Zoo Maintenance System

Table of contents:

1.0 Introduction

A zoo maintenance system effectively manages and handles the functionalities of the zoo.
Like monitoring the animal details and locating a particular animal inside the zoo etc.

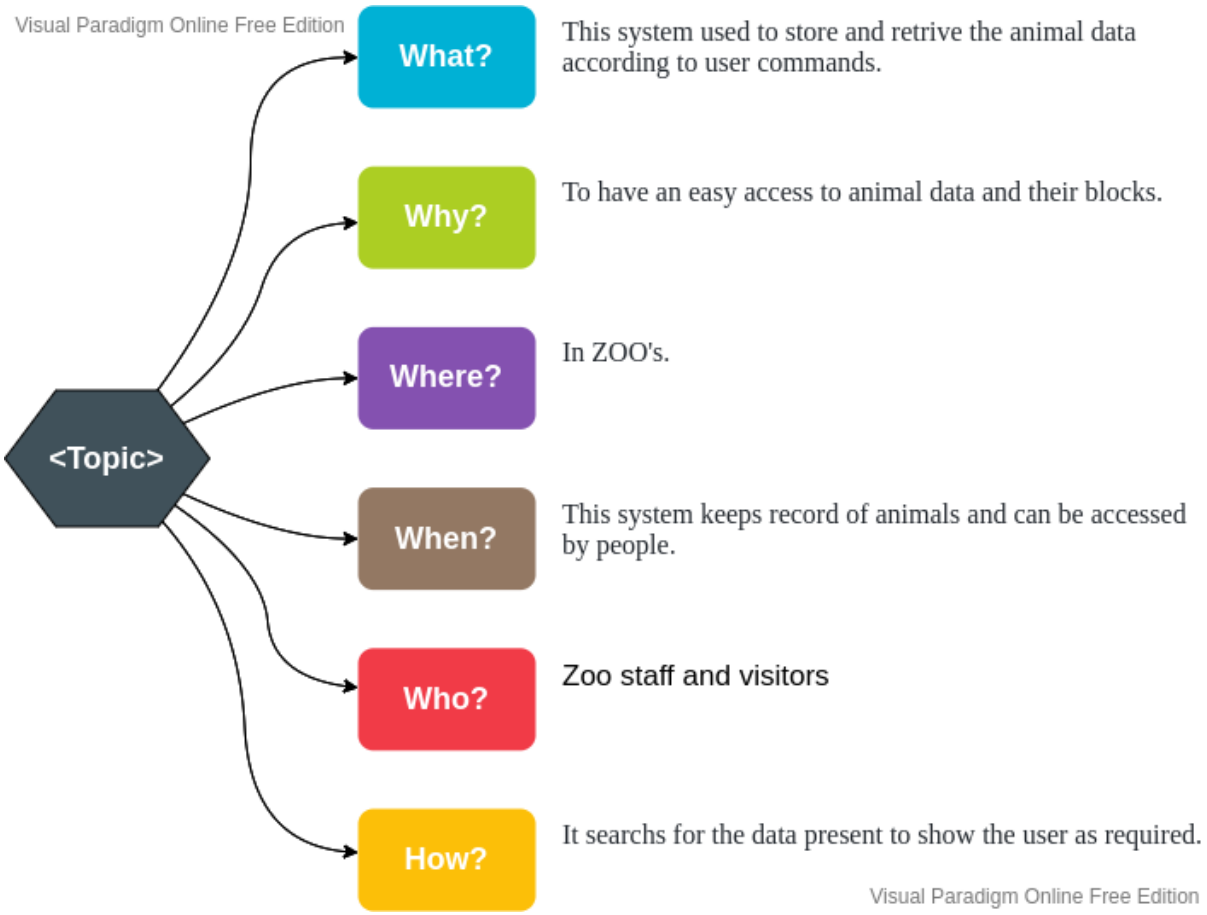
1.2 Abstract

Zoological parks provide an opportunity to open a whole new world of curiosity and interest, and sensitise visitors regarding the value and need for conservation of wildlife. A zoo maintenance system effectively manages and handles the functionalities of the zoo.

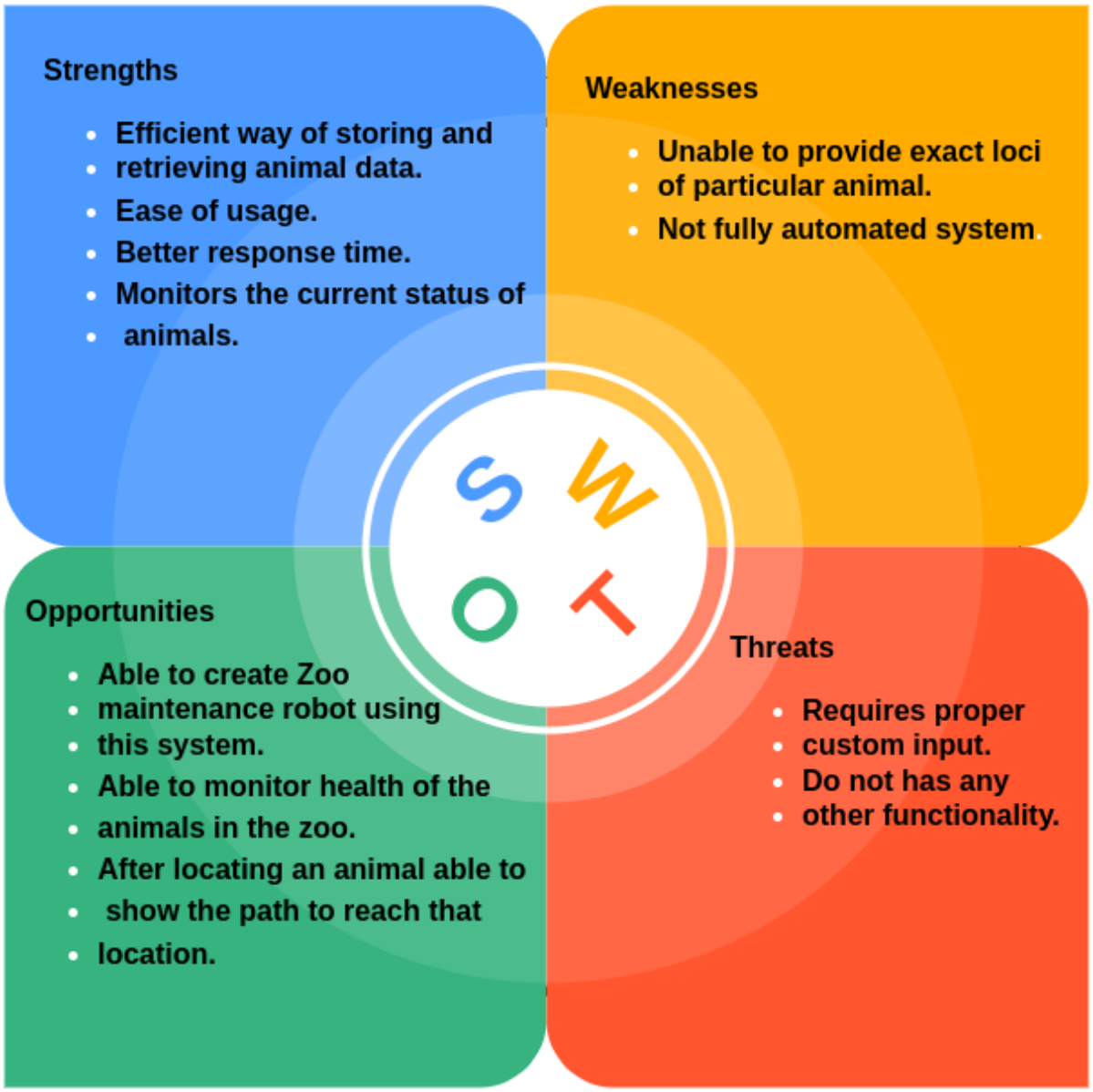
1.3 Description

The main problem and motivation is basically lack of information about the various animals present in the zoo. This project is based on the various information related to the animals which are present in the zoo. Such as when we visit any zoo then we are not aware of which animal is placed at which place or we don't know the loci of that particular animal. To know this information, the system looks for data present in the zoo, but with this information system we can easily search the zoo by filling the name of the animal, such as Lion, Tiger, etc. then by running this user command one can easily find the block in which the animal is present.

2. 5 W's and 1 H


Visual Paradigm Online Free Edition
<Topic>
What?
This system used to store and retrive the animal data according to user commands.
Why?
To have an easy access to animal data and their blocks.
Where?
In ZOO's.
When?
This system keeps record of animals and can be accessed by people.
Who?
Zoo staff and visitors
How?
It searchs for the data present to show the user as required.
Visual Paradigm Online Free Edition

3. SWOT Analysis



4.Requirements

4.1 High level requirements

Test ID	Description	Category
HLR_1	Update Animals details	Functional
HLR_2	View Animal/Animals details	Functional
HLR_3	Check for Animals presence	Functional
HLR_4	Search for animal	Functional

4.2 Low level requirements

Test ID	Description	HLR ID	Category
LLR_1	Add Animal details	HLR_1	Technical
LLR_2	Edit Animal details	HLR_1	Technical
LLR_3	Delet Animal details	HLR_1	Technical
LLR_4	View Animal details Block wise	HLR_2	Technical
LLR_5	Display Animal present in the zoo or not	HLR_3	Technical

LLR_6	Search through Animal data details present in the system	HLR_4	Technical

5. Test Plan and Output:

5.1 High level test plan:

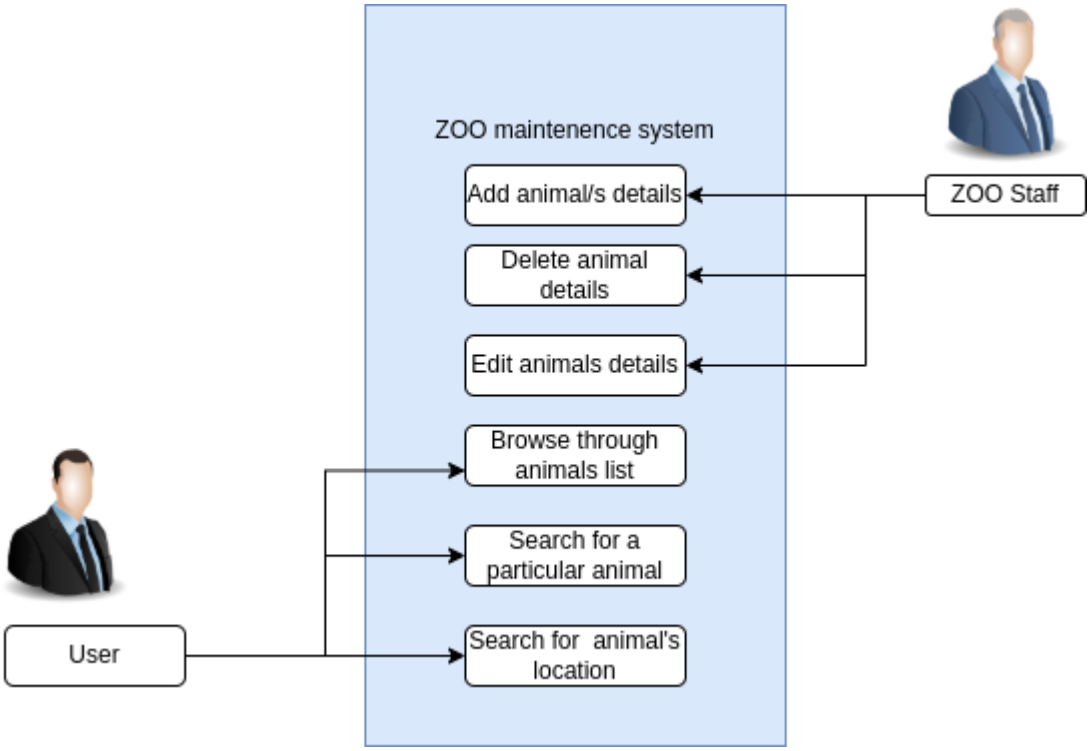
Test ID	Description	Input	Expected output	Actual Output
HLTP_01	Updating animal's details	Animal's details	Animal's details in a list	Updated list with animal's details
HLTP_02	View animal's details	Open animals list	Animal's details in a list display	Updated list with animal's details displays.
HLTP_03	Check for Animals presence	Animal's name	Animal's details in a list if animal exists in zoo Or not found.	Animal's details in a list if animal exists in zoo Or not found.
HLTP_04	Search for animal	User types animals name	Found or Not found	Found or Not found

5.2 LOW LEVEL TEST PLAN

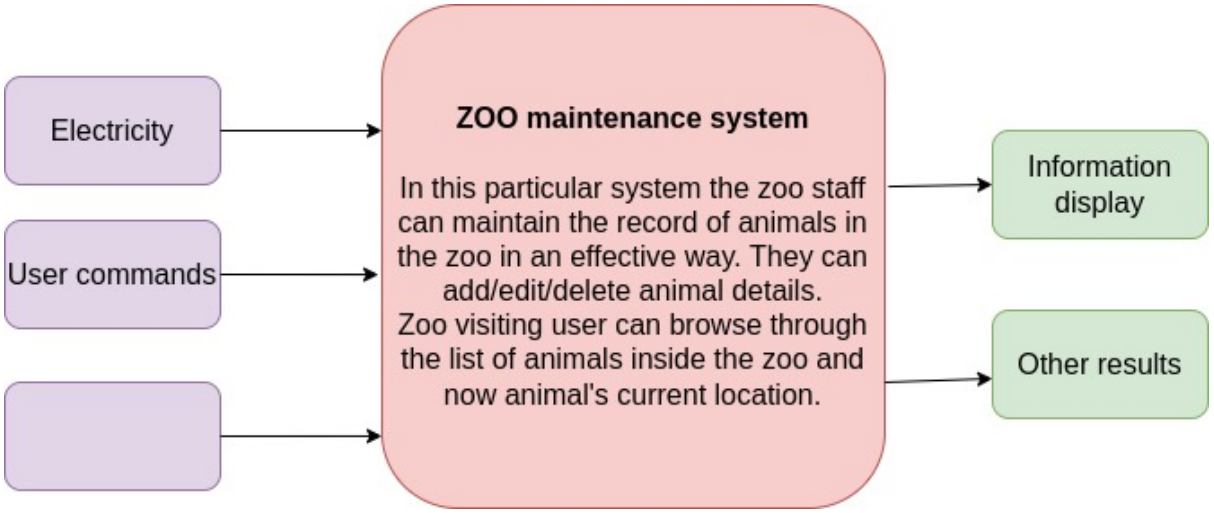
Test ID	Description	HLTP ID	Input	Expected output	Actual Output	Status
LLTP_01	Add animal's details	HLTP_01	Animal's details	Animal's details in a list	Updated list with animal's details	✔️
LLTP_02	Edit animal's details	HLTP_01	Animal's details to be edited	Updated list with animal's details displays.	Updated list with animal's details displays.	✔️
LLTP_03	Delete Animal details	HLTP_01	Animal's details to be deleted	Updated list with animal's details displays.	Updated list with animal's details displays.	✔️
LLTP_04	Search for animal	HLTP_04	User types animals name	Search for an animal and display Found or Not found	Found or Not found	✔️
LLTP_05	Viewing animal list	HLTP_02	Option for display feature	Animal list displays	Animal list displays	✔️

LLTP_06	Check animals presence	HLTP_03	Option for animal check feature and type animal's name	Search for an animal and display Found or Not found	Found or Not found	✔

Use case diagram

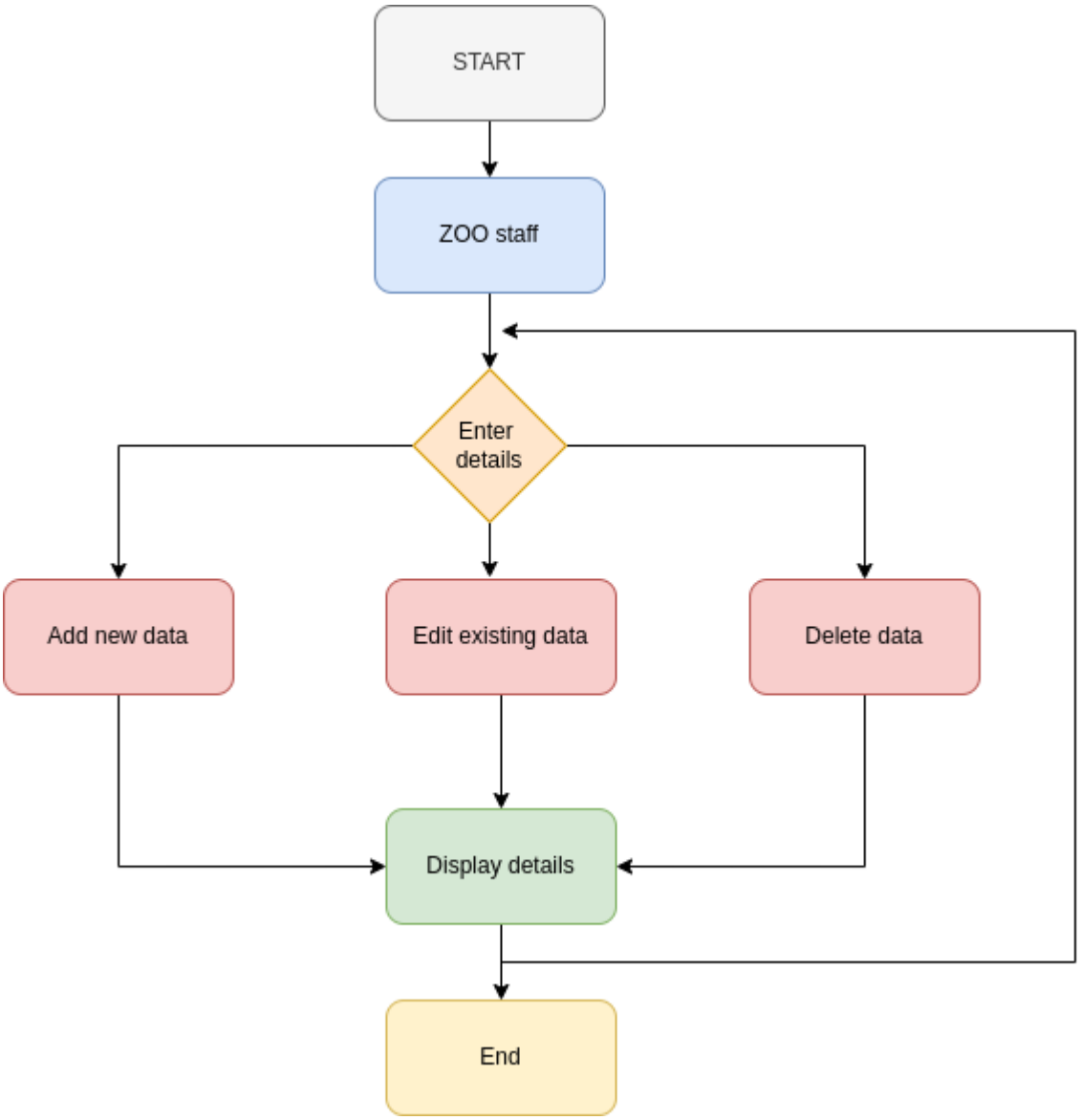


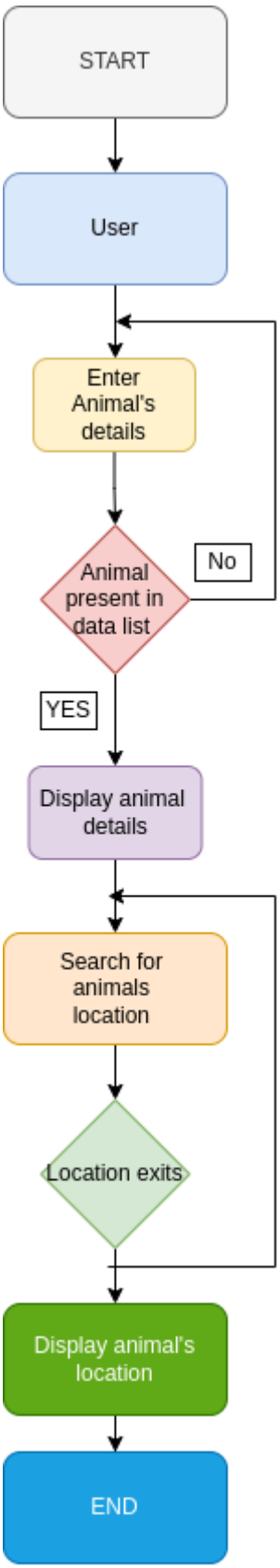
Black box



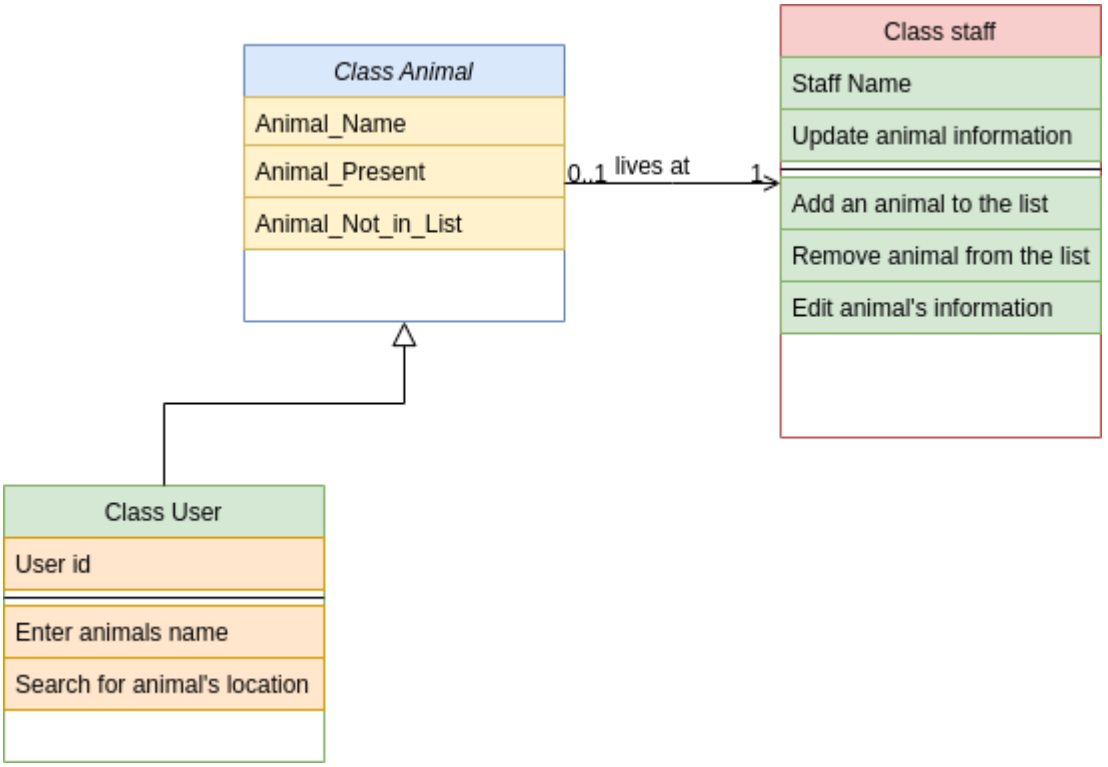
Program

Flow charts


START
ZOO staff
Enter details
Add new data
Edit existing data
Delete data
Display details
End


START
User
Enter Animal's details
Animal present in data list
No
YES
Display animal details
Search for animals location
Location exits
Display animal's location
END

Class diagram



6. Results:

This system was successfully able to maintain the data related to zoo animals such as storing animal details, updating details, deleting details and created an interface to zoo visitors to search animals, their locations and caretakers.

7. Conclusion:

This system is developed using the concepts of Object oriented programming in python languages. By creating different classes and objects to store and retrieve zoo related information, the data has been stored and used in an efficient manner.