**CHAPTER-1**

**INTRODUCTION**

1. **INTRODUCTION**

It is an online medical lab reporting system is designed to automate the integral data of the laboratory. Here patients get an option to register on the site and login using registered details. Once registered with their address and contact details, the patients may now see a variety of tests. The system allows patients to do the bookings for various tests like CBC, Blood Glucose, thyroid and cultures. The system allows users to book any test needed.

The main objective is to provide the technician a pleasant and efficient way of working and generating reports. It can be used in any Laboratory for maintaining Patients details and their test results. This project is used for New Patient record Entry, Patient ID and Report Generation according to requirements.

**1.1 PROJECT OVERVIEW**

It is an online medical lab Medical Laboratory Technology presents the development in the medical laboratory science. In many countries, there are two main types of labs that process the majority of medical specimens. Hospital laboratories are attached to a hospital, and perform tests on patients. Private (or community) laboratories receive samples from general practitioners, clinical research sites and other health clinics for analysis. The online medical lab reporting system is built for the purpose of effective and clear data saving and manipulating. The online medical lab reporting system project highly minimize time and resource by which, searching the reports and data you can get the data in quickest time. And almost the resources are wise used since most actions are done on the medical laboratory management system. Some of the resources minimize the manpower and paper work. The feature use to storing data in a secure way.The application can be used in any Hospital that has medical/laboratory within the hospital for maintaining patients’ tests and test report detail by customizing some of the features.

The online medical lab reporting system is designed to automate the integral data of the laboratory. Here patients get an option to register on the site and login using registered details. Once registered with their address and contact details, the patients may now see a variety of tests. The system allows patients to do the bookings for various tests like CBC, Blood Glucose, thyroid and cultures. The system allows users to book any test needed.

The main objective is to provide the technician a pleasant and efficient way of working and generating reports. It can be used in any Laboratory for maintaining Patients details and their test results. This project is used for New Patient record Entry, Patient ID and Report Generation according to requirements.

**1.2** **PROJECT SPECIFICATION**

The system after careful analysis has been identified to be presented with the 3 modules. They are:

**1.Admin module**

Admin login to the page, view the details of registered users and technicians, add tests , maintain booking details of users, Admin view the users booking details, Admin keep the result, Admin approve the booking details from the users, logout.

**2. User module**

User login to the page and register his/her profile, test selection option, User can view his/her result, View the profile, User can view his/her booking details and user can enter their feedback, logout.

**3. Lab Technician module**

Technician login to the page and register his/her profile, view the booking details from users, publish the result, logout.

**CHAPTER-2**

**SYSTEM STUDY**

**2.1 INTRODUCTION**

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minute’s detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is loop that ends as soon as the user is satisfied with proposal.

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies, a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

**2.2 EXISTING SYSTEM**

The existing system is all type of service entries are registered to directly by people. And all records are keeping manually. So it’s very difficult and time consuming. Since all these services are carried out by the normal people they are suffered from long distance journey. Client side validations are performed to check entry of invalid fields or incorrect format. The user id and password are checked thoroughly for existing users to ensure security. Stored procedures and classes are used to increase the speed of processing.

**2.3 DRAWBACKS OF EXISTING SYSTEM**

**The limitations of current system are mentioned below:**

* Require more space to keep the paper documents.
* More time consuming.

**2.4 PROPOSED SYSTEM**

This system provides good and efficient information and thus making the service smarter. It is an interface for customers to browse the catalogue and buy the site online. It provides an end to end solution to manage. By following this new approach, the information can be accessed from anywhere just with a mouse click. This helps the users by saving lot of time and providing the user with up to date information. Some of the Features,

* High Speed response for users.
* Easy to handle and flexible.
* Fast and convenient
* High accuracy and saving.
* Fully online service

**2.5 ADVANTAGES OF PROPOSED SYSTEM**

The system is very simple in design and to implement. The system requires very low system resources and the system will work in almost all configurations. It has got following features:

####  Better security: -

For data to remain secure measures must be taken to prevent unauthorized access. Security means that data are protected from various forms of destruction. The system security problem can be divided into four related issues: security, integrity, privacy and confidentiality. Username and password requirement to sign in ensures security. It will also provide data security as we are using the secured databases for maintaining the documents.

 **Accuracy: -**

The level of accuracy in the proposed system will be higher. All operation would be done correctly and it ensures that whatever information is coming from the center is accurate

####  User friendliness and interactive: -

The proposed system will help the user to reduce the workload and provides user friendly environment so that they can easily do their jobs..

####  Minimum time required: -

The data are management is in such a way that a particular registered user can buy site very easily.

**CHAPTER 3**

**REQUIREMENT ANALYSIS**

**3.1 FEASIBILITY STUDY**

The feasibility study is concerned with the consideration made to verify whether the system fit to be developed in all terms .Once the idea to develop software is put forward, the question that rises first will pertain to be the feasibility aspects. A feasibility study is conducted to select the best system that meets the system performance requirements. This entitles an identification description, an evaluation of candidate systems and the selection of the best system for the job. It also helps in identifying the risk factors involved in developing and deploying the system. So a feasibility study is a report which could be used by the senior or top persons in the organization. This is because based on the report the organization decides about cost estimation, funding and other important decisions which is very essential for an organization to run profitably and for the system to run stable. There are different types in feasibility study. They are,

**3.1.1 Economical Feasibility**

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

* The costs conduct a full system investigation.
* The cost of the hardware and software.
* The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

At your service will be a simple platform for users to access services for their huge needs. It is completely free. Using this system large number people can solve their problems with free of cost.

**3.1.2 Technical Feasibility**

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input, output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed.

Technical issues raised during the investigation are:

* Does the existing technology sufficient for the suggested one?
* Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project requires High Resolution Scanning device and utilizes Cryptographic techniques. Through the technology may become obsolete after some period of time, due to the fact that newer version of same software supports older versions, the system may still be used. So there are minimal constraints involved with this project. The system has been developed using php in front end and MySql in server in back end, the project is technically feasible for development.

* + 1. **Behavioral Feasibility**
  + Is there sufficient support for the users?
  + Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

**3.2SYSTEM SPECIFICATION**

**3.2.1 HARDWARE SPECIFICATION**

The selection of hardware configuration is a very important task related to software development. Particularly insufficient RAM may affect adversely on the speed and correspondingly on the efficiency of the entire system. The processor should be powerful to handle all the operations. Hard disk should have sufficient capacity to store database and application. The network should be well sufficient to handle the communication fast.

Minimum hardware specification required to run this application:

|  |  |  |
| --- | --- | --- |
| Processor | : Dual core or compatible | |
| Speed | : 800 | MHz or above |
| Hard disk | : 180 | GB |
| RAM | : 512 MB | |
| Monitor | : Compatible monitor with 600 x 800 resolutions | |
| Internet | : 56 kbps/above | |

**3.2.2 SOFTWARE SPECIFICATION**

After the collection of all the required information regarding the software to be developed and has removed all incompleteness, inconsistencies and anomalies from the specification and then starts to systematically organize the requirements in the specification. Software’s required to run this application:

Operating System Front End

Back End

: Windows 7

: PHP

: MYSQL

**3.3 SOFTWARE DESCRIPTION**

**3.3.1 PHP**

PHP, also known as the Hypertext Preprocessor, is a widely used computer programming language. It is used to create dynamic Web pages, or Web pages that update and display information depending on the user's activity. PHP is mainly a server side language, allowing it to process Web pages faster and easier, providing a better user experience.

Rasmus Ledorf, a Danish computer programmer, introduced the first version of PHP in 1995. He called the initial set of codes the "Personal Home Page Tools," which he used as a code to track visitors looking at his online resume. After some time, he added more functionality to the script and released the source code to everyone. Ledorf continued improving and developing PHP by himself until 1997.

During that same year, two more programmers, Andi Gutmans and Zeev Zuraski, coordinated with Ledorf to create PHP 3.0. This version was the first to resemble the current popular PHP most programmers use today. It has more functionality and capabilities and was officially called "PHP: Hypertext Preprocessor." The latest PHP is version 5, with several release updates. It still resembles the functionality of version 3.0, but with added features. It mainly runs through the "Zend Engine 2.0."

There are four major advantages when using PHP: accessibility, compatibility, simplicity and extensive community support. Because PHP is open source, access has no restrictions. Programmers interested in using PHP only need to download the scripts, without paying a single cent. It runs in all types of Operating Systems and is also compatible with the most popular Web servers, including Apache and IIS. Programmers also don't have to spend a lot of time learning PHP because its language is related to Perl and C, two of the most popular coding languages. Lastly, PHP has a large online community support. Since it's free, a lot of other programmers share scripts and give advice on how to improve the system. As of May 2011, there are more than 25 million websites that run on PHP. Most webmasters prefer to use the language because PHP was created to work on Web development. PHP can also be integrated to HTML codes, allowing easier coding and flexibility when building websites. Although using PHP is simple, experienced programmers are usually required when building

complex and large websites. PHP has a lot of extensions and other add-ons, and an inexperienced programmer may get confused sorting through and checking all of the available scripts. Also, codes can sometimes become complex when programmers want to achieve a very specific function.

Lastly, because PHP is mainly a language designed for Web development, building large websites requires programmers to know HTML, JavaScript and CSS.

**3.3.2 MYSQL**

MySQL officially, but also called My Sequel is (as of July 2013) the world's second most widely used [relational database management system](http://en.wikipedia.org/wiki/Relational_database_management_system) (RDBMS) and most widely used open-source RDBMS. It is named after co-founder [Michael Widenius's](http://en.wikipedia.org/wiki/Michael_Widenius) daughter, MY. The [SQL](http://en.wikipedia.org/wiki/SQL) acronym stands for [Structured Query Language.](http://en.wikipedia.org/wiki/Structured_Query_Language)

The MySQL development project has made its [source code](http://en.wikipedia.org/wiki/Source_code) available under the terms of the [GNU General Public License,](http://en.wikipedia.org/wiki/GNU_General_Public_License) as well as under a variety of [proprietary](http://en.wikipedia.org/wiki/Proprietary_software) agreements. MySQL was owned and sponsored by a single [for-profit](http://en.wikipedia.org/wiki/Business) firm, the Swedish company [MySQL AB,](http://en.wikipedia.org/wiki/MySQL_AB) now owned by [Oracle Corporation.](http://en.wikipedia.org/wiki/Oracle_Corporation)

MySQL is a popular choice of database for use in web applications, and is a central component of the widely used [LAMP](http://en.wikipedia.org/wiki/LAMP_(software_bundle)) open source web application software stack (and other ['AMP'](http://en.wikipedia.org/wiki/List_of_AMP_packages) stacks). LAMP is an acronym for ["Linux,](http://en.wikipedia.org/wiki/Linux) [Apache,](http://en.wikipedia.org/wiki/Apache_HTTP_Server) MySQL,Perl/PHP/Python." [Free-software](http://en.wikipedia.org/wiki/Free_software)-open source projects that require a full-featured database management system often use MySQL.For proprietary use, several paid editions are available, and offer additional functionality.

Applications which use MySQL databases include:

[TYPO3,](http://en.wikipedia.org/wiki/TYPO3) [MODx,](http://en.wikipedia.org/wiki/MODx) [Joomla,](http://en.wikipedia.org/wiki/Joomla) [WordPress,](http://en.wikipedia.org/wiki/WordPress) [phpBB,](http://en.wikipedia.org/wiki/PhpBB) [MyBB,](http://en.wikipedia.org/wiki/MyBB) [Drupal](http://en.wikipedia.org/wiki/Drupal) and other software. MySQL is also used in many high-profile, large-scale [websites,](http://en.wikipedia.org/wiki/Website) including [Google](http://en.wikipedia.org/wiki/Google) (though not for searches), [Facebook,](http://en.wikipedia.org/wiki/Facebook) [Twitter,](http://en.wikipedia.org/wiki/Twitter) [Flickr,](http://en.wikipedia.org/wiki/Flickr) and [YouTube.](http://en.wikipedia.org/wiki/YouTube)

MySQL is a [relational database management system](http://en.wikipedia.org/wiki/Relational_database_management_system) (RDBMS), and ships with no [GUI](http://en.wikipedia.org/wiki/Graphical_user_interface) tools to administer MySQL databases or manage data contained within the databases. Users may use the included [command line](http://en.wikipedia.org/wiki/Command_line) tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back up data,

inspect status, and work with data records. The official set of MySQL front-end tools, [MySQL](http://en.wikipedia.org/wiki/MySQL_Workbench) [Workbench](http://en.wikipedia.org/wiki/MySQL_Workbench) is actively developed by Oracle, and is freely available for use.

* **The MySQL Database Server is very fast, reliable, scalable, and easy to use.**

If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available. MySQL can also scale up to clusters of machines, networked together.

MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

* **MySQL Server works in client/server or embedded systems*.***

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different back ends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs).

We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

**CHAPTER 4**

**SYSTEM DESIGN**

**4.1 INTRODUCTION**

Software design is a process of problem-solving and planning for a software solution. After the purpose and specifications of software is determined, a plan for the solution is developed. If the software is ‘semi-automated’ or user-centered, software design may involve user experience design yielding a story board to help to determine those specifications. If the software is completely automated, a software design may be as simple as a flow chart or text describing a planned sequence of events. In either case, some documentation of the plan is usually the product of the design. A software design may be platform-independent or platform-specific, depending on the availability of the technology called for by the design. There are fundamentally two different approaches to software design. They are Function Oriented Design and Object Oriented Design. Each of these two techniques may be made applicable at different stages of the design process.

**4.2 MODULE DESIGN**

In the object oriented design approach, system is viewed as a collection of objects. The system state is decentralized among the objects and each object manages its own state information. In fact, the functions defined for one object cannot refer or change data of other objects. Objects have their own internal data which define their state. Similar objects constitute a class.

In other words, each object is a member of sub class. Classes may inherit features from a super class. Conceptually, objects communicate by message passing. Dividing the main system into sub system is the fundamental principle for good design. Decomposition of main system into modules gives more advantages in the system of proposed system. Different modules are dependent on each other by common attribute, so that each one can understand easily.

**Input Design**

We defined system design as the process of developing specification for a system that meet the criteria established in system analysis. The major step in design is the preparations of input and the design of output in a form acceptable to the user. The input design is the process of converting user oriented inputs to a computer based formats.

The goal of designing input data is to make automation as easy and free from error as possible. For providing a good input design for the application easy data input and selection features are adopted. The input design requirement such as user friendliness, consistent format and interactive dialog for giving the right message and help for user at right time are also considered for the development of this project.

Most common cause of errors in a system is invalid user input. Maximum care is taken to prevent invalid data from entering in to the system. This was achieved by making proper validation checks on the user inputs. Error messages are displayed when and where an invalid user entry /action are encountered. Online Medical Lab Reporting System, has 2 input subsystems

**1.Registration**

In Online Medical Lab Reporting System, we are providing a registration form, in which the user can give his/her personal details and create an account by giving user name and password.

**2.Login**

In our system we are provided a login for secured use of the system. The user is identified by his/her email id and password. Once the authenticated person is logged in he/she can manage the system as per their needs.

**Output Design**

Computer output is the most important and direct source of information to the user efficient output design should improve the system’s relationships with the user and help in decision making. A major form of output is hardcopy from the printer. Printout should be designed around output requirement of the user. Output design concepts:

* Output is the most visible component of the information system.
* It is the basis of management’s final assessment of the system
* It is designed by rapidly constructing prototypes
* During system design output are modeled as data flows

**Report Generation(Output)**

This module helps the user to view their results.

**4.3 DATA FLOW DIAGRAM**

Data Flow Diagram (DFD) is a diagram that describes the flow of data and the processes that change data throughout a system. It’s a structured analysis and design tool that can be used for flowcharting in place of or in association with information. Oriented and process oriented system flowcharts. When analysts prepare the Data Flow Diagram, they specify the user needs at a level of detail that virtually determines the information flow into and out of the system and the required data resources. This network is constructed by using a set of symbols that do not imply physical implementations. The Data Flow Diagram reviews the current physical system, prepares input and output specification, specifies the implementation plan etc.

The purpose of the design is to create architecture for the evolving implementation and to establish the common tactical policies that must be used by desperate elements of the system. We begin the design process as soon as we have some reasonably completed model of the behavior of the system. It is important to avoid premature designs, wherein develop designs before analysis reaches closer. It is important to avoid delayed designing where in the organization crashes while trying to complete an unachievable analysis model.

Throughout the project, the context flow diagrams, data flow diagrams and flow charts have been extensively used to achieve the successful design of the system. In our opinion, “efficient design of the data flow and context flow diagrams helps to design the system successfully without much major flaws within the scheduled time”. This is the most complicated part in a project. In the designing process, our project took more than the activities in the software life cycle. If we design a system efficiently with all the future enhancements, the project will never become junk and it will be operational.

The data flow diagrams were first developed by Larry Constantine as way for expressing system requirements in graphical form. A data flow diagram also known as “bubble chart” has

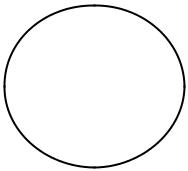
purpose of clarifying system requirements and identifying major transformations that will

become programs in system design. It functionally decomposes the requirement specification down to the lowest level. DFD depicts the information flow, the transformation flow and the transformations that are applied as data move from input to output. Data Flow Diagram is quite effective, especially when the required design is unclear and the user and analyst need a notational language for communication. It is used to model the system components such as the system process, the data used by the process, any external entities that interact with the system and information flows in the system.

Four basic symbols are used to construct data flow diagrams. They are symbols that represent data source, data flows, and data transformations and data storage. The points at which data are transformed are represented by enclosed figures, usually circles, which are called nodes.

**Main symbols used in the data flow diagram are:**

1. Circle represents a process that transforms incoming data flows in to outgoing data flows.



2. A square defines a source and destination of system data.

3.Arrow identifies data in motion.



.

4.An open rectangle defines a data store, data at rest or temporary repository of data

**Steps to Construct Data Flow Diagrams:-**

Four steps are commonly used to construct a DFD:

* Process should be named and numbered for easy reference. Each name should be representative of the process.
* The destination of flow is from top to bottom and from left to right.
* When a process is exploded in to lower level details they are numbered.
* The names of data stores, sources and destinations are written in capital letters.

**Rules for constructing a Data Flow Diagram**

* Arrows should not cross each other.
* Squares, circles and files must bear names.
* Decomposed data flow squares and circles can have same names.
* Draw all data flow around the outside of the diagram.

**CONTEXT DIAGRAM**

Context diagram is also called LEVEL-0 DFD. It represents the entire system in a single bubble. The various entities are with external which the system interacts and the data flows occurring between the systems and external entities are also represented. A system Context Diagram (SCD) is the highest level view of a system, showing the system as a whole and they should always be produced as DFDs. Context Diagrams show the interactions between a system and other factors with which the system is designed to face. SCD is very helpful in understanding the context in which the system will be part of Software engineering.

Context diagrams are used early in a project to get agreement on the scope under investigation. Context diagrams are typically included in a requirements document. These diagrams must be read by all project stakeholders and thus should be written in a plain language so the stakeholders can understand items within the document.The best System Context Diagrams are

used to display how system interoperates at a very high level or how systems operate and interact logically. The System Context Diagram is a necessary tool in developing a baseline interaction between systems and actors; actors and systems and systems.

**Context Diagram**



**User level-1**



**Technician-1**



**Admin-1**



**4.5 UML DIAGRAM**

A use case diagram is a graphic depiction of the interactions among the elements of a system. A [use case](http://searchsoftwarequality.techtarget.com/definition/use-case) is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service [Web site](http://searchsoa.techtarget.com/definition/Web-site). Use case diagrams are employed in [UML](http://searchsoftwarequality.techtarget.com/definition/Unified-Modeling-Language) (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems.

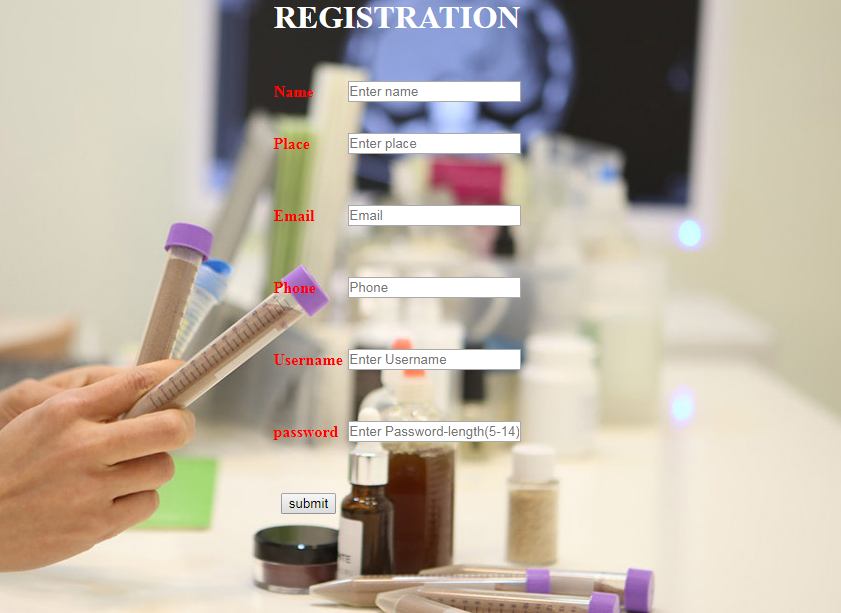
System objectives can include planning overall requirements, validating a [hardware](http://searchcio-midmarket.techtarget.com/definition/hardware) design, testing and [debugging](http://searchsoftwarequality.techtarget.com/definition/debugging) a [software](http://searchsoa.techtarget.com/definition/software) product under development, creating an online help reference, or performing a consumer-service-oriented task. For example, use cases in a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components.

The boundary, which defines the system of interest in relation to the world around it.

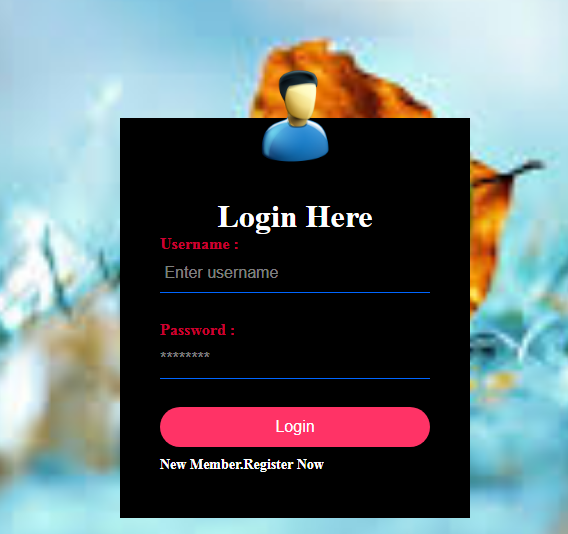
* The actors, usually individuals involved with the system defined according to their roles.
* The use cases, which are the specific roles played by the actors within and around the system.
* The relationships between and among the actors and the use cases.

**4.5.1 USER INTERFACE DESIGN**

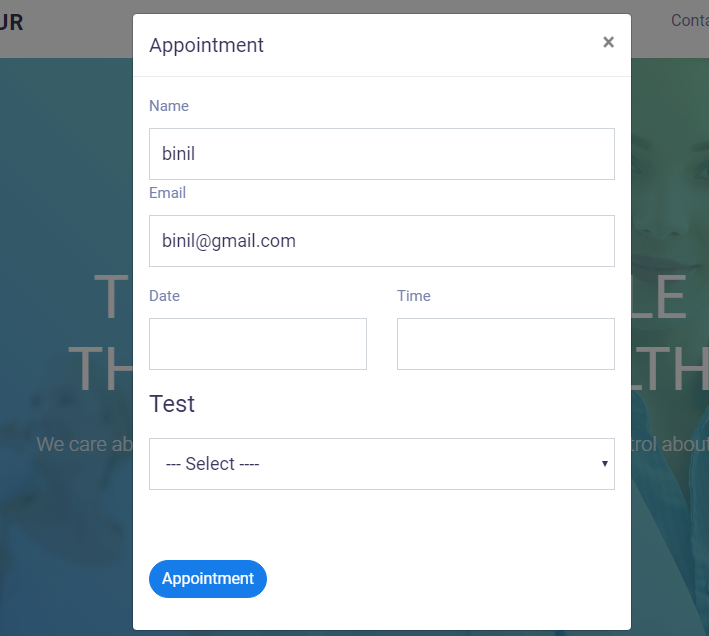
**User registration**

****

**Login**

****

**Appointment form**

****

**4.6 Database Design**

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

* Data Integrity
* Data independence

#### Relational Database Management System (RDBMS)

A relational model represents the database as a collection of relations. Each relation resembles a table of values or file of records. In formal relational model terminology, a row is called a tuple, a column header is called an attribute and the table is called a relation. A relational database consists of a collection of tables, each of which is assigned a unique name. A row in a tale represents a set of related values.

#### Relations, Domains & Attributes

A table is a relation. The rows in a table are called tuples. A tuple is an ordered set of n elements. Columns are referred to as attributes. Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity. A domain D is a set of atomic values. A common method of specifying a domain is to specify a data type from which the data values forming the domain are drawn. It is also useful to specify a name for the domain to help in interpreting its values. Every value in a relation is atomic, that is not decomposable.

#### Relationships

* Table relationships are established using Key. The two main keys of prime importance are Primary Key & Foreign Key. Entity Integrity and Referential Integrity Relationships can be established with these keys.
* Entity Integrity enforces that no Primary Key can have null values.
* Referential Integrity enforces that no Primary Key can have null values.
* Referential Integrity for each distinct Foreign Key value, there must exist a matching Primary Key value in the same domain. Other key are Super Key and Candidate Keys.

##### **Normalization**

Data are grouped together in the simplest way so that later changes can be made with minimum impact on data structures. Normalization is formal process of data structures in manners that eliminates redundancy and promotes integrity. Normalization is a technique of separating redundant fields and breaking up a large table into a smaller one. It is also used to avoid insertion, deletion, and updating anomalies. Normal form in data modelling use two concepts, keys and relationships. A key uniquely identifies a row in a table. There are two types of keys, primary key and foreign key. A primary key is an element or a combination of elements in a table whose purpose is to identify records from the same table. A foreign key is a column in a table that uniquely identifies record from a different table. All the tables have been normalized up to the third normal form.

As the name implies, it denotes putting things in the normal form. The application developer via normalization tries to achieve a sensible organization of data into proper tables and columns and where names can be easily correlated to the data by the user. Normalization eliminates repeating groups at data and thereby avoids data redundancy which proves to be a great burden on the computer resources. These include:

* Normalize the data.
* Choose proper names for the tables and columns.
* Choose the proper name for the data.

**First Normal Form**

The First Normal Form states that the domain of an attribute must include only atomic values and that the value of any attribute in a tuple must be a single value from the domain of that attribute. In other words 1NF disallows “relations within relations” or

“relations as attribute values within tuples”. The only attribute values permitted by

1NF are single atomic or indivisible values. The first step is to put the data into First Normal Form. This can be donor by moving data into separate tables where the data is of similar type in each table. Each table is given a Primary Key or Foreign Key as per requirement of the project. In this we form new relations for each non-atomic attribute or nested relation. This eliminated repeating groups of data. A relation is said to be in first normal form if only if it satisfies the constraints that contain the primary key only.

#### Second Normal Form

According to Second Normal Form, for relations where primary key contains multiple attributes, no non-key attribute should be functionally dependent on a part of the primary key. In this we decompose and setup a new relation for each partial key with its dependent attributes. Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it. This step helps in taking out data that is only dependent on a part of the key. A relation is said to be in second normal form if and only if it satisfies all the first normal form conditions for the primary key and every non-primary key attributes of the relation is fully dependent on its primary key alone.

#### Third Normal Form

According to Third Normal Form, Relation should not have a non-key attribute functionally determined by another non-key attribute or by a set of non-key attributes. That is, there should be no transitive dependency on the primary key. In this, I decompose and set up relation that includes the non-key attributes that functionally determines other non-key attributes. This step is taken to get rid of anything that does not depend entirely on the Primary Key. A relation is said to be in third normal form if only if it is in second normal form and more over the non key attributes of the relation should not be depend on other non-key attribute.

TABLES

1.**login**

Table no : 1

Table name : tbl\_login

Primary key :id

Table Description : To store login details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Size** | **Description** |
| login\_id | Int | 4 | Primary key |
| Username | varchar | 50 | User name |
| Password | varchar | 50 | password |
| user\_type | Int | 4 | usertype |
| Status | Int | 4 | status |

2.**User registration**

Table no : 2

Table name : tbl\_uregi

Primary key :id

Table Description : To store user details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Size** | **Description** |
| user\_id | int | 4 | Primary key |
| Name | varchar | 50 | Name |
| Place | varchar | 50 | Place of user |
| phoneno | varchar | 50 | Phone Number |
| Email | varchar | 50 | Email id |
| username | varchar | 50 | User Name |
| password | varchar | 50 | Password |
| Status | int | 4 | Present/not |

3.**Technician registration**

Table no : 3

Table name : tbl\_tecregi

Primary key :tec\_id

Table Description : To store technician details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Size** | **Description** |
| tec\_id | varchar | 50 | Primary key |
| name | varchar | 50 | Name |
| place | varchar | 50 | Place of technician |
| gender | varchar | 20 | Male/female |
| qualification | varchar | 50 | Qualification |
| experience | varchar | 20 | Experience |
| regi\_no | int | 10 | Registration number |
| email | varchar | 50 | Email id |
| username | varchar | 50 | username |
| password | varchar | 50 | password |
| status | int | 4 | Present/not |

4.**Quqlification**

Table no : 4

Table name : tbl\_quali

Primary key :q\_id

Table Description : To store qualification details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Size** | **Description** |
| qualify\_id | int | 4 | Qualification id |
| qualification | varchar | 50 | Qualification |

5.**Add test**

Table no : 5

Table name : tbl\_test

Primary key :test\_id

Table Description : To store test details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Size** | **Description** |
| test\_id | int | 4 | test id |
| testtype | varchar | 20 | Test details |
| normal | varchar | 20 | Normal range of test |

6.**Appointment**

Table no : 6

Table name : tbl\_appointment

Primary key :app\_id

Table Description : To store appointment details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Size** | **Description** |
| app\_id | int | 4 | Primary key |
| bookingdate | varchar | 10 | Date of booking |
| time | int | 20 | Time of booking |
| name | varchar | 20 | name |
| test | int | 10 | Select test |

7.**Result**

Table no : 7

Table name : tbl\_result

Primary key :result\_id

Table Description : To store result details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Size** | **Description** |
| result\_id | Int | 4 | Primary key |
| id | int | 4 | userid |
| name | varchar | 20 | name |
| result | varchar | 20 | result |

8.**Feedback**

Table no : 8

Table name : tbl\_feedback

Primary key :f\_id

Table Description : To store feedback details

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Field Type** | **Size** | **Description** |
| f\_id | int | 4 | Feedback id |
| name | varchar | 40 | User\_name |
| message | varchar | 50 | feedback |

**CHAPTER 5**

# SYSTEM TESTING

## 5.1 INTRODUCTION

Software Testing is the process of executing software in a controlled manner, in order to answer the question - Does the software behave as specified? Software testing is often used in association with the terms verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, analysis, inspections, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

Software testing should not be confused with debugging. Debugging is the process of analysing and localizing bugs when software does not behave as expected. Although the identification of some bugs will be obvious from playing with the software, a methodical approach to software testing is a much more thorough means for identifying bugs. Debugging is therefore an activity which supports testing, but cannot replace testing.

Other activities which are often associated with software testing are static analysis and dynamic analysis. Static analysis investigates the source code of software, looking for problems and gathering metrics without actually executing the code. Dynamic analysis looks at the behaviour of software while it is executing, to provide information such as execution traces, timing profiles, and test coverage information.

Testing is a set of activity that can be planned in advanced and conducted systematically. Testing begins at the module level and work towards the integration of entire computers based system. Nothing is complete without testing, as it 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 intent of finding an error.
* A good test case is one that has high possibility of finding an undiscovered error.
* A successful test is one that uncovers an undiscovered error.

If a testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrate that the software function appear to be working according to the specification, that performance requirement appear to have been met.

There are three ways to test program.

* For correctness
* For implementation efficiency
* For computational complexity

Test for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

## 5.2 TEST PLAN

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The Test Plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers is always responsible for testing the individual units of the programs, ensuring that each performs the function for which it was designed. There is an independent test group (ITG) which is to remove the inherent problems associated with letting the builder to test the thing that has been built. The specific objectives of testing should be stated in measurable terms. So that the mean time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours per regression test all should be stated within the test plan.

The levels of testing include:

* Unit testing
* Integration Testing
* Data validation Testing
* Output Testing

### 5.2.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module.

The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm’s execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, all error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur. Boundary testing is the last task of unit testing step. Software often fails at its boundaries.

Unit testing was done in Sell-Soft System by treating each module as separate entity and testing each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of the modules were found and were rectified. After coding each module is tested and run individually. All unnecessary code where removed and ensured that all modules are working, and gives the expected result.

### 5.2.2 Integration Testing

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expanse of entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop.

After performing unit testing in the System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

### 5.2.3 Validation Testing or System Testing

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly known as Black Box testing or System tests.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

### 5.2.4 Output Testing or User Acceptance Testing

The system considered is tested for user acceptance; here it should satisfy the firm’s need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points:

* Input Screen Designs,
* Output Screen Designs,

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

**CHAPTER 6**

**IMPLEMENTATION**

### 6.1 INTRODUCTION

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

At this stage the main work load, the greatest upheaval and the major impact on the existing system shifts to the user department. If the implementation is not carefully planned or controlled, it can create chaos and confusion.

Implementation includes all those activities that take place to convert from the existing system to the new system. The new system may be a totally new, replacing an existing manual or automated system or it may be a modification to an existing system. Proper implementation is essential to provide a reliable system to meet organization requirements. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to the new system. The system can be implemented only after through testing is done and if it is found to be working according to the specifications. The system personnel check the feasibility of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required to implement the three main aspects: education and training, system testing and changeover.

The implementation state involves the following tasks:

* Careful planning.
* Investigation of system and constraints.
* Design of methods to achieve the changeover.
* Training of the staff in the changeover phase.

### 6.2 IMPLEMENTATION PROCEDURES

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system. In many organizations someone who will not be operating it, will commission the software development project. In the initial stage people doubt about the software but we have to ensure that the resistance does not build up, as one has to make sure that:

* The active user must be aware of the benefits of using the new system.
* Their confidence in the software is built up.
* Proper guidance is imparted to the user so that he is comfortable in using the application.

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not up running on the server, the actual process won’t take place.

#### 6.2.1 User Training

User training is designed to prepare the user for testing and converting the system. To achieve the objective and benefits expected from computer based system, it is essential for the people who will be involved to be confident of their role in the new system. As system becomes more complex, the need for training is more important. By user training the user comes to know how to enter data, respond to error messages, interrogate the database and call up routine that will produce reports and perform other necessary functions.

#### 6.2.2 Training on the Application Software

After providing the necessary basic training on computer awareness the user will have to be trained on the new application software. This will give the underlying philosophy of the use of the new system such as the screen flow, screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the ways to correct the date entered. It should then cover information needed by the specific user/ group to use the system or part of the system while imparting the training of the program on the application. This training may be different across different user groups and across different levels of hierarchy.

#### 6.2.3 System Maintenance

Maintenance is the enigma of system development. The maintenance phase of the software cycle is the time in which a software product performs useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for system maintenance is for it to make adaptable to the changes in the system environment. Software maintenance is of course, far more than "Finding Mistakes".

# 

# 

# CHAPTER 7

**CONCLUSION**

**AND FUTURE SCOPE**

**7.1 CONCLUSION**

The software reduces the time consumption and the manual efforts of lab test. It will be a simple platform for users to access services for their huge needs.

* The benefits, we can obtain from the new system are:
* Timely and accurate information will be available
* Reduced data loss
* The access time and process time is highly reduced
* Quick data view
* Error free output

The proposed system is expected to replace manual system and provide more efficient performance and services.

## 7.2 FUTURE SCOPE

* The system is designed in such a way that the payment of bidding service should be done in online mode.
* The patient test is booked and the lab may now collect samples from patients registered address.
* After successful testing, the patient or the user gets a notification of test result through an email.
* Chat with technician and users.

**CHAPTER 8 BIBLIOGRAPHY**

**BOOKS/REFERENCES:**

* Gary B. Shelly, Harry J. Rosenblatt, “*System Analysis and Design*”, 2009.
* Roger S Pressman, “*Software Engineering*”, 1994.
* PankajJalote, “So*ftware engineering*: a precise approach”, 2006.
* James lee and Brent ware Addison, “Open source web development with LAMP”, 2003
* IEEE Std 1016 Recommended Practice for Software Design Descriptions.

**WEBSITES:**

* www.w3schools.com
* www.jquery.com
* http://homepages.dcc.ufmg.br/~rodolfo/es-1-03/IEEE-Std-830-1998.pdf
* www.agilemodeling.com/artifacts/useCaseDiagram.htm