Chapter 3

# Introduction

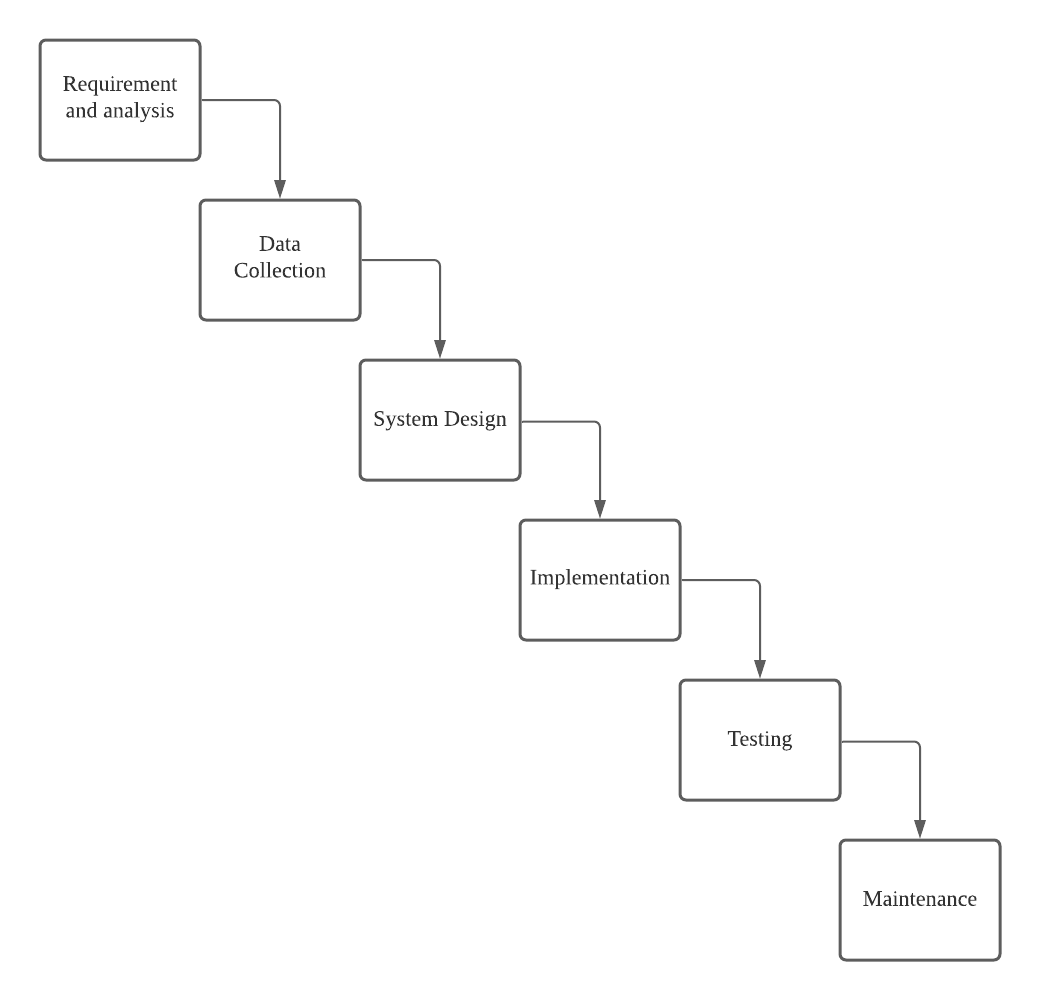
In this chapter, we will examine the approach that will be employed in this project. Besides, this chapter will outline the stages of the implementation and methodologies that will be employed in this project. Furthermore, all the implementation details will be given in this chapter as well.

# Development Methodology

To create, test and release high-quality applications, software developers have been encouraged to use Software Development Life Cycle (SDLC). The SDLC's goal is to provide high-quality software that meets or exceeds expectations while staying on schedule and staying under budget. When it comes to creating top-notch applications, the Software Development Life Cycle (SDLC) is the method used by the industry. Software that meets or exceeds expectations while staying on budget and on schedule are the goals of the SDLC. There are a variety of software development life cycle models that may be used throughout the process of creating software. They are also known as "Software Development Process Models" (SDPM). Each process model has a unique set of stages that must be followed to assure the success of the software development process. Most often used SDLC models include Waterfall Model, Spiral Model, V-Model and Big Bang Model, among others.

# Project Framework

For this research it will focus more on the Waterfall model, where phase-to-phase progress cannot be made if the previous phase has not been finished. The Waterfall Model is a step-by-step approach to software development that uses predefined stages to keep things organized. Before the following phase may begin, the previous one must be finished with no overlap. The SDLC is divided into phases, each of which has a particular task to accomplish. Winston Royce first debuted it in 1970 (Rungta, K. 2021).



*Figure 3.1: Waterfall Model*

## Requirement Analysis and Planning

This is the most crucial and foundational step of the SDLC, as well as the most time-consuming. An examination of the previous chapters of this study has been carried out in order to determine what kind of application is going to be used, the issue statements, the scope of this investigation, as well as the methodologies and software that will be appropriate for this research. The information gathered from the core project methodology and product feasibility studies is used to plan and execute the project once it has been formed. The planning step considers the project's quality assurance requirements as well as any potential dangers or risks. To properly execute the project with the least level of risk, the technical feasibility study must identify the several technological approaches that might be applied.

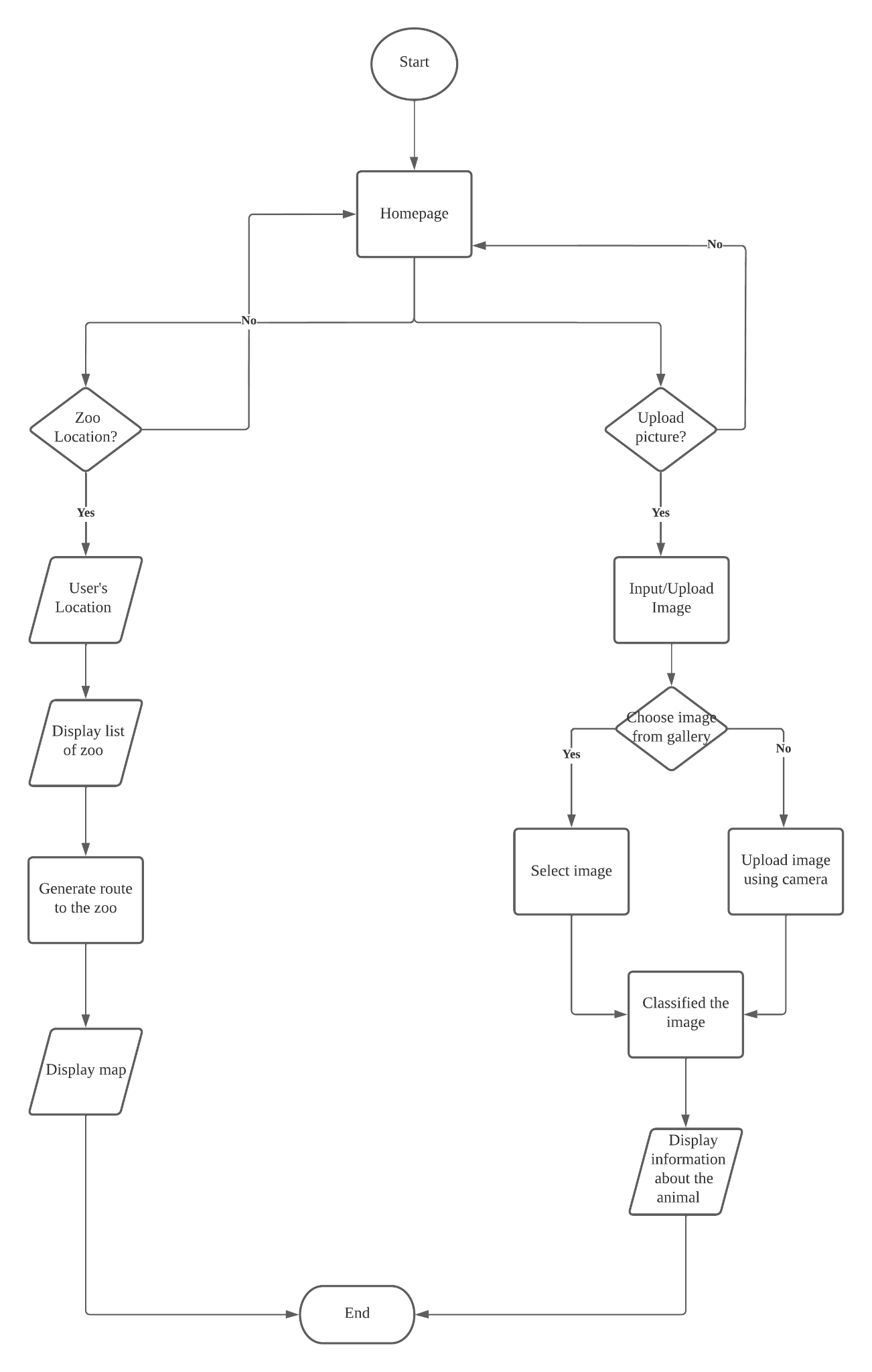
## Data Collection

A particular species of animal in Malaysia, namely the mammal, will be the focus of this research's data gathering efforts. Google images will be utilized as the datasets for training the image classification model, with images from Google Images serving as the source of the datasets for training. For the reason that Google photos are free and open source, it will be utilized in this research.

## System Design

Product requirements, including scenarios, layouts, and data models, guide the work of software developers. In the first place, a higher-level or logical design is made to define what the project is for and what it will accomplish, as well as the overall traffic flow of each part and the interconnection points. For this phase, identify the details of the system flow and design it is needed. It is because Flowcharts help programmers and businesspeople communicate better. In the programming of a problem, flowcharts play an important role and are very useful in grasping the logic of complex and long issues. The use case diagram also needs to be included in this phase where it can aid in the comprehension of how a user could engage with the technology designed. Ultimately, it should assist in developing and arranging requirements. It's a good initiative to create the application's user interface now that the system's flowchart and use case diagram have been completed.

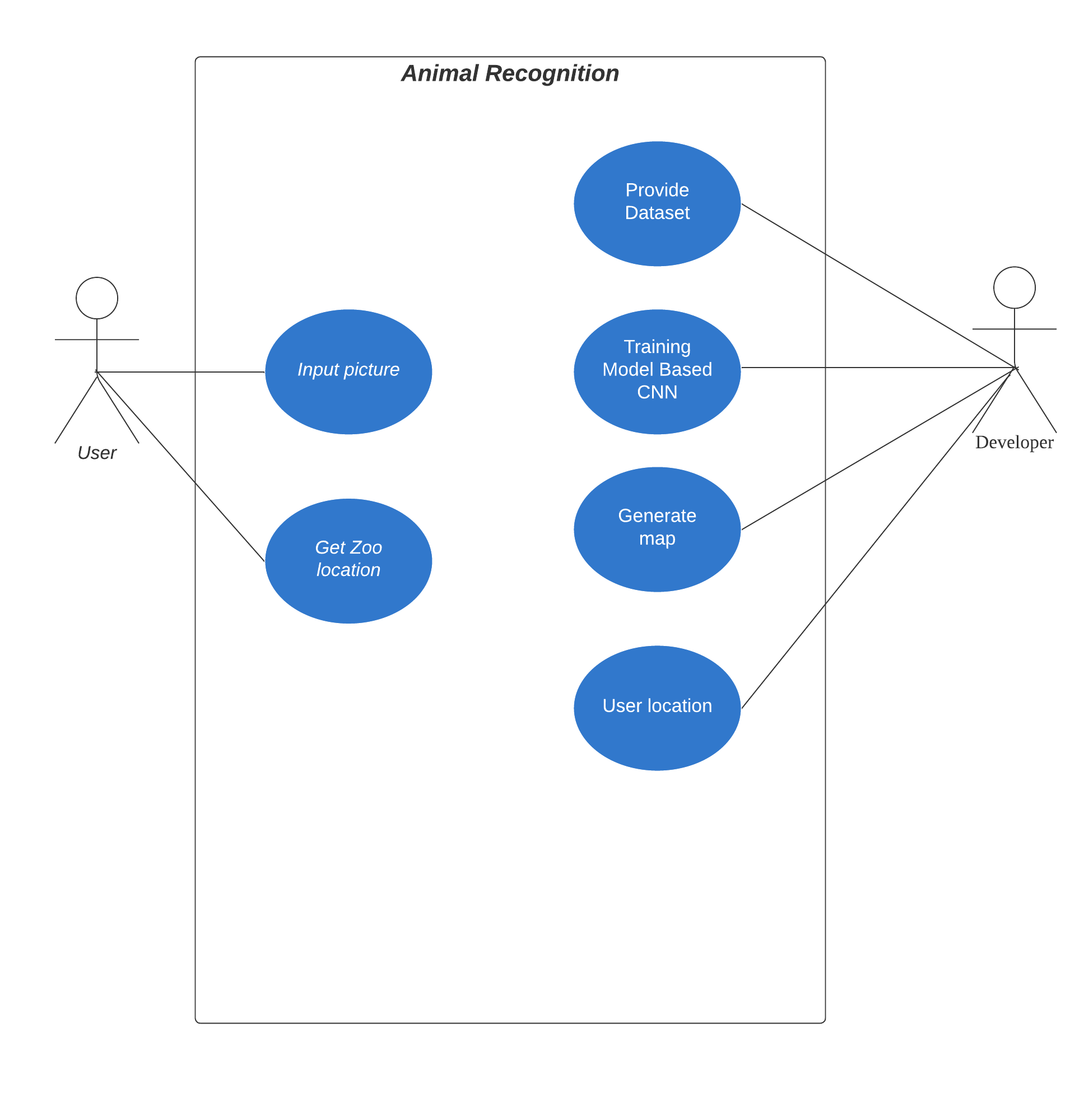
### System Flowchart



*Figure 3.2: Flowchart of the system*

Based on the Figure 3.2, it shows a brief about the system flow where it starts with displaying the homepage of the application then users have been given options to choose such as zoo location and upload a picture of an animal to start the process of the image recognition then display the output of the recognition and the details about the animal. Users need to give permission to turn on their location to know the location of the zoo. For the image recognition, users are allowed to upload an image to start the prediction animal then it will display the result of the prediction.

### Use case Diagram



*Figure 3.3: Use case diagram*

Based on the figure 3.3, it shows what functions and pages that will provide to the user.

|  |  |
| --- | --- |
| Use Case ID | UC 001 |
| Use Case | Upload Picture |
| Purpose | To upload picture for image recognition |
| Actor | User |
| Precondition | Click the upload option button on the navigation bar. |
| Main Flow | Choose whether upload image by camera or choose image from gallery |

|  |  |
| --- | --- |
| Use Case ID | UC 002 |
| Use Case | Get Zoo location |
| Purpose | User get the location and route to the zoo selected using map |
| Actor | User |
| Precondition | 1. Users need to click the option zoo location on the navigation bar. |
| Main Flow | 1. Users need to choose zoo the they preferred 2. Display Google map and route to the preferred zoo |

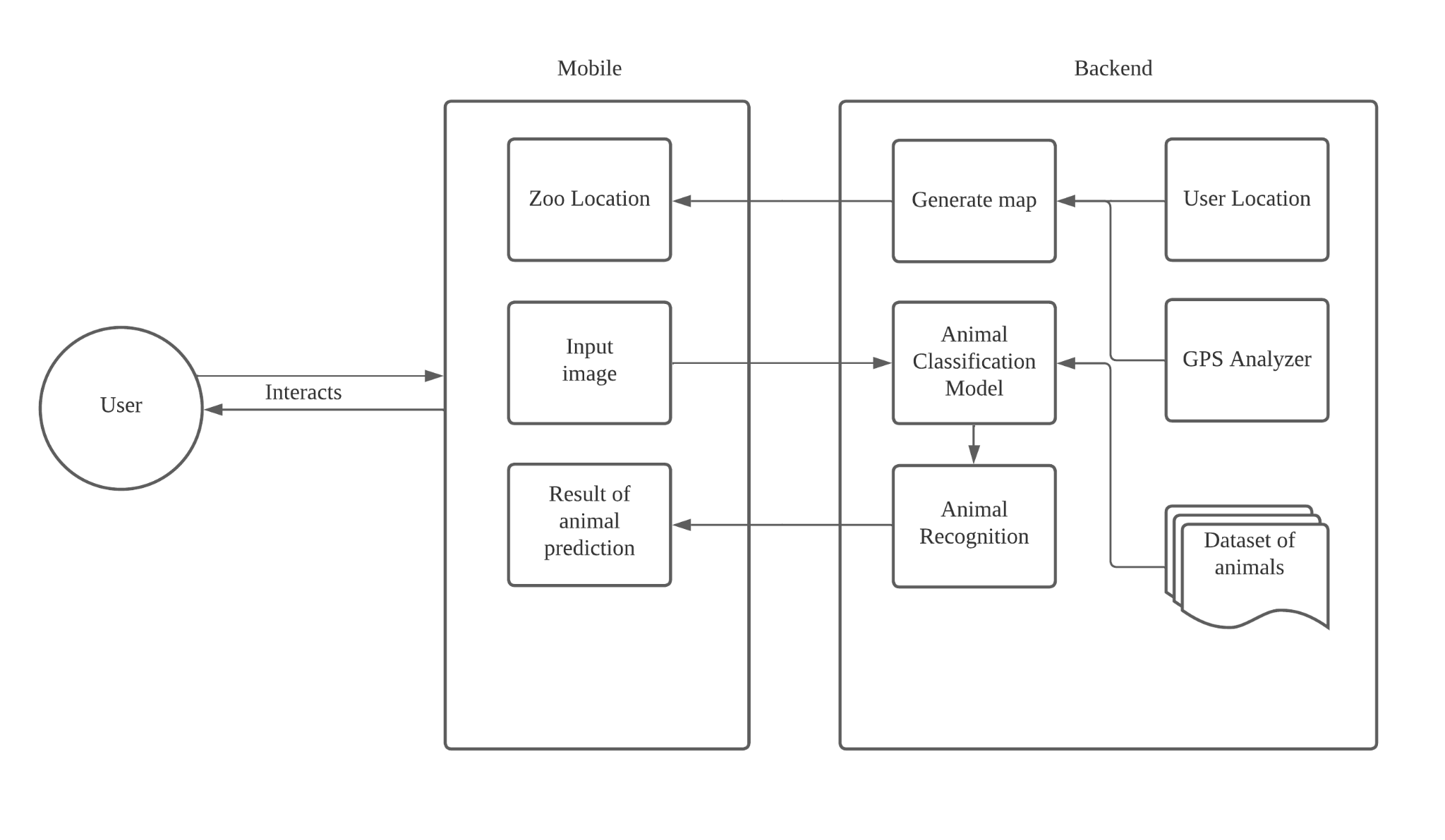
|  |  |
| --- | --- |
| Use Case ID | UC 003 |
| Use Case | Provide dataset |
| Purpose | Dataset of animals images to train and test the Convolutional Neural Networks (CNN) model |
| Actor | Developer |
| Precondition | 1. Download all images of animals in Google |
| Main Flow | 1. Make a folder of the dataset 2. Make subfolder for each animals to classify |

|  |  |
| --- | --- |
| Use Case ID | UC 004 |
| Use Case | Generate map |
| Purpose | Display user’s location, zoo location and route to the zoo |
| Actor | Developer |
| Precondition | Get Google map APIs key from Google API |
| Main Flow | 1. Allow to get location of the user 2. Get the location of the zoo 3. Make google map using Google APIs   Make route from the user’s location to the zoo |

|  |  |
| --- | --- |
| Use Case ID | UC 005 |
| Use Case | Training Convolutional Neural Networks Model |
| Purpose | To make classification for the animals using images |
| Actor | Developer |
| Precondition | Dataset for training, testing and validation |
| Main Flow | 1. Import TensorFlow libraries   Create convolutional neural network model using TensorFlow |

### System Architecture

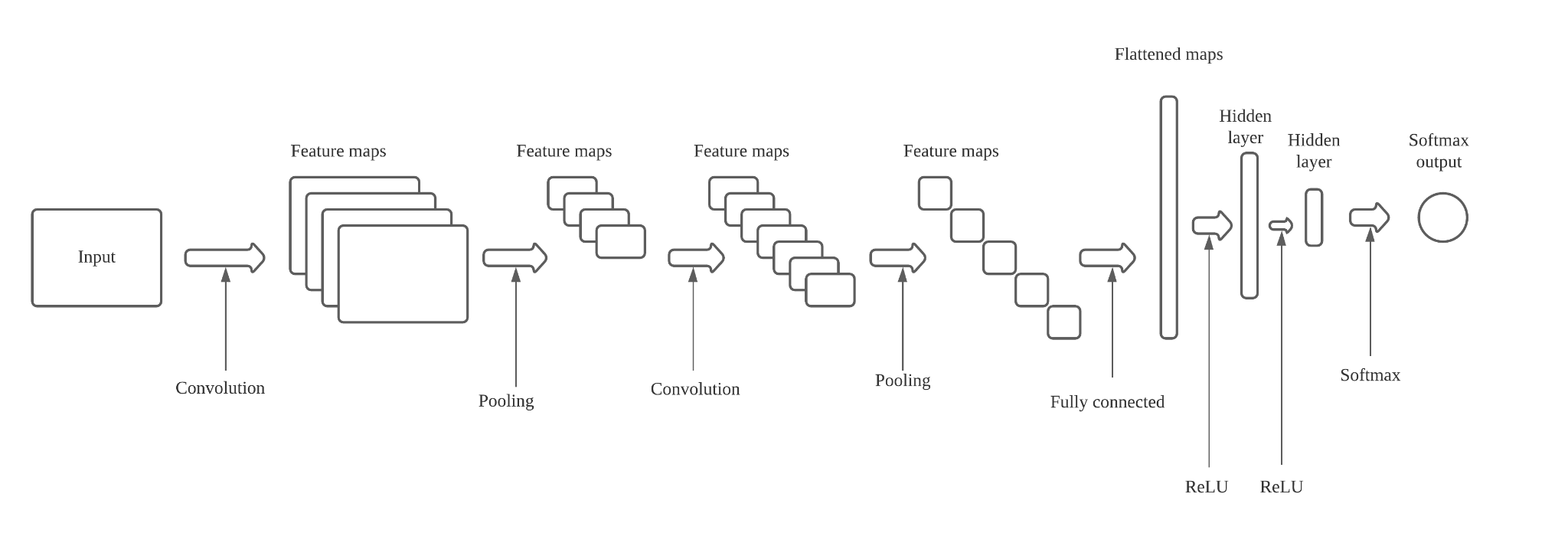
The architecture of a system is defined by its conceptual model, which identifies the system's structure, behavior, among other operational details. Abstract representations that may be used to draw conclusions about the structure and behavior of systems are known as architecture descriptions, which are formally described and represented systems.



*Figure 3.4: System Architecture diagram*

Based on Figure 3.4, it shows an overview of how the picture that was uploaded by the user was predicted by utilizing TensorFlow and Keras, and then how the result of the prediction was stored in the database and then displayed to the application.

### Image Classification using Convolutional Neural Networks (CNN)



*Figure 3.4: Convolutional Neural Networks (CNN) Architecture diagram*

The image's raw data is stored in the Input Layer. Afterward, the Convolution Layer, which computes volume by calculating the dot product between each filter and picture patch, follows. It's possible to create an output volume of 32 x 32 x 12 by using 12 filters on this layer. When it gets to Pool Layer, it does so on a periodic basis, which is essential for speeding up computation while reducing memory use and preventing overfitting. Max pooling and average pooling are two of the most popular forms of pooling layers. Next Full-Connected Layer, which is a normal neural network layer that receives input and computes class scores and outputs a 1-D array of size equal to the number of classes. Activation Function Layer is next in line of progression where the output of the convolution layer will be activated by an element-wise activation function in this layer. RELU is a commonly used activation function.

### User Interface Design

|  |  |
| --- | --- |
| **Sketch** | **Explanation** |
| *Figure 3.5: Homepage* | This is the homepage of the application where it will have three option button for each page of the application which is homepage, animal image recognition and zoo location |
| *Figure 3.11: Nearest zoo page* | This is the Zoo location page where it will display a list of zoo locations. |
| *Figure 3.12: Nearest zoo map* | This is the zoo location map where it will appear after users click on one of the zoo location options. |
| *Figure 3.13: Upload image page* | This is an upload image page where users need to upload an image of an animal that they want to predict. |
| *Figure 3.14: Prediction page* | After users upload an image and click on the predict button, it will display the classification of the image and give the details about the animal. |

## Pseudocode

|  |
| --- |
| Animal Recognition |
| BEGIN  IF user select gallery  Go to gallery  Choose image  Upload image THEN print result of recognition  ELSE IF select takepicture  Take picture THEN print result of recognition  ENDIF  END |

|  |
| --- |
| Maps |
| BEGIN  IF user select zoo negara  Users need to turn on GPS location  Get the users location  THEN display route to the zoo negara  ELSE IF user select zoo melaka  Users need to turn on GPS location  Get the users location  THEN display route to the zoo melaka    ELSE IF user select zoo taiping  Users need to turn on GPS location  Get the users location  THEN display route to the zoo taiping  ENDIF  END |

## Implementation

For the implementation phase, all systems based on the requirements from the system design such as zoo location and animal classification will be developed, then integrate all the systems together.

### Animals Image Recognition Application

In this step, mobile application's interfaces and certain functionality, such as logging in or registering, logging out, and entering an image for the image recognition will be developed. A mobile app for the Animals Image Recognition will be built on Android Studio in order to fulfill this level which will use mostly Java as its front end and back end of the application.

### Image Recognition using Convolutional Neural Networks (CNN)

As part of my study for Chapter 2, it will use the Convolutional Neural Networks technique to detect the animals in photos that will get the input from the user, which will be used to identify the animals. This part will be coded in the Python programming language, with various open-source libraries such as TensorFlow, Matplotlib, and OpenCV being used as resources. TensorFlow will allow me to design a model that will be used to train and test on the dataset that will be created.

### Zoo Location

GPS technology to locate users' whereabouts for this feature. It will use this information to determine the location of the users, after which it will display all the zoo locations in the specified state. Users will be prompted for permission to switch on their GPS location for the system to know which state they are in in order to do this operation. This will be coded using Java and Google APIs for the geolocation and GPS.

* + 1. Gantt Chart

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Week**    **Task** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** |  |
| Semester 5 | | | | | | | | | | | | | | | |
| **Mutual**  **Acceptance Form**  **Submission (F1)** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Chapter 1:**  **Introduction** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Project Motivation Evaluation Form**  **Submission (F2)** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Submission of**  **Chapter 1** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Outline of**  **Chapter 2** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Chapter 2 –**  **Literature Review** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Submission of**  **Chapter 2** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Literature Review**  **Evaluation Form**  **(F3)** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Chapter 3 -**  **Methodology** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Submission of**  **Chapter 3** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Methodology**  **Evaluation Form**  **(F4)** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Submission of**  **Full Report** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Presentation of**  **Final Proposal** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Plagiarism**  **Checking** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Writing Proposal**  **Report** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

### Hardware

*Table 3.1: List of hardware*

|  |  |  |
| --- | --- | --- |
| **No** | **List of hardware** | **Specification** |
| 1 | Processor (CPU) | Intel i3-6006U |
| 2 | Graphic Card (GPU) | Nvidia GeForce 920mx |
| 3 | RAM | 12 GB |

### Software

*Table 3.2: List of software*

|  |  |  |
| --- | --- | --- |
| **No** | **List of software** | **Details** |
| 1 | Windows | Windows 10 Pro edition 64-Bit Operating System |
| 2 | React Native | Platform to develop a mobile application |
| 3 | MySQL | Database for the application |
| 4 | Sqlite3 | Database form the application |
| 5 | Android Studio | To deploy the phone emulator |
| 6 | Canva | To design the mobile application |

## Testing

An application's TEST CASE is a sequence of activities carried out to validate a certain feature or operation of the program. A test case comprises the procedures, data, preconditions, and postconditions produced for a given test scenario to check any requisites that may be specified. To assess if a software product is meeting the needs of the client, testing engineers use a set of variables or conditions to compare predicted and actual outcomes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Scenario** | **Steps** | **Input** | **Output** | **Result** |
| 1 | Launch the application | Open the phone emulator on the Android Studio | None | The application will display the homepage | Success or fail |
| 2 | Animal recognition | Click on the upload image option then choose either to upload image using camera or pick image from gallery | Animal image | Display the result of the recognition | Success or fail |
| 3 | Zoo location | Users need to give permission to turn on their GPS location. | User’s GPS location | Display all list of zoos | Success or fail |
| 4 | Map Route | User need to lick on zoo location option | User’s  location | Route from user;s location to the zoo location | Success or fail |
| 5 | Image prediction accuracy | Using TensorFlow libraries to get the prediction accuracy | Animal Image | Accuracy of the prediction | Success or fail |

# Summary

This chapter explains the project's methodology and the Waterfall model of the Software Development Life Cycle (SDLC). This project's structure includes the project's goals, phases, activities, and results. In this project, the five processes of analysis, data collecting, system design, implementation and testing are covered in depth. Animals in Malaysia, hardware and software requirements have been examined throughout the requirements analysis process. The use case for the application, the system flowchart, the system architecture, and the CNN architecture were all created and discussed throughout the system design phase. The platform will be used to build a mobile app in React Native using open-source frameworks and Python as the language to train and test the model in the implementation phase. During the testing process, this application will check for the mobile app's operation and determine the recognition accuracy.

# References

Rungta, K. (2021, December 11). What is the waterfall model in SDLC? Advantages and disadvantages. Meet Guru99 - Free Training Tutorials & Video for IT Courses. https://www.guru99.com/what-is-sdlc-or-waterfall-model.html