

A Project report on

ONLINE EXAMINATION SYSYTEM

Submitted in partial fulfillment of the requirements

for the award of the Degree of

**Bachelor of Science
IN
COMPUTER SCIENCE**

Submitted By

K.UMA GANESH (2173080)

Under the Esteemed Guidance of

Kum.V.LALITHA , M.B.A.

Lecturer in Computer Science



U.G. DEPARTMENT OF COMPUTER SCIENCE

VSM COLLEGE (A)

(Recognized by Adikavi Nannaya University, Rajamahendravaram)

Ramachandrapuram-533255, E.G.Dist (A.P).

2017-20

VSM COLLEGE (A)

(Recognized by Adikavi Nannaya University, Rajamendravaram)

Ramachandrapuram -533255, E.G.Dist (A.P).

U.G. DEPARTMENT OF COMPUTER SCIENCE

CERTIFICATE



This is to certify that the project entitled “*ONLINE EXAMINATION SYSTEM*” that is being submitted by *K.UMA GANESH (2173080)*

In Partial Fulfillment the requirements for the award of the Degree of **Bachelor of Science** in “**Computer Science**” in the academic year 2017-20 to **VSM College(A)** affiliated to **Adikavi Nannaya University, Rajamahendravaram**, is a record of Bonafied work carried out by him/her under my guidance and supervision.

The results embodied in this project have not been submitted to any other University or Institute for the award of any degree or diploma.

INTERNAL GUIDE

(**Kum.V.LALITHA** M.B.A.)

PROJECT CO-ORDINATOR

(**Sri.M Phani Kumar** M.C.A.)

HEAD OF THE DEPARTMENT

(**Sri.MVV Chowdary** M.C.A.,M.Tech.,M.B.A.)

EXTERNAL EXAMINER

ONLINE EXAMINATION SYSTEM

ACKNOWLEDGEMENTS

I would like to take the privilege of the opportunity to express my gratitude into Project work of “*ONLINE EXAMINATION SYSTEM*” enabled us to express our special thanks to our Principal **Sri.NSV Kiran Kumar** who has shown keen interest in us and encouraged us by providing all the facilities to complete my project successfully.

I owe my gratitude to our beloved Head of the Department Computer Science **Sri. MVV Chowdary** for assisting us in completing our project work.

I express my sincere thanks to my internal guide “**Kum.V.LALITHA**” who has been a source of inspiration for us throughout my project and for her valuable advices in making my project a success.

I wish to express my sincere thanks to all teaching and non-teaching staff of Computer Science Department. I wish to express my special thanks to all the Faculty members of our College for their concern in subjects and their help throughout my course.

I am very thankful to all my friends who had given me good co-operation and suggestions throughout this project and helped me in successful completion.

K.DURGA BHAVANI (2173069)

K.PAVAN KALYAN (2173070)

K.UMA GANESH (2173080)

Abstract

Online Examination system is a software solution, which allows a particular company or institute to arrange, conduct and manage exams through online. This project is helpful for educated peoples to practice different mock test. Online Examination was developed to examine the student's technical knowledge. The people can able to write online examinations only after registration. In current generation due to increasing the competition in society online examination came into existence for example likes SET, CAT, Aptitude test or placement purpose.

In Online Examination students are available with one question and more than two options or answers, students are independent to choose their answer, In Online Examination System the time management should be perfect. In Online Examination System makes the result available at minimum time. The results are more accurate.

TABLE OF CONTENTS

ABSTRACT

LIST OF FIGURES

LIST OF TABLE

TITLE.**PAGE NO****1. Introduction**

Purpose

Scope

Technologies to be used

Overview

2. Overall Description

2.1 Goals of proposal system

2.2 Problems in existing system

2.3 Methodology

Module Description

2.4 Hardware and Software requirements

3. Software requirements specification

3.1 Introduction

3.2 Data Collection

3.3. SRS Document

3.4 Role of SRS

4. Feasibility Study

4.1 Technical Feasibility

4.2 Economical Feasibility

4.3 Operational Feasibility

5. Data Flow Diagrams

Uml diagrams

Class diagrams

Sequential diagram

6. Tables

Mst_admin

Mst_question

Mst_result

Mst_subject

mst_test

mst_user

mst_useranswer

7. Forms

Home page

New user signup

Index page

Sublist.php page

Showtest page form

Quiz form

Result form

Review page

8. Coding

9. TESTING

INTRODUCTION

UNIT TESTING

WHITE BOX TESTING

BASIC PATH TESTING

CONDITIONAL TESTING

DATA FLOW TESTING

CONCLUSION

1. Introduction

Online Examination System is a multiple Choice Questions based Examination System. It provides an easy to use environment for both Test Conductors and Students appearing for examination. The Online examination System helps in speeding up the process of conducting Examination. Teachers will be able to create Examinations by compassing a set of Questions.

1.1 Purpose:-

The purpose of the System is to develop Online Examination System used to test the Domain Knowledge of the Students, and employees with respect to the particular technology. The manual procedure used for conducting exam is time consuming process and error prone due to human limitations. The purpose of this application is to provide an application which will considerable reduce the time required to give the exam and know the result. Being an integrated Online Examination System it will reduce the paper work. Response by the candidates will be checked automatically and instantly.

1.2 Scope:-

Online Examination System can be used in educational institutions as well as in corporate world. The system handles all the operations and generates reports as soon as the test is completed which saves the precious time of faculties spent on reviewing answer sheets. It is cost-effective and a popular means of mass-evaluation system. The administrator of the system prepares the tests and questions for each exam. The candidates can login through client computers with their admission number given to them by the college and can take exam.

1.3 Technologies to be used:-

Front end: - PHP

Back end: - MY SQL

1.4 Overview:-

The project “Online Examination System” assesses students by conducting online objective tests. The tests would be highly customizable. For students they give papers according to their convenience from any location by using internet and time there is no use of extra things like Paper, Pen etc.

The project allows based facilities’ to create their own tests. It would enable educational institutes to perform tests, quiz and create feedback forms. This project would be helpful for creating practice test, say for educational institutes and a hiiiis a feedback.

2. Overall Description

2.1 Goals of proposal system:-

In comparison to the current system, the proposed system will be less time consuming and will be more efficient. Analysis will be very easy in the system as it is automated and result will be very precise and accurate and will be declared in very short span of time because calculations and evaluations are done by simulator itself.

The proposed system is very secure as no chances of leakage of questions paper as it is dependent on the administrator only. The logs of appeared candidates and their marks are stored and can be backed up for future use. There is no need of papers, pens etc. separate panels for administrators and panels. Easy to add questions. Each student gets Random questions. Easy to manage exams and questions.

2.2 Existing system:-

The existing system of conducting examination process is manual. Existing system is a large man power process and is difficult to implement it at different platform. It has so many problems. So we introduce an Online Examination System (OLES), which is fully computerized.

The existing system is very time consuming. It is difficult to analyze the exam manually. Results are not precise as calculation and evaluations are done manually. Result processing after summation of exam takes more time as it is done manually. To the exam of more candidates more invigilators are required but no need of invigilators in case of Online Examination. The chances of paper leakages are more in current system as compared to proposed system.

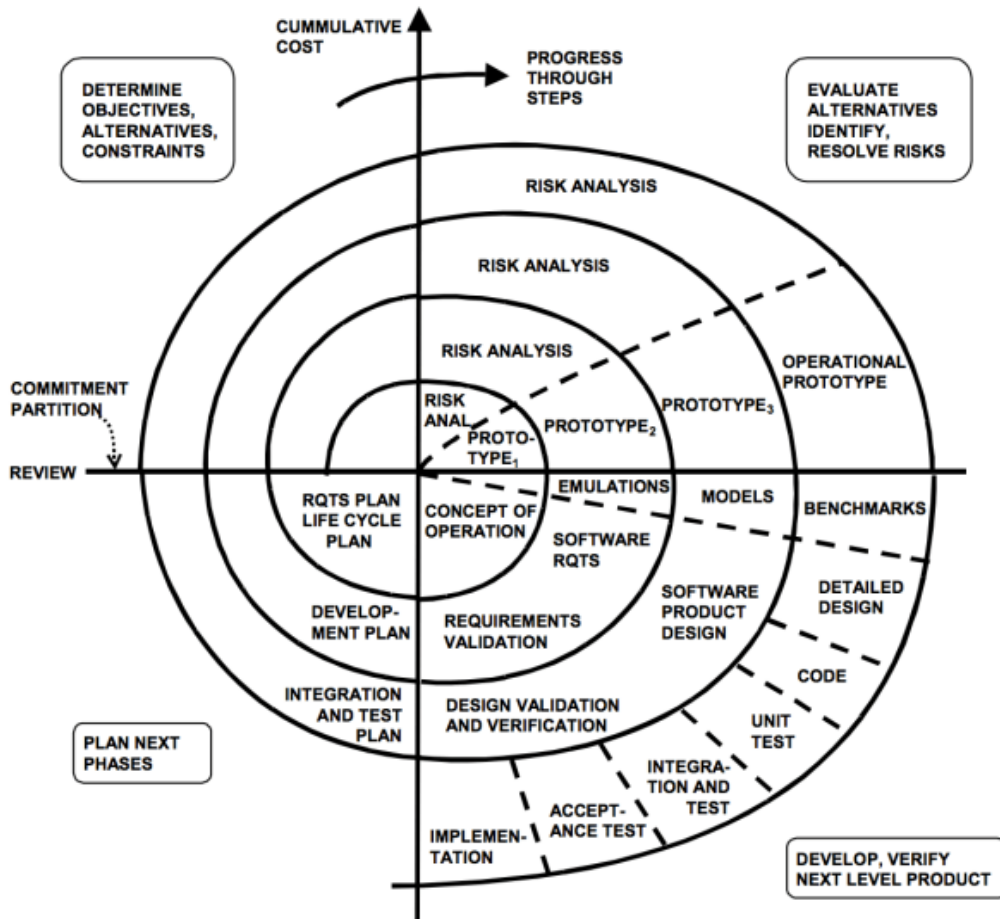
2.3 Methodology:-Spiral model is a risk drive model generator for software project based on the unique risk pattern of a given project spiral model guides a team to adopt elements of one or more process models.

The spiral model combines the ideal of iterative development with the systematic controller of the waterfall model.

Spiral model is not so well known as other SDLC (Software Development Life Cycle) model for example and here the reason spiral model can be partly costly to use and does not work well for small projects.

It is a risk-driven model which means that the overall success of a project highly depends on the risk analysis requires of a specific expertise on every iteration thus to review and analyze the project from time to time, special skills are needed.

In a few words spiral model can be characterized by repeatedly iterating a set of elemental development process and eliminating risk, so it is acting being reduced. To understand how you can get your goals using spiral model.



As you can see, the spiral model consists of four main software development life cycle phases. The whole development process repeatedly passes through these stages; each iteration is called a Spiral.

Four main phases are:-

1. Determine Objectives
2. Alternatives
3. Constraints
4. Planning Phase.

This is where everything starts. Team members try to gather the product objectives, requirements (e.g., Business requirements specifications or SRS), alternatives in design, etc. In the subsequent spirals, all requirements are generated according to the customer's feedback.

2.4 Module Description:-

In our project "Online Examination System". We can divide overall programs into modules

These modules are:

1. MST_ADMIN
2. MST_ques

3. MST_Sub
4. MST_Test
5. MST_User
6. MST_UserAns
7. MST_Result

MST_ADMINThis module contains the information about the admins of the examination system and the whole control of the examination system is managed by the admin. In this module the controller will Create, Delete and Edit admin for the exam dept with all of their details like Login name and password has been provided through which the admin can later login to perform all of their tasks.

MST_ques:

This module consists of the number of questions along with the question_id, test_id, Ques_desc, and answers with 2 or more options and true answer.

MST_Sub:

This module consists of different subjects that the exam writers are able to select their subject by selecting sub_id and sub_name.

MST_Test:

This module refers to which test are you going to write by using test id and consists of information like sub_id, test name and total number of questions.

MST_User:

This module is for user login for examination by using their self details correctly .After success login they are able write exam.

MST_UserAns:

This module consists of the user answers after the examination and all details about the user like test id, ques desc, answers along with correct and wrong answers.

MST_Result:

This module contains the result of the examination by the users along with the identification by using their login

3. SRS DOCUMENT

STRUCTURE

- Introduction

OVERALL DESCRIPTION

- Product Perspective
- Product function
- User Characteristics
- Constrains
- Assumption and Dependency

SPECIFICATION REQUIREMENTS

- External Interface
- Functions
- Performance requirements
- Logical Database of requirements
- Design Constraint
- S/w System attributes

Change Management System

Document Approvals

Supporting Information

3.1 Software Requirements Specification:-

INTRODUCTION:-

A Software requirements specification is a document that lays out the description of the s/w requirements that is to be developed as well the intention of the software requirement specification shows that the software is supposed to do as well as how it is supposed to perform .It is written down before the actual software development work starts.

Software requirements specification is important for developers have to expand to achieve desired software goals. It thus reduces development cost, the lesser the developers will change from the client. And if composed properly, an SRS ensures that these is less chance of mistake on the part of developers as they have clear idea on the functionalities and internalities of the software it also helps..

To clear any communication problems between the client and the developer. Furthermore an SRS. Serves to form a foundation of mutual argument between the client and the developers. It also serves as the document to verify the testing process. A format of SRS given by IEEE.

OVERALL DESCRIPTION:-

This selection of SRS describes the general factor that affects the product and its requirement. It does not state specific requirements. It summarizes them and makes them easier to understand. It is largely statement of what is in sections of SRS in summarized form.

PRODUCT PERSPECTIVE:-

This part describes the context within which the product is begin built. It also shows if the software is a part of a product family, are replacement for any already existing members or a completely new and unique product also if the SRS only describes a product of a large system, then this point will layout the requirements from the larger system for this part the operate effectively a diagram should be drawn here to show the chief components of the entire system the interlinking connection and also interfaces.

USERS CLASSES AND CHARACTERISTICS:-

This Part identifies the user classes that will use the product. These classes may be categorized based on which function they are, usage frequency, technical expertise, experience etc... The relevant characteristics of each class should be given here. Also the most important users. Classes should be identified here.

OPERATING ENVIRONMENT:-

This segment describes the environment in which the product will operate; such as the platform (Hardware) operating system version etc... Also any other components the application will cohabit.

DESIGN CONSTRAINTS:-

This segments layout constraints or issues that will limit options to developers.

Some of these might be hardware limitations, interfaces particular technology, parallel operations languages requirements, regularity policies, corporate, policies, security provisions etc..., user documentation.

This part shows the user's manual, factorials etc..., That will be provided a long with the software.

ASSUMPTION AND DEPENDENCIES:-

These Part lists of any assumptions that could affect the requirements started in the SRS. The software would not work to the desired level. If these assumptions are incorrect or change. Also tested to any dependency the Product will have an external factor.

SPECIFICATION REQUIREMENTS:-

1.External interfaces:-

External interface requirements specify hardware, software or database elements with which a system or component must interface. This section provides information to ensure that the system will communicate Properly with external components. If different portions of the product have different interfaces, incorporate an instance of this section with in details requirements for each portion.

2. USER INTERFACE:-

This part describes the logical characteristics of each interface between the software and the user. This might include screen images samples, Graphical user interface standards, screen layout constraints, buttons and functions, Key board shortcuts, messages. Display etc..., the software components. For which user interface is needed will also be defined.

3. SOFTWARE INTERFACE:-

This segment describe the connections between the product any other specific software components, be it operation systems, database libraries etc. Data items and message going into the system and going is identified and the purpose of each is described.

4. COMMUNICATE INTERFACE:-

This part describes any communication functions required by the software, such as e-mail, web browser, network server communication protocols, electronic forms etc. Any communication standard to be used is also identified communications security and or encryption issues are specified.

FUNCTIONS:-

- Content Management
- User Administration
- Web menu Administration
- Template Administration
- Selections and categories Administration
- General website options Administration

PERFORMANCE REQUIREMENTS:-

All the performance constraints on the software system are two types of performance requirements.

STATIC:-

- Do not impose constraint on the execution.
- Number of terminals and number of simultaneous users.
- These static requirements are called capacity requirements.

DYNAMIC:-

- Specify constraints on the execution behavior
- Generally on response time
- Must be in measurable terms.
- Repose time should be good

LOGICAL DATABASE OF REQUIREMENTS:-

This should specify the logical requirements for any information that is to be placed into a database . This may include the following.

- Types of information used by various functions.
- Frequency of use
- Accessing capabilities.
- Data entities and their relationships.
- Integrity constraints.
- Data retention requirements.

DESIGN CONSTRAINT :-

The information of all users and their examinations must be stored in a stored in a database that is accessible by the website.

- User may access from any compute. That internet browsing capabilities and an internet connection.
- User must have their correct user names and passwords to enter into their accounts and do actions

SOFTWARE SYSTEM ATTRIBUTES:-

A software requirements specification include in depth description of the software that will be developed. A system requirements specification collects information on the requirements for a system.

“SOFTWARE” and “SYSTEM” are sometimes used interchangeably as SRS. But, a software requirements specification provides greater detail that a system requirement specification.

CHANGE MANAGEMENT PROCESS:-

Determines the change management process in order, to identify, evaluate and update in the project scope and requirements.

Change management systems follow the standardized methods and procedures to incorporate the changes in the system. These systems perform an end to end. Impact analysis which in turn is helpful in taking effective measure. All the above mentioned features of change management system. Helps in making the product or system successful and also help in project management.

DOCUMENT APPROVALS:-

Provide information about the approvals of the SRS document with the details, such as approvers name, signature, date and soon.

Document approvals are a real sore point in most organization. The software system employed for managing the approval process can range from shared file folders to email to printouts not surprisingly no one is excited. About these systems as they tend to be wasteful, slow and opaque.

5.1 SUPPORTING INFORMATION:-

Provide Information , such as a table of contents index and soon. This is necessary especially when SRS is prepared for large and complex projects.

The purpose of the supporting information is to enable authors to provide and achieve supporting information such as data table, methods, information, figures, video or computer software in digital formats,

5.2 DATA COLLECTION:-

The data collection, ocquistation and analysis activates. That SRS administrator have grown over the past half century to address the changing data and information needs to policy makers, managers,, educators and researchers in the science and technology policy arena. The science and engineering enterprises was significantly reshaped during World War 2 to include a substantial public role to meet nation's health. To make informed decisions. Within the new context, federal policy markers required data on and analysis of science and engineering resources issues in narrow topic for SRS to meet the border needs, it will need to make further efforts to link and integrate its data sets.

5.3 SRS DOCUMENT:-

A Requirements specification for a software system is a complete description

Of the behavior of a system to be developer it includes set to use cases that describes all interactions that users will have with the software.

The software specification Requirements constitute of fictional requirements and non functional Requirements.

FUNCTIONAL REQUIREMENTS:-

It describes what the software will do and the what quality attributes are used. It describes functionality or system services. It depends on type of software, expected users.

NON FUNCTIONAL REQUIREMENTS:-

The non functional requirements are that used to evaluate the performance of the software, such as security, safety, portability these define system properties and constraints.

Eg:- Reliability, Response time, storage requirements.

5.4ROLE OF SRS:-

A software requirements specification is a description of a software system to be developed. It is modeled after business requirements specification also known as stakeholder Requirements specification.

A software requirements specification is a document that lays out the description of the software that is to be developed as well as the intention of the software under development. Software requirement specification shows what the software is supposed to do as well as how it is supposed to perform. It is written down before the actual software development work starts.

PURPOSE FOR ROLE OF SRS:-

This gives the purpose of the SRS document, not the software itself. It also states how much of the software is covered by the document, particularly saying whether it describes the entire software system or only a part of it. It also states the intended readers of the document.

Any additional quality characteristics of the product. That might be important are specified here. Such as adaptability interpretability,, availability, correctness, maintainability robustness, usability etc. These are written in a specific verifiable and quantitative way.

Feasibility Study

Feasibility is defined as the practical extent to which a project can be performed successfully.

- Analyze the software will meet the organizational requirement or not.
- Determine whether the software can be implemented using current technology and within the specified budget and scheduled (or) not.
- Determine whether the software can be integrated existing software (or) not.

Types of feasibility:

1. Technical feasibility
2. Operational feasibility
3. Economical feasibility

Technical feasibility:

Technical feasibility assesses the current resources and technology, which are required to accomplish user requirements in the software within the allocated time and budget

- Analyzes the technical skills and capabilities of software development team members.
- Determine whether the relevant technology is stable and established or not.
- The technology chosen for software development has large number of users so that they can be consulted when problems arise (or) improvements are required.

Operational Feasibility

Operational feasibility assesses the extent to which the required software performs series of steps to solve business problems and user requirements

This feasibility depends on human resources.

Determines whether the solution suggested by software development team is acceptable (or) not.

- Analyze whether users will adapt to new software or not.
- Determines whether, The organization is satisfied by the alternative solutions proposed by software development team (or) not.

Economic Feasibility:

Economic feasibility determines whether the required software is capable of generating financial gains for an organization (or) not involves in.

- Cost incurred on software development produces long term gains for an organization
- Cost required to conduct full software investigation.
- Cost hardware, software, development team and training.

Usage examples:

- After reviewing the economic feasibility of sending you to mars, we have concluded a better investment for humanity is dumping red eye into Olympic sized swimming pool and giving you a rubber ducky.
- They were looking at upgrading their computer systems, but once they looked at the economic feasibility they recognized and realized the additional revenues would not offset the costs.

- We did not know if there was enough economic feasibility to continue on with our PROJECT and do what we were gonna do.

DATA DESIGN PHASE

Data Flow Diagram

A data flow diagram (DFD) is a structured analysis and designed tool that can be used for flow charting. A DFD is a network that describe the flow of data and the processes that change or transform the data throughout a system. This network is constructed by using a set of symbols that do not imply any physical implementation it has the purpose of clarifying system requirements and identifying major transformations. So it is the starting point of the design phase that functionally decompose the requirements specifications down the lowest level of detail. DFD can be considered to an abstract of the logic of an information oriented or process oriented system flow chart. For these reasons DFD's are often referred to as logical data flow diagrams.

EXTERNAL ENTITY

A external entity is a source or destination of data flow. Only those entities which originate or receive data are represented on a data flow diagram. The symbol used is a rectangular box.

PROCESS

A process shows a transformation or manipulation of data flow within the system. The symbol used is an oval shape.

DATA FLOW

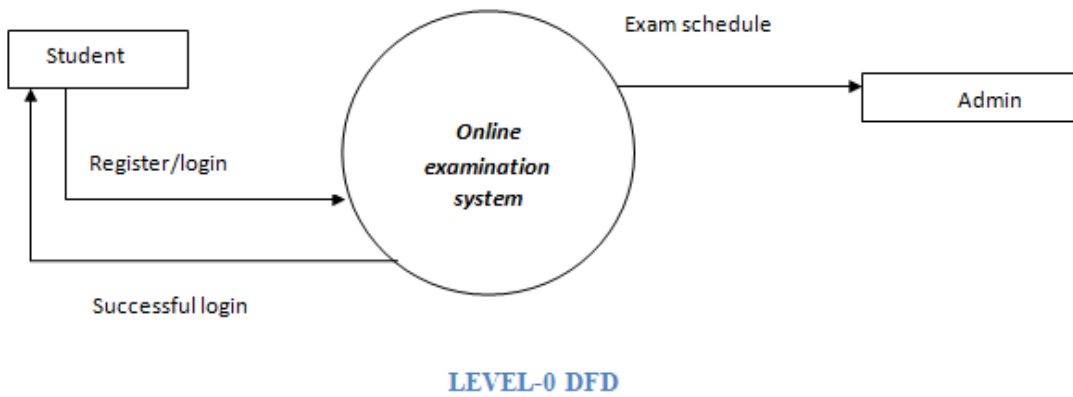
The data flow shows the flow of information from a source to its destination. Data flow is represented by a line, with arrow heads showing the direction of flow. Information always flow may be written, verbal or electronic. Each data flow may be referenced by the process or data stores at its head and tail, or by a description of its contents.

DATA STORE

A data store is a holding place for information within the system. It is represented by an open ended narrow rectangle, data stores may be long-term files suchas sales ledgers, or may be short-term accumulations for example batches of documents that are waiting to be processed, Each data store should be given a reference followed by an arbitrary number.

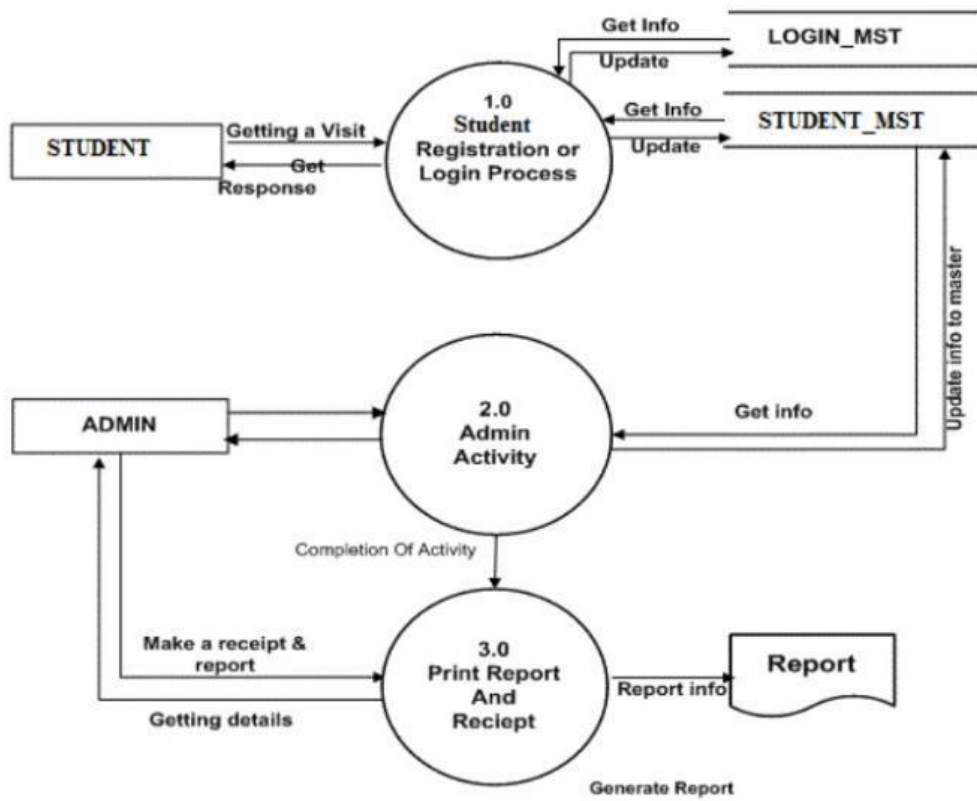
Levels of DFD:-

Level 0:- highest abstraction level DFD is known as level0 DFD, which depicts the entire information process as one diagram connecting all the underline details. Level 0 DFD's are also known as context level DFD's.



Level-1:-

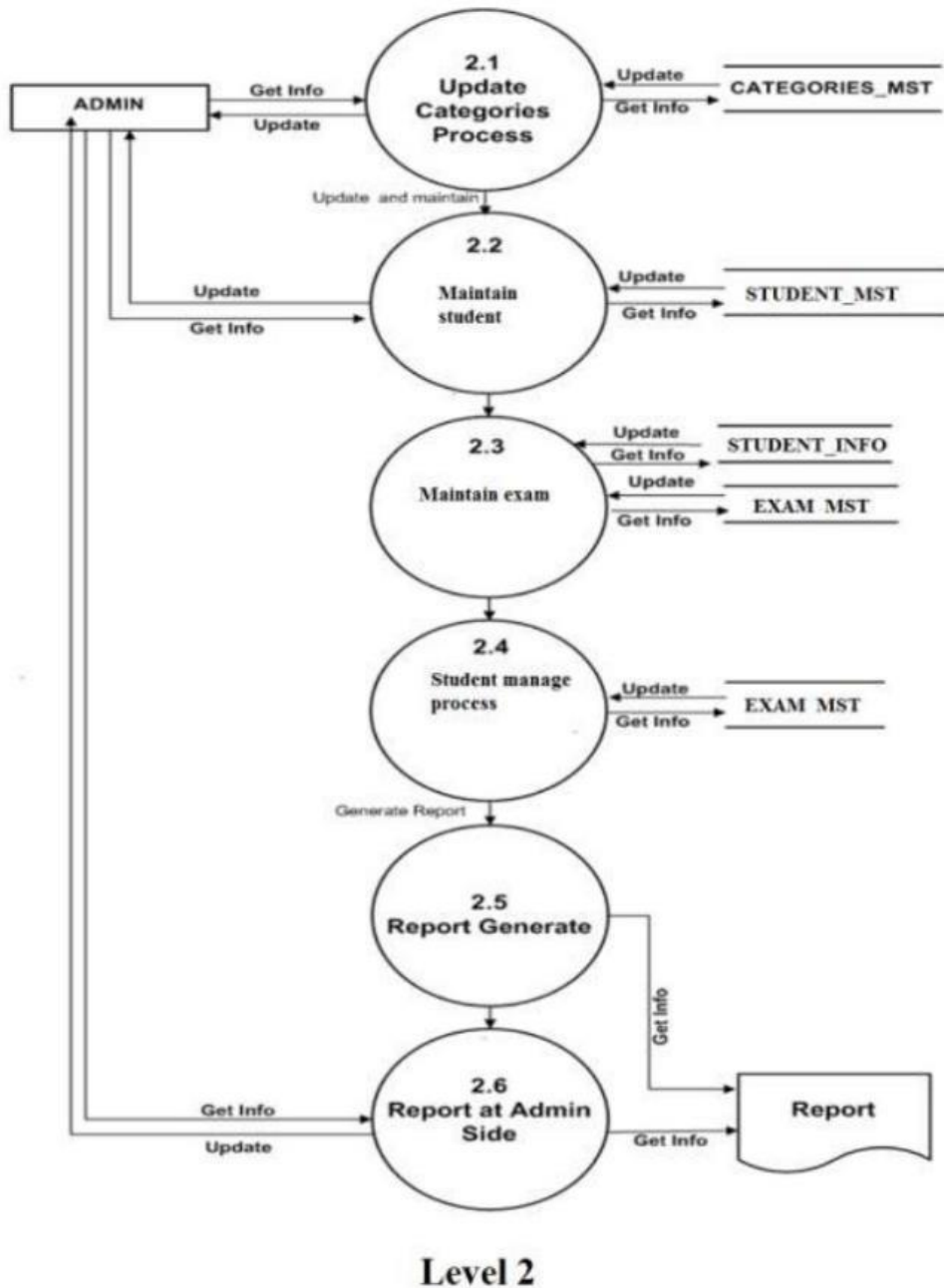
The level 0 DFD is broken into more specific, level-1 DFD depicts basic modules in the system and flow of data among various modules level-1 DFD mentions basic process and sources of information.



Level 1

Level-2:-

At this level DFD shows how data flows inside the modules mention in level-1.



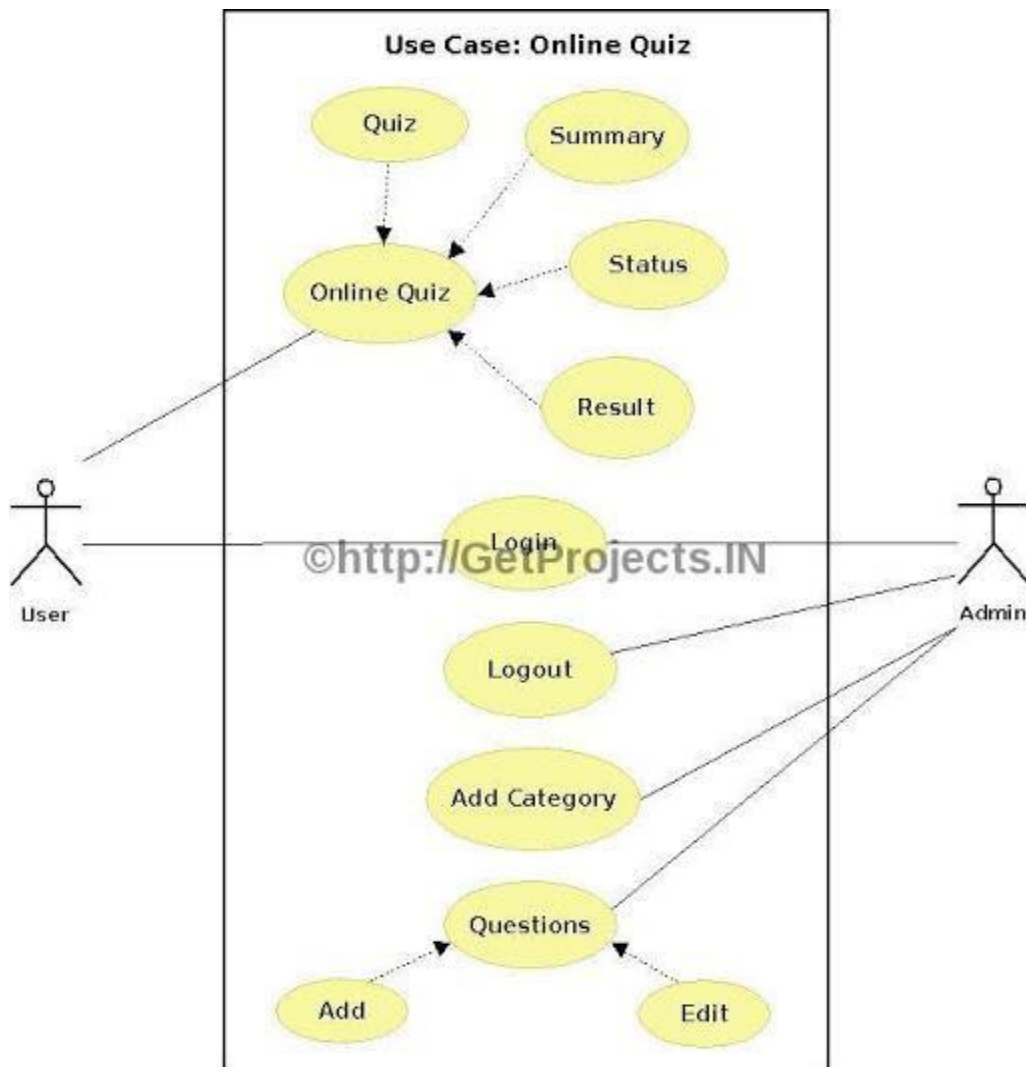
UML DIAGRAMS

A Uml Diagrams is a diagram based on the unified modeling language (UML) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes on order to better understand, alter, maintain or document information about the system.

1. Use Case Diagram:-

A Use case diagram as its simplest is a representation of a user's interaction with the system that shows the relationships between the user and the different use cases in which the user is involved.

A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.



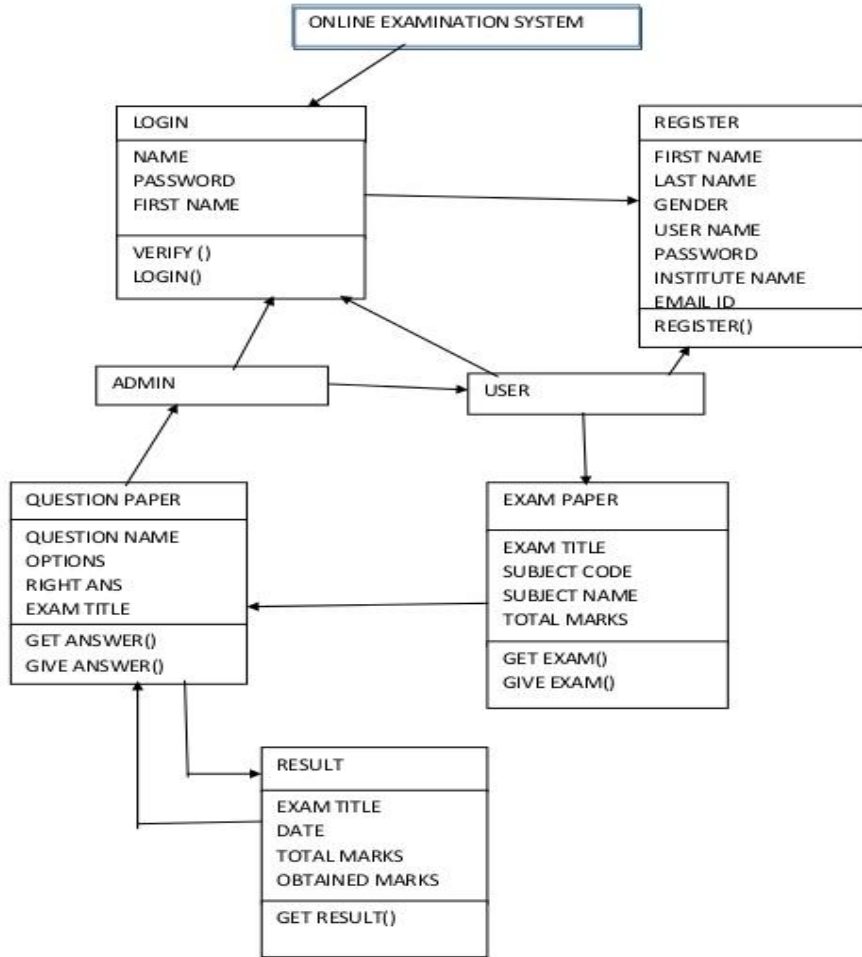
2.CLASS DIAGRAM:-

In software engineering, a **class diagram** in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

ONLINE EXAMINATION SYSTEM

CLASS DIAGRAM:



10

3. Activity Diagram:

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part

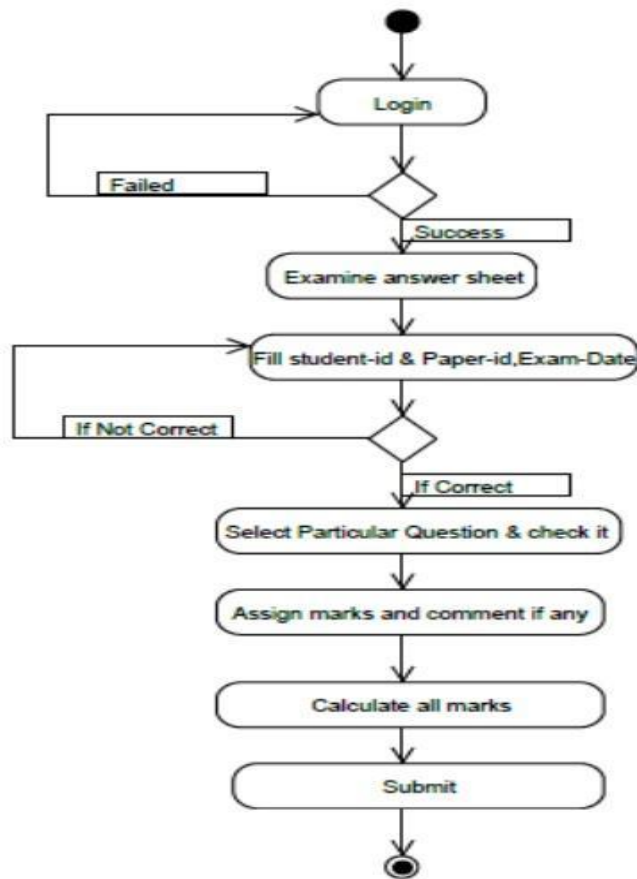
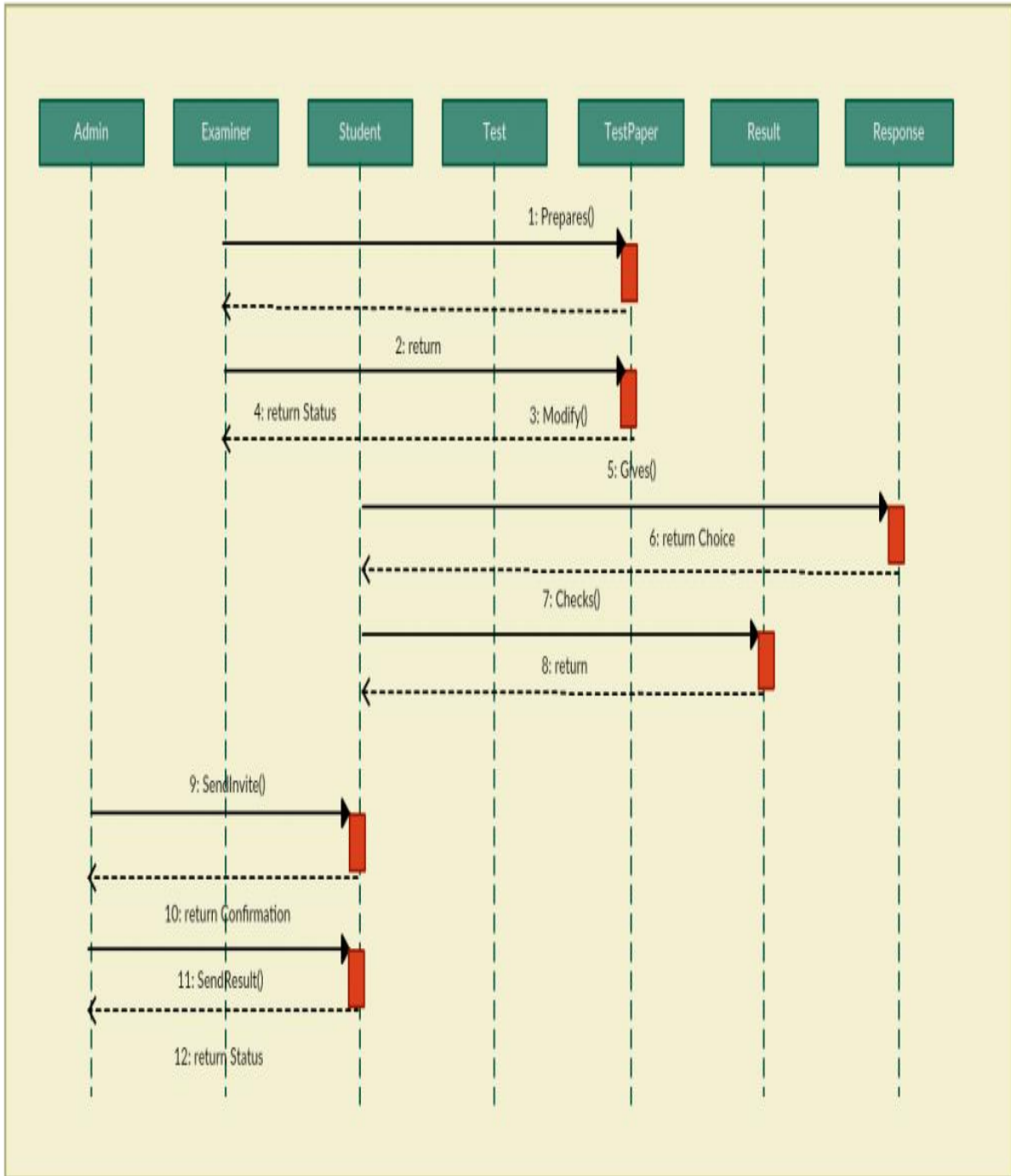


Fig. 3.7 Activity Diagram Evaluation (Created by Umllet software)

4. Sequential Diagram:

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

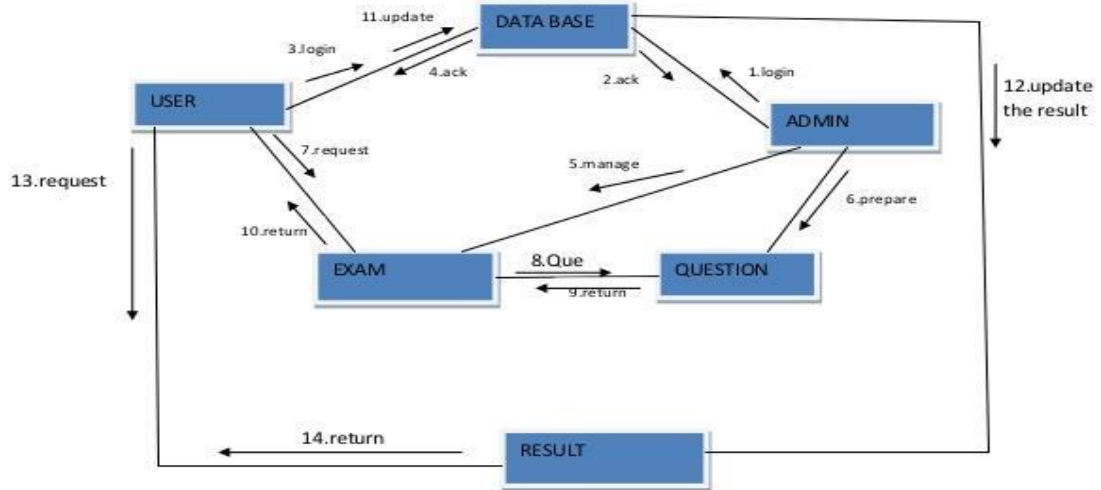


4.Collabroration diagram:

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software **objects** in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular **use case** and define the role of each object. Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.

ONLINE EXAMINATION SYSTEM

COLLABORATION DIAGRAM:



TABLES

Table1:mst_admin

FIELDS	DATA TYPE	DEFAULT	KEY	EXTRA
id	Int(11)	None		AUTO_INCREMENT
loginid	Varchar (50)	None		
pass	Varchar (50)	None		

Table2:mst_question:

FIELDS	DATA TYPE	DEFAULT	KEY	EXTRA
que_id	int(5)	None		AUTO_INCREMENT
test_id	int(5)	NULL		
Que_desc	Varchar (100)	NULL		
ans_1	Varchar (75)	NULL		
ans_2	Varchar (75)	NULL		
ans_3	Varchar (75)	NULL		
ans_4	Varchar (75)	NULL		
true_ans	int(1)	NULL		

Table3:mst_result:

FIELDS	DATA TYPE	DEFAULT	KEY	EXTRA
login	Varchar (5)	NULL		
test_id	int (5)	NULL		
test_date	date	NULL		
score	Int(3)	NULL		

Table4:mst_subject

FIELDS	DATA TYPE	DEFAULT	KEY	EXTRA
sub_id	int(5)	None		
sub_name	Varchar(25)	NULL		

Table5:mst_test

FIELDS	DATA TYPE	DEFAULT	KEY	EXTRA
test_id	int(5)	None		AUTO_INCREMENT
sub_id	int(5)	NULL		
test_name	Varchar (30)	NULL		
total_que	Varchar (15)	NULL		

Table 6:mst_user

FIELDS	DATA TYPE	DEFAULT	KEY	EXTRA
user_id	int(5)	None		AUTO_INCREMENT
Login	Varchar (20)	NULL		
Pass	Varchar (20)	NULL		
username	Varchar (30)	NULL		
address	Varchar (50)	NULL		
city	Varchar (15)	NULL		
Phone	int(10)	NULL		
email	Varchar (30)	NULL		

Table 7:mst_useranswer

FIELDS	DATA TYPE	DEFAULT	KEY	EXTRA
sess_id	Varchar (80)	NULL		
test_id	int(11)	NULL		
que_desc	Varchar (200)	NULL		
ans1	Varchar (50)	NULL		
ans2	Varchar (50)	NULL		
ans3	Varchar (55)	NULL		
ans4	Varchar (50)	NULL		
your_ans	int(11)	NULL		
true_ans	int(11)	NULL		

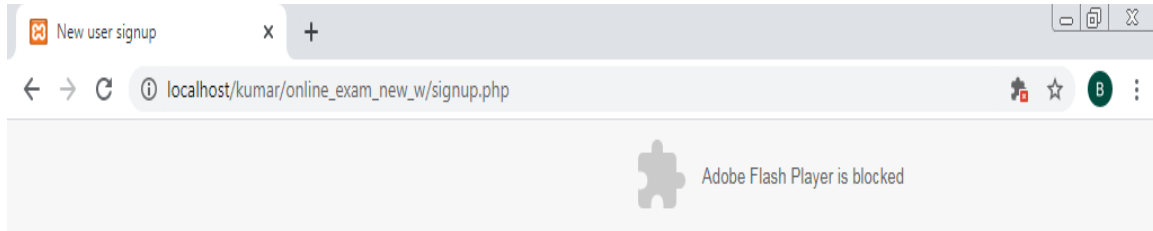
FORMS

Home page:

The screenshot shows a web browser window with the following elements:

- Browser Tab:** Wel come to Online Exam | lisenme.com
- Address Bar:** localhost/kumar/online_exam_new_w/index.php
- Notification:** Adobe Flash Player is blocked
- Main Content:**
 - Heading: **Wel come to Online Quiz Lisenme.com**
 - Image: A graphic showing a laptop, a magnifying glass, and a pencil.
 - Text: **Wel Come lisenme. This Tutorial will provide the Online exam for various subject of interest. You need to login for the take the online exam.**
- Right-Side Form:**
 - Section: **User Login**
 - Fields: **Login ID** and **Password** (both with input boxes)
 - Button: **Login**
 - Link: **New User ? Signup Free**
- Taskbar:** Shows icons for Internet Explorer, File Explorer, Chrome, Firefox, Word, and a custom icon. The system tray shows the time as 2:00 PM on 2/22/2020.

New user signup:

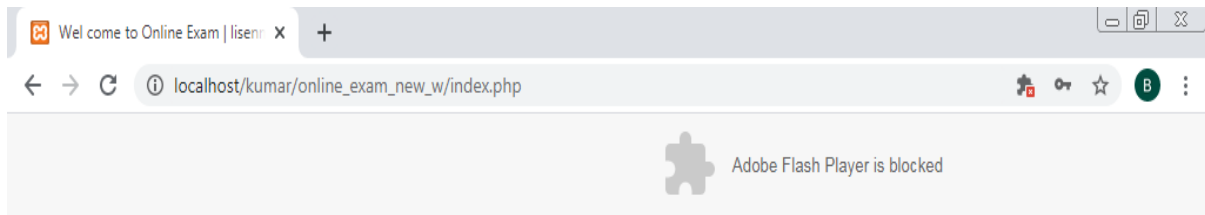


New User Signup


Login Id	<input type="text"/>
Password	<input type="password"/>
Confirm Password	<input type="password"/>
Name	<input type="text"/>
Address	<input type="text"/>
City	<input type="text"/>
Phone	<input type="text"/>
E-mail	<input type="text"/>
	<input type="button" value="Signup"/>



Index page:



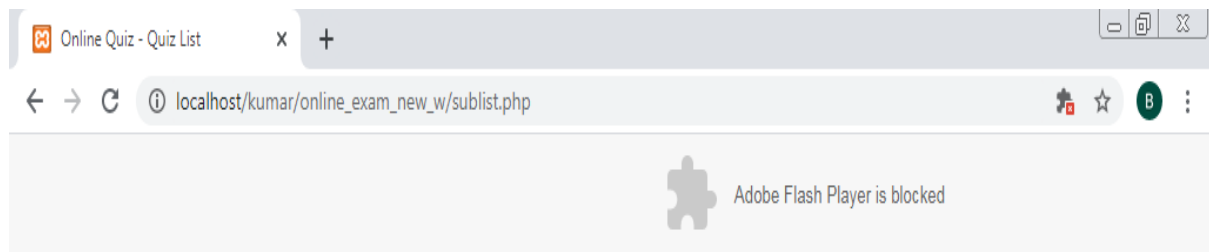
Wel come to Online Exam

 [Subject for Quiz](#)

 [Result](#)



Sublist.php page:



[Home](#) | [Signout](#)

Select Subject to Give Quiz

VB
Oracle
Java
PHP
Computer Fundamental
Networking
mysql



Showtest page form:



[Home](#) | [Signout](#)

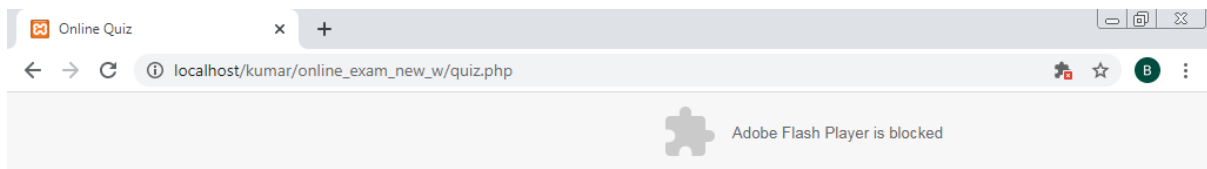
VB

Select Quiz Name to Give Quiz

VB Basic Test
Essentials of VB
Creating User Services



Quiz form:



[Home](#) | [Signout](#)

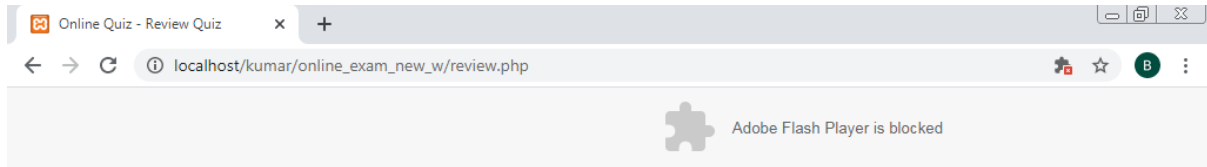
Que 1: What Default Data Type ?

- String
- Variant
- Integer
- Boolear



Result form:

Review page:



[Home](#) | [Signout](#)

Review Test Question

Que 1: What Default Data Type ?

[String](#)

[Variant](#)

[Integer](#)

[Boolear](#)

[Next Question](#)



Coding

Signup code:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>New user signup </title>

<script language="javascript">

function check()

{

if(document.form1.lid.value=="")

{

    alert("Plese Enter Login Id");

        document.form1.lid.focus();

        return false;

}

if(document.form1.pass.value=="")

{

    alert("Plese Enter Your Password");

        document.form1.pass.focus();

        return false;

}

if(document.form1.cpass.value=="")
```

```
{  
    alert("Plese Enter Confirm Password");  
    document.form1.cpass.focus();  
    return false;  
}  
  
if(document.form1.pass.value!=document.form1.cpass.value)  
{  
    alert("Confirm Password does not matched");  
    document.form1.cpass.focus();  
    return false;  
}  
  
if(document.form1.name.value=="")  
{  
    alert("Plese Enter Your Name");  
    document.form1.name.focus();  
    return false;  
}  
  
if(document.form1.address.value=="")  
{  
    alert("Plese Enter Address");  
    document.form1.address.focus();  
    return false;  
}  
  
if(document.form1.city.value=="")  
{  
    alert("Plese Enter City Name");  
    document.form1.city.focus();  
    return false;  
}
```



```
}

if(document.form1.phone.value=="")

{

    alert("Plese Enter Contact No");

        document.form1.phone.focus();

        return false;

}

if(document.form1.email.value=="")

{

    alert("Plese Enter your Email Address");

        document.form1.email.focus();

        return false;

}

e=document.form1.email.value;

        f1=e.indexOf('@');

        f2=e.indexOf('@',f1+1);

        e1=e.indexOf('.');

        e2=e.indexOf('.',e1+1);

        n=e.length;

        if(!(f1>0 && f2==-1 && e1>0 && e2==-1 && f1!=e1+1 && e1!=f1+1 && f1!=n-1 &&

e1!=n-1))

        {

            alert("Please Enter valid Email");

            document.form1.email.focus();

            return false;

        }

return true;

}

</script>
```

```

<link href="quiz.css" rel="stylesheet" type="text/css">

</head>

<body>

<?php

include("header.php");

?>

<table width="100%" border="0">

<tr>

<td width="132" rowspan="2" valign="top"><span class="style8"></span></td>

<td width="468" height="57"><h1 align="center"><span class="style8">New User
Signup</span></h1></td>

</tr>

<tr>

<td><form name="form1" method="post" action="signupuser.php" onSubmit="return
check();">

<table width="301" border="0" align="left">

<tr>

<td><div align="left" class="style7">Login Id </div></td>

<td><input type="text" name="lid"></td>

</tr>

<tr>

<td class="style7">Password</td>

<td><input type="password" name="pass"></td>

</tr>

<tr>

<td class="style7">Confirm Password </td>

<td><input name="cpass" type="password" id="cpass"></td>

</tr>

```

```
<tr>

  <td class="style7">Name</td>

  <td><input name="name" type="text" id="name"></td>

</tr>

<tr>

  <td valign="top" class="style7">Address</td>

  <td><textarea name="address" id="address"></textarea></td>

</tr>

<tr>

  <td valign="top" class="style7">City</td>

  <td><input name="city" type="text" id="city"></td>

</tr>

<tr>

  <td valign="top" class="style7">Phone</td>

  <td><input name="phone" type="text" id="phone"></td>

</tr>

<tr>

  <td valign="top" class="style7">E-mail</td>

  <td><input name="email" type="text" id="email"></td>

</tr>

<tr>

  <td>&nbsp;</td>

  <td><input type="submit" name="Submit" value="Signup">

  </td>

</tr>

</table>

</form></td>

</tr>
```

```
</table>
```

```
<p>&nbsp;</p>
```

```
</body>
```

```
</html>
```

SignupUser.php code:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

```
<html>
```

```
<head>
```

```
<title>User Signup</title>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
```

```
<link href="quiz.css" rel="stylesheet" type="text/css">
```

```
</head>
```

```
<body>
```

```
<?php
```

```
include("header.php");
```

```
extract($_POST);
```

```
include("database.php");
```

```
$rs=mysql_query("select * from mst_user where login='$lid'");
```

```
if (mysql_num_rows($rs)>0)
```

```
{
```

```
    echo "<br><br><br><div class=head1>Login Id Already Exists</div>";
```

```
    exit;
```

```
}
```

```
$query="insert into mst_user(user_id,login,pass,username,address,city,phone,email)
values('$uid','$lid','$pass','$name','$address','$city','$phone','$email)";
```

```
$rs=mysql_query($query)or die("Could Not Perform the Query");
```

```
echo "<br><br><br><div class=head1>Your Login ID $lid Created Sucessfully</div>";
```

```
echo "<br><div class=head1>Please Login using your Login ID to take Quiz</div>";
```

```
echo "<br><div class=head1><a href=index.php>Login</a></div>";

?>

</body>

</html>
```

Index.php code:

```
<?php

session_start();

?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>Wel come to Online Exam | lisenme.com</title>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">

<link href="quiz.css" rel="stylesheet" type="text/css">

</head>

<body>

<?php

include("header.php");

include("database.php");

extract($_POST);

if(isset($submit))

{

    $rs=mysql_query("select * from mst_user where login='$loginid' and pass='$pass'");

    if(mysql_num_rows($rs)<1)

    {

        $found="N";

    }

}
```

```

else
{
    $_SESSION[login]=$loginid;
}
}

if (isset($_SESSION[login]))
{
echo "<h1 class='style8' align=center>Wel come to Online Exam</h1>";

    echo '<table width="28%" border="0" align="center">

<tr>

    <td width="7%" height="65" valign="bottom"></td>

    <td width="93%" valign="bottom" bordercolor="#0000FF"> <a href="sublist.php"
class="style4">Subject for Quiz </a></td>

</tr>

<tr>

    <td height="58" valign="bottom"></td>

    <td valign="bottom"> <a href="result.php" class="style4">Result </a></td>

</tr>

</table>';

exit;

}

?>

<table width="100%" border="0">

<tr>

    <td width="70%" height="25">&nbsp;</td>

    <td width="29%" bgcolor="#1d91d0"><div align="center" class="style1">User Login
</div></td>

```

</tr>

<tr>

<td height="296" valign="top"><div align="center">

<h1 class="style8">Wel come to Online Quiz Lisenme.com</h1>

<param name="movie" value="english theams two brothers.dat">

<param name="quality" value="high">

<param name="movie" value="Drag to a file to choose it.">

<param name="quality" value="high">

<param name="BGCOLOR" value="#FFFFFF">

<p align="left" class="style5"> </p>

<blockquote>

<p align="left" class="style5">Wel Come lisenme. This Tutorial will provide the Online exam for various subject of interest.

You need to login for the take the online exam.</p>

</blockquote>

</div></td>

<td valign="top"><form name="form1" method="post" action="">

<table width="200" border="0">

<tr>

<td>Login ID </td>

<td><input name="loginid" type="text" id="loginid2"></td>

</tr>

<tr>

<td>Password</td>

<td><input name="pass" type="password" id="pass2"></td>

</tr>

```

<tr>

<td colspan="2"><span class="errors">

<?php
    if(isset($found))
    {
        echo "Invalid Username or Password";
    }
?>

</span></td>

</tr>

<tr>

<td colspan=2 align=center class="errors">

    <input name="submit" type="submit" id="submit" value="Login">

</td>

</tr>

<tr>

<td colspan="2" bgcolor="#1d91d0"><div align="center"><span class="style4">New
User ? <a href="signup.php">Signup Free</a></span></div></td>

</tr>

</table>

</form>

</td>

</tr>

</table>

</body>

</html>

```

Sublist.php code:

```

<?php
session_start();

```



```

?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>Online Quiz - Quiz List</title>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">

<link href="quiz.css" rel="stylesheet" type="text/css">

</head>

<body>

<?php

include("header.php");

include("database.php");

echo "<h2 class=head1> Select Subject to Give Quiz </h2>";

$rs=mysql_query("select * from mst_subject");

echo "<table align=center>";

while($row=mysql_fetch_row($rs))

{

    echo "<tr><td align=center ><a href=showtest.php?subid=$row[0]><font
size=4>$row[1]</font></a>";

}

echo "</table>";

?>

</body>

</html>

Showtest.php code:

<?php

session_start();

?>

```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>Online Quiz - Test List</title>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">

<link href="quiz.css" rel="stylesheet" type="text/css">

</head>

<body>

<?php

include("header.php");

include("database.php");

extract($_GET);

$rs1=mysql_query("select * from mst_subject where sub_id=$subid");

$row1=mysql_fetch_array($rs1);

echo "<h1 align=center><font color=blue> $row1[1]</font></h1>";

$rs=mysql_query("select * from mst_test where sub_id=$subid");

if(mysql_num_rows($rs)<1)

{

    echo "<br><br><h2 class=head1> No Quiz for this Subject </h2>";

    exit;

}

echo "<h2 class=head1> Select Quiz Name to Give Quiz </h2>";

echo "<table align=center>";

while($row=mysql_fetch_row($rs))

{

    echo "<tr><td align=center ><a href=quiz.php?testid=$row[0]&subid=$subid><font
size=4>$row[2]</font></a>";
```

```
}  
echo "</table>";  
?>  
</body>  
</html>
```

Quiz.css code:

```
body {  
    margin-left: 0px;  
    margin-top: 0px;  
}  
.style1 {  
    color: #FFFFFFF;  
    font-weight: bold;  
}  
.style2 {  
    color: #990000;  
    font-weight: bold;  
}  
.style4 {  
    color: #000000;  
    font-weight: bold;  
}  
.style5 {color: #990033}  
.style6 {color: #ECE9D8}  
.style7 {  
    color: #993300;  
    font-weight: bold;  
}
```

```
.style8 {  
  
    color: #6633CC;  
  
    font-weight: bold;  
  
}  
  
a:link {  
  
    color: #330066;  
  
    text-decoration: none;  
  
}  
  
a:hover {  
  
    color: #FFCC00;  
  
    text-decoration: underline;  
  
}  
  
a:visited {  
  
    color: #330066;  
  
    text-decoration: none;  
  
}  
  
.head1 {  
  
    font-family: Arial, Helvetica, sans-serif;  
  
    font-size: 24px;  
  
    color: #FF0000;  
  
    text-align: center;  
  
    font-weight: bold;  
  
}  
  
.errors {  
  
    color: #FF0000;  
  
    font-weight: bold;  
  
}
```

```
.tot {  
  
    font-family: Arial, Helvetica, sans-serif;  
  
    font-size: 16px;  
  
    color: #000000;  
  
}
```

```
.tot {  
  
    font-family: Arial, Helvetica, sans-serif;  
  
    font-size: 18px;  
  
    font-weight: bold;  
  
    color: #000000;  
  
}
```

```
.tans {  
  
    font-family: Arial, Helvetica, sans-serif;  
  
    font-size: 18px;  
  
    font-weight: bold;  
  
    color:#00CC66 ;  
  
}
```

```
.fans {  
  
    font-family: Arial, Helvetica, sans-serif;  
  
    font-size: 18px;  
  
    font-weight: bold;  
  
    color:#FF0000 ;  
  
}
```

Quiz.php code:

```
<?php  
  
session_start();  
  
error_reporting(1);
```

```
include("database.php");

extract($_POST);

extract($_GET);

extract($_SESSION);

/*$rs=mysql_query("select * from mst_question where test_id=$tid",$cn) or die(mysql_error());

if($_SESSION[qn]>mysql_num_rows($rs))

{

unset($_SESSION[qn]);

exit;

}*/

if(isset($subid) && isset($testid))

{

$_SESSION[sid]=$subid;

$_SESSION[tid]=$testid;

header("location:quiz.php");

}

if(!isset($_SESSION[sid]) || !isset($_SESSION[tid]))

{

    header("location: index.php");

}

?>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<title>Online Quiz</title>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">

<link href="quiz.css" rel="stylesheet" type="text/css">

</head>
```

```

<body>

<?php

include("header.php");

$query="select * from mst_question";

$rs=mysql_query("select * from mst_question where test_id=$tid",$cn) or die(mysql_error());

if(!isset($_SESSION[qn]))

{

    $_SESSION[qn]=0;

    mysql_query("delete from mst_useranswer where sess_id="" . session_id() .""") or
die(mysql_error());

    $_SESSION[trueans]=0;

}

else

{

    if($submit=='Next Question' && isset($ans))

    {

        mysql_data_seek($rs,$_SESSION[qn]);

        $row= mysql_fetch_row($rs);

        mysql_query("insert into mst_useranswer(sess_id, test_id,
que_des, ans1,ans2,ans3,ans4,true_ans,your_ans) values (".session_id().",
$tid,$row[2],$row[3],$row[4],$row[5], $row[6],$row[7],$ans)") or die(mysql_error());

        if($ans==$row[7])

        {

            $_SESSION[trueans]=$_SESSION[trueans]+1;

        }

        $_SESSION[qn]=$_SESSION[qn]+1;

    }

}

```

```

else if($submit=='Get Result' && isset($ans))
{
    mysql_data_seek($rs,$_SESSION[qn]);

    $row= mysql_fetch_row($rs);

    mysql_query("insert into mst_useranswer(sess_id, test_id,
que_des, ans1,ans2,ans3,ans4,true_ans,your_ans) values (".session_id().",
$tid,$row[2],$row[3],$row[4],$row[5], $row[6],$row[7],$ans)") or die(mysql_error());

    if($ans==$row[7])
    {

$_SESSION[trueans]=$_SESSION[trueans]+1;

    }

    echo "<h1 class=head1> Result</h1>";

    $_SESSION[qn]=$_SESSION[qn]+1;

    echo "<Table align=center><tr class=tot><td>Total Question<td>
$_SESSION[qn]";

    echo "
    <tr
    class=tans><td>True
Answer<td>".$_SESSION[trueans];

    $w=$_SESSION[qn]-$_SESSION[trueans];

    echo "<tr class=fans><td>Wrong Answer<td> ". $w;

    echo "</table>";

    mysql_query("insert into mst_result(login,test_id,test_date,score)
values('$login',$tid, ".date("d/m/Y").", $_SESSION[trueans])") or die(mysql_error());

    echo "
    <h1 align=center><a href=review.php> Review
Question</a> </h1>";

    unset($_SESSION[qn]);

    unset($_SESSION[sid]);

    unset($_SESSION[tid]);

    unset($_SESSION[trueans]);

    exit;

```



```

        }
    }

    $rs=mysql_query("select * from mst_question where test_id=$tid",$cn) or die(mysql_error());

    if($_SESSION[qn]>mysql_num_rows($rs)-1)

    {

        unset($_SESSION[qn]);

        echo "<h1 class=head1>Some Error Occured</h1>";

        session_destroy();

        echo "Please <a href=index.php> Start Again</a>";

        exit;

    }

    mysql_data_seek($rs,$_SESSION[qn]);

    $row= mysql_fetch_row($rs);

    echo "<form name=myfm method=post action=quiz.php>";

    echo "<table width=100%> <tr> <td width=30>&nbsp;<td> <table border=0>";

    $n=$_SESSION[qn]+1;

    echo "<tr><td><span class=style2>Que ". $n .": $row[2]</span>";

    echo "<tr><td class=style8><input type=radio name=ans value=1>$row[3]";

    echo "<tr><td class=style8> <input type=radio name=ans value=2>$row[4]";

    echo "<tr><td class=style8><input type=radio name=ans value=3>$row[5]";

    echo "<tr><td class=style8><input type=radio name=ans value=4>$row[6]";

    if($_SESSION[qn]<mysql_num_rows($rs)-1)

    echo "<tr><td><input type=submit name=submit value='Next Question'></form>";

    else

    echo "<tr><td><input type=submit name=submit value='Get Result'></form>";

    echo "</table></table>";

    ?>

</body>

```

```
</html>
```

Result.php code:

```
<?php
```

```
session_start();
```

```
?>
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

```
<html>
```

```
<head>
```

```
<title>Online Quiz - Result </title>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
```

```
<link href="quiz.css" rel="stylesheet" type="text/css">
```

```
</head>
```

```
<body>
```

```
<?php
```

```
include("header.php");
```

```
include("database.php");
```

```
extract($_SESSION);
```

```
$rs=mysql_query("select t.test_name,t.total_que,r.test_date,r.score from mst_test t, mst_result r  
where
```

```
t.test_id=r.test_id and r.login='$login','$cn) or die(mysql_error());
```

```
echo "<h1 class=head1> Result </h1>";
```

```
if(mysql_num_rows($rs)<1)
```

```
{
```

```
    echo "<br><br><h1 class=head1> You have not given any quiz</h1>";
```

```
    exit;
```

```
}
```

```
echo "<table border=1 align=center><tr class=style2><td width=300>Test Name <td>
Total<br> Question <td> Score";
```

```
while($row=mysql_fetch_row($rs))
```

```
{
```

```
echo "<tr class=style8><td>$row[0] <td align=center> $row[1] <td align=center> $row[3]";
```

```
}
```

```
echo "</table>";
```

```
?>
```

```
</body>
```

```
</html>
```

Review.php code:

```
<?php
```

```
session_start();
```

```
extract($_POST);
```

```
extract($_SESSION);
```

```
include("database.php");
```

```
if($submit=='Finish')
```

```
{
```

```
    mysql_query("delete from mst_useranswer where sess_id=" . session_id() ."'") or
die(mysql_error());
```

```
    unset($_SESSION[qn]);
```

```
    header("Location: index.php");
```

```
    exit;
```

```
}
```

```
?>
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

```
<html>

<head>

<title>Online Quiz - Review Quiz </title>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">

<link href="quiz.css" rel="stylesheet" type="text/css">

</head>

<body>

<?php

include("header.php");

echo "<h1 class=head1> Review Test Question</h1>";

if(!isset($_SESSION[qn]))

{

    $_SESSION[qn]=0;

}

else if($submit=='Next Question' )

{

    $_SESSION[qn]=$_SESSION[qn]+1;

}

$rs=mysql_query("select * from mst_useranswer where sess_id=" . session_id() . "" , $cn) or
die(mysql_error());

mysql_data_seek($rs,$_SESSION[qn]);

$row= mysql_fetch_row($rs);

echo "<form name=myfm method=post action=review.php>";

echo "<table width=100%> <tr> <td width=30>&nbsp;<td <table border=0>";

$n=$_SESSION[qn]+1;
```

```

echo "<tr><td><span class=style2>Que ". $n .": $row[2]</style>";
echo "<tr><td class=\".($row[7]==1?'tans':style8').\">$row[3]";
echo "<tr><td class=\".($row[7]==2?'tans':style8').\">$row[4]";
echo "<tr><td class=\".($row[7]==3?'tans':style8').\">$row[5]";
echo "<tr><td class=\".($row[7]==4?'tans':style8').\">$row[6]";
if($_SESSION[qn]<mysql_num_rows($rs)-1)
echo "<tr><td><input type=submit name=submit value='Next Question'></form>";
else
echo "<tr><td><input type=submit name=submit value='Finish'></form>";
echo "</table></table>";
?>

```

Test.html:

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<style type="text/css">
<!--
-->
</style>
<link href="quiz.css" rel="stylesheet" type="text/css">

<link href="quiz.css" rel="stylesheet" type="text/css">
</head>

```

```
<body>
```

```
<p>&nbsp;</p>
```

```
<table width="28%" border="0" align="center">
```

```
<tr>
```

```
<td width="7%" height="65" valign="bottom"></td>
```

```
<td width="93%" valign="bottom" bordercolor="#0000FF"> <a href="quiz.php" class="style4">Quiz </a></td>
```

```
</tr>
```

```
<tr>
```

```
<td height="58" valign="bottom"></td>
```

```
<td valign="bottom"> <a href="result.php" class="style4">Result </a></td>
```

```
</tr>
```

```
</table>
```

```
<p class="style2"></p>
```

```
<p class="style8"><strong></strong></p>
```

```
</body>
```

```
</html>
```

TESTING

INTRODUCTION:

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding, in fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

A strategy for s/w testing integrates s/w test case design methods into a well planned series of steps that results in the successful construction of software testing is the set of activities that can be planned in advance and conducted as systematically. The underlying motivation of program testing is to affirm software quality with method that can economically and effectively apply to both strategic to both large and small-scale system.

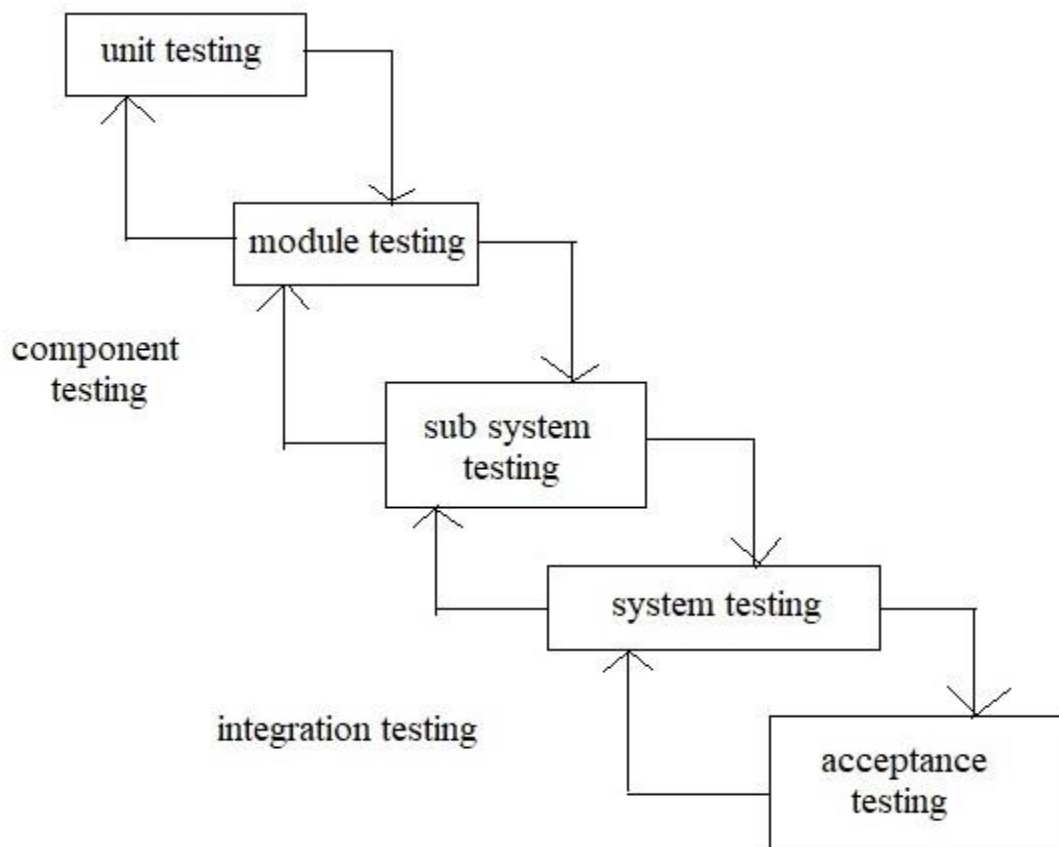
STRATEGIC APPROACH TO SOFTWARE TESTING:

The s/w engineering process can be viewed as a spiral initially system engineering defines the role of s/w and leads to s/w requirement analysis where the information domain, functions, behavior, performance, constraints and validation criteria are established. For s/w are established moving in ward along the spiral we come to design and finally to coding to develop computer software. We spiral in along stream lines that decrease the level of abstraction on each turn.

A strategy for s/w testing may be viewed in the context of the spiral unit testing begins at the vector of the spiral and concentrates on each unit of the software as implemented in source code, testing progress by moving outward along the spiral we encounter validation testing where requirements established as part of software requirements.

Analyses are validated against the s/w that has been constructed finally we arrive at system testing, where the software and other system elements are tested as whole.

TESTING:



UNIT TESTING :

Unit testing focuses verification on the smallest unit of software design, the module. The unit testing we have is white box oriented and some modules the steps are conducted in parallel.

WHITE BOX TESTING :

This type of testing ensures that.

- *All independent paths have been exercised at least once.
- *All logical decisions have been exercised on their true and sides.
- *All loops are executed at their boundaries and within their operational bounds.
- *All internal data structures have been exercise to assure their validity.

To follow the concept of white box testing we have created independently to verify that data flow is correct, All conditions are exercised to check their validity.

All loops are executed on their boundaries.

BASIC PATH TESTING:

Established technique of flow graph with cyclomatic complexity was used to derive test cases for all the functions the main steps in deriving test cases were use the design of the code. And draw correspondent flow graph.

Determine the cyclomatic complexity of resultant flow graph, using formula.

$$V(G) = E - N + 2P$$

$$V(G) = P + 1$$

$$V(G) = \text{Number of regions}$$

Where $V(G)$ is cyclomatic complexity, E is the number of edges.

N is the number of flow graphs nodes, P is the number of predicate nodes.

Determine the basis of set of linearly independent paths.

CONDITIONAL TESTING:

In this part of the testing each of the conditions were tested to both true and false aspects and all the resulting paths were tested. So that each path may be generated on particular condition is traced to uncover any possible errors.

DATA FLOW TESTING:

This type of testing selects that path of the program according to the location of definition and use of variables. This kind of testing was used only when some local variable were declared. The definition use chain method was need in this type of testing. These were particularly use full in nested statements.

BLACK BOX TESTING:

It is a test case design method based on the functional requirements of the s/w it will help a s/w engineer to derive set of input conditions that will exercise all the functional requirements of the program Black-Box testing attempts to find errors in the following

Categories:

Incorrect or missing functions.

Interface errors

Errors in data structures

Performance error

By black box testing we derive a set of test cases that satisfy the following criteria.

Test cases that reduce by count that is greater than one, the number of additional test cases that must be designed to achieve reasonable testing.

Test cases that tell us something about the presence or absence of classes of errors rather than errors associated only with a specific test or hand.

LOOP TESTING:

In this type of testing all the loops are tested all the limits possible. The following exercise was adopted for all loops.

*All the loops were tested at their limits, just above them and just below them..

*For nested loops the inner most loop first and then work outwards.

*For concatenated loops the values of dependent loops were set with the help of connected loop.

*Unstructured loops were resolved into nested loops or concatenated loops and tested as above.

CONCLUSION and FUTURE SCOPE

The "Online Examination System" has been successfully completed. The goal of the system is achieved and problems are solved. Final reports are generated as per the specification of the client. The package is developed in a manner that it is user friendly and required help is provided at different levels. The Online Examination project can be easily used in the process of decision making.

To modify the Online Examination System project to .Net platform to take the advantage of geographical remote area. By shifting the project to the Dot Net platform the project can be made into a Mobile Accessible Application by which the restrictions of the software & hardware requirements can be scaled down, which is not possible using ASP.

2) we can even apply the unique identity of the user by keeping certain formalities which must be answered by the user while entering the test (like the user need to answer again the questions which he did while registering himself for the test) the questions selected will be in random order from his personal data only (like the mole present on the user, etc ...) so that we it may confirm that the user itself is writing the test .

3) we can even use the thumb mark of the users to confirm their identities .

4) We can even add the photos of the users in this to confirm the user Identity.

Thus by applying these steps we can make sure that the user itself is writing the exam not the others.

A Project report on
VETERINARY INFORMATION MANAGEMENT SYSTEM

Submitted in partial fulfillment of the requirements

For the award of the Degree of

Bachelor of Science

IN

COMPUTER SCIENCE

Submitted By

D. S Lalitha aparna(2173042)

Under the Esteemed Guidance of

Kum.V.LALITHA, M.B.A.

Lecturer in Computer Science



U.G. DEPARTMENT OF COMPUTER SCIENCE

VSM COLLEGE (A)

(Recognized by Adikavi Nannaya University, Rajamendravaram)

Ramachandrapuram -533255, E.G.Dist (A.P).

2017-2020

VSM COLLEGE (A)

(Recognized by Adikavi Nannaya University, Rajamendravaram)

Ramachandrapuram -533255, E.G.Dist (A.P).

U.G. DEPARTMENT OF COMPUTER SCIENCE

CERTIFICATE



This is to certify that the project entitled “**VETERINARY INFORMATION MANAGEMENT SYSTEM**” that is being submitted by

D.ADINARAYANA (2173039)

D.S.LALITHA APARNA (2173042)

CH.INDHIRA (21730035)

CH. VENKATESH (2173034)

A.KASI VISWANADH (2173007)

In Partial Fulfillment the requirements for the award of the Degree of **Bachelor of Science** in “**Computer Science**” in the academic year 2017-20 to **VSM College(A)** affiliated to **Adikavi Nannaya University, Rajamahendravaram**, is a record of Benefited work carried out by him/her under my guidance and supervision.

The results embodied in this project have not been submitted to any other University or Institute for the award of any degree or diploma.

INTERNAL GUIDE
(Kum .V. LALITHA M.B.A.)

PROJECT CO-ORDINATOR
(Sri.M Phani Kumar M.C.A.)

HEAD OF THE DEPARTMENT
(Sri.MVV Chowdary M.C.A., M.Tech., M.B.A.)

EXTERNAL EXAMINER

ABSTRACT

A Prerequisite to the development of an efficient Animal health, food safety and traceability management system in the Animal food production chain is the implementation of an integrated veterinary informational management system (VIMS) capable for the capture, storage ,analysis and retrieval of data and providing the opportunity for the commutative gathering of the knowledge and capability for it interpretation.

A Veterinary management system is developed to aid in dispensary animals and their owners data management and viewing the system is aimed to help doctors to get register as well as view owner and their animal history. Our system is a standalone system that can be installed on doctor, owners system to be installed on doctor used for further login on installation. The application allows owner to open application allows doctor to insert various data fields regarding a to doctor details, availability details etc.. The application allows owner to insert various data fields regarding to owner registration animal registration etc. The system saves this doctor, owner animals related data in the android phone they may view this data as and when needed then they may check the details whenever needed. The application allows doctor to search owner and their animals by their id.

ACKNOWLEDGEMENTS

I would like to take the privilege of the opportunity to express my gratitude into Project work of "*ONLINE MARKS POSTAGE SYSTEM*" enabled us to express our special thanks to our Principal **Sri.NSV Kiran Kumar** who has shown keen interest in us and encouraged us by providing all the facilities to complete my project successfully.

I owe my gratitude to our beloved Head of the Department Computer Science **Sri. MVV Chowdary** for assisting us in completing our project work.

I express my sincere thanks to my internal guide "**Kum.V.LALITHA**" who has been a source of inspiration for us throughout my project and for her valuable advices in making my project a success.

I wish to express my sincere thanks to all teaching and non-teaching staff of Computer Science Department. I wish to express my special thanks to all the Faculty members of our College for their concern in subjects and their help throughout my course.

I am very thankful to all my friends who had given me good co-operation and suggestions throughout this project and helped me in successful completion.

D.ADINARAYANA (2173039)

D.S.LALITHA APARNA (2173042)

CH.INDHIRA (21730035)

CH. VENKATESH (2173034)

A.KASI VISWANADH (2173007)

CONTENTS

S. No	TOPIC	P.G.No
1	INTRODUCTION 1.1. PURPOSE 1.2. SCOPE 1.3. TECHNOLOGY TO BE USED 1.4. OVER VIEW	1 - 6
2	OVERALL DESCRIPTION 2.1 GOALS OF PROPOSED SYSTEM 2.2 EXISTING SYSTEM FEATURES 2.3 METHODOLOGY TO BE USED 2.4 MODULE DESCRIPTION	7 - 13
3	SRS DOCUMENT 3.1 INTRODUCTION 3.2 OVERALL DESCRIPTION 3.3 SPECIFICATION REQUIREMENTS 3.4 CHANGE MANAGEMENT PROCESS	14 - 19
4	FEASABILITY STUDY	20 - 24
5	DESIGN PHASE	25 - 30
6	CODING	31 - 54
7	FORMS	55 - 59
8	REPORTS	60 - 63
9	TESTING	64 - 75
10	CONCLUSION	76
11	REFERENCES	77

INTRODUCTION

Introduction:

Animal health, Animal welfare and safety have been recognized worldwide as a major issues that helps to project the health of consumers but that also may create international trade barriers for live animals, product of animal origin, animal genetic material and biological products. In the face of this complexity of the food supply, the application of up to data information system (IS) are becoming more and more important food production and trade of animals and food of animal origin. According to the literature data and preliminary research. In this world we want to shortly elaborate the new VIMS computerized module for notification of animal diseases, designed and constructed in 2005.

Purpose:

The program provides essential knowledge for effective decision making and on-the - job performance with a series of courses designed to increase the overall managerial effectiveness of veterinary professional veterinarians and practice management will gain confidence and competence in utilizing long-term strategic thinking to plan for the future. It is an opportunity to study the real world business issues, in a stimulating, interactive academic setting focus on the realities of working in the veterinary medical profession. Attend develop a better understanding of veterinary business by develop exploring the problems and opportunities in a rapidly changing industry.

A Hospital Management System not only provides an opportunity to the hospital to enhance their patient care but can also increase the profitability of the organization. These kinds of application enable hospitals and doctors to serve their patients better. Computers are being extensively used for routine administrative purposes by health care professionals. In veterinary medicine, hospital management software is designed for all types of practices, ranging for small, single-vet ones to large ones with multiple locations and vets, and covers a wide range of hospital administration and management processes. It is an integrated end-to-end management system that provides relevant information across the hospital to support effective decision making for patient care, administration and critical financial accounting in a seamless flow. Specific areas of application include scheduling of appointments, registration of cases including client and patient data, case history and electronic medical records, health care reports, reminders, inventory controls, reports and accounting.

Scope:

1. We help organization conduct studies and implement development projects in the field of animal health care, animal welfare, livestock development food safety and public health
2. During 2015, USA reported to world organization for animal health(OIE) a totalp of 8709 veterinarians out of which 75593186.9% are private veterinarians.
3. As per federation of veterinarians in Europe (FVE) survey of veterinary profession in Europe 2015 there are 24300 veterinarians in Europe.
4. A clear majority (60%) of those veterinarians , work in mostly private clinical practice which is predominantly of the companion animal

5. This document is the only that describes the requirements of the system. It is meant for use by the developer and will also be the basis for validating the final delivered system.

The developer is responsible for asking for clarifications, where necessary and will not make any alterations with the permission of the client.

Features of Leave Management system:

The veterinary management system project has been divided into five modules. They are :

- Administration module
- Doctor registration module
- Owner and their animals registration module
- Appointment module
- Search module.

Technologies to be used:-

Software requirements:

Platform- windows (2000/XP)/unix, Solaris

Software- HTML, PHP, CSS.

Hardware requirements:

Processor- Intel Celeron class processor with 2.0 GHZ

Ram- 256 mb.

Harddisk-40GB

Keyboard- Keys

Mouse- any pointing device.

Front-end-PHP

Back-end-MYSQL

Front-end-PHP:-

Hypertext Preprocessor is a general-purpose programming language origin all designed for web development. It was originally created by RASMUS LERDORF in 1994. The PHP references implementation is now produced by the PHP Group

- Scalability and extensibility
- Flexibility
- Robustness
- According to the organization requirements and the culture

- Must provide excellent reporting features with good profiling support
- Platform independent
- Easy to debug and maintain
- Event driven programming facility
- Front and must support some popular back and like MS access.

Back-end-MYSQL:-

MYSQL is an open source relational database management system. Its name is a combination of "MY "the name of co-founder MICHAEL WIDENIUS 'S daughter, and "SQL" there for standard Query language.

- Multiple user support
- Efficient data handling
- Provide inherent features for security
- Efficient data retrieval and maintenance
- Stored procedures
- Popularity
- Operating system compatible

Overview:-

The need of professional advancement the face of accelerating change has long been an accepted fact of the medical/scientific community .In this economy veterinary .professional will face in increasingly competitive business environment brought on by new and expensive advances in surgical equipment, medical treatments, computer and communication technology, and the progression of corporate ownership, practice consolidations, and pet superstores. As a consequence, veterinarians who lack a broad-based understanding of business will find the economic viability of their practices challenge.

To help meet those challenges the veterinary practice management program has been developed by Purdue's KRANERT School Management and college of veterinary medicine. It is an intensive four module, management education program offering graduate level instruction on core management issues. The program is held at Purdue University's west.

OVERALL DESCRIPTION:-

Goals of the proposed system:-

By developing the system we can attain the following features. Easy to handle and feasible cost reduction fast and convenient. The administrator can insert, delete and modify the patient records as per the requirements. The laboratory reports and building details can be easily maintained in the system this project will help to smoothen the process of the hospital activities.

- The new proposed system stores and maintains all the employees details
- The most important benefit of this system is that it reduces the redundancy of data within the data
- It is a broad statement that defines what you plan to do in a project. It gives an idea to the reader of what problem your organization intends to address.
- To save the time
- To reduce paper and file work
- To speed up the procedure
- To relieve the management from repetitive work.

Existing system:-

Existing system is a manual system where animal owner need to go to hospitals and take appointment for their animals and they don't know about all the vet doctor near to their areas.

In Existing system if they medicine about disease name, but they don't know what medicine to take for that information they have consult doctor for that need to go for doctor so it is time taking process and it is paper base system.

Advantages:-

- The current manual system has a lot of the paper work, and it does not deal with old new car purchase and sales
- Within the increase in database, it will become a massive job to maintain the database.
- The existing system needs to be changed.
- The demolition of existing buildings to make way of new office blocks
- It is change in errors.

Disadvantages:-

- Time taking process and if doctor not available waste for travelling time and waiting time.
- Paper records may be lost, no data recovery facility.
- Lack of integrity

- Lack of availability and continuity of service
- Lack of accuracy
- No control by an international civil body
- **Proposed system:-**

System provides information about nearest vet doctors information and animal owner can take appointment from anywhere where they want to and check availability of vet doctors, system generates districts wise disease and medicine reports about animals.

Our application is simple, secure and smart .our application is quite useful which helps for both doctors and animal owners. Owners can check various animal diseases and their medicines (or) treatments.

Advantages of proposed system:-

- Owner can get anywhere from register so it is time squaring process.
- A part from reduction in storage costs data outsourcing to the cloud also helps in reducing the maintenance.
- They can check doctor availability if not available they want come to hospital so no travelling cost and travelling time.
- Avoiding local storage of data.
- By reducing the costs of storage, maintenance, and personal.
- It reduces the chance of losing data by hardware failures.

Solutions of this problems:-

- Owners can get register from anywhere so it is time saving process.
- They can check doctors Availability if not available they won't come to hospital so no travelling cost and travelling time.
- This application provides a platform for animal owners to know about the kind of animal disease and medicines information.
- Understand everyone's interests.
- List the possible Solution.
- Evaluate the options.
- Document the agreements.
- Agree on contingences, monitoring, and evaluation.
- Identify the issues. Be clear about what the problem.

- Select any option or options.

General Methodology in developing s/w project-

The general methodology in developing a system is involved in different phases which describe the systems life cycle model for developing software project the concept including not only forward motion but also have the possibility to return that is cycle back to an activity previously completed.

This cycle back or feedback may occur as a result of the failure with the system to meet a performance objective or a result of changes in redefinition of the system activities like most system the life cycle of the computer based system also exhibits distinct phases. They are:

1. Requirement analysis phase
2. Design phase
3. Development phase
4. Coding phase
5. Testing phase
6. Design constraints
7. System analysis

1. Requirement analysis phase:-

This phase includes the identification of the problem in order to identify the problems, we have to know that information about the problem, the purpose of the evaluation for problem to be known we have to clearly know about the silently requirements and the objectives of the purpose.

The Requirement analysis phase begin when the previous phase objectives have been achieved. Documentation related to user requirements from the concept Development phase and the planning phase shall be used as the basis for further user needs analysis and the development of detailed requirements. Multiple_ release projects require only one iteration of the Requirement Analysis phase , which should involve requirements definition for all planned releases.

2. Design phase:-

S/W design is a process through which the requirements are translated into a representation of s/w one of the software requirements has been analyzed and specified the s/w design involves three technical activities design, coding generation and testing. The design of the system is in modulator form i.e. ., the s/w is logically partitioned into components that perform specific functions and sub functions.

The design phase leads to modules that exhibit independent functional characteristics .It even leads to interface that reduce the complexity of the connections between modules and with the external environment. The design phase is of main importance because in the activity decisions alternately after the success of s/w implementation and maintain

3 .Development phase:-

The development phase includes choosing of a suitable s/w to solve the particular problem given the various facilities problem and the sophistication in the selected s/w given a better development of the problem.

The development phase that a company goes through during the preliminary stage of its corporate life. Companies that are in this stage are characterized by their focus on early-stage business activities. Development phase companies are generally underfunded and likely to be on the lookout for sources of capital.

4. Coding phase:-

The coding phase is for similar. The coding phase is for similar the design of the system produced during the design phase into code in a given programming language which can be executed by a computer and which performs the computation specified by the design.

The coding phase is the software engineering paradigm is usually defined after the designing phase. In this phase the developers or the coders have to implement the software design practically using any computer languages so that the software can be created and the user can use it this is the main step to be performed and to ensure that this step is performed efficiently and is prone to minimum errors the rest of the steps included in the software lifecycle model .

5. Testing phase:-

Testing is done in various ways such as testing the algorithm programming code, sample data beginning is also one of the following the above testing.

The testing phase of the software development lifecycle (SDLC) is where you focus on investigation and discovery. During this testing phase, developers find out whether their code and programming work according to customer requirements. Before testing can begin the project team develops a test plan.

6. Design constraints:-

The veterinary management system requires huge resources as hundreds of the animals the services instantly, quick response time and needed the database should also be very large and to maintain very huge animals and drugs data.

A design constraint refers to a limitation on the requirements and under which a robot is expected to operate. A design constraint can, for example affect the robot shape the robot operation features and the robot functionality.

7. System analysis:-

Analysis is the detailed study of the various operations performed by a system and their relationships within and outside of the system .A key questions is what must be done to solve the problem ?are aspect of analysis is defining the boundaries of the system should consider other related system. During analysis, data are collected on the available files, decision points and transactions handled by the present system.

Its a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. It is a problem solving technique that improves the system ensures the at all the components of the system work efficiently to accomplish their purpose.



Spiral Model

Spiral model is a risk driven model generator for software projects based on the unique risk pattern of a given project spiral model guides a team to adopt elements of one (or) more process models.

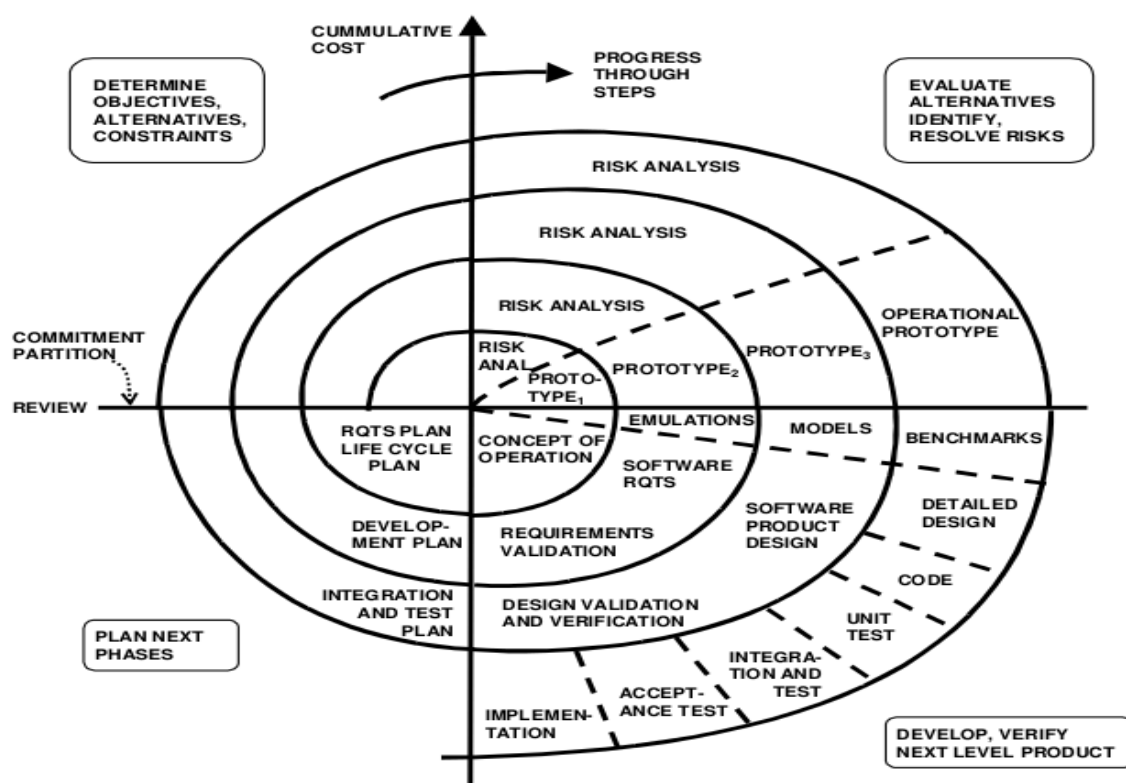
The spiral model combines the idea of iterative development with the systematic controller of the waterfall model spiral model is not so well known as other SDLC(Software development life cycle)model .for example and here's the reason spiral model can be pretty, costly to use and does not work well for small projects.

It is a risk driven model which means that the overall success of a project highly depends on the risk analysis requires specific expert on every iteration thus, the review and the project for time, special skills and needed.

At a first sight, if may like model is complicated and clumsy and these are no reasons to consider this approach as one of your options. But like any other SDLC model this once besides its disadvantages has its unique strong sides for example, there is a possibility to add some additional functionality at the last stages of product development, since risk maintaining and regular expertise are core characteristics of this approach the overall project becomes more transparent.

In a new words, spiral model can be characterize by repeatedly a set of elemental development process and eliminating risk. So it is actively being reduced.

To understand how you can get your goals using spiral model. Let us take a look at this diagram



STEPS FOLLOWED IN SPIRALMODEL:-

1. Requirement collection
2. Designing
3. Development
4. Testing

Requirement Collection : Requirement collection is the first step of spiral model .In this step requirements collected from the silent side as per module for designing. It is collected requirements in the form and convert into SRS (Software requirement specification)

It is a collection the process is not as straightforward as just asking the stakeholders what they want they system to do, as in many cases they are not aware of all the possibilities that exist, and may be limited by their immersion in the current state. For example asking people in the 9th century for their requirements for a self – propelled vehicle would have just resulted in the specification for a faster horse- drawn carriage rather than an automobile. Beware the old and age, “it’s everything I asked for, but not what I need”!

Designing: In designing step designs are prepared by the senior developer on this basis of requirement designs are will be changed as per requirements. Design is a visual look or a shape given to a certain object, in order to make it more attractive, make it more comfortable or to improve another characteristic. Designers use tools from geometry

Try and art... .Design is also a concept used to create an object.

Development In this step, development starts as per designs or requirements. Development is done by develop modules by module .After developers development feature module of the product , then only a developers can go on to develop the next module of the product .process of creating something over a period of time: the state of being created or made more advanced.

Testing: Testing performed for new modules features first .After that newly added features or module dependency will be tested with other feature or modules. In general testing is finding out how well something works in terms of human beings, testing tells what level of knowledge or skills has been acquired in computer hardware and software development testing, is used at key checkpoints in the overall process to determine whether objectives are being met. For example in software development product objectives are sometimes tested by product user representatives when the design is complete coding follows and the finished co

de is then tested at the unit or module level by each programmer, at the component level by the group of programmers involved and at the system level when all components are combined together At early or late stages a product or service may also be tested for usability.

Coding: The coding phase is for translating the design of the system produced during the design phase into code in a programming language which can be executed by the design.

It is the process of using a programming language to get a computer to behave how you want it to. Every line of code tells the computer to do something and a document full of lines of code is called a

script. People who make code are called programmers, coders, developers they all work with computers to create websites apps and even games today you will learn what code it what it is for and how to start learning code yourself.

Design constraints:-The veterinary management system requires .huge resources as hundreds of the animals the services instantly, quick response time should also very large.

A design constraints refers to limitation on the requirements and or operation conditions under which a robot is expected to operate a design constraints can. For example affect the robot shape, robot operation features and the robot functionality.

System analysis: Analysis is the detailed study of the various operations performed by a system an their relationship within and outside of the system. It is the process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components it is a problem solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose.

Advantages of Spiral Model:

- High amount of risk analysis hence, avoidance of Risk is enhanced.
- Good for large and mission – critical projects.
- Strong approval and documentation control.
- Addition functionality can be added at a later date.
- Software is produced early in the software life cycle.

Disadvantages of Spiral Model:

- Can be a costly model to use
- Risk analysis requires highly specific expertise.
- Doesn't work well for smaller projects.
- It is highly dependent on the risk analysis phase.

In this project with the help of veterinary doctors and following module:-

Modules:-

1. Administration Module
- 2 .Doctor Registration Module
- 3 Owner and their Animal Registration Module
4. Appointment Module
5. Search Module

1 .Administration Module:-

In this Module the admin Maintain the complete details and information about doctors, owners and their animals. It can have access update, delete.

The administration module is the administrators interface and allows to process all configuration operations of the system. Configuring the data model of the CMDB, in terms of classes, attributes, relations, including the definition of certain behaviors of the user interface

2 Doctors Registration Module:-

In this Module the Doctor can have Register themselves via this Module and can make their account here. Owner details related the account of module. In order to inform all the patients same information is important.

The doctor registration module provides a conceptual framework for entering data on those patients in a way that eases data entry & accuracy by matching the open MRS entry to the data source ties easily back to individual patient records to connect registers to patient data, and collects data elements to enable better supervision of treatment programs

3. Owners and their animal registration module:-

In this Module Owners can Registers their animals .After checking the animal type are in this. They give basic information about animals after getting Registration. By this id owners or doctors can search for animals.

Here we can register the new animal owners for their animals during registration we can enter the basic information regarding owners.

4. Appointment module:-

In this Module .First owners can check for doctor availability. If we doctor trimmings are availability they can fix appointment Id.

Our appointment module is computerized which allows us to efficiency make appointments for you and your pet. Our receptionists and team will attempt to accommodate all requests to the best of our ability. If you feel you have an emergency with your pet during business hours, please call us before coming in so that a staff member can advise you on particular emergency.

5. Search Module:-

In this module the application allows doctors to search animals and their owners to search required doctors .they can search about their Appointments.

The search module search for specific content on your site you can search both for users and for particular words when you are on the "users" tab of search you will be able to search the user names of registered users on your site, and if you have sufficient permissions also their email address.

Non- Functional Requirements:

Usability:

In this case of use and learns ability of humans made object the object of use can software application, website, book, tool, machine, process or anything a human interacts with a usability study may be conducted as primary job function by a usability analyst or as a secondary job function by designers, technical writers, marketing personal and the others.

Reliability:

Accomplishment of a given task measured against present standards of accuracy, completeness, cost and the speed.

Supportability:

To which the destine characteristics of a stand by or support system meet by the operational requirements of an organization.

Implementation:

Implementation is the realization of an application or execution of a plan, idea, model, design, specification, standard, algorithm, or policy.

Interface:

An interface refers to a point of interaction between components and is applicable at the level of both hardware and software this allows a component weather a piece of hardware such as graphic cards such as an internet browser to function independently.

Legal:

It is established by or founded upon law or official or accepted rule of or relating to jurisprudence of legal loophole having legal efficacy or characteristic of the profession of law the legal profession allowed by official rules a legal pass receiver.

3. SRS DOCUMENTATION

SOFTWARE REQUIREMENT SPECIFICATION (SRS):

A software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform. An SRS describes the functionality the product needs of fulfill all stake holders(business, users)needs .A typical SRS includes. A purpose an overall description. An SRS document which serves as a written contract between silent and organized on. The surface requirement specification establishes. The basis for an agreement between customers and contracts how the software product should function It also provide a realistic basis for estimating product costs , risks and schedules. SRS can help prevent software project failure.

The SRS is developed based the agreement between customer and contractors. It may include the use cases of how user is going to interact with software system. The software requirement Specification document consistent of all necessary requirements required for project development. To develop the software system we should have clear understanding of software system. To achieve this we need to continuous communication with customers together all requirements.

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

It is highly recommended to review or test SRS documents before start writing test cases and making any plan for testing . Let see how to test SRS and the important point to keep in mind while testing it.

- Correctness of SRS should be checked
- Ambiguity should be avoided
- Requirements should be complex
- Consistent requirements
- Verification of expected result
- Testing environment
- Requirements ID
- Assumption should be avoided

1. Product respective for Animals:-

Animal by -products (ABPs) are defined as entire bodies or parts of animals, products of animal origin or other products obtained from animals which are no intended for human consumption. Animal products are a potential source of risks to public and animal health. Legislation has been in place form a years to control these risks by setting out the requirements for a

storage transport, use and disposal of Animal by product

Animal by – products are carcasses and parts of carcasses from slaughter house, animal shelters, zoos and veterinarians, and products of animal origin not intended for human consumption, including catering waste. These products may go through a process known as “rendering” to be made into human and nonhuman foodstuffs, fats, and other material that can be sold to make commercial products such as cosmetics, paint, cleaners, polishes, glue, soap and ink. The sale of animal by- products allows the meat industry to compete economically with industries selling sources of vegetable protein.

2. Product function of Animals:-

An animal product is any material derived from the body of an animal .Examples are fat, flesh ,blood, milk, eggs, and lesser known products ,such as is in glass and rennet. Animal by- product. It is defined by the user USDA , are products harvested or manufactured from livestock other than muscle meat. Several diets prohibit the inclusion of some animal products , including vegetarian, kosher, and other balanced diet, such as veganism and the row exclude any material of animal origin. The word animals includes all species in the biological kingdoms. For Example, insects, and animals.

The left over pieces that come from the process of stripping meat from animals tends to get used for different purpose one of them is to put these parts into pet food. Many large, well- known pet food brands use animal by- products as protein sources in their recipes. This can include animal feet, livers, lungs, heads etc...

3. Characteristics of veterinary management:-

It is uses of pets, animals, farmers, Advice animal owners about general care , medical, conditions, and treatments , veterinarians, in private clinical practices treat the injuries of pets and other animals with a variety of medical equipment , including surgical tools and x-ray and ultrasound machines. They provided treatment for animal that is similar to the services a physician provides to that humans.

The veterinary profession is one of the most interesting, calling and diverse occupations in which one can pursue a career. It difficult, its challenging, and its filled with both happy and sad moments, stressful situations, mistakes, failures and pleasant success. There are thousands of successful veterinarians that managed to climb to the top of the ladder only with hard work and a lot of knowledge of course. There are 8 characteristics of the Veterinary :

- Detailed knowledge
- Skills for making quick decisions
- Works with private life
- Healthy living
- Passionate and compassionate

- Perfect manual dexterity
- Communication skills
- Business skills

4. Constraints on animal development-

It is a given tax development generally results in the production of individuals that are recognizable as members of that specific group.

* Vertebrate embryo can be recognized as such early in development, regardless of whether they will later become fish, birds, or animals.

* Regardless of the direction and magnitude of external selective pressure, it may be impossible for the organism to change because of internal constraints.

* Developmental constraints are any aspects of a developmental system that increase the probability of a particular outcomes and limit the production of variable phenotypes.

* Humans are nearly always have the same number of fingers and toes at birth and, although there are slight variations in the size shape of digits and limbs from one individual to the next.

* They are recognizable as human features for example, wings, hooves, and fins never appear in humans.

* Normal development seems to follow the same pathway, and resulting variation is limited.

5. Assumption & dependence

- **Assumption:-**

While developing the prerequisite website for veterinary management system, following Assumption were kept in mind for the administration. The user logged in with ID and Password is a valid user. An assumption is something that is believed to be true. It's something that you can expect to happen during a project. However, that certainty isn't supported by factual proof, it comes from experience. Just like dependencies and constraints, assumptions are events that are outside of the project manager's and teams control. But unlike constraints, which put restrictions on a project and can pose a danger to its successful completion. Assumptions open possibilities for it and make it possible for the project to finish successfully. This is how you can differentiate assumptions from constraint and dependencies.

- An assumption is something that you assume to be in case, even without proof. For example people might make the assumption that you're heard if you wear glasses, even though that's not true or very nice.
- Assumes uniform density and charge of background solves.
- If the geometry of the solution becomes non-spherical.
- Lacks angular and dihedrals polarizations
- Experimental samples many impurities while the model pristine.

- Requires material and experiment setup.
- It is under predict the solvation.
- The assumptions used in the development of IGM are listed also.

Managing project assumptions in 5 steps

1. Define:-

You will realize that you will have to make a lot of assumptions during the course of a project. There are some assumptions you definitely need

1. Human resource Availability : All key project team members are available and have the necessary skills and knowledge to work on the project. The access to the right resources for a given project at a given time with the communication skills .

2. Budget Availability : The determined budget is accurate and covers all project expenses. It is always at any time the amount pursuant to section. The next Revolving Loan, based on the most recently delivered budget, plus, to the extent available, the excess variance amount.

3. Scheduling accuracy: The set deadlines and milestones are achievable and the project can be finished on time. The scheduling accuracy is a comparison between the forecast that you are trying to satisfy in each interval and the number of staff that end up being planned for each interval. In theory, a good fit between schedule and forecast makes your call center more prepared to service customers.

4. Performance of contractors:-

Suppliers and vendors: All necessities and equipment goods are available when you need them. It is the fulfillment of the contractual obligations by the parties. It is one of the methods to discharge a contract. The parties have no further rights and liabilities once the contract is discharged.

5. Upper management support-You have the support and buy -in from the C -Level and the project sponsor, who will back you up when issues arise. It is defined as devoting time to the IS program in proportion to its cost and potential, reviewing plans, following up on. Results and facilitating the management problems involved.

2. Assessment

Make a list of all project assumptions and prioritize them. Which assumptions are almost 100% certain occur, and which are less certain? The whole project team should be involved in this step as they are the ones who actively work on the project and can contribute their combined experiences, this will allow for more accurate assumptions. There are two factors on which you can assess an assumption.

1. The probability of its occurrence

2. It is possible impact on the project

Assessment is the process of gathering and discussing information from multiple and diverse sources in order to develop a deep understanding of what students know, understand, and

can do with their knowledge as a result of their educational experiences, the process culminates when assessment results are used to improve.

3. Document

Write the key assumptions down in the project initiation document ,along with the project dependencies and constraints .This will help you keep an overview of all aspects that might restrict your projects and also help you identify all the possibilities that enable you to deliver the project on schedule and on budget.

A document is a form of information. A document can be put into an electronic form and stored in a computer as one or more files. Often a single document becomes a single file. An entire document or individual parts may be part of a database.

4. Monitor:

Documenting assumptions also enables you to monitor and control them better .Assumptions are often true ,but that doesn't mean that they can't turn out to be false during the course of the project .Remember that assumption are not facts. You need to constantly track and monitor assumptions .A tone point or another in the project's life cycle, you will be able to prove whether an assumption was true or false. Include the regular monitoring of assumptions in your project plan, e.g. schedule dates on which you will test whether your assumption was right.

Monitors are connected to the computer via VGA Digital visual interface (DVI) , HDMI, Display port, Thunderbolt, low voltage differential signaling (LVDS).

- Within the constraints of your budget and desk space, get the largest monitor.
- The purpose of monitor is to display text graphics produced by computer programs and to provide a visual interface for computer users.
- Viewing angle
- Brightness or luminance
- Contrast ratio
- Response rate
- Adjustability

5. Review:

As assumptions are based on experiences , it's vital that you review them at the end of the project .Which assumptions proved to be true and had the biggest impact on the project's out come? Which turned out to be false and had to be dismissed ? Answering these questions will help you make more accurate assumptions in future projects.

A review is an evaluation of a publication, service, or company such as a movie, videogame, musical composition, book, a piece of hardware, like a car, home appliance, or computer, or an event or performance, Often evaluates the work in question based on its strong and weak points, something ending with a recommendation.



EXTERNAL INTERFACE REQUIREMENTS

1 .User Interfaces:-

The user interface requirements for the VLTSW are dictated by [AD04] and 4.1.7 and described in their implementation in [AD08]. The ATCS shall in herewith them completely .Therefore, as already stated at the end of the General Requirements-control Software of this document ,the explicit requirement for the ATCS is to use the CCS / User Interface to create all the graphical panels , which will interact with the observer at the level of the ISS /User Interface .Since the user interface environment shall be organized during the development phase as a flexible ,easy -to-update ,dynamically configurable system, which access mainly value stored in the on-line database ,it is not necessary (if not worthless) to define the layout and hierarchy of panels at requirement level .Discussions with potential users and preliminary tests on the field will suggest how functions have to be grouped and how data have to be displayed to achieve the best ergonomics .

The graphical user interface will rely on the ESO Graphical UIF Common Conventions [AD08].For the implementation of the graphical user interface framework [AD08] the current standard will be used (TCI/TK [RDV03])

2. Hardware Interfaces:-

The hardware interfaces shall obey to the two statements written in [AD04], § 5.2, and repeated here for reader's convenience:

1. All hardware interfaces to the controlled devices shall be release I/O boards in the LCUs.
- 2 The Ws shall not have hardware interfaces to the controlled devices they shall control them in directly by means of the LCUs, which will receive commands through the LAN.

Reliable software device drivers shall be provided for every I/O board used the AT Control system. They shall be completely tested to prove the full access to the required board's functionality and the correct exploitation of its resources. All necessary device drivers are available from the VLT Common Software, since all boards for seen are VME Standard boards, or fully compatible with existing boards .The exception here is the beacon interface bard which is 100% compatible with the Digital I/O board)..

All interfaces between the ATCS and the Electro Mechanical Hardware are described in the specific Interface Control Document [AD20] .

2. Software Interfaces:-

The ATCS has different types of software interfaces (this term is used in a very broad meaning) to external packages depending how the interaction is released ;

1. Program interface the ATCS will access the function of external Software through library (CCS Astronomical catalogue library ASM)
2. Message interface the ATCS will receive commands from a software silent (interferometer supervisor software-ISS)
3. Database interface at ATCS will exchange information retrieving and storing Data from/in to a database (VLT Archive system with in the VLT Dataflow system)

Functional requirements:

- The SRS contains descriptions of function and capability that the product must provide. The document also defines constraints and assumptions. Non-functional requirements describe the general characteristics of a system. They are also known as quality attributes. Functional requirements describe how a product must behave what its features and functions.
- Business Rules
- Transaction corrections, adjustments and cancellations.
- Administrative functions
- Authentication
- Authorizing levels
- Audit tracking
- External interfaces
- Certification requirements.
- Reporting requirements
- Historical data
- Legal or regulatory requirements.

Performance requirements:

1. System availability
2. Speed
3. Cost
4. Usability
5. Performance

The programming language and its suitable to the user. Simple of installation and training.

Logical database of requirements:

A logical database is the collected information stored information on multiple physical disk files and hard drives with in a computer. A logical database can stretch over multiple physical hard disks and information files. To have a logical database all given hard disks and information.

Design constraints:

A design constraints describe a particular subcomponents of a system and for its interfaces with other subcomponent the SRS writer should clearly distinguish between identifying required design.

Software system attributes:

A software requirements specification on (SRS) is a document that describes the nature of a project software. This includes the purpose scope and functional and non-functional requirements software and hardware.

Change management process:

Change management process to be used to identifying, evaluate and update the SRS to reflect changes in physical, scope and requirements. In this section we can discuss about for change management process and everyone involved in approving the documentation and next section can supporting documentation.

FEASIBILITY STUDY

Introduction:

As the name implies a feasibility analysis is used to determine the availability of an idea, such as ensuring a project is legally and technically feasible as well as economically justifiable. It tells as whether a project is worth the investment – in some cases, a project may not be double. There can be many reasons for this including requiring too many resources.

The prime focus of the feasibility study is evaluating the practicality of the proposed system keeping in mind a number of factors. The following factors are taken into account before deciding in favor of the new system. A feasibility study is done before an idea is approved or implemented and it should be an objective and rational assessment of the practicality of the proposed project. Therefore we aim to make analysis with an objective attitude, by using factual and impartial sources of information. This feasibility report includes analysis on five different aspects, which will each give us an insight in the current situation in Kenya. We will learn about general issues on development work experienced in the past and that can these issues to influence our own project. Authors such as Sachs, Easterly, Kerosene, and have a great deal of experience in the field and also did a lot of research on the subject.

While the important to conduct both plans before setting up a company, a business plan should only be conducted once the business has been deemed viable by a feasibility study. Its studies are conducted by business organizations, other organizations can naturally benefit from it as well. Since the study aims to discover whether an action is viable, it can help organizations to avoid costly exhausting ventures.

The study is typically used in situations where an important strategic decision needs to be taken. This can vary and some of example situations include:

- Change in business location
- Purchase of new equipment or software
- Acquisition of another company
- Hiring of additional employees

Feasibility study is the perfect tool for situations where the impact is likely to be big in terms of operational or economic significance. In this study which is performed by an organization in order to evaluate whether a specific action makes sense from an economic or operational standpoint. The objectives of the study is to test the feasibility of a specific action and to determine and define any issues that would argue against that action. A business plan provides a planning function and defines the actions needed to take a business idea into reality, whereas a feasibility study provides an investigation into a specific function and whether it is viable.

They are five types of feasibility studies are

- Technical feasibility
- Economic feasibility

- Operational feasibility
- Legal feasibility
- Scheduling feasibility

1. Technical feasibility

The assessment assesses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working system. Technical feasibility also involves evaluation of the hardware, software and other technical requirements of the proposed system.

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at the point in time, not too many detailed design of the system, making it difficult to access the issues like performance, cost on a number of issues have to be considered while doing a technical analysis.

- Is the proposed technology or solution practical?
- Do we currently possess the necessary technology?
- Do we possess the necessary technical expertise, and is the schedule reasonable?

2. Economic feasibility:

This assessment typically involves a cost, benefits analysis of the project, helping organizations determine the availability cost and benefits associated with a project before financial resources are allocated. It also serves as an independent project credibility helping decision makers determine the positive economic benefits to the organization that the proposed project will provide.

It also serves as an independent project assessment and enhances project credibility helping decision makers determine the positive economic benefits to the organization that the proposed project will provide. The Economic feasibility study (EFS) is to demonstrate the net benefits of a proposed project for accepting or disbursing electronic funds, benefits, taking into consideration the benefits and costs to the agency other state agencies and the general public as a whole.

- The state commissioned a report on the economic feasibility of a single payer health system
- They were looking at upgrading their computer systems, but once they looked at the economic feasibility they realized the additional revenues would not offset the costs.

3. Operational feasibility:

This assessment involves undertaking a study and determine the whether and how well- the organizations needs can be met by completing the project operational feasibility studies also examine how a project plan satisfies the requirements analysis phase of system development.

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. It takes care that the management and the users support the project. The opportunities identified during scope opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of system development. It is needed to develop, purchase, install, or operate the system.

- Cost benefit analysis
- Long term returns
- Maintain cost

4. Legal feasibility:

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like data protection acts, or social media laws. Let's say an organization wants to.

Legal feasibility is a determination of whether a proposed project infringes on known Acts, status as well as any pending legislation. In some instances the project might appear sound, on closer investigation it may be found to infringe in several legal areas, Concerned with legal issues.

- Determines whether the proposed system conflicts with legal requirements.
- Copyrights or patent laws
- Antitrust laws
- Labor relations and regulations
- Safety regulation
- Union contracts
- National data and work laws

5. Scheduling feasibility:

This assessment is the most important for project success, after all, a project will fail if not completed on time, In scheduling feasibility, an organization estimates how much time the project will take to complete when these areas have all been examined, the feasibility analysis helps identify any constraints the proposed all a project will it is not completed on time, In scheduling feasibility on organization estimates how much time the project will take to complete. When these areas have all been examined feasibility analysis helps identify any constraints the proposed project may face, including.

- Internal project constraints:
Technical, Technology Budget Resource.
- Internal corporate
- constraints:
Financial, Marketing, Exports etc... .
- External constraints:
Logistics, Environment laws etc... .



Data Flow Diagrams

A data flow diagrams (DFD) maps out of the flow of information for any process. It uses defined symbols like rectangles, circles, and arrows, plus, short text label to shown data inputs, outputs storage points and the routes between each destination. Data flow charts can range from simple even hand drawn process, overviews to in depth, multi-level.

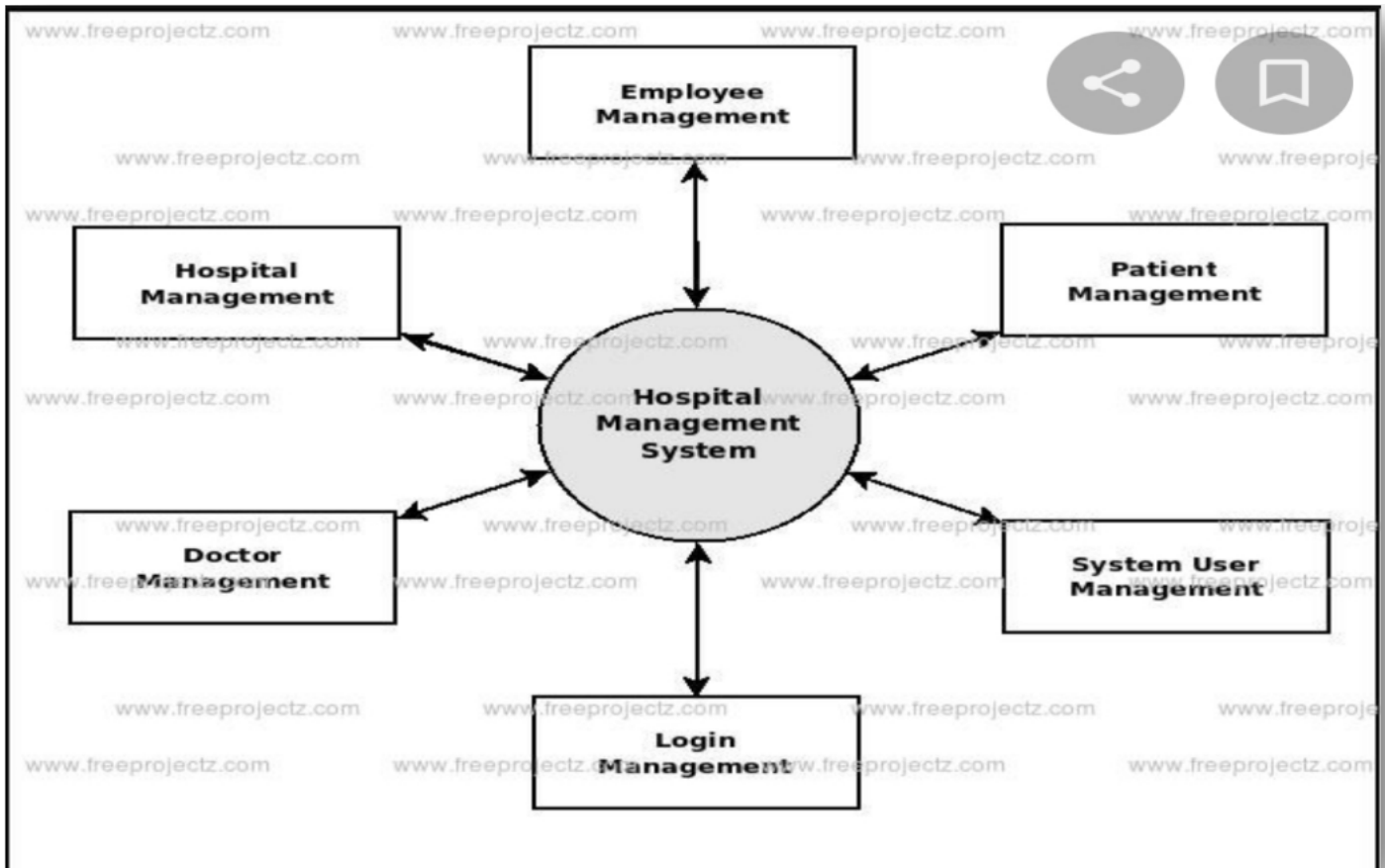
Level – 0 DFD:

High level entities and process flow of veterinary management system.

1. Managing all the employees
2. Managing all the patients.
3. Managing all the system users.
4. Managing all the doctors.
5. Managing all the hospitals.

Level– 1 DFD:





Level 1 DFD

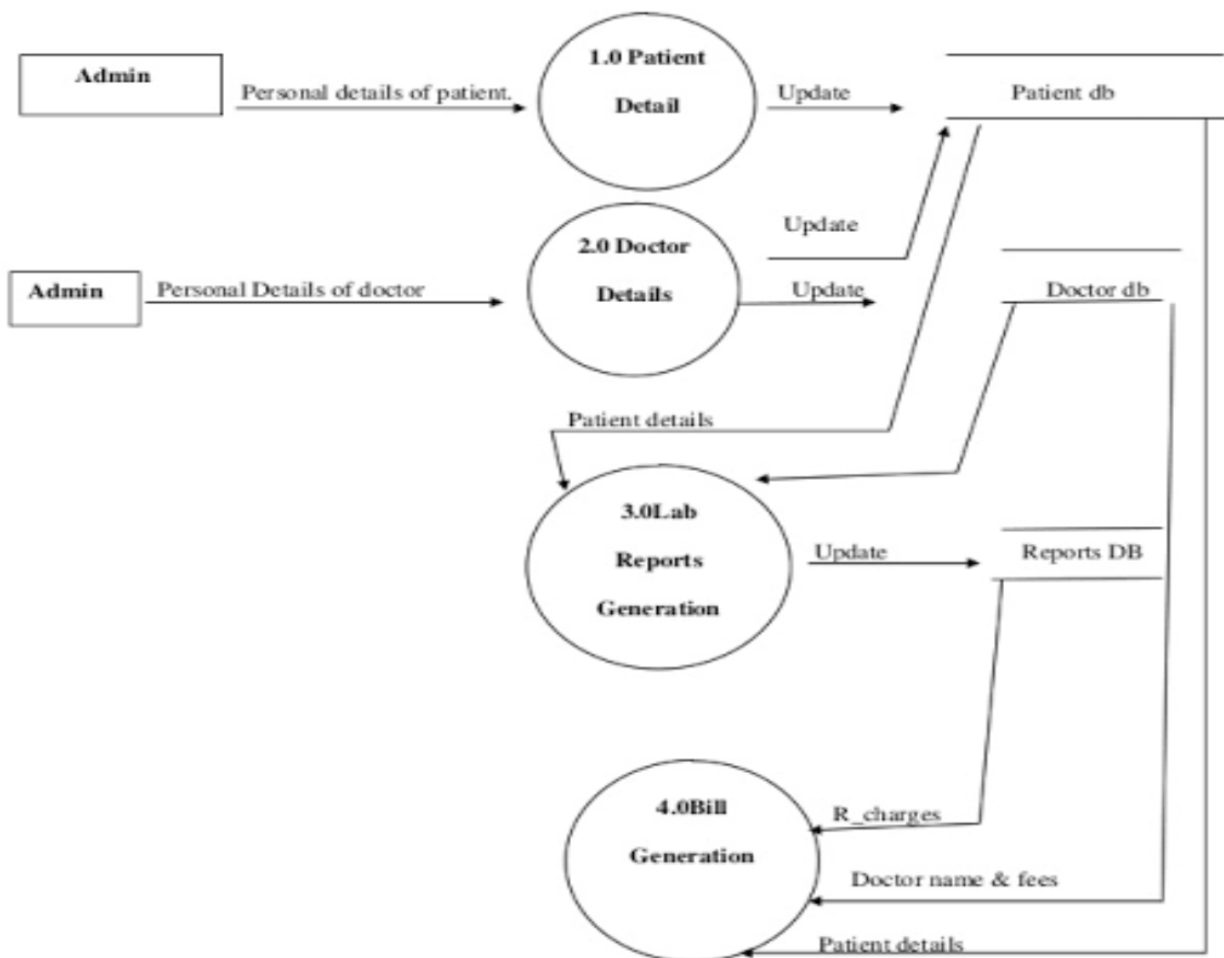


Fig. No.2.2

In this level – 1 DFD to form the patient details, person details of patient details are update also. In this lab report generation and update the doctor and consult the doctor DB. In this bill generation and doctor name & fields are in patient details also.

Level – 2 DFD:



DFD: Level 2

In this level 2 DFD to form the injection dates and injection database are retrieves and doctor the injection entry user the stores and medicine prescription are use in medicine labs. Diet advice and retrieve in diet advice db. And the test of diagnosis of db. In this so many levels and details of this topic.

DATABASE DESIGN

Pet care is a medium sized Veterinary surgery with six branches across London. There is the entity relationship model of the data held by pet care. Pet care want a database system developed to handle the records of the animals they are for, prescriptions and appointments. According to an initial analysis of pet care, the entities include owner, breed, animal type, appointment, Veterinary doctor, branch, prescription, drug and drug type.

Pet care provides various treatments for all types of pets. The details of each treatment include a treatment number, full description of the treatment, and the cost to the pet owner. Based on the result of the examination of a sick pet, the vet may propose one or more types of treatment. For each types of treatment, the information recorded includes the examination number and date.

The pet owner is responsible for the cost of the treatment given to a pet. The owner is invoiced for the treatment arising from each examination, and details recorded on the invoice include the invoice number, invoice date, owner number, owner name and full address, put number, put name, and details of the treatment given. The invoice provides the cost for each type

1. Owner and Animal is one to many Relationship: All animals have an owner.
2. Breed and Animal is one to many Relationship: Animals are defined by particular
3. Animal type and Breed is one to many Relationship: Animals are defined by type (dog, cat, rabbit etc...) and also by particular breed.
4. Animal and Appointment is one to many Relationship : The appointment for each animal are based upon the owner request.
5. Veterinary doctor and Appointment is one to many Relationships The appointments for each veterinary doctor are based upon the diagnosis made and the charge mode.
6. Appointment and prescription is one to many Relationship: The outcome of the appointment might be have a prescription or more than one prescription.
7. Drug type and Drug one too many Relationship: The drug type must be taken for based upon the drug type of prescription. Entity relationship and database modeling methods used to produce the types of conceptual schema, model, system, often a description and relational database and its requirements for top-down fashion or a combination of elements in a table whose purpose is to identity records from the table. A foreign key is a column in a table that uniquely identities.

Normalization:

In Order to avoid redundancy or data duplication we make use of real concept called Normalization. Data structure is redefined to a process called normalization data are grouped together in the simplest way so that the later changes can be made with minimum impact on data redundancy and promotes integrity. Normalization is the technique of separating redundant field and breaking up a large tables into a smaller one. It is also used to

avoid insertion, deletion, and updating animal

- **Entity:** It is a data aggregate that represents no. of data elements.
- **Attribute:** These are property associated with each entity.
- **Key:** It is a unique identities of the entity.
- **Primary key:** It is an orbit choice of one of the candidate key that helps to uniquely identify each record. In short the values for each of the 3NF are shown below.
- **Foreign key:** foreign key is a column or group of columns in a relational database table that provides link between two tables.

TABLE DESIGN

Table Name: Admin table

S.NO	Field name	Data type	Constraint	Description
1.	ADMIN NAME	VARCHAR	NOTNULL	ADMIN NAME
2.	PASSWORD	VARCHAR	NOTNULL	PASSWORD

Table Name: Owner

The table is used to store the owner details are available in books, internet.

S.NO	Field name	Data type	Type	Description
1	OWNER NAME	VARCHAR	20	OWNER NAME
2	EMAIL ID	VARCHAR	50	EMAIL ID
3	PHONE NO	VARCHAR	12	PHONE NO
4	DISTRICT	VARCHAR	40	DISTRICT
5	PINCODE	VARCHAR	40	PINCODE

Table Name: Doctor

The table is used to store the patient details are available in the system.

S.NO	Field Name	Type	Constraint	Description
1	DOCTOR NAME	VARCHAR	20	DOCTOR NAME
2	GENDER	VARCHAR	20	GENDER
3	DOCTOR CONTACT NO	VARCHAR	15	DOCTOR CONTACT NO
4	HOSPITAL NAME	VARCHAR	40	HOSPITAL NAME
5	CITY	VARCHAR	40	CITY

Table Name: Available

S.NO	Field Name	Type	Constraint	Description
1	AID	VARCHAR	40	AID
2	DOCTOR CODE	VARCHAR	20	DOCTOR CODE
3	DOC-DATE	VARCHAR	40	DOC-DATE

Table Name: Animal table

S.NO	Field Name	Type	Constraint	Description
1	ANIMALID	VARCHAR2(40)	FOREIGN KEY	AID
2	ANIMAL TYPE	VARCHAR2(40)	FOREIGN KEY	ANIMAL TYPE
3	OWNERID	VARCHAR2(40)	FOREIGN KEY	OWNERID
4	ANIMAL AGE	VARCHAR2(40)	NOT NULL	ANIMALAGE
5	BREED	VARCHAR2(20)	NOT NULL	BREED

Table Name: Appointment

S.NO	Field Name	Data type	Constraints	Description
1	APPID	VARCHAR2(20)	FOREIGN KEY	APPID
2	DOC-CODE	VARCHAR2(20)	FOREIGN KEY	DOC-CODE
3	OWNERID	VARCHAR2(20)	FOREIGN KEY	OWNERID
4	ANIMALID	VARCHAR2(20)	FOREIGN KEY	ANIMALID
5	ADATE	VARCHAR2(40)	NOT NULL	ADATE
6	ATIME	VARCHAR2(40)	NOT NULL	ATIME

Forms

HOME PAGE:-

localhost / 127.0.0.1 | phpM

veterinary

localhost/veter/kmn.html

VETERINARY

HOMEPAGE LOGIN ADMIN APPOINTMENT ANIMAL OWNER AVAILABLE ABOUT

VETERINARY INFORMATION MANAGEMENT SYSTEM

SIMPLIFY YOUR LIFE

Track Vaccinations And Prescription

REGISTER

Type here to search

20:43 29-03-2020

Admin Register.php

localhost / 127.0.0.1 / veter registration

localhost/veter/Registrar.php

Admin_Register

Username

Email

Password

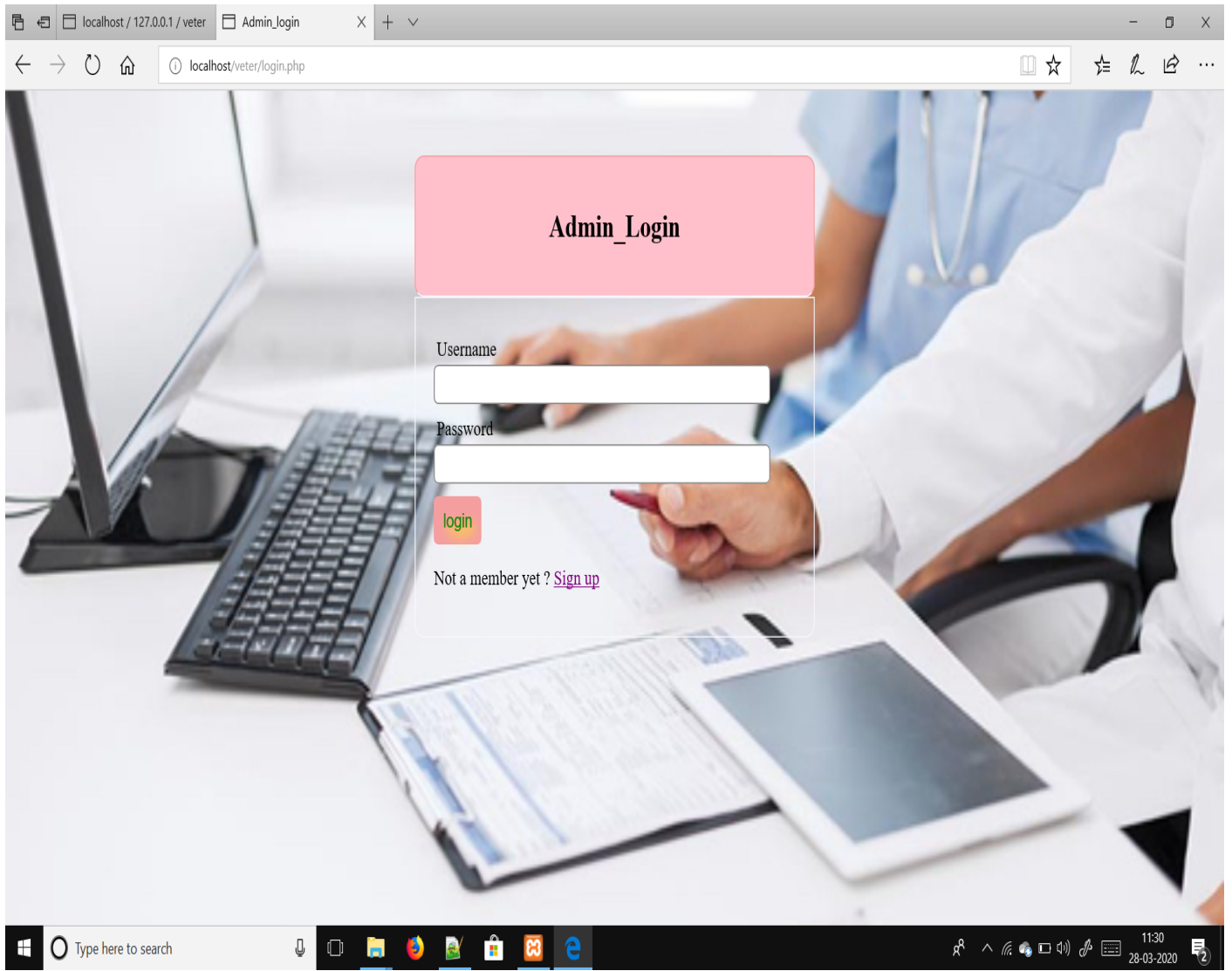
Conformpassword

Already a member? [sign in](#)

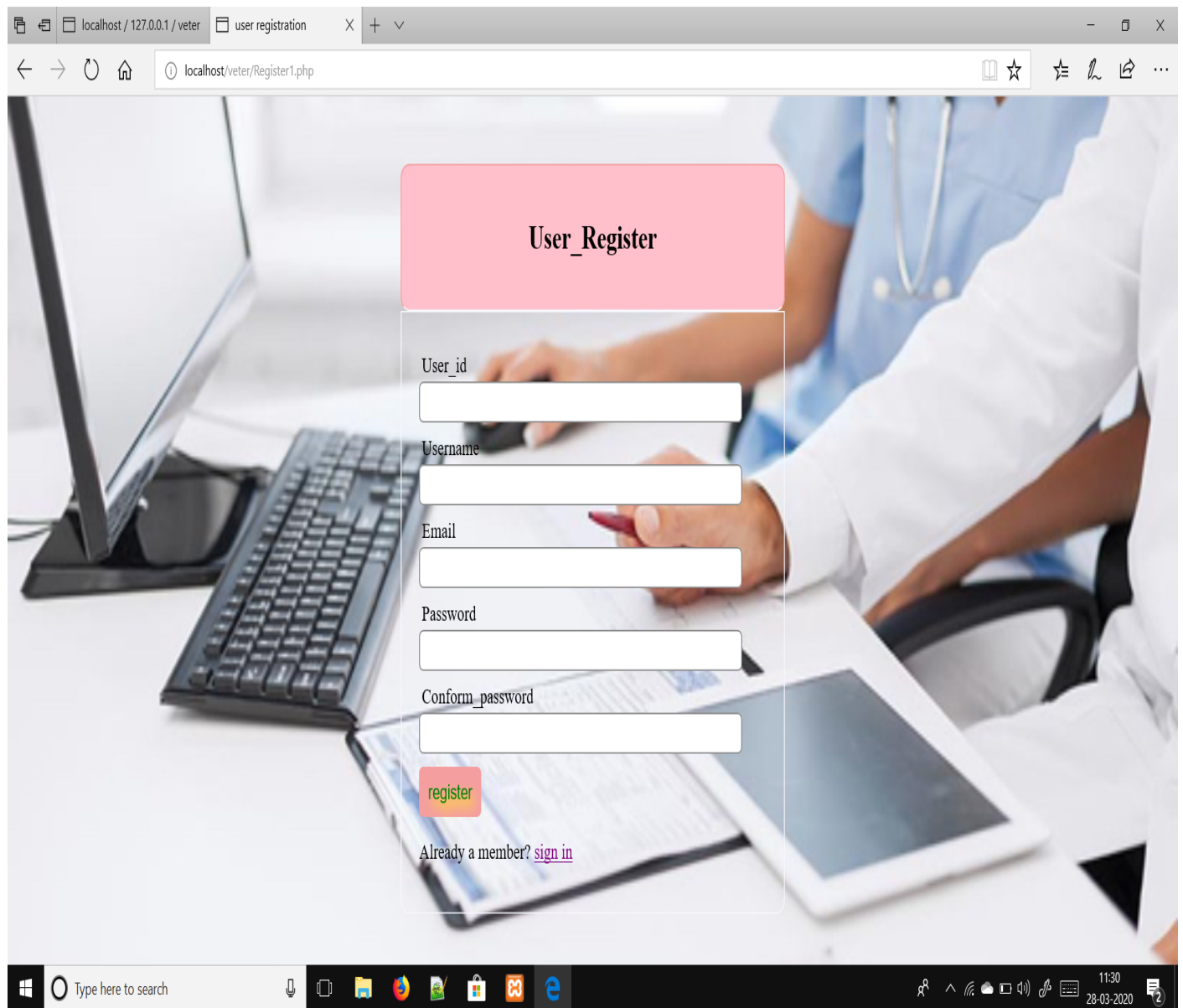
Type here to search

11:30
28-03-2020

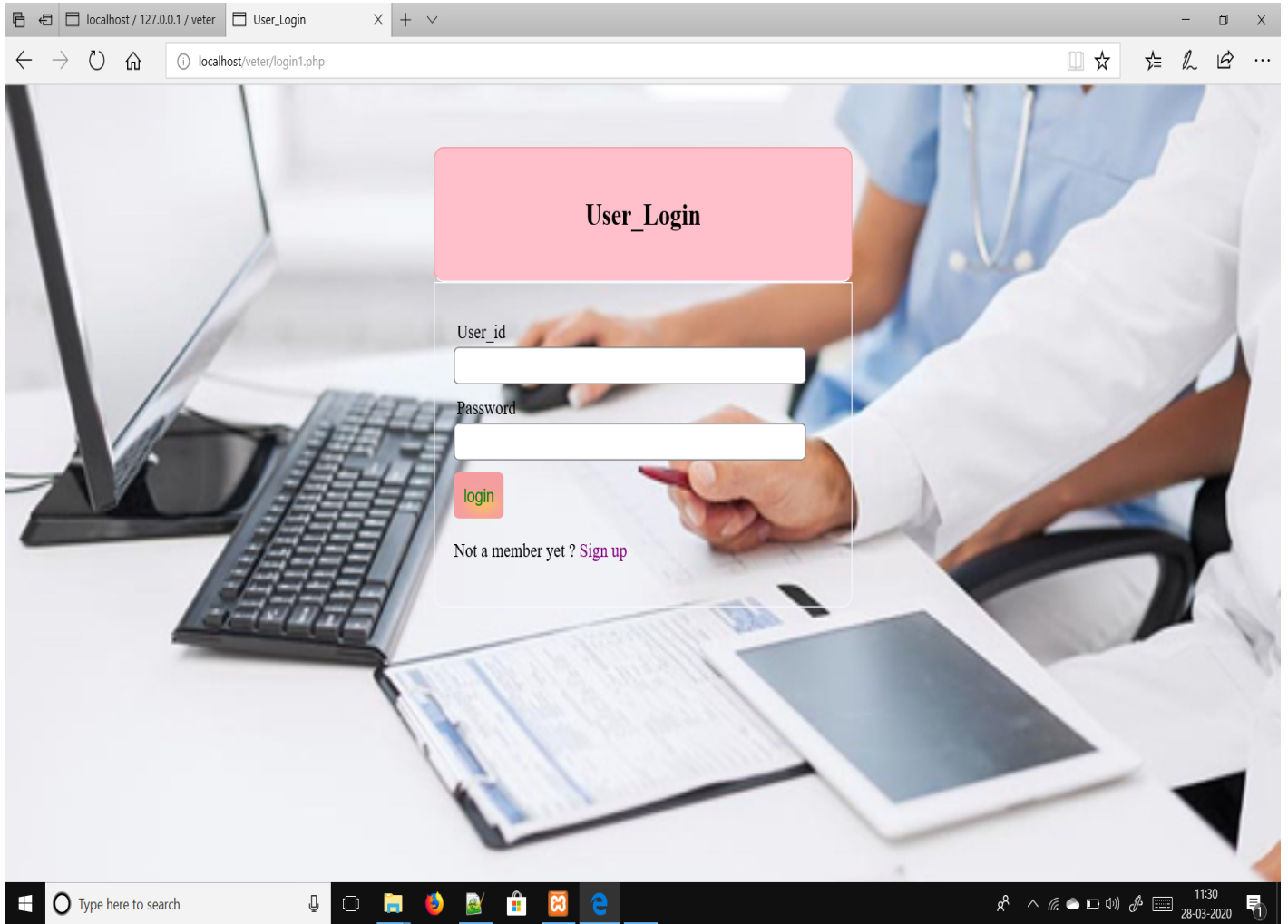
Admin Login.php



User Register1.php



User Login1 .php



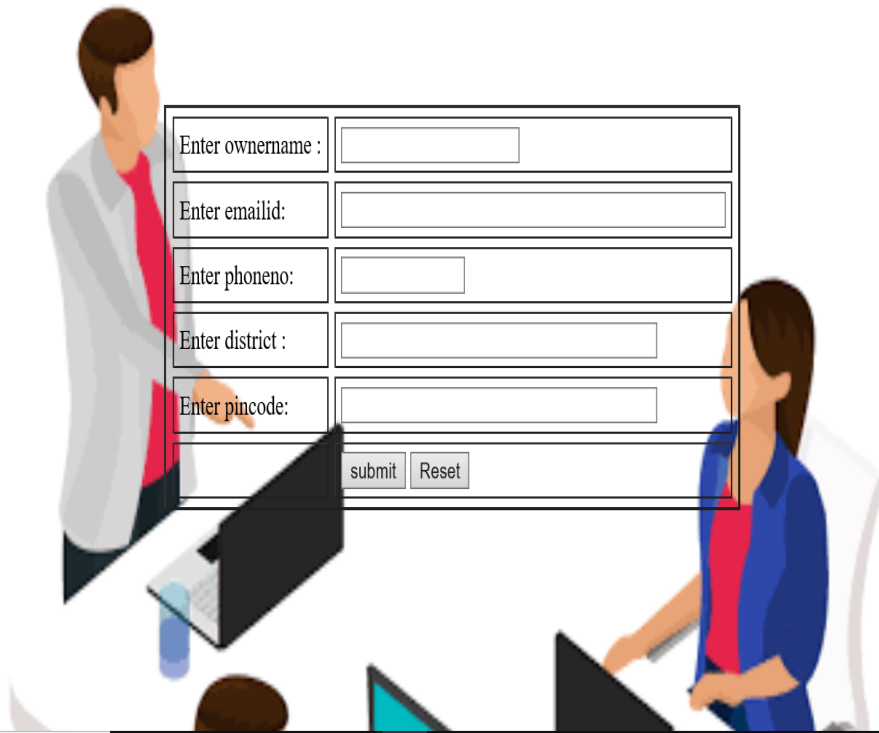
Owner info:-



//owner.php

VETERINARY INFORMATIONS MANAGEMENT SYSTEM

Owner Check Status



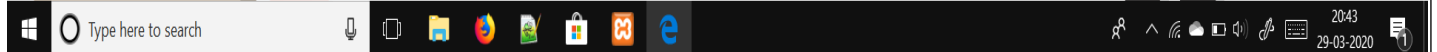
Doctor Appointment-



VETERINARY INFORMATIONS MANAGEMENT SYSTEM

Doctor Appointment

Enter doctorcode :	<input type="text"/>
Enter ownerid :	<input type="text"/>
Enter animalid :	<input type="text"/>
Enter adate:	<input type="text"/>
Enter atime:	<input type="text"/>
<input type="button" value="submit"/> <input type="button" value="Reset"/>	



LAvailability:-

VETERINARY INFORMATIONS MANAGEMENT SYSTEM

Best Software for Doctors to Manage their Practice

Enter Availability

Enter doctorcode :	<input type="text"/>
Enter doctordate:	<input type="text"/>
Enter 1ST shift :	<input type="text"/>
Enter 2ND shift :	<input type="text"/>
Enter 3RD shift:	<input type="text"/>
	<input type="button" value="submit"/> <input type="button" value="Reset"/>

Animal:-

localhost / 127.0.0.1 | phpM localhost

localhost/veter/animal1.php

VETERINARY INFORMATIONS MANAGEMENT SYSTEM

Enter Animal Details

Enter animal id :	<input type="text"/>
Enter animaltype :	<input type="text"/>
Enter ownerid:	<input type="text"/>
Enter animalage :	<input type="text"/>
Enter breed:	<input type="text"/>
	<input type="submit" value="submit"/> <input type="button" value="Reset"/>

Windows taskbar: Type here to search, 20:43, 29-03-2020

Appointment:-

VETERINARY INFORMATIONS MANAGEMENT SYSTEM

Doctor Appointment

Enter doctorcode :	<input type="text"/>
Enter ownerid :	<input type="text"/>
Enter animalid :	<input type="text"/>
Enter adate:	<input type="text"/>
Enter atime:	<input type="text"/>
	<input type="button" value="submit"/> <input type="button" value="Reset"/>



CODING

ADMIN REGISTRATION:-

```
<?php include('server.php');?>
<html>
<head>
<title>registration</title>
<link rel="stylesheet" type="text/css" href="Style.css" >
</head>
<body>
<div class="header" >
<h2> admin Register </h2>
</div>
<form method="post" action="Register.php" >
<?php include('errors.php'); ?>
<div class="input-group" >
<label> username</label>
<input type="text" name="username" value="<?php echo $username;?>" >
</div>
<div class="input-group" >
<label> Email </label>
<input type="text" name="Email" value="<?php echo $email;?>" >
</div>
<div class="input-group" >
<label> password </label>
<input type="password" name="password_1" >
```

```

</div>
<div class="input-group">
<label> conformpassword</label>
<input type="password" name="password_2" >
</div>
<div class="input-group">
<button type="submit" name="register" class="btn">
register</button></div>
<p>
Already a member? <a href="login.php">sign in </a>
</p>
</form>
</body>
</html>

```

ADMIN LOGIN:-

```

<?php include('server.php');?>
<html>
<head>
<title> login </title>
<link rel="stylesheet" type="text/css" href="Style.css" >
</head>
<body>
<div class="header">
<h2> admin login </h2>
</div>
<form method="post" action="login.php">
<?php include('errors.php'); ?>
<div class="input-group">
<label> username </label>
<input type="text" name="username" >

```

```

</div>
<div class="input-group">
<label> password </label>
<input type="password" name="password">
</div>
<div class="input-group">
<button type="submit" name="login" class="btn">
login</button></div>
<p>
Not a member yet ? <a href="Register.php"> Sign up </a>
</p>
</form>
</body>
</html>

```

ADMIN REGISTER/LOGIN SERVER:-

```

<?php
session_start();
$username="";
$email="";
$errors= array();

$db = mysqli_connect('localhost', 'root', '', 'veter');

if(isset($_POST['register'])){
    $username= mysqli_real_escape_string($db,$_POST['username']);
    $email= mysqli_real_escape_string($db, $_POST['Email']);
    $password_1 = mysqli_real_escape_string($db, $_POST['password_1']);
    $password_2 = mysqli_real_escape_string($db, $_POST['password_2']);

    if(empty($username)){

```



```

    array_push($errors, " Username is required");
}
if(empty($email)){
    array_push($errors, " Email is required");
}
if(empty($password_1)){
    array_push($errors, " Password is required");
}
if($password_1 != $password_2){
    array_push($errors, " Passwords did not match");
}

if(count($errors)==0){
    $password=md5($password_1);
    $sql= "INSERT INTO admin(username, email, password)
        VALUES (' $username', '$email', '$password)";
    mysqli_query($db, $sql);

    header(' location: kmn.html'); // redirect to homepage.
}
}

//log user in form login page
if(isset($_POST['login'])){
    $username= mysqli_real_escape_string($db, $_POST['username']);
    $password= mysqli_real_escape_string($db, $_POST['password']);

    //ensure that form fields are properly filled .

```

```

if(empty($username)){
    array_push($errors," Username is required");
}
if(empty($password)){
    array_push($errors," Password is required");
}
if(count($errors)==0){
    //encrypt password before comparing with that from database
    $password=md5($password);
    $query=" select * from admin WHERE username=' $username' AND
password=' $password' ";
    $result= mysqli_query($db,$query);
    if(mysqli_num_rows($result)==1)
    {

        header('location:doctor.php');}
else{
    array_push($errors,"invalid username and password");
    header('location:login.php');
}
}
}
}
//logout
if(isset($_GET['logout'])){
    session_destroy();
    unset($_SESSION['username']);
    header('location:kmn.html');
}
?>

```



USER REGISTRATION:-

```
<?php include('server1.php');?>
<html>
<head>
<title> user registration</title>
<link rel="stylesheet" type="text/css" href="Style.css" >
</head>
<body>
<div class="header">
<h2>user Register </h2>
</div>
<form method="post" action="Register1.php" >
<?php include('errors.php'); ?>
<div class="input-group">
<label> userid</label>
<input type="text" name="id" value="<?php echo $username;?>" >
</div>
<div class="input-group">
<label> username</label>
<input type="text" name="username" value="<?php echo $username;?>" >
</div>
<div class="input-group">
<label> Email </label>
<input type="text" name="Email" value="<?php echo $email;?>" >
</div>
<div class="input-group">
<label> password </label>
<input type="password" name="password_1" >
</div>
<div class="input-group">
```

```
<label> conformpassword</label>
<input type="password" name="password_2" >
</div>
<div class="input-group" >
<button type="submit" name="register" class="btn" >
register</button></div>
<p>
Already a member? <a href="login1.php" >sign in </a>
</p>
</form>
</body>
</html>
```

USER LOGIN:-

```
<?php include('server1.php');?>
<html>
<head>
<title> login </title>
<link rel="stylesheet" type="text/css" href="Style.css" >
</head>
<body>
<div class="header" >
<h2>user login </h2>
</div>
<form method="post" action="login1.php" >
<?php include('errors.php'); ?>
<div class="input-group" >
<label> userid </label>
```



```

<input type="text" name="id" >
</div>
<div class="input-group" >
<label> password </label>
<input type="password" name="password" >
</div>
<div class="input-group" >
<button type="submit" name="login" class="btn" >
login</button></div>
<p>
Not a member yet ? <a href="Register1.php" > Sign up </a>
</p>
</form>
</body>
</html>

```

USERS REGISTRATION/LOGIN SERVER:-

```

<?php
session_start();
$id="";
$username="";
$email="";
$errors= array();

$db = mysqli_connect('localhost', 'root', '', 'veter');

if(isset($_POST['register'])){
    $id= mysqli_real_escape_string($db,$_POST['id']);
    $username= mysqli_real_escape_string($db,$_POST['username']);

```



```

$email= mysqli_real_escape_string($db, $_POST['Email']);
$password_1 = mysqli_real_escape_string($db, $_POST['password_1']);
$password_2= mysqli_real_escape_string($db, $_POST['password_2']);

    if(empty($id)){
        array_push($errors, "Userid is required");
    }
if(empty($username)){
    array_push($errors, "Username is required");
}
if(empty($email)){
    array_push($errors, "Email is required");
}
if(empty($password_1)){
    array_push($errors, "Password is required");
}
if($password_1 != $password_2){
    array_push($errors, "Passwords did not match");
}

if(count($errors)==0){
    $password=md5($password_1);
    $sql= "INSERT INTO users(id,username, email, password)
        VALUES (' $id', '$username', '$email', '$password')";
    mysqli_query($db, $sql);

    $_SESSION['id']=$id;
    $_SESSION['success']="You are now logged in";
    header('location:index1 .php');// redirect to homepage.

```

```

}
}

//log user in form login page
if(isset($_POST['login'])){
    $id= mysqli_real_escape_string($db, $_POST['id']);
    $password= mysqli_real_escape_string($db, $_POST['password']);

    //ensure that form fields are properly filled .
    if(empty($id)){
        array_push($errors, "Userid is required");
    }
    if(empty($password)){
        array_push($errors, "Password is required");
    }
    if(count($errors)==0){
        //encrypt password before comparing with that from database
        $password=md5($password);
        $query=" select * from users WHERE id=' $id' AND password=' $password'";
        $result= mysqli_query($db, $query);
        if(mysqli_num_rows($result)==1 )
        {
            $_SESSION['id']=$id;
            $_SESSION['success']="You are now logged in";
            header('location:index1.php');}
    else{
        array_push($errors, "invalid userid and password");
        header('location:login1.php');
    }
}
}

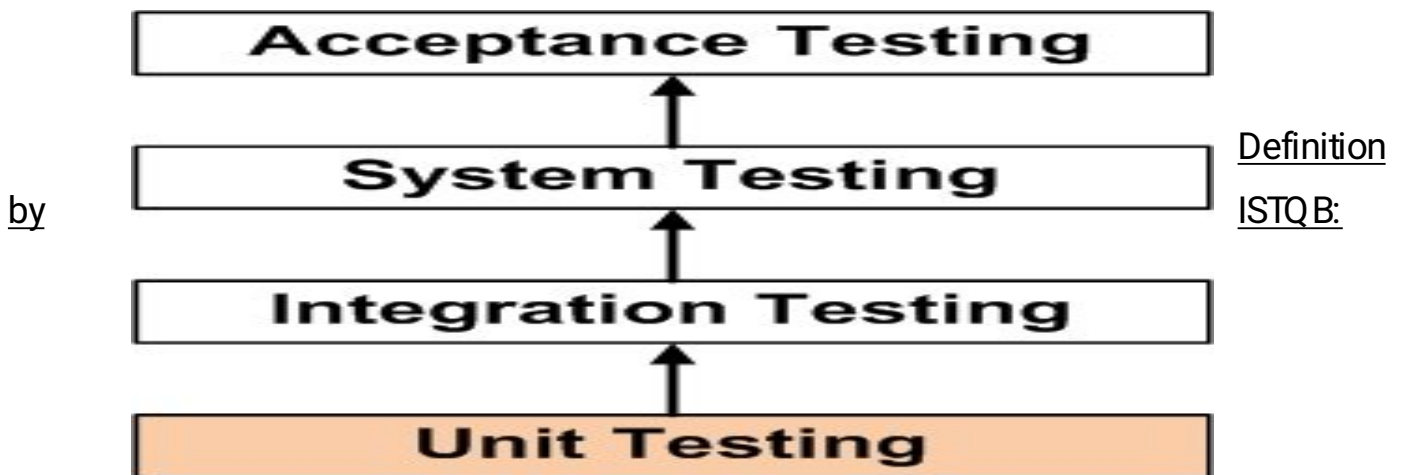
```

```
}  
  
//logout  
if(isset($_GET['logout'])){  
    session_destroy();  
    unset($_SESSION['id']);  
    header('location:login1.php');  
}  
?>
```


9. TESTING

9.1 Unit Testing

UNIT TESTING is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.



- **Unit testing:** See *component testing*.
- **Component testing:** The testing of individual software components.

Unit Testing Method

It is performed by using the White Box Testing method.

When is it performed?

Unit testing is the first level of software testing and is performed prior to Integration Testing.

Who performs it?

It is normally performed by software developers themselves or their peers. In rare cases, it may also be performed by independent software testers.

Unit Testing Tasks

- Unit Test Plan

- Prepare
- Review
- Rework
- Baseline
- Unit Test Cases/ Scripts
 - Prepare
 - Review
 - Rework
 - Baseline
- Unit Test
 - Perform

Unit Testing Benefits

- Unit testing increases confidence in changing/ maintaining code. If good unit tests are written and if they are run every time any code is changed, we will be able to promptly catch any defects introduced due to the change. Also, if codes are already made less interdependent to make unit testing possible, the unintended impact of changes to any code is less.
- Codes are more reusable. In order to make unit testing possible, codes need to be modular. This means that codes are easier to reuse.
- Development is faster. How? If you do not have unit testing in place, you write your code and perform that fuzzy 'developer test' (You set some breakpoints, fire up the GUI, provide a few inputs that hopefully hit your code and hope that you are all set.) But, if you have unit testing in place, you write the test, write the code and run the test. Writing tests takes time but the time is compensated by the less amount of time it takes to run the tests; You need not fire up the GUI and provide all those inputs. And, of course, unit tests are more reliable than 'developer tests'. Development is faster in the long run too. How? The effort required to find and fix defects found during unit testing is very less in comparison to the effort required to fix defects found during system testing or acceptance testing.
- The cost of fixing a defect detected during unit testing is lesser in comparison to that of defects detected at higher levels. Compare the cost (time, effort, destruction, humiliation) of a defect detected during acceptance testing or when the software is live.
- Debugging is easy. When a test fails, only the latest changes need to be debugged. With testing at higher levels, changes made over the span of several days/ weeks/ months need to be scanned.
- Codes are more reliable. Why? I think there is no need to explain this to a sane person.

9.2 WHITE BOX TESTING

- White Box Testing is defined as the testing of a software solution's internal structure, design, and coding. In this type of testing, the code is visible to the tester. It focuses primarily on verifying the flow of inputs and outputs through the application, improving design and usability, strengthening security. White box testing is also known as Clear Box testing, Open Box testing, Structural testing, Transparent Box testing, Code-Based testing, and Glass Box testing. It is usually performed by developers.
- It is one of two parts of the "**Box Testing**" approach to software testing. Its counterpart, **Black box testing**, involves testing from an external or end-user type perspective. On the other hand, White box testing is based on the inner workings of an application and revolves around internal testing.
- The term "White Box" was used because of the see-through box concept. The clear box or White Box name symbolizes the ability to see through the software's outer shell (or "box") into its inner workings. Likewise, the "black box" in "Black Box Testing" symbolizes not being able to see the inner workings of the software so that only the end-user experience can be tested.

9.3 BLACK BOX TESTING

- Black box testing is defined as a testing technique in which functionality of the Application Under Test (AUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software. This type of testing is based entirely on software requirements and specifications.



- In Black Box Testing we just focus on inputs and output of the software system without bothering about internal knowledge of the software program.
- The above Black-Box can be any software system you want to test. For Example, an operating system like Windows, a website like Google, a database like Oracle or even your own custom application. Under Black Box Testing, you can test these applications by just focusing on the inputs and outputs without knowing their internal code implementation. Consider the following video tutorial-

9.4 GREY BOX TESTING

Gray Box Testing is a technique to test the software product or application with partial knowledge of the internal workings of an application.

In this process, context-specific errors that are related to web systems are commonly identified. It will increase the testing coverage by concentrating on all of the layers of any complex system.

Gray Box Testing is a software testing method, which is a combination of both White Box Testing and Black Box Testing method.

- In White Box testing internal structure (code) is known
- In Black Box testing internal structure (code) is unknown
- In Grey Box Testing internal structure (code) is partially known



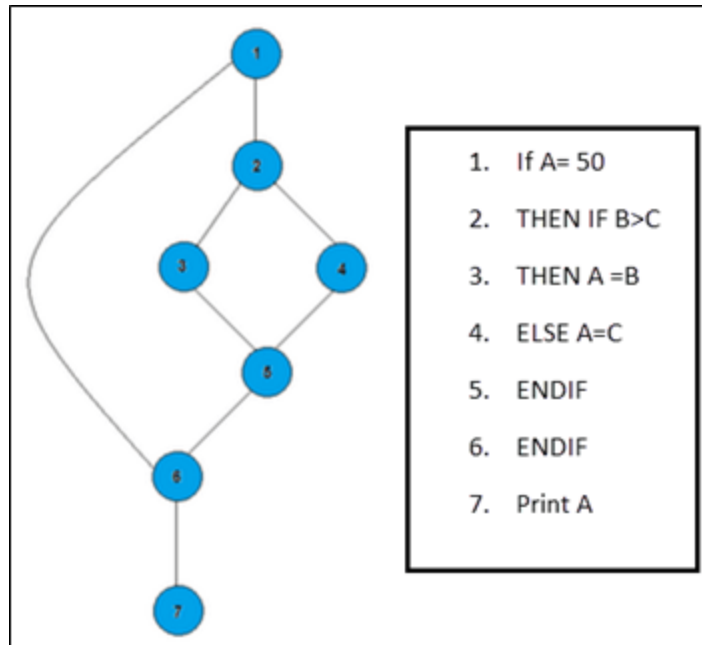
In Software Engineering, Gray Box Testing gives the ability to test both sides of an application, presentation layer as well as the code part. It is primarily useful in Integration Testing and Penetration Testing.

9.5 BASIS PATH TESTING

The basis path testing is same, but it is based on a White Box Testing method, that defines test cases based on the flows or logical path that can be taken through the program. In software engineering, Basis path testing involves execution of all possible blocks in a program and achieves maximum path coverage with the least number of test cases. It is a hybrid of branch testing and path testing methods.

The objective behind basis path in software testing is that it defines the number of independent paths, thus the number of test cases needed can be defined explicitly (maximizes the coverage of each test case).

Here we will take a simple example, to get a better idea what is basis path testing include



In the above example, we can see there are few conditional statements that is executed depending on what condition it suffice. Here there are 3 paths or condition that need to be tested to get the output,

- Path 1: 1,2,3,5,6, 7
- Path 2: 1,2,4,5,6, 7
- Path 3: 1, 6, 7

Steps for Basis Path testing

The basic steps involved in basis path testing include

- Draw a control graph (to determine different program paths)
- Calculate Cyclomatic complexity (metrics to determine the number of independent paths)
- Find a basis set of paths
- Generate test cases to exercise each path

Advantages of Basic Path Testing

- It helps to reduce the redundant tests
- It focuses attention on program logic
- It helps facilitates analytical versus arbitrary case design
- Test cases which exercise basis set will execute every statement in a program at least once

Conclusion:

Basis path testing helps to determine all faults lying within a piece of code.

9.6 CONDITIONAL TESTING

Types of Conditions

The method is easy to describe: It includes a ``checklist'' of cases to be covered for each kind of condition that one might find in source code, in the test for an `if-then` or `if-then-else` statement, or as the test for a `while-do` or `repeat-until` loop.

Simple Conditions

Boolean Variables

A Boolean variable, or (application of a) Boolean function, is one kind of simple condition.

Relational Expressions

Relational expressions are comparisons of two values of the same type. If v_1 and v_2 are values of the same type, and it is possible to test ``equality'' of values of this type, then two (kinds of) relational expressions involving this type are

- $v_1 == v_2$ (`` v_1 is equal to v_2 '')
- $v_1 != v_2$ (`` v_1 is not equal to v_2 '')

If the type is ``ordered'' then additional relational expressions are also possible:

- $v_1 < v_2$ (`` v_1 is strictly less than v_2 '')
- $v_1 <= v_2$ (`` v_1 is less than or equal to v_2 '')
- $v_1 >= v_2$ (`` v_1 is greater than or equal to v_2 '')
- $v_1 > v_2$ (`` v_1 is strictly greater than or equal to v_2 '')

In all cases, a truth value (either `true` or `false`) is obtained whenever you assign values of the given type to each of the variables v_1 and v_2 . Of course, `` v_1 '' and `` v_2 '' might also be (calls to) functions returning values of the given type, in the expressions given above.

9.7 Compound Conditions

Compound conditions are conditions that are obtained by combining together simple conditions, using ``logical connectives,'' such as ``and'' (`&&`), ``or'' (`||`), or ``not'' (`!`); additional connectives, such as ``implies'', ``and,'' or ``exclusive or'' might also be used.

9.8 Guidelines for Test Design

As mentioned above, a ``checklist'' or simple rule describing required test cases can be given for each of the kinds of conditions listed above.

Guidelines for Simple Conditions

Guidelines for Boolean Variables

In this case, two test cases should be covered:

- There should be a test such that the value of the Boolean variable (or function) included in the condition is `true` at some point when the condition is tested.
- There should be a test such that the value of the Boolean variable or function is `false` at some point when the condition is tested.

Guidelines for Relational Expressions

Suppose the relational expression to be considered is

Where e_1 and e_2 are expressions of the same type, and op is one of the comparisons (`=`, `!`, `<`, and so on) listed above.

If this is an ``ordered'' type, so that it makes sense to ask whether e_1 is ``less than'' e_2 , then three cases should be covered by tests:

- There should be a test case such that e_1 is strictly less than e_2 , at some point when the condition is tested.
- There should be a test case such that e_1 is equal to e_2 at some point when the condition is tested.
- There should be a test case such that e_1 is strictly greater than e_2 when the condition is tested.

On the other hand, if the type is not ``ordered,'' then only two cases need to be covered (or, can be distinguished):

- There should be a test case such that e_1 is equal to e_2 at some point when the condition is tested.
- There should be a test case such that e_1 is not equal to e_2 at some point when the

condition is tested.

Guidelines for Compound Conditions

Note that either two or three cases need to be covered for each one of the simple conditions listed above.

As described above, a compound condition connects some number of simple conditions together, using logical connectives. Suppose that a complex condition connects k simple conditions with three cases, and h simple conditions with two cases, together. Then, it is possible - in principle - to define a total of

$$3^k 2^h$$

Different combinations of these cases.

Some might be impossible to achieve. For example, if x , y , and z are variables of the same ordered type, and three of the relational expressions included in the compound condition are

- i. $x < y$,
- ii. $y < z$,
- iii. $x < z$

then one of the (twenty-seven) combinations of cases that you'd get by considering these three relational expressions would be equivalent to the following condition on x , y , and z :

$$(x < y) \ \&\& \ (y < z) \ \&\& \ (x > z)$$

It should be clear that there is no way to assign values to x , y and z such that the above condition is satisfied.

The guideline for ``compound combinations'' is now easy to state: For each of the above combinations of conditions that is *possible*, include a test such that this combination of conditions holds at some point (during execution of the program) when the condition is checked.

9.9 DATA FLOW TESTING:

Data flow testing is a family of test strategies based on selecting paths through the program's control flow in order to explore sequences of events related to the status of variables or data objects. Dataflow Testing focuses on the points at which variables receive values and the points at which these values are used.

Advantages of Data Flow Testing:

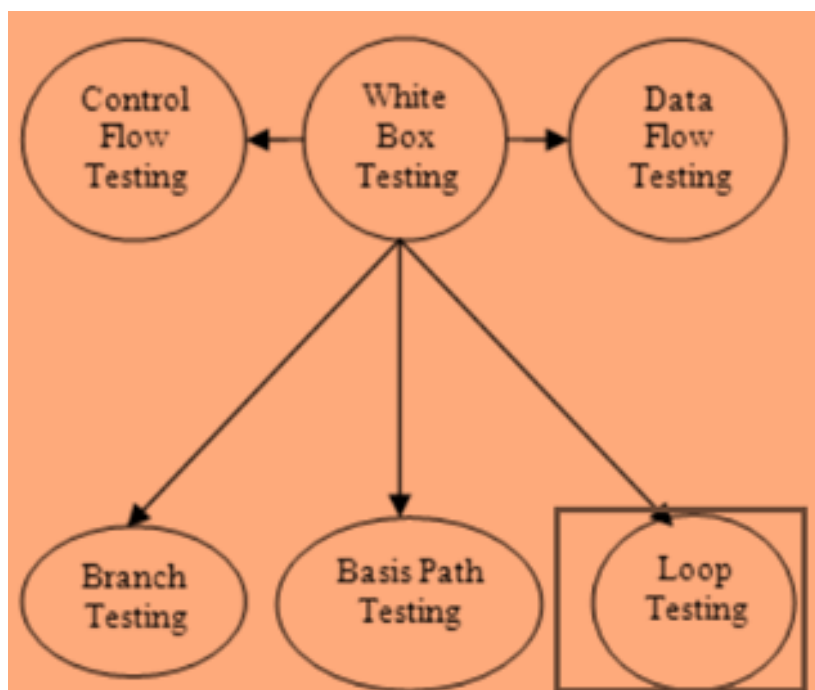
Data Flow testing helps us to pinpoint any of the following issues:

- A variable that is declared but never used within the program.
- A variable that is used but never declared.
- A variable that is defined multiple times before it is used.
- De allocating a variable before it is used.

9.10 LOOP TESTING

Loop Testing is defined as a software testing type, that completely focuses on the validity of the loop constructs. It is one of the parts of Control Structure Testing (path testing, data validation testing, condition testing).

Loop testing is a White box testing. This technique is used to test loops in the program.



Types of loop Tested

Examples of types of loop tested are,

- Simple loop
- Nested loop
- Concatenated loop
- Unstructured loop

Why do Loop Testing?

Loop Testing is done for the following reasons

- Testing can fix the loop repetition issues
- Loops testing can reveal performance/ capacity bottlenecks
- By testing loops, the uninitialized variables in the loop can be determined
- It helps to identify loops initialization problems.

CONCLUSION

All the suggestions forwarded in the s/w proposals have been completed and the final threshold the application has been crossed viewing through the system development a brief figure.

- It's a web –enabled project.
- This project offers user to enter the data through simple and interactive forms.
- It is more concerned about the validity of the data.
- Data storage and retrieved will become faster and easier.
- Through these features it will increase, the efficiency accuracy, and transparency.

REFERENCE

C.J THOMSON(2014). "usage of Reference management software at the university of Tornio". *JIS it is university of the Florence*. 4(1) .. 10. DOI. 4403 / *JIS It* 8679 Retrieved 2016-09-09

Meredith, Sandra (2014-01-21)." Critical review of referencing software when used with OSCOLA".

Technology 4(1). DOI; 10.2139/SSM.2184098,

ISSN 2042-115 X

<https://www.wikidata.org/wiki/Help>:

Sources// scientific- newspaper – or- article.

