1. **INTRODUCTION OF PROJECT**

**1.1 INTRODUCTION**

The project entitled “User Maintenance” is software developed for established in computer store. This software enables you to manage user’s record. This project is developed to manage all operations of the user’s details. Module to manage this application would be User Maintenance to:-

1. Work is done using computer.
2. New record can be easily added.
3. Can view all detail of any record. (Quick Access).
4. Administration role is there for maintaining records.
5. Provide the security from the un-authentication user.
6. User friendly.
7. Easy to use.

**Various modules**

* *Admin login*: First Administrator will enable the application by putting the security key. This key provides the security from the   
  unauthorized users.

By this project we have tried to give the practical shape to the computerized “User Maintenance”. The language (Advance java) used in the creation of this project, proved to be very useful and efficient in event handling. It has buttons and Text fields, which make it more dominating over other packages and languages. Through this project we have tried our best to include all the possible aspects of “User Maintenance” to overcome the existing manual system as regards its speed, efficiency and accuracy.

“User Maintenance” For Computer Store is a software application that maintains the records related of user’s information of computer store.

* 1. **EXSISTING SYSTEM**

The transactions related to add, update and returns are maintained manually at present along with maintaining the accounts of the user details. Problems with manual work are following:-

* All work was done with help of paper.
* Time consuming procedure.
* All process done manually.
* It should not provide quick access to the records maintained.
* Edition and deletion of a record was difficult.
* Data cannot accurate and reliable. So their security is also less.
  1. **OBJECTIVE**

The main objective of the application is to automate the existing system, maintain the records of the user details.

Main purpose of the proposed “**User Maintenance**” to overcome the drawback of the existing system, we have planned to make the whole operation of User maintenance as computerized. The proposed computerized system will help in reducing the establish cost workload and also should be beneficial in speedily search.

With the help of new system, it provides the facilities for modification, deletion, searching and faster retrieve of the records.

* 1. **SCOPE**

This application can be used by any computer store to automate the process of manually maintaining the records related to the subject of maintaining the stock, the records of the user details.

User Maintenance is designed for get information about the user’s detail on the system. The system handles all the operations like Add Record, update record etc. The existing system is weak whereas this system can make it very easily.

**1.5 Developer’s Responsibilities**

To design and develop the proposed system to solve the problem with the existing system, the developer is responsibility for following events:-

* Evaluates the project for feasibility.
* Analyses current system for problems and opportunities.
* Defines requirements for improving the system.
* Defines system interface, flow and procedure.
* Performs interviews and data gathering.
* Design database structure.
* Design to interface of computer system.
* Developing the system.
* Installing the software on the clients system.
* Maintaining the system after installing.

**2. FEASIBILITY STUDY**

In feasibility study we analyze our proposed solution for being feasible or not. Under this we take into consideration three types of feasibility studies.

* Operational Feasibility
* Technical Feasibility
* Economic Feasibility

The term “feasibility study” is used as a convenient description for the output for the work done; users of this toolkit should not apply preconceived notions of what a feasibility study consists of. Stated as simply as possible, the work done here must show that the project:

* is in accordance with predetermined needs;
* is the most suitable technical solution to the needs;
* can be implemented within any capacity constraints of the Institution which operates;

**2.1. Technical Feasibility**

Our project “user maintenance” is technical feasible, because in this it is technically feasible to Add record, Add user Details, New Record. If we want to add new Record, we simply open the Add window and add it to the database.

**2.2. Operational Feasibility**

The project is also operationally feasible, because in this project we use the GUI (Graphical User Interface) which is easy to use and understand. It also provides the user friendly interface. The user will easily use the system. In this we use the buttons, textbox, images which is easily understandable for end user.

**2.3. Economic feasibility**

“User Maintenance” is economically feasible. In this project we use the java language and the for data storage we use the MS SQL. It is economically feasible because java is a platform independent so every system has their JVM.

**3. SOFTWARE REQUIREMENT ANALYSIS**

The analysis part of designing of the software focuses on the requirement analysis of the software. It is based on studying of what is required from the software, the nature of program to be built, behavior, performance and interface. It is based on studying the existing system.

|  |  |  |  |
| --- | --- | --- | --- |
| **HARDWARE REQUIREMENTS** | | | |
| PROCESSOR | RAM | | DISK SPACE |
| Pentium IV or above version | 512 MB OR Higher | | 20 MB or above |
| **SOFTWARE REQUIREMENTS** | | | |
| OPERATING SYSTEM | | DATABASE | |
| Win-98, Win-XP or above version, window XP, window 7, | | My SQL 5.1 | |

Through graphical user interface it is easy to use and any user can understand it easily. GUI provides user friendly environment.

### **INTERFACE**

The user maintenance will use the standard input/output devices for a personal computer. This includes the following:

* Keyboard
* Mouse
* Monitor

### 

### **USER INTERFACE**

Table 1: Inventory Control User Interface Screens

|  |  |
| --- | --- |
| SCREEN NAME | DESCRIPTION |
| Login | Log into the system as an Admin. |
| Staring page | Starting page include all the structure. |
| User account detail | Admin add detail of user related to account. |

**4. DESIGN**

Once the project is properly defined, the Design step begins. This is the prototype phase of the project. Prototype concepts are basic design ideas; the first look at your project's possibilities. This phase diverts focus from the problem domain to the solution domain. It acts as a bridge between the requirement phase and its solution. The design phase focuses on the detailed implementation of the system recommended in the feasibility study.

Systems design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development. There is some overlap with the disciplines of systems analysis, systems architecture and systems engineering.

**4.1. The External Design**

External design consists of conceiving, planning out and specifying the externally observable characteristics of the software product. These characteristics include user displays or user interface forms and the report formats, external data sources and the functional characteristics, performance requirements etc. External design begins during the analysis phase and continues into the design phase.

**4.2. Physical design**

The physical design relates to the actual input and output processes of the system. This is laid down in terms of how data is input into a system, how it is verified/ authenticated, how it is processed, and how it is displayed as output. Physical design, in this context, does not refer to the tangible physical design of an information system. To use an analogy, a personal computer's physical design involves input via a keyboard, processing within the CPU, and output via a monitor etc. It would not concern the actual layout of the tangible hardware, which for a PC would be a monitor, CPU, motherboard, hard drive, modems, video/graphics cards, USB slots, etc.

**4.3. Logical design**

The logical design of a system pertains to an abstract representation of the data flows, inputs and outputs of the system. This is often conducted via modeling, which involves a simplistic (and sometimes graphical) representation of an actual system. In the context of systems design, modeling can undertake the following forms, including:

* Data flow diagrams
* Entity Relationship Diagrams

**4.4. Data Flow Diagrams**

Data Flow Diagramming is a means of representing a system at any level of detail with a graphics network of symbol showing data flows, data processes, and data sources/destination.

Data flow diagram (DFD) uses a number of symbols to represent the systems.

**Terms used in DFD**

Data Flow

Administrator

Entity

Database

Frame

Process

DFD (Login)

Admin

**login**

**Login**

**Administrator goes to Login**

**Retrieve**

**Username or Password is incorrect then show message in a label placed in Login**

**If Username and password are correct then redirect to Redirect to MainFrame**

**MainFrame**

DFD (MainFrame)

**Administrator goes to MainFrame**

**MainFrame**

Admin

**User Maintenance**

DFD (User Maintenance)

**Administrator goes**

**to MainFrame**

**MainFrame**

Admin

**User Maintenance**

**Redirect**

**Save**

**UserDetail**

**Retrieve**

**usermaster**

**4.5. ENTITY DIAGRAM**

**Terms used in DFD**

**Primary Key**

**Field**

**Table**

**!**

**ER – Model (login table)**

password

username

Login

**In Login frame , a Administrator fill the username and password to go to main frame if they authenticate then go to next frame other wise generate a error message in a label placed in Login frame .**

**ER – Model (usermaster table)**

Password

Username

**!**

ID

email

usermaster

Security\_Answer

Status



**In this table , a administrator enters the record of users in a desire fields like UserId, Username, Password, User Type ,User Status, Security question etc and also they will change update and retrieve a record in user master panel.**

**5. CODING**

The design is complete; most of the major decisions about the system have been made. **After designing the new system, the whole system is required to be converted into computer understanding language.** The goal of the coding phase is to translate the design of the system into code in a given programming language for a given design. The aim of this phase is to implement the design in the best possible manner. The coding phase affects both the testing &maintenance profoundly. A well written code reduce the testing & maintenance effort .since the testing &maintenance cost of software are much higher than coding cost .During the coding focus should be on developing programs that are easy to write

The design must be translated into machine readable form. The code generation step performs this task if the design is performed in a detailed manner, code generation can be accomplished without Much complication, programming language are used for coding & the right programming language is chosen

This is also called programming phase in which the programmer convert the program specification into computer instruction which be refer as programs. The program coordinates the data movements & controls the entire process in a system.

It is generally felt that the program must be modular in nature. This helps in fast development, maintenance & future change, if required.

**LANGUAGE**

**Java language is used for coding in this project.**

**6. TESTING**

Before actually implementing the new system into operation, a test run of the system is done removing all bugs, if any. It is an important phase of a successful system. After codifying the whole program of the system, a test plan should be developed on run on a given set of test data. The output of the test run should match the expected result.

Testing is a process of executing a program with the intent of finding an error. A good test case is on that has high probability of finding an as yet undiscovered error. No program or system is perfect, communication between user and designer is not always clear. The result is error and error. So testing performs a main role in developing software.

In this phase, the system is tested normally programs are written as series of individual modules, these subjects separated& detailed test .The system is then tested as whole .The separated modules are brought together & tested as complete system. The system is tested to ensure that interface between modules work (integration testing) the system work on the intended platform with the expected volume of the data (volume testing) & that the system does what the user require (beta testing). System testing involves testing hardware devices & debugging computer programs, testing information processing procedures.

**The people who are responsible for testing a software application:**

1. Software testing can be conducted by the developers of the system or an independent testing group who are part of the organization that has built the system.
2. Software testing can also be conducted by the client or the ultimate users of the system.
3. The team responsible for the different types of testing needs to be decided upon during the planning stage.
   1. **The various stages in testing**

Software testing is usually performed at different levels of abstraction of the application along the software development process by the builders of the system.

**There are five testing stages**

**6.1.1. Unit testing.**

**6.1.2. Integration testing.**

**6.1.3. System testing.**

**6.1.4. Acceptance testing.**

**6.1.5. Regression testing.**

The objective and the abstraction levels of the application to which these tests are performed are different.

* Unit tests are performed on the smallest individual units of the application. Unit testing uses code and detailed design as an input to check correctness of individual units.
* The integration tests on a group of modules and their interfaces. Integration testing uses the system design and the functional specification document as an input.
* The system tests are for the entire system and the interfacing external systems. System testing uses the overall functionality of the system as given in the functional specifications and software requirements. It also evaluates the non- functional requirements.
* Acceptance testing is the test conducted periodically by client representatives to check if client requirements have been met adequately.
* Regression testing, on the other hand, retests the tested sections of the software to ensure no unintended error has been introduced.

Here is another very important concept of software testing, that is, the test case. Test cases are scenarios that are executed by the testers on the completed application to determine if the application meets a specific requirement. One or more test cases may be required to determine if a requirement is satisfied.

A good test case is one that uncovers errors in a complete manner with minimum time and effort. Considering the earlier example of the completed house, the analysis, ‘check if the color of chimney is red’ is a test case. If for the same example, when the test case is written as ‘check if the door does not open with a wrong key’, becomes a negative test case. Hence, we learn that a test case is a statement specifying an input, an action, or an event and expects a specific response after execution.

**6.2. TEST CASES**

**Login**

|  |  |  |  |
| --- | --- | --- | --- |
| **Case No.** | **Security Key** | **Valid/Invalid** | **Response** |
| 1 | admin | Valid | Authentic |
| 2 |  | Invalid, because Security Key field is empty | Nothing |
| 3 | Manpreet | Invalid, Because Security Key is incorrect | Show message  “Not Recognised” |

1. **IMPLEMENTATION AND POST IMPLEMENTATION**

A crucial phase in the system development life cycle is the successful implementation of the new designed system implementation. It includes all those activities that take place to convert from the old system to the new system. The new system is here replacing an existing manual system. The proper implementation become necessary so that a reliable system based on the requirement of the organization can be provided. Successful implementation guarantees improvement in the organization working.

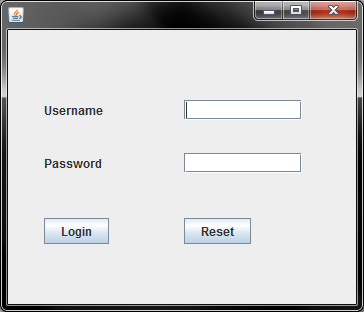
Finally the actual implementation requires launching of the web site and fulfills the required formalities with the concerned authority and feeding of all the information required in the database. Thus, implementation is vital step in ensuring the success of new system. Even well-designed system can fail if it is not properly implemented.

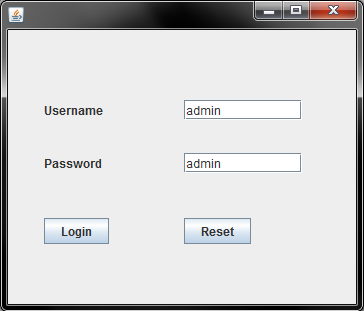
**Implementation Activities**

* Acquisition of hardware, software &services
* Software development or modification
* End user training
* System documentation & installing

1. **SNAPSHOTS**

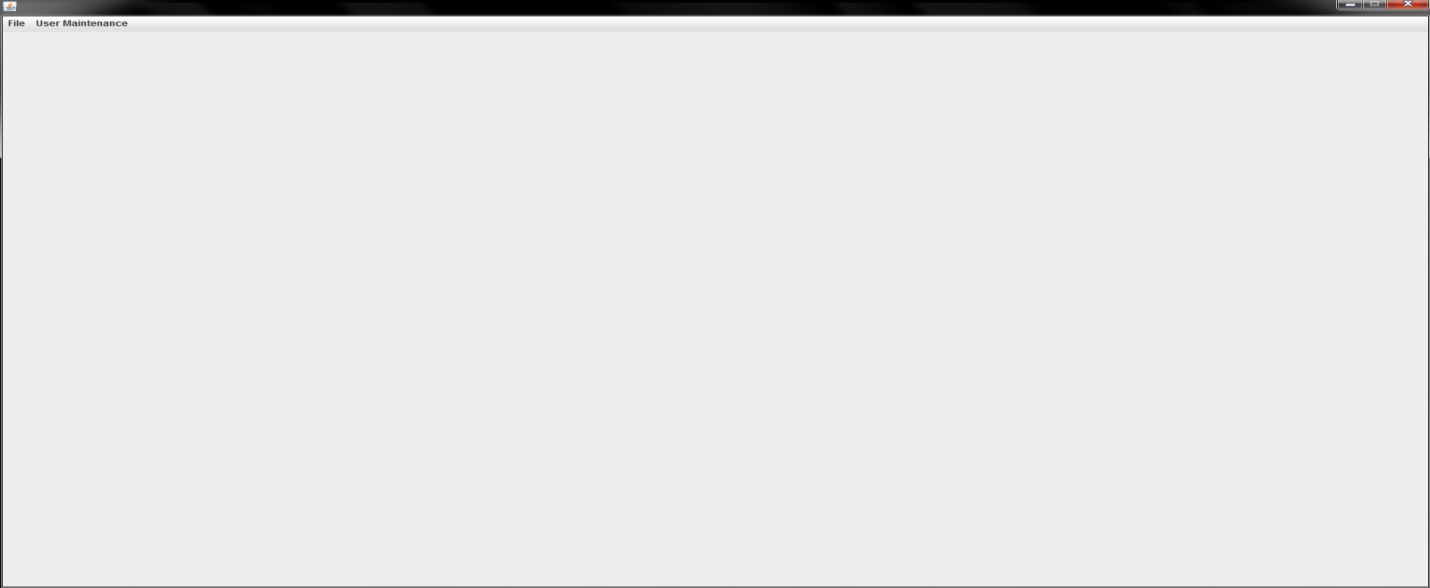
Login Frame





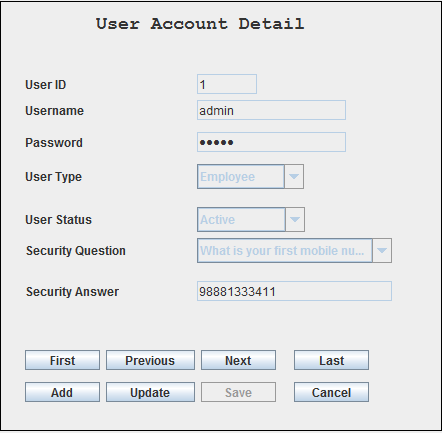
**Administrator fills username and password for login.**

**User Maintenance frame**

****

**Main Frame is used for to represent a menu and interface between panel and administrator.**

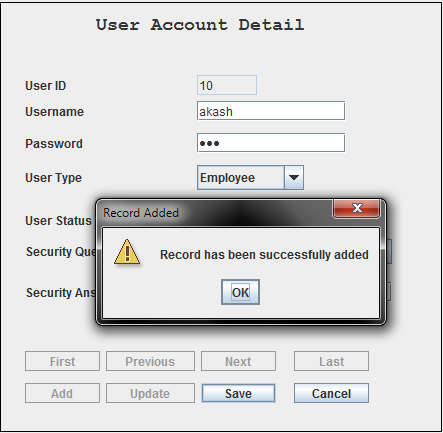
**User account detail**

****

**User account detail panel is used for to add, retries, update user record in any time.**

**User account detail (add button)**

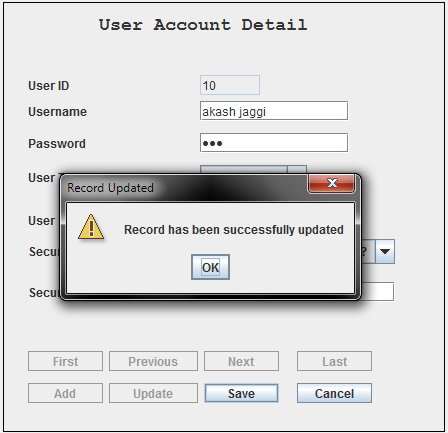
****

****

**User account detail panel (add button) is used for to add new user record in any time**

**User account detail (Update Button)**





**User account detail panel (update button) is used for to update any user record in any time.**