**“AniWorld, The World of Animals”**

**ADissertation**

**Submitted in partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering in Computer Science andEngineering**

**Submitted To**



**RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL (M.P.)**

**Submitted By:**

**Jyoti Patankar Manasvi Sadhwani**

**Enrollment No- 0821CS181032Enrollment no -0821CS181040**

**Under The Supervision Of:**

**Mr. Pratyush Sharma**

**(HOD, CSE Dept.)**

**Mr. Atish Sharma**

**(Sr. Asst. Professor/Asst. Professor, CSE Dept )**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING MALWA INSTITUTE OF TECHNOLOGY, INDORE**

**MALWA INSTITUTE OF TECHNOLOGY, INDORE**

**Department of Computer Science and Engineering**

### CERTIFICATE

This is to certify that the work embodies in this dissertation entitled **“AniWorld, The world of Animals ”** being submitted by **Miss Manasvi Sadhwani RollNo.:0821CS181040 and Miss Jyoti PatankarRoll No.:0821CS181032** for partial fulfillment of the requirement for the award of “**Bachelor of Technology in Computer Science and Engineering**” to “**Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal(M.P.)**” during the academic year 2020-21 is a record of bonafide piece of work, carried out by him under my supervision and guidance in the “**Department of Computer Science and Engineering** ”, Malwa Institute of Technology, Indore (M.P.).

**APPROVED & SUPERVISED BY:**

**Mr.Pratyush Sharma Mr. Atish Sharma**

(HOD, CSE Dept.) (Asst.Prof. CSE Dept.)

**Dr. Rajeev G Vishwakarma**

(Principal)

**MALWA INSTITUTE OF TECHNOLOGY, INDORE**

### Department of Computer Science and Engineering

### CERTIFICATE OF APPROVAL

The Dissertation entitled **“AniWorld, The World of animal”** being submitted by “**Manasvi Sadhwani” (Roll No.: 0821CS181040)** and **“Jyoti Patankar” (Roll No.: 0821CS181032)**has been examined by us and is hereby approved for the award of degree **“Bachelor of Technology in Computer Science and Engineering**”, for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed but approve the dissertation only for the purpose for which it has been submitted.

**(Internal Examiner) (External Examiner)**

**Date: Date:**

**MALWA INSTITUTE OF TECHNOLOGY, INDORE**

### Department of Computer Science and Engineering

### DECLARATION

### WeManasvi sadhwaniandJyoti Patankar, student ofBachelor of Technology in Computer Science and Engineering, in Malwa Institute of Technology, Indore (M.P.), here by declare that the work presented in this dissertation entitled AniWorld , The world of Animalis the outcome of our own work, is bonafide and correct to the best of my knowledge and this work has been carried out taking care of Engineering Ethics. The work presented does not infringe any patented work and has not been submitted to any other university or anywhere else for the award of any degree or any professional diploma.

**Student Name: Jyoti Patankar**

#### Enrollment No.: 0821CS181032

#### Student Name: Manasvi sadhwani

#### Enrollment No.: 0821CS181040

#### Date:

# **Acknowledgement**

Foremost, I would like to express my sincere gratitude to my advisor **Mr. Atish Sharma**  for the continuous support of our study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis.

I am ever grateful to **Dr. Rajeev G Vishwakarma,** Principal, Malwa Institute of Technology, Indorefor providing opportunity to explore this wonderful venture. And also I would like to give regards and remembers another source of inspiration **Mr. Pratyush Sharma** (HOD,CSE Dept.)

Last but not the least; I would like to thank my family: my parents, for giving birth to me at the first place and supporting me spiritually throughout my life.

This thesis would not have been possible without collaboration with my colleagues, and the support of my friends and family.

**Jyoti Patankar**

**Manasvi Sadhwani**

**Abstract**

This document is a review report on the research conducted and the project made in the field of computer engineering to develop a Web application forAnimal services. The report proposed the results and solutions on the limited implementation of the various techniques that are introduced in the project. Whereas the implementation of the project give the real world idea of how the system works and what changes can be done in order to improve the utility of the overall system.

Furthermore, the paper states the overview of the observations made by the authors in order to help further optimization in the mentioned field to achieve the utility at a better efficiency for a safer road.

**List of Figures:**

|  |  |  |
| --- | --- | --- |
| **FigureNo.** | **Name** | **PageNo.** |
| **1** | Use Case diagram |  |
| **2** | Activity diagram |  |
| **3** | Class diagram |  |
| **4** | Sequence Diagram |  |
| **5** | DFD |  |
| **6** | ER diagram |  |
|  |  |  |
|  |  |  |

**ABTRACT**

**Chapter 1Introduction**

* 1. Background……………………………....…….…..………………....5
  2. Need………………………..…………….……..….………..…….…..6
  3. Problem Descriptions……………………….…..…….….………..…...7
  4. Solution………………………………….…………...………………..12
  5. Description of the project……………………….……………..………13

**Chapter 2Literature Survey**

2.1 Literature Survey………………………..….………….…………… 15

2.2 S/W Requirements……………………….…..……..………………16

2.2.2H/W Requirements…………………………………………………

2.2.3 Functional Requirement…………………..…..…….…….………… 17

2.2.4 Non Functional Requirement……………..….…..……...……………18

**Chapter 3 Analysis & Design**

3.1 DFD…………………………………………..…..…………………22

3.2 ER Diagram…………….……………........……….………………… 24

3.3 USE CASE Diagram……………….....………....……………………35

3.4 SEQUENCE Diagram…………...…...…..………………………….39

3.5 ACTIVITY Diagram………….…….………….……………………. 44

3.7 CLASS Diagram……………..…….………..….……………………45

**Chapter 4 Process model used**

4.1 Agile model……………………………………………… 47

**Chapter 5 TechonologiesUsed**

5.1 Django…………….………………..……….….…………………………51

5.2 VScode……………………..……….……..……………………….52

5.3 Java script …………………….….…….…………………………..53

5.4 PostgreSQL…………………………………………………………..

5.5 PG admin……………………………………………………………

**Chapter6 Testing**

6.1 Testing objective………………………….……….………………..…..55

6.2 Testing scope………………….......……….…………..……….………55

6.3 Testing principal…………………………….……………….………….56

6.4 Test case………………………………...………………………………57

6.5 Type of testing…………………….……………………………………59

**Chapter7SnapShots**

**7.1 \_\_\_\_\_\_\_\_\_\_\_\_**

**7.2** \_\_\_\_\_\_\_\_\_\_\_\_

**CONCLUSION**……………………..…………………….……………………………67

**FUTURE ENHANCEMENT AND LIMITATIONS**…………….………………….68

**BIBLIOGRAPHY**……..…………..……………..…………………………………….. 70

**Chapter-1**

**Introduction**

India alone is home to an estimated 30 million street dogs and has the world's highest rate of human deaths from rabies. The biggest reason for the increasing population of stray dogs is open garbage. Municipal authorities deal with thousands of tonnes of garbage a day but are able to process less than half of it. Home composting and source segregation is yet to gain ground in India. That makes garbage a source of ready food for dogs. The only source of food for street dogs is garbage and their shelter is street.

According to the surveys, 70% of people love to have pets at their home and out of that only 20% are willing to have a street dog as their pet. These situations are rapidly increasing the lack of of food and shelter for street animals. This issue can be resolved if people start adopting pets instead of buying one. 30% of people don't tend to have better at their home because a pet cannot be left alone at home for days while the owner is out of station. Some people faces the issues of vaccination of pet and regular health care.

All these issues can be resolved if people start to adopt a pet instead of buying one and if ifa animal services are available to easily.

**1.1 Background**

This project is an initiative tu to save Street animals ants how to make the services easily available for the owners.

Ill-treatment:

10% of the street animals die every year because of lack of food, lack of shelter, and cruel fights with other animals as well as humans. The harsh treatment and cruelty towards street animals is commonly observed everywhere. The animals are being beaten up for no reason, animals are hit by the vehicles and people tend to lower down the humanity for animals.

Time Issues:

For the people who like to have a pet at home r are unable to have one because of the time issues. The major issue faced by the owners is that when they there place for few or more days then who would take care of their pets.

Healthcare:

Proper healthcare services are not provided for the the animals in every area. A lot of owners don't vaccinate their pets because of unavailability of of healthcare organisations and veterinary doctors why some of them delay their pets regular check up and vaccination by some days because of time conflicts.

Unadopted pets:

Some people adopt Street animals ants some people provide regular foods to the street animals in their areas. But it is seen that the cruelty against Street animals is rapidly growing in the hearts of human beings. These animals need to be saved from ill treatment and along with food they require some regular care for living.

**1.2 Need**

The street animals need to be saved from ill treatments, weather conditions, food shortages and a lot more. We need a proper system to provide all the needed services for these animals. A bunch of people should come together with their services which can be paid as well as free of cost to rescue and save the animals.

**1.3Problem Description**

After understanding the current scenarios and pass surveys we can conclude that the animals around us are continuously being ill treated and their rescue is must. Some government as well as private organisations are taking an initiative to provide food shelter and save the street animals but wind leaves more efforts from government as well as non governmentalorganisations, health care organisations, and common people.

The agenda is to provide all the possible services through an online mode so that owner as well as these organisations don't face any trouble.

.

**1.4Solution**

Providing all the services is a difficult task because neither some organisations and common people want to take any initiative and want to make efforts for the ones who are unspoken. The challenging part is to provide all the services through one platform and the complete cooperation and coordination of all the people and organisation.

The solution is is to provide these services which can have be paid as well as free of cost.

**1.5Description of Project**

Our website, AniWorld is is an initiative to solve this problems. In our project, there are different departments for providing different services for the pets, their owners, as well as street animals.

Adoption Feature:

Through this adoption feature, user can adopt the street animal through our website from anywhere in our country. The details of the animal would be provided. The user can visit that option page or can directly check the animals for adoption through the home page. The details of each adoptable pet is available on the adoption page as well as us specific page is designed for each animal. User can easily avail this service. For adopting the user needs to contact the adoption department and can contact the admin for any query.

Caretaker feature:

This teacher is for the owner of the pets. The issue faced by the owners is that how and where their pet would live if the owner is out of town for some days. To resolve this issue we have a feature called caretaker through which owner can contact a caretaker by visiting the caretaker page. Information cards of each category are available on home pages too. The owner can read all the necessary information about the caretaker and then contact the caretakers through the form provided at the bottom of the page. Once the owner is satisfied and comfortable in leaving their pets with the caretaker, owner can do the same. it is completely dependent on the caretaker and owner to schedule some meetings to check if the pet would be comfortable with the cat and the owner my visit the caretakers place to check if the care taker is good enough to take care of their pet or not. The caretakers are available free of cost as well as paid.

For any query owner as well as the caretaker can contact the admin.

NGO:

Since the street animals are continuously being ill treated and beaten up up by by humans and in some areas the animals die because of lack of of food and shelter. If the user is interested in helping those sweet animals, one is free to do so. But sometimes one person or a family or a group of friends are unable to help and rescue all the treat animals of that area. so we have provided a service of contacting the NGOs for animal welfare.

A person are free inform the NGO or bite take some initiative and work with the NGO for animal welfare through our website.

Veterinary:

The veterinary services are provided by the healthcare teams and the veterinary doctors for the street animals as well as the pets and their owners.

A user can contact any veterinary doctor as per their choice for veterinary services like regular check up, vaccination and treatment.

Contact:

All the interested people and organisations can contact the admin to be the part of our project, AniWorld.

NGOs, caretakers and veterinary doctor's information and the details would be checked legally and can be the part of AniWorld after the verification.

Sign up and sign in services are provided for the user. In case of any query or dispute, admin can be contacted.

**Chapter-2**

**Literature Survey**

**2.1Literature Survey**

This survey is done to comprehend the need and prerequisite of the general population, and to do as such, we went through different sites and applications and looked for the fundamental data. Based on these data, we made an audit that helped us get new thoughts and make different arrangements for our task. We reached the decision that there is a need of such application and felt that there is a decent extent of progress in this field too.

**2.2Requirements**

Requirements are documented in a formal artifact called a Requirements Specification (RS), which will become official only after validation. A RS can contain both written and graphical (models) information if necessary. Example: [Software requirements specification](https://en.wikipedia.org/wiki/Software_requirements_specification) (SRS).

* **Functional Requirements**
* **Non-Functional Requirements**
* **Software Requirements**

**2.2.1 Software requirements**

* Django
* VScode
* PostgreSQL
* PGadmin

**2.2.3Functional Requirement**

The functional Requirements of the system, project specifies the actions or the task that the system fulfilled. In other words that purpose for which the system is designed in the first place is defined by functional requirements.

The Functional requirements of the project are:-

* Implement python to achieve the desired result.

**2.2.4Non-Functional Requirements**

The Non - Functional Requirements of a system or project specifies the quality actions or the quality characteristics of the system. These are those requirements that are not the main purpose as to why the project is created , however fulfilling these requirements enhances the properties of the system.

The Non-Functional Requirements of the project are:-

* Privacy and security.
* Easy to use.
* Fast response to any activity.
* Attractive and User friendly system and setup.
* Platform independent.
* Reliability.

**Chapter-3**

**Analysis & Design**

**3.1 Use-case Diagrams**

In software and systems engineering, a use case is a list of actions or event steps, typically defining the interactions between a role (known in the Unified Modeling Language as an *actor*) and a system, to achieve a goal. The actor can be a human, an external system, or time. In systems engineering, use cases are used at a higher level than within software engineering, often representing missions or stakeholder goals. Another way to look at it is a use case describes a way in which a real-world actor interacts with the system. In a system use case you include high-level implementation decisions. System use cases can be written in both an informal manner and a formal manner.

A use case is a description of how end-users will use a software code. It describes a task or a series of tasks that users will accomplish using the software and includes the responses of the software to user actions.

Use cases have been used extensively over the past few decades. The advantages of Use cases includes:

* The list of goal names provides the shortest summary of what the system will offer
* It gives an overview of the roles of each and every component in the system. It will help us in defining the role of users, administrators etc.
* It helps us in extensively defining the user’s need and exploring it as to how it will work.
* It provides solutions and answers to many questions that might pop up if we start a project unplanned.

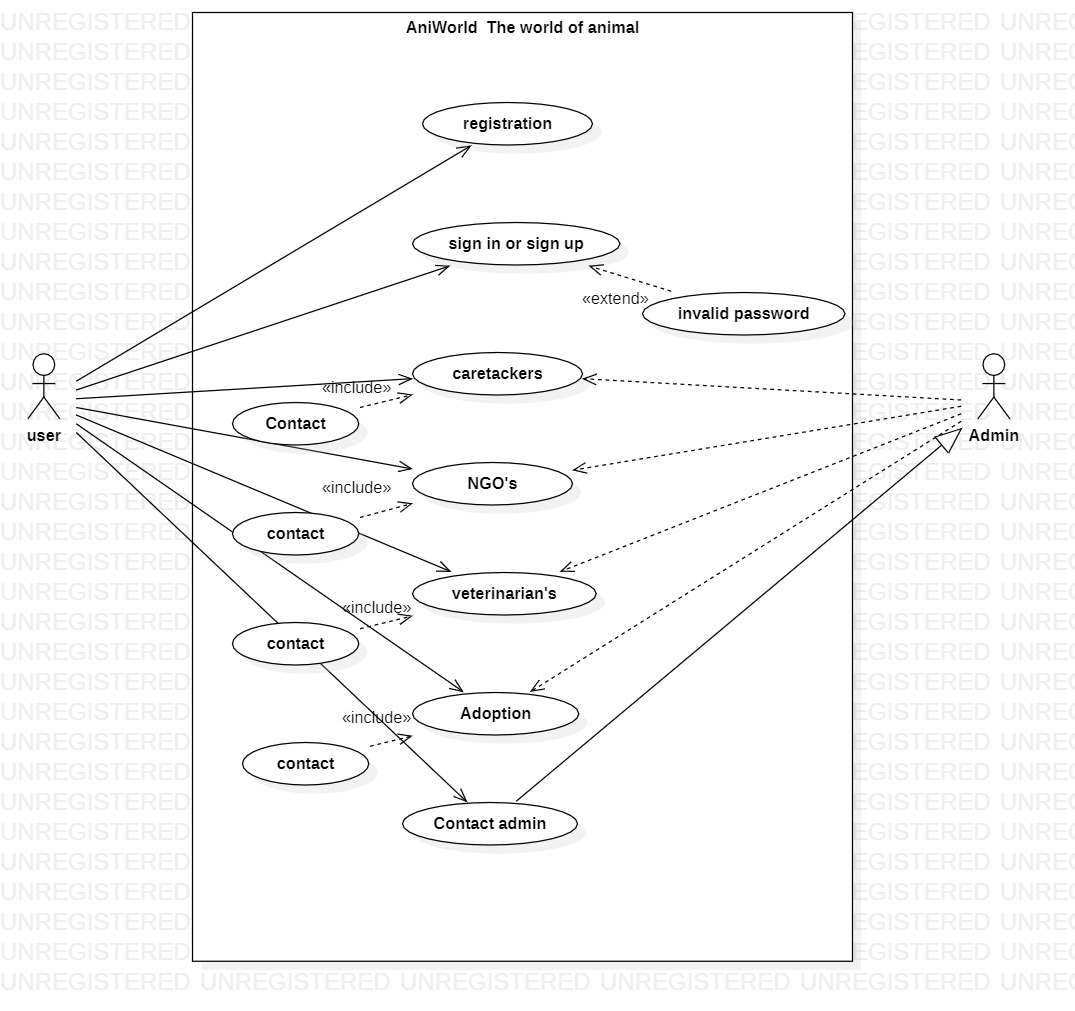


Figure 3.1 USE CASE diagram

**3.2Activity Diagrams**

Activity diagram are graphical representation of workflows of stepwise activities and action with support for choice iteration and concurrency.

We use Activity Diagrams to illustrate the flow of control in a system and refer to the steps involved in the execution of a use case. We model sequential and concurrent activities using activity diagrams. So, we basically depict workflows visually using an activity diagram. An activity diagram focuses on condition of flow and the sequence in which it happens. We describe or depict what causes a particular event using an activity diagram.

UML models basically three types of diagrams, namely, structure diagrams, interaction diagrams, and behavior diagrams. An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system.

An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. We can depict both sequential processing and concurrent processing of activities using an activity diagram. They are used in business and process modeling where their primary use is to depict the dynamic aspects of a system.

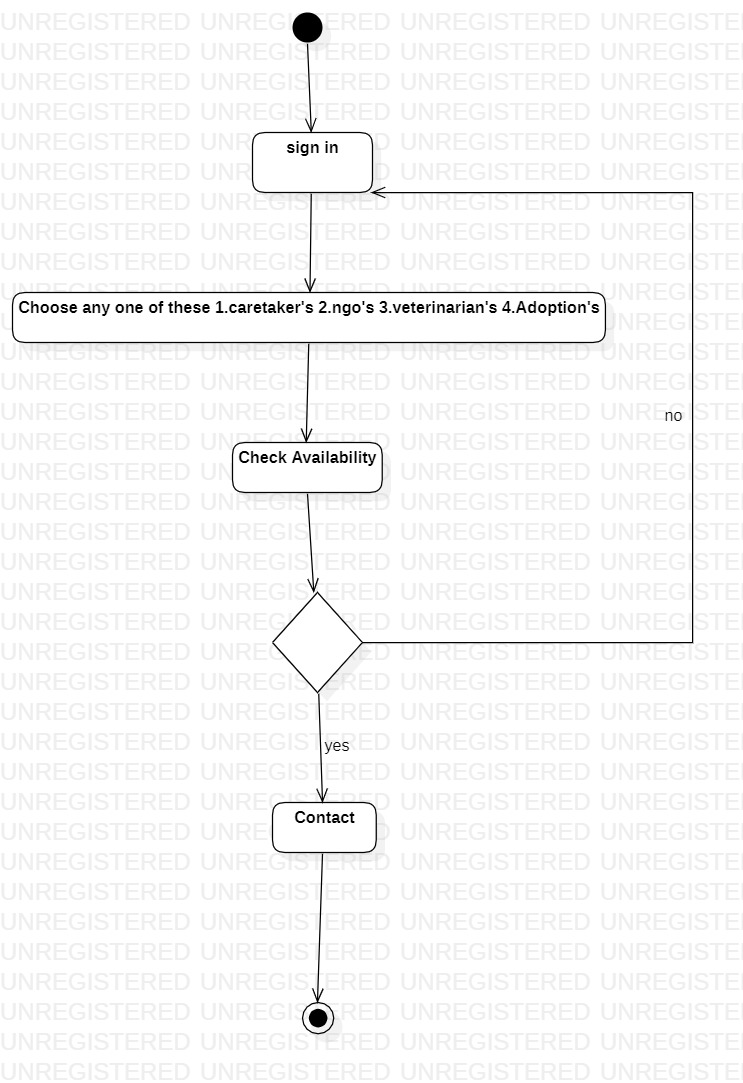


Figure 3.2 Activity Diagram

**3.3Class Diagrams**

In the [Unified Modeling Language](http://en.wikipedia.org/wiki/Unified_Modeling_Language) (UML), a class diagram is a type of static structure diagram that describes the structure of a system by showing the system's [classes](http://en.wikipedia.org/wiki/Class_(computer_science)), their attributes, and the [relationships](http://en.wikipedia.org/wiki/Object-oriented_programming) between the classes.

Class diagrams are the main building blocks of every object oriented methods. The class diagram can be used to show the classes, relationships, interface, association, and collaboration. UML is standardized in class diagrams. Since classes are the building block of an application that is based on OOPs, so as the class diagram has appropriate structure to represent the classes, inheritance, relationships, and everything that OOPs have in its context. It describes various kinds of objects and the static relationship in between them.

The main purpose to use class diagrams are:

* This is the only UML which can appropriately depict various aspects of OOPs concept.
* Proper design and analysis of application can be faster and efficient.
* It is the base for development and component diagram.

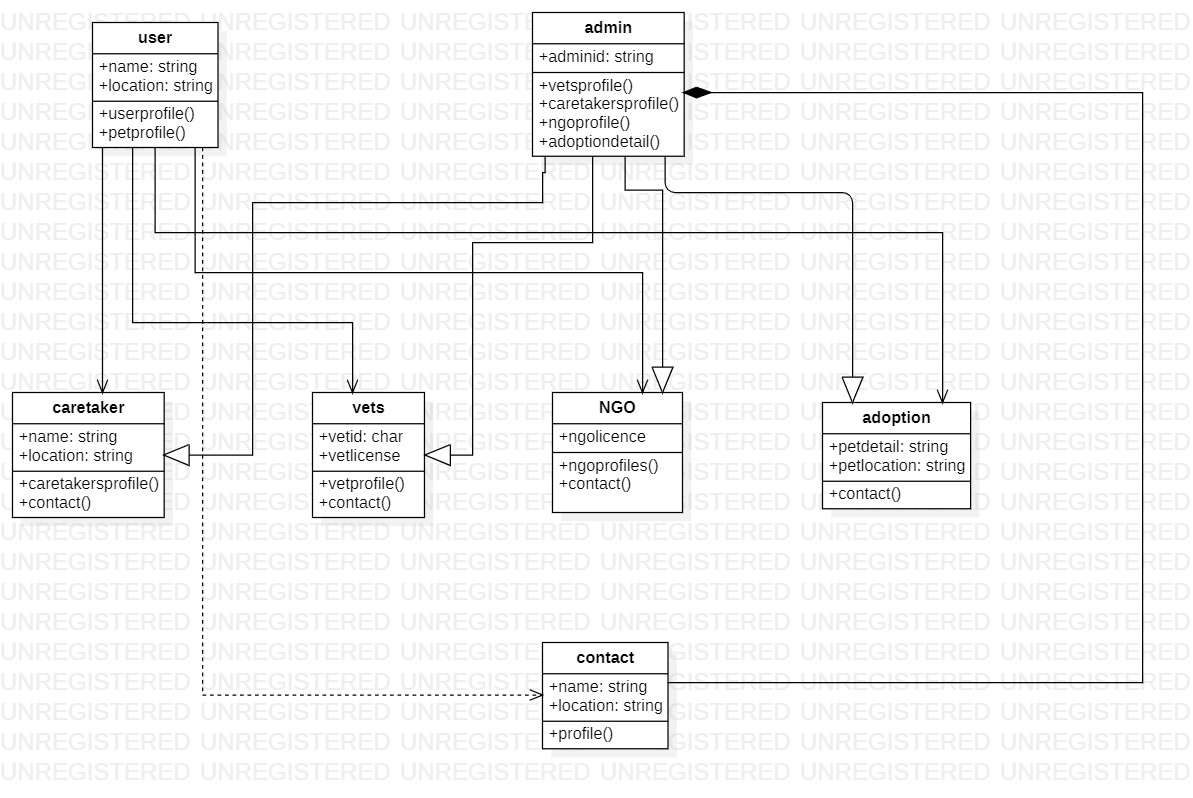


figure 3.3 Class diagram

**3.4 Sequence diagram**

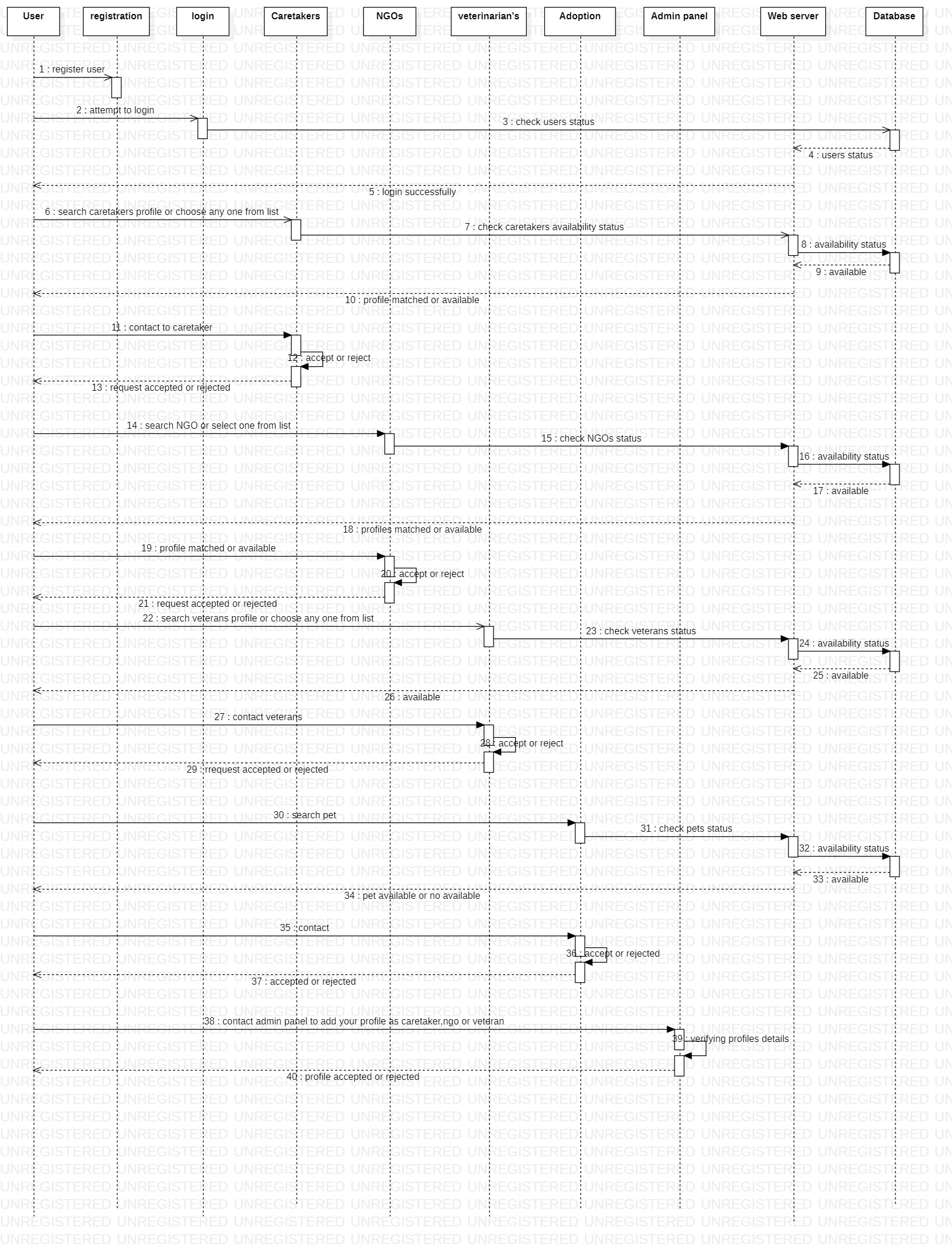
[**UML**](https://en.wikipedia.org/wiki/Unified_Modeling_Language) Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

Sequence Diagrams captures:

* the interaction that takes place in a collaboration that either realizes a use case or an operation (instance diagrams or generic diagrams)
* high-level interactions between user of the system and the system, between the system and other systems, or between subsystems .

### Purpose of Sequence Diagram

* Model high-level interaction between active objects in a system
* Model the interaction between object instances within a collaboration that realizes a use case
* Model the interaction between objects within a collaboration that realizes an operation
* Either model generic interactions (showing all possible paths through the interaction) or specific instances of a interaction (showing just one path through the interaction)



Sequence diagram

**3.5 Data Flow Diagram**

DFD graphically representing the functions, or processes, which capture, manipulate, store, and distribute data between a system and its environment and between components of a system. The visual representation makes it a good communication tool between User and System designer. Structure of DFD allows starting from a broad overview and expand it to a hierarchy of detailed diagrams. DFD has often been used due to the following reasons:

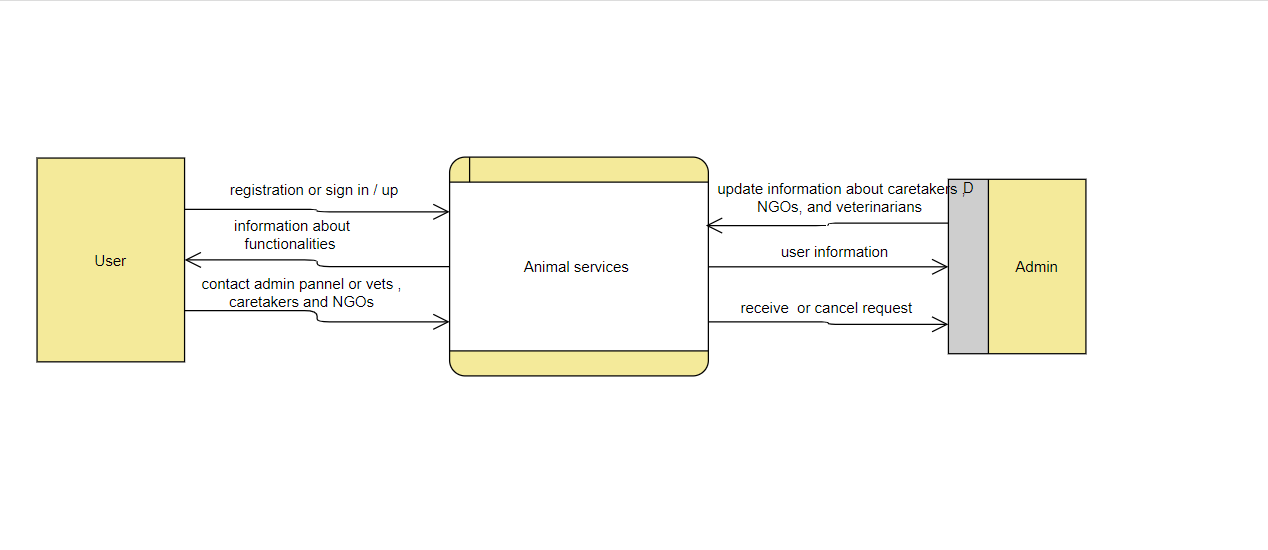
* Logical information flow of the system
* Determination of physical system construction requirements
* Simplicity of notation
* Establishment of manual and automated systems requirements

## **Logical vs Physical Data Flow Diagrams**

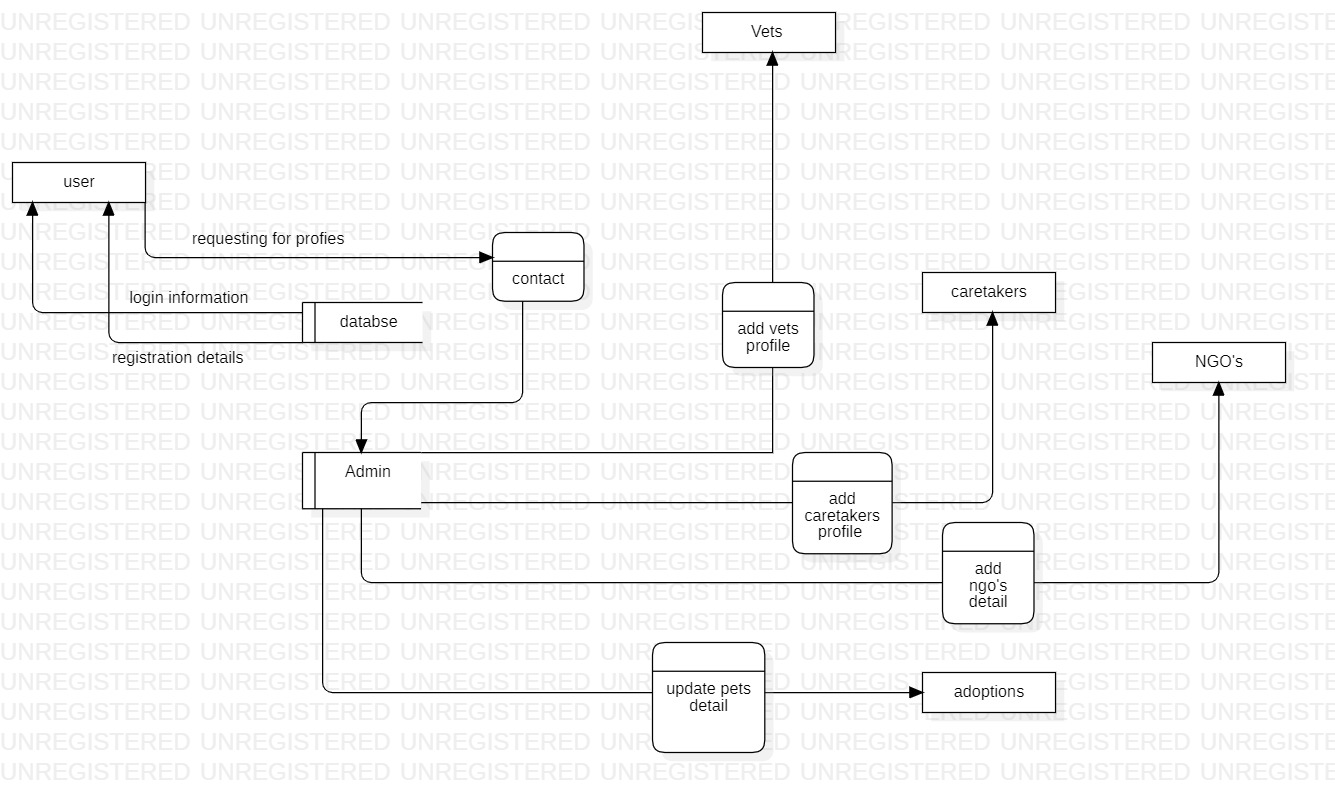
Data flow diagrams are categorized as either logical or physical. A logical data flow diagram focuses on the business and how the business operates. It is not concerned with how the system will be constructed. We can ignore implementation specifics such as, computer configuration, data storage technology, communication or message passing methods by focusing on the functions performed by the system, such as, data collection, data to information transformation and information reporting.

A physical data flow diagram shows how the system will be implemented, including the hardware, software, files, and people in the system. It is developed such that the processes described in the logical data flow diagrams are implemented correctly to achieve the goal of the business.

Level 0



Level 1:



**3.6 ER diagram**

Database is absolutely an integral part of software systems. To fully utilize ER Diagram in database engineering guarantees you to produce high-quality database design to use in database creation, management, and maintenance. An ER model also provides a means for communication.

Entity Relationship Diagram, also known as ERD, ER Diagram or ER model, is a type of structural diagram for use in database design. An ERD contains different symbols and connectors that visualize two important information: **The major entities within the system scope**, and the **inter-relationships among these entities**.

And that's why it's called "Entity" "Relationship" diagram (ERD)!

When we talk about entities in ERD, very often we are referring to business objects such as people/roles (e.g. Student), tangible business objects (e.g. Product), intangible business objects (e.g. Log), etc. "Relationship" is about how these entities relate to each other within the system.

In a typical ER design, you can find symbols such as rounded rectangles and connectors (with different styles of their ends) that depict the entities, their attributes, and inter-relationships.

**Chapter-4**

**Software Method Used**

**4.1Methodology**

**Agile model**

Agile model is chosen for its development.

AniWorld mainly consists of five modules to be developed and tested namely:

* Adoption Module
* Caretaker Module
* NGO Module
* Vets Module
* Contact Admin Module

Following are the Agile Manifesto principles-

* **Individuals and Interactions –** In Agile development, self-organization and motivation are important, as are interactions like co-location and pair programming.
* **Working Software –** Demo working software is considered the best means of communication with the customers to understand their requirements, instead of just depending in documentation.
* **Customer Collaboration –** As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous customers interaction is very important to get proper product requirements.
* **Responding to Change –** Agile development is focused on quick responses to change and continuous development.

**Characteristics of Agile Model**

1. Iterative development builds the project’s ultimate deliverable in small increments, repeating basic steps over and over.
2. Modularity is considered one of the key elements of a good process. Modularity is the element that allows the components to break down and that broken component is called activities.
3. Agile software development is considered an upgraded version of the traditional approach with time constraints add on. Impossible deadlines are not attempted for rapid delivery.
4. High Transparency – This is one of the main characteristics of Agile and Scrum approach. It allows various stakeholders to contribute and be completely involved throughout the protect.
5. Better Visibility – Agile and Scrum development methodology are well- known for actively involving users/clients throughout the life of the product’s development process. This leads to better visibility for key stakeholders involved in the project to ensure that everyone’s expectations are managed effectively.

**Advantages**

1. It is very realistic approach to software development.
2. Functionality can be developed rapidly and demonstrated.
3. Resource requirements are minimum
4. Suitable for fixed or changing requirements.
5. Enables concurrent development and delivery within an overall planned context.

**Disadvantages**

1. An overall plan, an agile PM practice is must without which it will not work.
2. Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.
3. Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction.
4. Transfer of technology to new team members may be quite challenging due to lack of documentation.
5. There is a very high individual dependency, since there is minimum documentation generated.

**Tasks**

1. Communication: helps to understand the objective.
2. Planning: required as many people(software teams) work on the same project but different function at same time.
3. Modelling: involves business modelling, data modelling, and process modelling.
4. Construction: this involves the reuse software components and automatic code
5. Deployment: integration of all the increments.

**Chapter – 5**

**Technology Used**

**5.1 Django**

Django is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support.

.

**5.2 VS code**

Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE.

**5.3JavaScript**

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

**5.4 PostgreSQL**

It is a highly stable database management system, backed by more than 20 years of community development which has contributed to its high levels of resilience, integrity, and correctness. **PostgreSQL** is **used as** the primary data store or data warehouse for many web, mobile, geospatial, and analytics applications.

**5.5 PG admin**

**pgAdmin** is the most popular and feature rich Open Source administration and development platform for PostgreSQL, the most advanced Open Source database in the world. **pgAdmin** may be used on Linux, Unix, macOS and Windows to manage PostgreSQL and EDB Advanced Server 9.5 and above.

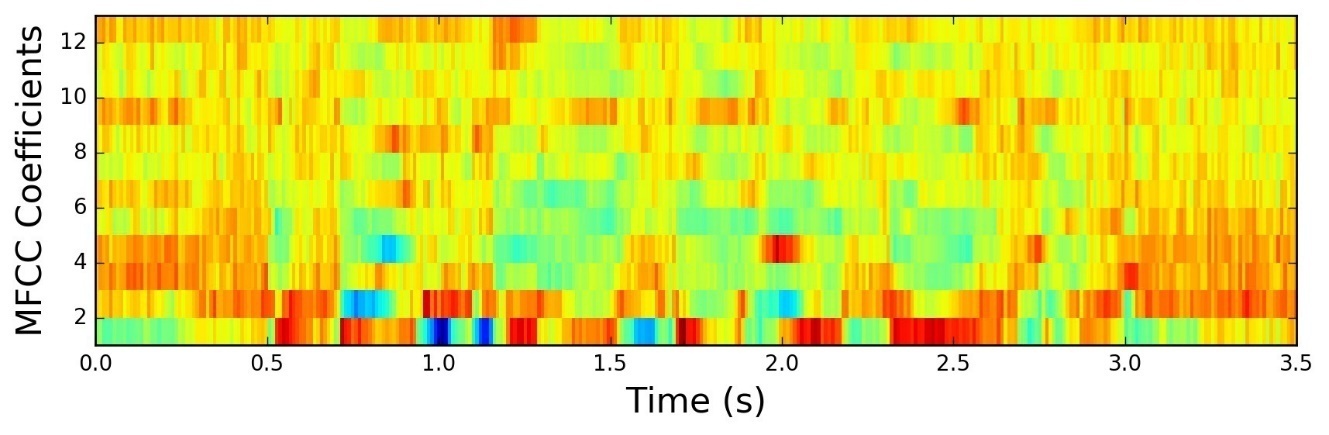
**Chapter – 6**

**Testing**

**6.1 Implementation**

For implementation, feature selection is the consideration and constantly emphasize topic. The music is basically in a different format of mp3, wav and wma. Different terms and characterized methods, learning technique, algorithms and attributes apply to work out on genre classification based on feature distinguish. The features that we select should be able to distinguish and characterize the different form of an audio signal considering the amplitude, temporal and spectral and include Mel-frequency cepstral coefficient (MFCCs), spectrogram, spectrum power, zero crossing rate, spectral centroid, spectral roll off and chroma frequencies. Mel-Frequency Cepstral Coefficients (MFCCs) is a spectral power and musical characteristics representation where an audio is transformed into a time and frequency domain to provide an output of an audio signal in a comprehension way.

To some extent, Mel-frequency cepstral has been able to demonstrate an effective calculation of pitch and frequency content in the audio signals by automatic recognition of speech and Modeling them.



There have been 2 stages of implementation, particularly one targeted on FVs optimization and therefore the different one was dedicated to evaluating musical style classification effectiveness. many algorithms were used within the pre-study part, particularly k Nearest Neighbors (k NNs) formula, Support Vector Machine (SVM), each algorithm with- and while not the co-training mechanism, further as Random Forests. The results achieved for musical style classification victimization these algorithms around at intervals constant vary of accuracy. Henceforth, the best effectiveness was obtained whereas victimization the Support Vector Machine formula with a co-training mechanism, consequently, the results for SVM (co-training), can solely be given.

**4.2 Testing**

Testing in software product is the process of evaluating and detecting the actual functionality of a system or application ensuring it perform what it is built for. The testing also intent to discover whether a software application met its requirements or not. The identification of software problems and defects is detected during this stage. The main reason for software testing is to ensure that a system application is productive and meet the quality standard by fulfilling the design modules and prototype.

**4.2.1 White Box Testing**

White Box Testing in also known as clear box, open box, glass box and structural testing in software application testing. White Box Testing only handle with the source code of internal structure. Test case is design by using skills of programming level of programmers and system internal outlook and frame. Generally, at unit level, testing is done. (Rajkumar, 2020) Unlike black box testing, tester can easily observe the code in this testing approach. White box testing performs its techniques by understanding the source code and creating test cases for execution. The tester must possess good knowledge of code and each process of testing to process application**.**

**4.2.2 Black Box Testing**

Black Box Testing is also known in other terminology as Behavioral testing. In this approach, the testing make assumption of setting specific input values or object and the system has to provide the respective results as per the functional design of system. If the output result match with actual resulting output then the tester approves and confirm system as 'ok' for solving problematic issues. The tester is not aware of the internal structure source code in this approach and they don't have any interference with the code. The end uses and tester their test on system application.

6.2.3 **Test Case**

Chart

Description automatically generated with low confidence

**Conclusion**

Conclusion It completely meets the objectives and requirements of the system. The framework has achieved an unfaltering state where all the bugs have been disposed of. The framework cognizant clients who are familiar with the framework and comprehend it's focal points and the fact that it takes care of the issue of stressing out for individuals having fatigue-related issues to inform them about the drowsiness level while driving.

**Bibliography**

**References:**

[**www.google.com**](http://www.google.com)

**web.learncodeonline.in**

**www.W3school.com**

**django documentation**