



Mini project report on
PET CLINIC

Submitted in partial fulfilment of the requirements for the award of degree of

Bachelor of Technology
in
Computer Science & Engineering
UE20CS352 –OOADJ Project

Submitted by:

TEAM 16

K VASANTH KARTHIK
V NIMISH BHASU
SURAJ P BHADANIA

PES2UG20CS515

PES2UG20CS479

PES2UG20CS470

Under the guidance of

Prof. Nivedita Kasturi

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

FACULTY OF ENGINEERING

PES UNIVERSITY



TABLE OF CONTENTS

Chapter No.	Title	Page No.
1.	INTRODUCTION	
2.	PROBLEM DEFINITION	
3.	USECASE MODELING	
4.	CLASS MODELING	
5.	ACTIVITY MODELING	
6.	STATE MODELING	
7.	IMPLEMENTATION	
8.	RESULTS SCREENSHOTS	

NOTE: Please add appropriate description for all diagrams where ever required. Only important class implementation needs to be added to IMPEMNTATION SECTION.

INTRODUCTION

Owning a pet comes with a lot of responsibilities, including managing their healthcare needs. Pet owners often struggle to keep track of their pet's schedule and appointments, which can lead to missed appointments and compromised pet health. With the advancement in technology, a pet clinic website can now offer a simple and efficient way to manage all aspects of pet healthcare. By providing an online platform for pet owners to book appointments, check their pet's schedule, and access medical information, pet clinics can make it easier for pet owners to stay on top of their pet's health needs.

Pet clinic website requests user to sign up first in order to access their facilities. The registered user can login and book doctor appointments for their pets , check the schedule of their pet and even check the commonly occurring diseases in their pet along with the medical procedure which needs to be followed when the disease is caused.

Pet clinic website is a maven project developed using spring-boot and it uses H2-database. The website also uses HTML/CSS in the jsp file format for some static webpages.

The website uses Model-View-Controller architecture and it uses the following design patterns:

1. Façade Design Pattern
2. Proxy Design Pattern
3. Template Design Pattern
4. Singleton Design Pattern

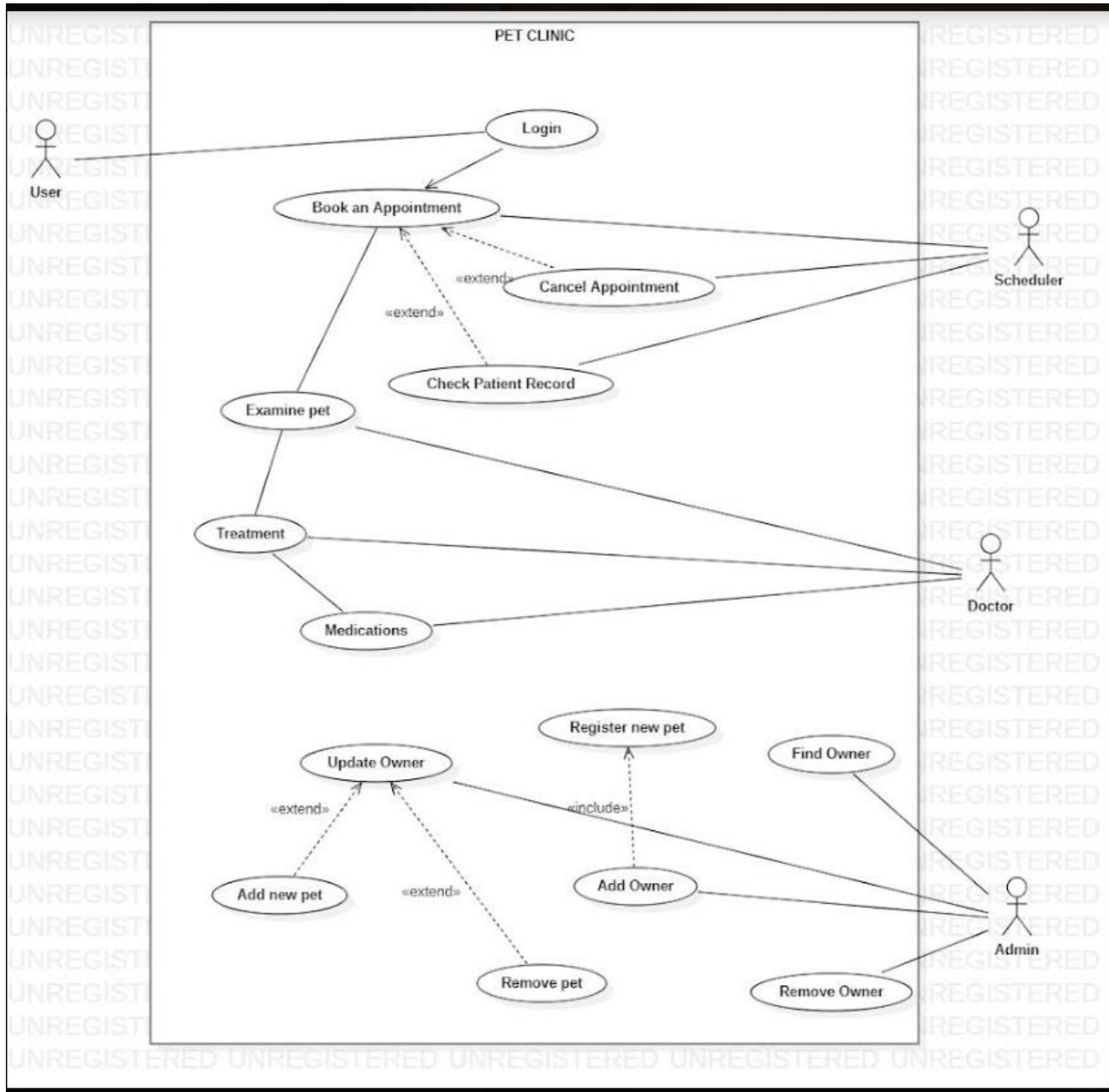
PROBLEM DEFINITION

The pet clinic requires an automated management system to handle their daily operations such as appointment scheduling, pet registration, medical history tracking etc.

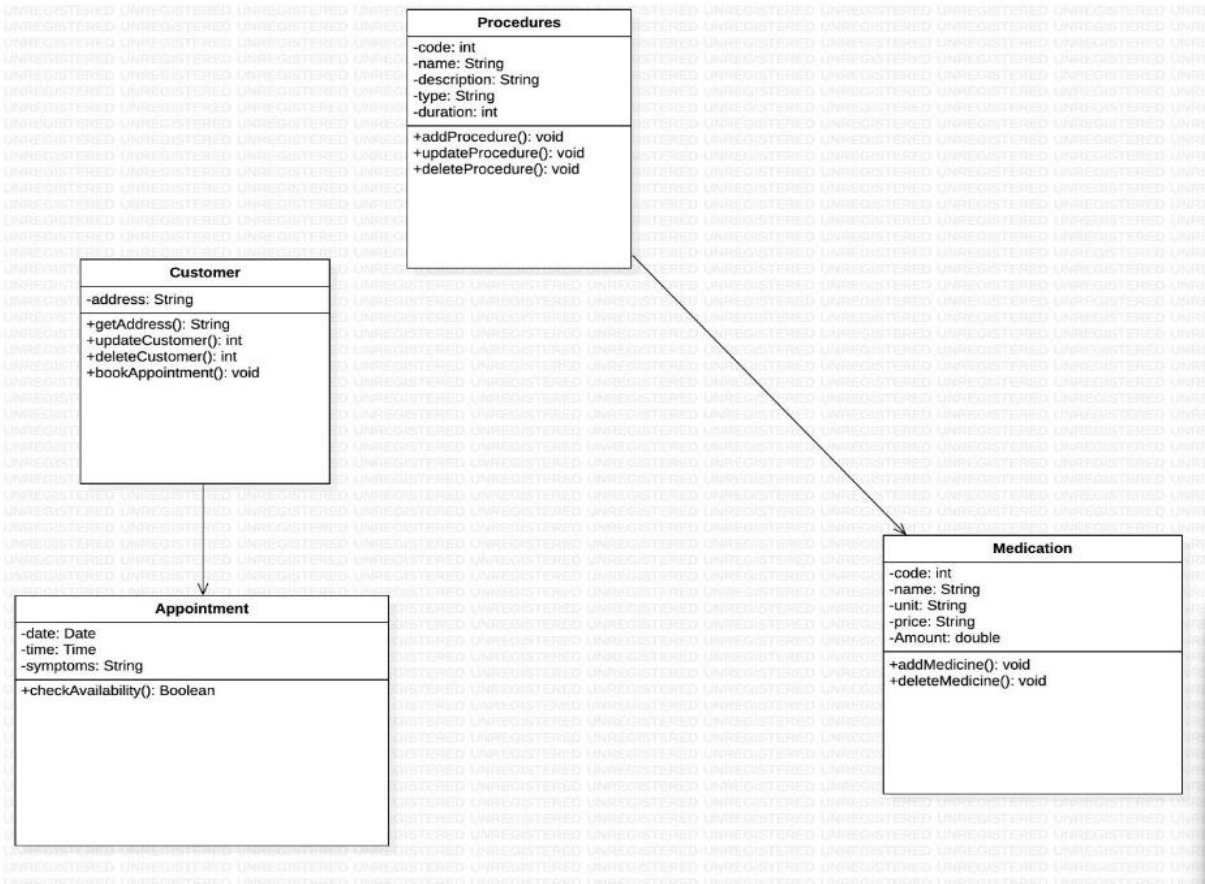
The solution must be built using the Spring Boot framework and the H2 database to ensure scalability, reliability, and maintainability. The system should have a user-friendly interface with role-based access control to guarantee data security.

The ultimate goal is to create a modern and efficient management system that enhances service quality, streamlines operations, and improves the customer experience.

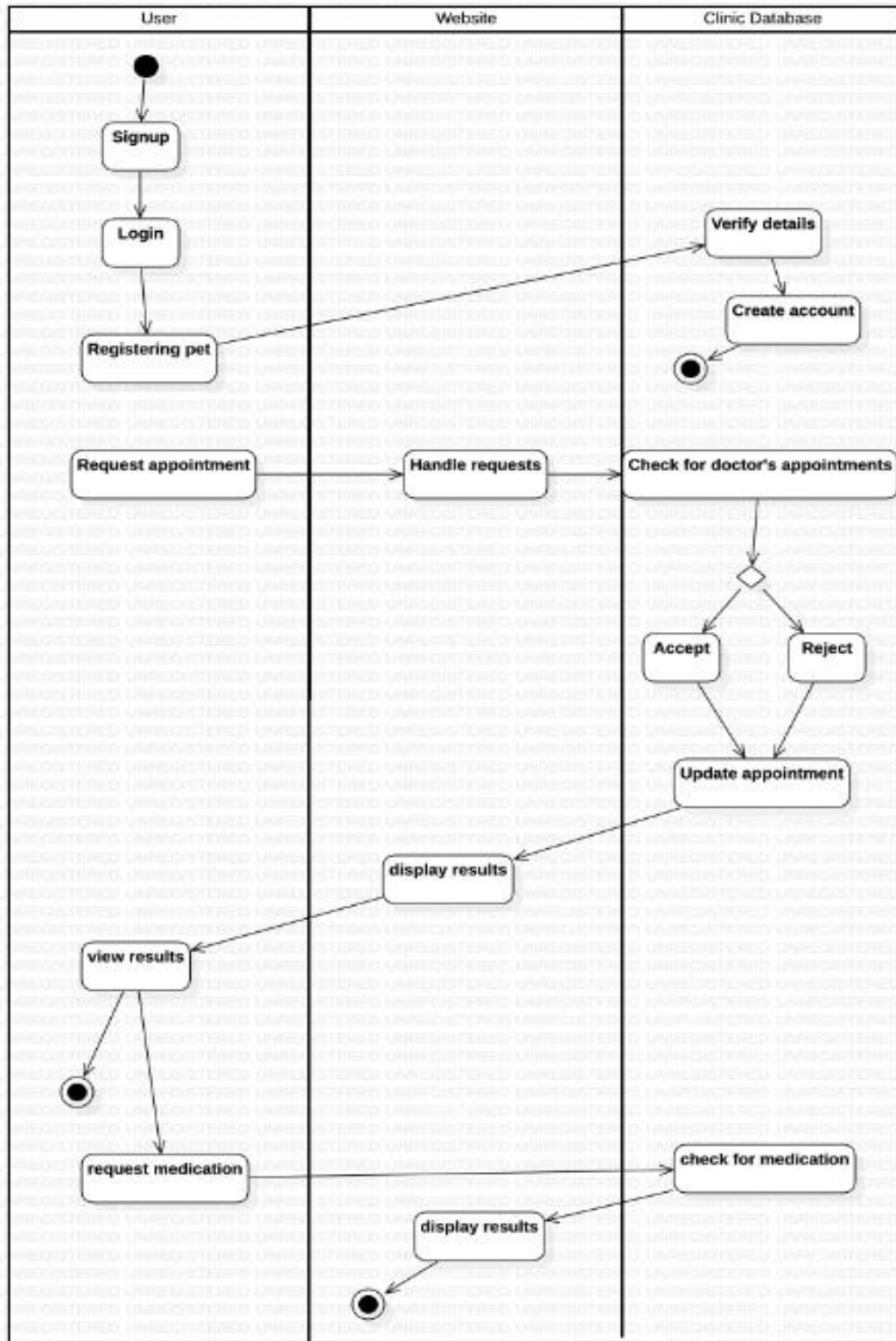
USECASE MODELING



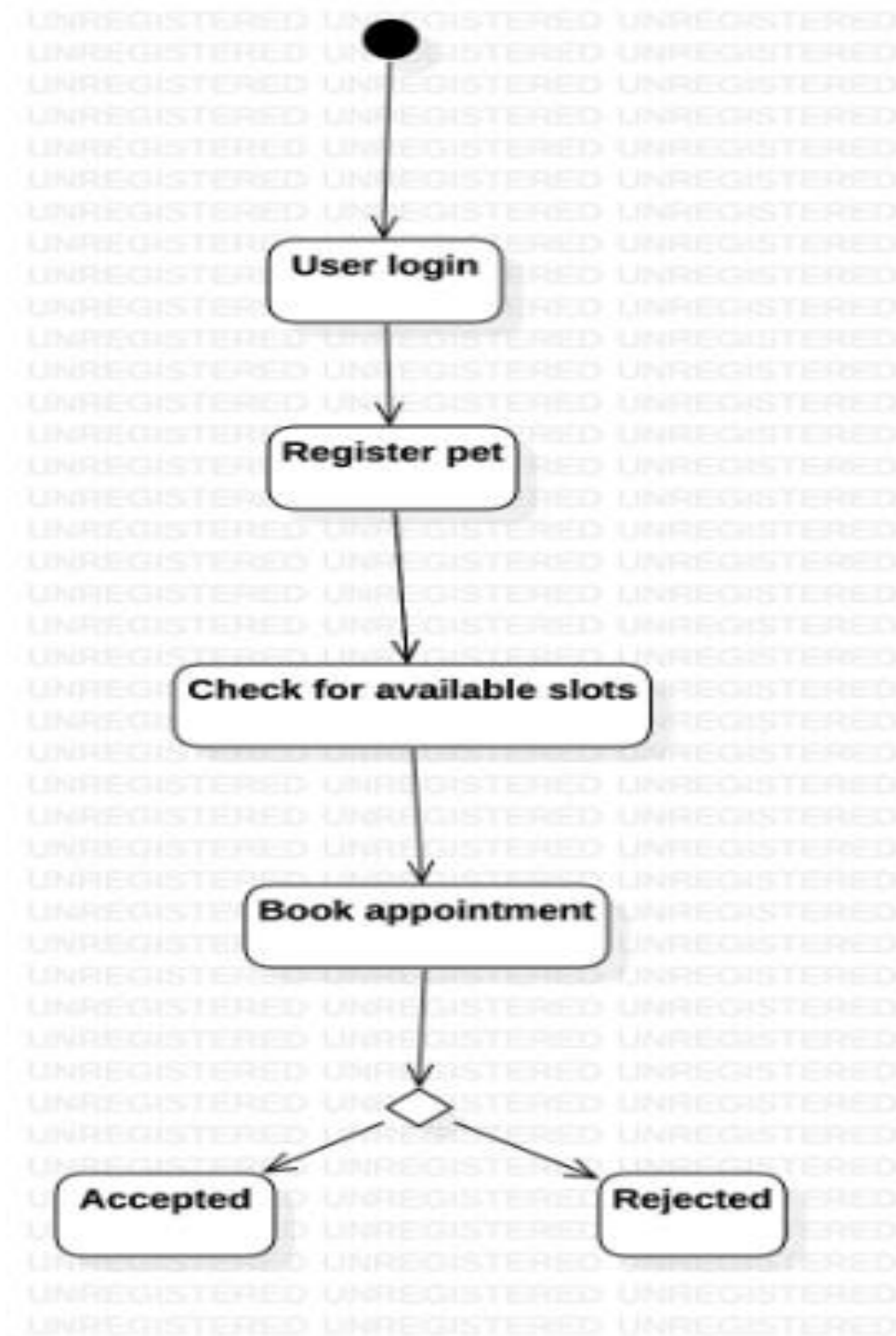
CLASS MODELING

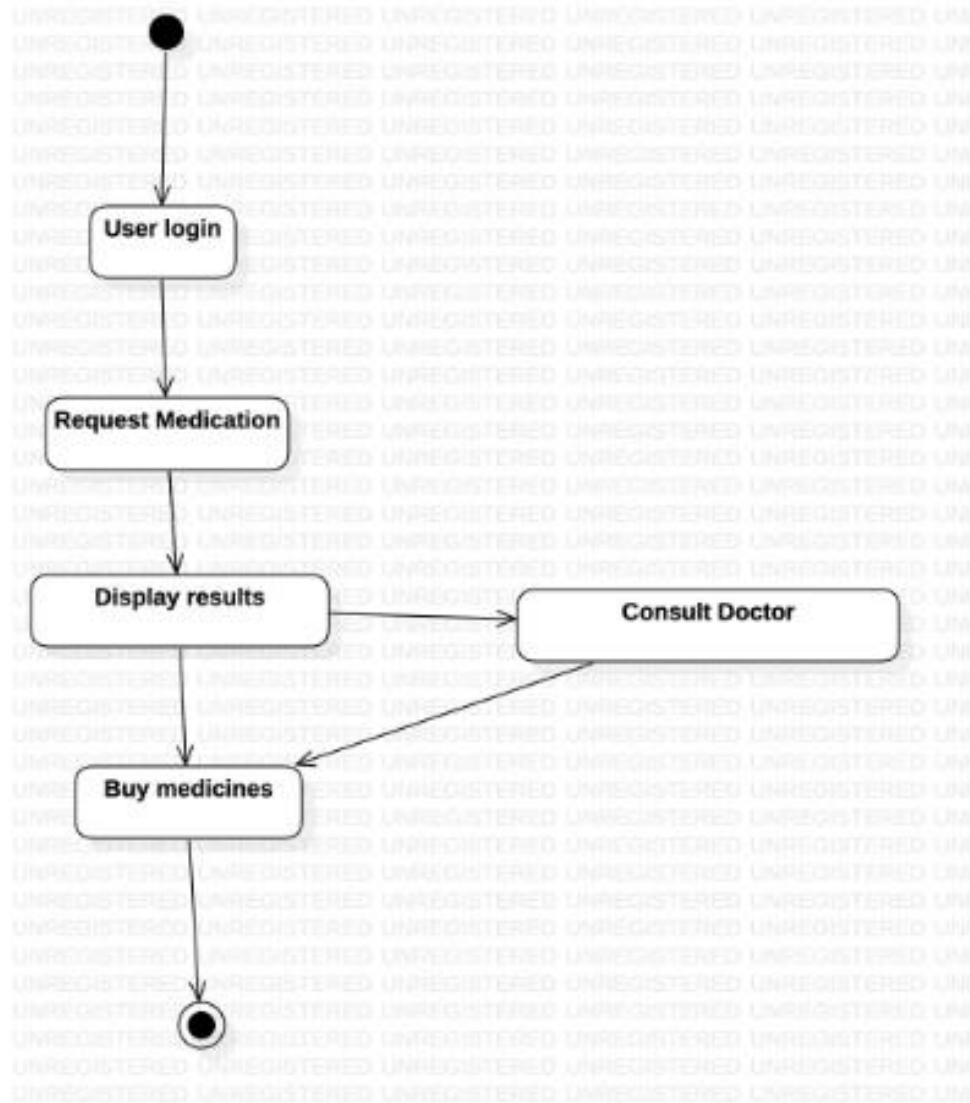


ACTIVITY MODELING



STATE MODELING





IMPLEMENTATION

1. **Demo application file:** This file is the entry point of a Spring Boot application. It is responsible for configuring and starting the Spring Boot application.

Code:

```
package io.vsn;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class UserRegistration1Application
{
    public static void main(String[] args)
    {
        SpringApplication.run(UserRegistration1Application.class, args);
    }
}
```

2. **User Controller:** The controller part of the MVC.

```
9. package io.vsn.controller;

import java.util.List;
import java.util.Random;

import javax.servlet.http.HttpSession;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.*;
import org.springframework.web.servlet.ModelAndView;
import io.vsn.model.User;
import io.vsn.repo.UserRepository;

@Controller
public class UserController
{
    @Autowired
    UserRepository urepo;

    @RequestMapping("/")
    public String home()
    {
        return "home";
    }

    @RequestMapping("/signup")
    public String getSignup()
    {
        return "signup";
    }

    @RequestMapping("/login")
```

```

    public String getLogin()
    {
        return "login";
    }

    @PostMapping("/addUser")
    public ModelAndView addUser(@RequestParam("user_email") String
user_email, User user)
    {
        ModelAndView mv=new ModelAndView("success");
        List<User> list=urepo.findByEMAIL(user_email);

        if(list.size() !=0)
        {
            mv.addObject("message", "Oops!  There is already a user registered
with the email provided.");
        }
        else
        {
            urepo.save(user);
            mv.addObject("message","User has been successfully registered.");
        }

        return mv;
    }
    @GetMapping("/dummy")
    public String dummy(HttpSession session)
    {
        if(session.getAttribute("username") != null)
        {
            return "dummy";
        }
        else
        {
            return "redirect:/login";
        }
    }

    @PostMapping("/dummy")
    public String goToAppointments()
    {
        return "appointments";
    }
    @GetMapping("/appointments")
    public String getAppointmentsPage(@SessionAttribute("username") String
username)
    {
        return "appointments";
    }
    @GetMapping("/bookAppointment")
    public ModelAndView bookAppointment() {
        ModelAndView mav = new ModelAndView("appointmentResponse");
        Random random = new Random();
        boolean isBooked = random.nextBoolean();
        mav.addObject("isBooked", isBooked);
        return mav;
    }
}

```

```

@RequestMapping("/schedules")
public String showSchedulesPage() {
    return "schedules";
}

@RequestMapping("/meds")
public String showMedsPage() {
    return "meds";
}

@RequestMapping("/dogs")
public String showDogsPage() {
    return "dogs";
}

@RequestMapping("/cats")
public String showCatsPage() {
    return "cats";
}

@PostMapping("/login")
public String login_user(@RequestParam("username") String
username, @RequestParam("password") String password,
                        HttpSession session, ModelMap modelMap)
{
    User auser=urepo.findByUsernamePassword(username, password);

    if(auser!=null)
    {
        String uname=auser.getUser_email();
        String upass=auser.getUser_pass();

        if(username.equalsIgnoreCase(uname) &&
password.equalsIgnoreCase(upass))
        {
            session.setAttribute("username",username);
            return "dummy";
        }
        else
        {
            modelMap.put("error", "Invalid Account");
            return "login";
        }
    }
    else
    {
        modelMap.put("error", "Invalid Account");
        return "login";
    }
}

@GetMapping(value = "/logout")
public String logout_user(HttpSession session)
{

```

```

        session.removeAttribute("username");
        session.invalidate();
        return "redirect:/login";
    }

}

```

3. User Repository : Used to push the data into the H2 database console.

```

package io.vsn.repo;

import java.util.List;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.data.jpa.repository.Query;
import io.vsn.model.User;

public interface UserRepository extends JpaRepository<User, Integer> {

    @Query("from User where user_email=?1")
    public List<User> findByEMAIL(String email);

    @Query("from User where user_email=?1 and user_pass=?2")
    public User findByUsernamePassword(String username, String password);

}

```

4. User.java: The model part of the MVC architecture.

```

package io.vsn.model;

import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;

@Entity
public class User
{
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Integer user_id;

    private String user_fname;
    private String user_lname;
    private String user_email;
    private String user_pass;
    private String user_mobile;
}

```

```

    public Integer getUser_id() {
        return user_id;
    }
    public void setUser_id(Integer user_id) {
        this.user_id = user_id;
    }
    public String getUser_fname() {
        return user_fname;
    }
    public void setUser_fname(String user_fname) {
        this.user_fname = user_fname;
    }
    public String getUser_lname() {
        return user_lname;
    }
    public void setUser_lname(String user_lname) {
        this.user_lname = user_lname;
    }
    public String getUser_email() {
        return user_email;
    }
    public void setUser_email(String user_email) {
        this.user_email = user_email;
    }
    public String getUser_pass() {
        return user_pass;
    }
    public void setUser_pass(String user_pass) {
        this.user_pass = user_pass;
    }
    public String getUser_mobile() {
        return user_mobile;
    }
    public void setUser_mobile(String user_mobile) {
        this.user_mobile = user_mobile;
    }
    @Override
    public String toString() {
        return "User [id=" + user_id + ", user_fname=" + user_fname + ",
user_lname=" + user_lname + ", user_email="
        + user_email + ", user_pass=" + user_pass + ", user_mobile=" +
user_mobile + "]\n";
    }
}

```

5. Façade design pattern(used in medications section):

```

package io.vsn.controller;
public class MedsFacade {

    public String getDiseaseInfo(String diseaseName) {
        String info = "";
        if (diseaseName.equalsIgnoreCase("Canine Distemper")) {
            info = "A highly contagious viral disease that attacks the respiratory
gastrointestinal, and nervous systems of the dogs.";
        }
    }
}

```

```

        } else if (diseaseName.equalsIgnoreCase("Feline Panleukopenia")) {
            info = "A highly contagious viral disease that attacks the immune
system, causing a decrease in white blood cells.";
        } else if (diseaseName.equalsIgnoreCase("Canine Parvovirus")) {
            info = "A highly contagious virus that attacks intestinal tract.";
        } else if (diseaseName.equalsIgnoreCase("Feline Leukemia Virus")) {
            info = "A contagious virus that attacks immune system, causing cancer
and other diseases.";
        } else if (diseaseName.equalsIgnoreCase("Heartworm Disease")) {
            info = "A parasitic disease transmitted by mosquitoes that affects
heart and lungs.";
        }
        return info;
    }

    public String getSymptomsInfo(String diseaseName) {
        String info = "";
        if (diseaseName.equalsIgnoreCase("Canine Distemper")) {
            info = "High fever, cough, runny nose and eyes, vomiting, diarrhea,
seizures, and paralysis.";
        } else if (diseaseName.equalsIgnoreCase("Feline Panleukopenia")) {
            info = "Loss of appetite, vomiting, diarrhea, fever, dehydration,
lethargy, and sudden death.";
        } else if (diseaseName.equalsIgnoreCase("Canine Parvovirus")) {
            info = "Vomiting, diarrhea (often bloody), fever, loss of appetite,
lethargy, and dehydration.";
        } else if (diseaseName.equalsIgnoreCase("Feline Leukemia Virus")) {
            info = "Loss of appetite, weight loss, vomiting, diarrhea, fever,
lethargy, and anemia.";
        } else if (diseaseName.equalsIgnoreCase("Heartworm Disease")) {
            info = "Coughing, fatigue, weight loss, difficulty breathing, and heart
failure.";
        }
        return info;
    }

    public String getMedicationInfo(String diseaseName) {
        String info = "";
        if (diseaseName.equalsIgnoreCase("Canine Distemper")) {
            info = "There is no cure for distemper, but supportive care such as
fluids, antibiotics, and anticonvulsants can help manage symptoms.";
        } else if (diseaseName.equalsIgnoreCase("Feline Panleukopenia")) {
            info = "There is no cure for feline panleukopenia, but supportive care
such as fluids, antibiotics, and antivirals can help manage symptoms.";
        } else if (diseaseName.equalsIgnoreCase("Canine Parvovirus")) {
            info = "There is no cure for canine parvovirus, but supportive care
such as fluids, antibiotics, and anti-nausea medication can help manage symptoms.
Vaccination is also available to prevent infection.";
        } else if (diseaseName.equalsIgnoreCase("Feline Leukemia Virus")) {
            info = "There is no cure for feline leukemia virus, but supportive care
such as fluids, antibiotics, and antivirals can help manage symptoms.";
        } else if (diseaseName.equalsIgnoreCase("Heartworm Disease")) {
            info = "There is no cure for heartworm disease, but preventative
medication can be given to dogs to protect them from infection.";
        }
        return info;
    }
}

```

6. Singleton design pattern (used in schedules section):

```
package io.vsn.controller;
public class Schedule {
    private static Schedule instance = null;

    private Schedule() {}

    public static Schedule getInstance() {
        if (instance == null) {
            instance = new Schedule();
        }
        return instance;
    }

    public String generateSchedule() {
        StringBuilder sb = new StringBuilder();
        sb.append("<h1>MY PET'S DAY</h1>");
        sb.append("<p>HEY I AM WALL-E AND I WILL BE TELLING YOU HOW TO TAKE CARE OF YOUR PET EVERYDAY.</p>");
        sb.append("<br><br><br>");
        sb.append("<p>PLEASE SELECT ONE OF THESE OPTIONS.</p>");
        sb.append("<br><br><br><br><br>");

        return sb.toString();
    }
}
```

7. Template design pattern(used in dogs section):

Super-class:

```
package io.vsn.controller;

public abstract class DogDailyRoutine {

    public void startDay() {
        wakeUp();
        goForAWalk();
        eatBreakfast();
    }

    public void endDay() {
        playWithOwner();
        sleep();
    }

    protected abstract void wakeUp();
    protected abstract void goForAWalk();
    protected abstract void eatBreakfast();
    protected abstract void playWithOwner();
    protected abstract void sleep();
}
```


Sub-class:

```
package io.vsn.controller;

public class LabradorRetrieverRoutine extends DogDailyRoutine {

    @Override
    protected void wakeUp() {
        System.out.println("The Labrador Retriever wakes up.");
    }

    @Override
    protected void goForAWalk() {
        System.out.println("The Labrador Retriever goes for a walk with its owner.");
    }

    @Override
    protected void eatBreakfast() {
        System.out.println("The Labrador Retriever eats high-quality dog food for breakfast.");
    }

    @Override
    protected void playWithOwner() {
        System.out.println("The Labrador Retriever plays fetch with a ball with its owner.");
    }

    @Override
    protected void sleep() {
        System.out.println("The Labrador Retriever goes to sleep for the night.");
    }

}
```

8. Proxy design pattern(used in cats section):

Cat.java class:

```
package io.vsn.controller;

public class Cat {
    public void sleep() {
        // Code for sleeping
        System.out.println("The cat will sleep at 9 p.m everyday.");
    }

    public void eat() {
        // Code for eating
        System.out.println("The cat is eating its food.");
    }

    public void play() {
        // Code for playing
        System.out.println("The cat is playing with a toy mouse.");
    }
}
```

Cat interface files:

```
package io.vsn.controller;

public interface CatProxy {
    void sleep();
    void eat();
    void play();
}
```

```
package io.vsn.controller;

import java.time.LocalDateTime;

public class CatProxyImpl implements CatProxy {
    private Cat cat;

    public CatProxyImpl(Cat cat) {
        this.cat = cat;
    }

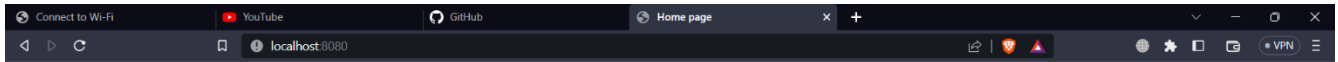
    @Override
    public void sleep() {
        cat.sleep();
    }

    @Override
    public void eat() {
        // Control access to eat method based on time of day
        LocalDateTime now = LocalDateTime.now();
        if (now.isAfter(LocalTime.of(8, 0)) && now.isBefore(LocalTime.of(20, 0))) {
            cat.eat();
        } else {
            System.out.println("It's not mealtime yet!");
        }
    }

    @Override
    public void play() {
        cat.play();
    }
}
```

RESULTS SCREENSHOTS

1. WELCOME PAGE



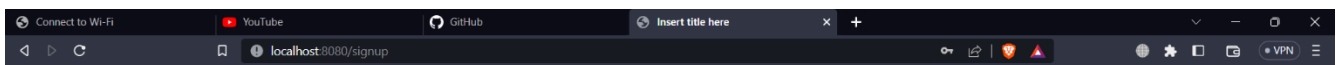
Welcome

Sign Up

Login



2.SIGN-UP PAGE:



Sign Up

Please fill in this form to create an account.

First Name

Enter First Name

Last Name

Enter Last Name

Email

Enter Email

Password

Enter Password

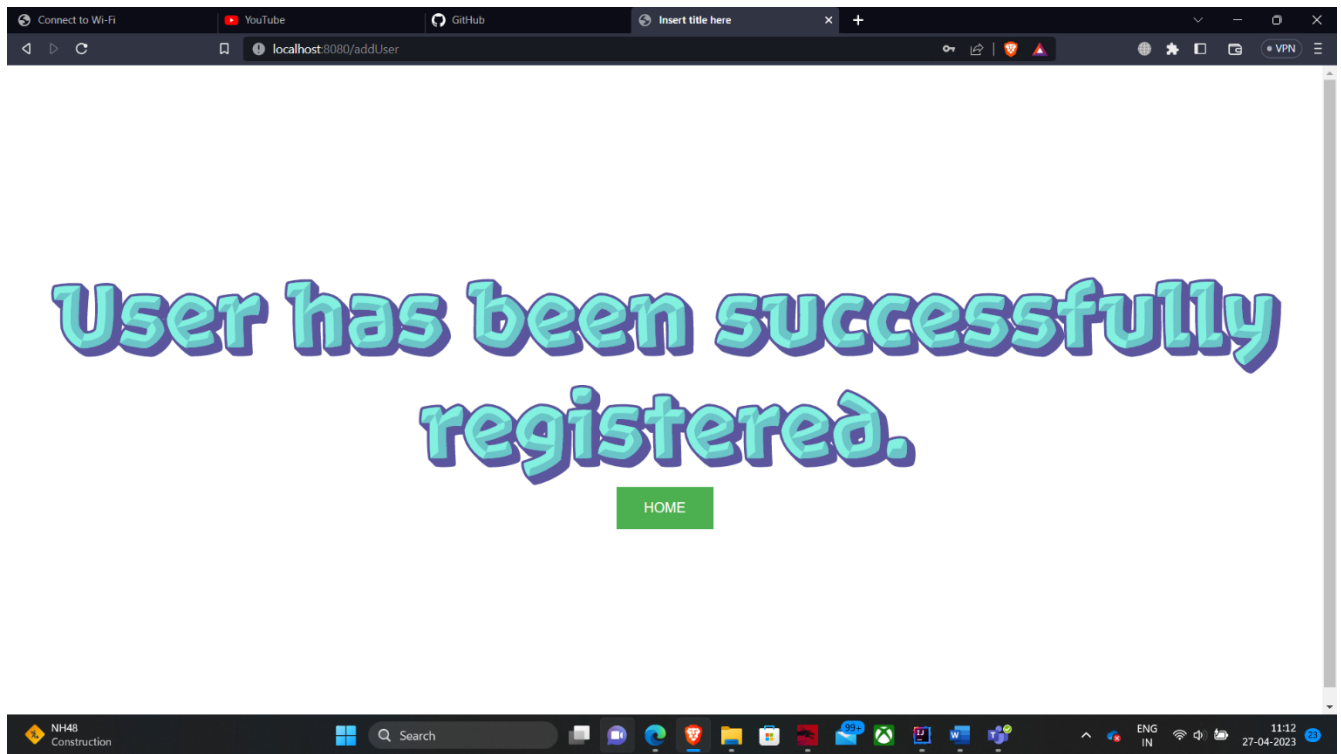
Mobile

Enter Mobile

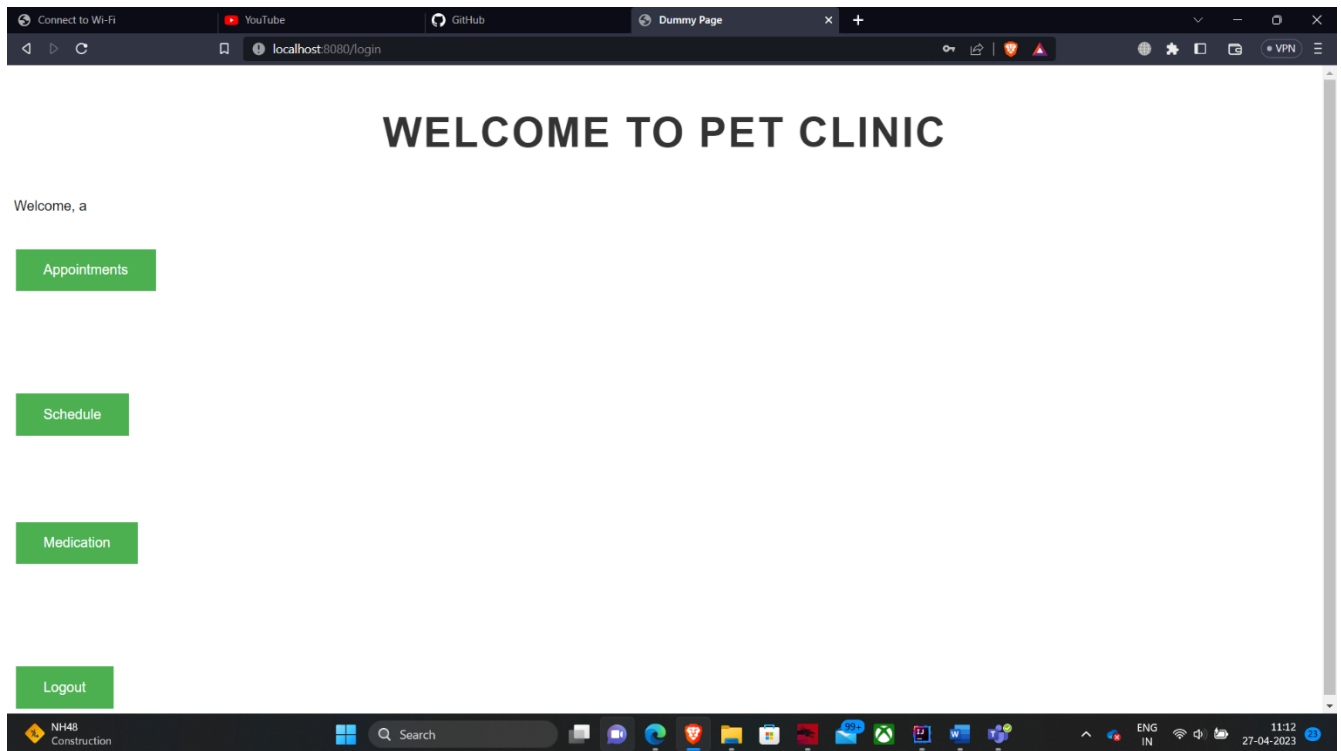
Cancel

Sign Up





3.Main webpage after logging in:



4. Appointments section:

Connect to Wi-Fi

YouTube

GitHub

Appointments

localhost:8080/dummy

VPN

Appointments

Doctor Name	Location	Action
Dr. John Smith	New York, NY	Schedule Appointment
Dr. Jane Doe	Los Angeles, CA	Schedule Appointment
Dr. Michael Lee	San Francisco, CA	Schedule Appointment
Dr. Lisa Patel	Chicago, IL	Schedule Appointment
Dr. Robert Johnson	Miami, FL	Schedule Appointment

Back

NH48 Construction

Search

ENG IN

11:12 27-04-2023

5.Schedules section: (DOGS AND CATS SECTION)

Connect to Wi-Fi

YouTube

GitHub

MY PET-Day Schedule

localhost:8080/schedules

VPN

MY PET'S DAY

HEY Y I AM WALL-E AND I WILL BE TELLING YOU HOW TO TAKE CARE OF YOUR PET EVERYDAY.

PLEASE SELECT ONE OF THESE OPTIONS.

DOGS

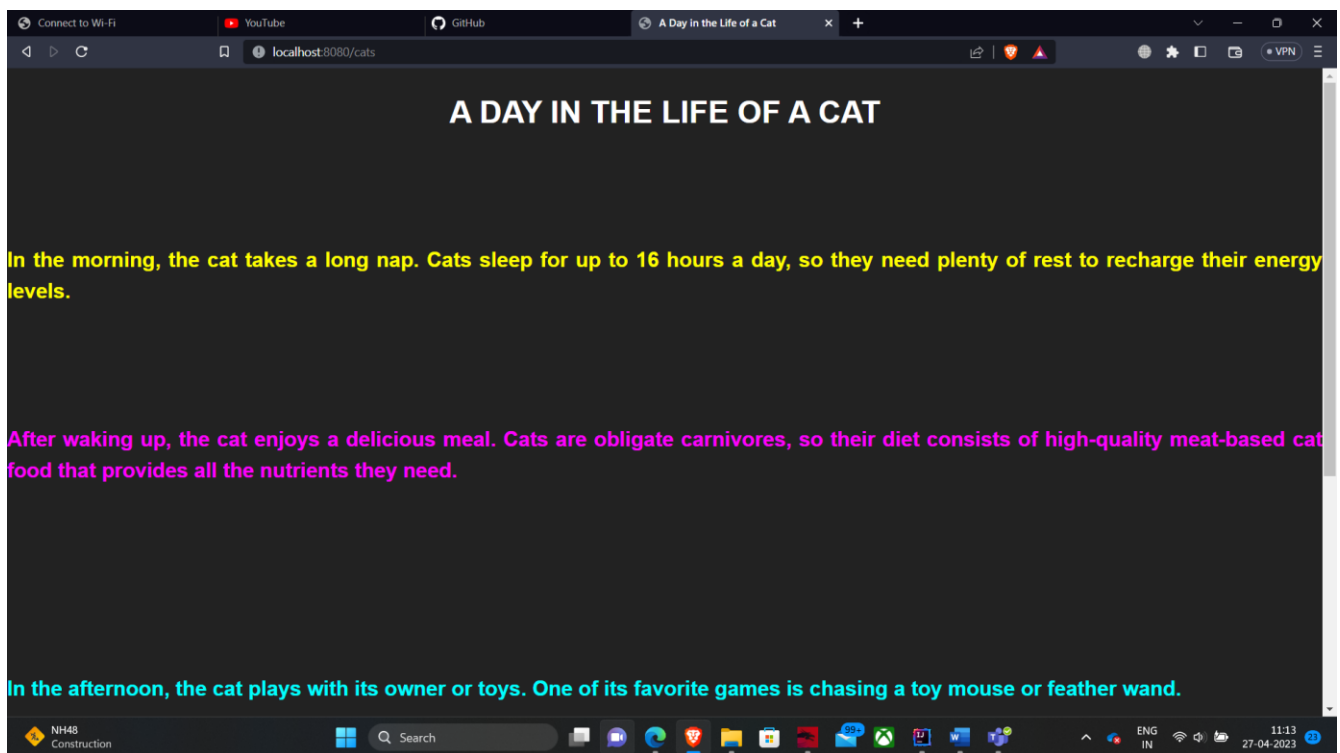
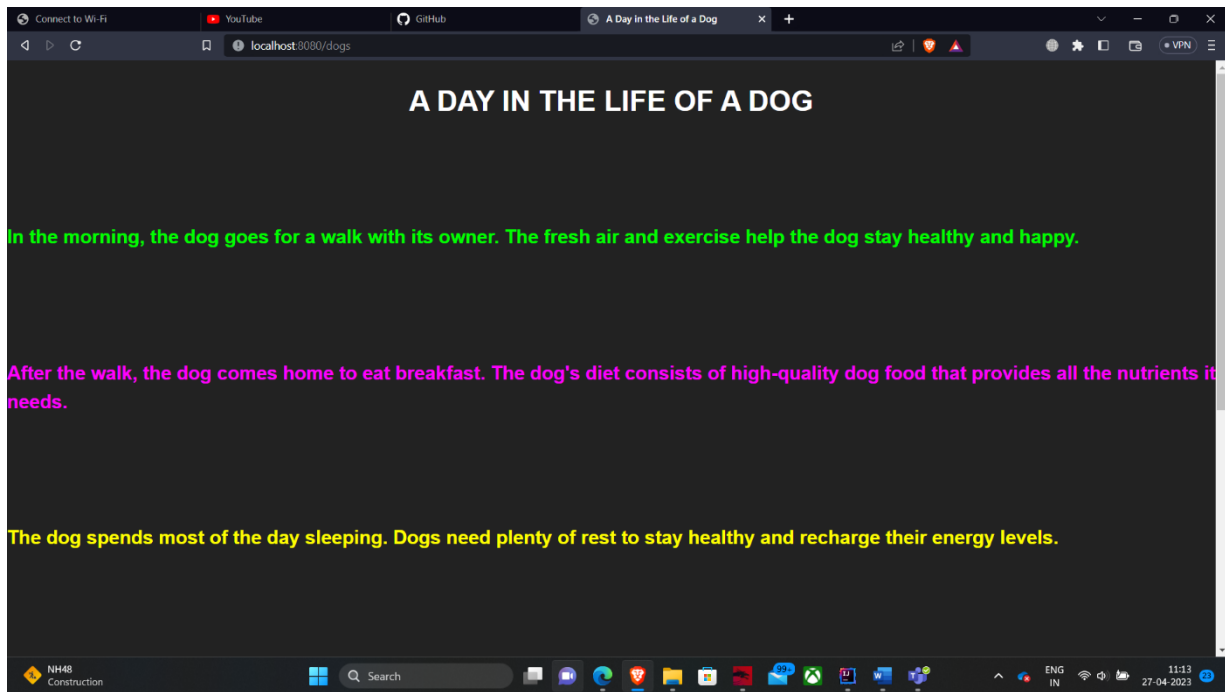
CATS

NH48 Construction

Search

ENG IN

11:12 27-04-2023



6.Medications section:

Connect to Wi-Fi

YouTube

GitHub

Sample Page

localhost:8080/meds

A highly contagious viral disease that attacks the respiratory gastrointestinal, and nervous systems of the dogs.

High fever, cough, runny nose and eyes, vomiting, diarrhea, seizures, and paralysis.

There is no cure for distemper, but supportive care such as fluids, antibiotics, and anticonvulsants can help manage symptoms.

KNOW MORE...

A highly contagious viral disease that attacks the immune system, causing a decrease in white blood cells.

Loss of appetite, vomiting, diarrhea, fever, dehydration, lethargy, and sudden death.

There is no cure for feline panleukopenia, but supportive care such as fluids, antibiotics, and antivirals can help manage symptoms.

KNOW MORE...

A highly contagious virus that attacks intestinal tract.

Vomiting, diarrhea (often bloody), fever, loss of appetite, lethargy, and dehydration.

There is no cure for canine parvovirus, but supportive care such as fluids, antibiotics, and anti-nausea medication can help manage symptoms. Vaccination is also available to prevent infection.

KNOW MORE...

NH48
Construction

Search

ENG
IN

11:12
27-04-2023