICAL STORE ----

Enter
1 - Purchase Medicine
These are the details of Medicine
NameParacetmol
Price=120
Available Quantity=30
Company=ABCD
Mfg Date=23-04-2022
Exp Date=24-04-2025
Medicine Review/Info=Good Medicine for Fever
Do you want to purchaseParacetmol
If Yes Enter 1 else any other number
1
Enter Quantity to Purchase
3
Total Price to be paid=360
To Continue with other Options Enter 1 Else any other number
1
```

```
---- SRM AP MEDICAL STORE ----

Enter
1 - Purchase Medicine
2 - Enter Information about a Medicine
3 - Stock of Medicine in Store
4 - Medicine Information
5 - Add Medicine
6 - Delete a Medicine
7 - Change Quantity of Medicine
2
Enter Name of the medicine you want to Review or include Info
Paracetmol
These are the details of Medicine
Name=Paracetmol
Price=120
Available Quantity=30
Company=ABCD
Mfg Date=23-04-2022
Exp Date=24-04-2025
Review Already Available!
If you want to Add a Review Enter 1 else Any other number
Good Medicine !
To Continue with other Options Enter 1 Else any other number
```

```
---- SRM AP MEDICAL STORE ----

Enter
1 - Purchase Medicine
2 - Enter Information about a Medicine
3 - Stock of Medicine in Store
4 - Medicine Information
5 - Add Medicine
6 - Delete a Medicine
7 - Change Quantity of Medicine
3
All Available Items are
Id=1
Name=Paracetmol         Price=120       Available Quantity=30
Company=ABCD            Mfg Date=23-04-2022     Exp Date=24-04-2025
Review or Info=Good Medicine for Fever
To Continue with other Options Enter 1 Else any other number
```

```
---- SRM AP MEDICAL STORE ----

Enter
1 - Purchase Medicine
2 - Enter Information about a Medicine
3 - Stock of Medicine in Store
4 - Medicine Information
5 - Add Medicine
6 - Delete a Medicine
7 - Change Quantity of Medicine
6
Enter Id to be deleted
2
Medicine with 2 is Deleted Successfully
To Continue with other Options Enter 1 Else any other number
```

```
---- SRM AP MEDICAL STORE ----

Enter
1 - Purchase Medicine
2 - Enter Information about a Medicine
3 - Stock of Medicine in Store
4 - Medicine Information
5 - Add Medicine
6 - Delete a Medicine
7 - Change Quantity of Medicine
4
Enter Name of the medicine you want to see Review and Info
Paracetmol
These are the details of Medicine
Name=Paracetmol
Price=120
Available Quantity=30
Company=ABCD
Mfg Date=23-04-2022
Exp Date=24-04-2025
Review or Info=Good Medicine for Fever
To Continue with other Options Enter 1 Else any other number
```

```
---- SRM AP MEDICAL STORE ----

Enter
1 - Purchase Medicine
2 - Enter Information about a Medicine
3 - Stock of Medicine in Store
4 - Medicine Information
5 - Add Medicine
6 - Delete a Medicine
7 - Change Quantity of Medicine
5
Enter Medicine Id
2
Enter Medicine Name
Dolo
Enter Company Name
SRM AP
Enter Manufactured Date
06-12-2022
Enter Expiry Date
06-12-2023
Enter Quantity
3
Enter Price
120
Medicine with id2Added Successfully
```

```
---- SRM AP MEDICAL STORE ----

Enter
1 - Purchase Medicine
2 - Enter Information about a Medicine
3 - Stock of Medicine in Store
4 - Medicine Information
5 - Add Medicine
6 - Delete a Medicine
7 - Change Quantity of Medicine
7
Enter id to change Details
```

```
Enter
1 - Change Quantity
2 - Change Price
3 - Change Name
4 - Change Company
5 - Change Manufaturing Date
6 - Change Expiry Date
7 - Change Info
Any other number to exit3
Enter Name to be changed
Paracetamol_a
Medicine Name changed Successfully
```

```
Enter
1 - Change Quantity
2 - Change Price
3 - Change Name
4 - Change Company
5 - Change Manufaturing Date
6 - Change Expiry Date
7 - Change Info
Any other number to exit6
Enter Expiry date to be changed
06-12-2025
Expiry Date changed Successfully
```

```
Enter
1 - Change Quantity
2 - Change Price
3 - Change Name
4 - Change Company
5 - Change Manufaturing Date
6 - Change Expiry Date
7 - Change Info
Any other number to exit1
Enter Quantity to be changed
23
Quantity changed Successfully
```

```
Enter
1 - Change Quantity
2 - Change Price
3 - Change Name
4 - Change Company
5 - Change Manufaturing Date
6 - Change Expiry Date
7 - Change Info
Any other number to exit4
Enter company to be changed
SRM AP
Company changed Successfully
```

```
Enter
1 - Change Quantity
2 - Change Price
3 - Change Name
4 - Change Company
5 - Change Manufaturing Date
6 - Change Expiry Date
7 - Change Info
Any other number to exit7
Enter Info to be changed(Less than 100 Characters)
Good for Fever
Info changed Successfully
```

```
Enter
1 - Change Quantity
2 - Change Price
3 - Change Name
4 - Change Company
5 - Change Manufaturing Date
6 - Change Expiry Date
7 - Change Info
Any other number to exit2
Enter Price to be changed
150
Price changed Successfully
Enter 1 to Change other Details Else any other number
```

```
Enter
1 - Change Quantity
2 - Change Price
3 - Change Name
4 - Change Company
5 - Change Manufaturing Date
6 - Change Expiry Date
7 - Change Info
Any other number to exit5
Enter Manufacturing date to be changed
06-12-2022
Manufacturing Date changed Successfully
```

4. Concluding Remarks

The SRM Medical Store Management System provides a robust solution for streamlining medical store operations. By automating tasks such as inventory tracking, billing, and reporting, the software not only reduces human errors but also improves operational efficiency. This system ensures real-time updates for stock and sales, enabling better decision-making and enhancing customer satisfaction**.** Additionally, the platform's features, such as expiration monitoring and transaction management, help medical store staff save time and ensure compliance with regulatory standards. The system is designed to be user-friendly, scalable, and adaptable, making it suitable for various medical stores and pharmacies. In summary, the project successfully demonstrates how automation can transform routine operations into an efficient and seamless experience for both staff and customers, providing a foundation for further enhancements like AI integration and mobile application support.

5. Future Work

Advanced capabilities could be added to this pharmacy management software in the future to further optimise workflows and enhance user satisfaction. Among the possible future projects are:

- **Integration with E-Prescriptions**: Allowing clients to submit electronic prescriptions expedites order processing and lowers human error in the dispensing of medications.
- Using machine learning algorithms to forecast supply needs based on past sales data, seasonal patterns, and consumer preferences is known as **AI-Powered Demand Forecasting.**
- **Mobile Application Development**: Presenting an easy-to-use mobile application that allows customers and pharmacists to place orders, monitor inventories, and access the system while on the road.
- **Multi-Language Support**: Including support for several languages to accommodate a wide range of users in various geographical areas.

References

1. Mahatme, M. S., et al. "Medical store management: An integrated economic analysis of a Tertiary Care Hospital in Central India." *Journal of Young Pharmacists* 4.2 (2012): 114-118.
2. Bhura, S., Yadav, R., Suchak, P., Kohad, C., & Moroliya, T. (2022). An AI Based Model for Medical Store Management System. *Asian Journal of Organic & Medicinal Chemistry*.

Github link: https://github.com/AP23110010268/Srm-medical-store-management-system-