



Kurdistan Region – Iraq
Ministry of Higher Education and Scientific Research
Lebanese French University
Department of Information Technology

Veterinary Clinic Management System

**A Graduation Project Submitted to the
Department of Information Technology (IT) /
Lebanese French University (LFU) as a Partial
Fulfilment of the Requirement of the BSc. Degree
in Information Technology**

By

Hardi Mustafa Jamil

Muhammed Salih Yassin

Mustafa Yassin Hamad Ali

Supervised By

Asst. Lect. Chiman Haydar Salih

April & 2023

Supervisor's Certification

I certify that the preparation of this graduation research project titled "Veterinary Management System " was made under my supervision at the department of Information Technology – Lebanese French University in partial fulfillment of the requirements for the degree of BSc. in Information Technology.

Signature:

Name: Asst. Lect. Chiman Haydar Salih

e-Mail: Chiman.salh@epu.edu.iq

Affiliation: College Of Science And Engineering – Department Of
Information Technology

Date: 13 / 04 / 2023

Examining Committee Report

We certify that we have read this graduation research project titled “Veterinary Management Web Application System “and as an examining committee, examined the students:

Hardi Mustafa Jamil, Muhammed Salih Yassin , Mustafa Yassin Hamad ALI

In it’s content and in what is related to it, and that in our opinion it meets the standard of a graduation research project for the degree of BSc in Information Technology.

Examining Committee

Signature:

Name: -Dr.S.T. Suganthi
Assistant Professor -

Date: / 04 / 2023

Signature:

Name: - Lect . Ahmed Najat Ahmed -

Date: / 04 / 2023

Head of Department

Signature:

Name: Lect. Ahmad Najat Ahmed
Date: / 04 / 2023

Dedication

This graduation research project is dedicated to:

This graduation research project is dedicated to all the individuals who have supported and encouraged me throughout my academic journey. To my family, friends, and professors who have believed in me and pushed me to be my best self, thank you for your unwavering support and guidance.

I also dedicate this project to the countless individuals who have contributed to the field of research that I have been exploring. Your hard work and dedication have paved the way for new discoveries and advancements, and I am honored to have had the opportunity to build upon your work.

Lastly, I dedicate this project to my future self, as a reminder of the hard work, dedication, and passion that went into completing this research. May it serve as a foundation for my future endeavors and a testament to the endless possibilities that come with pursuing knowledge and education.

1st Hardi Mustafa Jamil

2nd Muhammed Salih Yassin

3rd Mustafa Yassin Hamad ALI

Abstract

Veterinary Clinic Management System that proposed for this project. The proposed system is being developed to solve the problems faced by most of the veterinary clinics which are currently using the traditional method of operating their organizations in manual or physical filling that requires more time to manage the clinic. This is because the method has some drawbacks such as the data pet name, their owners, payments, etc.... written on the paper that can be easily misplaced, missing or the pet owner would forget to bring them before going to veterinary clinics in order to get a vaccination and deworming, etc...., services for their pets.

This veterinary management system allows practitioners to track patient information, store medical records securely, and schedule appointments easily. With this type of software in place, veterinarians are able to quickly access data on their patient's health history as well as financial records associated with each visit or procedure performed.

Table of Contents

Subject	Page No.
Supervisor's Certification	i
Examining Committee Report	ii
Dedication	iii
Abstract	iv
List of Figures	vi
List of Abbreviations	vii
Chapter One: An Overview	
1.1. Introduction	2
1.2. The Aim	2
1.3. Related Work	3
1.4. Problem Statement	4
1.5. Proposed Solution	4
Chapter Two: The System Design	
2.1. Introduction	6
2.2. The System Architecture	6
2.3. The Data Flow Diagram	7
Chapter Three: The Implemented System, Tests and Experimental Outcomes	
3.1. Introduction	9
3.2. Implementation Tools	9
3.3. Experimental Outcomes with Screen-shots	11
Chapter Four: Conclusions, Limitations and Future Works	
4.1. Conclusions	18
4.2. Limitations	18
4.3. Suggestions for Future Works	19
References	
References	20
Appendices	
Appendix A: Login Html codes	21
Appendix B : Login PHP codes	22
Appendix C : Add pets: Html , PHP Codes	24
Appendix D : Edit Pets Html , PHP Codes	26
Appendix E : Add Doctor Codes	30
Appendix F : All Doctor Codes	34

List of Figures

Figure No.	Figure's Title	Page No.
2-1	The Proposed System Architecture	6
2-2	Data Flow Diagram DFD	7
3-1	Log In Page	11
3-2	DashBoard Page	12
3-3	Add Pets Page	13
3-4	Add Doctor Page	13
3-5	Edit Pets Page	14
3-6	All Doctors Page	14
3-7	Error In Add Pets Page	15
3-8	Add Pets.php code	15
3-9	Correct Erros In Add Pets Page	16

List of Abbreviations

HTML Hyper Text Markup Language

CSS Cascading Style Sheets

PHP Hypertext Preprocessor

SQL Structured query language

JS Java Script

IDE Intrgrated development environment

SSMS SQL server management studio

Chapter One

An Overview

Chapter One

An Overview

1.1. Introduction

Veterinary management system is a software program designed to help veterinary clinics and hospitals to manage their daily operations efficiently. This system is designed to streamline the workflow of the clinic, including managing patient records, scheduling appointments, managing staff schedules, billing and invoicing, and other administrative tasks. The system also provides tools for managing medical records, including patient history, medication and treatment plans, lab results, and other clinical information. This software system provides a comprehensive solution for veterinary clinics to manage their business operations, improve their workflow, and deliver high-quality care to their patients. With the help of veterinary management system, veterinarians can focus more on providing medical care to animals rather than spending time on administrative tasks.

1.2. The Aim

The aim of Veterinary management system is to streamline and automate the operations of a veterinary clinic or hospital. It is designed to improve the efficiency and effectiveness of veterinary services by providing tools for managing patient records, appointments, inventory, billing and other important aspects of veterinary practice. With a veterinary management system, veterinarians can easily access patient information and medical history, schedule appointments, generate invoices and reports, manage inventory levels, and track financial transactions.

This system also helps to reduce paperwork, minimize errors, and improve communication between staff members. Ultimately, the aim of a veterinary management system is to ensure that veterinary clinics and hospitals can provide high-quality care to animals while also maximizing their profitability and success.

1.3. Related Work

The use of technology in veterinary management systems has gained significant attention in recent years. Several studies have explored different aspects of veterinary management systems, ranging from the design and development of software to the evaluation of their effectiveness in improving the delivery of veterinary services.

(Saleh Sulaiman S Aldhubayb and Mohammed Khaled Aldhubaib , 2020)

Examined the use of electronic medical records (EMRs) in veterinary practices. The study found that the use of EMRs improved the accuracy and completeness of medical records, reduced the time required to retrieve patient information, and improved communication among veterinary staff.

(Njeru Jerimiah Benard, 2018)

Rapid Application Development (RAD) is a form of agile software development methodology that prioritizes rapid prototype releases and iterations. Unlike the Waterfall method, RAD emphasizes the use of software and user feedback over strict planning and requirements recording .

(Anin Sofea Binti Azhari,2015)

Besides, for vaccination and deworming services, the pet owner will be given a "vaccination clinic card" which hold their pets' personal details and medical or treatment records. The appointment dates will also be recorded manually in the card(written down),as well as the record of prescriptions given [13]

1.4. Problem Statement

The lack of a proper system in a veterinary clinic leads to mismanagement of patient records, scheduling issues, and inefficient use of resources. This can result in delayed or missed appointments, inaccurate diagnosis or treatment, and poor customer satisfaction. Without a proper system, it becomes difficult for veterinary clinics to maintain accurate patient records and communicate effectively with pet owners.

This can lead to a loss of revenue as customers may seek services from other clinics that have a better system in place. Therefore, it is essential for veterinary clinics to adopt a suitable system that streamlines their operations and ensures efficient management of resources.

There have been some flaws in the current system:

1. It takes a lot of time to record and find information about animals.
2. Writing data on paper carries the risk of loss.
3. Lack of an obvious source of income.

1.5. Proposed Solution

After creating a veterinary management system, the proposed solution would be to implement it effectively in veterinary clinics and hospitals. This can be achieved by providing proper training to the staff members and veterinarians so that they can efficiently use the system. The next step would be to ensure that all patient data, medical histories, and treatment plans are entered into the system accurately and comprehensively. This will help in providing better treatment to the animals and will also enable the veterinarians to track the progress of the treatment.

1. To develop a system that allows us to easily record information. At the same time, you only need the (id) of the patient or doctor to find the information.
2. When data is entered into the system, it is stored in a safe place so that it can be reused.
3. There is a section of the system called payments where the source of income is identified.

Chapter Two

The System Design

2.3. The Data Flow Diagram

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled.

They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO. That’s why DFDs remain so popular after all these years. While they work well for data flow software and systems, they are less applicable nowadays to visualizing interactive, real-time or database-oriented software or systems. In this data flow diagram the admin or the person who manage the clinic need to create an account in sign up page , after that they can login and can see the home screen page , they can add data in specific part (add pets, add doctor, add appointments and add payment), after the data recorded they will be stored in the database , the data will retrieve to the system that can be see in view data pages(all doctor ,all pets, all appointments ,all payments),then if there have any mistakes of storing the data he can change or update the mistakes in edit pages.

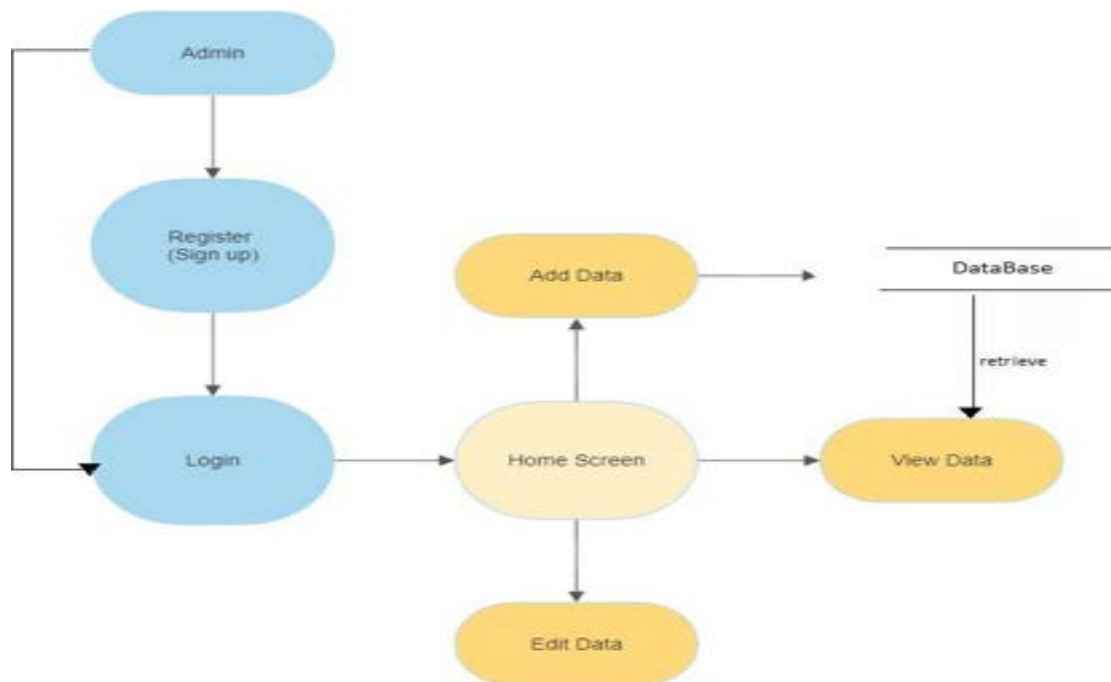


Figure (2.2) Data Flow Diagram DFD

Chapter Three

**The Implemented System,
Tests and Experimental
Outcomes**

Chapter Three

The Implemented System, Tests and Experimental Outcomes

3.1. Introduction

The aim of this chapter is to describe the implementation of the design system, which was given in details in chapter three. The system requirements and the implementation toolset are discussed also. The Veterinary Management System is presented and the interaction of user with the implemented system through the demonstration as test are discussed. Finally the empirical outcomes are discussed.

3.2. Implementation Tools

Implementation tools are software or hardware solutions that are used to execute a project plan or strategy. These tools help project managers and team members in effectively managing the project, tracking progress, and achieving the project goals, The tools automate many tasks and processes, making it easier for project managers to manage tasks, deadlines, and resources. Additionally, they provide visibility into project progress, allowing stakeholders to stay informed and take corrective actions if needed. Overall, implementation tools play a critical role in ensuring the success of a project.

3.2.1. Hardware Tools

Computer :

Device name DESKTOP-UCOPFAL

Processor Intel(R) Core(TM) i7-3632QM CPU @ 2.20GHz 2.20 GHz

System type 64-bit operating system, x64-based processor

Edition Windows 10 Pro

Version 21H2

OS build 19044.2130

Experience Windows Feature Experience Pack 120.2212.4180.0

3.2.2. Software Tools

Software tools are essential in programming languages because they help developers to automate and simplify the process of writing, testing, and debugging code. These tools range from text editors to integrated development environments (IDEs), compilers, debuggers, and version control systems.

IDEs (Integrated development environments) provide a comprehensive development environment that includes a code editor, compiler, debugger, and other tools that help developers to write, test, and deploy code efficiently. They also provide features such as syntax highlighting, code completion, and debugging tools that make it easier for developers to write error-free code.


Microsoft SQL Server Management Studio (SSMS) has an integrated development environment (IDE) that allows database administrators and developers to manage and manipulate SQL Server databases. The IDE provides a user-friendly interface for creating and executing queries, designing and modifying database schemas, and managing security and permissions. The SSMS IDE also includes tools for monitoring database performance and troubleshooting issues, making it a powerful tool for managing SQL Server databases.

Notepad++ is not an IDE. It is just a text editor, which can be used to edit source code. Notepad++ is a Windows-based and open-source software that helps businesses create and edit source codes using multiple programming languages, such as C++, Java, R, SQL, XML, Python, Pascal, JSON, and more. It enables staff members to find and replace text in files using dialog-based, incremental, and dialog-free search methodologies. Notepad++ natively supports languages such as XML, Python, CSS, HTML, Ruby, Pascal, Visual Basic, and more. We Used HTML ,CSS ,Java Script and PHP for creating the template for our system.

Windows is a graphical operating system developed by Microsoft. It allows users to view and store files, run the software, play games, watch videos, and provides a way to connect to the internet. It was released for both home computing and professional works.

3.3. Experimental Outcomes with Screen-shots

Figure (3.1) Login form: after the user created an account in sign up page , the user must enter their username and password to gain access to the system.[Appendix A],[Appendix B].



The screenshot displays a login form titled "Login" in a pink font. Below the title, there are two input fields: "Username" and "Password", both with light gray backgrounds. A prominent pink button labeled "Login" is positioned below the password field. At the bottom of the form, there is a link that reads "New User, Sign Up Here" in a blue font.

Figure (3.1) Log In Page

Figure (3.2) After the user(admin)has entered into the system , he can see the sidebar. As well as the data , such as :number of pets ,appointments ,total revenue and he can select the list of pets , doctors , appointments ,and payments he can add the data and view and edit in each of them.



Figure (3.2) DashBoard Page

Figure (3.3) The user(admin) can add the detail about the pets and pet owners such as (pet name, age , phone number ,address, choose the gender(sex) then add pet ,after that the data send to the database, and retrieve to the (all pets)page you can see there, [Appendix C].

The screenshot shows the 'Add Pets' form in the VET ERBIL application. The form is located in the main content area and contains the following fields:

- Pet Name:** A text input field.
- Age:** A text input field.
- Phone:** A text input field.
- Address:** A text input field.
- Gender:** Radio buttons for 'Male' and 'Female'.

A red 'Add Pet' button is positioned at the bottom right of the form. The left sidebar contains navigation options: Dashboard, Pets, Doctors, Appointments, Payments, and Login. Below the navigation is a 'Need Help' section with contact information: +9647028704248 and proclinic2023@site.com.

Figure (3.3) Add Pets

PageFigure (3.4) Admin can add doctor in add doctor page .[Appendix E]

The screenshot shows the 'Add Doctor' form in the VET ERBIL application. The form is located in the main content area and contains the following fields:

- Doctor Name:** A text input field.
- Specialization:** A text input field.
- Experience:** A text input field.
- Age:** A text input field.
- Gender:** Radio buttons for 'Male' and 'Female'.
- Phone:** A text input field.
- Address:** A text input field.

A red 'Add Doctor' button is positioned at the bottom left of the form. The left sidebar is identical to the 'Add Pets' form, showing navigation options and contact information.

Figure (3.4) Add Doctor Page

Figure (3.5) After we added the details about the pets or owner pets , if we want to changewe can do it in edit pets page .[Appendix D]

Figure (3.5) Edit Pets Page

Figure (3.6) The doctor details will retrieve by the database to the (all doctor)page ,then hecan see the data. .[Appendix F]

Doctor ID	Doctor Name	Experience (in years)	Phone	Specialization
1	Muhammed	2	22222	DR.
2	Acad	3	075870420	vet
3	same	3	077028321	urgery

Figure (3.6) All Doctors Page

Figure (3.7) We had an error in PHP codes on line 329 , 330

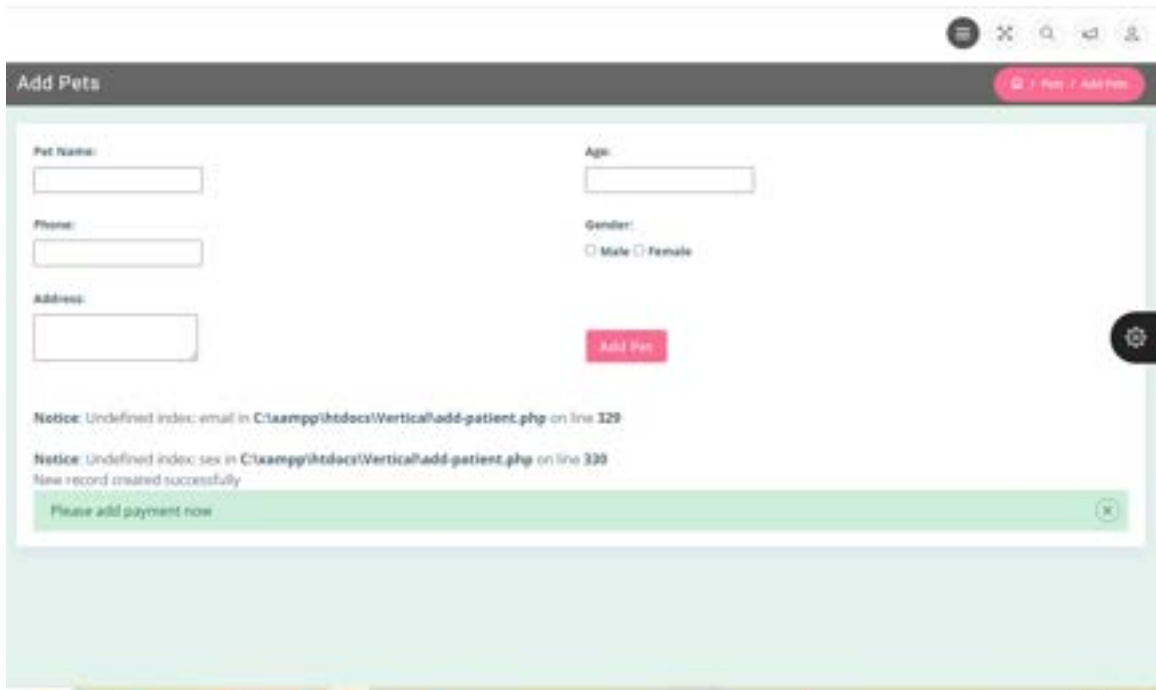


Figure (3.7) Error In Add Pets Page

Figure (3.8) The gender should be the same with the (sex):

```

<input type="radio" id="male" name="gender" value="male" >
<label for="male" >Male</label>
<input type="radio" id="female" name="gender" value="female" >
<label for="female">Female</label>
</div>
<div class="form-group col-md-6">
<label for="address">Address:</label><br>
<textarea id="address" name="address"></textarea><br>
</div>
<div class="form-group col-md-6 mb-3">
<br>
<button type="submit" class="btn btn-primary btn-lg" type="submit" name="
</div>
</div>
<?php
$host = 'localhost';
$user = 'amer';
$password = '12345';
$dbname = 'vetdb';

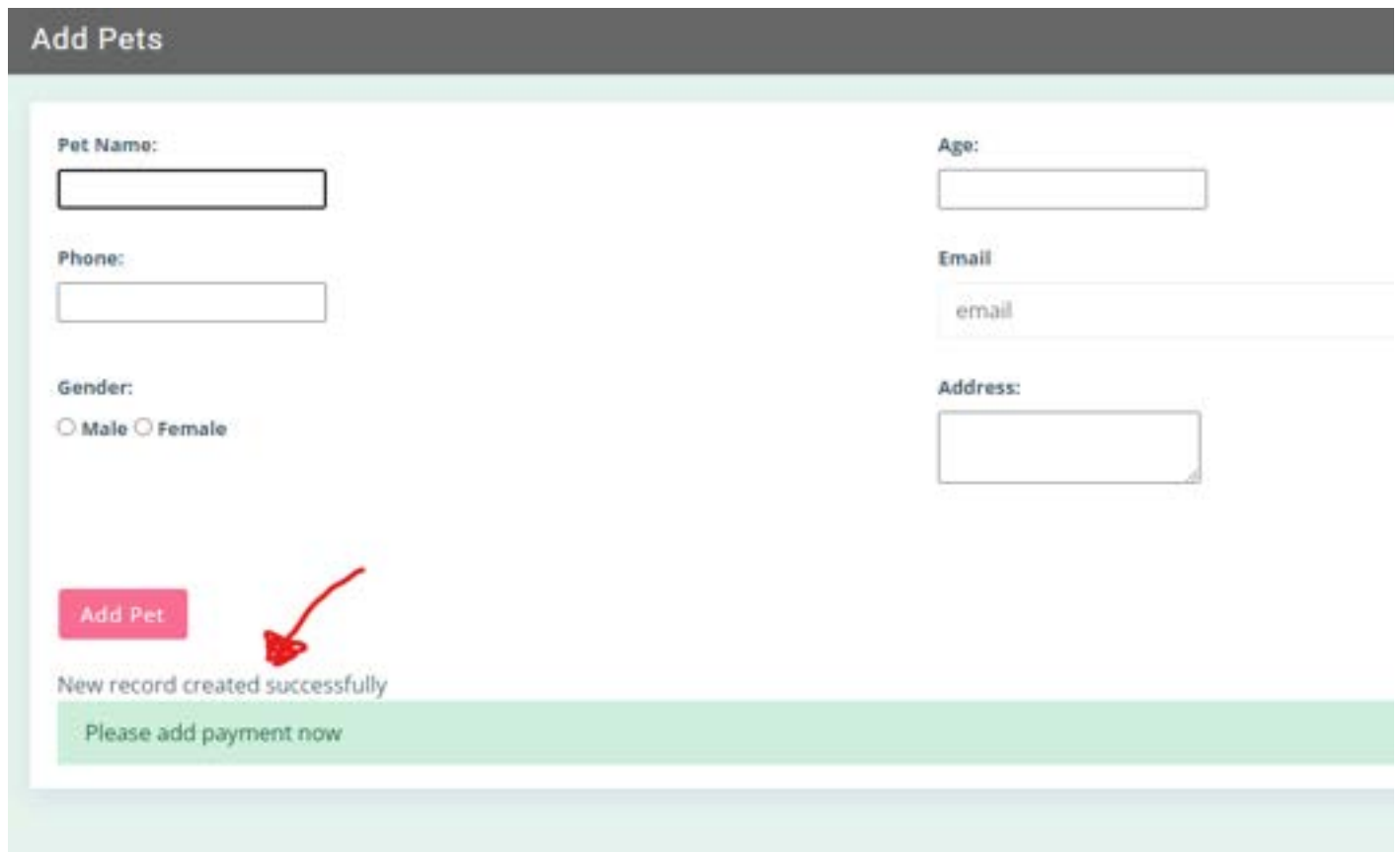
// Create a database connection
$conn = mysqli_connect($host, $user, $password, $dbname);
if (isset($_POST['submit'])) {

    $patientname = $_POST['patientname'];
    $age = $_POST['age'];
    $phone = $_POST['phone'];
    $email = $_POST['email'];
    $sex = $_POST['sex'];
    $address = $_POST['address'];

```

Figure (3.8) Add Pets.php code

Figure (3.9) Finally we solve the problem.



The image shows a web form titled "Add Pets" with a dark grey header. The form contains several input fields: "Pet Name:" (text box), "Age:" (text box), "Phone:" (text box), "Email" (text box with "email" entered), and "Address:" (text box). Below these is a "Gender:" section with radio buttons for "Male" and "Female". A pink "Add Pet" button is located at the bottom left. A red arrow points to the "Add Pet" button. Below the button, a message states "New record created successfully". At the bottom, a green banner contains the text "Please add payment now".

Figure (3.9) Correct Erros In Add Pets Page

Chapter Four

Conclusions, Limitations and Future Works

Chapter Four

Conclusions, Limitations, and Future Works

4.1. Conclusions

For this graduation research project (GRP), many points which are certain significance are drawn and concluded from this work, they are:

The Veterinary Management System is a powerful tool for veterinarians and their staff to better manage patient records, appointments, and inventory. It provides an efficient way of organizing data in one place. This system also helps keep track of medical history for each pet as well as providing the ability to create customized reports. By utilizing this system, veterinary practices can improve overall efficiency while reducing costs associated with manual record keeping processes.

In conclusion, the Veterinary Management System offers a comprehensive solution for managing all aspects of running a successful veterinary practice including patient records management, appointment scheduling and inventory tracking capabilities. Not only does it provide an efficient method of storing information but it also allows quick access when needed which saves time on administrative tasks such as compiling reports or searching through files manually. Overall this system makes life easier both in terms of organization and cost savings making it essential software for any veterinarian looking to run their business more efficiently while providing quality care to patients at lower costs than traditional methods would allow them to achieve.

4.2. Limitations

The researchers suffers from many limitations during the design and implement system, as listed in the following:

Projects involving the development of veterinary management systems are essential for managing veterinary practices and providing optimal care to animals. However, there are a number of limitations associated with these types of projects that must be taken into consideration when planning and executing them.

1- The first limitation is related to cost. Developing a comprehensive system can be expensive due to the need for specialized hardware, software, personnel resources and training costs associated with implementation. Additionally, ongoing maintenance costs may also arise if upgrades or modifications become necessary over time in order to keep up with changing technology or regulations within the industry.

2- Another limitation relates to data security concerns as sensitive information about both patients and clients must remain secure at all times in order for privacy laws such as HIPAA (Health Insurance Portability & Accountability Act) compliance regulations can be met properly . This requires additional measures such as encryption protocols being put into place which adds an extra layer complexity during development stages but is important nonetheless from a safety standpoint .

3- Finally , user adoption remains another key challenge since any new system will require staff members who use it on daily basis have proper understanding how it works so they able efficiently utilize its features maximum potential . If users not comfortable using particular feature set then performance impact could occur which leads unnecessary delays other issues impacting overall success project .

In conclusion , while developing veterinary management systems offers numerous benefits helping streamline operations improve patient care standards there still several limitations should kept mind throughout process ensure successful completion project goals objectives met desired results achieved end day.

4.3. Suggestions for Future Works

The designed and implemented system presents many fruitful lines of continued graduation research, and opens the door to a range if future work, as listed in the following:

1. Add credit cards to the system
2. People can register patients online for appointments in the system.
3. Also add other important or necessary features to the system such
as pharmaceuticals and online sales.
4. Enhanced security: The veterinary management web application system should continue to improve its security protocols to ensure that sensitive data is protected from unauthorized access and cyber-attacks.
5. Expansion to other cities: The application should be expanded to other cities and countries, catering to the needs of a wider range of clients and veterinary clinics.

References

- [1] Plav{i} B, 2005, Informational system in veterinary medicine, support in risk-analysis and management of the animal diseases prevention and control programmes, Master's thesis, Faculty of Veterinary Medicine, University of Belgrade, Belgrade.
- [2] AVImark Veterinary Practice Management Software" by Henry Schein Animal Health
- [3] Veterinary Emergency and Critical Care Society (VECCS)
- [4] Hospital Management System authentication by NJERU JERIMIAH BENARD ID:M21532121028 Ultra University Department Of Computer Science & Engineering (DFD)
- [5] Veterinary hospital management software by Santosh p.Sarangamath and vivek m. Patil - 'Information Technology in Veterinary Science' (2009), New India Publishing Agency, New Delhi. ISBN 978-81-908512-4-4
- [6] Mr. Doc: A Doctor Appointment Application System Article in International Journal of Computer Science and Information Security, • December 2016 (time slot)
- [7] June 2016Journal of Contemporary Medical Education 2

APPENDIX

Appendix A: Login Html code

```

<div class="wrapper">
<!-- Page Content -->
<div id="content">
<div class="container-fluid">
<div class="row">
<div class="col-sm-6 auth-box">
<div class="proclinic-box-shadow">
<h3 class="widget-title">Login</h3>
<form method="POST" class="widget-form" action="<?php echo
htmlspecialchars($_SERVER['PHP_SELF']); ?>">
<!-- form-group -->
<div class="form-group row">
<div class="col-sm-12">
<input name="username" placeholder="Username" class="form-control" required=""
data-validation="length alphanumeric" data-validation-length="3-12"
data-validation-error-msg="User name has to be an alphanumeric value (3-12 chars)"
data-validation-has-keyup-event="true">
</div>
</div><div class="form-group row">
<div class="col-sm-12">
<input type="password" placeholder="Password" name="password" class="form-control"
data-validation="strength" data-validation-strength="2"
data-validation-has-keyup-event="true">
</div></div>
<!-- /.form-group -->
<!-- form-group -->

```

```

<!-- /.form-group -->
<!-- Check Box -->
<!-- /Check Box -->
<!-- Login Button -->
<div class="button-btn-block">
<button type="submit" value="Login" class="btn btn-primary btn-lg btn-
block">Login</button>
</div>
<!-- /Login Button -->
<!-- Links -->
<div class="auth-footer-text">
<small>New User,
<a href="sign-up.php">Sign Up</a> Here</small>
</div>
<!-- /Links -->
</form>
</div>
</div>
</div>
</div>

```

Appendix B:Login PHP code

```

<?php
    // Database connection
    $host = "localhost"; // Database server host name
    $user = "amer"; // Database user name
    $password = "12345"; // Database password
    $database = "vetdb"; // Database name

```

```

$conn = mysqli_connect($host, $user, $password, $database);
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
// Form validation and authentication
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $username = $_POST["username"];
    $passwordd = $_POST["password"];
    $sql = "SELECT * FROM users WHERE username='$username' AND
password='$passwordd'";

    $result = mysqli_query($conn, $sql);
    if (mysqli_num_rows($result) == 1) {
        session_start();
        $_SESSION["username"] = $username;
        header("location: index.html");
        exit;
    } else {
        echo "<p style='color:red'>Invalid username or
password.</p>"; }
    }
    mysqli_close($conn);
?>

```

Appendix C:Add Pets

Html code , PHP code

```

<form method="post" enctype="multipart/form-data">

<div class="form-row">
<div class="form-group col-md-6">

    <label for="patientname">Patient Name:</label><br>

<input type="text" id="patientname" name="patientname" required><br>

</div>

    <div class="form-group col-md-6">

<label for="age">Age:</label><br>

        <input type="number" id="age" name="age" required><br>

</div>

    <div class="form-group col-md-6">

<label for="phone">Phone:</label><br>

<input type="number" id="phone" name="phone" required><br>

</div>

    <div class="form-group col-md-6">

<label for="sex">Gender:</label><br>

        <input type="radio" id="male" name="sex" value="male" >
<label for="male" >Male</label>

<input type="radio" id="female" name="sex" value="female" >
<label for="female">Female</label>

    </div>

<div class="form-group col-md-6">

    <label for="address">Address:</label><br><textarea id="address"
name="address"></textarea><br>

</div>

```



```

<div class="form-group col-md-6 mb-3">
<br>
<br> <button type="submit" class="btn btn-primary btn-lg" type="submit" name="submit"
>Add Patient</button>
</div> </div>
<?php
$host = 'localhost';
$user = 'amer';
$password = '12345';
$dbname = 'vetdb';
// Create a database connection
$conn = mysqli_connect($host, $user, $password, $dbname);
if (isset($_POST['submit'])) {
$patientname = $_POST['patientname'];
$age = $_POST['age'];
$phone = $_POST['phone'];
$sex = $_POST['sex'];
$address = $_POST['address'];
$sql = "INSERT INTO vet_table (patientname, age, phone, sex, address)
VALUES ('$patientname', '$age', '$phone', '$sex', '$address')";
if (mysqli_query($conn, $sql)) {
echo "New record created successfully";} else {
echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}
}
?>
</form>

```

Appendix D:Edit Pets HTML Code, PHP Code

```

<form><div class="form-row">
<div class="form-group col-md-6">
<label for="patient-name" >ID *&nbsp; &nbsp;</label>
<input type="number" name="id" required>
</div>
<div class="form-group col-md-6">
<label for="patient-name">Pet Name</label>
<input type="text" name="patientname">
</div>
<div class="form-group col-md-6">
<label for="age">Age &nbsp; &nbsp;</label>
<input type="text" name="age">
</div>
<div class="form-group col-md-6">
<label for="email">Email</label>
<input type="email" placeholder="email" class="form-control" id="Email" name="email">
</div>
<div class="form-group col-md-6">
<label for="phone">Phone No</label>
<input type="text" name="phone" >
</div>
<div class="form-group col-md-6">
<label for="gender">Sex</label>
<select class="form-control" name="sex">
<option selected>Male</option>

```

```

<option>Female</option>
    </select>
</div>
<div class="form-group col-md-12">
<label for="address">Address:</label><br>
    <textarea name="address"></textarea>
</div>
<div class="form-group col-md-6 mb-3">
<button type="submit" class="btn btn-primary btn-lg" type="submit" name="submit"
value="Submit">Update</button>
<button type="submit" class="btn btn-primary btn-lg" type="submit" name="delete"
value="delete">Delete</button>
</div>
<div class="form-group col-md-6 mb-3">
</div>
</div>
</form>
<?php
    // Check if the form is submitted
    if (isset($_POST['submit'])) {
        // Get the form data
        $id = $_POST['id'];
        $phone = !empty($_POST['phone']) ? $_POST['phone'] : '';
        $patientname = !empty($_POST['patientname']) ? $_POST['patientname'] : '';
        $age = !empty($_POST['age']) ? $_POST['age'] : '';
    }

```

```

$email = !empty($_POST['email']) ? $_POST['email'] : "";
$sex = !empty($_POST['sex']) ? $_POST['sex'] : "";
$address = !empty($_POST['address']) ? $_POST['address'] : "";

// Connect to the database
$host = 'localhost';
$user = 'amer';
$password = '12345';
$dbname = 'vetdb';
$conn = new mysqli($host, $user, $password, $dbname);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
// Prepare the SQL statement
$sql = "SELECT * FROM vet_table WHERE id='$id'";
$result = $conn->query($sql);

if ($result->num_rows == 1) {
    // Record exists, update it
    $row = $result->fetch_assoc();
    $patientname = !empty($patientname) ? $patientname :
$row['patientname'];
    $age = !empty($age) ? $age : $row['age'];
    $email = !empty($email) ? $email : $row['email'];
    $sex = !empty($sex) ? $sex : $row['sex'];
    $address = !empty($address) ? $address : $row['address'];
    $phone = !empty($phone) ? $phone : $row['phone'];
}

```

```
        $sql = "UPDATE vet_table SET patientname='$patientname',
age='$age', email='$email', sex='$sex', address='$address', phone='$phone' WHERE id='$id'";
        if ($conn->query($sql) === TRUE) {
            echo "Record updated successfully";
        } else {
            echo "Error updating record: " . $conn->error;
        }
    } else {
        // Record doesn't exist
        echo "Record with ID $id not found";
    }
    // Close the database connection
    $conn->close();
}
?>
```

Appendix E:Add Doctor

```

<form method="POST" >
<div class="form-row">
<div class="form-group col-md-6">
<label for="Doctor-name">Doctor Name</label>
<input type="text" required class="form-control" placeholder="Doctor name" id="name"
name="name" value="<?php echo isset($_POST['name']) ? $_POST['name'] : " ?>" required>
</div>

<div class="form-group col-md-6">
<label for="specialization">Specialization</label>

<input type="text" required placeholder="Specialization" class="form-control"
id="specialization" name="specialization" value="<?php echo isset($_POST['specialization']) ?
$_POST['specialization'] : " ?>" required>

</div>

<div class="form-group col-md-6">
<label for="experience">Experience</label>

<input type="number" placeholder="Experience" class="form-control"
id="experience" name="experience" value="<?php echo isset($_POST['experience']) ?
$_POST['experience'] : " ?>" required>

</div>

<div class="form-group col-md-6">
<label for="age">Age</label>

<input type="number" placeholder="Age" class="form-control" id="age"
name="age" value="<?php echo isset($_POST['age']) ? $_POST['age'] : " ?>" required>

```

```

    </div>

<div class="form-group col-md-6">
<label for="phone">Phone</label>

<input type="number" placeholder="Phone" class="form-control" id="phone"
name="phone" value="<?php echo isset($_POST['phone']) ? $_POST['phone'] : '' ?>" required>

</div>

<div class="form-group col-md-6">
<label for="email">Email</label>

    <input type="email" placeholder="email" class="form-control" id="email"
name="email" value="<?php echo isset($_POST['email']) ? $_POST['email'] : '' ?>" >

    </div>

    <div class="form-group col-md-6">
        <label for="gender">Gender</label>

        <select class="form-control" name="gender" id="gender">

            <option value="male" <?php if(isset($_POST['gender']) && $_POST['gender'] == 'male')
echo 'selected' ?>>Male</option>

            <option value="female" <?php if(isset($_POST['gender']) && $_POST['gender'] == 'female')
echo 'selected' ?>>Female</option>

        </select>
    </div>

<div class="form-group col-md-6">
<label for="about-doctor">Doctor Details</label>

    <textarea id="doctor-details" placeholder="Doctor Details" class="form-control"
name="details" rows="5"><?php echo isset($_POST['details']) ? $_POST['details'] : ''
?></textarea>

    </div>

    <div class="form-group col-md-6">
<label for="address">Address</label>

```

```
<textarea id="address" placeholder="Address" class="form-control" name="address"
rows="2"><?php echo isset($_POST['address']) ? $_POST['address'] : '' ?></textarea>
```

```
</div>
```

```
<div class="form-group col-md-6 mb-3"><br><br><br><br>
```

```
<button type="submit" class="btn btn-primary btn-lg" name="submit">Add
Doctor</button>
```

```
</div>
```

```
</div>
```

```
</form>
```

```
<?php
```

```
// Establish database connection
```

```
$servername = "localhost";
```

```
$username = "amer";
```

```
$password = "12345";
```

```
$dbname = "vetdb";
```

```
$conn = mysqli_connect($servername, $username, $password, $dbname);
```

```
// Check connection
```

```
if (!$conn) {
```

```
    die("Connection failed: " . mysqli_connect_error());
```

```
}
```

```
if (isset($_POST['submit'])) {
```

```
    $doctor_name = $_POST['name'];
```

```
    $specialization = $_POST['specialization'];
```

```
    $experience = $_POST['experience'];
```

```
    $age = $_POST['age'];
```

```
    $phone = $_POST['phone'];
```

```
    $email = $_POST['email'];
```

```
    $gender = $_POST['gender'];
```



```
$doctor_details = $_POST['details'];  
$address = $_POST['address'];  
  
// Check if phone already exists in database  
$sql = "SELECT * FROM doctors WHERE phone='$phone'";  
$result = mysqli_query($conn, $sql);  
if (mysqli_num_rows($result) > 0) {  
    echo "Phone number already exists in database."  
} else {  
    // Insert new doctor data into database  
    $sql = "INSERT INTO doctors (name, specialization, experience, age, phone, email, gender,  
details, address) VALUES ('$doctor_name', '$specialization', '$experience', '$age', '$phone',  
'$email', '$gender', '$doctor_details', '$address')";  
  
    if (mysqli_query($conn, $sql)) {  
        echo "New doctor added successfully."  
    } else {  
        echo "Error: " . $sql . "<br>" . mysqli_error($conn);  
    }  
}  
  
// Close database connection  
mysqli_close($conn);  
}  
?>
```

Appendix F:All Doctors

```

<li class="nav-item">
  <a class="dropdown-toggle" data-toggle="dropdown" href="#" role="button" aria-
  haspopup="true" aria-expanded="false">
    <span class="ti-announcement"></span>      </a>
  <div class="dropdown-menu proclinic-box-shadow2 notifications animated flipInY">
    <h5>Notifications</h5>
    <a class="dropdown-item" href="#">
      <span class="ti-wheelchair"></span> New Patient Added</a>
    <a class="dropdown-item" href="#">
      <span class="ti-money"></span> Patient payment done</a>
    <a class="dropdown-item" href="#">
      <span class="ti-time"></span>Patient Appointment booked</a>
    <a class="dropdown-item" href="#">
      <span class="ti-wheelchair"></span> New Patient Added</a>
  </div>
</li>
<li class="nav-item">
  <a class="dropdown-toggle" data-toggle="dropdown" href="#" role="button" aria-
  haspopup="true" aria-expanded="false">
    <span class="ti-user"></span>
  </a>
  <div class="dropdown-menu proclinic-box-shadow2 profile animated flipInY">
    <h5>John Willing</h5>
    <a class="dropdown-item" href="#">
      <span class="ti-settings"></span> Settings</a>
    <a class="dropdown-item" href="#">
      <span class="ti-help-alt"></span> Help</a>
    <a class="dropdown-item" href="#">
      <span class="ti-power-off"></span> Logout</a>

```

```
</div>
</li>
</ul>

</div>
</nav>
<!-- /Top Navigation -->
<!-- Breadcrumb -->
<!-- Page Title -->
<div class="row no-margin-padding">
<div class="col-md-6">
<h3 class="block-title">Patients</h3>
</div>
<div class="col-md-6">
<ol class="breadcrumb">
<li class="breadcrumb-item">
<a href="index.html">
<span class="ti-home"></span>
</a> </li>
<li class="breadcrumb-item">Patients</li>
<li class="breadcrumb-item active">All Patients</li>
</ol>
</div>
</div>
<!-- /Page Title -->

<!-- /Breadcrumb -->
<!-- Main Content -->
<div class="container-fluid">
```

```

<div class="row">
  <!-- Widget Item -->
  <div class="col-md-12">
    <div class="widget-area-2 proclinic-box-shadow">
      <h3 class="widget-title">Patients List</h3>
      <div class="table-responsive mb-3">
        <table id="tableId" class="table table-bordered table-striped">
          <thead>
            <tr>
              <th>Doctor ID</th>
              <th>Doctor Name</th>
              <th>Experience <small>(in Years)</small></th>
              <th>Phone</th>
              <th>Specialization</th>
            </tr>
          </thead>
          <tbody>

<?php
    $host = 'localhost';
    $user = 'amer';
    $password = '12345';
    $dbname = 'vetdb';

```

```
// Create a database connection
```

```
$conn = mysqli_connect($host, $user, $password, $dbname);
```

```
// Check connection
```

```
if (!$conn) {
```

```
    die("Connection failed: " . mysqli_connect_error());
```

```
}
```

```
// Select all records from the vet_table
```

```
$sql = "SELECT * FROM doctors";
```

```
$result = mysqli_query($conn, $sql);
```

```
echo "<tr>";
```

```
//echo "<td>".1$row["id"].echo"</td>";
```

```
if (mysqli_num_rows($result) > 0) {
```

```
    // Output data of each row
```

```
    while($row = mysqli_fetch_assoc($result)) {echo
```

```
"<tr><td>". $row["id"]. "</td><td>". $row["name"]. "</td><td>". $row["experience"]. "</td><td>". $row["phone"]. "</td><td>". $row["specialization"]. "</td></tr>";
```

```
    }
```

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

```
    } else {
```

```
        echo "0 results";
```

```
    }
```