Project Report

for

Medical Store Management System

Version 1.0 approved

Prepared by:

* Sunil Patil

29 May, 2020.

1. **INTRODUCTION**

In today’s highly competitive global industry, the demand for high quality products manufactured at low costs with shorter cycle times has forced manufacturing industries to consider various new product designs, manufacturing, information systems and management strategies. Most of industries will to increase productivity and implementing change can be a difficult task. Most of the Industries rely on changing work flow, increasing machines, improving human skills which can be an arduous task and there is need of investing money and time.

For this purpose, modeling and simulation is necessary to develop, validate and use it in practice. Various simulation software is available which can be used to study and compare alternative designs and shows the working of model without any prototype, thus saving cost and time. Simulation can be used to predict the way in which system will evolve and respond to its surrounding, so as to apply necessary changes that will help make the system perform the way that we want. Hence simulation has gained importance in past few years as it provides low cost, secure and fast analysis.

**1.1 Existing system**

The existing system for keeping the data of the medicines is manual which uses paper and requires more time to find medicines in a medical store, maintain their day to day transactions manually. A expired product need to be send back to the distributor which is difficult with the existing system. Many times proper work is required to maintain the store this stores need a full pledged software to maintain their day to day transactions.

**Limitations:-**

* Data is always collected in form of paper, so there is wastage of paper.
* Finding a product is difficult task.
* Billing is more time consuming.
* Tracking the management of store is difficult.
* Records may get lost.

**1.2 Proposed System**

The proposed help them in many ways. It help them do billing very easily. Account maintenance also become easier. They can keep track of their purchases, sales, stocks and account details. The system will is provided with all the master entries to enter any new product, or supplier, or customer to add or modify and delete. In the proposed system medicine will be identified by patient name, age which will be automatically generated.

**Advantages of proposed system:**

* Searching and manipulating patients medicine record is easy.
* The patient and medicine record can be store for longer.
* Billing record can be easily stored.
* Tracking of day to day transactions is easy.

1. **SYSTEM ANALYSIS**

Now a days we do all work with computerized system but with current system is very complicated to keep the track of all registers and handle them manually. As well as that medical shop management system project report work is time consuming & also expensive in that medical shop management system project report system report work may be not accurate and not fastest.

**2.1 Requirements**

In Medical store Management System the main purpose is to satisfy all the requirements needed to fulfill the system and so as to satisfy all needs. All the requirements are analyzed they are further specified thoroughly and finally the specified requirements are validated. Sales have the requirements to display daily sales of product whether the product is available in stock and to generate bills of products. Also there is requirement to give individual unique id for all products in the system so that track of all products is kept; while for customers also it is necessary to keep their own id’s and hence details of all purchases can be kept.

**2.1.1 Functional**

Functional requirements define the fundamental actions the system must perform. The functional requirements for the system are as below:

1. Add new product to the inventory
2. Remove a product from inventory
3. View all products in the inventory
4. Delete all products
5. Create new bill
6. View all bills
7. Remove a bill
8. Add new customer to the customer records
9. View all saved customers

**2.1.2 Non-functional**

Non-functional requirements define the needs in terms of performance, logical database requirements, design constraints, standards compliance, reliability, availability, security, maintainability and portability.

**2.2 Problem Definition:**

To develop Medical Store management system, for any shop with all requirements to be fulfilled like; adding new customer, sales management, product details, etc. Annual reports are to be generated on products managing older and newer customers. Product price is to be decided on the basis of the original price of the product and additional taxes (like GST, etc.). Inventories must be able to efficiently managed and unique id should be given to customers as well as every individual product.

**2.3 SRS- Software Requirement Specifications Document:**

A **software requirements specification** (SRS) is a detailed description of a software system to be developed with its functional and non-functional requirements. The SRS is developed based the agreement between customer and contractors. It may include the use cases of how user is going to interact with software system. The software requirement specification document consistent of all necessary requirements required for project development. To develop the software system, we should have clear understanding of Software system. To achieve this, we need to continuous communication with customers to gather all requirements.

A good SRS defines the how Software System will interact with all internal modules, hardware, communication with other programs and human user interactions with wide range of real-life scenarios. Using the Software requirements specification (SRS) document on QA lead, managers create test plan. It is very important that testers must be cleared with every detail specified in this document in order to avoid faults in test cases and its expected results.

1. Scope: The Medical Store Management System will be able to manage products in the inventory and allow to keep track of the customers and bills.
2. Overall description: The Medical Store Management System will allow users to add products to inventory, remove and view the products saved in the inventory. The system will also allow the user to create new bill, remove a bill and view all saved bills.
3. Specific requirements: The Medical Store Management software can be executed on Windows Operating System. The minimum hardware and software requirements are mentioned in the section 5.

**3.PROPOSED SYSTEM**

The proposed help them in many ways. It help them do billing very easily. Account maintenance also become easier. They can keep track of their purchases, sales, stocks and account details. The system will is provided with all the master entries to enter any new product, or supplier,or customer to add or modify and delete. In the proposed system medicine will be identified by patient name, age which will be automatically generated.

Some of the merits of proposed system are as follows:-

* **Accuracy:-**

It is one of the features of computer that medical store management system project report it provides accuracy result. It is often said, “Man does mistakes but machines never”.When a man works there seems numbers of errors and mistakes in performing calculation and other things, while it is not in case with computers.

* **Security:-**

The security is maintained vary comfortably in computers. Secretes are not disclosed off, as there are password settings in the computers. So the firm can get the benefit of non-disclosure of its secrets to the outsider competitors as well as other non-authorized persons.

* **Time saving:-**

The computerized system is time saving. On an immediate inquiry , the operator or any authorized person has to just click some switches and get the information in few seconds.

* **Perform Repetitive Task Very Well:-**

It is said that medical store management system project report computer never tires. I do the repetitive task very well as it never feels laziness. Performance of repetitive task by a man is not well

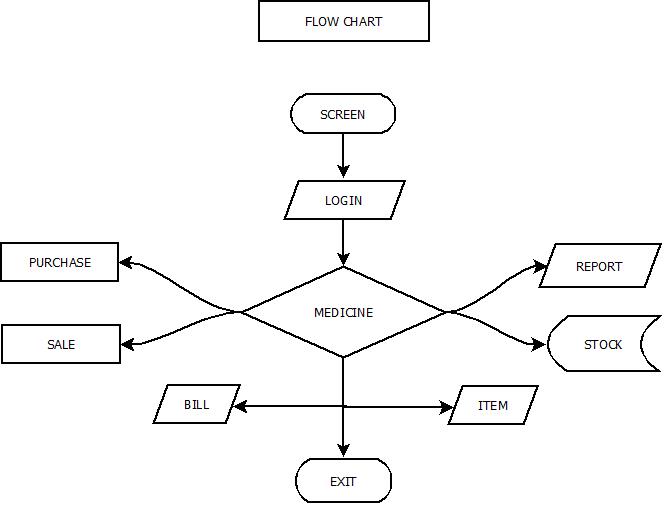
Performed, as main tires very quickly and feels lazy. It is not in the case with computer. So, proposed system is beneficial than Old one.

* **Low cost:-**
* The computer need not change year as it happens in case of registers. In register system clerk , employees and also managers are needed to fulfill the management activity while in computer system the two persons are enough for all the work. So in proposed system the two persons are enough for all the work . so in proposed system the two persons and expenses of registers are saved annually.

**PROPOSED SYSTEM**

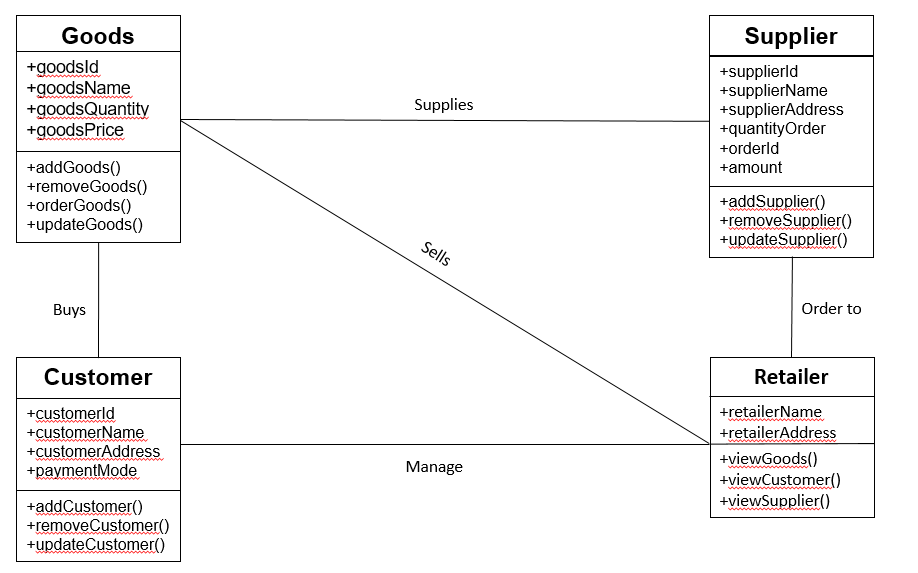
**3.1. DESIGN**

**3.1.1 System Design-DFD Level 1**

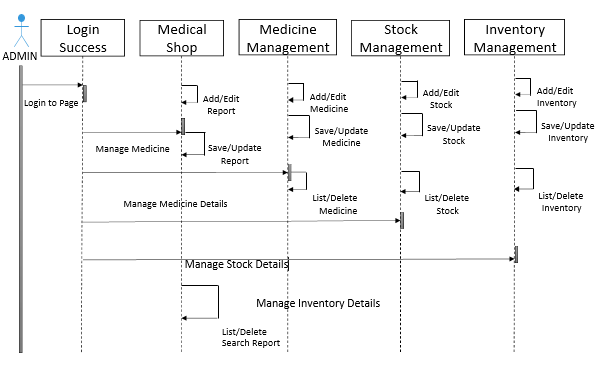


**3.2Design Document**

**3.2.1 Class Diagram:**



**3.2.2 Sequence Diagram**



**4.SYSTEM REQUIREMENT**

1 Hardware Requirement

* Intel processor
* 256 GB RAM
* 80 GB Hard Disk

2. Software Requirements:

|  |  |
| --- | --- |
| **Name of Component** | **Specification** |
| Programming Platform | C, C++ |
| Files | File handling in C |
| IDE | Dev C++, Code Blocks |
| Operating System | Windows 8.1, Windows 10, Linux |
| Browser | Google Chrome, Mozilla |

**5. IMPLEMENTATION**

The implementation of the Medical store Management System involves the writing and compilation of the program using C language .

We have used the following libraries functions

#include<stdio.h>

#include<conio.h>

#include<string.h>

#include<math.h>

#include<dos.h>

#include<time.h>

#include<ctype.h>

#include<windows.h>

**6. TESTING**

In a software development project, errors can be introduced at any stage during development. To ensure quality of the final delivered software these defects will have to be removed. During testing, the software under test(SUT) is executed with a finite set of test cases, and the behavior of the system of these test cases is evaluated to determine if the system is performing as expected.

While discussing testing, we commonly use terms like error, fault, failure, etc.

The term error is used in two different ways, it refers to the discrepancy between a computed, observed or measured value and the true specified or theoretically correct value. Fault causes a system to fail in performing its required function. The term error is also used to refer the defects. Failure is the inability of a system or component to perform are required function according to its specifications.

**6.1Verification:**

The verifying process includes checking documents design, code and program. It does not involve executing the code. Verification uses methods like review, walkthrough, inspection, desk checking, etc. It finds bugs early in the development cycle. Target is application and software architecture, specification complete design, etc.QA team does verification and make sure that software is as per requirement in SRS document. It comes before validation.

In inventory management all the requirements are verified check listed with the requirement document.

**6.1.1 Verification Techniques:**

1. **Formal** verification techniques rely on mathematical proof of correctness. In inventory management all requirements are checked whether they are correct and hence verified. Formally all the numerical values are checked if they are mathematically correct.
2. **Informal** verification techniques rely on subjective human reasoning. All the Human decisions and their opinion to review are taken into consideration and hence they are verified.
3. **Static** verification techniques assess the system by using the source code without executing it. The source code of the inventory is assessed line by line without execution and hence verification is done.
4. **Dynamic** verification techniques assess the system by executing it first. In inventory whole system is verified by assessing the system.

**6.1.2 Checklist for SRS:**

**General (SRS requirements) Checklist**

1. Is a functional overview of the system provided?

**Yes, all the required functions as per requirements are implemented.**

1. Have the software and hardware environments been specified?

**Yes**

1. If assumptions that affect implementation have been made, are they stated?

**No**

1. Are all the requirements, interfaces, constraints, definitions, etc. listed in the appropriate sections?

**Yes**

**Interface Checklist**

1. Are all inputs and outputs to the system specified?

**Yes**

1. Are all interface requirements between hardware, software, personnel, and procedures included?

**Yes**

**Behavioral Requirements Checklist**

1. Have all requirements described in the problem statement and in subsequent communications with the customer been specified?

**Yes, customer requirements are fulfilled**.

1. Are all inputs to a function sufficient to perform the required function?

**No.**

1. Are undesired events/inputs considered and their required responses specified?

**No.**

**Non-Behavioral Requirements Checklist**

1. Is the reliability specified, including the consequences of software failure, the vital information that needs to be protected from failure, and the strategy for error detection and recovery?

**No, since scope is limited.**

1. Are planned changes specified (i.e., maintainability)?

**Yes**

#### Requirements Quality:

1. Does each requirement avoid conflicts with other requirements?

**Yes, concurrency is provided.**

1. Does each requirement have a priority? (e.g. essential program functionality should not be on a low priority)

**Yes, as per priority and level each requirement is provided.**

1. Is each requirement testable? Will it be possible for independent testing to determine whether each requirement has been satisfied?

**Yes , required test cases are to be provided.**

**6.2 Validation:**

**Validation**is a dynamic mechanism of validating and testing the actual product. It always involves executing the code. It is computer-based execution of program. **Validation** uses methods like black box (functional) testing, grey box testing, and white box (structural) testing etc. **Validation** is to check whether software meets the customer expectations and requirements. It generally follows after **verification**.

**6.2.1 Levels of Testing**

The basic levels of testing are unit testing, integration testing, system testing and acceptance testing.These different levels of testing attempt to detect different types of faults.

1. The first level of testing is called unit testing.Unit testing is essentially for verification of the code produced by the individual programmers of the module.All the modules are integrated.
2. The next level of testing is called as integration testing.In this, many unit tested modules are combined into subsystem,which are then tested.The goal here is to see if the modules can be integrated properly, and thus the interfaces between modules.
3. The next levels are system testing and acceptance testing. Here the entire software system is tested. The reference document for is the requirement document. Acceptance testing is often performed with realistic data of the client.

In inventory management system every module is tested with unit testing.

Login page is tested and verified with valid username and password. Thus, further access is available.

Product module is tested whether all products are added with unique id and the required details and get added to file.

Customer module is also tested in isolation to see whether the new visiting customers are added properly with unique customer id.

It is checked whether the billing module is correctly, so GST with proper taxation is added for every product.

The acceptance testing is performed to validate the modules like product and bill perform in integration.

no of products -----> Bill generated with proper taxation

customers ----🡪Bill records and purchased list.

Acceptance testing is performed by verifying the client to satisfy by providing original details of client as customer and product are purchased and bill is generated.

Also, system testing, has been done to validate all requirements as provided in requirement document that is SRS.

**6.2.2Types of Testing**

There are two types of testing that is black box testing and the white box testing.

**Black Box Testing:**

In this testing type, the tester only knows only the inputs that can be given to the system and what output the system should give. The basis for deciding the test cases is the requirements or the specifications of the system or module. This form of testing is also called functional or behavioral testing.

In inventory management the admin login is the black box testing where only the inputs, different username with uppercase letters and password with whether satisfying all types of special characters and numeric values are tested and output is generated if successful; Also, customer is added with necessary input and output is generated.

**White Box Testing:**

White Box Testing on the other hand, is concerned with testing the implementation of the program. The intent of this testing is not to exercise all different input or output conditions but to exercise different programming structures used in the program. White box testing is also called structural testing.

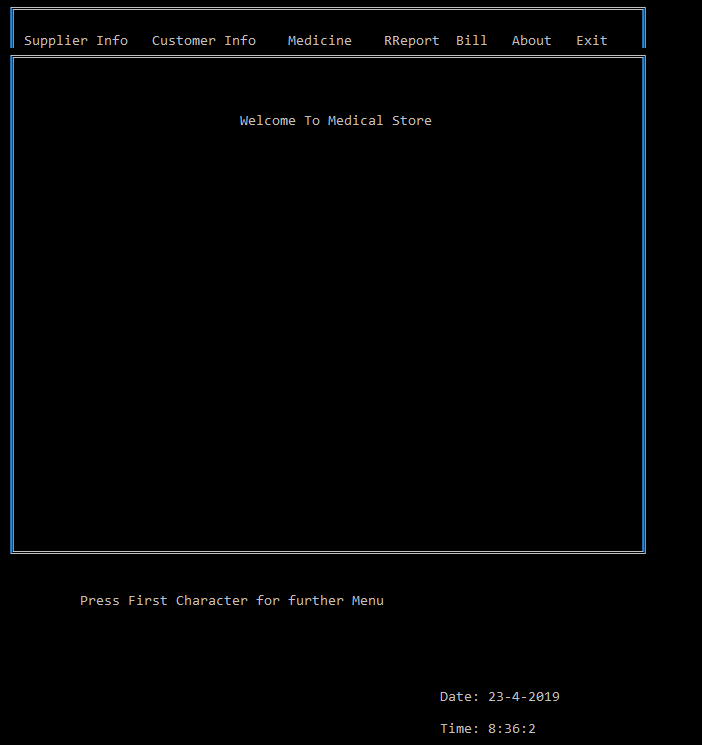
In inventory management the billing is tested using white box testing where the programming is implemented for adding GST taxation for every product and bill is calculated with current product price and reduced if any discount given.

Products for the customer is added and quantity is also given also new products with unque id are added to shop. Thus if any customer is old all the details are maintained and testing is provided whether old and new customers are identified.

**7. SCREENSHOTS**



User name and Password page



Main menu Page



Medicine menu page



Supplier Menu page

**8. Project Planning**

|  |  |
| --- | --- |
| **Month** | **Activity** |
| **January** | Topic search and selection |
| **February** | Gathering of required data |
| **March** | Implementation of coding |
| **April** | Testing and maintenance |

**9. Conclusion**

This system provides an easy way for the operator to interact with the database and manipulate the data in the database. The operator can add, delete and update the records in the database with ease. The medicines now can be easily monitored through software.

**10. References**

1. Becchetti, C.; Neri, A. (2013). "Chapter 6: Medical Software". [Medical Instrument Design and Development: From Requirements to Market Placements](https://books.google.com/books?id=P3VeLRVMdQ8C&pg=PT275). Chichester, U.K.: John Wiley & Sons Ltd. pp. 359–418. [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [9781119952404](https://en.wikipedia.org/wiki/Special:BookSources/9781119952404).
2. [Jump up to:**a**](https://en.wikipedia.org/wiki/Medical_software#cite_ref-VogelMed11_2-0) [**b**](https://en.wikipedia.org/wiki/Medical_software#cite_ref-VogelMed11_2-1) [**c**](https://en.wikipedia.org/wiki/Medical_software#cite_ref-VogelMed11_2-2) Vogel, D.A. (2011). "Chapter 3: The FDA Software Validation Regulations and Why You Should Validate Software Anyway". [Medical Device Software Verification, Validation, and Compliance](https://books.google.com/books?id=LYxH-zUSOTgC&pg=PA27). Boston, MA: Artech House. pp. 27–36. [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [9781596934238](https://en.wikipedia.org/wiki/Special:BookSources/9781596934238).
3. [Jump up to:**a**](https://en.wikipedia.org/wiki/Medical_software#cite_ref-JetleyMed13_3-0) [**b**](https://en.wikipedia.org/wiki/Medical_software#cite_ref-JetleyMed13_3-1) Jetley, R.; Sudarsan, S.; R., Sampath; Ramaswamy, S. (2013). "Medical Software - Issues and Best Practices". [Distributed Computing and Internet Technology: 9th International Conference, ICDCIT 2013, Bhubaneswar, India, February 5-8, 2013, Proceedings](https://books.google.com/books?id=5I25BQAAQBAJ&pg=PA69). Hyderabad, India: Springer. pp. 69–91. [ISBN](https://en.wikipedia.org/wiki/International_Standard_Book_Number) [9783642360718](https://en.wikipedia.org/wiki/Special:BookSources/9783642360718).
4. [Lockheed Hospital Information System](https://books.google.com/books?id=8vsFGwAACAAJ). Lockheed Aircraft Corporation. 1965. p. 82.