

A PROJECT ON
Veterinary Clinic Management System

SUBMITTED IN
PARTIAL FULFILLMENT OF THE REQUIREMENT
FOR THE COURSE OF DIPLOMA IN ADVANCED COMPUTING FROM CDAC



SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY
Hinjawadi

SUBMITTED BY:

Ahire Bhavesh Sunilbhai,

Gade Onkar Popat,

Anubhav Tewari,

Samarth Cherian

UNDER THE GUIDANCE OF:

Ms.Lalita Shinde

Faculty Member

Sunbeam Institute of Information Technology, Pune

ACKNOWLEDGEMENT

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We avail this opportunity to express our deep sense of gratitude towards Mr. Nitin Kudale (Center Coordinator, SIIT, Pune) and Mr. Yogesh Kolhe (Course Coordinator, SIIT ,Pune) .

We are deeply indebted and grateful to them for their guidance, encouragement and deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form.

Last but not the least we thank the entire faculty and the staff members of Sunbeam Institute of Information Technology, Pune for their support.

Ahire Bhavesh Sunilbhai,
Onkar Gade Popat,
Anubhav Tewari,
Samarth Cherian

SIIT Pune

A PROJECT ON
"Veterinary Clinic Management System"

SUBMITTED IN
PARTIAL FULFILLMENT OF THE REQUIREMENT
FOR THE COURSE OF
DIPLOMA IN ADVANCED COMPUTING FROM CDAC



SUNBEAM INSTITUTE OF INFORMATION TECHNOLOGY
Hinjawadi

SUBMITTED BY:

Ahire Bhavesh

Sunilbhai, Gade

Onkar Popat,

Anubhav Tewari,

Samarth Cherian

UNDER THE GUIDENCE OF:

Ms.Lalita

Shinde

Faculty Member
Sunbeam Institute of Information Technology, PUNE.



CERTIFICATE

This is to certify that the project work under the title '**Veterinary Clinic Management System**' is done by Ahire Bhavesh Sunilbhai, Gade Onkar Popat, Anubhav Tewari, Samarth Cherian in partial fulfillment of the requirement for award of Diploma in Advanced Computing Course.

Ms. Lalita Shinde

Project Guide

Mr. Yogesh Kolhe

Course Co-Coordinator

Date: 12-02-2025

Table of Contents:

1. INTRODUCTION	2
2. REQUIREMENTS	3
2.1 Functional Requirements.....	3
2.1 Admin Module	4
2.2 Doctor Module.....	6
2.3 Pet Owner Module	9
2.4 Receptionist Module	11
3. Non-Functional Requirements	12
3.3.1 Hardware and Software Interfaces.....	12
4. DESIGN.....	13
4.1 Database design.....	13
5. CODING STANDARD IMPLEMENTED.....	19
6. TEST REPORT.....	22
7. APENDIX A	
1. Entity Relationship Diagram.....	27
2. Data Flow Diagram.....	28
3. Class Diagram.....	29
8. APENDIX B	
UI Screenshots.....	30
9. REFERENCES	

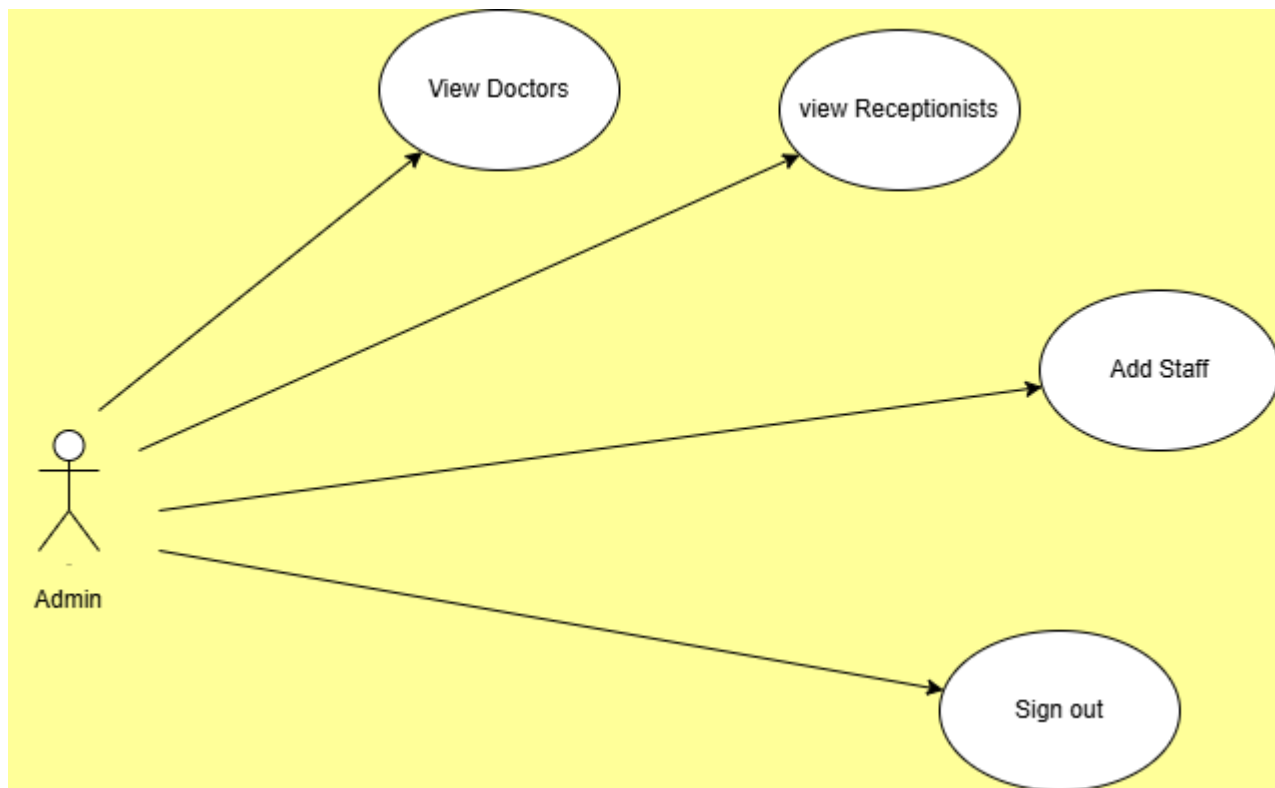
1. INTRODUCTION TO PROJECT

The **Veterinary Clinic Management System** is an application that helps to manage the day-to-day activities of a clinic, like adding appointments, managing patient records, prescriptions, and billings. The system ensures efficient management of veterinary services by automating administrative tasks, reducing paperwork, and improving overall workflow. It includes functionalities like digital medical records for pets, invoice generation, and secure data storage.

The Veterinary Clinic Management System is an application that helps manage day-to-day activities of a clinic, such as adding appointments, managing patient records, prescriptions, and billings. Built using Spring Boot for backend logic, REST APIs, and secure data processing, it uses JWT authentication for secure login and access control. The front end is developed with React for dynamic user interfaces, styled with Bootstrap to enhance design and responsiveness. Data is securely stored using MySQL as the database for reliable record handling. The system ensures efficient management of veterinary services by automating tasks, reducing paperwork, and improving workflow. It includes functionalities like digital medical records for pets, invoice generation, and secure data storage.

2. REQUIREMENTS

2.1 FUNCTIONAL REQUIREMENTS



2.1 Admin Flow

2.1.1 Home Page

- **Objective:** Display a list of Doctors, Receptionists, Add Staff
- **Features:**
 - Click to view List of Doctors, Receptionists, and with functionality to add staff

2.1.2 Doctors

- **Objective:** Allow admin to view and manage Doctors
- **Features:**
 - Click on a Doctors to see the list of Doctors.

2.1.3 Receptionists

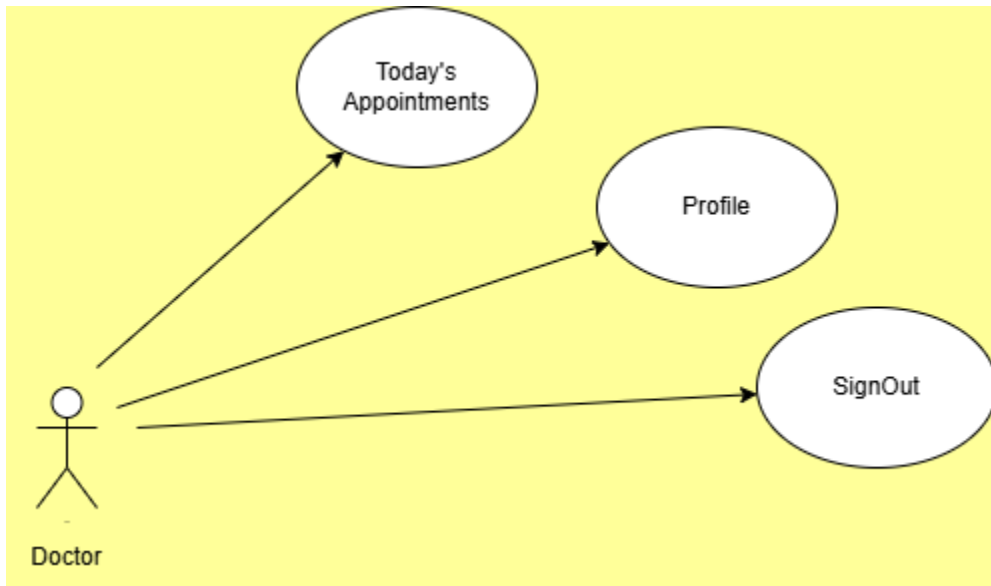
- **Objective:** Allow admin to view and manage Receptionists.
- **Features:**
 - Click on Receptionists to see the list of Receptionists.

2.1.4 Add Staff

- **Objective:** Allow admin to view and manage Staff of the clinic.
- **Features:** Admin can add or delete staff by adding their email and password.

2.1.5 Sign out

- **Objective:** Just to sign out.



2.2 Doctor Flow

2.2.1 Home Page

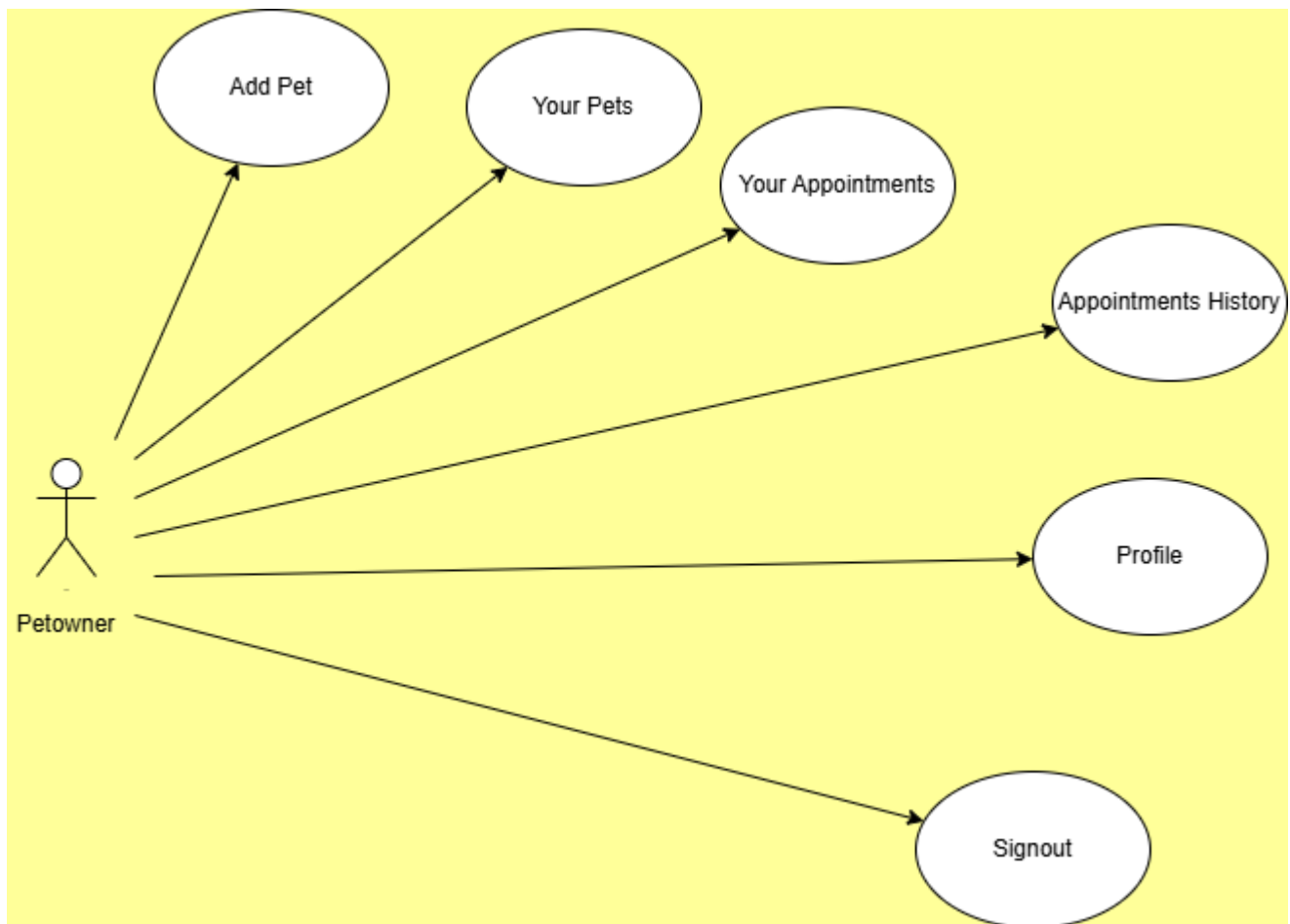
- **Objective:** Allow Doctors to manage their appointments.
- **Features:**
 - Displays the list of upcoming appointments

2.2.2 Doctor Profile Page

- **Objective:** To view and Edit Doctor's profile
To edit the profile

2.2.3 Doctor Signout

- **Objective:** To signout.



2.3 Pet Owner Flow

2.3.1 Add Pet

- **Objective:** Add Pet
- **Features:**
 - Add information about pet like gender ,age,breed etc.

2.3.2 Appointment Page

- To see all Appointments

2.3.3 Your Pets

- To see all your pets

2.3.4 Appointment History

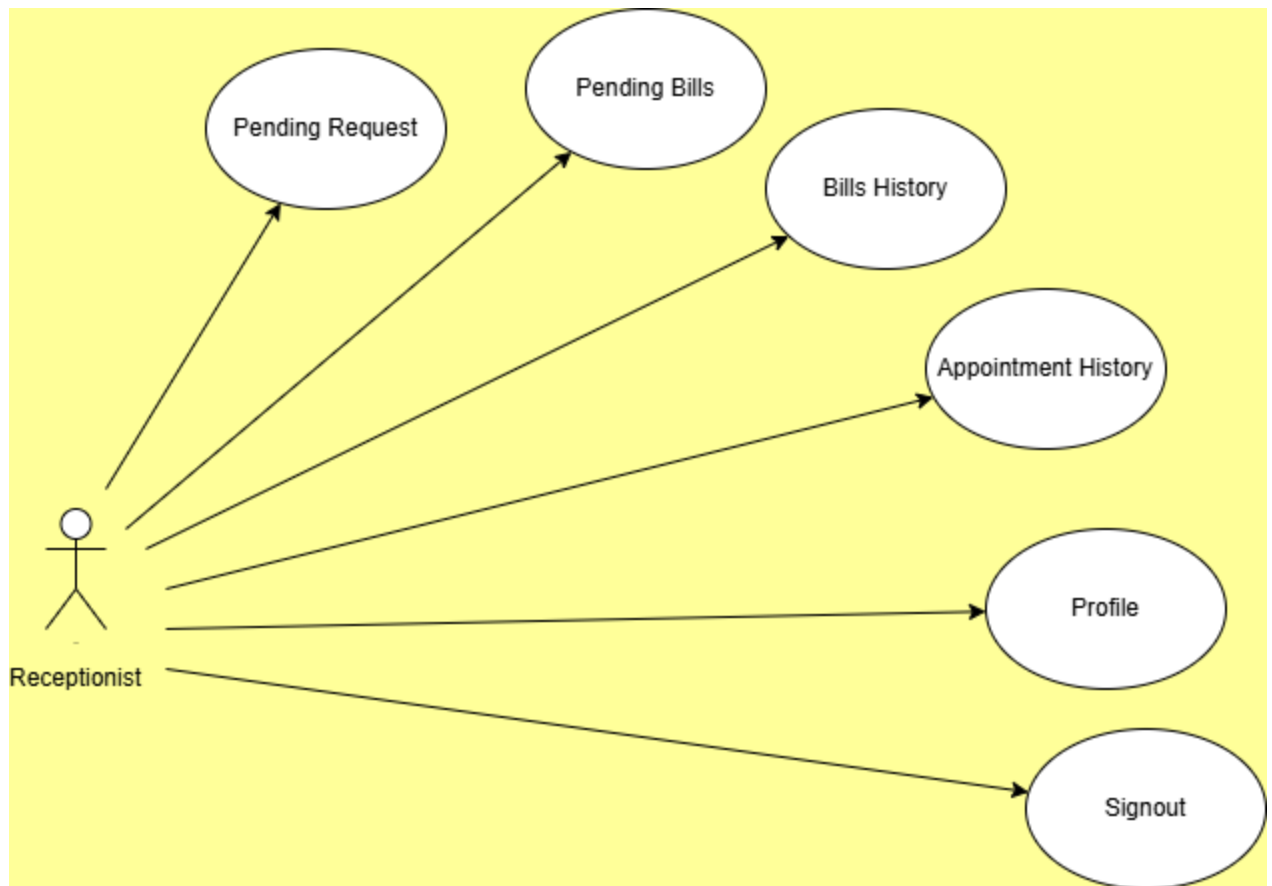
- To see your appointment history

2.3.5 Profile

- To view and modify your profile

2.3.6 Signout

- To signout



2.4 Receptionist Flow

2.4.1 Home Page

- **Objective:** TO see all Pending requests
- **Features:**
 - Receptionist can see all the pending request and approve them.

2.4.2 Profile

- **Objective:** Profile.
- **Features:** The receptionist can view and edit their profiles.

3. Non-Functional Requirements

3.1 Interface

- User interfaces must be intuitive and user-friendly.
Detailed designs are provided in Appendix B.

3.2 Performance

- **Number of Concurrent Users:** The system should handle at least 1000 transactions/inquiries per second.
- **System Resilience:** The application should be resilient to temporary server failures.

3.3 Constraints

- The system should maintain performance standards of handling 1000 transactions/inquiries per second.

3.4 Other Requirements

3.4.1 Hardware Interfaces

Requirements: Intel Core i5 or higher (or AMD equivalent), 8 GB RAM, 512 GB SSD or larger.

-

3.4.2 Software Interfaces

- **Operating Systems:** MS Windows 11, Ubuntu 22.04.
- **Database:** MySQL.
- **Server:** Embedded Tomcat.
- **Browsers:** Compatible with modern web browsers.

4. System Design

4.1 Architecture

- **Front-End:** Developed using React.js
- **Back-End:** Built with Spring Boot for server-side logic.
- **Database:** MySQL for storing user data, orders, and other system information.
- **Server:** Embedded Tomcat for hosting the application.

4. DESIGN

4.1 Database Design

The following table structures depict the database design.

Table 1: Appointments

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
appoint_date	date	YES		NULL	
appoint_time	time(6)	YES		NULL	
illness	varchar(100)	YES		NULL	
status	enum('APPROVED', 'COMPLETED', 'DENIED', 'PENDING')	YES		NULL	
doctor_id	bigint	YES	MUL	NULL	
owner_id	bigint	YES	MUL	NULL	
pet_id	bigint	YES	MUL	NULL	
receptionist_id	bigint	YES	MUL	NULL	

Table 2: Billings

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
billing_date	datetime(6)	YES		NULL	
status	enum('APPROVED', 'COMPLETED', 'DENIED', 'PENDING')	YES		NULL	
total_amount	double	YES		NULL	
presc_id	bigint	NO	UNI	NULL	

Table 3: Doctors

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
aadhar_no	varchar(12)	YES		NULL	
degree	varchar(20)	YES		NULL	
specialist	varchar(20)	YES		NULL	
uid	bigint	NO	UNI	NULL	

Table 4: Medicines

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
name_of_medicine	varchar(255)	NO	UNI	NULL	
qty	int	NO		NULL	
presc_id	bigint	NO	UNI	NULL	

Table 5:Pet Owners

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
uid	bigint	YES	UNI	NULL	

Table 6:Pets

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
age	int	YES		NULL	
breed	varchar(50)	YES		NULL	
gender	enum('FEMALE', 'MALE', 'OTHER')	YES		NULL	
name	varchar(50)	YES		NULL	
species	varchar(50)	YES		NULL	
owner_id	bigint	YES	MUL	NULL	

Table 7:Prescriptions

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
diagnosis	varchar(60)	YES		NULL	
pres_advice	varchar(255)	YES		NULL	
appointment_id	bigint	NO	UNI	NULL	

Table 8 : receptionists

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
aadhar_no	varchar(12)	YES		NULL	
qualification	varchar(20)	YES		NULL	
uid	bigint	NO	UNI	NULL	

Table 9: Users

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	date	YES		NULL	
is_available	bit(1)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
address	varchar(255)	YES		NULL	
dob	date	YES		NULL	
email	varchar(50)	NO	UNI	NULL	
firstname	varchar(50)	YES		NULL	
gender	enum('FEMALE', 'MALE', 'OTHER')	YES		NULL	
image	longblob	YES		NULL	
lastname	varchar(50)	YES		NULL	
password	varchar(255)	YES		NULL	
phone_no	varchar(13)	YES		NULL	
role	enum('ADMIN', 'DOCTOR', 'PETOWNER', 'RECEPTIONIST')	YES		NULL	

E-R Diagram, Dataflow diagram and Class Diagram:

Go to Appendix A

5. CODING STANDARDS IMPLEMENTED

Naming and Capitalization

Below summarizes the naming recommendations for identifiers in Pascal casing is used mainly (i.e. capitalize first letter of each word) with camel casing (capitalize each word except for the first one) being used in certain circumstances.

Identifier	Case	Examples	Additional Notes
Class	Pascal	UserController	Class names should be based on "objects" or "real things" and should generally be nouns . No '_' signs allowed. Do not use type prefixes like 'C' for class.

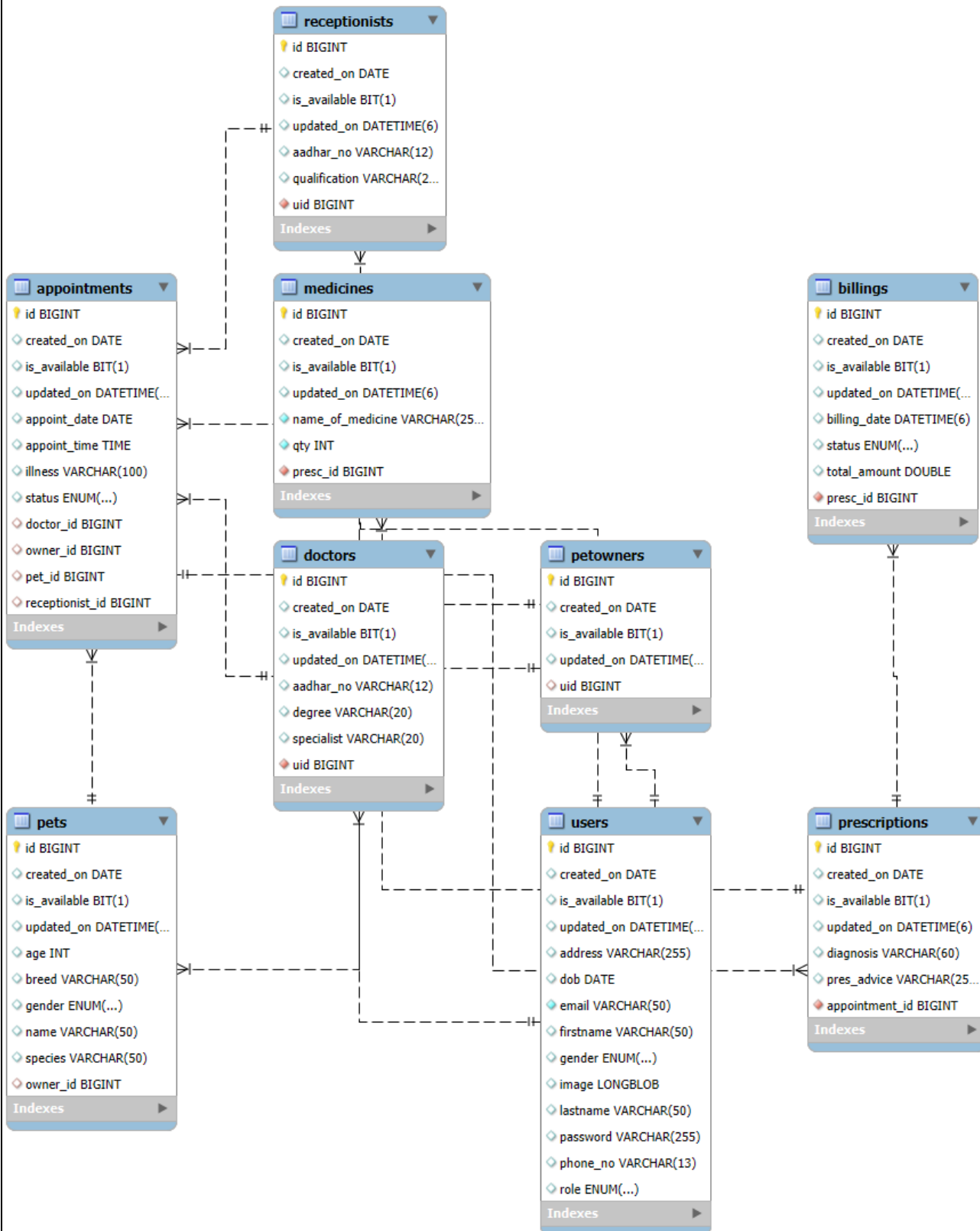
Method	Camel	SignUp, SignIn,	Methods should use verbs or verb phrases.
Parameter	Camel	firstName, lastName, email, password	Use descriptive parameter names. Parameter names should be descriptive enough that the name of the parameter and its type can be used to determine its meaning in most scenarios.
Interface	Pascal with "I" prefix	UserRepository, DoctotRepositor y, PetOwner Repository	Do not use the '_' sign
Annotation	Pascal	SpringBootApplication	Use @ at start of annotation
DTOs	Camel	ApiResponseDTO, SignUpReqDTO, OrderDetailsRes DTO	Use to transfer data between the processes
Exception Class	Pascal with "Exception" suffix	ResourceNotFoun dException	

Comments

- Comment each type, each non-public type member, and each region declaration.

DATE	WORK PERFORMED	SLC PHASE	Additional Notes
Sept 11, 2024	Project Allotment and User Requirements Gathering	Feasibility Study	Our team met the client Mr. Nitin Kudale (CEO, SIIT Pune) to know his requirements.
Sept 17, 2024	Initial SRS Document Validation and Team Structure Decided	Requirement Analysis	The initial SRS was presented to the client to understand his requirements better.
Sept 30, 2024	Designing the use-cases, Class Diagram, Collaboration Diagram, E-R Diagram, and User Interfaces	(Elicitation) Requirement Analysis & Design Phase	Database Design completed.
Oct 1, 2024	Business Logic Component Design Started	Design Phase	----- -
Oct 5, 2024	Coding Phase Started	Coding Phase	70% of Class Library Implemented.
Oct 15, 2024	Implementation of Web Application and Window Application Continued	Coding Phase and Unit Testing	Class Library Modified as per the need.
Dec 1, 2024	Implementation of Web Application and Window Application Continued After Ensuring	Coding Phase and Unit Testing	----- -
Dec 15, 2024	Functioning the Required Validation were Implemented	Coding Phase and Unit Testing	Module Integration was done by the Project Manager
Dec 24, 2024	The Project was by the respective Leaders and the Manager	Testing Phase (Module Testing)	----- -
Dec 30, 2024	The Project was	Testing Phase	The Project of Other Team was Taken up by the Team for Testing
Jan 2, 2025	Project Leader of Other Project Group For Testing	er (Acceptance Testing)	The Project was complete for submission
Jan 17, 2025	The Errors Found were Removed	Debugging	----- ----- -
Jan 12, 2025	Final Submission of Project		

Entity Relationship Diagram



UI

Signin Page:

Sign in

Email address
abhi@gmail.com

Password

SIGN IN

Don't Have Account? [Sign Up](#)

SignupPage (PetOwner) :

Sign Up

First Name

Last Name

Email

Date Of Birth

dd-mm-yyyy

Phone No

Password

Please fill out this field.

Re-Entered Password

Gender

Select Gender

Role

Pet Owner

Address

CLOSE

SIGN UP

Admin Page:

Admin Panel

 Doctors

 Receptionists

 Add Staff

 Sign Out

Add Staff :

Admin Panel

 Doctors  Receptionists  Add Staff  Sign Out

Role
Select Role 

Email

Password


Re-Enter Password

SUBMIT

View:Doctors

Admin Panel

 Doctors  Receptionists  Add Staff  Sign Out

No.	Name	Email	Phone No.	Status	Action
1	samarth cherian	samarth@gmail.com	8959846241	Active	 

View:Receptionists

Admin Panel

 Doctors  Receptionists  Add Staff  Sign Out

No.	Name	Email	Phone No.	Status	Action
1	siddharth jat	siddharth@gmail.com	8959846242	Active	 
2	akash singh	akash@gmail.com	8959846243	Active	 

Doctors:Sign in

Doctor Panel

☰ Today's Appointment 👤 Profile ➡ Sign Out

No.	Owner Email	Date	Time	Illness	Actions
1	ritvik@gmail.com	2025-02-11	13:26:00	sneezing	<div><div>VIEW HISTORY</div><div>ATTEND</div></div>

YOUR APPOINTMENTS

Doctors : Profile

Doctor Panel

Today's Appointment Profile Sign Out

User Profile



samarth cherian

samarth@gmail.com

First Name

samarth

Last Name

cherian

Email

samarth@gmail.com

Date of Birth

1993-02-21

Phone No

8959846241

Address

jabalpur

EDIT PROFILE

Pet History:

Pet History

No.	Diagnosis	Prescription
No records found		

CLOSE

Receptionist :approve appointments

Receptionist Panel

 Pending Requests  Pending Bills  Appointment History  Profile  Sign Out

No.	Owner Email	Appointment Date	Appointment Time	Illness	Actions
1	ritvik@gmail.com	2025-02-11	16:21:00	ill	<div><div>✓</div><div>✗</div></div>

Receptionist: Pending bills

Receptionist Panel

 Pending Requests  Pending Bills  Appointment History  Profile  Sign Out

No.	Owner Name	Amount	Actions
1	ritvik dasu	700	<button>PAID</button>
2	ritvik dasu	700	<button>PAID</button>

Receptionist:Appointment history

Receptionist Panel

 Pending Requests  Pending Bills  Appointment History  Profile  Sign Out

No.	Owner Email	Appointment Date	Appointment Time	Illness
1	abhi@gmail.com	2025-02-12	15:15:00	fever
2	abhi@gmail.com	2025-02-12	15:16:00	vomiting
3	ritvik@gmail.com	2025-02-11	12:07:00	fur problem
4	ritvik@gmail.com	2025-02-11	12:15:00	grooming
5	ritvik@gmail.com	2025-02-11	16:21:00	ill
6	ritvik@gmail.com	2025-02-11	13:26:00	sneezing
No More Data Found				

Receptionist:Profile

Receptionist Panel

⌵ Pending Requests 📄 Pending Bills 📅 Appointment History 👤 Profile 🚪 Sign Out

User Profile



akash singh

akash@gmail.com

First Name

akash

Last Name

singh

Email

akash@gmail.com

Date of Birth

Phone No

8959846243

Address

sadar jbp

EDIT PROFILE

Petowner: Login

Pet Owner Panel

[🐾 Your Pet](#) [📅 Your Appointment](#) [🕒 History](#) [👤 Profile](#) [🚪 Sign Out](#)

ADD NEW PET

TOMMY

Species: dog
Breed: rottweiler
Age: 9



CASPER

Species: cat
Breed: egyptian cat
Age: 4



Get connected with us on social networks:



VETERINARY CLINIC

A pet clinic is a healthcare facility for pets, offering services like check-ups, vaccinations, treatments, and surgeries. Staffed by veterinarians, it ensures pets' health and well-being.

USEFUL LINKS

[Your Account](#)
[About Us](#)
[Help](#)

CONTACT

[🏠 Hinjewadi, Pune 10012, India](#)
[✉ info@example.com](#)
[☎ + 01 234 567 88](#)
[📠 + 01 234 567 89](#)

Snipping Tool

Screenshot copied to clipboard
Automatically saved to screenshots folder.

Mark-up and share

PetOwner:Your Appointments

Pet Owner Panel

[🐾 Your Pet](#) [📅 Your Appointment](#) [🕒 History](#) [👤 Profile](#) [🚪 Sign Out](#)

📅 BOOK APPOINTMENT

No.	Appointment Date	Appointment Time	Pet Name	Status
1	2025-02-11	13:26:00	casper	APPROVED

Get connected with us on social networks:



VETERINARY CLINIC

A pet clinic is a healthcare facility for pets, offering services like check-ups, vaccinations, treatments, and surgeries. Staffed by veterinarians, it ensures pets' health and well-being.

USEFUL LINKS

[Your Account](#)
[About Us](#)
[Help](#)

CONTACT

🏠 Hinjewadi, Pune 10012, India
✉ info@example.com
☎ + 01 234 567 88
📠 + 01 234 567 89

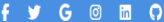
Petowner:Appointment History

Pet Owner Panel

[🐾 Your Pet](#) [📅 Your Appointment](#) [🕒 History](#) [👤 Profile](#) [🚪 Sign Out](#)

No.	Appointment Date	Appointment Time	Pet Name	Status
1	2025-02-11	12:07:00	casper	COMPLETED
2	2025-02-11	12:15:00	tommy	COMPLETED
3	2025-02-11	16:21:00	tommy	COMPLETED

Get connected with us on social networks:



VETERINARY CLINIC

A pet clinic is a healthcare facility for pets, offering services like check-ups, vaccinations, treatments, and surgeries. Staffed by veterinarians, it ensures pets' health and well-being.


USEFUL LINKS

[Your Account](#)
[About Us](#)
[Help](#)

CONTACT

🏠 Hinjewadi, Pune 10012, India
✉ info@example.com
☎ + 01 234 567 88
📠 + 01 234 567 89

User Profile



ritvik dasu

ritvik@gmail.com

First Name

ritvik

Last Name

dasu

Email

ritvik@gmail.com

Date of Birth

2004-01-13

Phone No

8959846249

Address

mumbai

EDIT PROFILE

6. REFERENCES

1. **Spring Boot Documentation**
URL: <https://spring.io/projects/spring-boot>
2. **React.js Documentation**
URL: <https://reactjs.org/docs/getting-started.html>
3. **Redux Documentation**
URL: <https://redux.js.org>
4. **Java Programming Language**
URL: <https://www.oracle.com/java/>
5. **MySQL Workbench Documentation**
URL: <https://dev.mysql.com/doc/workbench/en/>
6. **Spring Boot with React and Redux**
URL: <https://www.baeldung.com/spring-boot-react-and-redux>
7. **Java Persistence API (JPA) Documentation**
URL: <https://www.eclipse.org/eclipselink/documentation/2.7/>
8. **Swagger Documentation for Spring Boot**
URL: <https://springdoc.org/>
9. **MDN Web Docs**
URL: <https://developer.mozilla.org/>
10. **React Redux Integration Guide**
URL: <https://react-redux.js.org/>