Project Name: VET US

Batch PG-DAC Sep-2021

Documentation On

"VET US"

Submitted By

Group No: 15

- Nitin Dang.   210943120061
- Shreyas Limaye.   210943120096
- Omkar Kshirsagar.   210943120062
- Chaitanya Waghmode.   210943120023

UNDER THE GUIDENCE OF:

Mrs. Harshita Maheshwari
Faculty Member

Infoway Technologies, Pune.

<u>ACKNOWLEDGEMENT</u>

The successful completion of a project is not an individual's effort. It is an outcome of cumulative efforts of all group members, each having its own importance to the objective.

This section is a value of thanks and gratitude towards all members for contributed in their own special way towards the completion of the project. For their invaluable comments and suggestions, we wish to thank all group members. We owe our gratitude and appreciation to our guide Mrs. Harshita Maheshwari whose suggestions and encouragement helped us to co-ordinate our project. We would also like to express our gratitude towards Infoway faculty and management for their Extended Support. Our thanks and appreciations also go to all project members for developing the project.

Abstract

Online applications are playing an important role in our day to day life from online shopping to doctor booking which is saving time. In this project features for booking an appointment by checking the availability of a doctor and then select specific doctor specialization and a form is shown to the user who will fill the form based on animal condition and symptoms.

Table of Contents

1. <u>INTRODUCTION</u>

The goal of the system is to create a tool for managing the workflow of a veterinarian. There was a need for a light weight application that would keep track of patients, appointments, procedures and medication.

All these vets need a web application to manage their work, clients, patients, medication, timetables, receipts. In the second chapter we go through the methods, tools and techniques that were used developing the system.

The third chapter lists all the main features of the application and explains their functionality in more detail.

Fourth chapter tells about the planning of the project and includes diagrams, use case scenarios and lists that help the reader to get a better understanding of how the application works.

1.1 Problem Statement

VET US is a web application that helps to store and manage information and procedures that vets can book appointments. The pages are made by React JS, the database is handled by MySQL, the application logic is controlled by Spring boot.

1.2 Aims & Objectives –

- Provide convenient way of appointment reservation.
- Computerized patient's information and treatment review.
- Give clinic staff easy way in doing information maintenance and updates.
- Upgrade performance.
- Easy access to system options.

3.1. Existing System

There is already existing online website for the Clinic Management but not foe the Veterinary.

3.2. Proposed System

➢ **CUSTOMER –**

- Register/Login –

- View Doctors –

    - Based on Nearest Distance

    - Based on Feedback

    - Based on Availability

- Book Appointment –

    - Book Appointment

    - Schedule Appointment

➢ **DOCTOR –**

- Register/Login –

- Appointments –

    -Schedule Appointments

    -Confirm Appointments

    - Cancel Appointments

3.3. Software Requirement Specification

Hardware:

1. Intel i3 Processor 3rd Generation
2. 4 GB ddr3 ram.
3. Windows 10 Home edition or later.
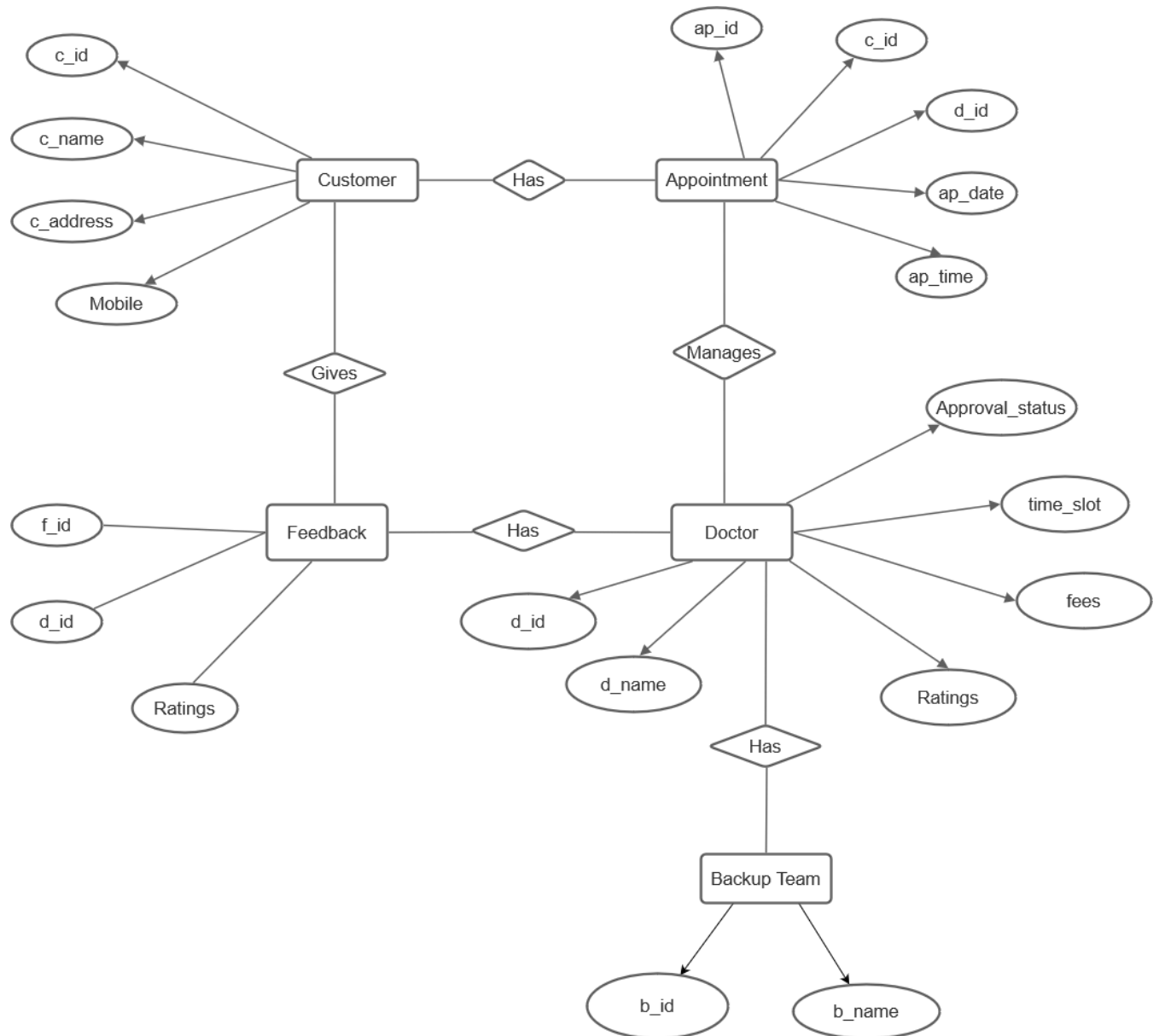4. 200 GB SATA HDD Space
5. Data Connection 200 Kbps

Software:

1. **Frontend-** REACTJS
2. **Backend-** Spring boot
3. **Database-** MySQL 5.7 with Workbench 8.0
4. **.**Apache Tomcat Server 9.0
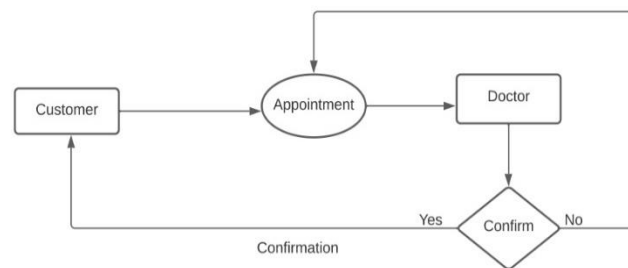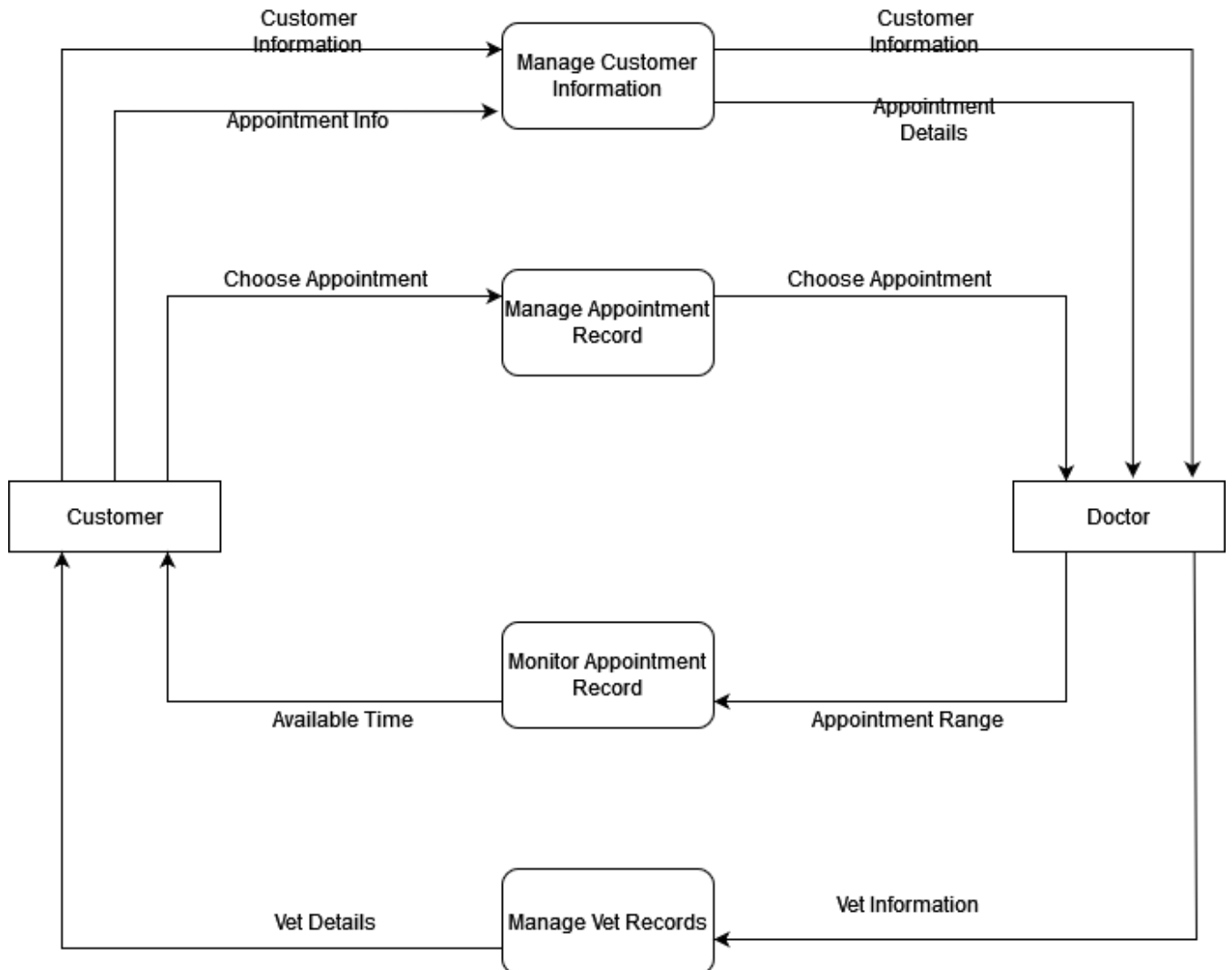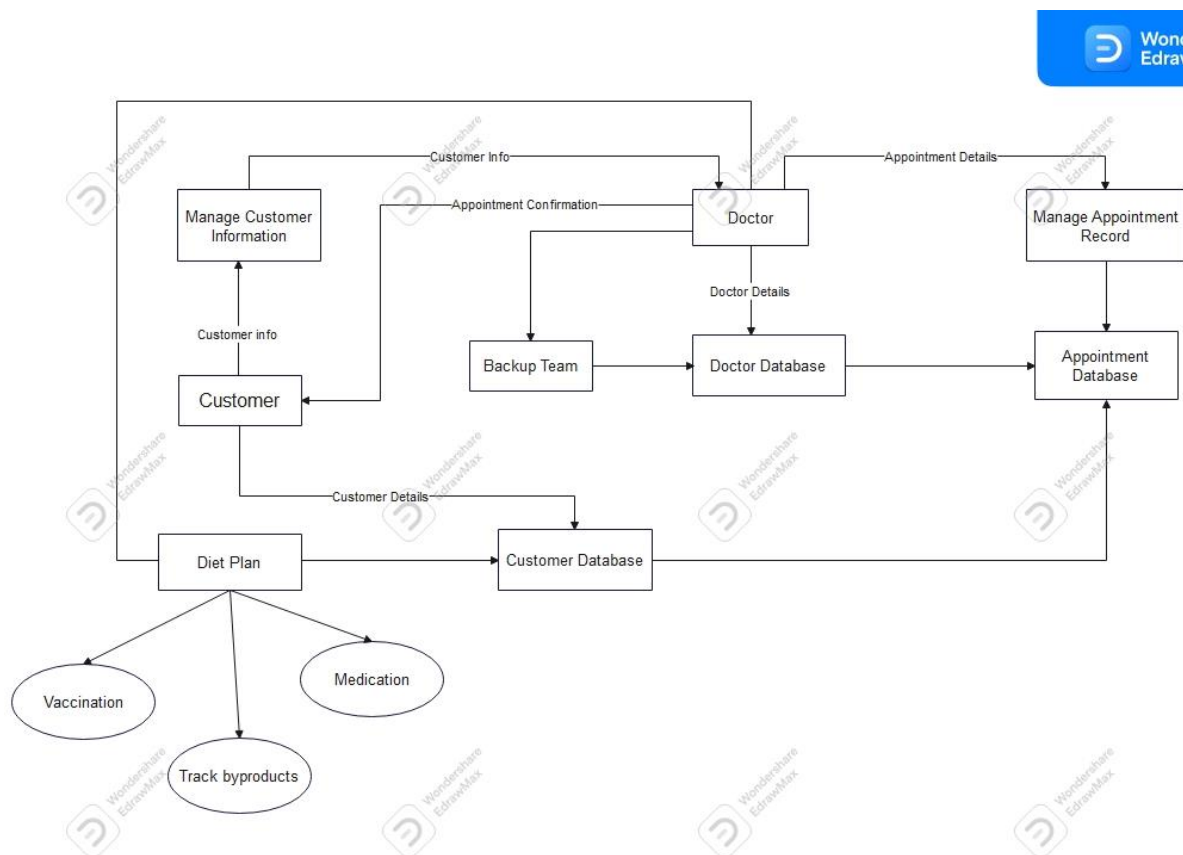
4. Design

4.DESIGN

4.1UML diagrams

4.1.1.ER-Diagram:

4.1.2.Data Flow Diagram(DFD):

Level 0:

Fig: DFD Level 0

Level 1:

Fig : DFD Level 1

Level 2:

5.Table Structure

Doctor:

```
+----------------+--------------+------+-----+---------+-------+
| Field          | Type         | Null | Key | Default | Extra |
+----------------+--------------+------+-----+---------+-------+
| d_id           | int          | NO   | PRI | NULL    |       |
| address        | varchar(255) | YES  |     | NULL    |       |
| contact_no     | varchar(255) | YES  |     | NULL    |       |
| email          | varchar(255) | YES  |     | NULL    |       |
| experience     | varchar(255) | YES  |     | NULL    |       |
| fname          | varchar(255) | YES  |     | NULL    |       |
| lname          | varchar(255) | YES  |     | NULL    |       |
| password       | varchar(255) | YES  |     | NULL    |       |
| specialization | varchar(255) | YES  |     | NULL    |       |
+----------------+--------------+------+-----+---------+-------+
```

Patient:

```
+------------+--------------+------+-----+---------+-------+
| Field      | Type         | Null | Key | Default | Extra |
+------------+--------------+------+-----+---------+-------+
| p_id       | int          | NO   | PRI | NULL    |       |
| address    | varchar(255) | YES  |     | NULL    |       |
| contact_no | varchar(255) | YES  |     | NULL    |       |
| email      | varchar(255) | YES  |     | NULL    |       |
| fname      | varchar(255) | YES  |     | NULL    |       |
| lname      | varchar(255) | YES  |     | NULL    |       |
| password   | varchar(255) | YES  |     | NULL    |       |
+------------+--------------+------+-----+---------+-------+
```

Appointment:

```
+-------+--------------+------+-----+---------+-------+
| Field | Type         | Null | Key | Default | Extra |
+-------+--------------+------+-----+---------+-------+
| ap_id | int          | NO   | PRI | NULL    |       |
| date  | datetime     | YES  |     | NULL    |       |
| time  | varchar(255) | YES  |     | NULL    |       |
| d_id  | int          | YES  | MUL | NULL    |       |
| p_id  | int          | YES  | MUL | NULL    |       |
+-------+--------------+------+-----+---------+-------+
```

Feedback:

```
+--------+--------------+------+-----+---------+-------+
| Field  | Type         | Null | Key | Default | Extra |
+--------+--------------+------+-----+---------+-------+
| f_id   | int          | NO   | PRI | NULL    |       |
| rating | varchar(255) | YES  |     | NULL    |       |
| d_id   | int          | YES  | MUL | NULL    |       |
+--------+--------------+------+-----+---------+-------+
```

Backup:

```
+--------+--------------+------+-----+---------+-------+
| Field  | Type         | Null | Key | Default | Extra |
+--------+--------------+------+-----+---------+-------+
| b_id   | int          | NO   | PRI | NULL    |       |
| b_name | varchar(255) | YES  |     | NULL    |       |
| d_id   | int          | YES  | MUL | NULL    |       |
+--------+--------------+------+-----+---------+-------+
```

5. Implementation

5.1. Modules

Our proposed system that is Customized-FirstChoice consists of three main modules listed as below.

1. Veterinary Module
2. Doctor Module

5.2. Module Description

5.2.1   Veterinary Module

• Veterinary can login or register.

• Select Appointment date and time.

• Book Appointment.

• Veterinary can able to see appointment status after book appointment.

5.2.2   Doctor Module

• Doctor can login or create his own account.

• Doctor can confirm Appointment.

 ## 5.3. <u>Introduction of technologies used</u>

5.3.1 Spring Boot Framework:

Spring Boot provides a good platform for Java developers to develop a stand-alone and production-grade spring application that you can **just run**. You can get started with minimum configurations without the need for an entire Spring configuration setup.

Spring enables you to build applications from "plain old Java objects" (POJOs) and to apply enterprise services non-invasively to POJOs. This capability applies to the Java SE programming model and to full and partial Java EE.

5.3.1.1   Features of Spring boot Framework:

Web Development

It is well suited Spring module for web application development. We can easily create a self-contained HTTP server using embedded Tomcat, Jetty or Undertow. We can use the spring-boot- starter-web module to start and running application quickly.

Spring Application

It is a class which provides the convenient way to bootstrap a spring application which can be started from main method. You can call start your application just by calling a static run () method.

Admin Support

Spring Boot provides the facility to enable admin related features for the application. It is used to access and manage application remotely. We can enable it by simply using spring.application.admin. enabled property.

Logging

Spring Boot uses Common logging for all internal logging. Logging dependencies are managed by default. We should not change logging dependencies, if there is no required customization is needed.

Security

Spring Boot applications are spring bases web applications. So, it is secure by default with basic authentication on all HTTP endpoints. A rich

set of Endpoints are available for develop a secure Spring Boot application.

Advantages of a Spring Boot application

- Fast and easy development of Spring-based applications;
- No need for the deployment of war files;
- The ability to create standalone applications;
- Helping to directly embed Tomcat, Jetty, or Undertow into an application;
- No need for XML configuration;
- Reduced amounts of source code;

1. The JDBC Template

The central class of the Spring JDBC abstraction framework is the **Jdbc Template** class that includes the most common logic in using the JDBC API to access data, such as handling the creation of connection, statement creation, statement execution, and release of resource. The **Jdbc-Template** class can be found in the **org.springframework.jdbc.core** package.

The **Jdbc Template** class instances are thread-safe once configured. A single **Jdbc Template** can be configured and injected into multiple DAOs.We can use the **Jdbc Template** to execute the different types of SQL statements. **Data Manipulation Language** (**DML**) is used for inserting, retrieving, updating, and deleting the data in the database such as **SELECT**, **INSERT**, or **UPDATE** statements

2.1 MySQL

MySQL, the most popular Open-Source SQL database management system, is developed, distributed, and supported by Oracle Corporation.

Features of MySQL:

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- **MySQL databases are relational.**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment.

- **MySQL software is Open Source.**
  Open-Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. The MySQL Database Server is very fast, reliable, scalable, and easy to use.
  MySQL Server works in client/server or embedded systems**.**

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

3. React JS

React JS is JavaScript library used for building reusable UI components. According to React official documentation, following is the definition −

React is a library for building composable user interfaces. It encourages the creation of reusable UI components, which present data that changes over time. Lots of people use React as the V in MVC. React abstracts away the DOM from you, offering a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using React Native. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding.

React Features

- **JSX** − JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.

- **Components** − React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.

- **Unidirectional data flow and Flux** − React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keeping your data unidirectional.

- **License** − React is licensed under the Facebook Inc. Documentation is licensed under CC BY 4.0.
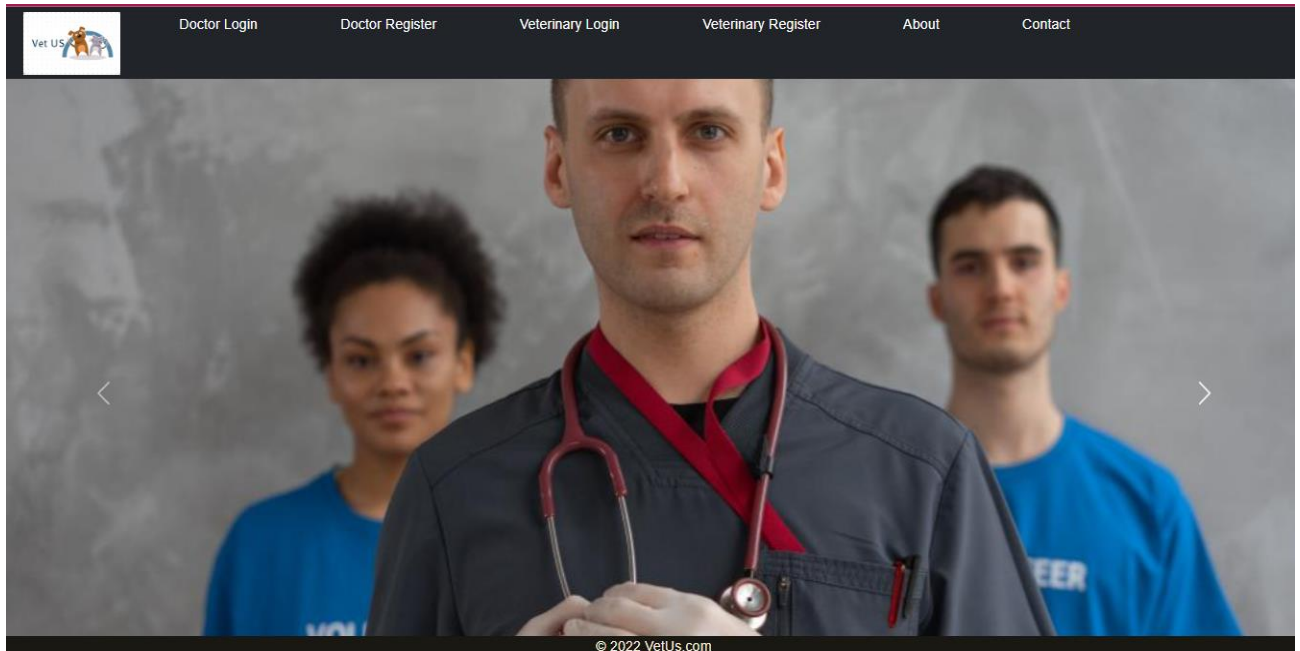
React Advantages

- Uses virtual DOM which is a JavaScript object. This will improve apps performance, since JavaScript virtual DOM is faster than the regular DOM.

- Can be used on client and server side as well as with other frameworks.

- Component and data patterns improve readability, which helps to maintain larger apps.

6.Test cases

Test case id	Test case Name	Test case description	Expected Result	Actual result	status
1	Doctor Login	Doctor need to register first and then enter his customer id and password for login.	Doctor can see his homepage	Doctor is on his home page	successful
2	Veterinary Login	Veterinary need to register first and then enter his customer id and password for login.	Veterinary can see his homepage	Veterinary is on his home page	successful
3	Book Appointment	Veterinary need to login first and then book Appointment	Veterinary can see his homepage	Veterinary can see Appointment status.	successful
4	Confirm Appointment	Doctor need to confirm/cancel/send backup to veterinary.	Appointment status show on Veterinary page.	Doctor can see Appointment status.	successful

7.Screenshots of Webpages

Homepage
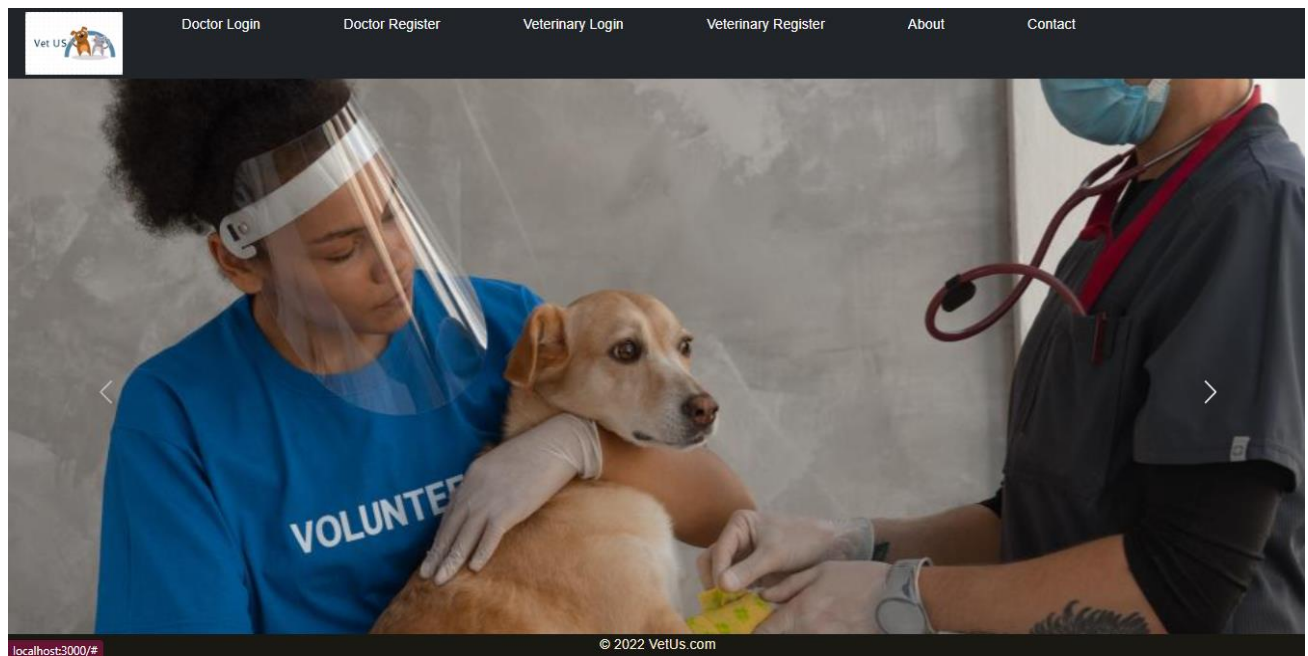
Doctor Registration:

Doctor Login:

<u>Veterinary Register:</u>

<u>Veterinary Login:</u>

<u>Appointment:</u>

Appointments

Appointment ID	First Name	Phone Number	Address	Appointment Date	Appointment Time	Status	Action
5	Nitin	9144461010	Kolhapur	2022-04-09T00:00:00.000+00:00	09:15 AM - 09:30 AM	Confirmed	Backup Confirm Cancel

© 2022 VetUs.com

"Never believe that animals suffer less than humans.
Pain is the same for them that it is for us.
Even worse, because they cannot help themselves.""

Appointment ID	Doctor Name	Appointment Date	Appointment Time	Appointment Status
5	Chaitanya Waghmode	2022-04-09T00:00:00.000+00:00	09:15 AM - 09:30 AM	Confirmed

© 2022 VetUs.com

TESTING

To build up our project we used software testing process for executing a program with the intent of finding error that is uncovering errors in a program makes it a feasible task and also typing to find the errors (whose presence is assumed) in a program. As it is a destructive process.
Types of testing we use in our project

Here we just mentioned that how the testing is related to this software and in which way we have test the software? In our project we have used five types of testing this are listed below –

UNIT TESTING –
Unit testing where individual program units or object class are tested here by using this testing, we have focus on testing functionality of the methods.

MODULE TESTING–
Where this is the combination of unit program is called module. Here we tested unit program is where the module program have dependency.

SUB SYSTEM TESTING –
Then we combined some module for the preliminary system testing in out project.

SYSTEM TESTING –
Where it is combination of two or more sub system and then it is tested here we tested the entire system a per requirement.

ACEEPTANCE TESTING –
Normally this type of testing is done to verify if system meets the customer specified requirements. After submitting this project to the user then they tested and to determine whether to accept the application. It is the system of testing performed by the customer to determine where they should accept the delivery of system.

Conclusion

VET US provides better platform to get connected with Doctors and Patients efficiently. Our System provides a very user-friendly platform where veterinary patient can easily look for the doctors available within the clinic, book an appointment.

Future Enhancement

1. If the animal have any major injury then user can also book the transport facility for their convenience.
2. We can also book medicine through this application.
3. We can also provide supplements.

Bibliography

https://www.javatpoint.com/spring-boot-tutorial

https://www.w3schools.com/REACT/DEFAULT.ASP

https://www.javatpoint.com/mysql-tutorial

https://www.youtube.com/watch?v=36WoQ1anwM0

Thank You!