

# **Pharmacy Management System**

## **Grpup\_project 30**

### **Proposal General View**

Our chosen proposal for our summative assessment was to make a complete Management System for a pharmacy that can handle all the financial and administrative operations of that pharmacy, starting from the employees, the customers, products, and the financial operations that may happen in that pharmacy, That will be in a simple way and in one application/program as the following and using the following classes:

- Our program will have a class to record all the authentication operations that will be needed to organize the employees and the HR-related processes. That can include adding or deleting employees, the ability to log in on the right screen, recording their operations, log\_in(s), and recording their absence, and so on.
- Our program will also have a class that will be responsible for managing the customers and their related operations, starting from adding customers, deleting them, editing their details and medication history, show their medication history to make sure giving them the right medication for his case, receive feedback from him, and even show a general summary for that customer if needed.
- We will also have a class for the general management of the products. This class will be responsible for managing the operations that will happen on the products, like adding a new product, deleting one, marking one as a missing product then ordering it automatically, or even showing an adjusted list of the current products.
- And as long as the pharmacy is a business that depends on selling medicines, we must have a class that will manage the selling or the cashier operations. This class will be responsible for identifying the current customer, choosing the medicines that he will buy and the quantity, processing the payment operation, creating and printing the receipt, and even recording the customer feedback for that time.
- Our last class will be the one that is very important for the pharmacy management, and the one that they will depend on to make sure everything is ok and so on. This last class will be the one that generates detailed reports and analyses. This class will be programmed to generate the profits report, salaries report, quantities report, rate and feedback report, and all different kinds of reports on a daily, monthly, or yearly basis, depending on the information provided by other previous classes.

With that application, we will be able to successfully manage our pharmacy and record every step taken by anyone inside the pharmacy, easily fix any problem we may face, and easily detect it.

---

## **Classes and Methods**

### **Class01\_Authentication: -**

In this class, we manage all the authentication operations that will be needed to organize the employees and the HR-related processes using the following classes:

- Add\_user  
A method for adding a new employee to the system so that he can log in and use the management system easily. In the adding process, the adding manager will have the chance to choose the type of employee he wants to add, his job, and his rank so that he limits what he can do and see depending on that.
- Delete\_user  
Just for deleting that user from the system if he has been fired or retired, with the ability to record the reason for deleting.
- Log\_in  
A method that will allow the authenticated user to log in, use the system and record his attendance, attendance time, and day automatically once logging in.
- Log\_out  
To log out from the system and be recorded as the end of the work-time.
- Point\_to\_Interface  
A method that comes after the user logs in to point him to the right interface for his position.

### **Class02\_Customers: -**

In this class, we manage all the authentication operations that will be needed to organize the employees and the HR-related processes using the following classes:

- Add\_customer  
Just for adding a new customer to the system, so that the pharmacy will be able to record the selling operation in his name
- Delete\_customer  
Just for deleting that customer from the system for any possible reason.
- Edit\_customer  
A method that will allow us to edit the customer's details, including the contact and the medication history and preferences if possible in case of any change.
- Show\_history  
To print a small overview report about the customer's medication history so that the doctor can have an idea about the best medication to give him, and take the right decision regarding his case.
- Show\_customers  
To show and print a brief report that shows all the customers recorded at the system and their number.

#### **Class03\_Product\_managment:** -

In this class, we will be able to manage all the operations that are related to managing the products in the warehouses of the pharmacy using the following classes:

- Add\_product  
To add any new product the pharmacy receives in its systems.
- Delete\_product  
For deleting any product that the pharmacy is not planning to sell it again.
- Edit\_product  
To edit any detail about any product, like the quantity, the price, and such details.
- Order\_product  
In case a product is missing from the pharmacy warehouse, by using this method they can order all the missing products from the exporter automatically.
- Show\_availability  
To show and print a brief report that shows all the available products, individually or all of it as a report.

#### **Class04\_Selling\_operations (Product\_managment, Customers):** -

In this class, we will be able to manage all the operations that are related to managing the products in the warehouses of the pharmacy using the following classes:

- Identify\_the\_customer  
To identify the customer who is buying in this current operation.
- The\_order  
To specify the products and the quantity that the customer is willing to order, in addition to defining the price with every product added to it.
- Pay  
To handle the payment process by choosing the payment type, the rest of the money, and recording it.
- Print\_receipt  
To create and print the receipt of the selling operation that happened, with all the details about the date, time, and the customer name in addition to the payment details.
- Rate\_feedback  
To take input from the user in two shapes: The first one is 1:5 rate, and the second one is written feedback. This step must be optional.

This class will inherit from the Product\_managment class because after the operation of selling we will have to edit the product quantity and order it if it was the last one to sell for example. In addition to that, it will inherit from the customer class so that we can add a new customer if the buyer is new or identify him from that class.

**Discarded..... (8/12/2021)**

**Class05\_Generating reports: -**

In this class, we will use all the provided classes above to generate some important reports and mathematical analysis that will have many benefits for understanding and improving the business and to take studied decisions. This will happen using the following methods:

- Profits  
This method will work in generating a full report about the profits that the pharmacy made in a specific amount of time. This report may include the pure income from all the products, the product that has the highest profits, and the month with the highest profits.
- Salaries\_report  
A report that will be generated each month will use the log-in information of the employees to calculate their salaries depending on the hours and the days they have worked.
- rate\_feedback\_report  
A method that will generate and print a report that contains some analysis and details about all the feedback the pharmacy got from the customers
- selling\_report  
A method that will generate and print a table of all the selling operations that happened in a specified amount of time.
- employees\_report  
A method that will generate and print a detailed report about a specific employee or all of them and their performance through the month/year.

