**INTRODUCTION**

1. **INTRODUCTION**

**1.1 ABOUT ORGANIZATION**

M.ES College, Nedumkandam, managed by Muslim Educational Society (Regd) Kozhikode, was started in 1982. The College is affiliated to Mahatma Gandhi University, Kottayam. Now the College has 7 graduate courses and 6 Post graduate courses.

The College being the only committed center of higher education in the area, serves the educational needs of generations. The College is situated in an ideal pleasant and calm location surrounded by cardamom and pepper plantations at Thazhakandam near Vattapara in Pampadumpara Panchayat of Udumbanchola Taluk in Idukki District.

The College is an eminent center playing a key role in the development of higher education.

**1.2 DEFINITION OF THE PROBLEM**

Today’s the technology is growing at a very vast speed therefore the work for the Pharmacy is becoming very much so to do the work in the accurate way and to improve the accuracy of the work and to make it more secure and safety Pharmacy management system was developed which manages the work flow of the Pharmacy. Many Pharmacies are still doing their whole work manually. Therefore, they need many workers to monitor all the process and to check the presence of each drug in Pharmacy.

So, when the new drugs or new batches of the drug arrive in the Pharmacy the manual entry is done in the register. And this also followed when the drug is given to any patients. When the month is completed the workers in the Pharmacy have to generate the list or report manually of the drugs in the Pharmacy shop. This work is done to maintain require stock in the Pharmacy. This kind of work may lead to mistake by workers and lead to a major problem.

Therefore, to solve this kind of problem the urgent need is to develop a Pharmacy management system that will prove beneficial for the Pharmacy. By using this software, we can generate bill, maintain the stock very well, we can do cost saving and maintain inventory control.

There are two main requirements of the software that is it must manage the stock of the Pharmacy and another is that it must produce the separate reports of purchase, sales and the stock.

1. **SYSTEM ANALYSIS**

**2.1 PRINCIPLES OF SYSTEM ANALYSIS:**

It is the most creative and challenging phase of the system life cycle. The analysis phase is used to design the logical model of the system whereas the design phase is used to design the physical model.

Many things are to be done in this phase. we began the designing process by identifying forms, reports and the other outputs the system will produce. Then the specify data on each were pinpointed. we sketched the forms or say, the displays, as expected to appear, on paper, so it serves as model for the project to begin finally we design the form on computer display, using one of the automated system design tool, that is VISUAL BASIC 6.0.

After the forms were designed, the next step was to specify the data to be inputted, calculated and stored individual data items and calculation procedure were written in detail. File structure such as paper files were selected the procedures were written so as how to process the data and procedures the output during the programming phase. The documents were design ion the form of charts.

Output design means what should be the format for presenting the results. It should be in most convenient and attractive format for the user. The input design deals with what should be the input to the system and thus prepare the input format. File design deals with how the data has to be stored on physical devices. Process design includes the description of the procedure for carrying out operations on the given data

**2.2 EXISTING SYSTEM**

Many pharmacies still use the manual system. Manual system requires the pharmacist to manually monitor everything in the pharmacy. This involves manual entry upon arrival of new batches of drugs and dispensing of drugs to patients etc. Upon a certain period (Ex: every month) the pharmacist is required to generate reports manually, on the movement of drugs. This is to monitor and refill the already diminishing stocks

Manual system is paper-based involving high amount of paper work and manpower requirement. This can usually lead to so many errors and the workload of pharmacist will increase. Even though computerized systems are used in some places, they are not efficient and are very complex and not user-friendly. So, the current pharmacy management software is very much user-friendly and flexible to meet user demands.

While in the existing system, it doesn’t have in built ability to look up database and work modules section and add new drugs upon confirmation of the admin. To store data to the database, one person has to look up the work and add data to populate database. Because of standalone system, no chances of using this system for multi user environment. Among all, the most important for any business process was not available such as daily logs report, not possible to make audit report and no functionality to prepare customized report as per business requirements.

**2.2.1 PROBLEM WITH THE EXISTING SYSTEM**

* Lack of security of data.
* Time consuming.
* Consumes large volume of paper work.
* Manual work
* No direct role for the higher officials.
* Very difficult to retrieve data from case files
* Redundancy of data may occur and this may lead to the inconsistency.

To avoid all these limitations and make the system working more accurately, it is needs to be computerized.

**2.2.2 OBJECTIVES OF THE PROPOSED SYSTEM**

The proposed system is very efficient and from the point of view of work, productivity and time.it is less tedious to manage operations in a medical store or pharmacy with such a system.it has all the required features and is developed with the aim of managing billing, stock management, report generation and relevant things in a medical store. With this software, the main operations in the pharmacy that has been creating hassles can be automated. Unlike the traditional system where the records are maintained manually, this system maintains records in database.

**2.3 PROPOSED SYSTEM**

Developing a system is feasible if and only if it is beneficial and removes all the drawbacks of the existing system and also enhances the way of operation to make the operation performing easy. The user should be satisfied with its functionality. If system does not fulfill this requirement, then developing that system would be futile. My basic aim is to develop this system is to improve its functionality and removes all the drawback of existing system.

The proposed system includes storing and retrieving medicine details, managing the stocks, and also computerized billing in the pharmacy. Our software has the facility to generate reports also. It includes a facility to know the stock details Of any item in the pharmacy. And the facility to search for any item in the database is also included.

It is accessible by an administrator and regular users by using a valid username and password. Only administrator can add and edit data, add users to system, and generate reports. The facilities for regular users are to view the stocks, and do the billing in the pharmacy. The interface is very user-friendly.

The proposed system is very easy to operate. Speed and accuracy are the main advantages of the proposed system. There is no redundancy of data. it can be easily retrieve and used at any time. The proposed system will easily handle all the data and store it safely for future reference. The proposed systems eliminate the drawbacks of the existing system at a great extent and it provides tight security to data.

**2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM**

The main Advantages of the proposed system can be enumerated as follows:

* + - * All the items are added to the system
      * Stock management is possible
      * New items can be added to the system at any time
      * Editing the stocks are also possible
      * Billing process can be managed easily
      * Avoid data Redundancy
      * Generating reports time to time is possible

**2.4 FEASIBILITY STUDY**

A feasibility analysis involves a detailed assessment of the need, value and practicality of a proposed enterprise, such as systems development. The process of designing and implementing record keeping systems has significant accountability and resource implications for an organization. Feasibility analysis will help you make informed and transparent decisions at crucial points during the developmental process to determine whether it is operationally, economically and technically realistic to proceed with a particular course of action.

Most feasibility studies are distinguished for both users and analysts. First, the study often presupposes that when the feasibility document is being prepared, the analyst is in a position to evaluate solutions. Second, most studies tend to overlook the confusion inherent in system development – the constraints and the assumed attitudes.

**2.4.1 OPERATIONAL FEASIBILITY**

People are inherently resistant to change, and computers have been known to facilitate change. An estimate should be made of how strong a reaction the user staff is likely to have toward the development of a computerized system. It is common knowledge that computer installations have something to do with turnover, transfers, retraining, and changes in employee job status. Therefore, it is understood that the introduction of a candidate system requires special effort to educate, sell and train the staff on new ways of conducting business.

**2.4.2 TECHNICAL FEASIBILITY**

Technical feasibility centers around the existing computer system (hardware, software, etc.) and to what extend it can support the proposed addition. For example, if the current computer is operating at 80 percent capacity – an arbitrary ceiling – then running another application could overload the system or require additional hardware. This involves financial considerations to accommodate technical enhancements. If the budget is a serious constraint, then the project is judged not feasible.

**2.4.3 COST/ BENEFIT ANALYSIS**

Economic analysis is the most frequently used method for evaluating the effectiveness of a candidate system. More commonly known as cost benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits overweigh costs, then the decision is made to design and implement the system. Otherwise, further justification or alterations in the proposed system will have to be made if it is to have a chance of being approved. This is an ongoing effort that improves in accuracy at each phase in the system life cycle.

**Costs:**

* Cost of new computer approximately Rs. 22,000/-
* Cost of operating system approximately Rs. 5000/-

**Benefits:**

* Avoids tedious typing task
* Faster document retrieval
* Saving storage space
* Keeps data secure
* Easy to use, update and maintain

**2.5 DATA FLOW DIAGRAM**

A data flow diagram is a graphical representation or technique depicting information flow and transform that are applied as data moved from input to output. The DFD are partitioned into levels that represent increasing information flow and functional details. The processes, data store, data flow, etc. are described in Data Dictionary.

**Data flow:**

Data moves in a specific direction from an origin to destination

**Process:**

Procedure s people or devices that use or transform data

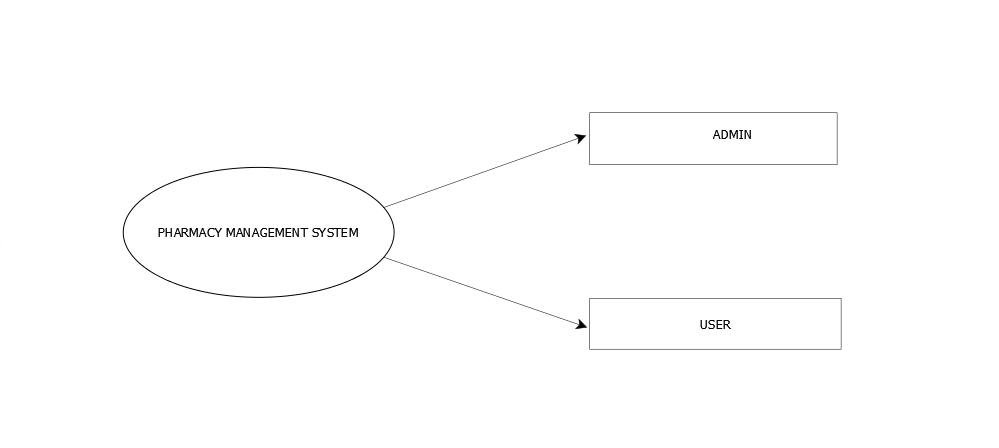
**External entity:**

This defines a source (originator) or destination of system data.

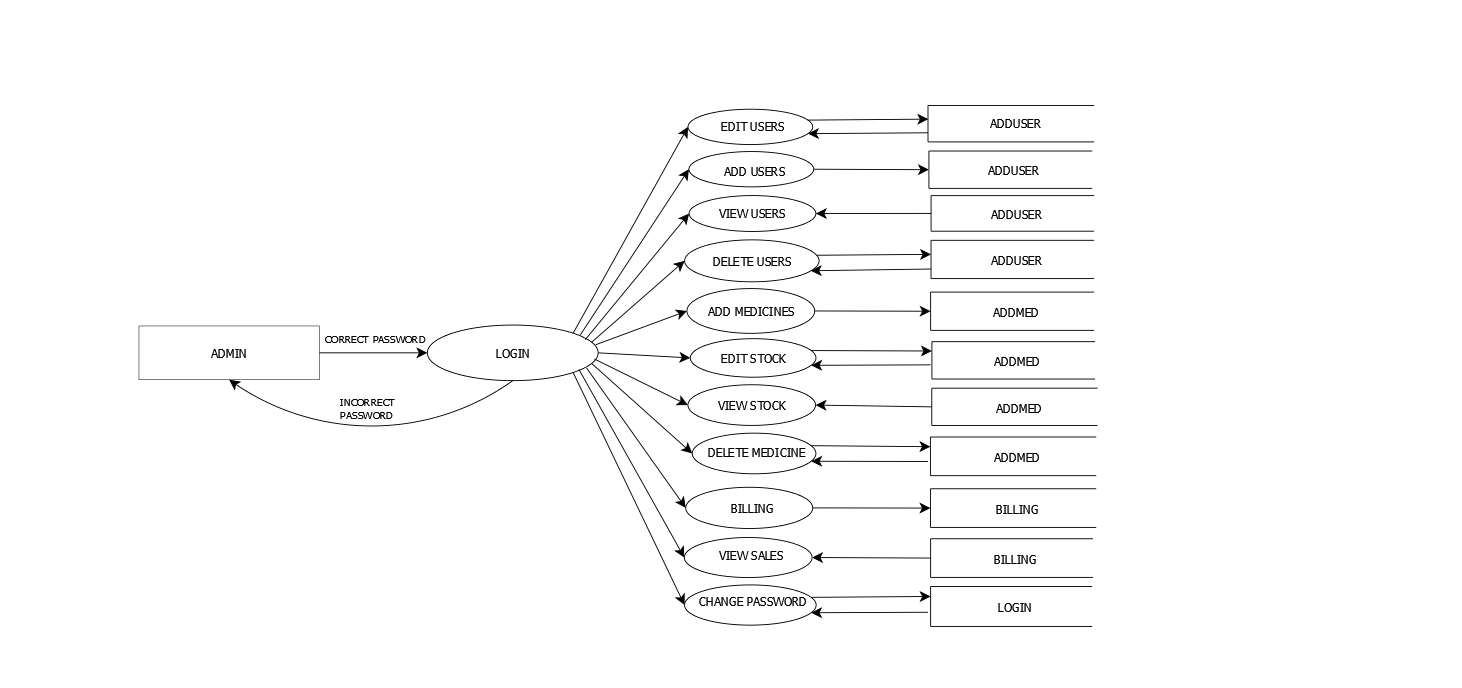
**Data Store:**

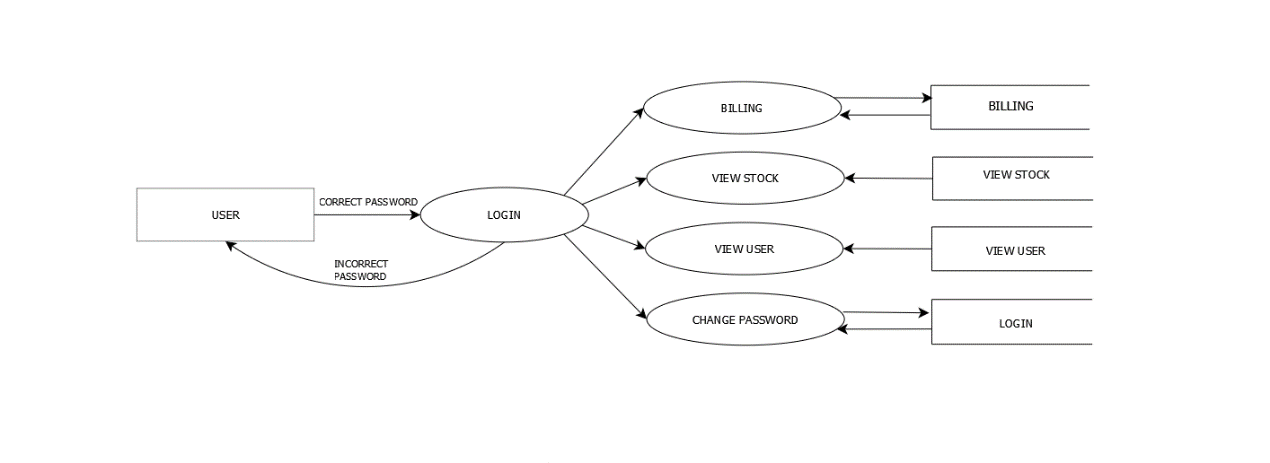
This indicates where data is stored in the system.

**CONTEXT DIAGRAM/ZEROTH LEVEL DFD**

****

**Level 1 DFD: Admin**

****

**Level 1 DFD: User**

**PROGRAMMING ENVIRONMENT**

* 1. **. PROGRAMMING ENVIRONMENT**
  2. **HARDWARE CONFIGURATION**
* CPU type : Intel i3 or AMD A8 Preferred
* Clock speed : 1Ghz minimum
* Ram size : 2gb
* Hard disk capacity : 40 GB
* Monitor type : SVGA Color
* Keyboard type : internet keyboard

**3.2 SOFTWARE CONFIGURATION**

* Operating System : Windows 10
* Language : Visual Basic 6.0
* Back End : SQL Server 2012
* Documentation : Microsoft Word 2016

**3.2 DBMS DESCRIPTIONS**

**SQL Server**

Relational database systems are the most important database systems used in the software industry today. One of the most outstanding systems is Microsoft SQL Server. SQL Server is a database management system developed and marketed by Microsoft.

The most important aspects of SQL Server are

* SQL Server is easy to use.
* SQL Server scales from a mobile laptop to symmetric multiprocessor systems.
* SQL Server provides data warehousing features that until now have only been available in Oracle and other more expensive DBMS.

A database system is an overall collection of different database software components and databases containing the parts viz. Database application programs, front-end Components, Database management systems, and Databases. A database system must provide the following features:

* A variety of user interfaces
* Query optimization
* Data integrity
* Concurrency control
* Backup and recovery
* Security and authorization.

SQL Server is a Relational Database Management System. SQL can query many rows from one or more tables using just one statement. This feature allows the use of this language at a logically higher level than procedural languages. Another important property of SQL is its non-procedurality. SQL contains two sub languages DDL and DML.

SQL Server works as a natural extension of Windows NT and windows 95/98.SQL Server is relatively easy to manage through the use of a graphical computing environment for almost every task of system and database administration. The SQL Server administrator’s primary tool for interacting with the system is Enterprise Manager. The Enterprise Manager has two main purposes: Administration of the database server and Management of database objects. SQL Server Query Analyzer provides a graphical presentation of the execution plan of a query and an automatic component that suggests which index should be used for a selected query.

This interactive component of SQL Server performs the tasks like:

* Generating and executing Transact-SQL statements
* Storing the generated Transact-SQL statements in a file
* Analyzing execution plans for generated queries
* Graphically illustrating the execution plan for a selected query.
* A stored procedure is a special kind of batch written in Transact-SQL using the SQL language and SQL extensions. It is saved on the database server to improve the performance and consistency of repetitive tasks. SQL Server supports stored procedures and system procedures. Stored procedures can be used for the following purposes: to control access authorization, to create an audit trial of activities in database tables, to separate data definition & data manipulation statements concerning a database & all corresponding applications.

The database object view can be used for:

* Restricting the use of particular columns and rows of tables –that is to control access to a particular part of one or more tables,
* To hide the details of complicated queries, to restrict inserted & updated values to certain ranges.

The Query Optimizer is the part of SQL Server that decides how to perform a query in a better way. It generates several query execution plans for the given query & selects the plan with the lowest cost.

SQL Server can operate in one of two security modes:

* Windows NT
* Mixed

Windows NT security mode exclusively uses Windows NT user accounts to log into the SQL Server system. Mixed mode allows users to connect to SQL Server using the Windows NT security system or the SQL Server system.

Additionally, Windows NT security provides three security facilities for controlling access to database objects:

* Transact-SQL statements GRANT, DENY, and REVOKE.
* Views.
* Stored procedures

A Windows NT user account or a SQL server login name allows a user to log into the SQL server system. A user who subsequently wants to access a database of the system needs a database user account to work in the DB. Therefore, users must have a DB user account for each DB they want to use. If there is no such account, the user may be allowed to work in the DB under the guest account.

Stored procedures can also be used to restrict data access. The restriction of data access using stored procedures is based upon the property that the permission to execute a stored procedure is independent of any permission for DB objects that are referenced by the stored procedure.

SQL server provides a mechanism called *a trigger* for enforcing procedural integrity constraints.

A DBMS handles 2 types of integrity constraints:

* Declarative Integrity constraints defined using CREATES & ALTER TABLE statements.
* Procedural integrity constraints handled by triggers.

SQL server keeps record of each change it makes to the DB during transactions. This is necessary in case an error occurs during the execution of the transaction. In this case, all previously executed statements within the transaction have to be rolled back. SQL server keeps all these records, in particular the before and after values, in one or more files called the transaction log. Each DB of the SQL server system has its own transaction log.

**3.4 FEATURES OF OPERATING SYSTEN**

Windows 10 delivers a refined, vastly improved vision for the future of computing with an operating system that's equally at home on tablets and traditional PCs -- and it's a free upgrade for most users. Windows 10 is the Goldilocks version of Microsoft's venerable PC operating system -- a "just right" compromise between the familiar dependability of Windows 7, and the forward-looking touchscreen vision of Windows 8.

This new Windows, is built from the ground up to pursue Microsoft's vision of a unified OS that spans all devices without alienating any one platform. It's an attempt to safeguard Microsoft's crumbling software hegemony, assailed on all sides by Google and Apple.

this new OS is chock-full of fresh features. To name just a few: a lean, fast Internet Explorer replacement called Edge; Microsoft's Siri-like voice-controlled virtual assistant, Cortana; and the ability to stream real-time games to your desktop from an Xbox One in another room. And in case you're wondering: there is no "Windows 9" -- Microsoft skipped it, going straight from 8 to 10.

Windows 10 is a welcome return to form. The Start menu, inexplicably yanked from 8, is back and working the way you expect it to. Those live tiles from the Windows 8 home screen still exist, but they've been attached to the Start menu, where they make a lot more sense. And the fiendishly hidden Charms bar has been morphed into the more straightforward (and easier to find) Action Center.

Windows 10 is the first step to an era of more personal computing, one in which Microsoft is moving Windows from its heritage of enabling a single device – the PC – to a world that is more mobile, natural, and grounded in trust. With Windows 10, applications, services, and content move across devices seamlessly and easily. Windows 10 features a universal app platform and universal store, providing a consistent experience across devices.

**3.5 Language Overview**

Visual Basic provides more of the actual code for a programmer than any other non-visual programming language. This makes it a widely-used programming language for visual applications

If you have ever programmed in the older BASIC or other command line programming language, then you will remember that the programmer had to write the code for entire user interface. Today’s windows, buttons, lists and other application features such as menus were not built-in to the BASIC programming language. Programmers had to create the code for these features on their own.

As much as 80% of a programmer’s time was spent writing code to create the user interface to his applications (the visual interface). To eliminate this huge drain on a programmer’s time, Microsoft had provided Visual Basic with built-in capability to create the user interface using nothing more than a mouse.

Visual Basic is itself a window application. You load and execute the VB system just as you do other Windows programs; we use this running VB program to create other programs. VB is just a tool, albeit an extremely powerful tool, that programmers use to write, test, and run windows applications.

**Visual Basic (VB)** is an event driven programming language and associated development environment created by Microsoft. In business programming, it has one of the largest user bases.

It is derived heavily from BASIC and enables rapid application development (RAD) of graphical user interface (GUI) applications, access to databases using DAO, RDO, or ADO, and creation of ActiveX controls and objects. A programmer can put together an application using the components provided with Visual Basic itself.

This built-in interface creation capability has had the future benefit of standardizing on the user interface to Windows applications. Today, user can move from one window program to another and see the same basic interface tools to work with- allowing them to concentrate solely on the unique capabilities of the application.

The bottom line is that you can create an entire application shell (the user interface) very quickly and then spend most of your time working on the features, which differentiate your application from its competitor. Today you need much more than just a language; you need a graphical development tool that can work inside the window system and applications that take advantage of all the graphical, multimedia, online and multiprocessor activities that windows offer.

Visual Basic is such a tool. More than a language, Visual Basic lets you generate applications that interact with every aspect of today’s windows operating systems.

**LANGUAGE FEATURES**

Visual Basic was designed to be usable by all programmers, whether novice or expert. The language is designed to make it easy to create simple GUI applications, but also has the flexibility to develop fairly complex applications as well. Programming in VB is a combination of visually arranging components on a form, specifying attributes and actions of those components, and writing additional lines of code for more functionality. Since default attributes and actions are defined for the components, a simple program can be created without the programmer having to write many lines of code. Performance problems were experienced by earlier versions, but with faster computers and native code compilation this has become less of an issue.

Although programs can be compiled into native code executables from version 5 onwards, they still require the presence of runtime libraries of approximately 2 MB in size. This runtime is included by default in Windows 2000 and later, but for earlier versions of Windows it must be distributed together with the executable.

Forms are created using drag and drop techniques. A tools palette is used to place controls (e.g., text boxes, buttons, etc.) on the form (window). Controls have attributes and event handlers associated with them. Default values are provided when the control is created, but may be changed by the programmer. Many attribute values can be modified during run time based on user actions or changes in the environment, providing a dynamic application. For example, code can be inserted into the form resize event handler to reposition a control so that it remains centered on the form, expands to fill up the form, etc. By inserting code into the event handler for a keypress in a text box, the program can automatically translate the case of the text being entered, or even prevent certain characters from being inserted.

A Visual Basic application can consist of one or more windows, or a single window that contains MDI child windows, as provided by the operating system. Dialog boxes with less functionality (e.g., no maximize/minimize control) can be used to provide pop-up capabilities. Controls provide the basic functionality of the application, while programmers can insert additional logic within the appropriate event handlers. For example, a drop-down combination box will automatically display its list and allow the user to select any element. An event handler is called when an item is selected, which can then execute additional code created by the programmer to perform some action based on which element was selected, such as populating a related list.

Alternatively, a Visual Basic component can have no user interface, and instead provide ActiveX objects to other programs via Component Object Model (COM). This allows for server-side processing or an add-in model.

The language is garbage collected using reference counting, has a large library of utility objects, and has basic object oriented support. Since the more common components are included in the default project template, the programmer seldom needs to specify additional libraries. Unlike many other programming languages, Visual Basic is generally not case sensitive, although it will transform keywords into a standard case configuration and force the case of variable names to conform to the case of the entry within the symbol table entry. String comparisons are case sensitive by default, but can be made case insensitive if so desired.

Visual Basic was designed to meet all real-world requirements with its key features, which are explained as follows: -

1. **Event Driven Programming Language:** -

Visual Basic makes it easy to locate event procedures code for controls on forms. Double-click any control to see one of its event procedures. For example, if you double-click the Exit command button, Visual Basic opens window and places the text cursor in the set of lines. The event procedures code, however, doesn’t do anything until runtime.

1. **Graphical user interface** -  
   In graphical user interface user can work with several windows at same time, within each window user can work on specific task. User can overlap windows, hide, change size and locations and even shrink down to icons. Using mouse user can move from one window to another, the window which is selected is an active window, most GUI are based on x-window system, x-window is a system used for supporting GUI.
2. **Database Connectivity:** -  
   Some databases, such as Microsoft Access, store all the related database files in a single global file called the database file. Inside the database, the individual groups of records and fields are called *tables*. Other database systems, such as dBase, keep track of a database’s data in multiple files. When you use database such as Microsoft Access, you must describe both the overall database and the individual table name within the database that the Data control is in use.
3. **Controls:** -  
   Controls are tools on the Toolbox window that you place on a form to interact with the user and control the program flow. The controls can be self-generated and can be used again and again in the same application. As you add controls to the Form window, the properties window updates to show the properties for the currently selected control. The selected control is usually the control you last placed on the form. Visual Basic lets you see a control’s properties in the Properties window by clicking to select the control or by selecting the control from the properties window’s drop-down list box.

**Benefits of Using VISUAL BASIC 6.0**

* VB now supports a true compiler that creates standalone runtime .exe files that execute more quickly than previous VB programs
* VB also includes several wizards that offer step-by-step dialog box questions that guide you through the creation of applications.
* VB’s development platform, a development environment called the Developer Studio.
* Now supports the same features as the advanced Visual C++ and Visual J++ Compilers.
* After you learn one of Microsoft’s visual programming products, you will have the skills to use the other language products without a long learning curve ahead of you.
* Visual Basic lets you generate applications that interact with every expect of today’s Windows operating systems.
* VB’s programming language is fairly simple and uses common English words and phrases for the most part.
* Microsoft Visual Basic 6.0, the latest and greatest incarnation of the old language, gives you a complete Windows application development system in one package.
* Visual Basic lets you write, edit, and test Windows applications. VB includes tools that can be used to write and compile help files, ActiveX controls and even Internet applications.

**SYSTEM DESIGN**

**4.0 SYSTEM DESIGN**

The design document that we will develop during this phase is the blueprint of the software. It describes how the solution to the customer problem is to be b built. Since solution to complex problems isn’t usually found in the first try, iterations are most likely required. This is true for software design as well. For this reason, any design strategy, design method, or design language must be flexible and must easily accommodate changes due to iterations in the design. Any technique or design needs to support and guide the partitioning process in such a way that the resulting sub-problems are as independent as possible from each other and can be combined easily for the solution to the overall problem.

The design of a system is essentially a blueprint or a plan for a solution for the system. The design activity begins when the requirements document for the software to be developed is available. Design is essentially the bridge between requirements specification and the final solution for satisfying the requirements.

The design process for software systems, often, has two levels. At the first level, the focus is on deciding which modules are needed for the system, the specifications of these modules, and how the modules should be interconnected. This is what is called the system design or top-level design. In the second level, the internal design of the modules, or how the specifications of the module can be satisfied, is decided. This design level is often called detailed design or logic design. Detailed design essentially expands the system design to contain a more detailed description of the processing logic and data structures so that the design is sufficiently complete for coding. Because the detailed design is an extension of system design, the system design controls the major structural characteristics of the system. The system design has a major impact on the testability and modifiability of a system, and it impacts its efficiency. Much of the design effort for designing software is spent creating the system design.

**4.1 DATABASE DESIGN**

Data design or architecture creates a model of data and information that is represented at a high of abstraction. This data model is then refined in to progressively more implementation specific representation that can be processed by the computer based system. The structure of data has always been an important part of software design. At the procedural level, the design of data structures and the associated algorithms required to manipulate them is essential to the creation of high quality application.

The general theme behind a database is to handle information as an integrated whole. A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and effectively. After designing the input and output, the analyst must concentrate on database design or how data should be organized around user requirements. The general objective is to make information access, easy, quick, inexpensive and flexible for other users. During database design, the following objectives are concerned: -

* + Controlled redundancy
  + Easy to learn and use
  + More information and low cost
  + Accuracy
  + Integrity

**4.2 INPUT DESIGN**

The input to the design phase is the specifications for the system to be designed. Hence, reasonable entry criteria can be that the specifications are stable and have been approved, hoping that the approval mechanism will ensure that the specifications are complete, consistent, unambiguous, etc. The output of the top-level design phase is the architectural design or the system design for the software system to be built. This can be produced with or without using a design methodology. A reasonable exit criteria for the phase could be that the design has been verified against the input specifications and has been evaluated and approved for quality.

A design can be object-oriented or function-oriented. In function-oriented design, the design consists of module definitions, with each module supporting a

functional abstraction. In object-oriented design, the modules in the design represent data abstraction.

**4.3 OUTPUT DESIGN**

Output Design of the computer is the most important and direct with the system. The output design phase of the system design is concerned with the conveyance of information to the end users in a user-friendly manner. The output design should be efficient, intelligible so that the systems relationship with the end user is improved and thereby enhancing the process of decision making.

The output design is an ongoing activity almost from the beginning of the project, and follows the principles of form design. Efficient and well-defined output design improves the relation of the system and the user, thus facilitating decision making. The primary considerations in the design of the output are the requirement of the information and the objectives of the end users.

**SYSTEM DEVELOPMENT**

**5.0 SYSTEM DEVELOPMENT**

**5.1 SYSTEM SPECIFICATIONS**

This program can be used in any pharmaceutical shops having a database to maintain. The software used can generate reports, as per the user’s requirements. The software can print invoices, bills, receipts etc. It can also maintain the record of stocks and maintain them accordingly.

**5.2 TABLES**

**Table Name: - Login**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Description** | **Data Type** | **Constraint** |
| userName | User Name | varchar(50) | Primary key |
| Password | Password | varchar(50) | Not Null |
| Type | User Type | varchar(50) | Not Null |

**Table Name :-Addmed**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Description** | **Data Type** | **Constraint** |
| Medid | Medicine ID | varchar(50) | Primary key |
| Medname | Medicine name | varchar(50) | Not Null |
| Bno | Batch number | varchar(50) | Not Null |
| Category | Medicine category | varchar(50) | Not Null |
| Mfgname | Name of mfg | varchar(50) | Not Null |
| Qty | Quantity in stock | varchar(50) | Not Null |
| Exp | Expiry date | Date | Not Null |
| Mfgdate | Mfg date | Date | Not Null |
| Buyrs | Buying price | Real | Not Null |
| Sellrs | Selling price | Real | Not Null |

**Table Name: - adduser**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Description** | **Data Type** | **Constraint** |
| stfname | Name of staff | varchar(50) | Not null |
| empid | Staff id | varchar(50) | Primary key |
| gender | Gender | varchar(50) | Not null |
| dob | Date of birth | Date | Not null |
| blood | Blood group | varchar(50) | Not null |
| address | Address | varchar(50) | Not null |
| cno | Contact no. | Int | Not null |
| email | Email | varchar(50) | Not null |
| joindate | Joining date | Date | Not null |
| salary | Salary | Real | Not null |

**Table Name :- billing**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Description** | **Type** | **Constraint** |
| Medname | Medicine name | varchar(50) | Not Null |
| cat | Catagery | varchar(50) | Not Null |
| mfgname | Manufacturer name | varchar(50) | Not Null |
| mfgdate | Date of Manufacture | date | Not Null |
| expdate | Expiry date | date | Not Null |
| custname | Customer name | varchar(50) | Not null |
| qty | Quantity bought | integer | Not Null |
| ppu | Price per unit | Integer | Not null |
| total | Total price | integer | Not Null |
| Billno | Bill number | Integer | Primary key |
| Billdate | Bill date | Date | Not null |

**SYSTEM IMPLEMENTATION**

**6.0 SYSTEM IMPLEMENTATION**

**6.1 TESTING, TRAINING & IMPLEMENTATION**

A crucial in the system life cycle is the successful implementation of the new system design. Implementation is the process of converting a new or a revised system design in to an operational one. The first task is implementation planning, deciding on the methods and time case to be adopted.

**IMPLEMENTATION PROCEDURES**

The main aspects of implementation are

* Conversion
* Post-implementation review
* Software maintenance

Implementation of a computer system to replace a manual system involve converting files, training users, creating accurate files and verifying printers for integrity.

**SOFTWARE MAINTENANCE**

Software maintenance is the enigma of software development. It holds the software industry captive, typing up programming resources .It is perceived as requiring either skill or experience. Maintenance covers wide range of activities, including correcting, coding and design errors, updating documentation and test data and upgrading user support.

**SYSTEM TESTING**

Testing is a set of activity that can be planned in advance and conducted systematically. Testing begins at the module level and work towards the integration of entire computers and based system. Nothing is complete without testing, as it is vital success of the system.

**TESTING OBJECTIVES:**

There are several rules that can serve as testing objectives. They are,

* + Testing is a process of executing a program with the context of finding an error.
  + A good test case is one that has high probability of finding an undiscovered error.
  + A successful test is one that uncovers an uncovered error.

If testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also, testing demonstrate that software functions appear to the working appear to the working according to the specification, that performance requirement appears to have been met. There are three ways to test a program,

* + Error correction
  + For implementation efficiency
  + For computational complexity

Tests for correctness are supposed to verify that a program does exactly what it was designed to do.

System testing is the stage of implementation, which aims at ensuring the accuracy of the system and its efficiency before the commencement of live operation. Testing is vital to the success of the system.

The entire testing process was divided into three phases.

Unit testing

Integrated testing

Validation testing

**UNIT TESTING:** It focuses verification efforts on the smallest unit of software design of a module. This is also known as module testing. The modules of the system are tested separately. This testing was carried out during programming stage itself. In this testing each module is found to work satisfactorily with test data then the testing become successful. Unit testing is successfully done for each module.

**INTEGRATION TESTING:** Data can be lost across an interface, one module can have an adverse effect on another, sub-function when combined, may not produce the devised major functions. Integration testing is systematic testing for constructing the program structure, while at the same time conducting testing to find undiscovered errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole. Here all the modules are integrated and tested. This testing provides the assurance that the application is well integrated functional unit with smooth transition of data. Integration testing is done after successful completion of unit testing.

**VALIDATION TESTING:** This provides the final assurance that the software meets all functional, behavioral and performance requirements. The software is completely assembled as a package. Validation succeeds when the software functions in a manner in which the user expects.

Validation refers to the process of using software in a live environment in order to find errors. During the course of validating the system, failures may occur and sometimes the coding has to be changed according to the requirement. Thus, the feedback from the validation phase generally produces changes in the software.

Once the application was made free of all logical and interface errors, inputting dummy data ensured that the software developed satisfied all the requirements of the user.

**Software Training:** Software Training and Support is important and a lot of developers fail to realize that. It would not matter how much time and planning a development team puts into creating software if nobody in an organization ends up using it. People are often resistant to change and avoid venturing into an unfamiliar area, so as a part of the deployment phase, it is very important to have training classes for\

new clients of your software. Another training element is training demonstration.

The third element is the resident expert. For example, one clerk read manual carefully, spent time on his practice and ended up being resident expert-a natural teacher.

**Documentation**: Software documentation or source code documentation is written text that accompanies computer software. It both explains how it operates or how to use it, and may mean different things to people in different roles. Documentation is an important part of software engineering. Types of documentation include:

* Requirements- Statements that identify attributes capabilities, characteristics, or qualities of a system. This is the foundation for what shall be or has been implemented.
* Architecture/Design- Overview of software. Includes relations to an environment and construction principles to be used in design of software components.
* Technical-Documentation of code, algorithms, interfaces, and APIs.

End user- Manuals for the end-user, system administrator and support staff.

**CONCLUSION AND FUTURE SCOPE**

**7.1 CONCLUSION AND FUTURE SCOPE**

* Detailed information gathering has to be done. Without that the purpose for using the software won’t be satisfied properly.
* However, it can give good profits in the long run.
* Implementing the software requires change in the business practices.
* Efficient organization of all knowledge is the analysis company and easy analysis access and retrieval of information is possible.
* In this project, we can also include BAR CODE facility using the bar code reader, which will detect the expiry date and the other information about the related medicines.
* Company using this software will always be able to plan in future and always be aware of their financial position in the market.
* It leads to streaming of business processes.
* The implementation and miniatous costs run very high (about 2 to 3 % of the company’s revenue.)

**BIBLIOGRAPHY**

**8.1 BIBLIOGRAPHY**

**Books**

* A Complete Visual Basic 6 Training Course: How to Programme: Package

-Harvey M deitel, Paul j deitel, Tem R Neito

* Advanced Programming Using Visual Basic: Version 6.0

-Julia Case Bradley

* Advanced MS Visual Basic

-Peter Morris

* Introduction: Visual Basic 6.0

-Garry Haggard; Wade Hutchison; Christy Shibata

* Microsoft SQL server black book

-Patrick Dalton

* Introducing Microsoft SQL Server 2014 Technical Overview

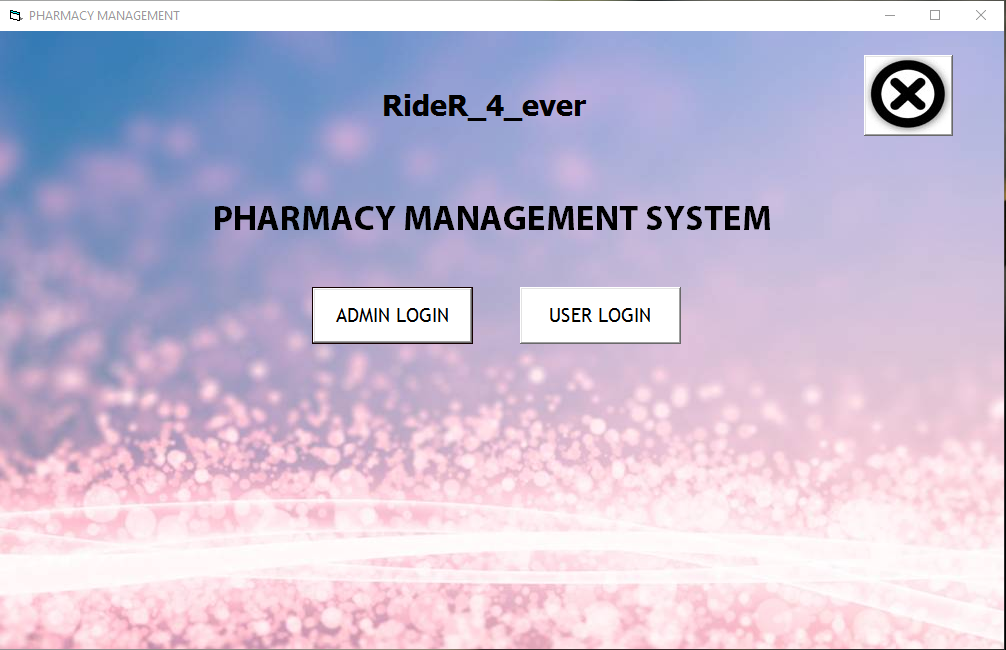
-Ross Mistry ; Stacia Minser

**Site Address**

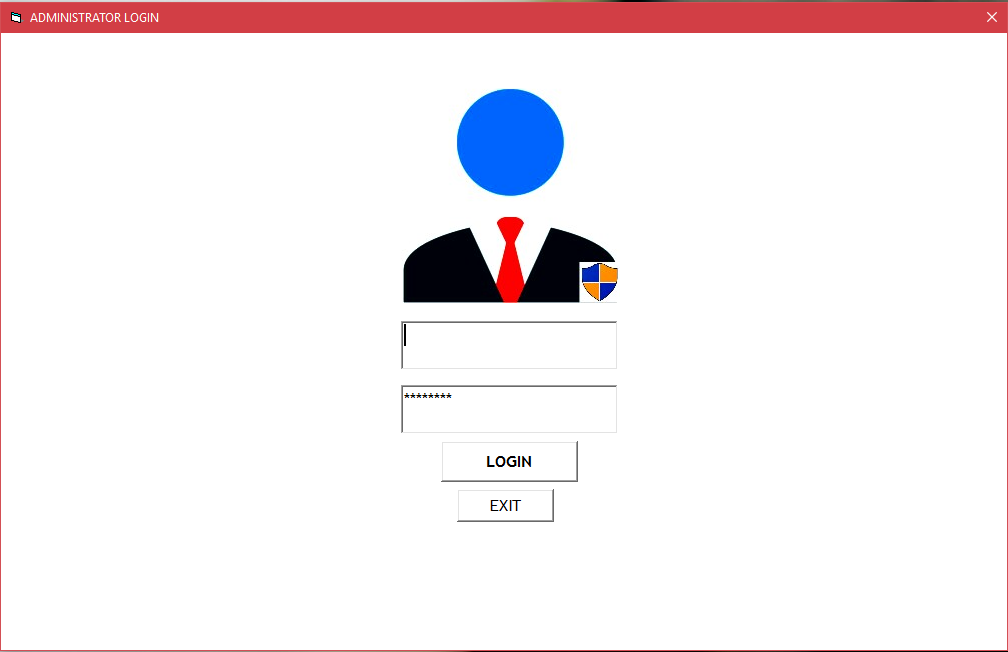
* [www.wikipedia.com](http://www.wikipedia.com)
* [www.visual-basic-6.com](http://www.visual-basic-6.com)
* [www.sqlmag.com](http://www.sqlmag.com)
* [www.techonthenet.com](http://www.techonthenet.com)

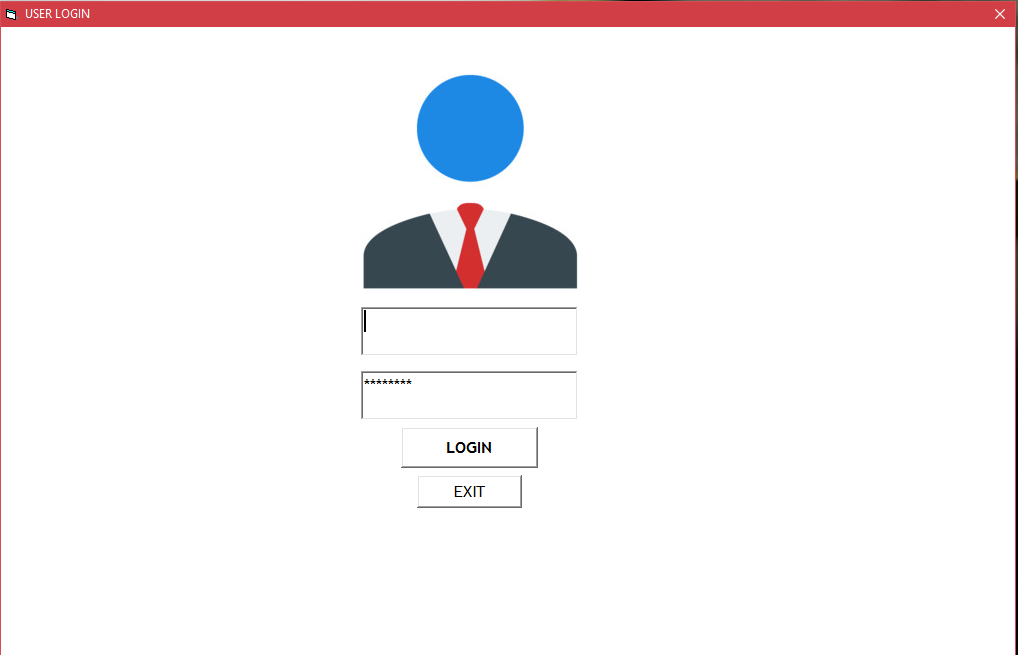
**SAMPLE SCREENSHOTS**

**9.**1 **SAMPLE SCREEN SHOTS**

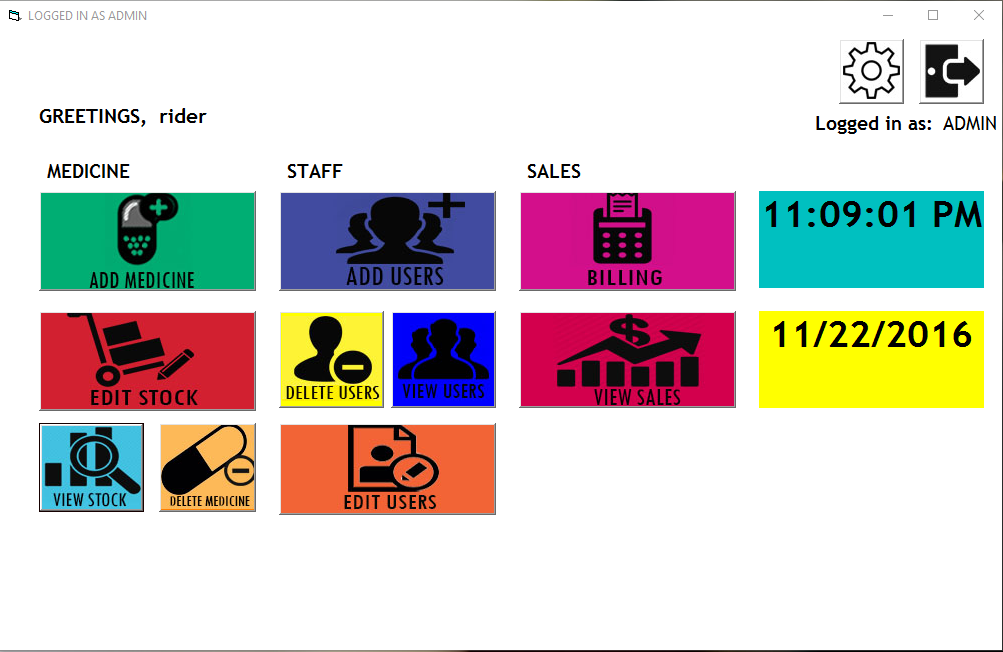
**START PAGE:-**

Admin login:-

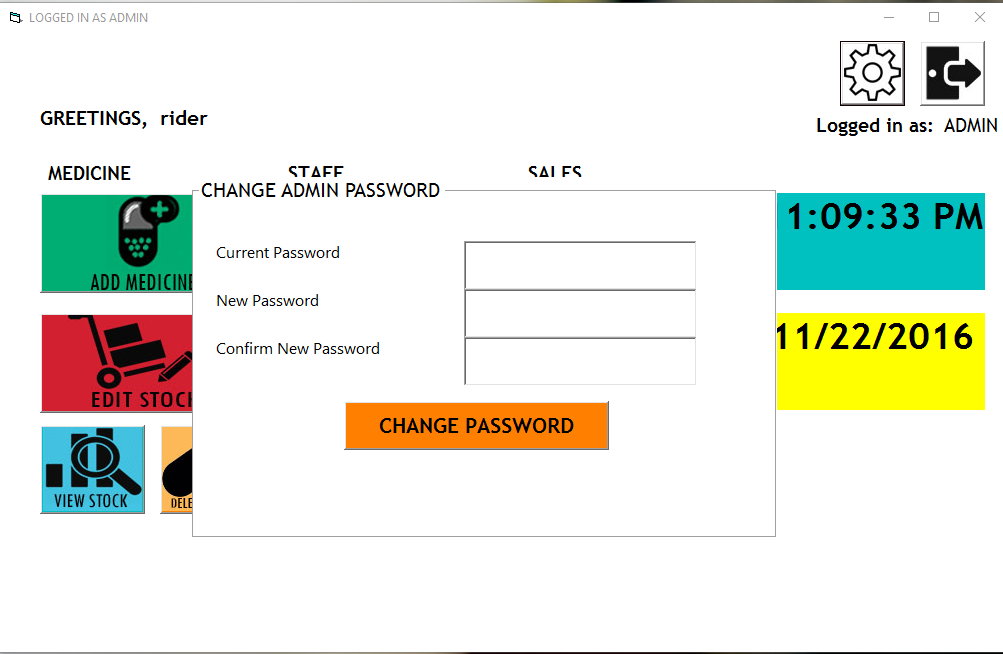


User login:-

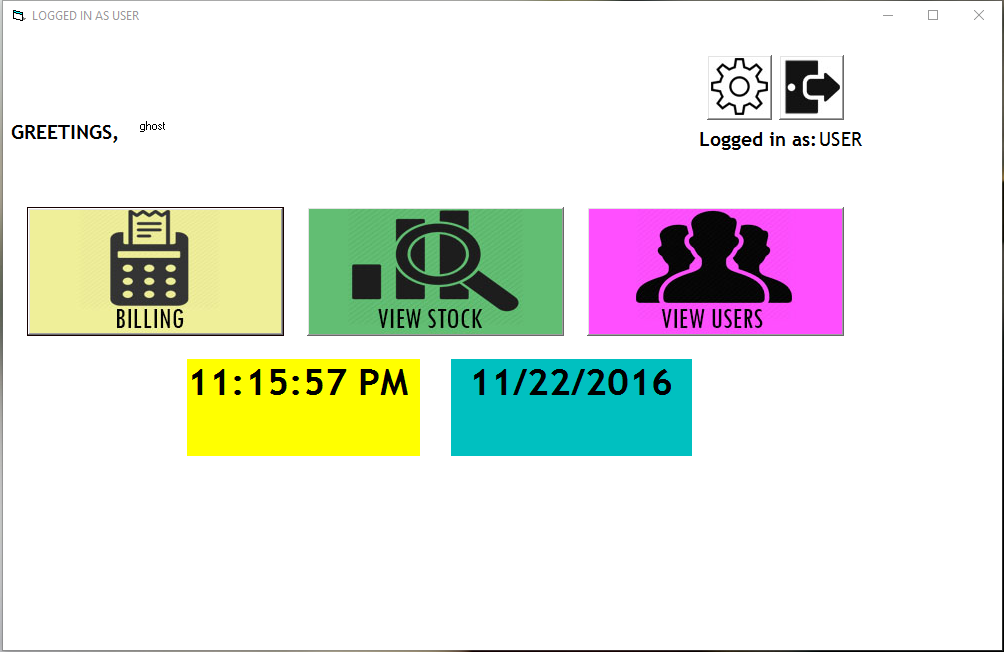
Admin Form:-



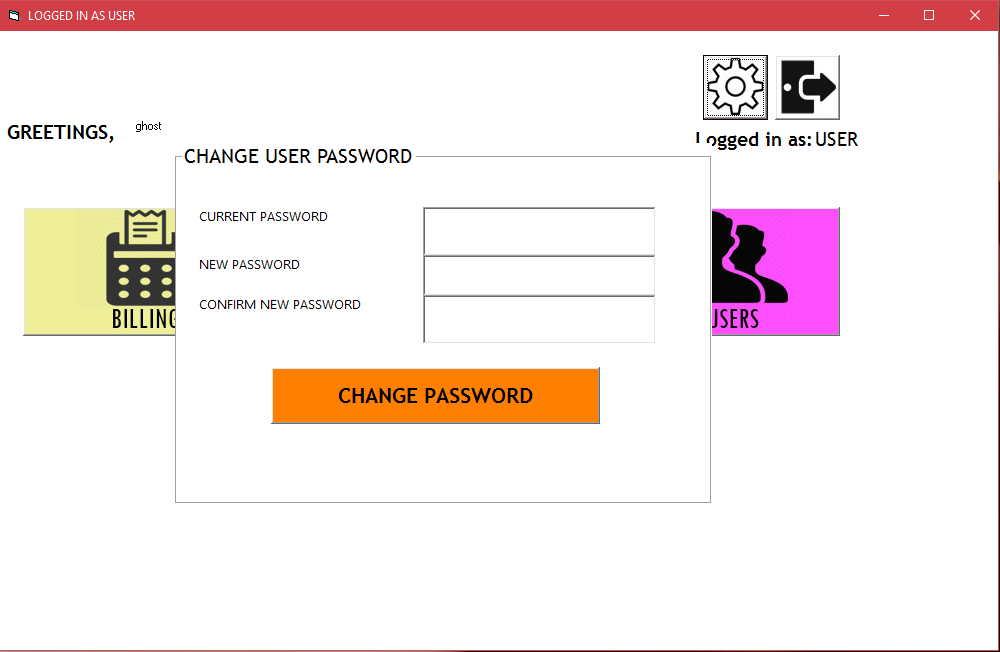
Change password for Admin:-



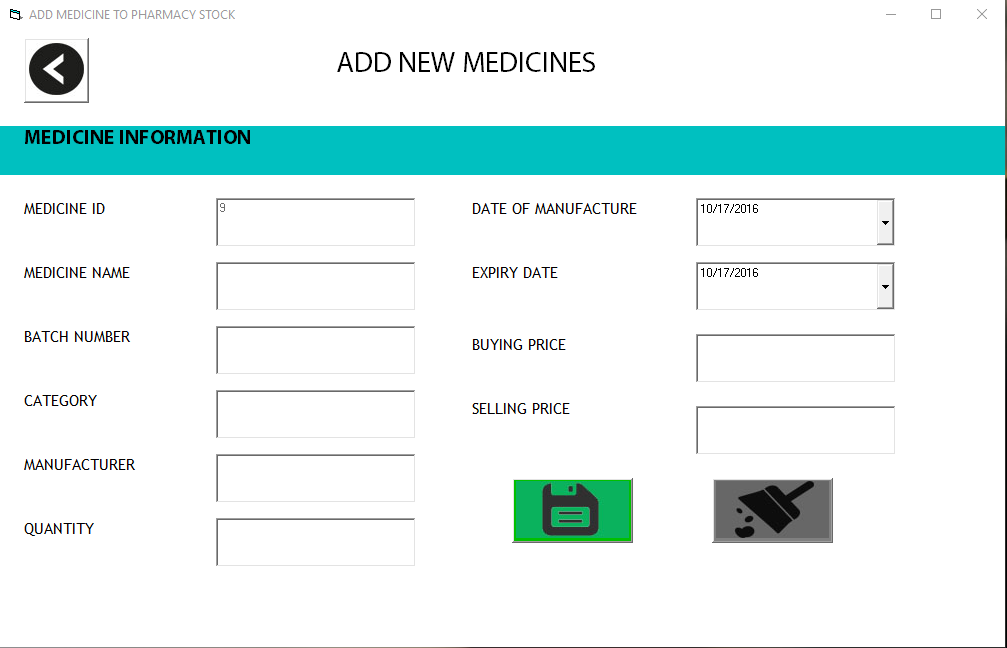
User form:-



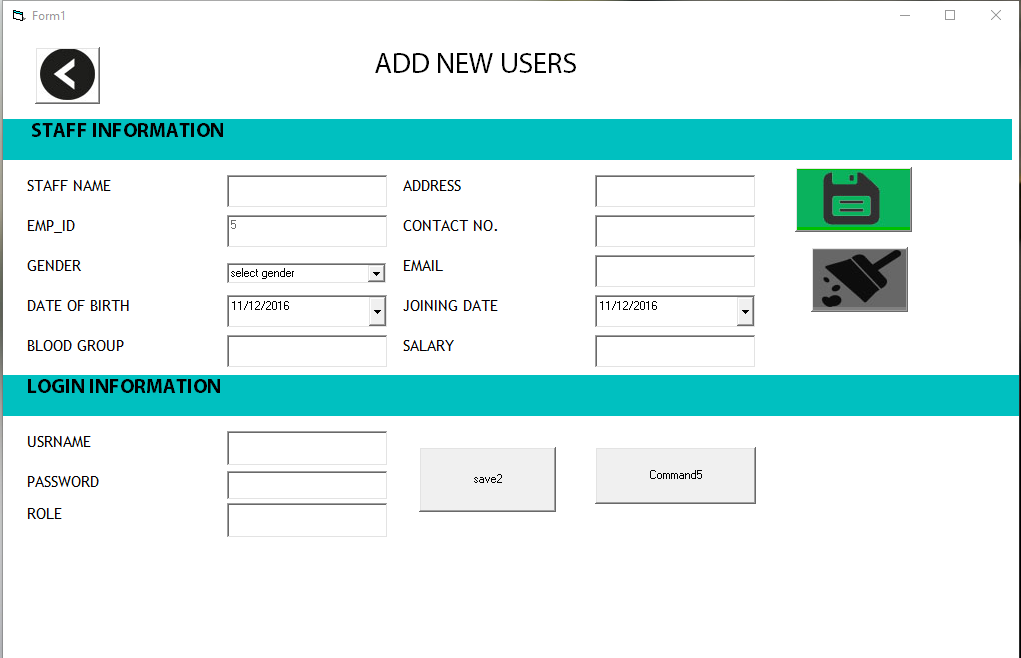
Change password for user:-



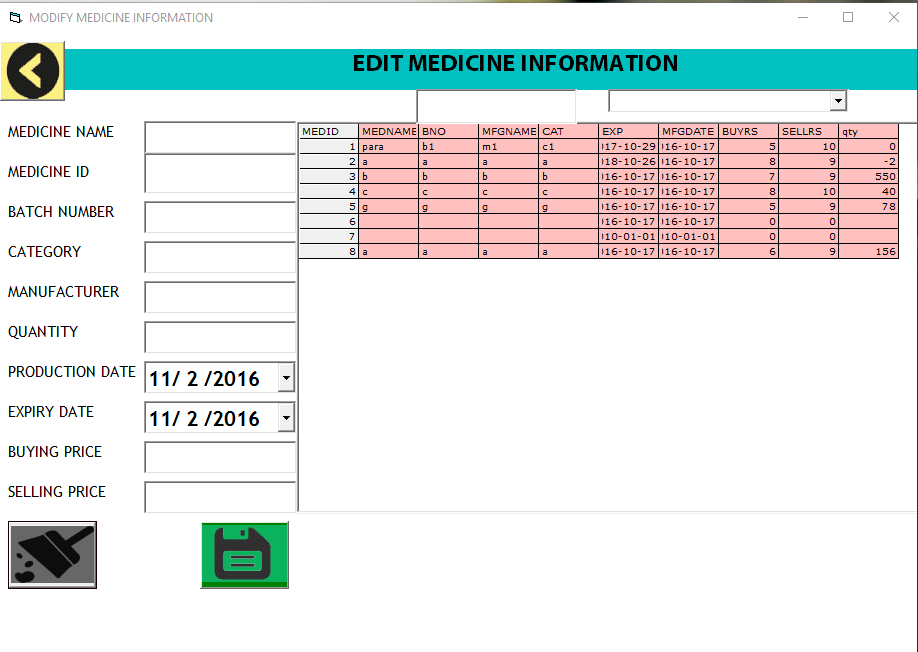
Add medicines:-



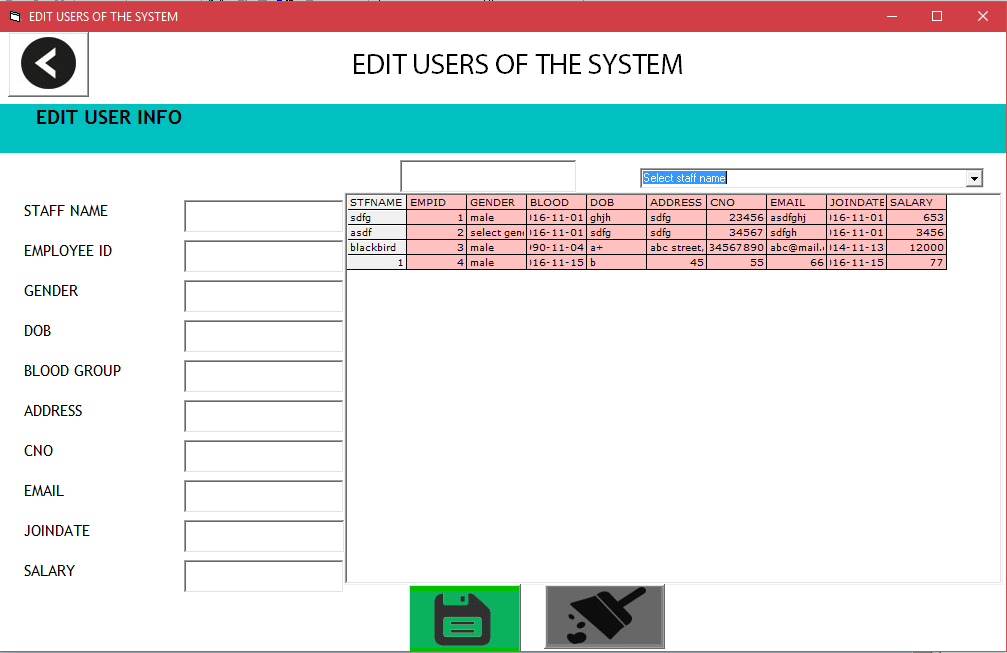
Add new users:-

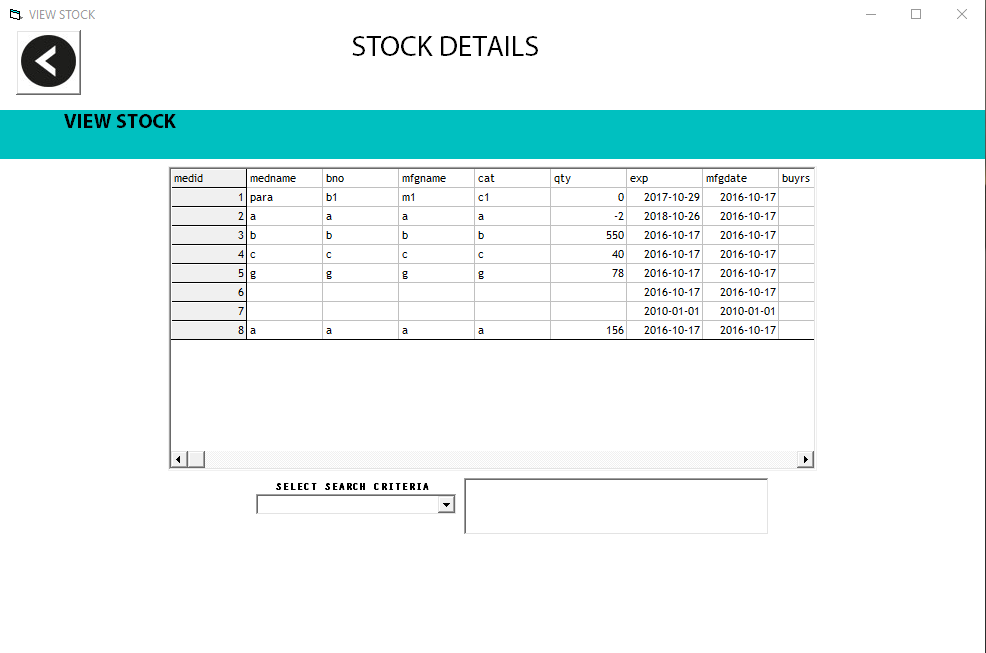


Edit stock:-

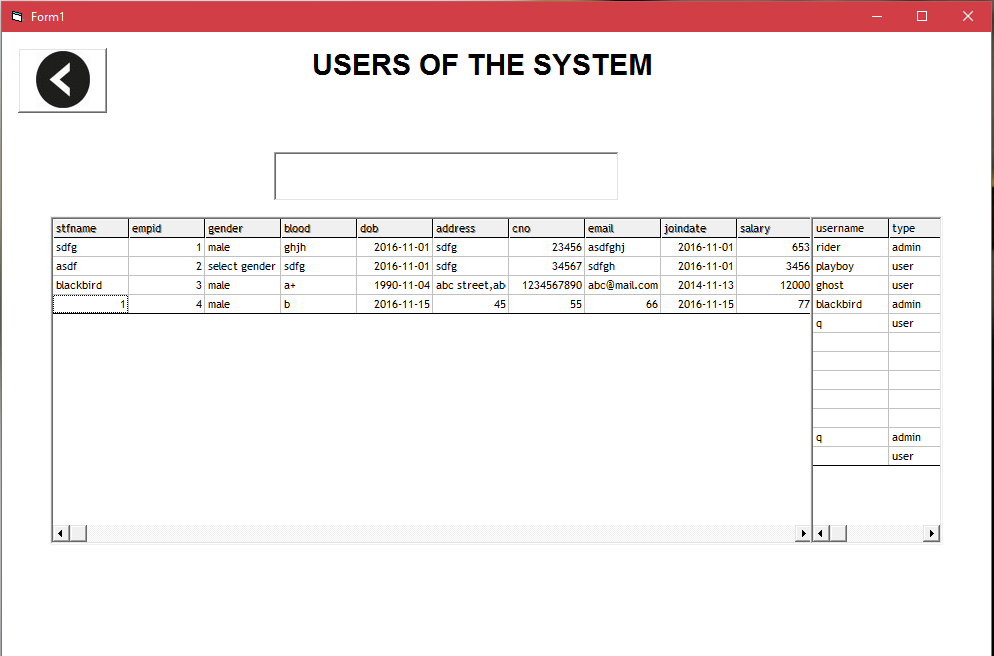


Edit users:-

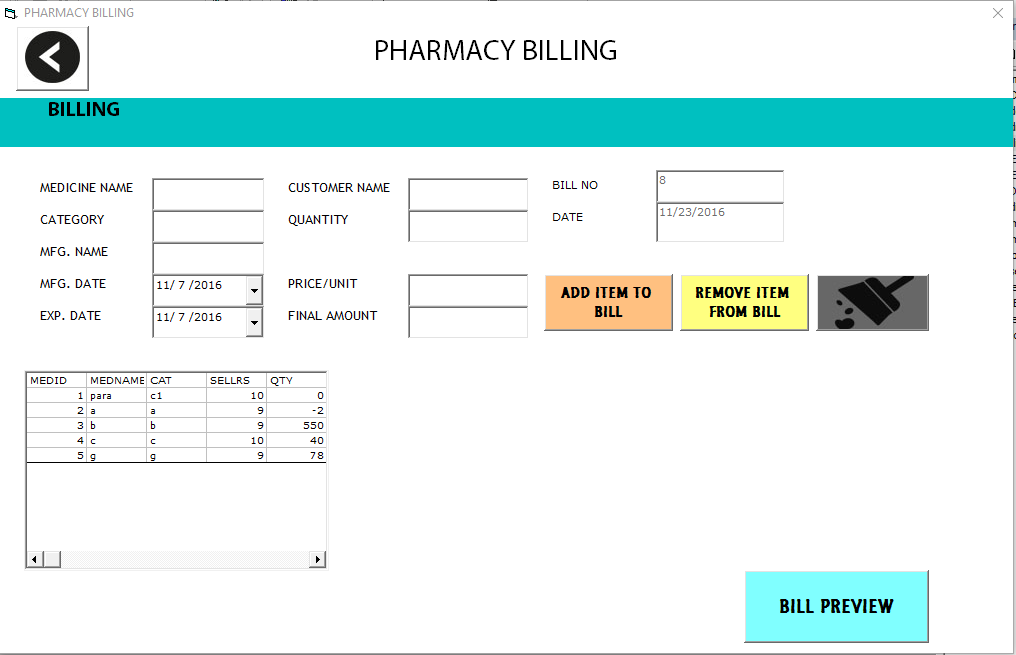


View stock:-

View users:-



Billing:-



View sales:-

