**C.V.RAMAN GLOBAL UNIVERSITY BHUBANESWAR, ODISHA- 752054**



TOPIC FOR CASE STUDY- MEDICAL STORE SYSTEM

BRANCH- COMPUTER SCIENCE & ENGINEERING

SEM- 4TH

GROUP :- 10

SUBJECT :- OOPS WITH JAVA (CSE23309)

GUIDED BY:- DR. DEBENDRA MUDULI

Dept. of computer science & engineering

SUBMITTED BY:-

1. PRATIK KUMAR SAHU - 2101020712

2. RAJNISH KUMAR – 2101020716

3. ARUN KUMAR BARIK – 2101020717

4. SATYA PRAKASH SAHU – 2101020718

5. DIBYA RANJAN SARANGI – 2101020719

6. SHUBHAJIT DAS – 2101020728

**Acknowledgement**

The Research Excellence and an effective report design always needs the assistance from various sides. It was not possible on my part to give a complete and perfect touch without any assistance. Therefore, I would liketo express my indebtedness to the following people for their unfailing assistance, enthusiasm and ingenuity. Firstly, I would like to express my heartiest gratitude to **Debendra Muduli** SIR, Department of Computer Science And Engineering for his constant support and guidance. I also thanks to my friends who helped me complete this report. Last but not the least I would like to extend my gratitude towards all the staff of C.V. Raman Global University (CGU), Bhubaneswar, Odisha for their timely cooperation, initiative, administration, assistance and suggestion and much needed encouragement.

**CONTENTS : -**

**❖ INTRODUCTION**

**❖ ALGORITHM/PROCEDURE**

**❖ IMPLEMENTATION USING JAVA**

**❖ OUTPUT OF THE PROGRAM**

**❖ TERMS USED FOR IMPLEMENTATION**

**❖ ADVANTAGES**

**❖ DISADVANTAGES**

**❖ APPLICATION AREA**

**❖ FUTURE SCOPES**

**❖ CONCLUSION**

**❖ REFERENCES**

**Introduction:-**

A medical store system in Java can be designed to automate various tasks performed in a medical store, such as maintaining inventory, managing sales, generating bills, and tracking customer information. Here is an overview of the system:

* Inventory Management: The system should have a feature to manage the inventory of medicines and other products available in the store. It should be able to add, edit, and delete items from the inventory, as well as track their stock levels.
* Sales Management: The system should have a feature to manage sales, including generating bills, tracking orders, and maintaining customer records. The system should also be able to provide reports on sales, such as daily, weekly, or monthly sales reports.
* User Management: The system should have a user management feature to ensure that only authorized personnel can access sensitive information, such as customer data, inventory levels, and financial records.
* Payment and Billing: The system should be able to generate bills for customers and process payments through various payment methods, such as cash, credit/debit cards, or mobile wallets.
* Reports: The system should provide reports on various aspects of the business, such as inventory levels, sales, and revenue. These reports can be generated on a regular basis or on demand.

**ALGORITHM/PROCEDURE:-**

Here is a high-level algorithm for a medical store system:

* Start the system.
* Initialize the user interface.
* Display the login screen.
* Prompt the user to enter the username and password.
* Authenticate the user credentials.
* If the credentials are correct, proceed to the main menu.
* If the credentials are incorrect, display an error message and prompt the user to re-enter the credentials.
* Display the main menu options, including inventory management, sales management, customer management, billing, and reporting.
* Prompt the user to select an option.
* If the user selects the inventory management option:

1. Display the inventory management screen.
2. Allow the user to add, edit, or delete products.
3. Update the inventory database accordingly.

* If the user selects the sales management option:

1. Display the sales management screen.
2. Allow the user to process orders, generate bills, and manage customer information.
3. Update the sales database and customer database accordingly.

* If the user selects the customer management option:

1. Display the customer management screen.
2. Allow the user to add, edit, or delete customer information.
3. Update the customer database accordingly.

* If the user selects the billing option:

1. Display the billing screen.
2. Allow the user to generate invoices and process payments.
3. Update the sales database and payment database accordingly.

* If the user selects the reporting option:

1. Display the reporting screen.
2. Allow the user to generate reports on various aspects of the business, such as sales, inventory levels, and revenue.

* After the user has completed their tasks, display a confirmation message.
* Log out the user and end the session.

Terminate the system.

**IMPLEMENTATION USING JAVA:-**

**import java.util.\*;**

**public class MedicalStoreSystem1 {**

**private static List<Medicine> medicineList = new ArrayList<>();**

**static List<Medicine> requestList = new ArrayList<>();**

**static List<Customers> customerList = new ArrayList<>();**

**static{**

**Medicine medicine = new Medicine("Dolox",500,10);**

**medicineList.add(medicine);**

**Customers c1 = new Customers("Arun    ", 20);**

**customerList.add(c1);**

**Customers c2 = new Customers("Satya   ", 18);**

**customerList.add(c2);**

**Customers c3 = new Customers("Rajnish ", 22);**

**customerList.add(c3);**

**Customers c4 = new Customers("Pratik  ", 24);**

**customerList.add(c4);**

**Customers c5 = new Customers("Subhajit", 20);**

**customerList.add(c5);**

**Customers c6 = new Customers("Dibya   ", 24);**

**customerList.add(c6);**

**}**

**public static void main(String[] args)throws InterruptedException**

**{**

**Scanner sc = new Scanner(System.in);**

**int choice = 0;**

**System.out.println("ENTER YOUR PASSWORD");**

**String pass = sc.nextLine().toLowerCase();**

**first:if(! pass.equalsIgnoreCase("ShopkeeperPass") && (pass.contains("buy medicine") || pass.contains("i don't have the password") ))//For Naive Users**

**{**

**do second:{**

**System.out.println("Welcome To Our Newly Built Medical Store System , Below is the menu");**

**System.out.println("        =========================       ");**

**System.out.println("\*\*\*\*\* | Medical Store System Menu | \*\*\*\*\*");**

**System.out.println("        =========================       ");**

**System.out.println("1. Search Medicine");**

**System.out.println("2. Buy Medicine");**

**System.out.println("3.Request For any medicine Not available");**

**System.out.println("4. Exit");**

**System.out.println("5.Enter Shopkeeper Mode");**

**System.out.print("Enter your choice: ");**

**choice = sc.nextInt();**

**switch (choice) {**

**case 1:**

**searchMedicine(sc);**

**break;**

**case 2:**

**BuyMedicine(sc);**

**break;**

**case 3:**

**reqMedicine(sc);**

**break;**

**case 4:**

**System.out.println("Thank you for using Medical Store System!");**

**break;**

**case 5:**

**System.out.println("You need to prove your Identity\nEnter Your Password");**

**sc.nextLine();**

**String shopPass = sc.nextLine().toLowerCase();**

**if(shopPass.equalsIgnoreCase(("shopkeeperpass")))**

**{**

**break second;**

**}**

**default:**

**System.out.println("Invalid choice! Please try again.");**

**break;**

**}**

**} while (choice != 4);**

**}**

**else if(pass.equalsIgnoreCase("ShopkeeperPass"))**

**{**

**do {**

**// System.out.println("Welcome To Our Newly Built Medical Store System");**

**System.out.println("        =========================       ");**

**System.out.println("\*\*\*\*\* | Medical Store System Menu | \*\*\*\*\*");**

**System.out.println("        =========================       ");**

**System.out.println("1. Add Medicine");**

**System.out.println("2. Search Medicine");**

**System.out.println("3. Sell Medicine");**

**System.out.println("4. Customers List");**

**System.out.println("5. Exit");**

**System.out.print("Enter your choice: ");**

**choice = sc.nextInt();**

**switch (choice) {**

**case 1:**

**addMedicine(sc);**

**break;**

**case 2:**

**searchMedicine(sc);**

**break;**

**case 3:**

**sellMedicine(sc);**

**break;**

**case 4:**

**System.out.println("NAME               POINTS");**

**for(Customers c : customerList)**

**{**

**System.out.println(c.getName()+"            "+c.getPoints());**

**}**

**break;**

**case 5:**

**System.out.println("Thank you for using Medical Store System!");**

**break;**

**default:**

**System.out.println("Invalid choice! Please try again.");**

**break;**

**}**

**} while (choice != 5);**

**}**

**sc.close();**

**}**

**private static void addMedicine(Scanner scanner) {**

**System.out.println("\*\*\*\*\* Add Medicine \*\*\*\*\*");**

**System.out.print("Enter medicine name: ");**

**String name = scanner.next();**

**System.out.print("Enter quantity: ");**

**int quantity = scanner.nextInt();**

**System.out.print("Enter price: ");**

**double price = scanner.nextDouble();**

**Medicine medicine = new Medicine(name, quantity, price);**

**medicineList.add(medicine);**

**System.out.println("Medicine added successfully!");**

**}**

**private static void reqMedicine(Scanner scanner) {**

**System.out.println("\*\*\*\*\* Request The Medicine \*\*\*\*\*");**

**System.out.print("Enter medicine name: ");**

**String name = scanner.next();**

**System.out.print("Enter quantity: ");**

**int quantity = scanner.nextInt();**

**System.out.print("Enter price: ");**

**double price = scanner.nextDouble();**

**Medicine medicine = new Medicine(name, quantity, price);**

**requestList.add(medicine);**

**System.out.println("We will get those Medicines for you in 1-2 working days!");**

**}**

**private static void searchMedicine(Scanner scanner) {**

**System.out.println("\*\*\*\*\* Search Medicine \*\*\*\*\*");**

**System.out.print("Enter medicine name: ");**

**String name = scanner.next();**

**for (Medicine medicine : medicineList) {**

**if (medicine.getName().equalsIgnoreCase(name)) {**

**System.out.println("Medicine found:");**

**System.out.println("Name: " + medicine.getName());**

**System.out.println("Quantity: " + medicine.getQuantity());**

**System.out.println("Price: " + medicine.getPrice());**

**return;**

**}**

**}**

**System.out.println("Medicine not found!");**

**}**

**private static void sellMedicine(Scanner scanner) {**

**System.out.println("\*\*\*\*\* Sell Medicine \*\*\*\*\*");**

**System.out.print("Enter medicine name: ");**

**String name = scanner.next();**

**for (Medicine medicine : medicineList) {**

**if (medicine.getName().equalsIgnoreCase(name)) {**

**System.out.print("Enter quantity to sell: ");**

**int sellQuantity = scanner.nextInt();**

**if (sellQuantity > medicine.getQuantity()) {**

**System.out.println("Insufficient stock!");**

**System.out.println("Remaining Stock : "+ (sellQuantity-medicine.getQuantity()));**

**return;**

**}**

**scanner.nextLine();**

**double totalCost = sellQuantity \* medicine.getPrice();**

**System.out.println("Enter The Name Of Customer");**

**String n=scanner.nextLine();**

**for(Customers c : customerList)**

**{**

**if(c.getName().trim().equalsIgnoreCase(n))**

**{**

**c.points+=(int)totalCost/10;**

**System.out.println("You earned : "+(int)totalCost/10+" points");**

**System.out.println("Do You Wanna Use Your Points?");**

**System.out.println("Press Y for Yes And N for No");**

**char ch = scanner.next().charAt(0);**

**if(ch=='Y')**

**{**

**totalCost-=c.points;**

**System.out.println("Total cost : " + totalCost);**

**System.out.println("Remaining Points : "+c.points);**

**}**

**else{**

**System.out.println("Total cost : " + totalCost);**

**System.out.println("Remaining Points : "+c.points);**

**}**

**return;**

**}**

**}**

**System.out.println("Total cost: " + totalCost);**

**int points = (int)totalCost/10;**

**System.out.println("You earned : "+points+" points");**

**Customers c = new Customers(n, points);**

**customerList.add(c);**

**medicine.quantity -= sellQuantity;**

**System.out.println("Medicine sold successfully!");**

**return;**

**}**

**}**

**System.out.println("Medicine not found!");**

**}**

**private static void BuyMedicine(Scanner scanner)throws InterruptedException {**

**System.out.println("\*\*\*\*\* Buy Medicine \*\*\*\*\*");**

**System.out.print("Enter medicine name: ");**

**String name = scanner.next();**

**for (Medicine medicine : medicineList) {**

**if (medicine.getName().equalsIgnoreCase(name)) {**

**System.out.print("Enter quantity You Wanna Buy: ");**

**int buyQuantity = scanner.nextInt();**

**if (buyQuantity > medicine.getQuantity()) {**

**System.out.println("Insufficient stock!");**

**System.out.println("Remaining Stock : "+ (buyQuantity-medicine.getQuantity()));**

**return;**

**}**

**scanner.nextLine();**

**double totalCost = buyQuantity \* medicine.getPrice();**

**System.out.println("Enter The Name Of Customer");**

**String n=scanner.nextLine();**

**for(Customers c : customerList)**

**{**

**if(c.getName().trim().equalsIgnoreCase(n))**

**{**

**c.points+=(int)totalCost/10;**

**System.out.println("You earned : "+(int)totalCost/10+" points");**

**System.out.println("Do You Wanna Use Your Points?");**

**System.out.println("Press Y for Yes And N for No");**

**char ch = scanner.next().charAt(0);**

**if(ch=='Y')**

**{**

**totalCost-=c.points;**

**System.out.println("Total cost : " + totalCost);**

**System.out.println("Remaining Points : "+c.points);**

**}**

**else{**

**System.out.println("Total cost : " + totalCost);**

**System.out.println("Remaining Points : "+c.points);**

**}**

**return;**

**}**

**}**

**Customers c = new Customers(n,(int)totalCost/10);**

**customerList.add(c);**

**System.out.println("You earned : "+(int)totalCost/10+" points");**

**System.out.println("Do You Wanna Use Your Points?");**

**System.out.println("Press Y for Yes And N for No");**

**char ch = scanner.next().charAt(0);**

**if(ch=='Y')**

**{**

**totalCost-=c.points;**

**System.out.println("Total cost : " + totalCost);**

**System.out.println("Remaining Points : "+c.points);**

**}**

**else{**

**System.out.println("Total cost: " + totalCost);**

**System.out.println("Complete The Payment And take your Receipt and after that you can take the medicine by providing your receipt");**

**System.out.println("Do You Want To Proceed? Y for yes | N for no");**

**char x=scanner.next().charAt(0);**

**if(x=='Y')**

**{**

**System.out.print("Completing Transaction..");**

**for(int i =1;i<=3;i++)**

**{**

**Thread.sleep(1000);**

**System.out.print(".");**

**}**

**System.out.println("\nTransaction Complete");**

**Thread.sleep(1000);**

**}**

**else**

**return;**

**}**

**medicine.quantity -= buyQuantity;**

**System.out.println("You can Take your medicines now by providing your receipt!");**

**System.out.println("Thank You");**

**Thread.sleep(1000);**

**return;**

**}**

**}**

**System.out.println("Medicine not found!");**

**}**

**}**

**/\***

**Cars.add("BMW")**

**Cars.remove(0)**

**Cars.get(0)**

**Cars.size()**

**Cars.contains("BMW")**

**Cars.set("1","VOLVO")**

**Cars.**

**for(String i : Cars)**

**System.out.println(i)**

**\*/**

**OUTPUT OF THE PROGRAM:-**

**ADVANTAGES:-**

There are several advantages of a medical store system, including:

1. Efficient management: A medical store system helps in the efficient management of medicines and other healthcare products. It helps to maintain an accurate inventory of the products, which reduces the risk of overstocking or stock-outs.

2. Improved patient care: With a medical store system in place, healthcare providers can quickly access the required medication for their patients, reducing the waiting time and improving the quality of care.

3. Cost-effective: The system helps to reduce the cost of storing and managing medical products by optimizing inventory levels and reducing wastage.

4. Improved accuracy: With a medical store system, there is a lower risk of errors in dispensing medicines as the system helps in verifying the accuracy of the medicines before dispensing.

5. Better record-keeping: A medical store system can help in maintaining accurate records of the medicines dispensed, which is useful for monitoring patient history and ensuring compliance with regulations.

6. Improved profitability: With better management of inventory levels, the system can help to increase profitability by reducing the cost of holding excess inventory and minimizing the risk of stock-outs.

Overall, a medical store system can help to improve the efficiency, accuracy, and profitability of a healthcare facility while also enhancing patient care.

**DISADVANTAGES:-**

While there are many advantages to a medical store system, there are also a few potential disadvantages that should be considered, including:

1. Cost: Implementing a medical store system can be expensive, especially for smaller healthcare facilities or clinics. This cost includes the initial investment in hardware and software, as well as ongoing maintenance and support costs.

2. Technical issues: Medical store systems rely heavily on technology, which means that technical issues such as software glitches or hardware failures can cause disruptions in service. This can lead to delays in patient care and can be frustrating for healthcare providers.

3. Training requirements: Healthcare providers need to be trained on how to use the system, which can be time-consuming and require additional resources. Additionally, staff turnover can lead to ongoing training requirements.

4. Security risks: Medical store systems contain sensitive patient data, including personal and medical information, which can be vulnerable to cyber threats such as hacking or data breaches. This can put patients at risk and damage the reputation of the healthcare facility.

5. Dependency on electricity: Most medical store systems are dependent on a constant supply of electricity. In areas where power outages are common, this can be a significant disadvantage as it can disrupt the operations of the healthcare facility.

Overall, while medical store systems offer many advantages, there are also some potential disadvantages that should be taken into account when considering the implementation of such a system.

**APPLICATION AREA:-**

Medical store systems are widely used in a variety of healthcare settings, including hospitals, clinics, pharmacies, and other healthcare facilities. Some of the common application areas of medical store systems include:

1. Inventory management: Medical store systems are primarily used to manage the inventory of medicines and other healthcare products. These systems can help to track the movement of products, monitor expiry dates, and reduce the risk of overstocking or stock-outs.

2. Patient care: Medical store systems can help to improve patient care by ensuring that the required medications are available when needed. These systems can also provide healthcare providers with real-time access to patient medication history, helping to improve treatment outcomes.

3. Billing and reimbursement: Medical store systems can help to streamline the billing and reimbursement process by accurately recording the medications dispensed to patients. This can help to reduce errors and improve the efficiency of the billing process.

4. Regulatory compliance: Medical store systems can help healthcare facilities to comply with regulatory requirements related to the storage and dispensing of medicines. These systems can provide accurate records of medicines dispensed, helping to ensure compliance with regulations.

5. Data analytics: Medical store systems can provide valuable data that can be used for data analytics and reporting. This data can help healthcare facilities to identify trends, track usage patterns, and make informed decisions about inventory management and other operational processes.

Overall, medical store systems are an essential tool in healthcare management and are used in a wide range of applications to improve patient care, increase efficiency, and ensure regulatory compliance.

**FUTURE SCOPES:-**

The future scope of medical store systems is vast and is expected to revolutionize the healthcare industry. Here are some of the potential developments that could shape the future of medical store systems:

1. Artificial intelligence (AI): AI could play a significant role in the future of medical store systems. AI algorithms could be used to analyze data on patient history, medication use, and inventory management, helping to optimize processes and improve patient outcomes.

2. Internet of Things (IoT): The IoT could enable medical store systems to be more interconnected, allowing for real-time monitoring of inventory levels, dispensing of medications, and tracking of patient outcomes. This could lead to improved efficiency, reduced waste, and better patient care.

3. Blockchain technology: Blockchain technology could be used to improve the security of medical store systems, ensuring that patient data is kept secure and protected against cyber threats.

4. Mobile technology: Mobile technology could enable medical store systems to be accessed from mobile devices, allowing healthcare providers to access patient information and medication records from anywhere, at any time.

5. Personalized medicine: Medical store systems could be used to support the delivery of personalized medicine, by tracking patient data and tailoring medication regimens to individual needs.

Overall, the future scope of medical store systems is exciting and holds great promise for improving patient outcomes, enhancing efficiency, and reducing costs in the healthcare industry. As technology continues to advance, we can expect to see more innovative developments in this field.

**CONCLUSION:-**

A medical store system is a software application that is designed to help manage the day-to-day operations of a medical store or pharmacy. The system can assist in inventory management, customer management, sales tracking, and reporting.

In conclusion, a medical store system is a valuable tool for any medical store or pharmacy to manage their operations efficiently. It can help store owners to keep track of their inventory, provide excellent customer service, and increase their revenue. With the right system in place, medical stores can streamline their operations and focus on providing quality service to their customers.

**REFERENCES :-**

1. "Pharmacy Management System: An Overview." International Journal of Scientific and Research Publications, vol. 5, no. 6, 2015, pp. 1-4.

2. "Medical Store Management System." International Journal of Engineering Research and General Science, vol. 3, no. 4, 2015, pp. 1546-1553.

3. "Design and Development of a Pharmacy Inventory Control System." Journal of Pharmaceutical Sciences and Research, vol. 8, no. 11, 2016, pp. 1288-1291.

4. "Evaluation of Pharmacy Management Information Systems in Hospitals." Iranian Journal of Pharmaceutical Research, vol. 15, no. 1, 2016, pp. 1-10.

5. "Pharmacy Management Systems: A Systematic Review." Journal of Medical Systems, vol. 42, no. 4, 2018, pp. 1-12.