

SAVITRIBAI PHULE PUNE UNIVERSITY

A PROJECT REPORT ON

**'APPLICATION FOR MANAGING A  
VETERINARY POLYCLINIC'**

SUBMITTED TOWARDS THE  
PARTIAL FULFILLMENT OF THE REQUIREMENTS OF

**BACHELOR OF ENGINEERING (Computer Engineering)**

**BY**

Shalini Prasanna

Exam No:B120204364

T. Pavithra

Exam No:B120204409

Supriya Waghmare

Exam No: B120204404

Ashlesha Waikos

Exam No:B120204424

**Under The Guidance of**

Prof. Pranjali Deshpande



**DEPARTMENT OF COMPUTER ENGINEERING**

**MKSSS'S CUMMINS COLLEGE OF ENGINEERING  
FOR WOMEN  
PUNE**

## CERTIFICATE

This is to certify that the Project Entitled

### **‘APPLICATION FOR MANAGING A VETERINARY POLYCLINIC’**

Submitted by

Shalini Prasanna	Exam No:B120204364
T. Pavithra	Exam No:B120204409
Supriya Waghmare	Exam No:B120204404
Ashlesha Waikos	Exam No:B120204424

is a bonafide work carried out by Students under the supervision of Prof. Pranjali Deshpande and it is submitted towards the fulfilment of the requirement of Bachelor of Engineering (Computer Engineering).

Prof. Pranjali Deshpande  
Internal Guide  
Dept. of Computer Engg.

Dr. Supriya Kelkar  
H.O.D  
Dept. of Computer Engg.

Dr. Madhuri Khambete  
Director  
MKSSS'S Cummins College of engineering for Women

Signature of Internal  
Examiner

Signature of External  
Examiner

# PROJECT APPROVAL SHEET

A Project Title

'APPLICATION FOR MANAGING A VETERINARY POLYCLINIC'

Is successfully completed by

Shalini Prasanna	Exam No:B120204364
T. Pavithra	Exam No:B120204409
Supriya Waghmare	Exam No:B120204404
Ashlesha Waikos	Exam No:B120204424

at

DEPARTMENT OF COMPUTER ENGINEERING

MKSSS'S Cummins College of engineering for women

SAVITRIBAI PHULE PUNE UNIVERSITY,PUNE

ACADEMIC YEAR 2015-2016

Prof. Pranjali Deshpande  
Internal Guide  
Dept. of Computer Engg.

Prof. Supriya Kelkar  
H.O.D  
Dept. of Computer Engg.

## ABSTRACT

Computed medical record database management is increasingly important as information and computer technology progress rapidly. Medical records contain treatment history and relevant experiences pertaining to the patients need to be documented accurately and efficiently. In many parts of the world clinical documentations are still being handwritten on forms and filed into paper medical records. The shortcomings of paper records are well identified, for example, illegibility and poorly organized records may lead to information gaps or inconsistencies, when used by another or even the same user later on. Hardcopies of documents are slow to be transferred around, with higher risk of being misplaced. In addition, when handled by multi-users in multi-locations, the quality of safekeeping can be significantly affected. The advent of computer technology has introduced enormous possibilities for electronic documentation and usage of electronic medical records. A computerized database management system is designed and implemented to replace the existing traditional hand recorded system in a veterinary polyclinic. The system provides easy access to patient registration, diagnosis, appointment and report generation. The new system has higher efficiency in terms of time taken, cost and data security protection as compared to the traditional hand recorded system.

## ACKNOWLEDGMENTS

*It gives us great pleasure in presenting the final project report on ‘**APPLICATION FOR MANAGING A VETERINARY POLYCLINIC**’.*

*I would like to take this opportunity to thank my internal guide **Prof. Pranjali Deshpande** for giving me all the help and guidance I needed. I am really grateful to them for their kind support. Their valuable suggestions were very helpful.*

*I am also grateful to **Dr. Supriya Kelkar**, Head of Computer Engineering Department, MKSSS’S Cummins College of Eng. for Women for her indispensable support, suggestions.*

*In the end our special thanks to **Mr. Arun Waikos** for providing various resources for Our Project.*

Shalini Prasanna  
T. Pavithra  
Supriya Waghmare  
Ashlesha Waikos  
(B.E. Computer Engg.)

# Contents

<b>1 Synopsis</b>	<b>1</b>
1.1 Project Title . . . . .	2
1.2 Project Option . . . . .	2
1.3 Internal Guide . . . . .	2
1.4 Technical Keywords (As per ACM Keywords) . . . . .	2
1.5 Problem Statement . . . . .	2
1.6 Abstract . . . . .	2
1.7 Goals and Objectives . . . . .	3
1.8 Relevant mathematics associated with the Project . . . . .	3
1.9 Names of Conferences / Journals where papers can be published . . . . .	7
1.10 Review of Conference/Journal Papers supporting Project idea . . . . .	7
1.11 Plan of Project Execution . . . . .	7
<b>2 Technical Keywords</b>	<b>8</b>
2.1 Area of Project . . . . .	9
2.2 Technical Keywords . . . . .	9
<b>3 Introduction</b>	<b>10</b>
3.1 Project Idea . . . . .	11
3.2 Motivation of the Project . . . . .	11
3.3 Literature Survey . . . . .	11
<b>4 Problem Definition and scope</b>	<b>12</b>
4.1 Problem Statement . . . . .	13
4.1.1 Goals and objectives . . . . .	13
4.1.2 Statement of scope . . . . .	13
4.2 Major Constraints . . . . .	13
4.3 Methodologies of Problem solving and efficiency issues . . . . .	13
4.4 Outcome . . . . .	14
4.5 Applications . . . . .	14
4.6 Hardware Resources Required . . . . .	14
4.7 Software Resources Required . . . . .	14
<b>5 Project Plan</b>	<b>15</b>
5.1 Project Estimates . . . . .	16
5.1.1 Reconciled Estimates . . . . .	16
5.1.2 Project Resources . . . . .	16

5.2	Risk Management w.r.t. NP Hard analysis . . . . .	16
5.2.1	Risk Identification . . . . .	16
5.2.2	Risk Analysis . . . . .	17
5.2.3	Overview of Risk Mitigation, Monitoring, Management . . . . .	17
5.3	Project Schedule . . . . .	19
5.3.1	Project task set . . . . .	19
5.3.2	Task network . . . . .	19
5.3.3	Timeline Chart . . . . .	21
5.4	Team Organization . . . . .	21
5.4.1	Team structure . . . . .	21
5.4.2	Management reporting and communication . . . . .	21
<b>6</b>	<b>Software requirement specification</b>	<b>22</b>
6.1	Introduction . . . . .	23
6.1.1	Purpose and Scope of Document . . . . .	23
6.1.2	Overview of responsibilities of Developer . . . . .	23
6.2	Usage Scenario . . . . .	23
6.2.1	User profiles . . . . .	24
6.2.2	Use-cases . . . . .	24
6.2.3	Use Case View . . . . .	25
6.3	Data Model and Description . . . . .	26
6.3.1	Data Description . . . . .	26
6.3.2	Data objects and Relationships . . . . .	26
6.4	Functional Model and Description . . . . .	26
6.4.1	Data Flow Diagram . . . . .	26
6.4.2	Activity Diagram: . . . . .	27
6.4.3	Non Functional Requirements: . . . . .	28
6.4.4	State Diagram: . . . . .	28
6.4.5	Design Constraints . . . . .	28
6.4.6	Software Interface Description . . . . .	29
<b>7</b>	<b>Detailed Design Document using Appendix A and B</b>	<b>30</b>
7.1	Introduction . . . . .	31
7.2	Architectural Design . . . . .	31
7.3	Data design (using Appendices A and B) . . . . .	31
7.3.1	Global data structure . . . . .	32
7.3.2	Temporary data structure . . . . .	32
7.3.3	Database description . . . . .	32
7.4	Component Design . . . . .	32
7.4.1	Class Diagram . . . . .	32
<b>8</b>	<b>Project Implementation</b>	<b>34</b>
8.1	Introduction . . . . .	35
8.2	Tools and Technologies Used . . . . .	35
8.3	Methodologies/Algorithm Details . . . . .	35
8.3.1	Algorithm 1/Pseudo Code . . . . .	35
8.3.2	Algorithm 2/Pseudo Code . . . . .	36

<b>CONTENTS</b>	<b>8</b>
8.4 Verification and Validation for Acceptance . . . . .	36
<b>9 Software Testing</b>	<b>37</b>
9.1 Type of Testing Used . . . . .	38
9.2 Test Cases and Test Results . . . . .	38
<b>10 Results</b>	<b>40</b>
10.1 Screen shots . . . . .	41
10.1.1 Login Page . . . . .	41
10.1.2 Admin Login . . . . .	42
10.1.3 Doctor Login . . . . .	43
10.2 Outputs . . . . .	44
10.2.1 Reports . . . . .	44
10.2.2 Messaging . . . . .	46
<b>11 Deployment and Maintenance</b>	<b>47</b>
11.1 Installation and un-installation . . . . .	48
<b>12 Conclusion and future scope</b>	<b>49</b>
<b>Annexure A References</b>	<b>51</b>
<b>Annexure B Laboratory assignments on Project Analysis of Algorithmic Design</b>	<b>53</b>
<b>Annexure C Project Planner</b>	<b>56</b>
C.0.1 Project Plan . . . . .	57
C.0.2 Project Plan . . . . .	58
<b>Annexure D Plagiarism Report</b>	<b>59</b>
D.0.1 Plagiarism Report . . . . .	60
<b>Annexure E Term-II Project Laboratory Assignments</b>	<b>61</b>
E.0.1 Code Snippet . . . . .	64
E.0.2 Code Snippet . . . . .	66
E.0.3 Code Snippet . . . . .	67
<b>Annexure F Information of Project Group Members</b>	<b>68</b>

# List of Figures

5.1	Timeline chart . . . . .	21
6.1	Use case diagram . . . . .	25
6.2	Data Flow diagram . . . . .	26
6.3	Activity diagram . . . . .	27
6.4	State transition diagram . . . . .	28
7.1	Architecture diagram . . . . .	31
7.2	Class Diagram . . . . .	33
10.1	Login Page . . . . .	41
10.2	Admin Login . . . . .	42
10.3	Doctor Login . . . . .	43
10.4	Reports . . . . .	44
10.5	Messaging . . . . .	46
C.1	Project Plan . . . . .	57
C.2	Project Plan . . . . .	58
D.1	Plagiarism Report . . . . .	60
E.1	Code Snippet . . . . .	64
E.2	Code Snippet . . . . .	66
E.3	Code Snippet . . . . .	67

# List of Tables

4.1	Hardware Requirements . . . . .	14
5.1	Risk Table . . . . .	17
5.2	Risk Probability definitions [?]	17
5.3	Risk Impact definitions [?]	17
5.4	Task Network . . . . .	20
6.1	Use Cases . . . . .	25
9.1	Test Cases . . . . .	39
B.1	IDEA Matrix . . . . .	54

# **CHAPTER 1**

## **SYNOPSIS**

## 1.1 Project Title

'APPLICATION FOR MANAGING A VETERINARY POLYCLINIC'

## 1.2 Project Option

INTERNAL PROJECT

## 1.3 Internal Guide

Prof. Pranjali Deshpande

## 1.4 Technical Keywords (As per ACM Keywords)

1. Computer Systems Organization
  - (a) COMPUTER-COMMUNICATION NETWORKS
    - i. Distributed Systems
      - A. Client/server
      - B. Distributed applications
      - C. Distributed databases
      - D. Network operating systems
      - E. Distributed file systems
      - F. Security and reliability issues in distributed applications

## 1.5 Problem Statement

To manage a veterinary polyclinic using customized web application.

## 1.6 Abstract

- A computerized database management system is designed and implemented to replace the existing traditional hand recorded system in a veterinary polyclinic. The system provides easy access to patient registration, diagnosis, appointment and report generation. The new system has higher efficiency in terms of time taken, cost and data security protection as compared to the traditional hand recorded system.

## 1.7 Goals and Objectives

- In veterinary polyclinics, the hospital records are managed manually. The records are hand-written and the reports are made by the doctors.
- We have proposed a system that will automate these manual tasks. Since, it is a web application, the doctors can access it remotely.
- The application coordinates the activities of different branches of the hospital and communicates between them.
- This will ease the processes of hospital management. Since the system is providing reports of the hospital records, much of the work will be reduced.

## 1.8 Relevant mathematics associated with the Project

### MATHEMATICAL MODEL

$S = \{s, e, X, Y, F, DD, NDD, MEMShared, corecnt, Success, Failure\}$

$s$  = Initial state

$e$  = End state

$X$  = Input state

$y$  = Output state

$F = F_{me} + F_{friend}$

$F_{me}$  = Main function

$F_{friend}$  = Support function

$DD$  = Deterministic Data

$NDD$  = Non Deterministic Data

$CPUcorecount$  = No. of cores in the CPU

$Success$  = Desired output is obtained

$Failure$  = Desired output is not obtained

$$F_{me} = \{F_{insert}, F_{query}\}$$

Formally given sets D1, D2 .... Dn.

A relation r is a subset of

D1 X D2 X D3 ..... Dn

Thus a relation is a set of n tuples (a1, a2.....an) where each ai ∈ Di

A1, a2 ....an are attributes.

Therefore, R= (a1, a2, a3,...,an) is a relation schema.

### Basic Operators

$\sigma$ : select       $\Pi$ : Project      U: Union

- : Set difference      x : Cartesian product

Notation :  $\sigma_p(r)$

p is called Selection predicate where p os the formula in prepositioned calculus consisting of terms connected by ^ and V (or).

$\sim$  (not).

Example of selection:

$\sigma_{animal} = OPD$

1) Finsert

$F_i \in F_{me}$

S= {Artificialinseminationregister, CalfBirthdetails, Infertilitydiagnosisregister, LabsTable, Liquidnitrogensupplydetails, OPD, Pregnancydiagnosis, Villagevaccinetable}

Artificialinseminationregister= {SrNo, AINo, OPDNo, Monthly, Progrssively, LLLAPLLadies, Cattletypeidentificationmark, ReproductiveOrgancondition, AI10daysproofoutcome, 280/310proofdate, Calfgender}

CalfBirthdetails= {SrNo, CBDNo, OPDNo, Dateofinsemination, Birthdate, Registrationno, Motherdetails, Comment}

Infertilitydiagnosisregister= {SrNo, IDRNo, OPDNo, Monthly, Progressively, CattleDescription, CheckupDate, Reproductiveorganscheckupresult, Checkupresult, Medicationdetails, Proofrecords,details}

LabsTable= {SrNo, LTNo, OPDNo, PDNO, Inference, Operation, AI, Cast, Vaccination, Health, Lab, SG, XRAY, EMR, PM, Total, Remark}

Liquidnitrogensupplydetails= {SrNo, LNSDNo, Date, Provider, LN2containertypeinch, LN2Containertypelitre, ReceivedLN2litre, TotalLN2inch, TotalLN2litre, TotalusedN2litr, RemainedN2inch, RemainedLN2litres, Comments}

OPD= {SrNo, MonthlyNo, OPDNo, Ownername, Owneroccupation, Ownercategorycaste, Owneraddress, Animal, Disease, DateIN, DateOUT, Horsecount, Cattlecount, Otherscount, Totalcount, comment}

Pregnancydiagnosis= {SrNo, OPDNo, PDNo, MonthlyNo, Date, Cow/Buffalo, ArtificaialInseminationdate, Checkupoutput}

Villagevaccinetable= {SrNo, VNo, OPDNo, Yearly, Monthly, Cowage, Cowcalf, Buffalo, Buffalocalf, Others, Total, Amount, Comment}

Consider table OPD

X = {{Iopd}, Ii}

Where

Iopd= {Iopd1, Iopd2, ..., Iopdn}

Where n is number of records.

Where

Isrno ∈ I<sup>+</sup>

I owner ∈ {a-z,A-Z}

I animaltype ∈ {words}

Finsert {I cust U Ii}

and, Y= {Iopd | Iopd = Iopd U Ii}

e= same as S

Success = data record inserted successfully.

## 2) Fquery

Fi ∈ Fme

S = {{ Artificialinseminationresgister, CalfBirthdetails, Infertilitydiagnosisregister, LabsTable, Liquidnitrogensupplydetails, OPD, Pregnancydiagnosis, Villagevaccinetable}}

Where

Artificialinseminationresgister= {SrNo, AINo, OPDNo, Monthly, Progrssively, LLLAPLadies, Cattletypeindentificationmark, ReproductiveOrgancondition, AI10daysproofoutcome, 280/310proofdate, Calfgender}

CalfBirthdetails= {SrNo, CBDNo, OPDNo, Dateofinsemination, Birthdate, Registrationno, Motherdetails, Comment}

Infertilitydiagnosisregister= {SrNo, IDRNo, OPDNo, Monthly, Progressively, CattleDescription, CheckupDate, Reproductiveorganscheckupresult. Checkupresult, Medicationdetails, Proofrecords,details}

LabsTable= {SrNo, LTNo, OPDNo, PDNO, Inference, Operation, AI, Cast, Vaccination, Health, Lab, SG, XRAY, EMR, PM, Total, Remark}

Liquidnitrogensupplydetails= {SrNo, LNSDNo, Date, Provider, LN2containertypeinch, LN2Containertypelitre, ReceivedLN2litre, TotalLN2inch, TotalLN2litre, TotalusedN2litr, RemainedN2inch, RemainedLN2litres, Comments}

OPD= {SrNo, MonthlyNo, OPDNo, Ownername, Owneroccupation, Ownercategorycaste, Owneraddress, Animal, Disease, DateIN, DsteOUT, Horsecount, Cattlecount, Otherscount, Totalcount, comment}

Consider table OPD

X = {{Iopd}, Ii}

Where

Iopd= {Iopd1, Iopd2, ...., Iopdn}

Where n is number of records.

Where

Isrno  $\in$  I<sup>+</sup>

I owner  $\in$  {a-z,A-Z}

I animaltype  $\in$  {words}

Therefore, Fquery = {op (OPD) p is predicate and p(Iopd)}

And Y= {Iopd | Iopd  $\in$  op (OPD)}

E= Same as S

Success= Selected data successfully displayed.

Ffriend = {F1}

F1 = username i x password i

Success F1 = True

= access granted

Failure F1 = False

= access denied

## 1.9 Names of Conferences / Journals where papers can be published

- [1] IEEE EMBS International Conference on Biomedical Engineering and Sciences
- [2] International Conference on Cloud Computing and Big Data.
- [3] International Conference on Robotics, Automation and Sciences.

## 1.10 Review of Conference/Journal Papers supporting Project idea

- [1] Database management system is currently implemented in the Radiology department in the Malaysian hospitals as mentioned in 'Computerized Brain Database Management System for Radiological Department in Hospital'[1].
- [2] According to Paulo Silva in 'Hospital database workload and fault forecasting'[2], a system maintains the hospital records. It is called the Hospital Information System (HIS). It consists of 4 processes: Care, Clinical process, Management and Resource. AIDA (Agency for Integration, Diffusion and Archive of Medical Information).
- [3] The ship building industry uses legacy system to introduce web based mobile technology. It is a cost involving process. SOAP based services have a standardized approach and can replace RPC, mentioned in 'Developing a Prototype of REST-based Database Application for Shipbuilding Industry: A Case Study'[3].
- [4] In 'Web database application system optimization research'[4], it has to face a large group of users. Therefore, system needs optimization. Optimization of server and Web application system optimization.

## 1.11 Plan of Project Execution

- Creating the database and required tables using Mysql.
- Developing the user interface of the application with respect to the veterinary polyclinic.
- Setting up the server and creating a server-client application.
- Check for connectivity of application.
- Generating the required reports.
- Building a messaging system.

## **CHAPTER 2**

## **TECHNICAL KEYWORDS**

## 2.1 Area of Project

Database Management Systems

## 2.2 Technical Keywords

1. Computer Systems Organization
  - (a) COMPUTER-COMMUNICATION NETWORKS
    - i. Distributed Systems
      - A. Client/server
      - B. Distributed applications
      - C. Distributed databases
      - D. Network operating systems
      - E. Distributed file systems
      - F. Security and reliability issues in distributed applications

# **CHAPTER 3**

## **INTRODUCTION**

### 3.1 Project Idea

- A computerized database management system is designed and implemented to replace the existing traditional largely hand-recorded system in a veterinary polyclinic in Jalna.

### 3.2 Motivation of the Project

- In veterinary polyclinics, the hospital records are managed manually. The records are hand-written and the reports are made by the doctors. We have proposed a system that will automate these manual tasks. Since it is a web application, the doctors can access it remotely. The application coordinates the activities of different branches of the hospital and communicates between them. This will ease the processes of hospital management. Since the system is providing reports of the hospital records, much of the work will be reduced.

### 3.3 Literature Survey

- [1] Database management system is currently implemented in the Radiology department in the Malaysian hospitals as mentioned in 'Computerized Brain Database Management System for Radiological Department in Hospital'[1].
- [2] According to Paulo Silva in 'Hospital database workload and fault forecasting'[2], a system maintains the hospital records. It is called the Hospital Information System (HIS). It consists of 4 processes: Care, Clinical process, Management and Resource. AIDA (Agency for Integration, Diffusion and Archive of Medical Information).
- [3] The ship building industry uses legacy system to introduce web based mobile technology. It is a cost involving process. SOAP based services have a standardized approach and can replace RPC, mentioned in 'Developing a Prototype of REST-based Database Application for Shipbuilding Industry: A Case Study'[3].
- [4] In 'Web database application system optimization research'[4], it has to face a large group of users. Therefore, system needs optimization. Optimization of server and Web application system optimization.

## **CHAPTER 4**

## **PROBLEM DEFINITION AND SCOPE**

## 4.1 Problem Statement

To design a web based system to manage the operations of veterinary polyclinic.

### 4.1.1 Goals and objectives

Goal and Objectives:

- In veterinary polyclinics, the hospital records are managed manually. The records are hand-written and the reports are made by the doctors.
- We have proposed a system that will automate these manual tasks. Since, it is a web application, the doctors can access it remotely.
- The application coordinates the activities of different branches of the hospital and communicates between them.
- This will ease the processes of hospital management. Since the system is providing statistics and reports of the hospital records, much of the work will be reduced.

### 4.1.2 Statement of scope

- Web application for the veterinary polyclinic that will generate timely reports. The application also includes the functionality of a messaging system wherein the patients are notified with their next visiting date and other information.

## 4.2 Major Constraints

- Good Internet Connection needed for insertion of data and the execution of queries.
- Not compatible with mobile phones.

## 4.3 Methodologies of Problem solving and efficiency issues

- The patients can be notified by using 2 technologies-
  - Sending text messages on mobile phones.
  - Sending Emails.

We have implemented a text messaging system because it is efficient and more friendly to our clients.

## 4.4 Outcome

- Reports are generated on the basis of the hospital data by applying time constraints.
- Text messages are sent to patients on time.

## 4.5 Applications

- The project is being specifically designed for a government veterinary poly-clinic in Jalna.

## 4.6 Hardware Resources Required

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	2 GHz	Intel Pentium(released since 2009)
2	RAM	3 GB	For Efficiency
3	Number of cores	2	For Client and server

Table 4.1: Hardware Requirements

## 4.7 Software Resources Required

Platform :

1. Operating System: OS Independant
2. IDE: Netbeans
3. Programming Language: HTML,CSS,JavaScript,PHP

## **CHAPTER 5**

## **PROJECT PLAN**

## 5.1 Project Estimates

### 5.1.1 Reconciled Estimates

#### 5.1.1.1 Cost Estimate

Suscription to Textlocal.in - Rs. 250 per month

#### 5.1.1.2 Time Estimates

- (a) 1 to 6 months : data acquiring, problem defining, normalization, designing the flow
- (b) 7 to 12 months : Implementation

### 5.1.2 Project Resources

1. People - Group members,Project Guide,Consultant at the Polyclinic(Jalna),Project Supervisors.
2. Hardware- 2 to 3 computers for client server implementation and concurrency analysis.
3. Software- Texmaker,Textlocal

## 5.2 Risk Management w.r.t. NP Hard analysis

Project developed by 4 team members who collectively performed different modules of the project. Project platforms include Linux OS (Fedora), Apache 2 server, HTML, CSS, MySQL, PHP for implementation of system.

### 5.2.1 Risk Identification

1. Have top software and customer managers formally committed to support the project? YES
2. Are end-users enthusiastically committed to the project and the system/product to be built? YES
3. Are requirements fully understood by the software engineering team and its customers? YES
4. Have customers been involved fully in the definition of requirements? YES
5. Do end-users have realistic expectations? YES
6. Does the software engineering team have the right mix of skills? YES
7. Are project requirements stable? YES

8. Is the number of people on the project team adequate to do the job? YES
9. Do all customer/user constituencies agree on the importance of the project and on the requirements for the system/product to be built? YES

### 5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Lack of Internet	15	High	High	High
2	Simultaneous access	10	Low	Low	Low
3	Data Theft(internally)	50	Low	Low	Low

Table 5.1: Risk Table

Probability	Value	Description
High	Probability of occurrence is	> 75%
Medium	Probability of occurrence is	26 – 75%
Low	Probability of occurrence is	< 25%

Table 5.2: Risk Probability definitions [?]

Impact	Value	Description
Very high	> 10%	Schedule impact or Unacceptable quality
High	5 – 10%	Schedule impact or Some parts of the project have low quality
Medium	< 5%	Schedule impact or Barely noticeable degradation in quality Low Impact on schedule or Quality can be incorporated

Table 5.3: Risk Impact definitions [?]

### 5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Risk ID	1
Risk Description	Unavailability of internet will disallow access to system.
Category	Development Environment.
Source	Software requirement Specification document.
Probability	Low
Impact	High
Response	Schedule impact. Discrepancy in the data stored.
Strategy	Switch to mobile internet.
Risk Status	Occurred

Risk ID	2
Risk Description	Schedule impact. Discrepancy in the data stored
Category	Development environment.
Source	Access synchronization.
Probability	Medium
Impact	High
Response	Scheduling access
Strategy	Pause Concurrent access
Risk Status	Identified

Risk ID	3
Risk Description	Internal accessors have complete rights to view data.
Category	Data access
Source	Database
Probability	Low
Impact	Medium
Response	Mitigate
Strategy	Enable access rights
Risk Status	Identified

## 5.3 Project Schedule

### 5.3.1 Project task set

Major Tasks in the Project stages are:

- Task 1: Data gathering for the design of system.
- Task 2: Defining scope and determining technologies for building the system.
- Task 3: Normalization of data and creating databases.
- Task 4: Building Client-side and Server-side modules.
- Task 5: Deployment.
- Task 6: Testing.

### 5.3.2 Task network

SR. No.	Task	Period	Expected	Target achieved
1	Gathering of hospital data	Dec 25th to Dec 31st 17	Structure of the system to be understood	Formats of records acquired
2	Restructuring of data	Jan 1st to Jan 15th 18	Minimal representation to be formed	Normalized tables obtained and created
3	Identifying different users	Jan 17th to Jan 20th 18	Defining views for different users	Different data, views for different users
4	Building the user interface	Jan 21st to Feb 20th	Building the front end of system	Different modules of UI built
5	Back end scripting	Feb 21st to March 15th 18	Connectivity between front and back end and transaction defining	Querying and connectivity established
6	Statistical Report generation	16th March to April 2nd 18	Generation of summary tables	Vaccination summary and lab summary created
7	Testing	April 3rd to May 15th	Testing all modules separately	Login module and admin module tested successfully

Table 5.4: Task Network

### 5.3.3 Timeline Chart

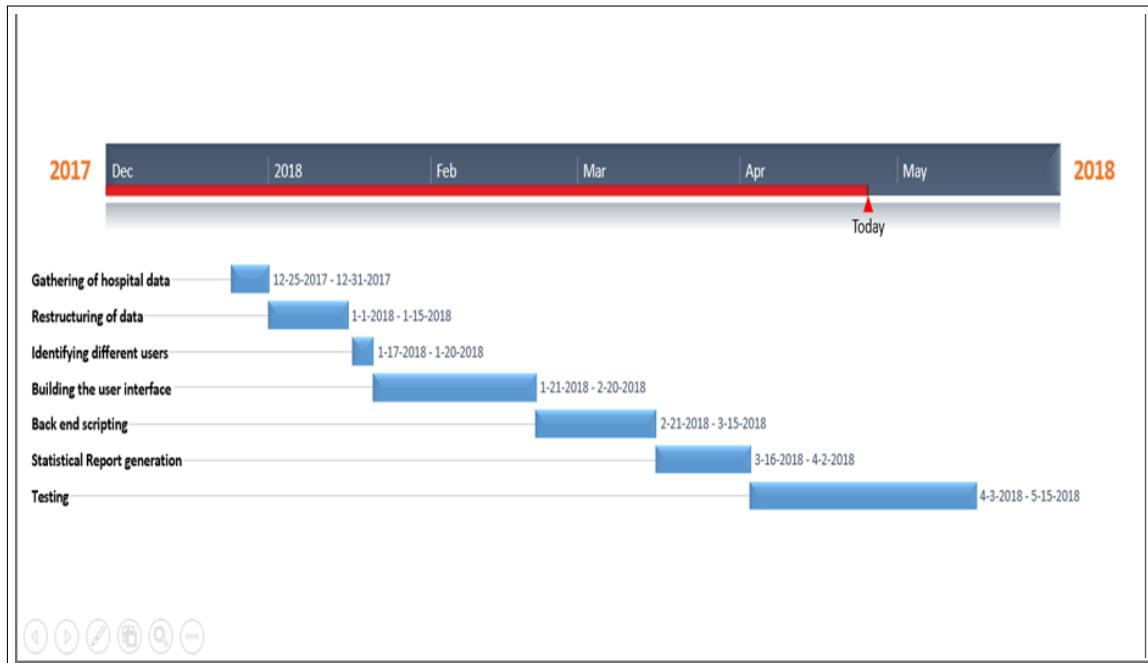


Figure 5.1: Timeline chart

## 5.4 Team Organization

The manner in which staff is organized and the mechanisms for reporting are noted.

### 5.4.1 Team structure

Following tasks did by the group members:

1st member: Data gathering and Documentation

2nd member: Normalization and Server side scripting

3rd member: Client side scripting and Documentation

4th member: Client side scripting and Database creation

### 5.4.2 Management reporting and communication

Internal Guide: Communication actively done mostly personally. The work done is also presented via emails.

Amongst group members: Communication done mostly by meeting personally and through emails and sharing on Drive.

**CHAPTER 6**

**SOFTWARE REQUIREMENT  
SPECIFICATION**

## 6.1 Introduction

Computed medical record database management is increasingly important as information and computer technology progress rapidly. Medical records contain treatment history and relevant experiences pertaining to the patients need to be documented accurately and efficiently. In many parts of the world clinical documentations are still being handwritten on forms and filed into paper medical records. The shortcomings of paper records are well identified, for example, illegibility and poorly organized records may lead to information gaps or inconsistencies, when used by another or even the same user later on. Hardcopies of documents are slow to be transferred around, with higher risk of being misplaced. In addition, when handled by multi-users in multi-locations, the quality of safekeeping can be significantly affected. The advent of computer technology has introduced enormous possibilities for electronic documentation and usage of electronic medical records. A computerized database management system is designed and implemented to replace the existing traditional hand recorded system in a veterinary polyclinic. The system provides easy access to patient registration, diagnosis, appointment and report generation. It can also collect, calculate and plot statistical data. The new system has higher efficiency in terms of time taken, cost and data security protection as compared to the traditional hand recorded system.

### 6.1.1 Purpose and Scope of Document

This document serves as a record for the software requirement specification related to the Veterinary polyclinic management. The scope of the document is to create a Web application for the veterinary polyclinic that will generate the final reports. The application might also include the functionality of a messaging system.

### 6.1.2 Overview of responsibilities of Developer

The developer should take into consideration most of the needs of the clients. The final software should meet most of the constraints and give satisfactory results as explained by the client. The system should be fault tolerant, and easy maintainable. It should pass most of the test cases. Overview of Functional Requirements: The developed application should allow-

- Easy insertion of data.
- Update the new data into the database.
- Produce the final reports.

## 6.2 Usage Scenario

This section provides various usage scenarios for the system to be developed.

### 6.2.1 User profiles

User personas and Characteristics:

- Doctor- Doctor will use the application to insert the details and look through the data.
- Administrator- Admin will manage the server machine. The admin will be responsible for managing and allotting the sign in details. Admin is also responsible for entering the patient details into the database.

### 6.2.2 Use-cases

- A doctor will do the following actions:
  1. Enter patient disease, day out and the type of treatment.
  2. Enter calf details.
  3. Enter pregnancy, artificial insemination, infertility diagnosis details.
- An assistant doctor will do the following actions:
  1. Edit liquid nitrogen stock table.
  2. Edit vaccine stock table.
  3. Edit labs table.
- The admin will do the following actions:
  1. Add staff.
  2. Delete staff.
  3. Edit staff information.
  4. Distribute username and password to the staffs.
  5. Enter the patient details into the database.

Sr No.	Use Case	Description	Actors	Assumptions
1	Use Case 1	Save the input data into the database	Admin	Connection to the database is successful
2	Use Case 2	Read data from the database	Admin,Doctor	Connection to the database is successful
3	Use Case 3	Generate patientno	Admin	Write operation is successful
4	Use Case 4	Generate reports	Doctor,System	Read, write operations are successful
5	Use Case 5	Send sms remainder	System	Successful read of dayin

Table 6.1: Use Cases

### 6.2.3 Use Case View

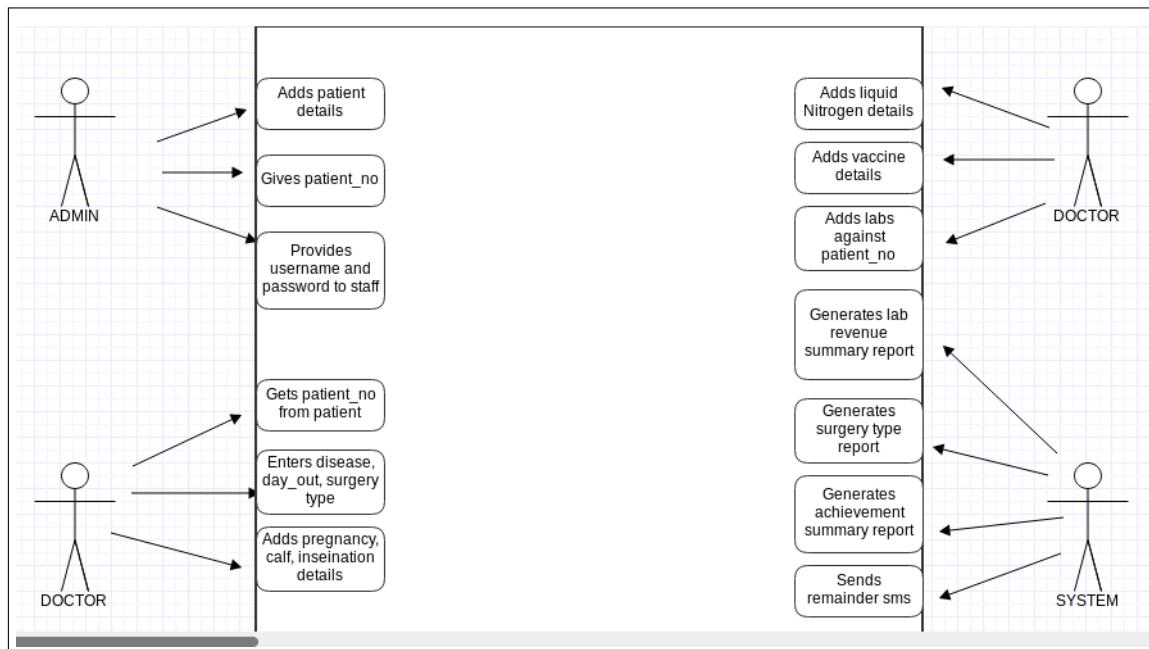


Figure 6.1: Use case diagram

## 6.3 Data Model and Description

### 6.3.1 Data Description

**Data Requirements:** The veterinary polyclinic data that is to be operated upon which includes all the patient details and details under different departments of the clinic. The developed application should allow the doctor to login to the application using his sign in details and let him insert the details of a new patient. Any new details should be updated directly to the database. Final reports should be generated as per the doctors requirements showing all the statistical data for brief analysis.

### 6.3.2 Data objects and Relationships

Data objects and their major attributes and relationships among data objects are described using an ERD- like form.

## 6.4 Functional Model and Description

The summary report, lab revenue reports and the surgery report will be generated from the following tables: 1. Patient table 2. Labs table

### 6.4.1 Data Flow Diagram

#### 6.4.1.1 Level 0 Data Flow Diagram

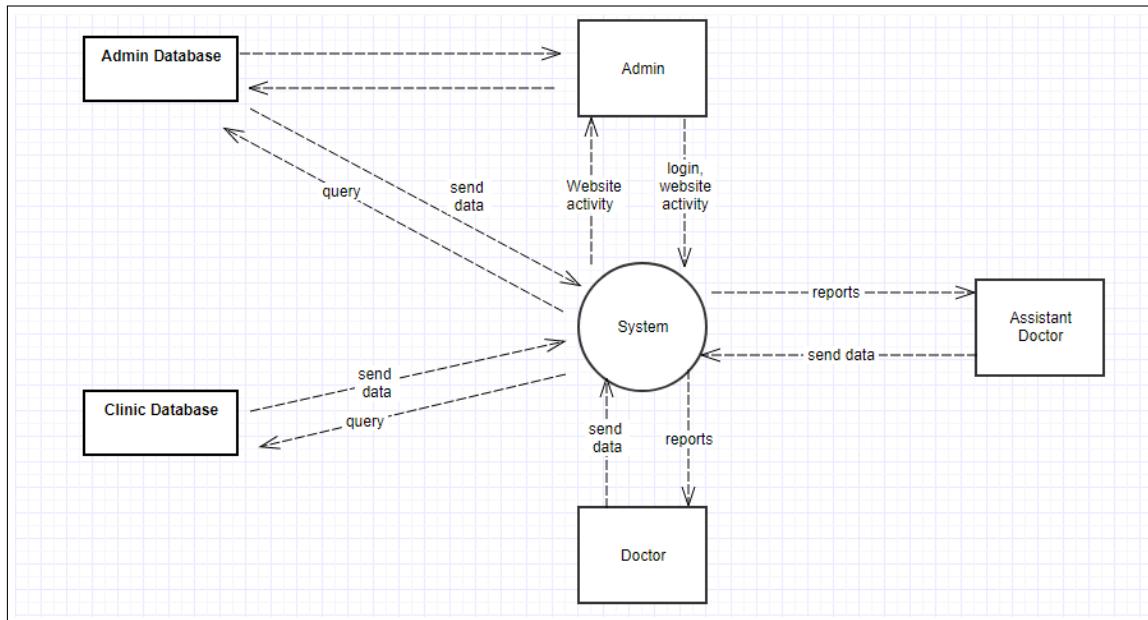


Figure 6.2: Data Flow diagram

#### 6.4.1.2 Level 1 Data Flow Diagram

#### 6.4.2 Activity Diagram:

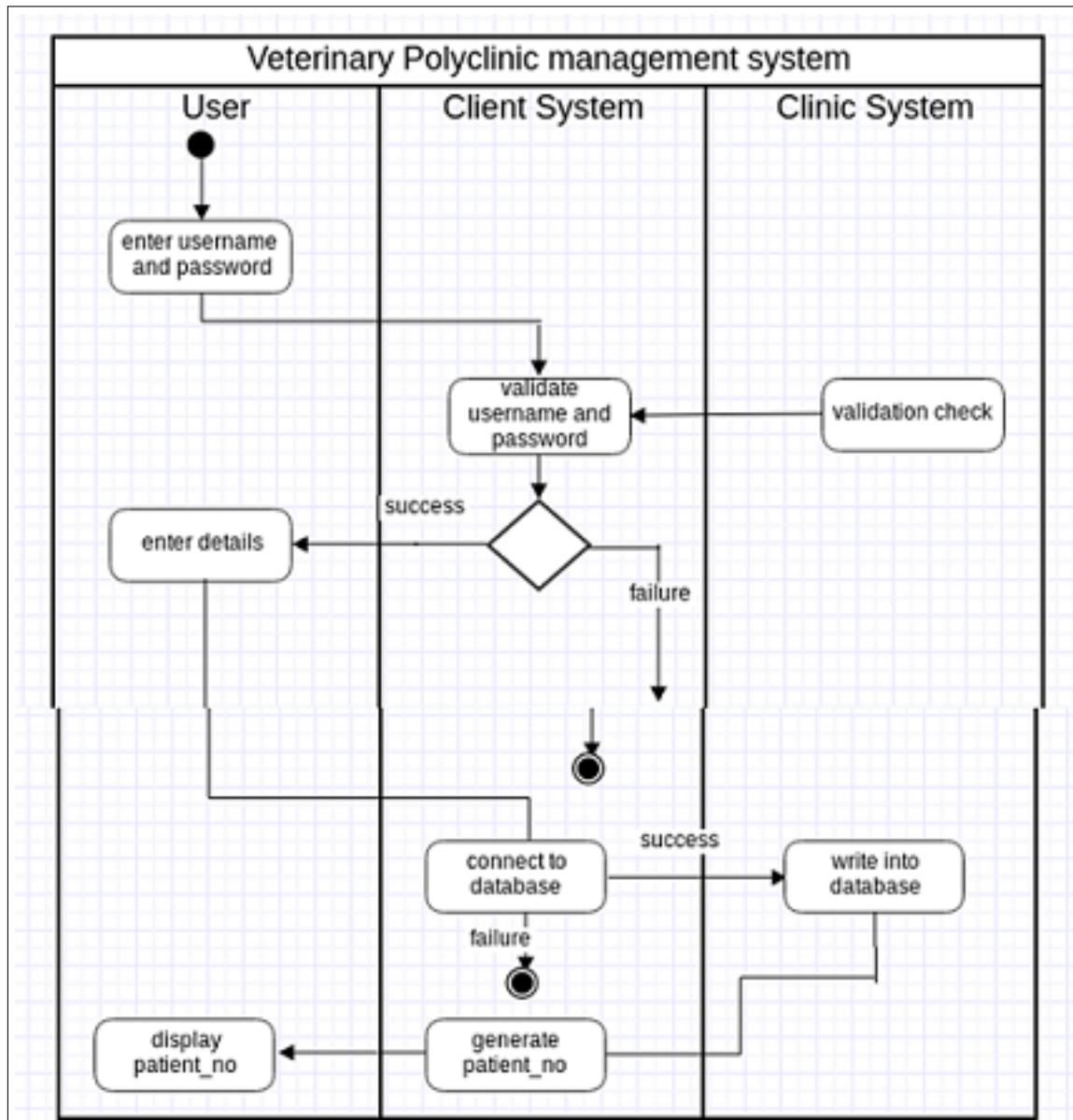


Figure 6.3: Activity diagram

### 6.4.3 Non Functional Requirements:

- Interface Requirements
- Performance Requirements
- Software quality attributes such as availability [ related to Reliability], modifiability [includes portability, reusability, scalability] , performance, security, testability and usability[includes self adaptability and user adaptability]

### 6.4.4 State Diagram:

State Transition Diagram

Fig.6.4 example shows the state transition diagram of Cloud SDK. The states are represented in ovals and state of system gets changed when certain events occur. The transitions from one state to the other are represented by arrows. The Figure shows important states and events that occur while creating new project.

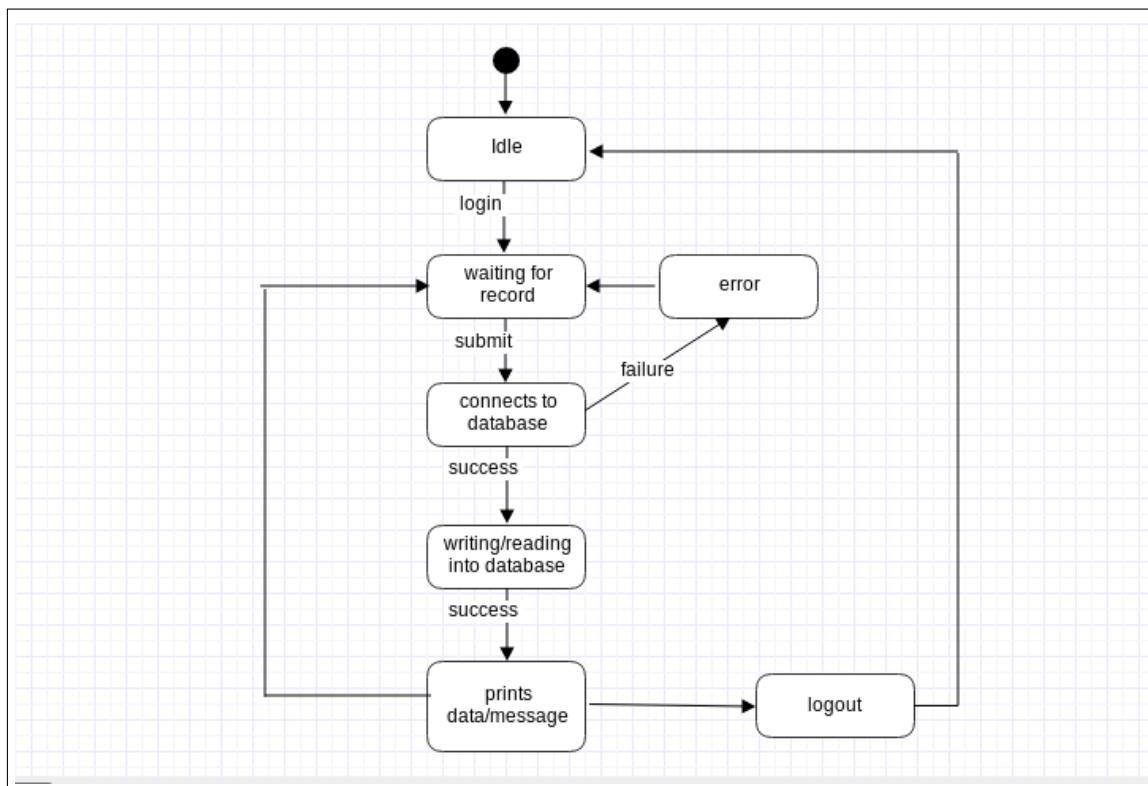


Figure 6.4: State transition diagram

### 6.4.5 Design Constraints

- The doctors should be educated on how to use the system,

- Hardware requirements: (i) Processor: Intel Core i3 (ii) OS Type: 64 bit (iii) RAM: 4 GB
- Software requirements: Netbeans IDE, Apache Tomcat, Mysql DBMS, Client side scripting: HTML, CSS Server side scripting: PHP

#### 6.4.6 Software Interface Description

1. Login Page:
  - (i) A login page for authorization.
  - (ii) Input details compared with the saved staff record.
2. Registration Module:
  - (i) For registering new client details.
  - (ii) Client details saved in a database.
3. Departments/inventory module:
  - (i) Mysql database for every department.
  - (ii) Records can be viewed, updated for all departments.
  - (iii) Inventory details are maintained in a separate database.
4. Report generation/statistics:
  - (i) Final reports generated.
5. Messaging System:
  - (i) Text messages sent to patients.
  - (ii) Textlocal.in used for subscription.

**CHAPTER 7**

**DETAILED DESIGN DOCUMENT**

**USING APPENDIX A AND B**

## 7.1 Introduction

In veterinary polyclinics, the hospital records are managed manually. The records are hand-written and the reports are made by the doctors. We have proposed a system that will automate these manual tasks. Since it is a web application, the doctors can access it remotely. The application coordinates the activities of different branches of the hospital and communicates between them. This will ease the processes of hospital management. Since the system is providing statistics and reports of the hospital records, much of the work will be reduced.

## 7.2 Architectural Design

A description of the program architecture is presented. Subsystem design or Block diagram, Package Diagram, Deployment diagram with description is to be presented.

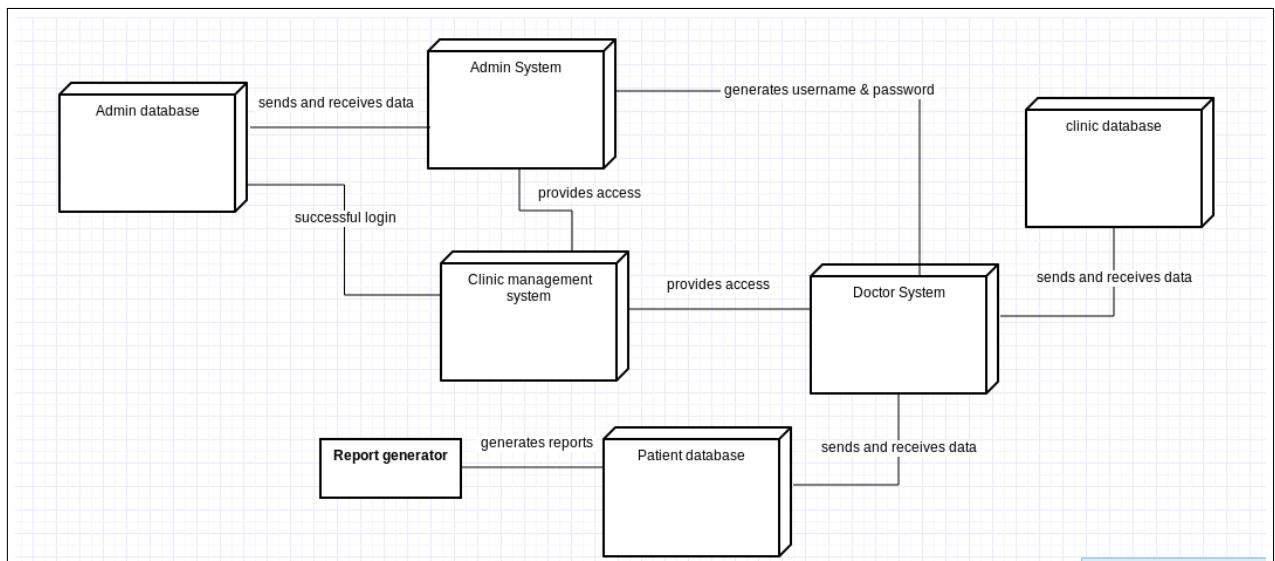


Figure 7.1: Architecture diagram

## 7.3 Data design (using Appendices A and B)

Quality Attributes:

- Design Qualities: Conceptual Integrity, Maintainability, Reusability
- Runtime Qualities: Availability, Performance
- System Qualities: Supportability, Testability

### **7.3.1 Global data structure**

Doctors page and admin page will be available at all times.

### **7.3.2 Temporary data structure**

Interim php files are created for handling the connectivity and querying.

### **7.3.3 Database description**

1. Patient table: Stores the details of patients like ownername, address, phoneno etc.
2. ai table: Stores the details of artificial insemination.
3. infertility table: Stores the infertility table details.
4. labs table: stores the details about the labs that are used by the patients.
5. calfdetails table: stores the details about the calf birth.
6. vaccination: keeps the records of vaccination.
7. pregnancy table: stores the information about the pregnancy details.
8. liquid nitrogen table: stores the stock information of the liquid nitrogen.
9. admin: stores the login details of the staff

## **7.4 Component Design**

Class diagrams, Interaction Diagrams, Algorithms. Description of each component description required.

### **7.4.1 Class Diagram**

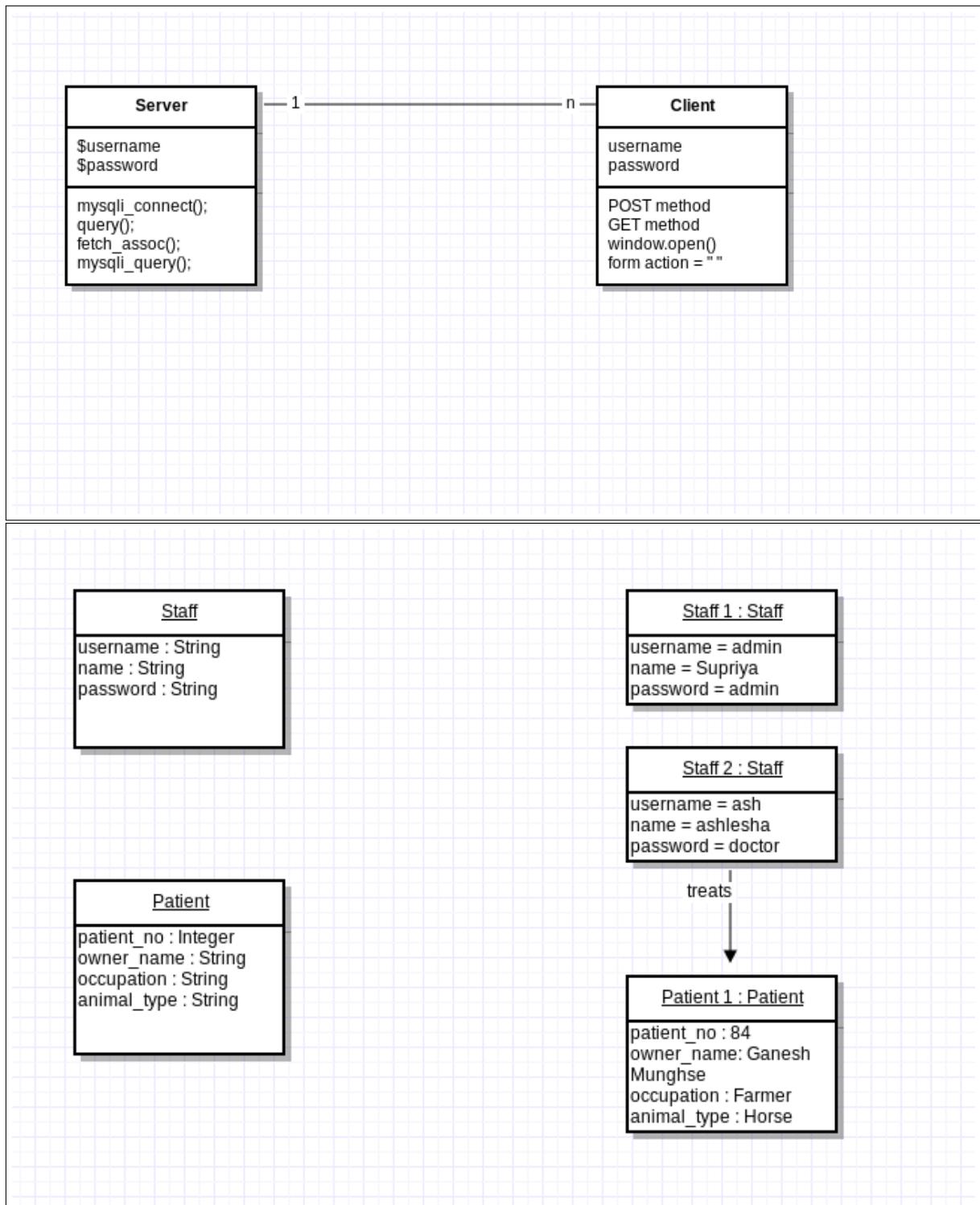


Figure 7.2: Class Diagram

# **CHAPTER 8**

# **PROJECT IMPLEMENTATION**

## 8.1 Introduction

We have implemented this project using various tools and technology like HTML, CSS for client side scripting, PHP for server side scripting, Apache 2 server that works on the central system and MySQL database.

## 8.2 Tools and Technologies Used

1. HTML: HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages. A web form or HTML form on a web page allows a user to enter data that is sent to a server for processing. Forms can resemble paper or database forms because web users fill out the forms using checkboxes, radio buttons, or text fields. Used for writing forms which form the basis of data storing from the user.
2. CSS: A CSS framework is a pre-prepared software framework that is meant to allow for easier, more standards-compliant web design using the Cascading Style Sheets language. Most of these frameworks contain at least a grid. More functional frameworks also come with more features and additional JavaScript based functions, but are mostly design oriented and unobtrusive. This differentiates these from functional and full JavaScript frameworks. Used for styling of the User Interface.
3. PHP: Hypertext Preprocessor (or simply PHP) is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. Used for back end for connectivity and transactions.
4. MySQL: MySQL is an open-source relational database management system (RDBMS). MySQL is a central component of the LAMP open-source web application software stack (and other "AMP" stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python". MySQL is also used in many high-profile, large-scale websites, including Google (though not for searches), Facebook, Twitter, Flickr, and YouTube. Used for storing the main data for the system.

## 8.3 Methodologies/Algorithm Details

### 8.3.1 Algorithm 1/Pseudo Code

1. Start

2. Enter the URL
3. Login as admin
4. Case
  - 4.1 Enter new patient
  - 4.2 Enter details
  - 4.3 Submit
  - 4.4 Update database
5. View details
  - 5.1 View all the patients
6. Enter new doctor
  - 6.1 Add new doctor details
  - 6.2 Create new entry
7. Logout

### 8.3.2 Algorithm 2/Pseudo Code

1. Start
2. Enter the URL
3. Login as doctor
4. Case
  - 4.1. Select from different departments
  - 4.2. Enter patient details
  - 4.3. Submit
5. Calculate fees
6. View reports
7. Logout

## 8.4 Verification and Validation for Acceptance

This system has been built based on the requirement of the Polyclinic on the lines of their current system which is inefficient. Our system caters to the needs of the polyclinic thereby providing additional functions which thoroughly improves the quality and quantity of the deliverables.

# **CHAPTER 9**

# **SOFTWARE TESTING**

## 9.1 Type of Testing Used

There are different types of testing we use in our project. Some testing types are as follows:

1. UNIT TESTING: Unit testing is the testing in which we do an individual unit or group of related units. It falls under the class of white box testing. It is often done by the programmer to test that the unit she has implemented is producing expected output against given input. So here we do the individual function testing.
2. INTEGRATION TESTING: Integration testing is testing in which a group of components are combined to produce output. Also, the interaction between software and hardware is tested in integration testing if software and hardware components have any relation. It may fall under both white box testing and black box testing.
3. FUNCTIONAL TESTING: Functional testing is the testing to ensure that the specified functionality required in the system requirements works. It falls under the class of black box testing.
4. SYSTEM SYSTEMS: System testing is the testing to ensure that by putting the software in different environments (e.g., Operating Systems) it still works. System testing is done with full system implementation and environment. It falls under the class of black box testing.
5. PERFORMANCE TESTING: Performance testing is the testing to assess the speed and effectiveness of the system and to make sure it is generating results within a specified time as in performance requirements. It falls under the class of black box testing. It checks the performance of our system.
6. USABILITY TESTING: Usability testing is performed from the perspective of the client, to evaluate how the GUI is user-friendly? How easily can the client learn? After learning how to use, how proficiently can the client perform? How pleasing is it to use its design? This falls under the class of black box testing.
7. ACCEPTANCE TESTING: Acceptance testing is often done by the customer to ensure that the delivered product meets the requirements and works as the customer expected. It falls under the class of black box testing.

## 9.2 Test Cases and Test Results

Test case No.	Description	Precondition	Expected result	actual result
TC1	To check whether user has logged in	User should be on login page	Login Successful	Login Successful
TC2	Check user-name and password are valid or not	User should be on login page	Display Error message (Wrong user-name or password)	Display Error message (Wrong user-name or password)
TC3	Check all the fields are filled or not	All the fields should be properly filled	Entered data is correct	Entered data is correct
TC4	To check if the count is given against the correct type of animal	Provide separate input methods for each animal type	Data provided successfully	Data provided successfully
TC5	Check whether patients data is successfully stored in the database or not	Data should be stored in the database	Data stored successfully	Data stored successfully
TC6	Performance testing for page loading time	Page shall be loaded within acceptable time range	Page loaded in minimum response time	Page loaded in minimum response time
TC7	Performance testing for database query execution time	Database query execution time shall be within acceptable time range	File retrieval is fast	File retrieval is fast

Table 9.1: Test Cases

# **CHAPTER 10**

# **RESULTS**

## 10.1 Screen shots

### 10.1.1 Login Page

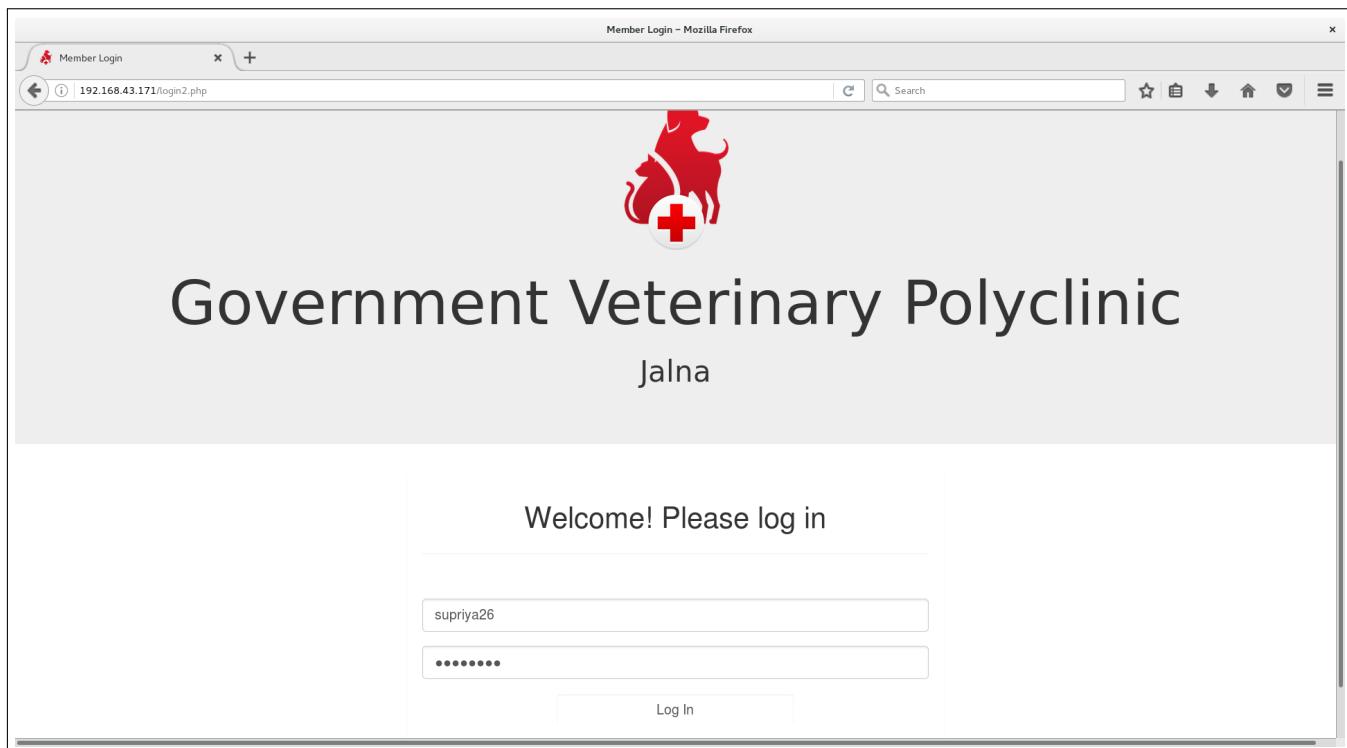
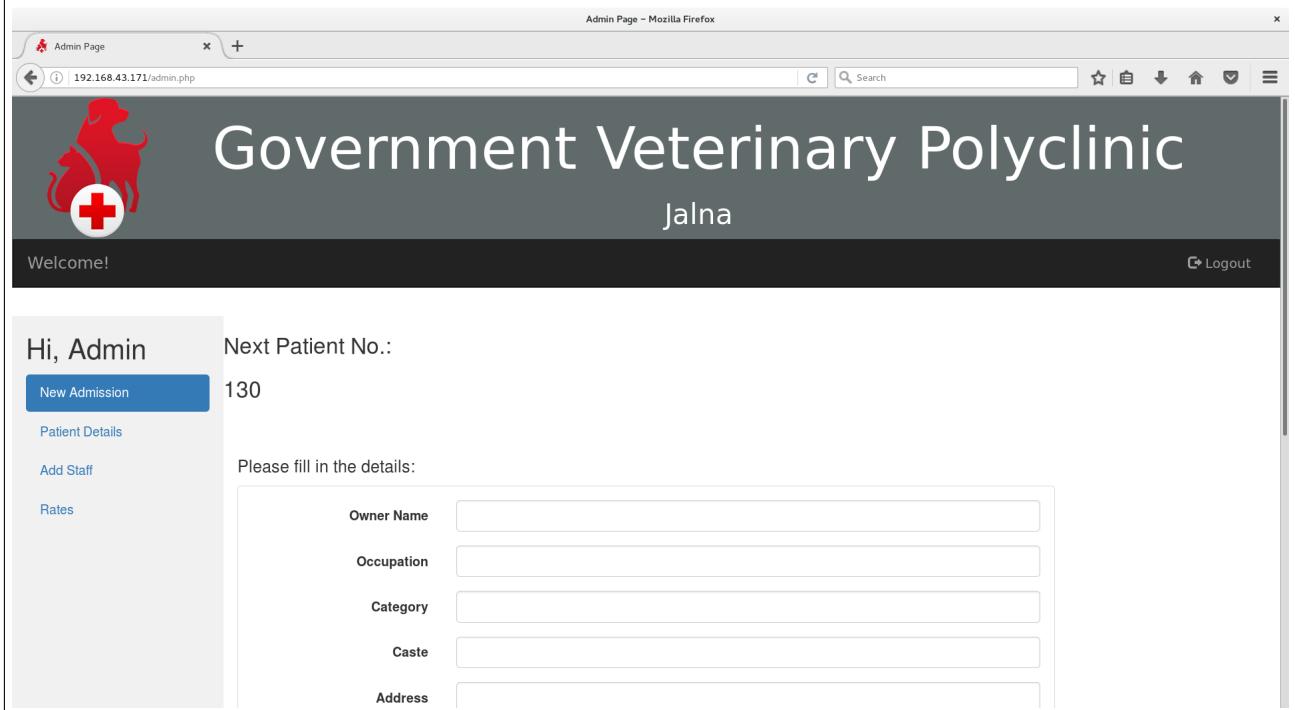
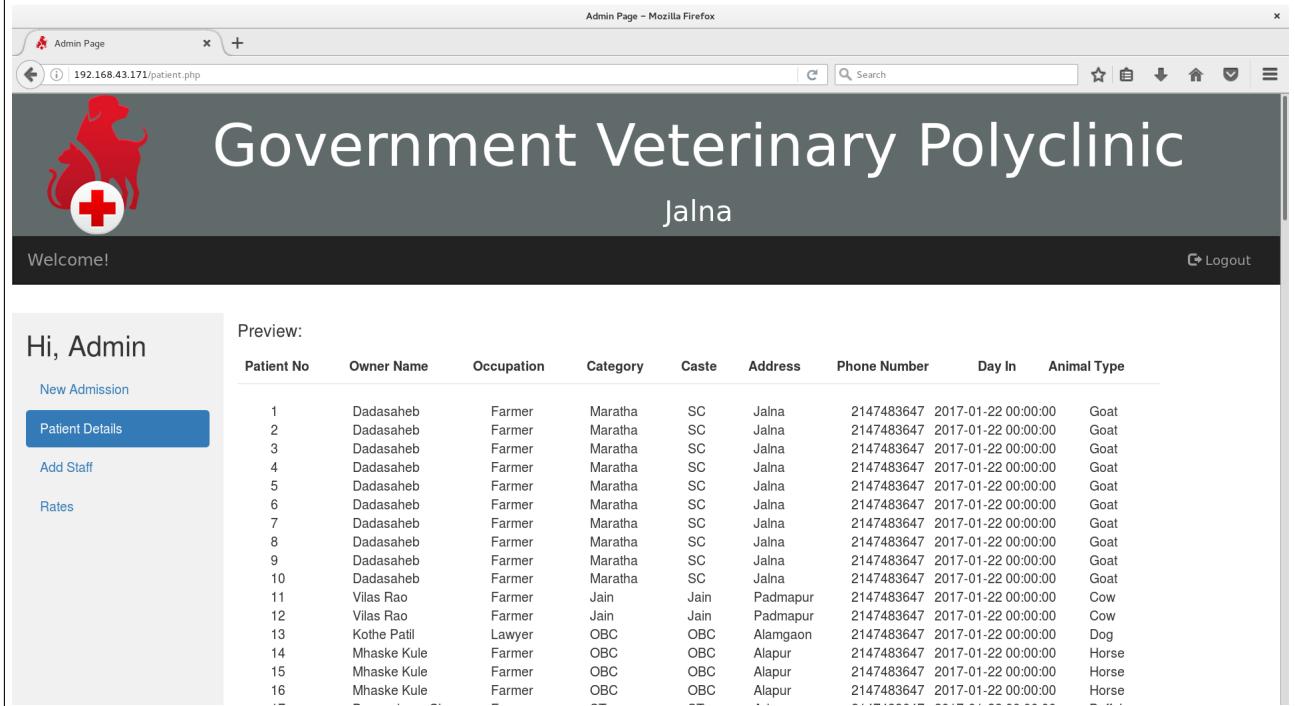


Figure 10.1: Login Page

### 10.1.2 Admin Login



The screenshot shows the 'Admin Page' interface for the 'Government Veterinary Polyclinic' located at 192.168.43.171/admin.php. The page features a logo of two red dogs with a white cross on their chest. The title 'Government Veterinary Polyclinic' and location 'Jalna' are prominently displayed. A 'Welcome!' message is on the left, and a 'Logout' link is on the right. The main area contains a form for 'New Admission' with fields for 'Owner Name', 'Occupation', 'Category', 'Caste', and 'Address'. The 'Owner Name' field is populated with '130'.

The screenshot shows the 'Patient Page' interface for the 'Government Veterinary Polyclinic' located at 192.168.43.171/patient.php. The layout is similar to the admin page. The 'Welcome!' message is on the left, and a 'Logout' link is on the right. The main area displays a table titled 'Preview:' showing a list of patients with columns for Patient No, Owner Name, Occupation, Category, Caste, Address, Phone Number, Day In, and Animal Type. The data is identical to the form data shown in the previous screenshot.

Patient No	Owner Name	Occupation	Category	Caste	Address	Phone Number	Day In	Animal Type
1	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
2	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
3	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
4	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
5	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
6	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
7	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
8	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
9	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
10	Dadasaheb	Farmer	Maratha	SC	Jalna	2147483647	2017-01-22 00:00:00	Goat
11	Vilas Rao	Farmer	Jain	Jain	Padmapur	2147483647	2017-01-22 00:00:00	Cow
12	Vilas Rao	Farmer	Jain	Jain	Padmapur	2147483647	2017-01-22 00:00:00	Cow
13	Kothe Patil	Lawyer	OBC	OBC	Alamgaon	2147483647	2017-01-22 00:00:00	Dog
14	Mhaske Kule	Farmer	OBC	OBC	Alapur	2147483647	2017-01-22 00:00:00	Horse
15	Mhaske Kule	Farmer	OBC	OBC	Alapur	2147483647	2017-01-22 00:00:00	Horse
16	Mhaske Kule	Farmer	OBC	OBC	Alapur	2147483647	2017-01-22 00:00:00	Horse

Figure 10.2: Admin Login

### 10.1.3 Doctor Login

The image displays two screenshots of a web application for a 'Government Veterinary Polyclinic' located in 'Jalna'. Both screenshots are taken from Mozilla Firefox, showing the URL 192.168.43.171/doc.php.

**Screenshot 1 (Top): Patient Details**

This screenshot shows a form for entering patient details. The header includes the clinic's logo (a red silhouette of a dog with a white cross) and name ('Government Veterinary Polyclinic Jalna'). A 'WELCOME, DOCTOR' message is at the top left, and a 'Logout' link is at the top right. On the left, a sidebar menu lists: PATIENT DETAILS (highlighted in blue), PATIENT LIST, VACCINATION, INSEMINATION, PREGNANCY, INFERTILITY, CALF BIRTH, NITROGEN SUPPLY, LABS, and REPORTS. The main content area has a heading 'Please fill Patient No.:'. It contains a text input field labeled 'Patient No.' and a 'SUBMIT' button. Below this is a section labeled 'Details:' with two empty text input fields, one for 'Patient No.' and one for 'Disease'.

**Screenshot 2 (Bottom): Reports**

This screenshot shows a report generation interface. The header and sidebar are identical to the first screenshot. The main content area has a heading 'REPORTS:' followed by a bulleted list of report types: • SURGERY, • LAB REVENUE, • VACCINATION, and • NITROGEN SUPPLY.

Figure 10.3: Doctor Login

## 10.2 Outputs

### 10.2.1 Reports

The figure consists of two vertically stacked screenshots of a web-based veterinary management system. Both screenshots show a header with the logo of a red dog with a white cross, the text "Government Veterinary Polyclinic Jalna", and a "WELCOME, DOCTOR" message. A "Logout" link is in the top right corner.

**Screenshot 1: Surgery Report (surgery.php)**

This screenshot shows a report for surgeries. It includes fields for "Date from" and "Date To" (both YYYY-MM-DD), and a "Submit" button. Below these are two tables:

Animal Type	Minor Surgery
Cow	18
Goat	62
Horse	17
others	1

Animal Type	Major Surgery
buffalo	1
cow	1

**Screenshot 2: Revenue Report (revenue.php)**

This screenshot shows a report for revenue. It includes fields for "Date from" and "Date To" (both YYYY-MM-DD), and a "Submit" button. Below these is a table:

Lab	Count	Cost
health	59	1770
sonography	54	2160
xray	52	2600
emr	63	630
post_mortem	46	920

A total revenue of 8080 is displayed at the bottom.

Figure 10.4: Reports

Doctor Page - Mozilla Firefox

192.168.43.171/vacrep.php



# GOVERNMENT VETERINARY POLYCLINIC

## Jalna

WELCOME, DOCTOR [Logout](#)

[PATIENT DETAILS](#)

[PATIENT LIST](#)

[VACCINATION](#)

[INSEMINATION](#)

[PREGNANCY](#)

[INFERTILITY](#)

[CALF BIRTH](#)

[NITROGEN SUPPLY](#)

[LABS](#)

**REPORTS**

Date from

Date To

Submit

Animal	Vaccines	Cost
Cow	510	40800
Buffalo	531	42480
Goat	118	9440
Others	238	19040

Total Revenue: 111760

Total Vaccines: 1397

Doctor Page - Mozilla Firefox

192.168.43.171/nit.php



# Government Veterinary Polyclinic

## Jalna

WELCOME, DOCTOR [Logout](#)

[PATIENT DETAILS](#)

[PATIENT LIST](#)

[VACCINATION](#)

[INSEMINATION](#)

[PREGNANCY](#)

[INFERTILITY](#)

[CALF BIRTH](#)

[NITROGEN SUPPLY](#)

[LABS](#)

**REPORTS**

Date from

Date To

Submit

Total Received in litres: 4909.47

Total Used in litres: 4837.67

Total Remaining in litres: 71.8

### 10.2.2 Messaging

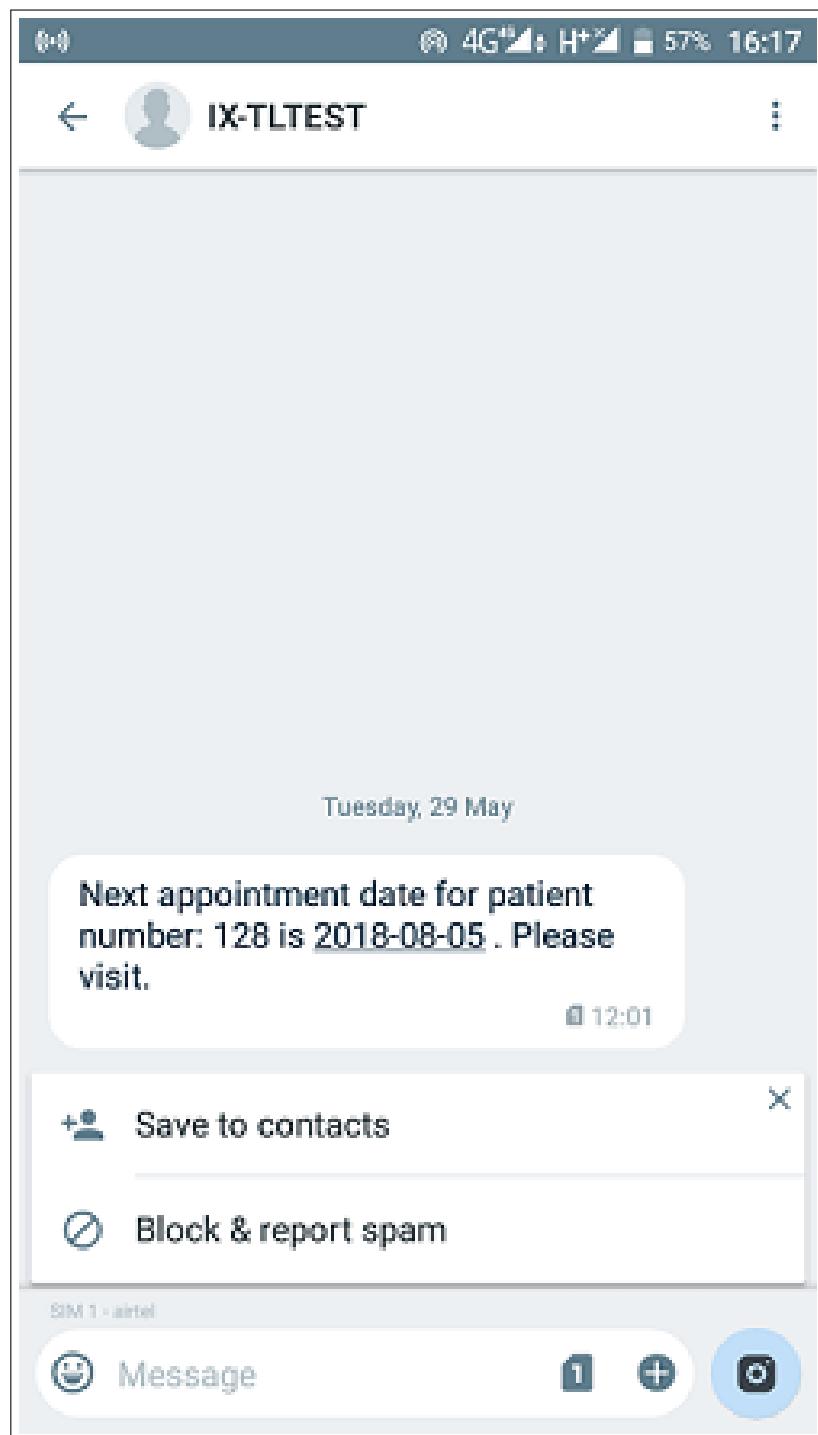


Figure 10.5: Messaging

# **CHAPTER 11**

## **DEPLOYMENT AND MAINTENANCE**

## 11.1 Installation and un-installation

1. The server machine need Apache HTTPD Server installed.
2. The server should have MYsql database system installed.
3. Subscription to Textlocal.in.
4. Subscription to URL domain(optional)

## **CHAPTER 12**

## **CONCLUSION AND FUTURE SCOPE**

## 1. CONCLUSION :

Thus, the digital system gets implemented and the traditional hand written system gets atomised. Separate user interfaces for doctors and administrator is created to simplify the work. The project also considers the Inventory management aspect of the veterinary polyclinic. A systematic approach is followed for the process starting from insemination stage, infertility, pregnancy diagnosis and finally the calf birth. The reports get generated as per the users requirements which reduce a lot of work for the doctors. An automated messaging system is implemented which helps in notifying the patients about their next infertility check up which is due in 3 months. Hence, this way our project becomes the first web application for a veterinary polyclinic in Maharashtra. Hence, this is our small contribution to the Digital India campaign.

## 2. FUTURE SCOPE :

1. A mobile application can be developed for the same.
2. Medical Inventory can be added as a part of clinical dbms.
3. The patients can make appointments using the system will be a part of future development.

**ANNEXURE A**

**REFERENCES**

- [1] K. S. Sim, T. K. Kho, F. S. Abas, F. F. Ting, V. Teh and C. S. Ee, Computerized Brain Database Management System for Radiological Department in Hospital, International Conference of Robotics, Automation and Sciences(ICORAS), 2016.
- [2] Paulo Silva, Cesar Quintas, Julio Duarte, Manuel Santos, Jose Neves, Antonio Abelha and Jose Machado, Hospital database workload and fault forecasting, IEEE-EMBS Conference on Biomedical Engineering and Sciences, pp. 63-68, 2012.
- [3] Jeong-cheol Jeon, Jaehwa Chung, Developing a Prototype of REST-based Database Application for Shipbuilding Industry: A Case Study, International Conference on Platform Technology and Service (PlatCon), pp. 1-6, 2017.
- [4] Cheng Zhi, The Web database application system optimization research, Seventh International Conference on Measuring Technology and Mechatronics Automation, pp. 1329-1332, 2015.

**ANNEXURE B**

**LABORATORY ASSIGNMENTS ON**

**PROJECT ANALYSIS OF**

**ALGORITHMIC DESIGN**

## ANNEXURE B. LABORATORY ASSIGNMENTS ON PROJECT ANALYSIS OF ALGORITHMIC

- To develop the problem under consideration and justify feasibility using concepts of knowledge canvas and IDEA Matrix.

I	D	E	A
Increase : Efficiency	Drive :Disadvantages in hand-written records	Educate :Educate the doctors on how to use the system.	Accelerate : Report generation process
Improve :Storage, Time efficiency, Fetch time,	Deliver :Automated reports, Web application.	Evaluate :Testing for errors	Artists :Doctors, Admin
Ignore :Internet unavailability	Decrease :Search time.	Eliminate :Redundancy	Avoid :Arithmetic errors

Table B.1: IDEA Matrix

- Project problem statement feasibility assessment using NP-Hard, NP-Complete or satisfy ability issues using modern algebra and/or relevant mathematical models.

Application for managing veterinary polyclinic

1. Aim: Project problem statement feasibility assessment using NP-Hard, NP-Complete or satisfiability issues using modern algebra.
2. NP class : A problem is assigned to the NP (nondeterministic polynomial time) class if it is solvable in polynomial time by a nondeterministic Turing machine. A P-problem (whose solution time is bounded by a polynomial) is always also NP.
3. NP-Complete Class: In computational complexity theory, an NP-complete decision problem is one which is in the NP complexity class and which is also NP-hard. In this context, NP stands for "nondeterministic polynomial time". The set of NP-complete problems is often denoted by NP-C or NPC.
4. NP Hard Class: The complexity class of decision problems that are intrinsically harder than those that can be solved by a nondeterministic Turing machine in polynomial time. P Class: It contains all decision problems that can be solved by a deterministic Turing machine using a polynomial amount of computation time, or polynomial time.
5. Complexity of query processing

In practice, we deal with the data complexity of the Query Evaluation Problem for Relational Calculus, because we typically have a small fixed collection of queries to answer (while the database instances vary). The

## *ANNEXURE B. LABORATORY ASSIGNMENTS ON PROJECT ANALYSIS OF ALGORITHMIC*

complexity of query evaluation problem depends upon two factors- the query complexity and the data complexity. The combined complexity is given as NP-complete.

### 6. Client-server application

the proposed system is a 3-tier application. It involves client server communication. The communication problems are reduced to NP-complete problems using Graph theory.

Thus the overall system complexity becomes NP complete.

**ANNEXURE C**

**PROJECT PLANNER**

### C.0.1 Project Plan

The figure consists of two screenshots of the Wrike platform, illustrating the project plan for 'Veterinary Polyclinic Database Management'.

**Screenshot 1: List View**

This view shows a detailed list of tasks across various sprints:

Task	Start Date	Due Date	Duration	Status	Assigned To
Completed Sprints				Completed	Pavithra T
Sprint 00				In Progress	Pavithra T
Current Sprint				In Progress	Pavithra T
Develop front-end logic	Feb 21, 2018	Mar 15, 2018	23d	In Progress	Pavithra T
Develop front-end markup	Jan 21, 2018	Feb 01, 2018	12d	In Progress	Pavithra T
Develop server-side handle	Dec 25, 2017	Dec 31, 2017	7d	In Progress	Pavithra T
Prepare for release	Jan 17, 2018	Jan 19, 2018	3d	In Progress	Pavithra T
Next Sprint	Mar 16, 2018	Apr 02, 2018	18d	In Progress	Pavithra T
Back end scripting	Jan 01, 2018	Jan 11, 2018	11d	In Progress	Pavithra T
Building a user interface	Apr 03, 2018	May 15, 2018	43d	In Progress	Pavithra T
Gathering of Hospital data				Planned	Pavithra T
Identifying different users				Planned	Pavithra T
Report generation				Planned	Pavithra T
Restructuring of Data				Planned	Pavithra T
SMS module and testing				Planned	Pavithra T

**Screenshot 2: Timeline View**

This view provides a visual timeline from May 20-26 to Jun 10-16, showing task progress and dependencies:

- May 20-26:** Tasks include 'Gathering of Hospital data', 'Restructuring of Data', 'Identifying different users', 'Building a user interface', 'Back end scripting', 'Report generation', and 'SMS module and testing'.
- May 27-Jun 2:** Tasks include 'Collect feedback', 'Final testing & release preparation', and 'Release to production'.
- Jun 3-9:** Task 'Sprint 00' is shown.
- Jun 10-16:** Task 'Completed Sprints' is shown.

Figure C.1: Project Plan

## C.0.2 Project Plan

The figure consists of two screenshots of the Wrike application interface.

**Screenshot 1: Dashboard Overview**

This screenshot shows the 'Veterinary Polyclinic Database Management' project dashboard. It features a circular progress chart indicating 15 tasks in total, with 7 backlogged, 1 completed, and 7 overdue. Below the chart, there are sections for 'TASKS' (14 unassigned, 7 overdue) and 'STATISTICS' (936h duration).

**Screenshot 2: Task List**

This screenshot shows the task list for the same project. The tasks are categorized by due date: Overdue, Today, and Later. The tasks listed are:

- Overdue: Getting Started (Welcome to Wrike!), Jun 04
- Today: Collect feedback (Backlog), New
- Later: Develop front-end logic (Current Sprint), Development; Develop front-end markup (Current Sprint), Development; Develop server-side handle (Current Sprint), Testing; Final testing & release preparation (Backlog), New; Prepare for release (Current Sprint), Reopened; Release to production (Backlog), New

At the bottom of the task list, it says 'Total: 15 tasks'.

Figure C.2: Project Plan

**ANNEXURE D**

**PLAGIARISM REPORT**

## Plagiarism report

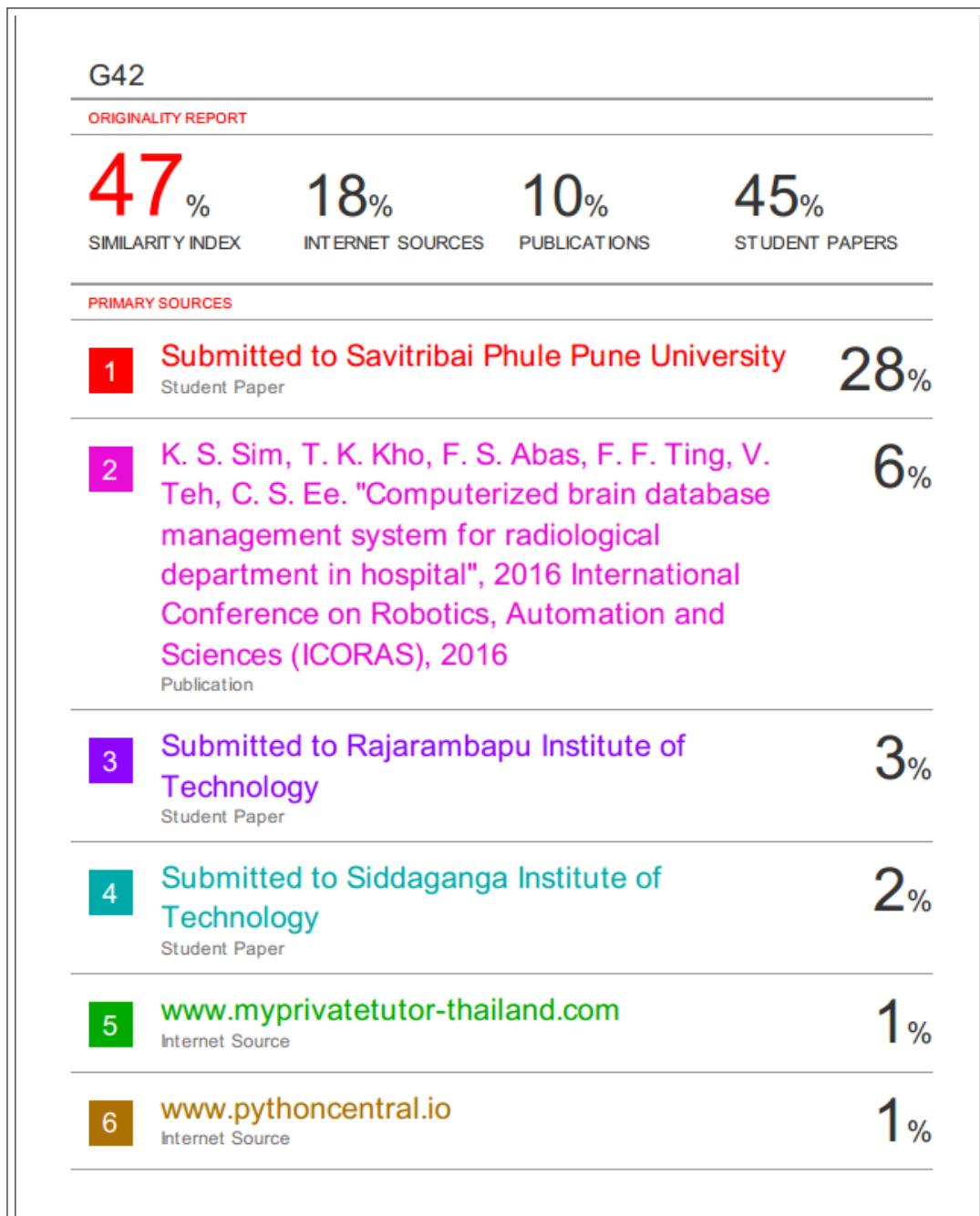
**D.0.1 Plagiarism Report**

Figure D.1: Plagiarism Report

**ANNEXURE E**

**TERM-II PROJECT LABORATORY  
ASSIGNMENTS**

1. Review of design and necessary corrective actions taking into consideration the feedback report of Term I assessment, and other competitions/conferences participated like IIT, Central Universities, University Conferences or equivalent centers of excellence etc.
  - (i.) Registration module: The Admin will log-in to the system and enter the details of the patient. On clicking the submit button, the details get stored into the patient database. Staff (doctors) details are added by the Admin. Unique usernames and passwords can be set by the Staff.
  - (ii.) SDLC Model followed: The software development life cycle model followed is the Waterfall Model. The Waterfall model involves the following phases Gathering requirements, designing, implementation, verification, testing and maintenance.
  - (iii.) Deployment: Local machines belonging to the same network are connected to the server. Servers ip address is used to connect locally.
  - (iv.) Completeness: The project is NP Complete.
  - (v.) Testing: Use of Selenium testing tool to test the test cases of log-in, report generation and submitting the forms.
  - (vi.) Multiusers: Use of sessions to provide multi-user environment. Different views provided to different users based on their Status.
2. Project workstation selection, installations along with setup and installation report preparations.
  - (i.) Project workstation selection:  
System specifications of developmental machine:  
Processor Intel Core i3-5010U CPU @ 2.10GHz 2.10 GHz, Installed memory (RAM) 8.00 GB, System type 64-bit Operating System, x84-based processor, Operating System details Linux Fedora 20
  - (ii.) Software requirements: (client)  
Any machine with any modern operating system that supports modern web browsers. Latest operating system versions, web browsers versions are preferred for efficient working.
  - (iii.) Hardware requirements: (min)  
Processor - Intel Pentium 2009, Speed 2GHz, No. of cores 2
  - (iv.) Software requirements: (server)
    - i. MySQL database system should be installed.
    - ii. Apache Httpd server should be installed.
    - iii. Internet connectivity should be set up.
  - (v.) Setup:
    - i. Set up a wireless network connection.
    - ii. Connect the local clients to the same network.
    - iii. Type the IP address of the server and the login page at the web

browser of the client.

- iv. The connection to the server is established when the login page is viewed at the clients side.
- v. The system can now be used by logging in.

3. Programming of the project functions, interfaces and GUI (if any) as per 1 st Term term-work submission using corrective actions recommended in Term-I assessment of Term-work.

(a) The system is a web-application. Its first module is the access verification module. The details of the system users along with their login credentials are stored in a database at the server side. The users can access the system only after authentication. The system has two types of dedicated users: Admin Doctors

The system functions are user-specific.

(b) Functions for admin

- 1. Add patient When a new patient arrives, enter the basic details of the owner and patient. These details get stored in a database and can be accessed by the doctors on demand.
- 2. Add new doctor When there is a need of adding a new doctor or removing an old one, the admin can enter username and create a new entry or delete one from the database.
- 3. Update lab test rates and vaccination rates The polyclinic has different labs. The patients are charged as per the treatment in these labs. Also, the polyclinic maintains and makes provision for certain vaccinations. Using this functionality it can set/update rates for these vaccinations/labs.

### E.0.1 Code Snippet

```

//algorithm
inputs={x,y}
x=username
y=password
Check password against username in database
if match:
echo "Successful!"
else:
echo "Failed. Try again."

//code snippet
$sql="SELECT patient_no, owner_name, occupation, caste, category, address, phone_no,
day_in, day_out, animal_type from patient";
$result=$conn->query($sql);

//code snippet
$sql = "INSERT INTO admin (user_name, name, pwd, email, status, phone, address) VALUES
('$value', '$value2', '$value3', '$value4', '$value5', '$value6', '$value7')";
$sql = "delete from admin where user_name = '$value'";

//code snippet
$array = array('xray', 'sono', 'health', 'emr', 'post', 'vaccine');
$costs = array($value2, $value3, $value4, $value5, $value6);
$sql = "UPDATE rates SET cost='$costs[$i]' WHERE entity='$array[$i]'";

```

Figure E.1: Code Snippet

(c) Functions for doctor

1. Adding patient details When a patient arrives at the doctor, the doctor accesses his basic details from the database. He diagnoses the animal and treats him or provides treatment details. Doctors module is not a heavy one since he has to focus more on treatment of the patient. This includes mentioning of disease and suggesting the type of surgery if needed. This is implemented using checkboxes and textboxes.
2. Listing the patients If the doctors want to see the patients recorded for the day they can switch to this tab.
3. Listing animals for vaccination Entering the number of different types of animals that have arrived at the hospital for the purpose of vaccination. These numbers are stored against different animals in the database. These are then used to calculate the cost of vaccination for an owner.
4. Insemination Entering the cattle details for Artificial Insemination. Here, an automated messaging module is implemented which notifies

the owner of the pregnancy checkup visit date after the gestation period.

5. Pregnancy The doctor has to enter the details of pregnancy checkup of the cattle.
6. Infertility Details of infertility of the cattle are entered here.
7. Calf Birth Details of the calf being born are entered. This helps in estimating the cattle population.
8. Maintaining of nitrogen supply The details of the nitrogen dealer and the stock details are entered here.
9. Labs The labs which will be assigned for the treatment of any animal are specified here with the help of checkboxes. This is required to calculate the fees for treatment
10. Reporting Different types of reports are generated. This is time based reporting. The doctor enters the period for which he wants the reports and the result will be displayed.

### E.0.2 Code Snippet

```

//code snippet
$sql="SELECT patient_no,owner_name,occupation,caste,category,address,phone_no,day_in,animal_type from patient";

//code snippet
$sql = "INSERT INTO vaccine(v_no,date,cow_count,buffalo_count,goat_count,others_count,comment) VALUES('$count1', '$value1',
'$value2', '$value3', '$value4', '$value6', '$value9')";

//code snippet
$sql = "INSERT INTO ai(ai_no,patient_no,repr_organ,cond,comments) VALUES('$count1', '$value1', '$value2', '$value4')";

//code snippet
$sql = "INSERT INTO pregnancy(pd_no,patient_no,breed,ai_no,checkup_result,medication_details,proof_records_details)
VALUES('$count1', '$value1', '$value2', '$value3', '$value4')";

//code snippet
$sql = "INSERT INTO infertility(inf_no,patient_no,breed,ai_no,checkup_result,medication_details,proof_records_details)
VALUES('$count1', '$value1', '$value2', '$value3', '$value4')";

//code snippet
$sql = "INSERT INTO calf_birth_details(ch_no,patient_no,ai_no,birth_date,comment) VALUES('$count1', '$value1', '$value2', '$value3',
'$value4')";

//code snippet
$sql = "INSERT INTO nitrogen(ln_no,date,provider,cont_type_inch,cont_type_ltr,received_ltr,total_inch,total_ltr,comments)
VALUES('$count1', '$value2', '$value3', '$value4', '$value5', '$value6', '$value7', '$value8', '$value11')";

//code snippet
$checkbox1 = $_POST['check_list1'];
$query = "INSERT INTO labs(lt_no,patient_no,health,sonography,xray,emr,post_mortem,total) VALUES
('$count1','$value1','$checkbox1','$checkbox2','$checkbox3','$checkbox4','$checkbox5', '$c')";

```

Figure E.2: Code Snippet

(d) Automated messaging module

### E.0.3 Code Snippet

```
//code snippet
$username = "pavi_104@yahoo.co.in";
$hash = "7a0cf73491a38033eab253f70271da0dce226fdc9a6ac0d6f4af42b1580c9483";

// Config variables. Consult http://api.textlocal.in/docs for more info.
$test = "0";

// Data for text message. This is the text message data.
$sender = "TXTLCL"; // This is who the message appears to be from.
$numbers = $phone; // A single number or a comma-separated list of numbers
$message = "Next appointment date for patient number: $value1. is $eff. Please visit.";
// 612 chars or less
// A single number or a comma-separated list of numbers
$message = urlencode($message);
$data =
"username=$username.&hash=$hash.&message=$message.&sender=$sender.&numbers=$numbers.&test=$test;
$ch = curl_init('http://api.textlocal.in/send/?');
curl_setopt($ch, CURLOPT_POST, true);
curl_setopt($ch, CURLOPT_POSTFIELDS, $data);
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
$result = curl_exec($ch); // This is the result from the API
curl_close($ch);
```

Figure E.3: Code Snippet

**ANNEXURE F**

**INFORMATION OF PROJECT GROUP  
MEMBERS**



1. Name : SHALINI PRASANNA
2. Date of Birth : 03/08/1996
3. Gender : Female
4. Permanent Address : Kumar presidency phase 2,1101,Meera Nagar,koregaon park.
5. E-Mail : shalini.prasanna@cumminscollege.in
6. Mobile/Contact No. : 9923124646
7. Placement Details : Not Placed



1. Name : T. PAVITHRA
2. Date of Birth : 10/04/1996
3. Gender : Female
4. Permanent Address : Bloom 203, My World, Opp to Cummins Office, Balewadi High Street, Baner Road, Pune
5. E-Mail : t.pavithra@cumminscollege.in
6. Mobile/Contact No. : 9960822880
7. Placement Details : Cybage



1. Name : SUPRIYA WAGHMARE
2. Date of Birth : 26/01/1996
3. Gender : Female
4. Permanent Address : B-51,Ramnagar Colony,Ghugus,Chandrapur.
5. E-Mail : supriya.waghmare@cumminscollege.in
6. Mobile/Contact No. : 7028291278
7. Placement Details : Infosys



1. Name : ASHLESHA WAIKOS
2. Date of Birth : 30/04/1996
3. Gender : Female
4. Permanent Address : G801, Jacobina, Nyati Equatorial, Bavdhan, Pune.
5. E-Mail : ashlesha.waikos@cumminscollege.in
6. Mobile/Contact No. : 9764486484
7. Placement Details : Not Placed