Software Description

Pharmacy Management System will be able to provide quick service to the patients with the help of previously saved records. However, new records can be added and old ones modified whenever a new patient or a previous patient visits the pharmacy. To identify the patient, each patient will be provided with a Patient Number and during the data entry process, all their required information will be added into the file in the binary format.

With the help of the Patient Number, the system will be easily able to retrieve their patient’s data and prepare for the bills.

Only the Admin will have the access to the entry, modification and retrieval of the data.

Work FlOW

**Admin Login**

**Opening Screen**

**Main Menu**

**Add Record Display Records Display Medicines Edit Record**

**prescribed**

**Password Re-authentication**

**Modify Record Delete Record**

**Billing EXIT**

Data Structure

This system is designed to ease the work in a pharmacy. It stores the details such as the patient number, name, diagnosis, medicine prescribed, name of the doctor consulted, and the contact number of the patients who visit the pharmacy in a binary file “patient.dat”.

The software consists of the following class :

* **class pharmacy :** stores the information of the patient.

1. patno – patient no. of integer type.
2. name – patient’s name as a character array
3. prob – diagnosis of the patient as a character array.
4. presmed – medicine prescribed as a character array.
5. doc – name of the doctor consulted as a character array.
6. phone - contact no. of the patient as a character array.

The class member functions used are as follows :

* void main\_menu() : To display the main menu.
* void add() : to input patient details.
* void display() : to display a patient’s record.
* void pres\_med() : to display all prescribed medicines.
* void edit() : to edit a patient’s record. ( modify or delete, based on choice)
* void modify(int) : to modify patient record.
* void delete\_patrec(int) : to delete a patient’s record.
* int check(int) : to check patient no. status (to check if patient number is already taken, while assigning a new patient number)
* void hosp\_header() : to display the opening screen of the software.
* void calc() : for billing.

The non-member functions (outside the class) are as follows :

* void authenticate() : to authenticate the admin by accepting the username and password.

**Data Files –**

1. patient.dat **:** contains the following elements –
2. patno – patient no. of integer type.
3. name – patient’s name as a character array
4. prob – diagnosis of the patient as a character array.
5. presmed – medicine prescribed as a character array.
6. doc – name of the doctor consulted as a character array.
7. phone - contact no. of the patient as a character array.

Project Requirement

* **Software requirements :**
* Requires a version of Borland C++ .
* Operating System : Windows XP/Vista/7/8/10.
* **Hardware requirements :**
* Processor : intel dual core or above.
* Processor speed : 1.0 GHZ or above.
* RAM : 1GB or above.
* Hard Disk : 20GB hard disk or above.

Bibliography

* <http://www.cplusplus.com>
* <https://www.learncpp.com/>
* Textbook for C++ by Sumita Arora.

Index

1. Project Requirement
2. Software Description
3. Data Structure
4. Workflow
5. Source Code
6. Output
7. Bibliography