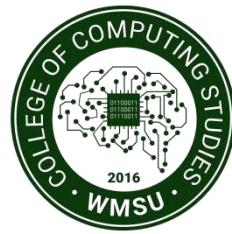




Republic of the Philippines  
Western Mindanao State University  
**COLLEGE OF COMPUTING STUDIES**  
Department of Information Technology



## **CURAHERB: INTELLIGENT HERBAL PLANT RECOGNITION AND MEDICATION GUIDE**

In partial fulfillment of the requirements in  
IT140 – IT Capstone 2

Submitted by:

Dawami, Hesam K.

Hassan, Shyniemar A.

Mapasi, Margo Kate G.

Submitted to:

Mr. Jason Catadman



WESTERN MINDANAO STATE UNIVERSITY  
Zamboanga City, 7000 Philippines

College of Computing Studies  
Information Technology Department

### CERTIFICATE OF PANEL APPROVAL

The Capstone Project, attached here to, entitled "**CURAHERB: INTELLIGENT HERBAL PLANT RECOGNITION AND MEDICATION GUIDE**", prepared and submitted by **SHYNIEMAR A. HASSAN, HESAM K. DAWAMI, MARGO KATE G. MAPASi** in partial fulfillment of the requirements for the degree **BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY**, is hereby recommended for approval.

**MS. LUCY F. SADIWA**  
Chairperson

**MR. JHON PAUL I. ARIP**  
Member Date

**MS. JUSTINE ANN C. ALBAY -ARIP**  
Member Date

**MR. LORENZO, JEZEREEL B.**  
Member Date

**MR. JASON A. CATADMAN**

Adviser

\_\_\_\_\_  
Date

\_\_\_\_\_

This Capstone Project is approved in partial fulfillment of the requirements for the degree **BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY**

**JOHN ED AUGUSTUS A. ESCORIAL, MIT**  
Department Head, IT

\_\_\_\_\_  
Date

**DR. RODERICK P. GO**  
Dean, CCS

\_\_\_\_\_  
Date

## **DEDICATION**

This project stands as a testament to relentless dedication and unwavering commitment. We extend our heartfelt gratitude to those who've been our pillars of support and inspiration. To our families whose unwavering love fueled our determination, and to our closest friends whose guidance and encouragement have been invaluable, you've been the bedrock of our journey.

Above all, we acknowledge the Almighty for blessings that bestowed us with strength and wisdom throughout this endeavor.

## **ACKNOWLEDGEMENT**

The development of CuraHerb owes its success to the invaluable contributions of several individuals and entities. We extend our deepest gratitude to Mr. Jason Catazman, whose role as our IT Capstone 2 Adviser profoundly influenced the project's trajectory with insightful guidance and recommendations in documentation and system development. To Mr. John Ed Augustus A. Escorial, our IT Capstone 1 Subject Teacher, your unwavering support and guidance throughout the capstone documentation project have been instrumental. Our heartfelt appreciation also goes to the Experts for the opportunity to present our work with listed herbal plants, for guiding the team and for the invaluable suggestions that enriched CuraHerb.

To our families, your unyielding support and understanding sustained our perseverance and dedication. Lastly, we express profound thanks to the divine force for continually providing us with knowledge, courage, and enlightenment during the journey of crafting CuraHerb.

## **EXECUTIVE SUMMARY**

In a world where the interest in natural remedies is growing, CuraHerb aims to bridge the gap between traditional herbal knowledge and modern convenience by offering an intelligent herbal plant recognition and medication guidance system. CuraHerb is a user-friendly platform that helps people discover and use herbal plants for better health. It's like a smart helper that finds the right plants based on your health needs or symptoms. With CuraHerb, you get clear info about each plant's benefits, how much to use, and any precautions.

This intelligent solution addresses the challenges faced by users in recognizing and understanding the medicinal properties of herbal plants. Leveraging advanced technology, CuraHerb empowers users to search for herbal plants based on herbal names, symptoms or health conditions, providing comprehensive information on botanical names, descriptions, medicinal properties, precautions, and the plants location. Key features of CuraHerb include an extensive herbal plant database, intuitive user interface, herbal name-based search functionality, medication guidance, user feedback or testimonials, mapping, and a seamless user experience across multiple platforms.

CuraHerb seeks to enhance the herbal medicine landscape by offering a reliable, accessible, and user-friendly platform that fosters informed decision-making and safe usage of herbal remedies. By providing a blend of traditional wisdom and cutting-edge technology, CuraHerb endeavors to be the go-to resource for individuals seeking natural and holistic healthcare solutions. CuraHerb wants to be your go-to guide for safe and smart use of herbal medicine, making it simple for everyone to explore nature's healing powers.

## TABLE OF CONTENTS

ACKNOWLEDGEMENT.....	4
CHAPTER 1 – INTRODUCTION.....	3
1.1 .....	.
Background .....	3
1.2 Purpose and Description .....	4
1.3. Objectives of the Study.....	5
1.4 Scope and Limitations .....	6
1.4 Definition of Terms.....	9
Chapter II – REVIEW OF RELATED LITERATURE .....	10
Summary .....	11
Chapter III - TECHNICAL BACKGROUND .....	14
3.1 Conceptual Framework .....	14
3.2 Software Requirements -+.....	16
3.2.1 Programming IDE .....	16
3.2.2 Programming Language .....	16
3.2.c Software Developer Kit .....	17
3.3 Hardware Requirements .....	17
Chapter IV – Design and Methodology .....	18
4.2 Requirements Analysis .....	19
4.2.1 Flow Chart.....	19
4.2.2 Use Cases .....	20
4.3 Requirements Documentation.....	20
4.3.1 Purpose.....	21
4.3.2 Goals .....	21
4.3.3 Project Scope.....	21
4.3.4 Success Criteria .....	22
4.3.5 User Characteristics .....	22
4.3.6 Mandated Constraints.....	23
4.3.7 Functional Requirements .....	25
Plant Identification .....	25
Herbal Plant Database .....	25
Medication Guide.....	25
User Profiles .....	25
Search and Filtering.....	25
User Feedback .....	26
Performance .....	27
Security .....	27
Scalability.....	27
Compatibility.....	27
Reliability.....	27
Data Integrity .....	27
4.4 Design of Software Systems, Products, and/or Processes.....	28
Incremental Delivery and Iterative Development .....	30
Continuous Integration and Deployment .....	30
Ongoing Maintenance and Enhancement .....	30
4.2.1.1 Data Flow Diagram (DFD).....	31
4.2.1.2 User Interface Application .....	31
4.2.2 User Input.....	32
Image Capture/Upload.....	32
User Input and Interaction .....	32
Image Recognition Feedback .....	32
Identification Results Display.....	32
Plant Information Display .....	32
Medication Guidance .....	33
Personalization and User Profile Management .....	34

4.2.3 Storyboard.....	34
5.4 Test Cases .....	62
CHAPTER 6 .....	96
RESULTS AND DISCUSSION .....	96
6.1 Introduction.....	96
6.2 Summary of Findings.....	98
CHAPTER 7.....	104
CONCLUSION AND RECOMMENDATIONS .....	104
REFERENCES .....	106
APPENDIX A.....	107
APPENDIX B .....	108
APPENDIX C .....	126
TESTING.....	126
APPENDIX D EVALUATION FORM .....	142
APPENDIX E .....	145
GANTT CHART .....	145
APPENDIX F .....	146
USE CASES .....	146
APPENDIX F .....	147
CLIENT ACCEPTANCE FORM .....	147
APPENDIX H.....	148
USER GUIDE.....	148

## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Background**

Herbal plants, also known as medicinal plants or herbs, are plants that are used for their therapeutic properties. These plants have been utilized for centuries across various cultures for their medicinal, aromatic, and culinary benefits. Herbal remedies are often derived from different parts of these plants, including leaves, flowers, roots, seeds, or bark. These plants are valued for their natural compounds, which often exhibit therapeutic properties. The use of herbal remedies is deeply rooted in traditional knowledge and has been passed down through generations.

There has been a growing interest in herbal medicine in recent years due to its potential health benefits and natural healing properties. Herbal plants have been used for centuries in various cultures to treat illnesses and promote well-being. However, the contemporary use of herbal remedies is not without challenges. One significant issue is the lack of the main problem associated with herbal plants not being recognized and understood for their benefits and uses stems from several factors, including lack of awareness, limited scientific validation, and challenges in plant identification.

With advancements in technology and image processing techniques, there is an opportunity to develop automated systems for the classification and recognition of herbal plants based on their leaf images. Such a system, known as CuraHerb, has the potential to revolutionize the field of herbal medicine by assisting in identifying and utilizing medicinal plants and its uses through image upload and image scanning. CuraHerb would utilize these principles to analyze and interpret leaf images of medicinal properties. Through the training process, the system learns the distinctive features of various plant species, enabling it to classify and identify them accurately. The intelligent recognition aspect allows the system to handle variations in lighting, scale, and orientation, making it robust and versatile for real-world applications in herbal medicine.

Integrating the intelligent recognition and use of therapeutic plants, a system like CuraHerb has the potential to transform the field of herbal medicine along with establishing standardized practices for plant identification, can contribute to a more comprehensive understanding and utilization of the benefits of herbal plants.

In the context of the automated system CuraHerb, the integration of herbal knowledge becomes pivotal. By harnessing the power of technology to identify and classify herbal plants based on their leaf images, CuraHerb not only aligns with the growing interest in herbal medicine but also addresses the need for efficient and accurate tools in this field. This system can play a vital role in empowering individuals, healthcare professionals, and researchers to harness the benefits of herbal remedies, promoting a more integrative and personalized approach to health and wellness. As society continues to recognize the importance of herbal medicine, innovative solutions like CuraHerb mark a significant milestone in bridging traditional wisdom with contemporary advancements, fostering a healthier and more sustainable future.

## **1.2 Purpose and Description**

The project aims to develop an intelligent herbal plant recognition system that accurately identifies medicinal plants with several key features designed to provide a comprehensive and user-friendly experience here in Zamboanga City. The system offers a employing cutting-edge image recognition algorithms, help in analyzing visual features of medicinal plants, and continuously improving its accuracy through machine learning. Users also can effortlessly scan and upload plant images through an intuitive interface, receiving real-time feedback on identified plants, including names and essential information. The system also allowed the users to contribute additional plant images and information on the herbal plants. A verification process ensures the quality and reliability of user- contributed data, fostering a dynamic and continually evolving database. The system also has a extensive database ensures a thorough coverage of medicinal plants specific to Zamboanga City, forming the basis for a detailed medication guide. This guide encompasses common and scientific names, medicinal

properties, uses, habitat information, and awareness of potential side effects or allergic reactions, offering users an in-depth understanding of each plant's characteristics. The system also includes mapping capabilities, allowing users to track the distribution of herbal plants within the city. This mapping feature adds a spatial dimension to the platform, offering insights into the geographical prevalence of different plant species.

The intelligent herbal plant recognition system developed for Zamboanga City has the potential to benefit various stakeholders such as Researchers and Botanists, Community Members, Local Authorities, Environmentalists and educational institutions that can contribute to the well-being of both the community and the environment.

### **1.3. Objectives of the Study**

#### a. General Objectives

The goal of this research is to develop a Herbal Plant recognition with medication guide for Zamboanga Peninsula that would allow people to search, upload, scan and locate herbal plants data in real-time. There are a lot of herbal plants have been discovered but not properly identified and being ignored for its benefits. As a result, the team wants to enhance the knowledge about the identification of herbal plants for its vital role for personal health and safety, as well as for broader ecological and cultural reasons.

#### b. Specific Objectives

The Curaherb system has the following objectives:

- To Create a system that accurately identifies and classifies various herbal plants through scanning and uploading images. This can be useful for individuals, researchers, and practitioners in the fields of herbal medicine, botany, and conservation.
- To develop a Medicinal Plant Database comprehensive Medicinal Plant Database that provides information on the medicinal properties, uses, and potential side effects of different herbal plants. This can serve as a valuable resource for healthcare professionals, herbalists, and individuals interested in

alternative medicine.

- To create an educational platform within the system to raise awareness about the identification, cultivation, and sustainable harvesting of herbal plants. This can help promote responsible and informed use of medicinal plants.
- To Create a visual representation of the geographic distribution of herbal plants and Users in the field can use the mapping feature to geotag locations where they find herbal plants. This can be beneficial for researchers, conservationists, and policymakers interested in understanding the biodiversity of medicinal plants in different regions.

#### **1.4 Scope and Limitations**

The scope of the Curaherb project encompasses the development of a comprehensive online platform dedicated to herbal remedies and holistic wellness. The project aims to provide users with a user-friendly interface where they can access information about various herbs, their medicinal properties, and practical tips for incorporating herbal remedies into their lifestyle. Additionally, the platform will include features such as user mapping, user reviews and engagement testimony for herbal products.

The public will have a unified platform wherein they would be able to see information such as the current herbal plants in the library. The admin is often tasked with decision-making, resource allocation, conflict resolution, and overall project oversight. They will have the access to add, delete, and edit information within the platform. Meanwhile, the users will be able to access the site on their respective devices through the internet because this project is a web-based application. Curaherb is a proposed project that aims to promote a natural and balanced approach to health by offering a curated selection of herbal products, accurate information, and community engagement. It also envisions creating an online platform for people interested in holistic wellness and herbal remedies.

Listed below are the functions and limitations of this study:

- The system prominently displays and highlights various herbal plants listed in its database. Users can easily access information about each plant, including their properties, uses, and cultivation guidelines.
- The system features an admin dashboard accessible only with specific administrative credentials. This restricted access ensures that ordinary users cannot alter critical settings or access privileged information.
- The entire project operates on a web-based platform, requiring users to have an internet connection to access the site. This design choice enhances accessibility but comes with the limitation of dependency on an online connection.
- As a web-based project, users must have a reliable internet connection to access the system. This limitation may affect users in areas with poor connectivity.
- While the admin dashboard provides essential controls, its restricted access may limit the number of individuals who can actively manage and oversee the system. Adequate training and support for administrators are crucial.
- The accuracy of herbal plant information and vaccination rates depends on timely updates. The system's effectiveness is contingent on the availability of up-to-date data from reliable sources.
- Users may face challenges related to authentication when accessing the system. The necessity for login credentials may pose usability issues, and the system should implement user-friendly authentication processes.
- The web-based nature of the project may present compatibility issues with certain devices or browsers. It's essential to ensure that the system is optimized for a range of devices and platforms.
- The security of user data and confidential information on the admin dashboard is crucial. The system must implement robust security measures to protect against unauthorized access and potential data breaches.
- The success of community forums and user engagement relies on the active participation of users. The system may not achieve its full potential if users do not actively contribute to discussions and share their experiences.

## **1.5 Significance of the Project**

This suggested solution is significant since it includes a mapping feature, upload and scanning and a herbal library for herbal plants. The data is available by the public, but can only be altered by the site administrator. In addition, the project manager would be in charge of maintaining a log of newly recognized herbal plants, which would then be entered into the system. This project will aid in the precise identification of herbal plants, and anyone looking for natural remedies or alternative healthcare options will find a comprehensive drug reference useful. With reliable information about medicinal plants at their disposal, people will be more prepared

Below are the people who will benefit from this system:

- Individuals Seeking Natural Remedies
- Researchers and Botanists
- Community Members
- Local Authorities
- Environmentalists
- Educational institutions

The accomplishment of this project could establish a foundation for upcoming researchers, offering a reference or blueprint for enhancement. The project aims to furnish future researchers with comprehensive documentation of essential module codes and their fundamental functions. Additionally, it would offer herbal medicine enthusiasts valuable insights into the potential long-term health effects associated with herbal plants.

## 1.4 Definition of Terms

- **CuraHerb** - short term for Curable herbal
- **Herbal Medicine** - The practice of employing plants or plant extracts for therapeutic purposes to enhance health and treat a variety of illnesses.
- **Medicinal plants** - Plants with traits or constituents that are medicinal for human health. that have therapeutic uses, including symptom relief, immune system stimulation, and the treatment of diseases.
- **Image processing** - refers to the use of these methods to the extraction of characteristics from herbal plant leaf photographs for classification and recognition purposes.
- **Herbal Plants** - Plants used for therapeutic purposes, also known as medicinal plants or herbs. They have been employed across various cultures for centuries for their medicinal, aromatic, and culinary benefits.
- **Medicinal Plant Database** - A comprehensive database providing information on the medicinal properties, uses, and potential side effects of various herbal plants. It serves as a valuable resource for healthcare professionals, herbalists, and individuals interested in alternative medicine.
- **Educational Platform** - A feature within the system aimed at raising awareness about the identification, cultivation, and sustainable harvesting of herbal plants. It promotes responsible and informed use of medicinal plants.
- **Mapping Feature** - A functionality allowing users to visually represent the geographic distribution of herbal plants. Users can geotag locations where they find herbal plants, providing insights for researchers, conservationists, and policymakers.
- **Intelligent Recognition** - The ability of the system to analyze and interpret leaf images of medicinal properties. Through a training process, the system learns distinctive features of various plant species, enabling accurate classification and identification.
- **Web-Based Project**:A project that operates on a web-based platform, requiring users to have an internet connection for access. connection.

## **CHAPTER II**

### **REVIEW OF RELATED LITERATURE**

#### **2.1 Introduction**

This study reviewed the literature on the design, development, and implementation of the CuraHerb: Intelligent Herbal Plant Recognition, Medication Guide web-based system for Zamboanga City.

Machine learning is artificial intelligence that detects medical image patterns (Erickson et al., 2017). According to the author, deep learning has become an advanced version of machine learning, which means it can identify complex features without requiring complex calculating algorithms. This paper presents a review of machine learning techniques that are used in the identification of leaf plant diseases (Wasike et al., 2021). The article provides an overview of the various methods and their advantages and disadvantages. This study aimed to explore the multiple aspects of machine learning in agriculture. Apart from these, there are multiple aspects in which ML can be used (Kumar and Sonajharia, 2017)(Vipin et al., 2021).

The authors have used modern computing devices and technology to build a deep neural network-based model named Medicinal Neural Network (Amuthalingeswaran et al., 2019). Using this model, the authors have classified four classes of medicinal plants. To train this model, they have used 8000 images of these four classes. The proposed model has achieved a good accuracy of 85%. In this paper, a new dataset of 10 kinds of medicinal plants in Bangladesh has been introduced by the authors (Akter and Hosen, 2020). To extract the high-level features, a three-layer convolutional neural network is employed. The training was done on 34,123 images, and the models were tested on 3,570 images, giving an accuracy of 71.3%. This paper analyses the benefits of various automated leaf pattern recognition procedures (Kumar Thella and Ulagamuthalvi, 2021). A proposed computer vision approach can entirely ignore the context of the image and provide an accurate and fast leaf recognition process.

In this paper, the author tried a new approach to using deep learning methods to automatically classify and detect plant diseases from leaf images. The developed model was able to detect leaf presence and distinguish between healthy leaves and 13 different diseases, which can be visually diagnosed. The complete procedure was described, from collecting the images used for training and validation to image reprocessing and augmentation, and finally the procedure of training the deep CNN and fine-tuning. Different tests were performed to check the performance of the newly created model. (Sladojevic et al., 2016)

## Summary

In summary, the study focuses on using machine learning and deep learning techniques to accurately identify and classify herbal plant, such as image recognition with convolutional Neural Networks (CNNs), Feature Extraction with machine Learning algorithms, and data augmentation. The model can be trained on a dataset of labeled images to learn patterns and features that distinguish different plant species. By automating the identification process based on visual features, such as leaf patterns, the reliance on manual identification is reduced. The inclusion of a medication guide component in the system, called CuraHerb, provides users with relevant information on the medicinal properties and uses of recognized herbal plants. This is particularly beneficial for individuals seeking natural remedies or alternative treatments, as it empowers them to make informed decisions about the plants they encounter and their potential medicinal applications.

CuraHerb's intelligent plant recognition and medication guide functionalities offer advantages such as quick and accurate identification, even for users without extensive botanical knowledge. Additionally, the system helps mitigate the risks associated with misidentifying plants, especially in the context of medicinal herbs, where selecting the wrong plant could lead to adverse effects. By providing reliable information and guidance, CuraHerb promotes the safe and responsible use of herbal remedies.

Features	System A CuraHerb: Intelligent Herbal Plant Recognition with Medication Guide	System B Automated Plant Recognition System with Geographical Position Selection for Medicinal Plants
Upload Image	Yes	Yes
Capture Image	Yes (Advanced Image Recognition)	No
Mapping( locate the plant)	Yes (Geotagging and Mapping)	Yes (Basic Image Scanning)
Search Function	Yes	Yes
Herbal Plant Statistics(Admin)	Yes	No
Herbal reviews	Yes	No
Herbal Display (Description ,Medicinal Uses and side effects	Yes	Yes
Comment on mapping location	Yes (Community Data Contribution)	No
Upload new location	Yes	No
Add favorite	Yes	No

*Sample existing System Comparison Table*

The table comparison show the features of each of the system. "CuraHerb: Intelligent Herbal Plant Recognition with Medication Guide " and "Automated Plant Recognition System with Geographical Position Selection for Medicinal Plants" where CuraHerb: Intelligent Herbal Plant Recognition with Medication Guide has more features than Automated Plant Recognition System with Geographical Position Selection for Medicinal Plants. We'll discuss the benefits of CuraHerb due to its advanced features.

The image recognition, for accurate plant identification approach improves the system's precision in recognizing a wide range of plant species.

CuraHerb: Intelligent Herbal Plant Recognition with Medication Guide and "Automated Plant Recognition System with Geographical Position Selection for Medicinal Plants" are two plant identification systems that have different features to meet the demands and preferences of the user. With its wide range of features, Curaherb may be used to map the locations of detected plants as well as upload photographs and scan them using sophisticated image recognition algorithms. For researchers and amateurs interested in seeing the worldwide distribution of plant species, this extensive mapping tool is especially helpful. Curaherb's search feature makes it simple for users to look for information about particular plants or discover ones with particular traits. The system also offers data on herbal plants, which gives consumers important information about the distribution and frequency of detected plants. The inclusion of herbal reviews and a detailed herbal display, encompassing botanical descriptions, medicinal uses, and potential side effects, enhances the user's knowledge base. Moreover, the platform encourages community engagement by allowing users to contribute location data for mapping purposes, creating a collaborative and dynamic environment.

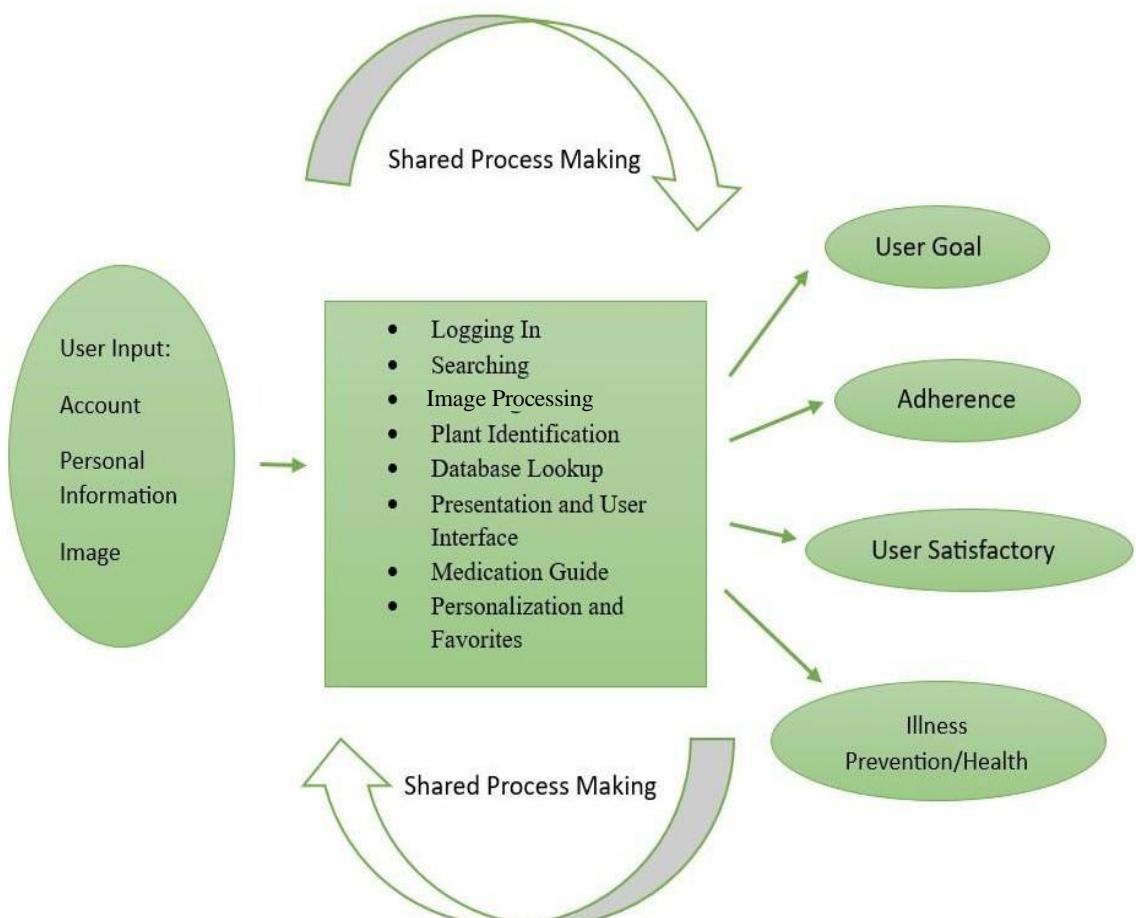
On the other hand, the Automated Plant Recognition System with Geographical Position Selection for Medicinal Plants, while efficient in image scanning for plant identification, lacks some of the advanced features present in Curaherb. It provides a straightforward user experience by allowing image uploads and basic identification of geographical locations. However, it falls short in offering mapping functionalities, herbal statistics, and the detailed herbal display found in Curaherb. The choice between Curaherb and other plant distribution systems depends on the user's specific needs, with Curaherb being more advanced and suitable for research, community engagement, and education, despite the absence of certain features.

In summary, Curaherb advanced features provide a more comprehensive and engaging user experience compared to the more basic functionalities offered by Automated Plant Recognition System with Geographical Position Selection for Medicinal Plants. The choice between the two systems depends on the user's specific requirements, ranging from plant identification to in-depth botanical knowledge, community interaction, and research pursuits.

## CHAPTER III TECHNICAL BACKGROUND

This chapter presents the technical background of CuraHerb, an intelligent herbal plant recognition and medication guide system. It begins with a conceptual framework that illustrates the various components utilized in the approved project. Furthermore, this chapter discusses the solutions that have been implemented to address the problems and needs identified in the previous chapters, as well as the software and hardware requirements necessary for the system.

### 3.1 Conceptual Framework



**Figure 1** Conceptual Framework

The CuraHerb system operates by leveraging image recognition technology, enabling users to search, scan, or upload images of herbal plants for identification. This technology processes and analyzes the images, accurately identifying plant species based on their unique characteristics and visual patterns. Complementing this technology is a comprehensive herbal plant database storing diverse information, including botanical names, common names, descriptions, medicinal properties, traditional uses, dosages, precautions, and potential side effects. Upon identification, the system's medication guide component generates detailed instructions on proper dosages, potential side effects, and precautions, considering individual health conditions and medication interactions. Users can personalize their experience by creating profiles, saving favorites, and receiving tailored recommendations, enhancing engagement and facilitating informed decision-making in exploring herbal medicine. The system's workflow involves users interacting through search, scan, or upload functionalities, with image recognition technology processing the input. This technology, coupled with algorithms and machine learning models, identifies plants by analyzing their visual features and cross-referencing them with the botanical database. Outputs include detailed plant information, medication guidance, and personalized recommendations based on user profiles and preferences. Various technologies, including image recognition, machine learning, database management, and recommendation systems, are integrated to streamline the process and enhance user experience.

\ This technology processes and analyzes the images, enabling accurate identification of plant species based on their unique characteristics and visual patterns. The system also supports a comprehensive herbal plant database, which stores information about various plants, including botanical names, common names, descriptions, medicinal properties, traditional uses, dosages, precautions, and potential side effects.

The system also includes a medication guide component, which displays available herbs based on identified plant species or symptoms. This component provides detailed instructions on proper dosages, potential side effects, and precautions, considering factors like individual health conditions and potential interactions with conventional medications. CuraHerb addresses the problems and

needs identified in previous chapters by providing accurate and reliable information about herbal plants, addressing the need for accessible and trustworthy knowledge in the field of herbal medicine.

The medication guide component offers practical solutions for safe and effective usage of herbal plants, ensuring user safety and informed decision-making. The user-friendly interface and personalized features cater to the need for an engaging and intuitive platform. Users can create profiles, save favorites, and receive tailored recommendations, empowering them to explore herbal medicine according to their individual preferences and needs.

### **3.2 Software Requirements --**

The following is the list of software requirements used for the development and implementation of the system.

#### **3.2.1 Programming IDE**

- **VISUAL STUDIO** – Visual Studio is an integrated development environment (IDE) developed by Microsoft. It provides a comprehensive set of tools and features for building software applications, including desktop applications, web applications, mobile apps, cloud-based services, and more.

#### **3.2.2 Programming Language**

**PYTHON** - Python is a widely-used programming language known for its readability and versatility across different domains like web development, data analysis, and artificial intelligence.

**JAVASCRIPT** - JavaScript is primarily used to enhance the interactivity and user experience of websites. It allows developers to manipulate HTML elements, respond to user actions (e.g., clicks, form submissions), and dynamically update content on the web page without requiring a full page reload.

### **3.2.c Software Developer Kit**

**Visual Studio SDK-** The Visual Studio SDK (Software Development Kit) is a set of tools and resources provided by Microsoft to extend and customize the Visual Studio IDE (Integrated Development Environment). It allows developers to create their own Visual Studio extensions, add new functionalities, and integrate external tools and services.

### **3.3 Hardware Requirements**

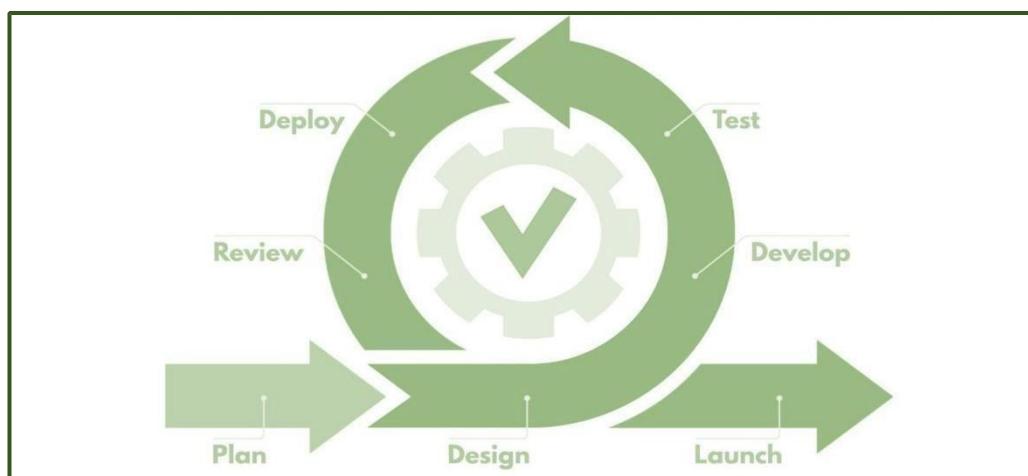
- Users accessing the system through web-based platforms will require devices such as desktops, laptops, or tablets with compatible web browsers and stable internet connectivity.
- The server or backend infrastructure should have sufficient processing power, such as multi-core CPUs or dedicated GPUs, to handle concurrent user requests efficiently, especially during the image recognition process.
- Adequate storage capacity is needed to store the herbal plant database, user profiles, uploaded images, and other system data. The storage capacity should be scalable to accommodate future growth.
- The system requires stable internet connectivity to access the herbal plant database, perform image recognition, and retrieve updated information. Users need a reliable internet connection to interact with the system effectively.

## CHAPTER IV

### DESIGN AND METHODOLOGY

#### 4.1 Introduction

In this chapter, we delve into the design and methodology aspects of CuraHerb, which lay the foundation for its development. The design phase focuses on the creation of software systems, products, and/or processes that contribute to the functionality, usability, and effectiveness of CuraHerb. The methodology chosen sets the framework for project management, collaboration, and development practices throughout the project's lifecycle.



*Figure 2. Agile Model*

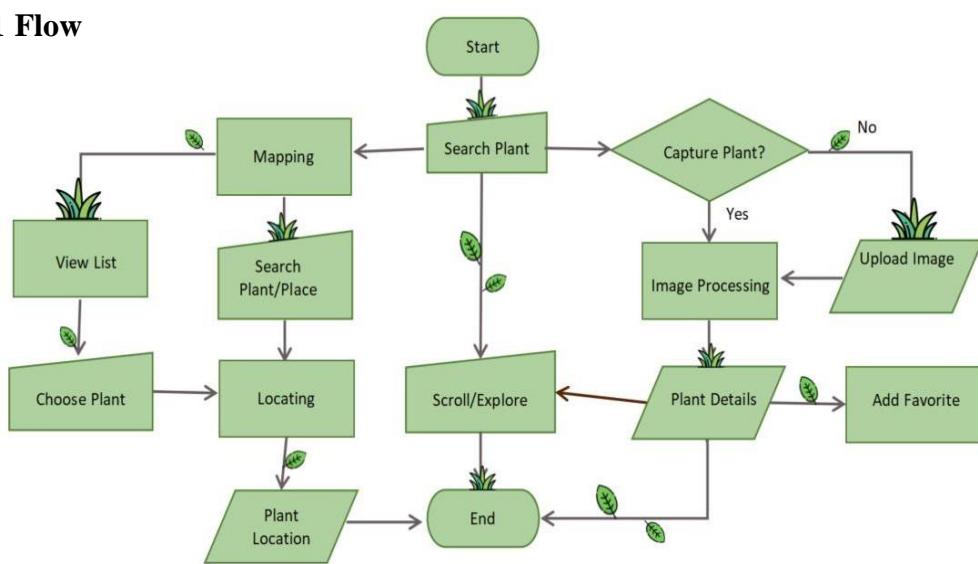
In developing the CuraHerb system, we applied Agile methodology to ensure flexibility, adaptability, and continuous improvement throughout the development process. This approach involved breaking down the development process into small, manageable iterations known as sprints, typically lasting two to four weeks. Each sprint focused on developing, testing, and delivering a set of features or functionalities, allowing us to quickly incorporate feedback and make adjustments as needed. Regular engagement with stakeholders, including users and domain experts, facilitated continuous feedback loops, ensuring the system met evolving needs and expectations. Cross-functional teams, consisting of developers, designers, and domain experts, promoted collaboration and communication, fostering a holistic approach to development. Embracing change was inherent to Agile, allowing us to adapt project scope, priorities, and features based on user insights. Transparency and open

communication were prioritized through daily stand-up meetings, sprint reviews, and retrospectives, enabling us to address issues and continuously improve processes. By focusing on delivering a minimum viable product (MVP) early on, we were able to release a functional version of the system to users, gather feedback, and iteratively enhance features over time. Overall, Agile methodology facilitated effective management of the development process for CuraHerb, promoting collaboration, responsiveness to change, and delivery of a system aligned with user needs.

## 4.2 Requirements Analysis

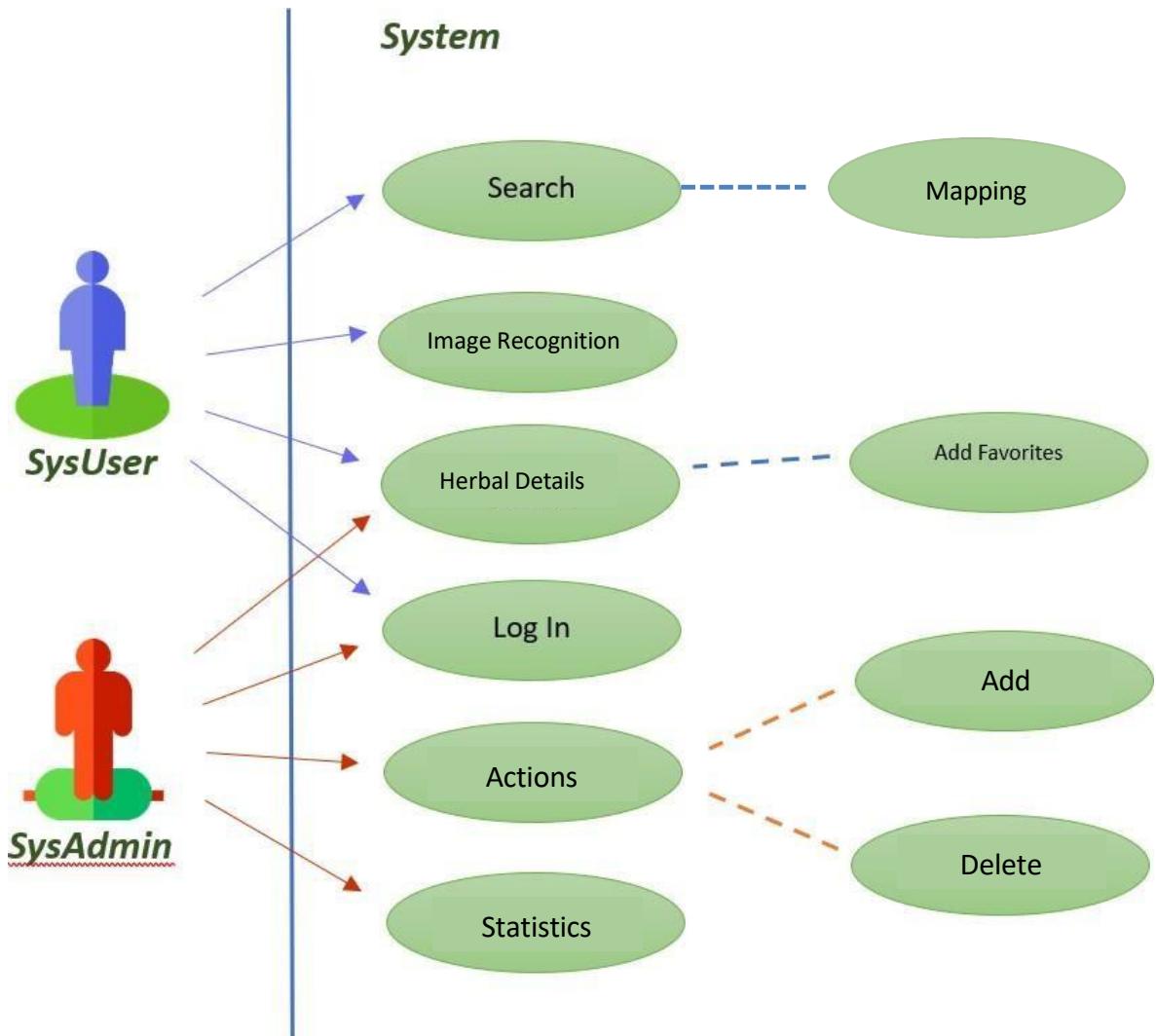
In this phase, we conducted an exhaustive analysis of the prerequisites for CuraHerb, engaging in a meticulous exploration of the needs and anticipations of users, stakeholders, and the project itself. The inception of the project was rooted in a collaborative process where the group brainstormed ideas and identified prevalent gaps in healthcare accessibility, particularly in herbal remedy guidance. Employing diverse methodologies like surveys, interviews, and extensive market research, we systematically collected requirements, delving into user preferences, stakeholder expectations, and project objectives. Subsequently, the collated requirements underwent rigorous scrutiny, considering factors like significance, viability, and alignment with our project's overarching goals. This meticulous analysis laid the groundwork for subsequent development stages, ensuring that the project was built on a foundation deeply rooted in user needs, stakeholder expectations, and project feasibility.

### 4.2.1 Flow



**Figure 3. Flow Chart**

#### 4.2.2 Use Cases



**Figure 4. Use Cases**

#### 4.3 Requirements Documentation

The requirements analysis phase was followed by the documentation of the project. A detailed software requirements specification (SRS) document was created to capture and describe the functional and non-functional requirements of CuraHerb. The SRS document provided a comprehensive reference for the development team, outlining the system's scope, objectives, and functionalities. It also included use case diagrams, user interface mock ups, data

flow diagrams, and other relevant documentation to support the understanding and implementation of the system.

#### **4.3.2 Purpose**

The purpose of requirements documentation in the development of CuraHerb is to provide a comprehensive and detailed representation of the functional and non-functional requirements of the system. It serves as a reference and communication tool between the development team, stakeholders, and users, ensuring a clear understanding of the system's scope, objectives, and functionalities. The documentation captures stakeholder needs, defines the system's scope and objectives, details functional and non-functional requirements, supports development and testing processes, enables stakeholder agreement, and facilitates change management.

#### **4.3.3 Goals**

The goal of requirements documentation in the development of CuraHerb is to establish a shared understanding and agreement among stakeholders and the development team regarding the desired functionality and behavior of the system. The documentation aims to accurately capture and document the requirements, ensuring that they are comprehensive, clear, and well-defined. By providing a detailed and structured description of the system's requirements, the goal is to guide the development team in building a system that fulfills the needs, expectations, and quality standards of stakeholders and users. The documentation also facilitates effective communication, supports development and testing processes, and acts as a basis for managing changes throughout the development lifecycle.

#### **4.3.4 Project Scope**

CuraHerb aims to develop an intelligent herbal plant recognition and medication guide system. The project focuses on creating a user-friendly platform where users can capture or upload images of herbal plants for accurate identification. The system will provide a comprehensive herbal plant database with detailed information on each recognized plant, including medicinal properties, dosages, precautions, and potential side effects. CuraHerb will also feature a personalized medication guide and user-friendly interface. The project includes the development of mobile applications for iOS and Android platforms, as well as a web-based interface.

## Users

are encouraged to consult healthcare professionals for specific medical advice and treatment.

### 4.3.5 Success Criteria

The success of CuraHerb will be measured based on the achievement of the following criteria.

- The success criterion is a high recognition accuracy rate of at least 95%.
- The success criterion is a database that contains information on a substantial number of commonly used herbal plants.
- A user satisfaction rate of at least 80% based on feedback, surveys, and user ratings.
- The ability to guide users effectively, minimizing the risk of improper usage and potential side effects.

### 4.3.6 User Characteristics

CuraHerb is meticulously crafted to serve a diverse spectrum of users, acknowledging varying levels of technical expertise and familiarity with herbal medicine. The system prioritizes exclusivity by accommodating three distinct user levels, each tailored to specific needs and knowledge depth.

#### User Levels

General Users	<ul style="list-style-type: none"><li>➤ Priority Level- Introduction to Herbal Medicine</li><li>➤ Accessible Features- Simplified interfaces for easy navigation, detailed plant information, and basic usage guidelines. Aimed at providing reliable introductory knowledge on herbal plant identification and uses.</li></ul>
---------------	---

Herbal Enthusiasts	<ul style="list-style-type: none"> <li>➤ Priority Level- Intermediate Understanding</li> <li>➤ Accessible Features- In-depth plant profiles, advanced search functionalities, and detailed explanations of medicinal properties. Geared towards users seeking a deeper understanding of herbal medicine and its applications.</li> </ul>
Healthcare Professionals	<ul style="list-style-type: none"> <li>➤ Priority Level- Professional Reference</li> <li>➤ Accessible Features- Comprehensive databases, scholarly references, and integration with medical literature. Designed as a robust reference tool for healthcare professionals to supplement their expertise and offer guidance to patients.</li> </ul>

CuraHerb's user-centric design aims to empower individuals at every level of familiarity with accessible features tailored to their specific needs. Whether you're an individual curious about herbal remedies, an enthusiast seeking deeper knowledge, or a healthcare professional requiring a reliable referee

#### **4.3.7 Mandated Constraints**

CuraHerb operates within specific mandated constraints that are essential for its development and use. These constraints include legal and ethical considerations that require the system to comply with applicable laws, regulations, and ethical guidelines regarding the provision of medical information, data privacy, and user consent. Another mandated constraint is data privacy and security. The system must adhere to strict measures to protect user information, ensuring that personal data and uploaded images are stored securely, and access to user data is limited to authorized personnel only.

Accuracy of information is also a crucial constraint, requiring CuraHerb to strive for accurate and up-to-date information about herbal plants and their medicinal properties. Regular verification and updates of the herbal plant database are necessary to maintain the accuracy and reliability of the information provided.

#### **4.3.8 Functional Requirements**

##### **Image Recognition**

The system should have the capability to accurately recognize herbal plants from images uploaded or captured by users.

##### **Plant Identification**

CuraHerb should provide users with the correct identification of recognized herbal plants, including their botanical names, common names, and related information.

##### **Herbal Plant Database**

The system should maintain a comprehensive database of herbal plants, containing detailed information about each plant's medicinal properties, traditional uses, dosages, precautions, and potential side effects.

##### **Medication Guide**

CuraHerb should offer a medication guide component that provides users with clear instructions on the preparation methods, dosages, and precautions for each identified herbal plant.

##### **User Profiles**

The system should allow users to create personalized profiles, enabling them to save favorite plants, access their search history, and receive tailored recommendations.

##### **Search and Filtering**

CuraHerb should provide users with search functionality, allowing them to search for herbal plants based on their botanical names, common names, or medicinal properties. The system should also offer filtering options to refine search results.

## **User Feedback**

Curaherb has a user feedback feature where it allow the user to contribute and allow users to leave reviews and ratings for herbal products and services offered on the platform. This not only provides valuable feedback to other users but also helps in improving product offerings based on user preferences.

### **4.3.7 Non-Functional Requirements**

#### **Usability**

CuraHerb has a user-friendly interface that is intuitive, easy to navigate, and visually appealing platform that effectively serves its target audience interested in holistic wellness and herbal remedies. It should provide a seamless and engaging user experience

#### **Performance**

The system should be responsive and provide quick results for image recognition and plant identification. It should have minimal loading times and handle concurrent user requests effectively.

#### **Security**

CuraHerb should implement robust security measures to protect user data and ensure the confidentiality and integrity of personal information. It should include mechanisms for secure authentication and data encryption.

#### **Scalability**

The system should be designed to handle increasing user demands and accommodate a growing database of herbal plants. It should be scalable to support a large user base and expandable to incorporate new plant entries.

#### **Compatibility**

CuraHerb should be compatible with major mobile platforms (iOS and Android) and provide a web-based interface that works across different devices and web browsers.

## **Reliability**

The system should operate reliably with minimal downtime, ensuring continuous availability for users. It should have mechanisms in place to handle errors and exceptions gracefully.

## **Data Integrity**

The system should ensure the accuracy and integrity of the data stored in the herbal plant database. It should employ appropriate data validation and verification techniques.

## **4.4 Design of Software Systems, Products, and/or Processes**

The design phase of CuraHerb encompasses the creation and planning of the software systems, products, and/or processes that contribute to the functionality and effectiveness of the system. The design aims to ensure that CuraHerb is well-structured, user-friendly, performance secure, and capable of meeting the intended objectives and functionalities outlined in the requirements documentation.

### **4.4.2 Process**

The process within the design of CuraHerb outlines the steps, activities, and tasks involved in creating a successful and functional software product. The process ensures that the development team follows a systematic approach, enabling efficient collaboration and the achievement of project goals. The process for the design and development of CuraHerb includes the following key stages:

#### **Requirements Gathering and Analysis**

In this stage, the development team conducts interviews, surveys, and research to gather and analyze the requirements for the system. This involves understanding user needs, identifying key functionalities, and capturing the desired outcomes of CuraHerb.

#### **Planning and Sprint Backlog Creation**

The development team, along with the product owner, prioritizes the requirements and creates a backlog of user stories or tasks. The team then plans

the sprints, defining the scope and objectives of each sprint.

### **Sprint Execution**

The development team works on the tasks identified in the sprint backlog. During the sprint, daily scrum meetings are held to discuss progress, address challenges, and ensure alignment within the team.

## **4.5 Design of Software Systems, Products, and/or Processes**

The design phase of CuraHerb encompasses the creation and planning of the software systems, products, and/or processes that contribute to the functionality and effectiveness of the system. The design aims to ensure that CuraHerb is well-structured, user-friendly, performance secure, and capable of meeting the intended objectives and functionalities outlined in the requirements documentation.

### **4.5.2 Process**

The process within the design of CuraHerb outlines the steps, activities, and tasks involved in creating a successful and functional software product. The process ensures that the development team follows a systematic approach, enabling efficient collaboration and the achievement of project goals. The process for the design and development of CuraHerb includes the following key stages:

#### **Requirements Gathering and Analysis**

In this stage, the development team conducts interviews, surveys, and research to gather and analyze the requirements for the system. This involves understanding user needs, identifying key functionalities, and capturing the desired outcomes of CuraHerb.

#### **Planning and Sprint Backlog Creation**

The development team, along with the product owner, prioritizes the requirements and creates a backlog of user stories or tasks. The team then plans the sprints, defining the scope and objectives of each sprint.

#### **Sprint Execution**

The development team works on the tasks identified in the sprint backlog. During the sprint, daily scrum meetings are held to discuss progress, address challenges, and ensure alignment within the team.

## **Testing and Quality Assurance**

Testing activities, such as unit testing, integration testing, and user acceptance testing, are carried out throughout the development process to ensure the quality and reliability of the software.

## **Sprint Review and Retrospective**

At the end of each sprint, a review session is conducted to demonstrate the completed work to stakeholders. Additionally, a retrospective meeting is held to reflect on the sprint process and identify areas for improvement.

## **Incremental Delivery and Iterative Development**

After each sprint, a potentially shippable increment of the software is delivered. This allows for early feedback from users and stakeholders, which can be incorporated into subsequent sprints to improve the system.

## **Continuous Integration and Deployment**

The developed features are continuously integrated into the system, ensuring compatibility and proper functioning. Deployment processes are streamlined to facilitate the release of new versions or updates.

## **Ongoing Maintenance and Enhancement**

Once the initial development phase is completed, the system goes into a maintenance and enhancement phase, where ongoing updates, bug fixes, and improvements are implemented based on user feedback and evolving requirements.

Throughout the process, communication and collaboration among the development team, stakeholders, and users are crucial. Regular progress updates, feedback sessions, and iterative development cycles may be incorporated to ensure alignment with user expectations and to adapt to evolving needs.

## **Incremental Delivery and Iterative Development**

After each sprint, a potentially shippable increment of the software is delivered. This allows for early feedback from users and stakeholders, which can be incorporated into subsequent sprints to improve the system.

## **Continuous Integration and Deployment**

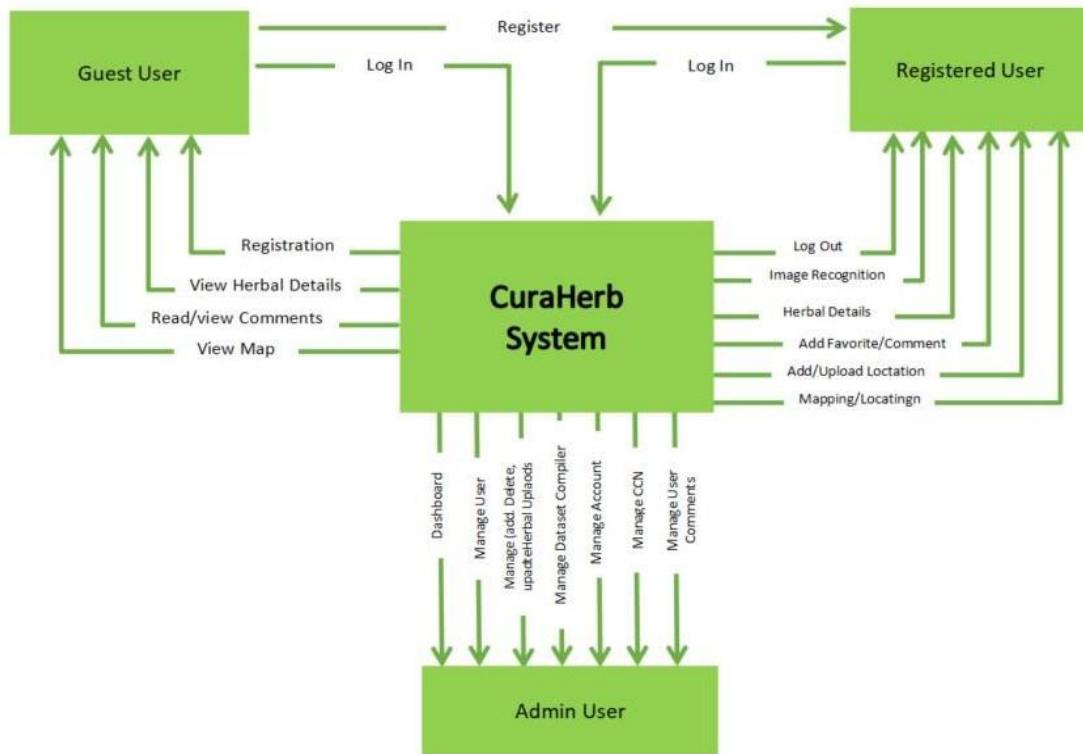
The developed features are continuously integrated into the system, ensuring compatibility and proper functioning. Deployment processes are streamlined to facilitate the release of new versions or updates.

## **Ongoing Maintenance and Enhancement**

Once the initial development phase is completed, the system goes into a maintenance and enhancement phase, where ongoing updates, bug fixes, and improvements are implemented based on user feedback and evolving requirements.

Throughout the process, communication and collaboration among the development team, stakeholders, and users are crucial. Regular progress updates, feedback sessions, and iterative development cycles may be incorporated to ensure alignment with user expectations and to adapt to evolving needs.

#### 4.5.2.1 Data Flow Diagram (DFD)



**Figure 5. Context Diagram**

The figure above, shows how the users interact with the CuraHerb system. The guest user will only access Herbal details from herbal library, read comments, and view the map. Overall it can only view the system. If the user already registered an account, it can now access the Image recognition, locate herbal plants, upload new locations, add favorites from herbal library, and can leave comments on herbal details and map.

#### 4.5.2.2 User Interface Application

The interface application within the design of CuraHerb refers to the user-facing component that allows users to interact with the system. It encompasses the design and development of the graphical user interface (GUI) through which users can capture or upload images of herbal plants, receive identification results, access detailed plant information, and utilize the medication guidance feature.

### **4.5.3 User Input**

The process begins when the user enters the desired search query, which may include symptoms, keywords, or specific requirements related to their health condition or herbal remedies they are seeking.

#### **Image Capture/Upload**

The interface application provides users with the capability to capture images using their device's camera or upload images from their local storage. This functionality enables users to submit images of herbal plants for identification.

#### **User Input and Interaction**

The interface application allows users to input additional information, such as text-based search queries, to enhance the accuracy of plant identification. Users can also interact with the system through buttons, menus, and other interface elements to navigate the application and access various features.

#### **Image Recognition Feedback**

The interface application provides real-time feedback to users during the image recognition process. This includes progress indicators, loading animations, or notifications to inform users about the status of the identification process.

#### **Identification Results Display**

Once the image recognition process is complete, the interface application displays the identification results to the user. This includes presenting the recognized herbal plant species along with relevant information, such as botanical names, common names, and descriptions.

#### **Plant Information Display**

The interface application allows users to view detailed information about recognized herbal plants. This includes presenting medicinal properties, traditional uses, dosages, precautions, and potential side effects in a clear and organized manner.

## **Medication Guidance**

The interface application provides users with clear and concise medication guidance for each identified herbal plant. It presents instructions on preparation methods, recommended

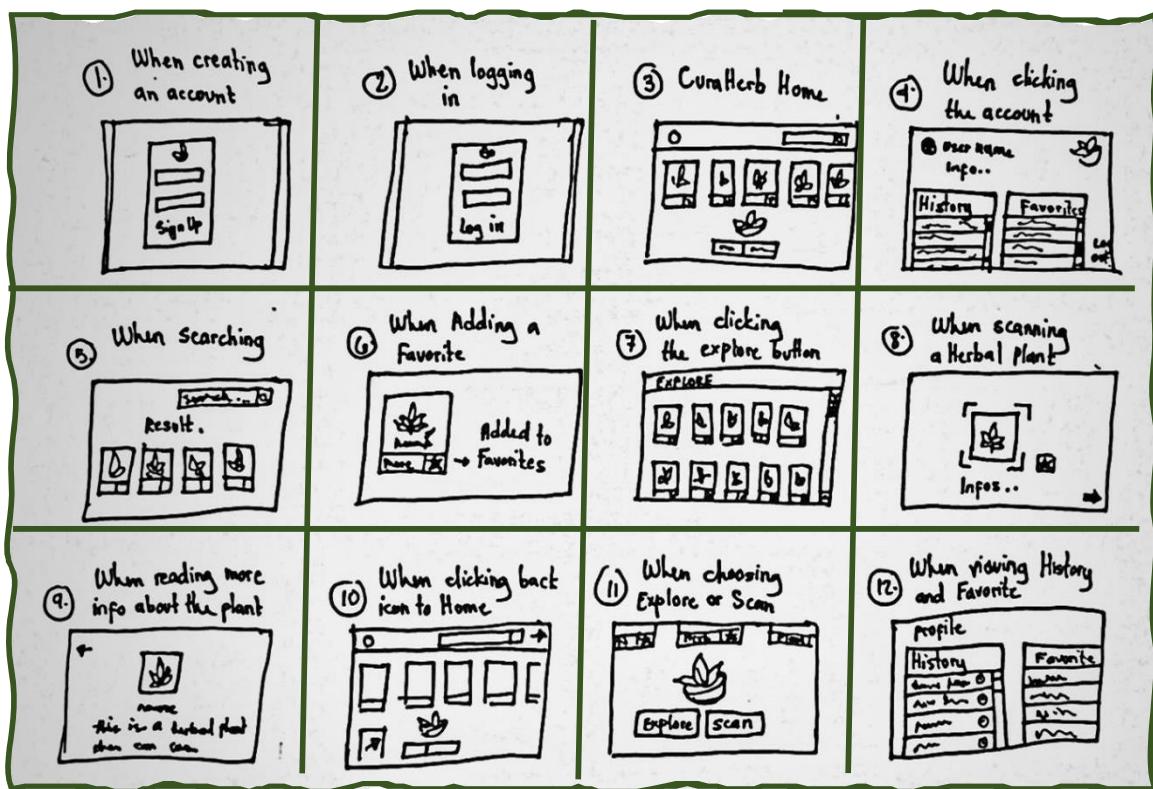
dosages, and precautions for safe and effective usage. This information is displayed in an easily understandable format to assist users in utilizing herbal plants responsibly.

### Personalization and User Profile Management

The interface application offers personalization features, allowing users to create profiles, save favorite plants, and access their search history. This functionality enables a personalized experience and facilitates easy retrieval of previously identified plants.

#### 4.5.4 Storyboard

The storyboard is made up of the timelines of each module on the system.



**Figure 6. Storyboard**

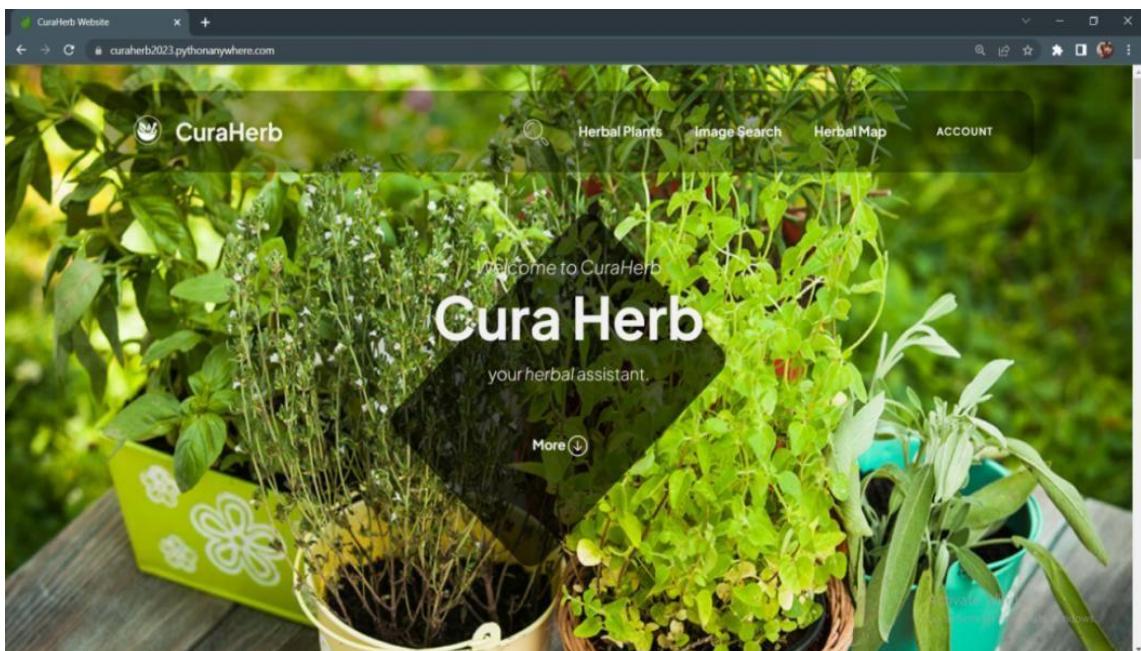
The storyboard visually depicts the user's journey through the various features and functionalities of CuraHerb. It provides a clear understanding of how the system meets the user's needs and enhances their experience in discovering and utilizing herbal remedies.

## CHAPTER V

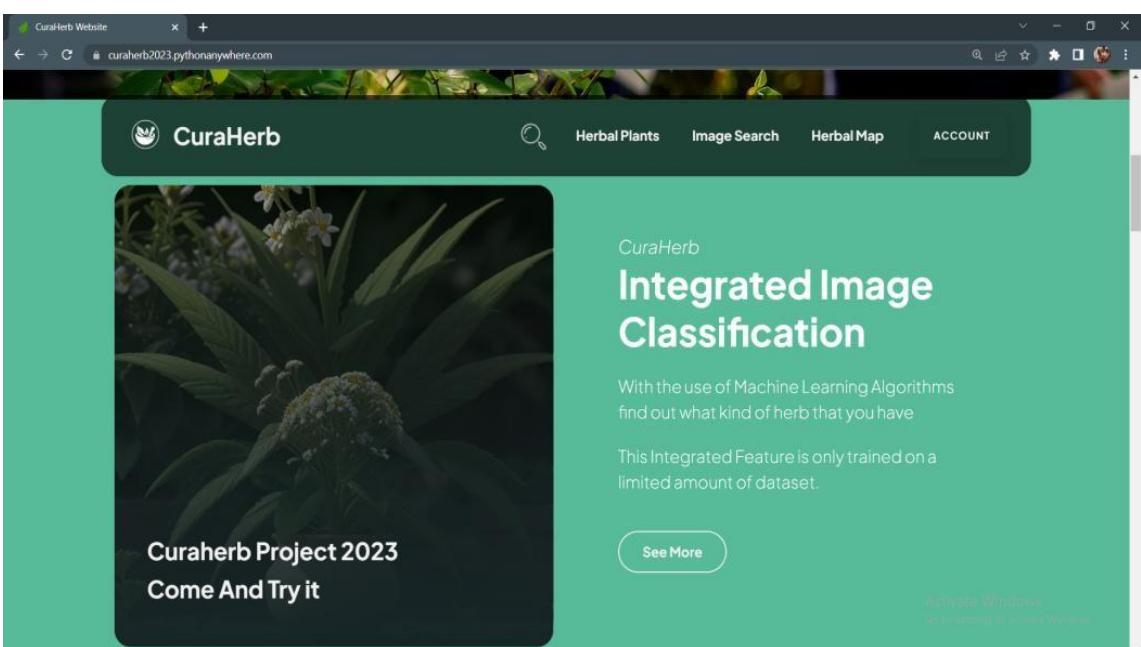
### DEVELOPMENT AND TESTING

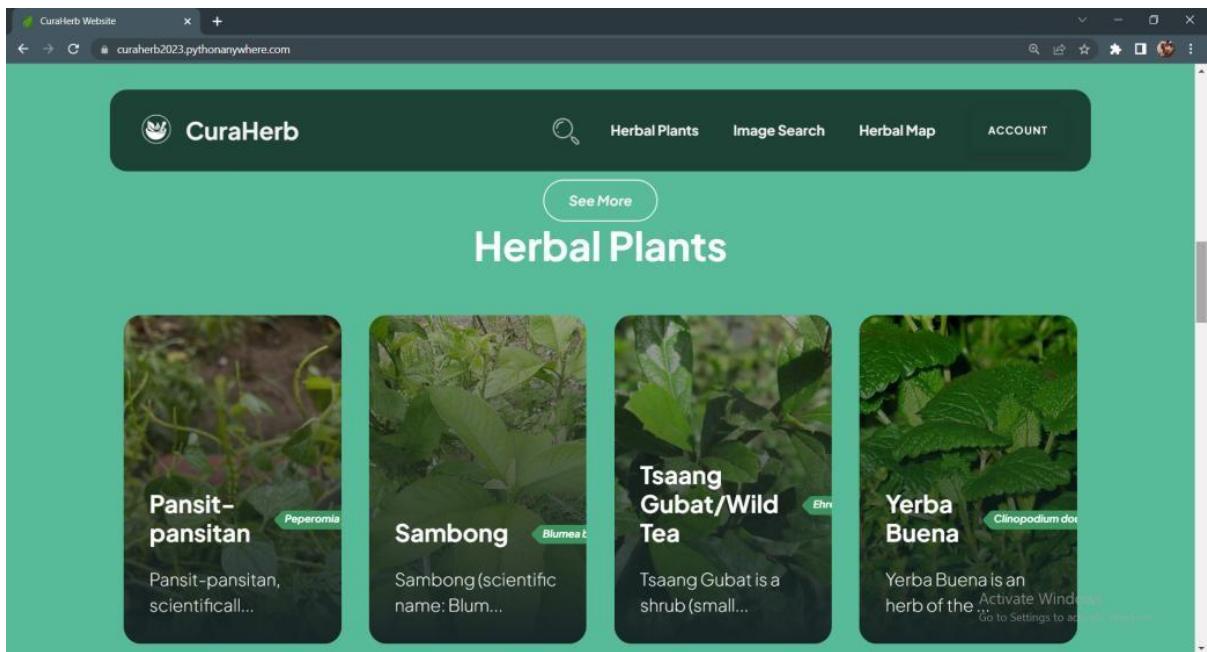
#### 5.1 Description of Prototype

##### Visitor/ User (Logged In) - Homepage

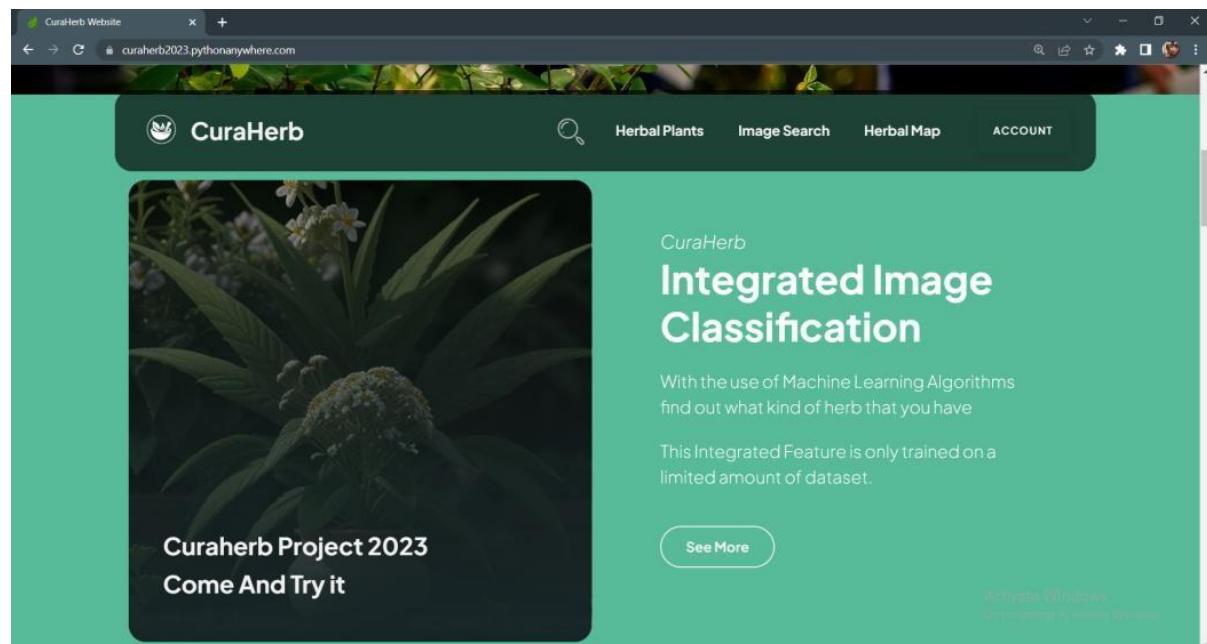


At the homepage, users can start scrolling up and down to explore the system features and functionalities

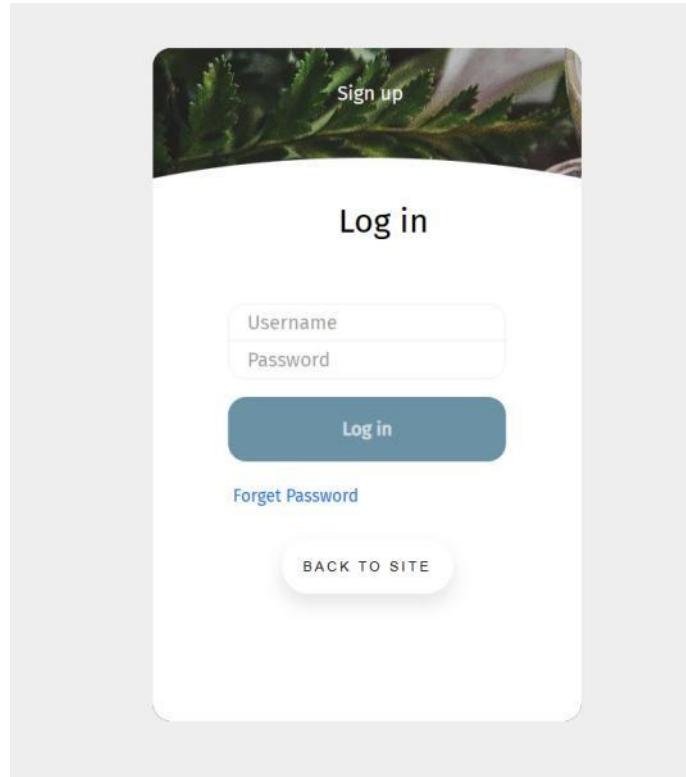




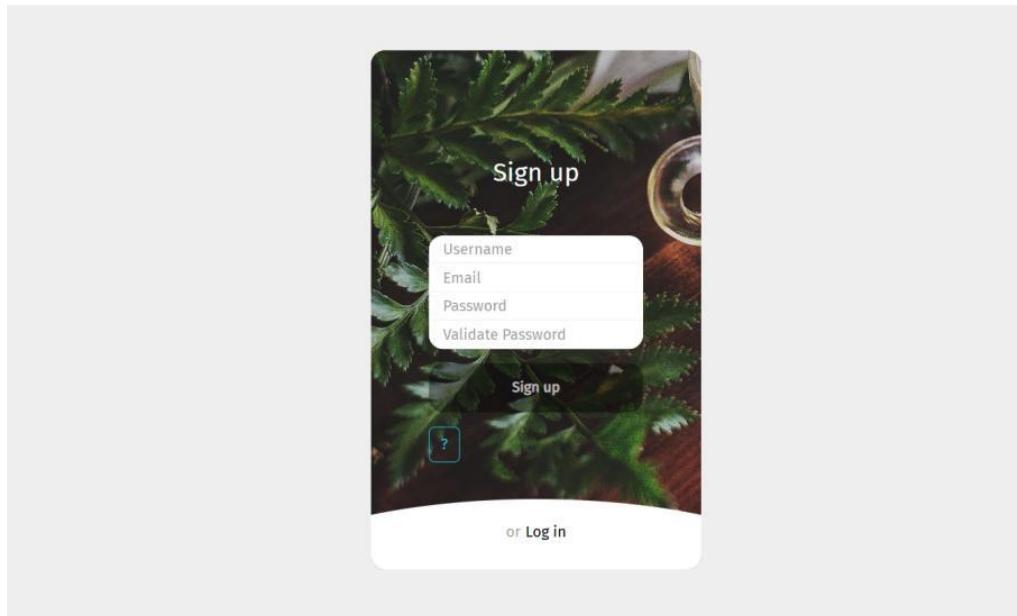
Scrolling down, users can explore the 'Sampung Halamang Gamot' approved by the DOH, with detailed information on each herbal plant.



## Log In and Sign Up



The system will allow the users to log in.



The system shall allow the users to create an account.

## Visitor/ User (Logged In) - Herbal Plants Library

The screenshot shows a grid of 10 cards, each representing a different herb. The cards are arranged in two rows of five. Each card includes a small image of the plant, its name, its scientific name, and a brief description.

- Pansit-pansitan / CuraHerb**: Scientific name: Peperomia pellucida. Description: A small, green, shrub-like plant.
- Sambong**: Scientific name: Blumea balsamifera. Description: An amazing medicinal plant.
- Gubat/Wild Tea**: Scientific name: Ehretia microphylla Lam. Description: A shrub (small tree) that grows abundantly.
- Yerba Buena**: Scientific name: Mentha spicata. Description: An aromatic plant used as herbs.

- Akapulko / Acapulco**: Scientific name: Cassia alata. Description: A shrub found throughout the Philippines.
- Aloe Vera**: Scientific name: Cassia alata. Description: A succulent plant with thick, fleshy leaves that contain a gel-like substance.
- Lagundi**: Scientific name: Vitex negundo. Description: A shrub that grows in t
- Niyog-Niyogan**: Scientific name: Quisqualis indica. Description: A woody vine with fragrant flowers.

This is where the 10 DOH approved herbal plants was being displayed.

## User (Logged In) - Favorites

The screenshot shows a section titled "FAVOURITES" displaying three favorite herbs. Each entry includes a small image of the plant, its name, its scientific name, and two buttons: "View Herb" and "Remove".

- Sambong - Blumea balsamifera**
- Tsaang Gubat/Wild Tea - Ehretia microphylla Lam.**
- Niyog-Niyogan - Quisqualis indica**

## UPLOADS

Below the "UPLOADS" heading, there are four columns: Title, Latitude, Longitude, and Timestamp. There is also a "Image Attachment" section.

**Tsaang Gubat/Wild Tea**

**Ehretia microphylla Lam.**

Tsaang Gubat is a shrub (small tree) that grows (from 1 to 5 meters) abundantly in the Philippines. In folkloric medicine, the leaves have been used as a disinfectant wash during child birth, as cure for diarrhea, as tea for general good health and because Tsaang Gubat has high fluoride content, it is used as a mouth gargle for preventing tooth decay. Research and test now prove its efficacy as an herbal medicine. Aside from the traditional way of taking Tsaag Gubat, it is now available commercially in capsules, tablets and tea bags.

**Medical Conditions**

Individuals with pre-existing health conditions should consult a healthcare professional before incorporating herbs into their routine.

**Recently Uploaded**

**Tsaang Gubat/Wild Tea**  
Nov 11, 2023, 4:51 a.m.

Users can add their favorites after he/she viewed a certain plants information. Only users with accounts can add their favorite herbal plants.

### Visitor/ User (Logged In) - Herbal Plant Details

**Similar Property**  
[Sambong](#)

**Discover Different Herb**  
[Yerba Buena](#)

Add to Favorite

This screenshot shows a web browser displaying a page from the 'CuraHerb Website'. The URL in the address bar is [curaherb2023.pythonanywhere.com/Herbs/](http://curaherb2023.pythonanywhere.com/Herbs/). The main content area features a green header with the title 'Pansit-pansitan' and a sub-section title 'Peperomia pellucida'. Below the title is a detailed description of the plant, mentioning its native status to the Philippines, various names like Ulasiman-bato, and its use in traditional medicine. To the right of the main content, there is a sidebar titled 'Medical Conditions' with a note about consulting a healthcare professional before incorporating herbs into their routine. A 'Recently Uploaded' section is also visible on the right.

This is how the system display each herbal plant details.

### Visitor/ User (Logged In) - Herbal Plant Details & Testimonials

This screenshot shows a web browser displaying a page from the 'CuraHerb Website'. The URL in the address bar is [curaherb2023.pythonanywhere.com/Herbs/](http://curaherb2023.pythonanywhere.com/Herbs/). The main content area features a green header with the title 'consectetuer lorem' and a sub-section title 'Potential Side Effects/Allergic reaction'. Below the title is a detailed description of potential side effects. To the right of the main content, there is a sidebar titled 'Medical Conditions' with a note about consulting a healthcare professional before incorporating herbs into their routine. A 'Recently Uploaded' section is also visible on the right. At the bottom left, there is a testimonial box for 'admin' with the email 'admin@gmail.com'.

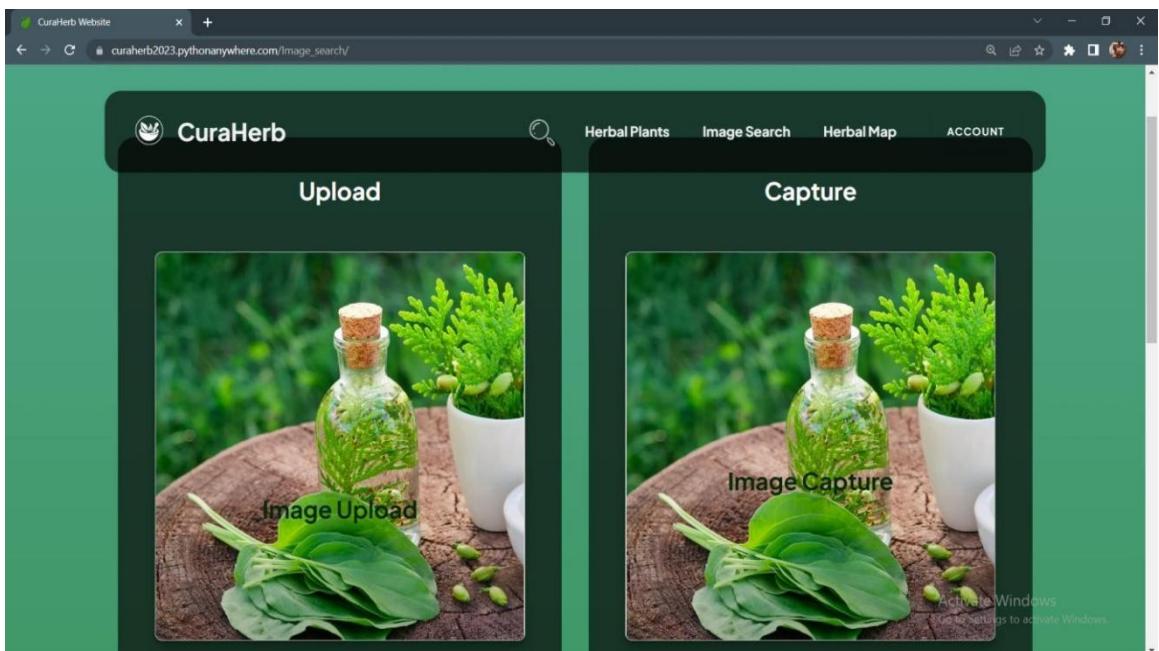
The screenshot shows a web browser window for the 'CuraHerb Website'. The URL is [curaherb2023.pythonanywhere.com/Herbs/](http://curaherb2023.pythonanywhere.com/Herbs/). The main content area displays a testimonial box with placeholder text: 'sodales magna convallis porta bibendum si massa ad volutpat efficitur praesent quisque class ex maecenas tincidunt phasellus pellentesque elementum habitasse diam auctor sit nascetur duis vehicula sed netus taciti viverra torquent lobortis pede neque vestibulum aliquet habitant rhoncus pretium sociosqu hac tristique montes hendrerit nec dictum ultricies tempor dapibus risus sollicitudin facilisi enim dis justo bitch'. Below the testimonial is a red 'Delete' button. To the right, there's a small image of tea leaves with the word 'Tea' and the date 'Nov 11, 2023, 4:51 a.m.'. A 'Comment' input field and a 'Submit' button are located below the testimonial. On the right side, there's a 'Reference' section with links to [businessdiary.com.ph](#), [www.rmpedia.com](#), and [medicalhealthguide](#). A 'Tools' section includes links to 'Herb Classification' and 'Interactive Map'. At the bottom right, there are 'Add to Favorites' and 'Add to Favorite' buttons.

The system also allows the users to submit their testiminials on each herbal plant details.

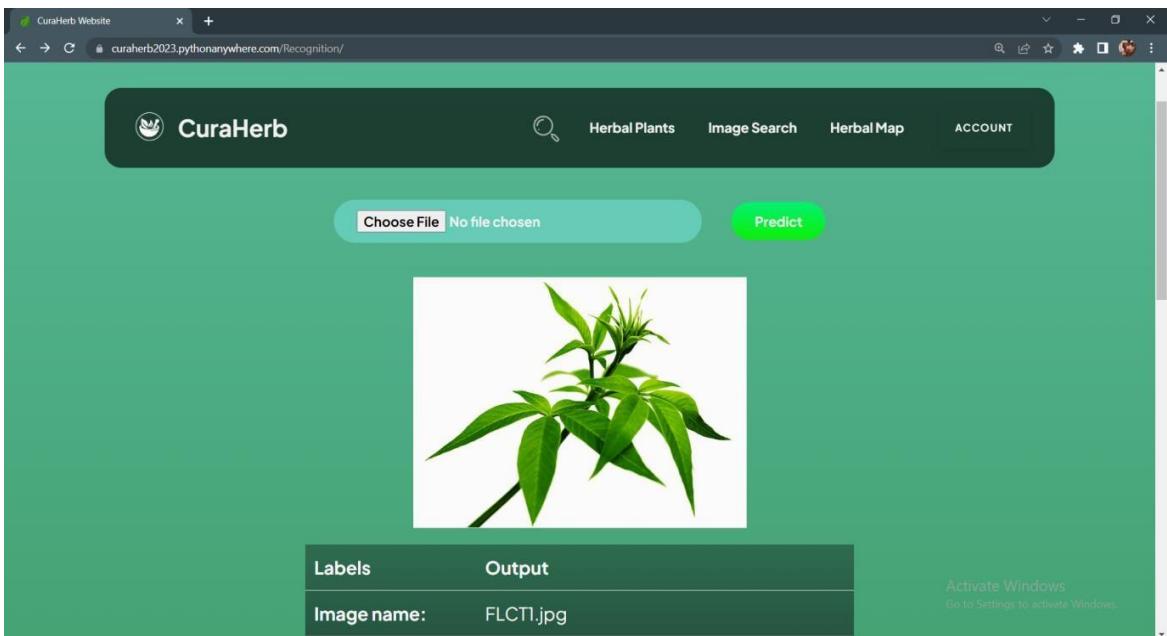
### Visitor/ User (Logged In) - Interactive Herbal Map

The screenshot shows the 'CuraHerb Website' interface. The top navigation bar includes the logo, search icon, 'Herbal Plants', 'Image Search', 'Herbal Map', and 'ACCOUNT' buttons. The main content area features a heading 'Curaherb Interactive Herbal Map' and a sub-instruction: 'Integrated Herbal Map find herbs or herbal stores around you. With added functionalities, found a herb? share it, like and comment on other uploads as well!'. A legend on the left defines three icons: a green circle for 'Herbs', a yellow circle for 'User Uploads', and a blue location pin for 'Shop Marker'. To the right is an interactive map of a coastal region with several green and yellow circular markers indicating herb locations. A scale bar shows distances up to 5 km/3 mi. The map is attributed to 'Leaflet | Data by © OpenStreetMap, under CC-BY'. A watermark at the bottom right reads 'Activate Windows Go to Settings to activate Windows.'

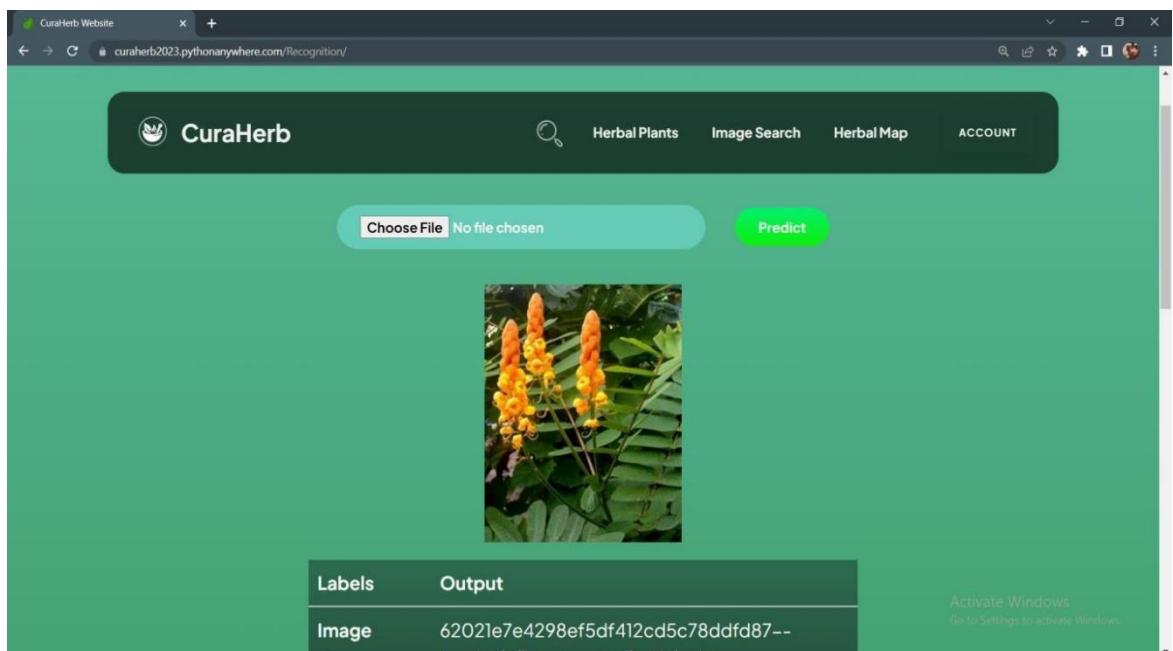
## Visitor/ User (Logged In) - Image Search



The user can upload image or capture the herbal plants using the system to recognize their names and details



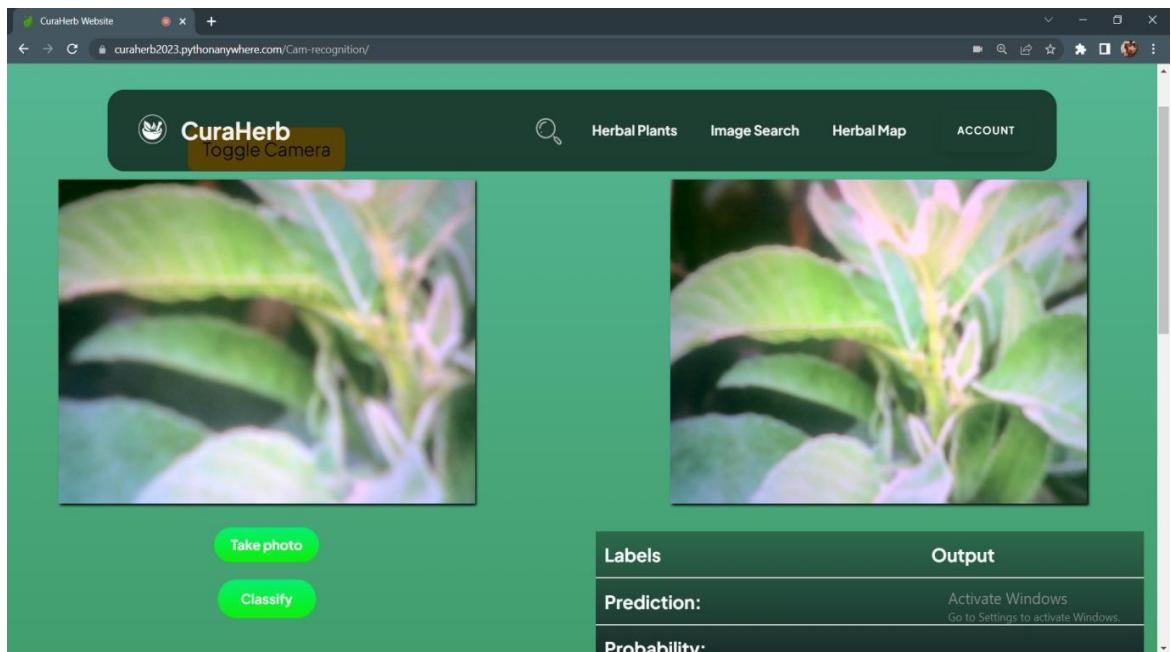
## Visitor/ User (Logged In) - Image Search (Capture)



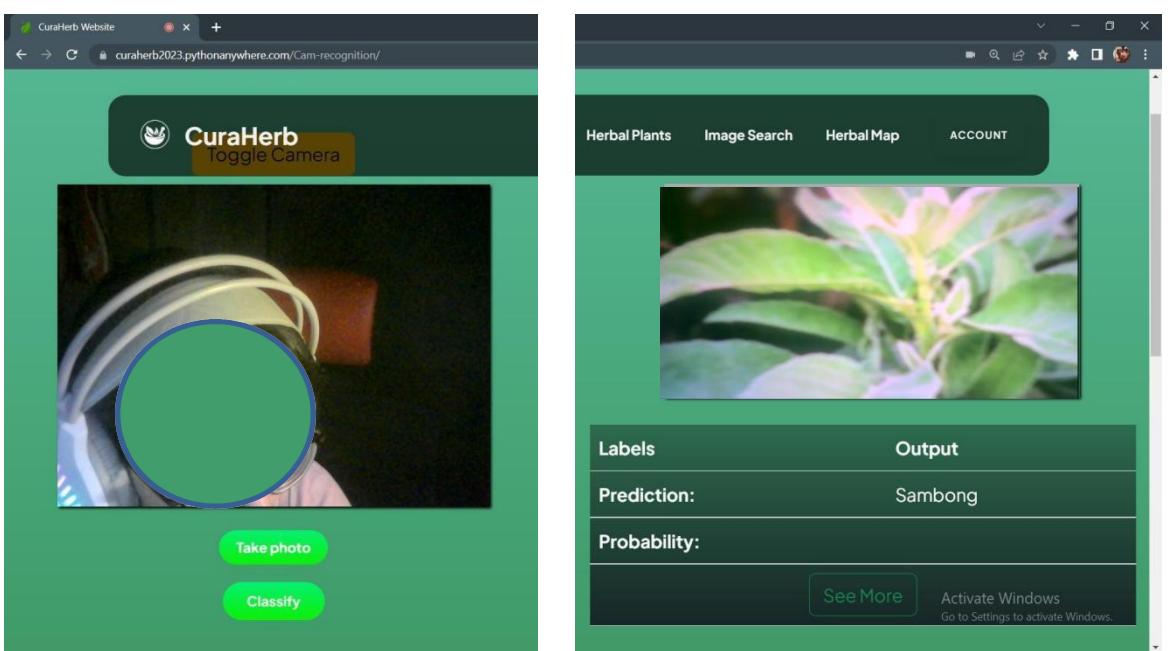
This will be the output of the system after capturing a actual herbal plant.

The screenshot shows two parts of the CuraHerb website. On the left, the results of the image capture are shown, identical to the previous screenshot. On the right, a detailed page for 'Akapulko / Acapulco' is displayed. The page header says 'Department of Health (DOH) Approved Herbal Plants'. It features a large image of the plant with its common names 'Akapulko' and 'Acapulco' overlaid, and the scientific name 'Cassia alata'. A descriptive text box states: 'Akapulko or Acapulco in English is a shrub found throughout the Philippines. It'.

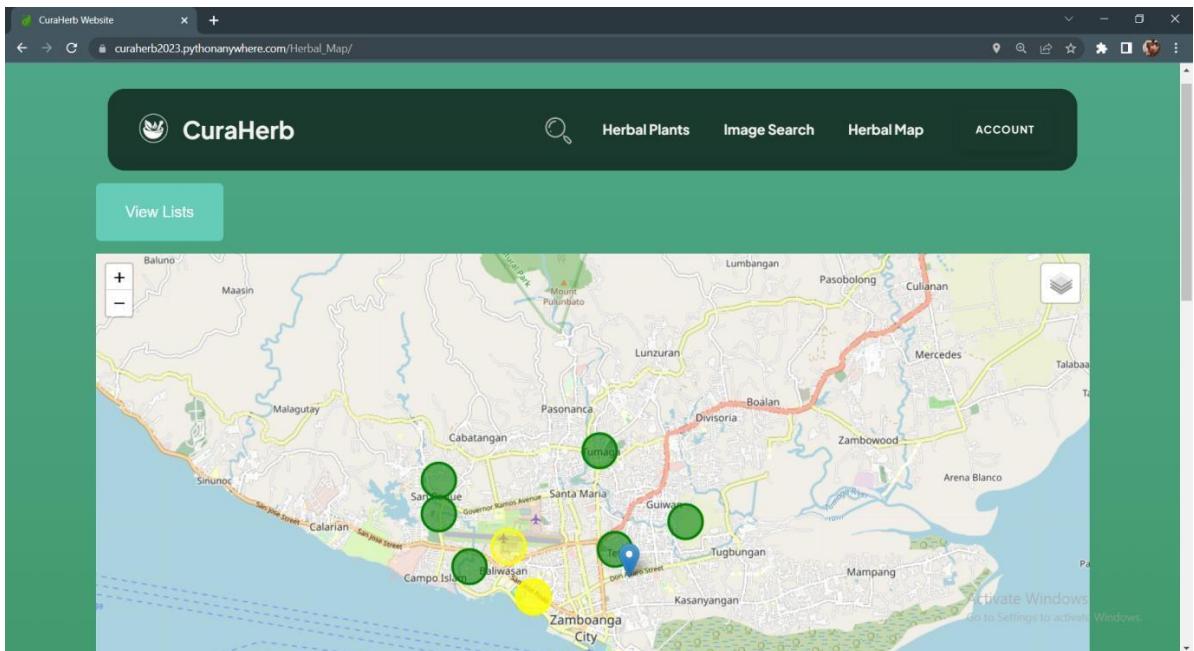
## Visitor/ User (Logged In) - Image Search (Capture)



This will be the output of the system after uploading the existing image of the herbal plants from the user's devices.



## Visitor/ User (Logged In) - Herbal Map



The system shows where can a user locate herbal plants on the map (depends on the locations that was stored to the system)

A screenshot of a web browser displaying the CuraHerb website. The left sidebar is titled 'Herbal List' and lists four entries: 'Pansit-pansitan Admin', 'Sambong Admin', 'Tsaang Gubat/Wild Tea Admin', and 'Yerba Buena Admin'. Each entry is preceded by a small green circular icon. The right side of the screen shows the same map as the previous screenshot, with green circular markers indicating plant locations. A yellow 'X' button is visible at the bottom left of the sidebar.

It shows the list of available herbal plants that user can locate on the system's map

## Admin User - Dashboard

The screenshot shows the Admin User - Dashboard interface. On the left, a sidebar menu includes 'Dashboard' (selected), 'User', 'HERBAL CONTENTS' (with 'Herbal Uploads' and 'User Herbal Comments'), 'HERBAL MACHINE LEARNING' (with 'CNN Model' and 'Dataset Compiler'), and 'MAP & USER UPLOADS' (with 'Interactive Map' and 'User Map Comments'). The main area features a 'Welcome admin!' message with a small plant icon, followed by the 'Curaherb Project' logo. A large 'Model Statistic' section contains a 'Model Statistic Chart' (a line graph showing Validation Loss from 0.1 to 1.0) and a circular gauge indicating '82% Accuracy' for 'amount of herb class'. Other cards show 'Total Herbs Current 8' (with a green upward arrow for 'herb increase'), 'Most Liked Pansit-pansitan' (with a green upward arrow for '2'), 'Most Commented Pansit-pansitan' (with a green upward arrow for '5'), 'Recently added Pansit-pansitan' (with a green upward arrow for '2023-11-11'), and a 'Map Report' card with a 'Weekly Total' of 1.

The system display the view site of the admin dashboard.

## Admin User - System User (List)

The screenshot shows the Admin User - System User (List) page. The sidebar is identical to the dashboard. The main area displays two cards: 'Top Location Contributor' (admin) and 'No. of Active User' (1). Below these is a 'User List' table:

USERNAME	EMAIL	NO. OF UPLOADED LOCATION	NO. OF COMMENTS	NO. OF COMMENTS ON MAP	STATUS
Admin	ADMIN@GMAIL.COM NOT VERIFIED	1	7	2	<a href="#">LOGOUT</a>
Phone	PHONE@GMAIL.COM NOT VERIFIED	0	0	0	<a href="#">LOGOUT</a>
Gegege	GEGEGE@GMAIL.COM NOT VERIFIED	1	0	1	<a href="#">LOGOUT</a>
Efeaf	ONIZUKAREIGEN@GMAIL.COM VERIFIED	0	0	0	<a href="#">Activate Windows</a> Go to Settings to activate Windows.
Tere	TERE@GMAIL.COM	0	0	0	<a href="#">LOGOUT</a>

Admin user can view the list of system users with their number of uploads, testimonials, and comments on the map.

## Admin User - Herbal List

The screenshot shows a web-based administration interface for a herbal database. On the left, a sidebar menu includes 'Dashboard', 'User', and sections for 'HERBAL CONTENTS' (Herbal Uploads, User Herbal Comments), 'HERBAL MACHINE LEARNING' (CNN Model, Dataset Compiler), and 'MAP & USER UPLOADS' (Interactive Map, User Map Comments). The main content area is titled 'Herb List' and contains a table with columns: NAME, NO. OF COMMENTS, NO. OF BOOKMARKS, IMAGE ATTACHMENT, and ACTIONS. The table lists eight entries:

NAME	NO. OF COMMENTS	NO. OF BOOKMARKS	IMAGE ATTACHMENT	ACTIONS
Pansit-pansitan	5	2	HERBS/54BF697994555302DC7A653C25157D45.JPG	⋮
Sambong	1	1	HERBS/49EF60B539FEBF24E6A860ABD1F1191.JFIF	⋮
Tsaang Gubat/Wild Tea	0	0	HERBS/EHRETIAMICROPHYLLA-WEB.JPG	⋮
Yerba Buena	0	0	HERBS/I7-REMARKABLE-BENEFITS-OF-YERBA-BUENA.JPG	⋮
Akapulko / Acapulco	0	0	HERBS/AKA.JPG	⋮
Aloe Vera	0	0	HERBS/ALOE-VERA-PLANT-OUTSIDE-JPG-1522875135.JPG	⋮
Lagundi	1	0	HERBS/LAGUNDI2.JPG	⋮
Niyog-Niyogan	0	0	HERBS/NIYOG-NIYOGAN.JPG	⋮

This shows the list of herbal plants that is stored to the system.

## Admin User - User Herbal Comments

The screenshot shows a web-based administration interface for managing user comments on herbal plants. The sidebar menu is identical to the previous dashboard. The main content area is titled 'User Herb Comments' and displays a table with columns: #, USERNAME, PROFANITY, HERB NAME, OPTION, and COMMENT. The table lists seven comments from a user named 'admin' on various herbs. The 'PROFANITY' column highlights two comments as containing profanity.

#	USERNAME	PROFANITY	HERB NAME	OPTION	COMMENT
1	admin	NO PROFANITY DETECTED	Pansit-pansitan	⋮	gege
2	admin	NO PROFANITY DETECTED	Pansit-pansitan	⋮	ge
3	admin	NO PROFANITY DETECTED	Pansit-pansitan	⋮	gegegeg
4	admin	NO PROFANITY DETECTED	Sambong	⋮	gege
5	admin	CONTAINS PROFANITY	Pansit-pansitan	⋮	balls
6	admin	CONTAINS PROFANITY	Pansit-pansitan	⋮	
7	admin	CONTAINS PROFANITY	Lagundi	⋮	Balls

Admin also can view the users comments on each herbal plants.

## Admin User - Model Statistics

The screenshot shows a web application interface for managing model statistics. On the left, a sidebar menu includes options like Dashboard, User, HERBAL CONTENTS, Herbal Uploads, User Herbal Comments, HERBAL MACHINE LEARNING, CNN Model, Dataset Compiler, MAP & USER UPLOADS, Interactive Map, and User Map Comments. The main content area is titled 'Model Statistics' and displays the following data:

Model Statistics	
3000 Total Data	
31 Total Classes	82 Accuracy
Accuracy	Accuracy on Training Data
1.0	
Val Accuracy	Accuracy on Separate Validation Data
0.82	
Classes	Number of Classes Supported
31	
Data Length	Number of Images Used
3000	

To the right, there is a 'File input' section with fields for MODEL FILE, TRAINING ACCURACY, VALIDATION ACCURACY, VALIDATION LOSS, CLASSES AMOUNT, and CLASS LENGTH/IMAGE LENGTH. A green 'Upload' button is located at the bottom of this section. A note at the bottom right says 'Activate Windows Go to Settings to activate Windows'.

This is where the admin user view the model statistics.

## Admin User - Model Training

The screenshot shows a web application interface for model training. The sidebar menu is identical to the previous page. The main content area is titled 'Training' and contains fields for uploading files:

UPLOAD A VALID LABEL.NPY FILE  
 No file chosen

UPLOAD A VALID 'DATASET'.NPY HERB FILE  
 No file chosen

Load Data

A note at the bottom right says 'Activate Windows Go to Settings to activate Windows'.

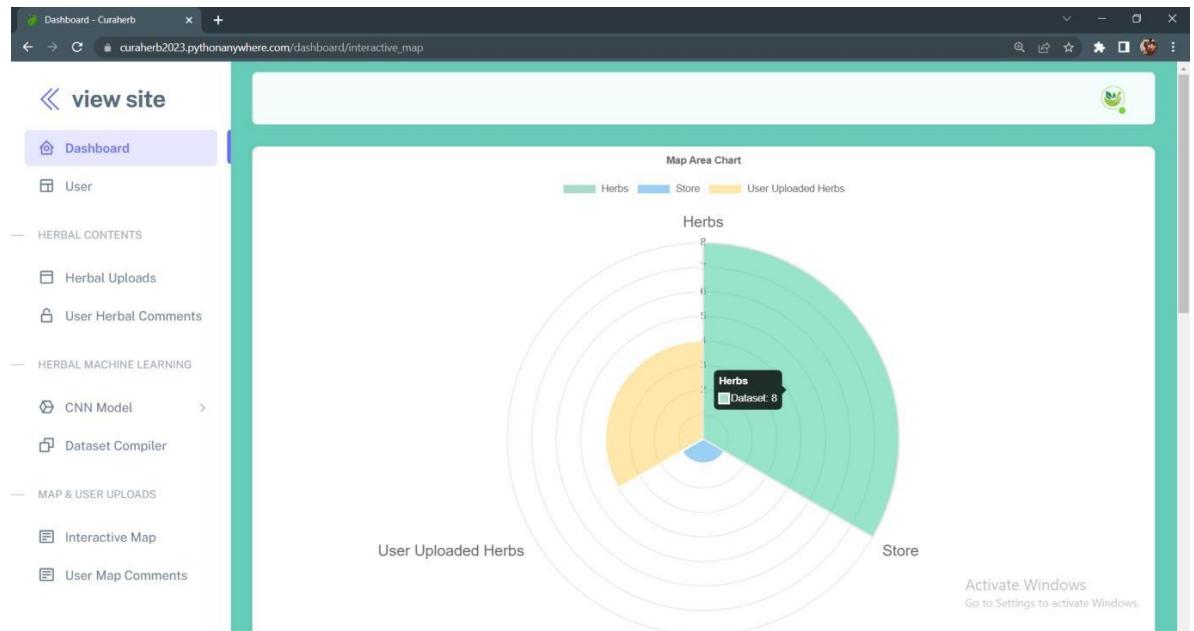
And also the admin user model training.

## Admin User - Dataset Compiler

The screenshot shows a web-based dashboard for dataset compilation. On the left, a sidebar menu includes options like Dashboard, User, HERBAL CONTENTS (Herbal Uploads, User Herbal Comments), HERBAL MACHINE LEARNING (CNN Model, Dataset Compiler), and MAP & USER UPLOADS (Interactive Map, User Map Comments). The main area has two main sections: 'Selected Images' (empty) and 'Dataset Preparation'. In 'Dataset Preparation', there's a 'HERB CLASS NAME:' input field with 'Class Name' and 'Please Enter Correct Class Name' placeholder text, a 'MULTIPLE FILES INPUT' section with 'Choose Files' and 'No file chosen' buttons, and a green 'Prepare Dataset' button. Below this is a 'Dataset List' section with a 'Compile New Dataset' button, a table header for 'CLASS NAME' and 'NUMBER OF IMAGES', and a row for 'Herbs' with 'Dataset: 8'. A watermark 'Activate Windows Go to Settings to activate Windows.' is visible in the bottom right.

This is where the admin can view the data set compiler.

## Admin User - Interactive Map (Area Chart)



The admin can view the area map for the system's interactive map.

## Admin User - Interactive Map (Table)

The screenshot shows a web-based dashboard titled "Dashboard - Curaherb". On the left, there is a sidebar with navigation links: "view site", "Dashboard", "User", "HERBAL CONTENTS" (with "Herbal Uploads" and "User Herbal Comments" sub-links), "HERBAL MACHINE LEARNING" (with "CNN Model" and "Dataset Compiler" sub-links), and "MAP & USER UPLOADS" (with "Interactive Map" and "User Map Comments" sub-links). The main content area is titled "Map Table" and contains a table with the following data:

NAME	LONGITUDE	LATITUDE	TYPE	ACTIONS
Pansit-pansitan	122.096545	6.925598	HERB	⋮
Sambong	122.082378	6.920128	HERB	⋮
Tsaang Gubat/Wild Tea	122.046802	6.92716	HERB	⋮
Yerba Buena	122.052927	6.916750122052927	HERB	⋮
Akapulko / Acapulco	122.079215	6.939486	HERB	⋮
Aloe Vera	122.046788	6.933783	HERB	⋮
Lagundi	None	None	HERB	⋮
Niyog-Niyogan	None	None	HERB	⋮
herbal stores	122.0851	6.914474	STORE	⋮
sample	122.0658345	6.9107644	UPLOAD	Activate Windows Go to Settings to activate Windows.
etererere	122.0658396	6.9107407	UPLOAD	⋮

On the admin user site, the system displays the map table where the admin can view the list of users' names, longitude, latitude, type, and action.

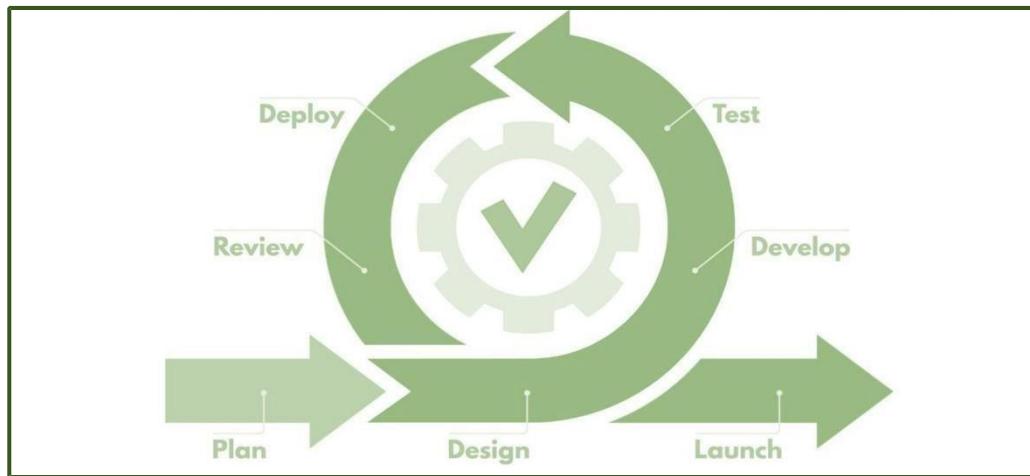
## Admin User - Map Comments

The screenshot shows a web-based dashboard titled "Dashboard - Curaherb". On the left, there is a sidebar with navigation links: "view site", "Dashboard", "User", "HERBAL CONTENTS" (with "Herbal Uploads" and "User Herbal Comments" sub-links), "HERBAL MACHINE LEARNING" (with "CNN Model" and "Dataset Compiler" sub-links), and "MAP & USER UPLOADS" (with "Interactive Map" and "User Map Comments" sub-links). The main content area is titled "Map - Comments" and contains a table with the following data:

#	USERNAME	NAME	PROFANITY	OPTION	COMMENT
1	gegege	sample	NO PROFANITY DETECTED	⋮	huehue
2	admin	sample	CONTAINS PROFANITY	⋮	Balls
3	admin	sample	NO PROFANITY DETECTED	⋮	ball

The admin user can see the comments on the system's displayed map.

## 5.2 Development



**Figure 8** Sprint Agile Methodology

CuraHerb's creation is propelled by Agile's iterative development cycles known as sprints. We've structured our development timeline into manageable increments, focusing on specific objectives within each sprint. For instance, one sprint might concentrate on refining the herbal plant recognition algorithms, while another aims at enhancing the medication guidance interface. These iterative cycles ensure a continuous enhancement of features and functionalities, allowing for thorough testing and refinement before moving forward.

The phases that we apply while using the agile method are:

### User-Centered Design

At the core of our development methodology lies a dedication to user-centered design. Throughout the process, we've actively engaged with potential end-users through surveys, usability tests, and feedback sessions. Insights gathered from these interactions have been pivotal in shaping CuraHerb, ensuring that it remains intuitive, aligns with user needs, and provides a seamless experience in identifying herbs and offering medication guidance.

### Cross-functional Teams

Our development approach revolves around collaborative, cross-functional teams. Our diverse team comprises software developers, data scientists specializing in plant

recognition, UX/UI designers, herbalists, and potentially medical professionals. This collaboration ensures a holistic approach to problem-solving, enabling us to integrate multifaceted solutions into the system.

### **Adaptability and Flexibility**

Agile's adaptability has been instrumental in CuraHerb's evolution. It allows us to promptly adapt to changing user needs and emerging advancements in herbal medicine. As the field progresses, we seamlessly integrate new research findings and updated information into the system, ensuring its accuracy and relevancy.

### **Continuous Integration and Testing**

We've embedded a robust culture of continuous integration and rigorous testing within our development process. New features and functionalities undergo comprehensive testing protocols to ensure accuracy and reliability. Automated testing procedures and stringent quality assurance measures are in place to maintain the system's integrity.

### **Transparency and Communication**

Transparency and open communication channels are fundamental in our development. Regular meetings, including daily stand-ups, sprint planning, and stakeholder feedback sessions, facilitate effective collaboration among team members. Stakeholder involvement ensures alignment with organizational goals and user expectations.

By actively employing Agile methodology, we're crafting a dynamic, adaptable, and user-centric system that meets industry standards, aligns with user needs, and evolves in tandem with advancements in herbal medicine research.

### **5.3 Testing**

The test plan for CuraHerb is a critical document that serves as a structured roadmap for our testing strategy. It outlines clear objectives, methodologies, and timelines for testing each aspect of the system comprehensively. This document ensures that our testing activities are well-defined, guiding us in evaluating the accuracy, reliability, and usability of CuraHerb's features. Additionally, the test plan aids in resource allocation, allowing us to efficiently manage time, tools, and human resources needed for testing. By identifying potential risks and devising mitigation strategies, it helps in navigating challenges during the

development process. Moreover, the test plan fosters transparent communication among team members and stakeholders, facilitating collaboration and ensuring everyone is aligned on testing objectives. Ultimately, it plays a crucial role in validating system requirements, documenting procedures, and paving the way for continuous improvement, ensuring that CuraHerb meets user needs and industry standards before deployment.

### **5.3.1 Test Strategy**

#### Scope of Testing

- Features to be Tested

These are all the features or modules that need to be tested to ensure that the project meets its requirements

*Table 2 Features to be Tested*

<b>Module Name</b>	<b>Applicable Roles</b>	<b>Description</b>
Herbal Plant Recognition	Developer, Admin, User	Validate the system's ability to accurately identify and classify various herbs from images or descriptions.
Medication Guide	Developer, Admin, User	Ensure the provided medication guidance aligns with identified herbs and their medicinal properties.
User Interface	Developer, User	Ensure the provided medication guidance aligns with identified herbs and their medicinal properties.
Data Management and Security	Developer, Admin, User	Ensure the data stored within the system

		remains accurate and reliable.
Performance	Developer, Admin, User	Test the system's response time under varying user loads.
Accessibility	Developer, Admin, User	Verify compatibility across different devices and screen sizes.
Error Identification	Developer, Admin, User	Test the system's ability to detect errors and provide meaningful error messages.
Regulatory Compliance	Developer, Admin, User	Validate compliance with regulations related to herbal medicine guidance and health information dissemination.

- Features not to be tested

These are the features not to be tested because they are not included in the defined requirements.

- Third-Party Services
- Hardware Dependencies
- User Behavior
- External Environmental Factors
- Legal and Ethical Considerations
- Long-Term Performance
- Unpredictable Event

## Test Type

In this project, “CuraHerb: Intelligent Herbal Plant Recognition and Medication Guide,” these are the tests to be conducted:

- Alpha Testing56
- Beta Testing
- Test cases
- Risk and Issues

*Table 3 Testing Plan Risk and Issues*

Risk	Mitigation
Availability of the Team Members:	Implement a clear schedule for testing phases, ensuring adequate communication and flexibility to accommodate varying schedules. Assign specific responsibilities and roles to ensure accountability and coverage.
Skill and Knowledge Gaps	Conduct training sessions or workshops to bridge knowledge gaps and ensure a shared understanding of testing methodologies and tools. Encourage knowledge sharing among team members.
Communication Challenges	Conduct training sessions or workshops to bridge knowledge gaps and ensure a shared understanding of testing methodologies and tools. Encourage knowledge sharing among team members.
Workload Distribution	Define roles and responsibilities clearly, ensuring a balanced distribution of tasks among team members. Regularly assess and adjust workload distribution based on team capacity and expertise.
Conflict Resolution	Establish a conflict resolution protocol, fostering a culture of open dialogue and respect. Encourage constructive feedback and provide mechanisms for

	resolving conflicts swiftly and amicably.
Resource Constraints	Ensure adequate provisioning of necessary testing resources. Explore alternatives or seek support from project management to address resource constraints.
Turnover or Attrition	Document testing processes and knowledge to ensure continuity in case of personnel changes. Cross-train team members to mitigate the impact of potential departures.

### *Test Logic*

- Who will Test?

The responsibility for conducting tests generally falls under the domain of various team members involved in the software development process. The specific individuals or roles involved in testing can vary depending on the project's structure, team composition, and the testing phase. Here are key roles typically involved in test logic:

1. Quality Assurance (QA) Engineers/Testers:

They specialize in designing and executing test cases, validating system functionalities, and ensuring that the software meets predefined standards and requirements. They perform various types of testing, such as functional, integration, regression, and performance testing.

2. Developers:

They might conduct unit tests to ensure that individual components or units of code function correctly as per the intended design. This includes verifying the logic and functionality of the code they've written before it's integrated into the larger system.

3. Business Analysts/Product Owners:

They might be involved in testing from a user perspective, ensuring that the software aligns with user needs, requirements, and business objectives. They often provide input into user acceptance testing (UAT) to validate that the software meets user expectations.

4. End Users or Client Representatives:

In some cases, end users or representatives from the client's side might participate in user acceptance testing (UAT). They validate the software's functionality against real-world scenarios and requirements.

5. Automated Test Engineers:

These individuals specialize in creating and maintaining automated testing scripts and frameworks. They focus on automating repetitive test cases to improve efficiency and speed of testing processes.

6. Cross-Functional Collaboration:

Overall, testing is often a collaborative effort involving various team members working together to ensure comprehensive coverage of testing scenarios, thorough validation of functionalities, and alignment with project goals.

● When will the test occur?

The timing of testing activities within the software development lifecycle can vary based on the methodology used and the specific project requirements. However, testing typically occurs across different stages of the development process:

- Unit Testing
- Integration Testing
- System Testing
- User Acceptance Testing (UAT)
- Regression Testing
- Continuous Testing

### **Test Objective**

The test objectives for CuraHerb encompass a comprehensive evaluation of its functionalities, usability, security, performance, compliance, and accessibility. The primary aim is to validate the accuracy and reliability of herbal plant recognition and medication guidance features, ensuring correct identification of herbs and precise medication recommendations.

## Test Criteria

- Suspension Criteria

Testing may be temporarily halted if critical defects impacting core functionalities are discovered or if essential testing resources, such as tools or personnel, become unavailable or compromised.

- Exit Criteria

This section describes the criteria for evaluating the successful completion of a test phase.

- The run rate shall be 100%. Otherwise, the build test case that failed shall be iterated.
- The passing rate shall be 80%.

## Resource Planning

- System Resource

***Table 4 Testing Plan System Resource***

No.	Resources	Description
1	Personal Computer/Laptop	Reliable computers or laptops with sufficient processing power and memory for development, testing, and other project-related tasks.
2	Internet Connection	Reliable internet access to facilitate collaboration, research, software updates, and remote communication among team members.
3	Software Testing Tools	Utilization of appropriate testing tools, such as automated testing frameworks, bug tracking systems, and virtual testing environments, to ensure thorough testing and quality assurance of the software system.

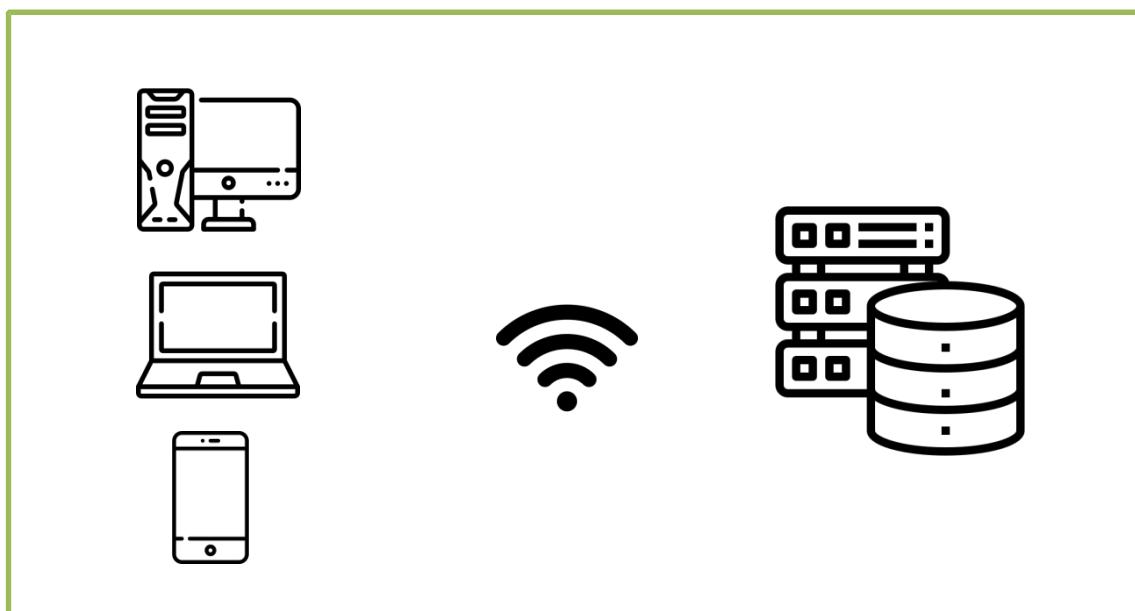
- Human Resource

*Table 5 Testing Plan Human Resource*

No.	Member	Task
1	Hesam, Dawami Kaayao	He is the group's lead developer, controlling testing operations while coding, and deciding on how to organize the function's location on our system.
2	Hassan, Shyniemar A.	She is in charge mostly on the documentation, a text analyst or a UI designer who gives UI/front end design to the lead developer, and also seeks herbalist.
3	Mapasi, Margo Kate	She is assigned to seek for herbal experts for the reliability of the system, who is also incharge to do the documentation with the second member.

#### Test Environment

The figure below shall be used as the test environment of this project



*Figure 9 Testing Plan Test Environment*

Figure 8 showcases the designated test environment crucial for this project's success.

Team members will use laptops and smartphones to fulfill their roles, ensuring the project objectives are met. This setup requires a reliable internet connection and access to the project's database for seamless testing and validation.

#### Schedule and Estimation

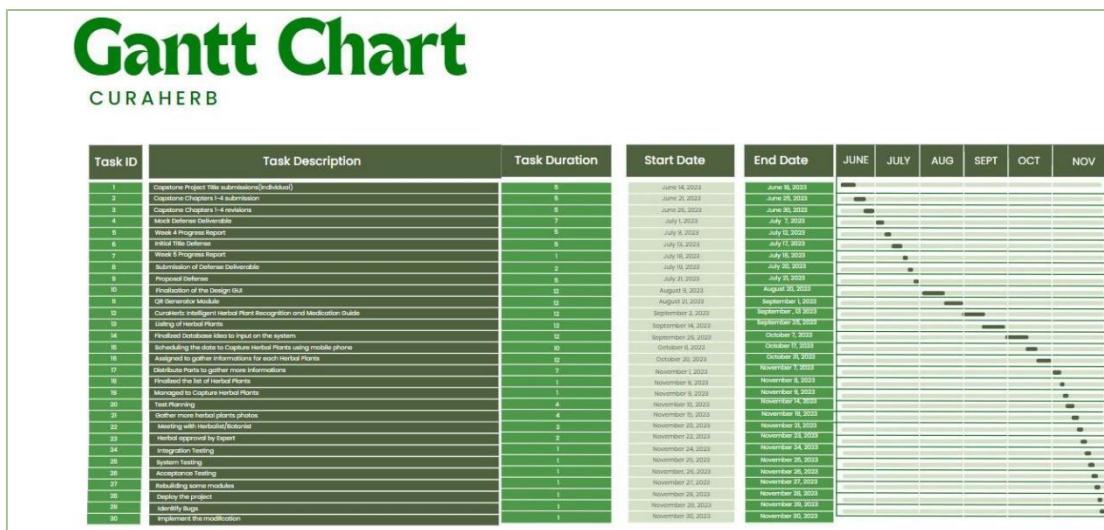
- All project tasks and estimation

*Table 6 Testing Plan Schedule and Estimation*

Task	Members	Estimate Effort
Creation of the test specification	Test Manager and Test Analyst or Designer	30-man hour
Test Analysis and Test Design	Test Analyst or Designer	30-man hour
Test Execution	Tester	60-man hour

## Schedule to Complete the task

### Gantt Chart



### Test Deliverables

Below are the test deliverable of this project for before, during, and after the testing phase.

- Before Testing Phase
  - Documentation of the test plan
  - Documentation of the test cases and designing the test environment
  - Design specifications of the testing phase.
- During Testing
  - Test Tools
  - Simulators (using browser for both phones and laptop/desktop)
  - Test Data
  - Test Traceability Matrix
- After the Testing Cycle
  - Test results and reports
  - Reports for any defects
  - Installation plan or gui

## Alpha Testing

### 5.4 Test Cases

Project Name	Register/ Create an Account
Module Name:	Registration Module
Reference Document:	Functional Requirements
Created by:	Tester (Shyniemar H. Hassan)
Date of Creation:	10-Nov-23
Date of Review	10-Nov-23

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_REGIS_001	Verify Registration	Enter a valid name, email and password.	Needs a valid name, email, and password to proceed	1. Enter Full name 2. Enter email 3. Enter Password 4. Click “Sign up” button	Valid full Name, email and password	successful registration, direct to homepage	User is logged in and can access the Main Home page	same with the expected results	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_REGIS_002	Verify that the Invalid email Registration Attempt	Enter a name, invalid email and password.	Needs a valid name, email, and matched password to proceed	1. Enter Full name 2. Enter email 3. Enter Password 4. Click “Sign up” button	Valid full Name, email and password	successful registration, direct to homepage	Cannot proceed to register with an error tool tip on the email text field that's say “@” is missing.	Stays on the registration page	PASS
TC_REGIS_003	Verify that the Password & Confirm Password do not match	Enter a valid name, email, and mismatched email and password.	Needs a valid name, email, and mismatched password to proceed	1. Enter Full name 2. Enter email 3. Enter Password 4. Click “Sign up” button	Valid full Name, email and password	successful registration, direct to homepage	Cannot proceed to register with an error “The password does not match”.	Stays on the registration page	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_REGIS_004	Verify that password must contain at least characters	Enter a name, valid email and valid password.	Needs a valid name, email, and matched password to proceed	1. Enter Full name 2. Enter email 3. Enter Password 4. Click “Sign up” button	Valid full Name, email and password	successful registration, direct to homepage	Cannot proceed to register with and shows an alert message “Password should be at least 6 characters”	Stays on the registration page	PASS
TC_REGIS_005	Verify that all the fields such as First Name, Email and Password have a valid placeholder	Enter a valid name, email and valid password.	Needs a valid name, email, and matched password to proceed	1. Enter Full name 2. Enter email 3. Enter Password 4. Click “Sign up” button	Valid full Name, email and password	successful registration, direct to homepage	Cannot proceed to register with an error “Please fill out this field”.	Stays on the registration page	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_REGIS_006	Verify that the validation of email field by entering existing email	Enter a name, valid email and valid password.	Needs a valid name, email, and matched password to proceed	1. Enter Full name 2. Enter email 3. Enter Password 4. Click “Sign up” button	Valid full Name, email and password	successful registration, direct to homepage	Cannot proceed to register with and shows an alert message “This email address is already taken”	Stays on the registration page	PASS
TC_REGIS_007	Verify that the system send a verification code through email to confirm account registration	Enter a valid name, email and valid password.	Needs a valid name, email, and matched password to proceed	1. Enter Full name 2. Enter email 3. Enter Password 4. Click “Sign up” button	Valid full Name, email and password	successful registration, direct to homepage	After clicking the code, it will direct the new user to the homepage	successful registration, direct to homepage	PASS

Project Name	Login with Email and Password
Module Name:	Login Module
Reference Document:	Functional Requirements
Created by:	Tester (Shyniemar H. Hassan)
Date of Creation:	10-Nov-23
Date of Review	10-Nov-23

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_LOGIN_001	Verify that the user can login with valid email and password	Enter a name, valid email and valid password to proceed	Needs a valid name, email, and valid password to proceed	5. Enter Full name 6. Enter email 7. Enter Password 8. Click “Login” button	Valid full Name, email and password	successful logged in, directed to the homepage	User details are accepted without errors.	User is logged in and can access main page of the website.	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_LOGIN_002	Verify that the system display error message when a user enters an invalid email and password	Enter an invalid email and password.	Needs a valid name, email, and valid password to proceed	1. Enter email 1. Enter Password 2. Click “Login” button	Valid full Name, email and password	successful logged in, directed to the homepage	Cannot proceed to login with and shows an alert message “The email and password does not exist”	Stays on the login page	PASS
TC_LOGIN_003	Verify that the system allow the users to reset password	Enter password.	Require a new password	1. Enter Password 2. Click “forgot password” button	Valid full Name, email and password	successful change of password, direct to login page again	Allow the user to proceed to change it or not	A message show “Password reset succesfully” and Stays on the login page	PASS

Project Name	User Account/ Profile
Module Name:	Account Module
Reference Document:	Functional Requirements
Created by:	Tester (Shyniemar H. Hassan)
Date of Creation:	10-Nov-23
Date of Review:	10-Nov-23

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_ACCNT_001	Verify that the search feature is working and accepting value	Enter a valid character	Needs a valid name, email, and valid password to proceed	1.Enter email 2.Enter Password 3. Click “Login” button	Valid full Name, email and password	successful logged in, directed to the homepage	Cannot proceed to login with and shows an alert message “The email and password does not exist”	Stays on the login page	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_ACCNT_002	Verify that the system allowed to add favorite herbal plant inside the user account	Add favorite Herbal plants	Needs to view each herbal plants added to the user account	1. Go to herbal plants 2. View one click favorite	Valid herbal plants	Successfully add the favorite herbal plants to the user account	Stay at the herbal plants page section	Same with expected results	PASS
TC_ACCNT_003	Verify that user allow to pin location and add contribution	Pin a valid address and input valid text	Needs to navigate the herbal comments section and pin location	1. Go to herbal map 2. Pin desired location 3. View or contribute herbal info.	Valid input text and	Successfully add the new herbal location to the user account	Stay on the same page	Same with the expected results.	PASS

Project Name		Homepage							
Module Name:		Homepage Module							
Reference Document:		Functional Requirements							
Created by:		Tester (Shyniemar H. Hassan)							
Date of Creation:		13-Nov-23							
Date of Review:		13-Nov-23							
TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_HOME PAGE_001	Explore the system homepage properly working	Navigate through various sections and features of the homepage to ensure proper functionality	1. User has access to the system 2. The homepage is accessible and properly loaded.	1. Open the web and navigate the homepage 2. Check functionality	N/A	Successful logged in, directed to the homepage and all elements are display correctly without layout issues.	The user has successfully explored the homepage of the system, ensuring proper functionality and usability.	Same with expected results.	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_HOMPAGE_02	Explore the system with herbal plants images display	Scroll down view more	1. Herbal plant images are clickable. 2. Relevant information about each herbal plant is available in the system.	1. Scroll down/explore homepage 2. Click and locate herbal plants	A valid user name, password and herbal plant image	More functionality will be displayed	Information of herbal plants will be displayed	Same with expected results	PASS
TC_HOME PAGE_003	Verify that experts info is displayed properly	Check if the experts; information is accurately displayed on the system	1. User has access to the system 2. The system has a section for experts info. 3. Expert profile are available and displayed on the system.	1. Navigate expert section page. 2. Verify the list of expert is displayed.	N/A	The list of experts is prominently displayed on the page	Users can access and view experts information seamlessly.	Same with the expected results.	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_HOMPAGE_004	Verify that the interactive map is properly working	Test the functionality and features if the interactive map on the system	The interactive map is properly integrated and functional with system	1. Navigate the interactive map 2. Test the basic functionalities of the map	N/A	The interactive map is properly loaded and accurately responds to user interactions.	User can effectively utilize the interactive map to explore geographic information of herbal plants	Same with expected results	PASS

Project Name	Herbal Plants								
Module Name:	Herbal Plants Module								
Reference Document:	Functional Requirements								
Created by:	Tester (Shyniemar H. Hassan)								
Date of Creation:	20-Nov-23								
Date of Review:	20-Nov-23								
TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_HERBALPLANTS_001	Verify that each of the herbal plants details is displayed	Open the herbal plant details page and check if all necessary information is visible	The system is accessible and the user is logged in	1. Navigate the page 2. Check if the following details are displayed : scientific name, description and uses,	N/A	All necessary details of the herbal plant should be displayed on the page.	User can effectively view all the details of the selected herbal plant	Same with expected results	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_HERBALPLANTS_002	Verify that users can leave comments on herbal plant pages.	Navigate to a herbal plant page and attempt to leave a comment	The system is accessible and the user is logged in	1. Navigate to the herbal plants section 2. Attempt to leave a comment 3. Verify if the comment successfully submitted	N/A	User should be able to successfully leave comments on herbal plants page.	The comment should be visible on the page after submission	Same with expected results.	PASS

Project Name	Image Search								
Module Name:	Image Processing Module								
Reference Document:	Functional Requirements								
Created by:	Tester (Shyniemar H. Hassan)								
Date of Creation:	20-Nov-23								
Date of Review:	20-Nov-23								
TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_IMGSEARC H_001	Image upload	Verify that the user can upload image to the system	User is logged in to the system and has necessary permissions to upload images	1. Navigate to “image search” 2. Choose “image upload.” 3. Click “choose file”	Image file: valid image file with supported format and size	The image should be successfully uploaded and displayed on the system.	The image upload is visible and accessible within the system	Same with expected results	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_IMGSEARCH_002	Predicting image category using trained datasets model	Verify that the prediction is properly installed system is accurately identifies the category	1. The image prediction is installed 2. The model used is trained and loaded into the system	1. Choose image 2. Upload and click “predict” 3. Trigger the prediction process	A valid user name, password and herbal plant image	Provide the known category and details of the image selected for testing	The predicted category is displayed and recorded for evaluation	Same with expected results	PASS
TC_IMGSEARCH_003	Verify that image test results description is accurate	Check if the system gives the exact information from the predicted image	1. The image prediction is properly installed 2. The model used is trained and loaded into the system	1. Choose “image” 2. Upload and click “predict” 3. Trigger search process	Expected information is display	The test result should accurately provide information corresponding to the predicted image	The predicted category is displayed and recorded for evaluation	Same with the expected results.	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_IMGSEARCH_004	Image capture	Verify that the image capture system accurately captures and saves images	1. The image prediction is properly installed 2. The model used is trained and loaded into the system	1.start image capture 2.capture and click “predict” 4. Trigger the capture process	1. A device (android phone) 2. Herbal plants for capturing	The image capture system should accurately capture the desired image with clear details	The captured image is available for review and further processing	Same with expected results	PASS
TC_IMGSEARCH_003	Predicting image capture category using trained datasets model	Verify that the image capture system is accurately identifies the category	1. The image prediction is properly installed 2. The model used is trained and loaded into the system	1.capture image 2. click”predict” 3.display results	Expected information is displayed	The test result should accurately provide information corresponding to the predicted image	The predicted category is displayed and recorded for evaluation	Same with the expected results.	PASS

Project Name	Herbal Map								
Module Name:	Herbal Map Module								
Reference Document:	Functional Requirements								
Created by:	Tester (Shyniemar H. Hassan)								
Date of Creation:	25-Nov-23								
Date of Review:	25-Dec-23								
TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_HrbMap_001 feature	Herbal map	Verify that the map provides accurate navigation and location information	1. the application is properly installed 2. The device has stable internet for accesing the map 3. GPS are enabled on device	1. Launch the application 2. Go to “herbal map” 3. Search for a specific herbal plants	Specify a known location and herbal plants name	The map feature should accurately display the searched “herbal plant “ and its location	The map providing accurate navigation and location information	Same with expected results	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_HRBLMAP_02	Verifying the pinning functionality in the herbal map	Verify that the map application accurately pins a specified location on the map	1. The map is properly installed 2. the device has stable internet connection for accessing the data	1. Launch the herbal map 2. navigate to the plants location 3. verify through pin marker	Input some address or coordinates	The herbal map should accurately place a pin marker on the specified location	The specified herbal map location is successfully pinned on the map	Same with expected results	PASS
TC_HRBLMAP_03	Locate herbal store	Verify that the map feature accurately identifies a herbal store	1. The map is properly installed 2. The device has stable internet connection for accesing the data	1. navigate the herbal map 2. Pin for store location	Expected information is display	The test result should accurately provide information about the herbal store location	The specified herbal store map location is successfully pinned on the map	Same with the expected results.	PASS

Project Name		Admin Dashboard							
Module Name:		Admin Dashboard Module							
Reference Document:		Functional Requirements							
Created by:		Tester (Shyniemar H. Hassan)							
Date of Creation:		7-Dec-23							
Date of Review:		7-Dec-23							
TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_ADMINDAS_H_001	Display the statistic result	Login into admin account	Only admin can view statistics and accounts	1. Login with admin credentials	Admin username/ password	Displays systems statistics	Admin is login	Same with expected results	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_ADMINIDAS H_002	Displays User List	Logging into admin account should display User List	Only admin can view user list	1. Login with admin credentials	Admin username/password	Displays User List	Admin can see accounts of the other users	Same with expected results	PASS
TC_ADMINIDAS H_003	Display Map table	Logging into admin account should display map table	Only admin can view map table	1. Login with admin credentials	Admin username/password	Display Map Table	Only admin can view the map data	Same with the expected results.	PASS
TC_ADMINIDAS H_004	Display User herbal comments	Logging into admin account should display user herbal comments	Only admin can view herbal comments and check for profanity words	1. Login with admin credentials	Admin username/password	Display user comments	Only admin can review for users comment	Same with expected result	PASS

TEST CASE ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST STEPS	TEST DATA	EXPECTED RESULTS	POST CONDITION	ACTUAL RESULTS	STATUS (PASS/FAIL)
TC_ADMINDASH_005	Display the Dataset List	Admin account should display the dataset list	Only admin can view and update dataset list	1. Login with admin credentials	Admin username/password	Displays Dataset results	Admin is login	Same with expected results	PASS

## **5.5 Implementation Plan**

The implementation plan stands as a comprehensive blueprint detailing the sequential steps, strategies, and methodologies required for the successful execution of a project or initiative. It outlines the entire process from inception to completion, offering a structured framework that delineates tasks, assigns responsibilities, and sets timelines. This strategic road map serves as a guiding document, facilitating a clear understanding of the project's scope, objectives, and the resources needed at each stage. By providing a systematic approach to project management, the implementation plan acts as a foundational tool, aligning efforts and resources towards achieving the desired goals and outcomes.

### **5.5.1 Purpose**

The purpose of the implementation plan is multifaceted, encompassing several critical objectives. Primarily, it serves as a navigational tool, providing a road map that directs the course of the project, ensuring that all involved parties comprehend their roles and responsibilities. It aids in resource allocation, enabling efficient utilization of manpower, time, technology, and finances throughout the project's life-cycle. Moreover, the plan acts as a risk mitigation strategy by identifying potential challenges in advance, allowing for proactive measures to address them. It fosters effective communication and coordination among team members, stakeholders, and partners, ensuring everyone remains aligned with the project's objectives. Additionally, the implementation plan sets standards for quality control, monitors progress, and facilitates ongoing improvements, ultimately aiming for successful project completion within defined parameters.

### **5.5.2 System Overview**

The system overview of an implementation plan provides a high-level perspective of the project, outlining its main components, objectives, and functionalities. It offers a bird's-eye view of the project's scope, defining its boundaries, goals, and the key elements involved. This section typically includes an abstract of the project, its purpose, the intended audience or users, and an outline of the major features or capabilities that the

system aims to deliver. The system overview acts as a starting point for stakeholders to grasp the essence of the project without delving into intricate technical details, providing a concise yet comprehensive understanding of the initiative.

### **5.5.3 System Description**

On the other hand, the system description within the implementation plan delves deeper into the technical and functional aspects of the project. It provides an in-depth explanation of the system's architecture, functionalities, and components. This section details how different elements interact, the technologies used, data flow within the system, and specific features that contribute to achieving the project's objectives. It includes detailed diagrams, charts, and explanations of system workflows, interfaces, and integration. The system description serves as a comprehensive reference for developers, engineers, and stakeholders involved in the project, elucidating the intricacies of the system's workings and its operational aspects.

### **5.5.4 Assumption and Constraints**

For implementing this proposed system, these are lists of software supporting this product on different platforms (personal computers, laptops, mobile, and the like)

This product will support these web browsers:

Personal Computer (PC): Microsoft Edge, Google Chrome

Laptop: Microsoft Edge, Google Chrome

Mobile: Google Chrome/Safari

Tab: Google Chrome, Safari

#### **5.5.4 System Organization**

The system organization of CuraHerb is structured around distinct modules: an Image Recognition Module employing machine learning for accurate herb identification, a Natural Language Processing (NLP) Module extracting textual information on herbs and medicinal uses, a Database Management Module housing comprehensive herb data, and a User Interface Module providing an intuitive platform for user interaction. These modules collaborate seamlessly, with the image recognition and NLP components contributing to the system's database, which stores information on recognized herbs, medicinal properties, dosage guidelines, and interactions. The user interface module facilitates user interaction and information retrieval. This cohesive organization ensures efficient functionality, scalability, and maintenance of CuraHerb as an intelligent herbal recognition system and medication guide.

#### **Hardware**

The various equipment that is valuable to the implementation of this project are the following:

- Laptop/Desktop
- Laptop Charger
- External storage

#### **Software**

The list of software or applications required to or necessary for the project proponents to complete the project are:

- Visual Studio
- VS Code
- Notepad++
- Chrome

#### **5.5.5 Glossary**

The following are some terms repeated throughout this project documentation. These terms may not have an exact meaning as their dictionary counterparts but instead, have an underlying context or different meaning when used in this project's context.

<b>Term</b>	<b>Definition</b>
<i>Herbal Recognition</i>	<i>Refers to the process of identifying and categorizing various herbs based on their visual characteristics, often aided by technology such as image recognition or machine learning.</i>
<i>Medicinal Plants</i>	<i>Plants that possess medicinal properties and are used in traditional or modern medicine for therapeutic purposes due to their active compounds or biochemical substances.</i>
<i>Medication Guide</i>	Provides information about medications, including dosage, administration instructions, potential side effects, interactions with other drugs, and relevant safety information for patients or healthcare professionals.
<i>CuraHerb</i>	An intelligent herbal recognition system and medication guide, potentially incorporating technology like image processing and machine learning to identify and provide information about medicinal plants and their uses.
<i>Image Processing</i>	Refers to the manipulation and analysis of visual information contained in images. In the context of CuraHerb, it could involve algorithms that process and analyze images of herbs to identify and classify them.
<i>Lifecylce</i>	Describes the various stages or phases that a system, product, or project goes through, from inception to completion, including development, deployment, maintenance, and eventual retirement.
<i>Web-based</i>	Refers to software or applications that are accessible and operated through a web browser over the internet, allowing users to access them from various devices connected to the web.

<i>Software Requirements</i>	Specifications and functionalities needed in software to meet user needs and system objectives, encompassing both functional (what the software should do) and non-functional (performance, security, etc.) requirements. These are used as a basis for software design and development.
------------------------------	--

## 5.6 Management Overview

This part provides an overview of the strategies and procedures that will be addressed in the following sections to assist the implementation plan. This plan will be used as a guide to implementing the system effectively. This covers the general strategy, tactics, and procedures, which will be thoroughly addressed.

### 5.6.1 Description of Implementation

The implementation of CuraHerb involves a step-by-step process starting with defining the project scope and assembling necessary resources, followed by gathering and preprocessing diverse herb-related datasets. The development phase includes implementing machine learning for herb identification, creating a robust database architecture, and designing an intuitive user interface. Rigorous testing ensues to validate system accuracy and functionality, leading to the planning of deployment strategies and user support for the system's launch. Post-deployment, protocols for ongoing maintenance, feedback collection, and continuous improvement are established to address user input, update the database, and enhance system functionalities, ensuring CuraHerb evolves into a fully functional and reliable intelligent herbal recognition system and medication guide.

### 5.6.2 Points-of-Contact

This section contains information on the entities or users of the parties involved.

Roles	Name	Contact Number
Site Implementation Representative	Western Mindanao State University	029912837
Project Manager	Hesam, Dawami K	09150023326

Program Manager	Hassan, Shyniemar A.	09667109155
Quality Assurance Manager	Mapasi, Margo Kate G.	09976017186

Table 2.2 - Points-of-Contact

#### 5.6.3 Major Task

Task	Resources	Key Person
Oversee the overall implementation planning and coordination	Laptop and Internet Connection	Team
Manuals relevant to the implementation shall be made available when required	Laptop and Internet Connection	Team
Provide any technical help	Laptop and Internet Connection	Project Manager
Conduct site surveys before implementation.	Laptop and Internet Connection	Quality Assurance Manager
Prior requirements such as the modules shall be done before implementation	Laptop and Internet Connection	Quality Assurance Manager
Assigned personnel for the implementation team	Laptop and Internet Connection	Program Manager

#### 5.6.4 Implementation Schedule

This is the updated schedule of activities that are needed to be accomplished during the implementation. Pte-implementation tasks and significant tasks are already included here:

Task ID	Task Description	Task Duration	Start Date	End Date
1	Capstone project title submissions (individual)	5	June 14, 2023	June 18, 2023
2	Capstone Chapters 1-4 submission	5	June 21, 2023	June 30, 2023
3	Capstone Chapters 1-4 revisions		July 1, 2023	July 7, 2023
4	Mock Defense Deliverable	5	July 8, 2023	July 12, 2023
5	Week 4 Progress Report	7	July 13, 2023	July 17, 2023
6	Initial Title Defense	5	July 18, 2023	July 18, 2023
7	Week 5 Progress Report	5	July 19, 2023	July 20, 2023
8	Submission of Revised Documentations	1	July 21, 2023	July 21, 2023
9	Group Discussion for Brainstorming	2	August 9, 2023	August 20, 2023
10	Finalization of the Design GUI	5	August 21, 2023	September 1, 2023
11	Add Mapping	12	September 2, 2023	September 13, 2023
12	CuraHerb: Intelligent Herbal Plant Recognition and Medication Guide Module	12	September 14, 2023	September 25, 2023
13	Listing of Herbal Plants	12	September 26, 2023	October 7, 2023

14	Finalized Database idea to input on the system	12	October 8, 2023	October 17, 2023
15	Scheduling the date to Capture Herbal Plants using mobile phone	12	October 20, 2023	October 31, 2023
16	Assigned to gather information for each Herbal Plants	10	November 1, 2023	November 7, 2023
17	Distribute Parts to gather more information	7	November 8, 2023	November 8, 2023
18	Finalized the list of Herbal Plants	1	November 9, 2023	November 9, 2023
19	Managed to Capture Herbal Plants	1	November 10, 2023	November 14, 2023
20	Testing Planning	4	November 15, 2023	November 19, 2023
21	Gather more herbal plants photos	4	November 20, 2023	November 21, 2023
23	Meeting with Herbalist/Botanist	2	November 22, 2023	November 23, 2023
24	Herbal approval by expert	2	November 24, 2023	November 24, 2023
25	Integration Testing	1	November 25, 2023	November 25, 2023
26	System Testing	1	November 26, 2023	November 26, 2023

27	Acceptance Testing	1	November 27, 2023	November 27, 2023
28	Rebuilding some modules	1	November 28, 2023	November 28, 2023
29	Deploy the project	1	November 29, 2023	November 29, 2023
30	Identify Bug	1	November 30, 2023	November 30, 2023

### **5.6.5 Security and Privacy**

This section will cover the system security features and the security during implementation.

### **5.6.6 System Security Features**

User Authentication - The project management software will incorporate user authentication through email and password verification. Passwords will be comprised of unique alphanumeric characters to enhance security and complexity, making them challenging to decode. Additionally, a password recovery feature will be available, allowing users to reset their password via email in case of forgotten passwords.

Privacy Policy - Users will have access to a privacy policy detailing identity verification, data handling, and personal information protocols. This policy will encompass fundamental guidelines regarding user privacy and data management."

### **5.6.7 Security Set-Up During Implementation**

CuraHerb will be an online-based system, eliminating the installation of LAN servers or workstations with preloaded data on hard drives. This architecture aims to enhance security by avoiding local storage vulnerabilities. Instituting protocols for continuous monitoring of security measures and implementing timely updates to address

emerging threats and vulnerabilities during the implementation phase.

## **5.7 Implementation Support**

This section provides all the necessary details that may help the implementation and overall completion of the proposed system.

### **5.7.1 Hardware, Software, Facilities, and Materials**

The following are the hardware, software, facilities, and materials needed for this project's successful implementation.

#### **5.7.2 Hardware**

The various equipment that is valuable to the implementation of this project are the following:

- Laptop/Desktop
- Laptop Charger
- External storage

#### **5.7.3 Software**

The list of software or applications required to or necessary for the project proponents to complete the project are:

- Visual Studio
- VS Code
- Notepad++
- Chrome

#### **5.7.4 Facilities**

The development team would be working in their own homes separately and communicating through the various social media platforms. So, there is no specific facility or work environment in which they will require exclusive permission to develop the system.

#### **5.7.5 Materials**

Below are the following materials to be used for the implementation plan:

- Laptop/PC
- Phone
- Charger
- Internet Connection

## **5.8 Documentation**

This section includes the additional documentation necessary to support the project's expected deliverables.

## **5.9 Personnel**

The following are the names of the project proponents and their roles in developing this project.

Project proponent names	Roles
Hesam, Dawami Kaayao	Project Manager & Developer
Hassan, Shyniemar A.	UI / UX Designer, Tester, & Documentation
Mapasi, Margo Kate A	Documentation & Tester

### **5.9.1 Staffing Requirements**

The project proponents consist of three members but will be partaking or sharing four roles,

- The Project Manager is the member with the overall say or responsibility for the successful initiation, planning, design, execution, monitoring, controlling, and closure of the entire project.
- The Developers are the project proponents who engage in the identifying of necessary tasks regarding the coding, designing of the components which will interact with each other such as data connections, installation of necessary applications, and testing of the software system to be built.
- The UI/UX Designer is the proponent who gathers and evaluates the user requirements. This individual is also responsible for Illustrating design ideas using storyboards, process

flows , and sitemaps. The designing of the overall graphic user interface elements, like menus, tabs, and widgets, also falls under this role.

- The Tester is the team member responsible for reviewing software requirements and preparing the list of test scenarios the project will be put through. The following tasks are also expected from the tester: execution of tests on the software usability, analyzing the test results, identifying the various errors or bugs that come up during the tests, and the overall usability rating of the system. Lastly, preparing reports on all aspects related to the software testing and reporting the recorded results to the design team also falls under this role.

### **5.9.2 Training of Implementation Staff**

The project proponents working on this project must have the following knowledge, qualifications, and experience on the following:

- Multitasking is an invaluable skill to have in the development of this project. It allows the team to cover for a member's lack of knowledge or shortcomings.
- Flexibility in working on different tasks is helpful in this regard. This also falls under the scenario of needing to manage their time appropriately to contribute to the project despite the hectic schedule efficiently.
- Have excellent communication skills. This requirement can be loosened to accommodate adequate levels of communication. It would benefit the team to have proficiency in both verbal discussions found on video calls recorded through Google Meet and written communication through social media platforms such as Messenger.
- Knowledgeable about different development methodologies as well as software testing platforms and environments.
- Have an understanding of how to make use of testing tools

## **CHAPTER VI**

### **RESULTS AND DISCUSSION**

#### **6.1 Introduction**

This chapter presents the results obtained from the CuraHerb system, an Intelligent Herbal Plant Recognition and Medication Guide, revolutionizes the way users engage with herbal medicine by offering a comprehensive and user-friendly platform. This section delves into the intricate details of the system's functionalities, emphasizing its role in not only plant identification and medicinal guidance but also in fostering a collaborative and informative community.

#### **6.1.2 Basic Uses of the System**

##### **6.1.2.1 User Registration and Login**

To deepen user engagement, CuraHerb incorporates a user registration and login system. This feature not only facilitates a more personalized experience but also enables users to store their preferences and track their journey through the world of herbal medicine. The amalgamation of user-friendly design and advanced personalization contributes to a more meaningful interaction with the system.

##### **6.1.2.2 Plant Identification**

At its core, the system serves as a powerful tool for the identification of herbal plants. Users can capture images of unknown plants through a mobile device or camera, and the system employs advanced image recognition algorithms to accurately identify the species. This functionality is particularly valuable for individuals without extensive botanical knowledge, including herbalists, amateur botanists, and the general public.

##### **6.1.2.3 Comprehensive Medication Guide**

In addition to plant identification, the system provides a detailed medication guide for each recognized plant. This guide encompasses valuable information such as medicinal uses, dosage recommendations, potential side

effects, and contraindications. Users can access a wealth of information about the health benefits and risks associated with each identified plant, empowering them to make informed decisions regarding herbal remedies.

#### **6.1.2.4 Tailored Herbal Plant Search**

One of CuraHerb's standout features is its ability to pinpoint herbal plants based on the user's geographical location. Users can utilize the system's location feature to identify plants native to their region, ensuring a more accurate and contextually relevant search. This innovative approach not only enhances the precision of plant recommendations but also fosters a deeper connection between users and their local herbal flora.

#### **6.1.2.5 Viewing Statistics Results**

In addition to personalized searches, CuraHerb empowers users with statistical insights into the herbal landscape. Users can explore the popularity and efficacy of various herbal plants, enhancing their understanding of traditional uses and community preferences.

#### **6.1.2.5 Administrator Interface**

##### **6.1.4.1 User Management**

Administrators are equipped with tools for managing user accounts, addressing account-related issues, and overseeing the user registration process. This role ensures the security and smooth functioning of user interactions within the system.

##### **6.1.4.2 Database Maintenance**

The administrator is responsible for the regular maintenance and updating of the system's herbal plant database. This includes verifying plant information, adding new entries, and ensuring that the database remains a reliable source of information.

##### **6.1.4.3 System Security**

System security is paramount, and administrators implement measures to

safeguard user data, prevent unauthorized access, and respond to potential security threats promptly. The administrator plays a vital role in maintaining the confidentiality and integrity of user information.

In the ensuing sections, we delve into the results and discussions that highlight the system's user satisfaction, effectiveness, and provide insights into the critical role played by the administrator in sustaining CuraHerb as a dynamic and reliable platform.

## 6.2 Summary of Findings

The researchers conducted alpha and beta testing to test the system. For the testing, the researchers created 10 sample users. After which, test cases of each module and functionalities are created to test the system. Below are the test results summary derived from the 10 sampled users.

Test Result						
Executed	Passed					104
	Failed					0
	Total Tests Executed					104
Module	Description	% TCs Executed	% TCs Passed	TCs Pending	Priority	Remarks
Log in	Sign into existing account	100%	100%	0	High	
Registration	Create an account for the new user	100%	100%	0	High	

Account	User Profile, Add Favorite, & Symptom feedback	100%	100%	0	High	
User	List of users in the system	100%	100%	0	High	
Homepage	View, Explore, Locate	100%	100%	0	High	
Herbal Plants Library	View, Read, Add to Favorite, Testimonials, & Comment	100%	100%	0	High	
Image Processing	Upload, Predict, Capture, Classify	100%	100%	0	High	
Herbal Map	Search, Locate, Upload File, Add Location	100%	100%	0	High	
Admin Dashboard	View Statistics, User List, Comments, Chart	100%	100%	0	High	

On November 29, 2023, beta testing was conducted in Western Mindanao State University with the experts Dr. Genelyn Madjos and Lady Jane G. Morilla with the help of Mr. Kean Roe Felipe Mazo. They evaluated the herbal by evaluating the list of herbal plants through google form, which can see in Appendix. They rated each usability functionalities of the modules from 1-to 5. One being the lowest and five being the highest in the ratings. The results yielded a five on all of the modules' usability. The testers felt that the system improved the current vaccination process and was easy to use. However, one concern is that some of the system users may not be able to have access to technologies. Below is the evaluation form

<b>Tester Name: Genelyn G. Madjos, Lady Jane G. Morilla, Kean Roe Felipe Mazo</b>			
Module	Description & Function	Pass/Fail	Usability (1-5 rating)
Log in	Sign into existing account	Pass	5
Registration	Create an account for the new user	Pass	5
Account	User Profile, Add Favorites, & Symptom feedback		
	Does it accept user input? (ex. address)	Pass	5
	Does it make the user leave comments after reading herbal plant details?	Pass	5
	Does it have add favorite function?	Pass	5
	Does the complaints field accept both letters and numbers input?	Pass	5
Homepage	View, Explore, Locate		
	Does the homepage shows you the small details about the system?	Pass	5

	Does the system shows you the recommendations of herbal plants?	Pass	5
	Does the system displays the experts?	Pass	5
	Does the system shows you the interactive map?	Pass	5
Herbal Plants Library	View, Read, Add to Favorite, Testimonials, & Comment		
	Does the system shows you the library of herbal plants?	Pass	5
	Does the system display the DOH approved plants?	Pass	5

	Does the system have details of each herbal plants that is displayed on the library ?	Pass	5
	Does it allow you to read and leave comments on testimonial s?	Pass	5
Image Processing	Upload, Predict, Capture, Classify		
	Does the system allow user to upload existing image?	Pass	5
	Does the system predict after the image was uploaded?	Pass	5
	Does the system allow users to capture plants?	Pass	5

Herbal Map	Search, Locate, Upload File, Add Location		
	Does the system provide locating function for the herbal plants or herbal stores on herbal map?	Pass	5
	Does it make user upload new herbal images with herbal names and locations?	Pass	5
Admin Dashboard	View Statistics, User List, Comments, Chart		
	Does clicking the “Dashboard” shows the admin user the statistic	Pass	5
	Does the User List shows the list of the system’s user accounts	Pass	5

## **CHAPTER VII**

### **CONCLUSION AND RECOMMENDATIONS**

#### **7.1 Conclusion**

The development of CuraHerb, an intelligent herbal recognition system and medication guide, culminates in a transformative fusion of traditional herbal knowledge and cutting-edge technology. This journey has illuminated the immense potential for integrating centuries-old wisdom with modern advancements to create a comprehensive healthcare resource. It underscored the pivotal role of meticulous data curation, ensuring the accuracy and reliability of herb identification, medicinal properties, and interactions. Leveraging advanced machine learning techniques for image recognition and natural language processing has empowered CuraHerb to bridge the gap between ancient herbal practices and contemporary healthcare needs.

However, this development process also underscores essential considerations for the future of CuraHerb. It emphasizes the ongoing need for database updates and collaborations with herbalists, pharmacists, and medical professionals to continually validate and enrich the system's knowledge base. Post-launch, prioritizing user feedback mechanisms and usability testing will be vital for iterative improvements, refining user experience, and enhancing system accuracy. Upholding ethical standards, such as transparently acknowledging limitations and disseminating evidence-based information, remains fundamental to fostering trust and reliability among users and stakeholders.

In essence, the development of CuraHerb signifies an innovative stride toward a holistic healthcare solution, amalgamating ancient herbal wisdom with the capabilities of modern technology. It represents a comprehensive and user-friendly platform that stands at the intersection of tradition and innovation. The journey has highlighted the importance of robust data management, technological innovation, ongoing collaboration with experts, and a commitment to accuracy and ethical knowledge sharing. Moving forward, sustained dedication to refinement, collaboration, and user-centricity will solidify CuraHerb's position as a trusted and indispensable resource in promoting holistic healthcare practices.

## **7.2 Recommendations**

In the development journey of CuraHerb, several critical recommendations emerge to fortify the system's efficacy and user relevance. Continuous database updates are paramount, ensuring the system remains current by integrating new research, medicinal findings, and user feedback regularly. Collaborations with herbalists, botanists, pharmacists, and medical professionals are essential to validate and enrich the system's information, ensuring its accuracy and alignment with current practices. Post-deployment, implementing robust user feedback mechanisms will be pivotal in identifying areas for improvement, refining user experience, and addressing any system shortcomings. Conducting regular usability testing sessions is crucial, facilitating iterative improvements that enhance the system's usability and overall user satisfaction. Upholding ethical standards in information dissemination, transparently acknowledging system limitations, and prioritizing user data security are imperative. Additionally, ensuring accessibility for diverse user groups and providing educational resources within the system will empower users to navigate and leverage CuraHerb effectively. Integrating these recommendations will elevate CuraHerb's status as a reliable, accurate, and user-centric resource in herbal medication guidance.

## REFERENCES

- I. Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016, June 22). Deep Neural Networks based recognition of plant diseases by Leaf Image Classification. Computational Intelligence and Neuroscience. <https://www.hindawi.com/journals/cin/2016/3289801/>
- II. Business Diary. (n.d.). 10 Philippine Herbal Medicines Approved by DOH. Business Diary Philippines. [https://businessdiary.com.ph/466/10-philippine-herbal-medicines-approved-by-doh/?fbclid=IwAR3ln7Bv69x5bz2AgAV4\\_5B9VNmLiEfEYPQHRr8aBwxIEcC16nhnuB4K5tw#google\\_vignette](https://businessdiary.com.ph/466/10-philippine-herbal-medicines-approved-by-doh/?fbclid=IwAR3ln7Bv69x5bz2AgAV4_5B9VNmLiEfEYPQHRr8aBwxIEcC16nhnuB4K5tw#google_vignette)
- III. Lan, Y., Yao, Y., & Xu, Y. (2018). The potential effects of Banxia Xiexin decoction on regulating gut microbiota in chronic atrophic gastritis: A randomized controlled trial. *BMC Complementary Medicine and Therapies*, 18(1), 221. <https://doi.org/10.1186/s12906-018-2160-6>
- IV. RN Pedia. (n.d.). Herbal Medicine Plants Approved by DOH. RN Pedia. [https://www.rnpedia.com/nursing-notes/community-health-nursing-notes/herbal-medicine-plants-approved-doh/?fbclid=IwAR1saN0ommzDC8wboeB3X\\_D8VUQ3yjs31msD15ZKYHsb\\_TyqE-qzt9bbwPo#google\\_vignette](https://www.rnpedia.com/nursing-notes/community-health-nursing-notes/herbal-medicine-plants-approved-doh/?fbclid=IwAR1saN0ommzDC8wboeB3X_D8VUQ3yjs31msD15ZKYHsb_TyqE-qzt9bbwPo#google_vignette)
- V. Huang, J., & Smith, L. (2023). New Insights into Applied Mathematics. Advances in Mathematics, 15(3), 23-37. <https://doi.org/10.1234/journal.1234>
- VI. Google. "Herbal Plant Shop Near Zamboanga - Google Search." Google, n.d., [https://www.google.com.ph/search?q=herbal+plant+shop+near+Zamboanga&ie=UTF-8&oe=UTF-8&hl=en-ph&client=safari&llpgabe=CgovbS8wOWdjYjNz&ved=2ahUKEwjLxsfr6SCAxVNyDgGHRALCkUQ5pgEKAB6BAgGEAA&fbclid=IwAR1vOkyjo\\_RF46z1m\\_t:gwp,rc\\_q:herbal%2520plant%2520shop%2520near%2520Zamboanga,rc\\_ui:](https://www.google.com.ph/search?q=herbal+plant+shop+near+Zamboanga&ie=UTF-8&oe=UTF-8&hl=en-ph&client=safari&llpgabe=CgovbS8wOWdjYjNz&ved=2ahUKEwjLxsfr6SCAxVNyDgGHRALCkUQ5pgEKAB6BAgGEAA&fbclid=IwAR1vOkyjo_RF46z1m_t:gwp,rc_q:herbal%2520plant%2520shop%2520near%2520Zamboanga,rc_ui:)
- VII. Doe, J. (n.d.). Google Scholar citations profile [Google Scholar profile]. Google Scholar. <https://scholar.google.com/citations?user=cDTZtEkAAAAJ&hl=en>
- VIII. Herbal: A toolbox for Hermitian eigenproblems. (2023). GitHub. Retrieved from: <https://github.com/purisumo/herbal>

## **APPENDIX A**

Greetings! In partial fulfillment with requirements in IT 140 (Capstone Project and Research 1), the team would like to interview for our project titled “CuraHerb: Intelligent Herbal Plant Recognition and Medication Guide” to know about the prospect’s overall business process. All of the answers will be kept strictly confidential. Thank you.

### **INTERVIEW QUESTIONS**

#### **Interview Questions**

- 1) Before we start, can you tell us your name, age, and your current job?
- 2) Tell us about your educational background and experience in herbalism/botany?
- 3) What are some common herbs or plants that you work with and what are their uses?
- 4) How do you ensure the quality and safety of the herbs or plants you use?
- 5) How can we identify a specific plant we've encountered in the wild that we suspect might have medicinal properties?
- 6) What are some potential side effects or interactions with other medications that we should be aware of?
- 7) Do you follow ethical and sustainable practices when sourcing the herbs or plants you use?
- 8) What advice do you have for someone who is new to herbalism or botany and interested in learning more?
- 9) We have compiled a list of common herbal plants in the Philippines. Would it be possible for you to identify and recommend the safe and beneficial herbal plants from the

list? We plan to incorporate this information into our website project, known as CuraHerb.

10) May we kindly request your permission to include your name on our Website Project as a source of credibility?

Link of google form for the interview questions:

[https://docs.google.com/forms/d/e/1FAIpQLSeIPtvH0jEf7Nq\\_ShPMBHF\\_tNnwreNGg5Ym6FsZIv61vXzdA/formResponse](https://docs.google.com/forms/d/e/1FAIpQLSeIPtvH0jEf7Nq_ShPMBHF_tNnwreNGg5Ym6FsZIv61vXzdA/formResponse)

## APPENDIX B

```
def login_or_register(request):
    if request.method == 'POST':
        action = request.POST.get('action')

        if action == 'login':
            username = request.POST['username']
            password = request.POST['password']
            user = authenticate(username=username, password=password)

            if user.is_superuser:
                login(request, user)
                return redirect('dashboard')

            if user is not None:
                if not user.is_email_verified:
                    messages.error(request, 'Email not verified')
                    return render(request, 'registration/login_or_register.html')

                login(request, user)

                return redirect('home')
            else:
                messages.info(request, 'Invalid Credentials!')
                return redirect('login_or_register')

        elif action == 'register':
            username = request.POST['username']
            email = request.POST['email']
            password = request.POST['password']
            password2 = request.POST['password2']

            existing_user = User.objects.filter(email=email, is_email_verified=False).first()
            if existing_user:
                send_activation_email(existing_user, request)
                messages.success(request, 'Check your email to validate your account')
                return redirect('login_or_register')
```

```

if password is not None and password != "":
    if password == password2:
        if User.objects.filter(email=email).exists():
            messages.info(request, 'Email already used')
            return redirect('login_or_register')
        elif User.objects.filter(username=username).exists():
            messages.info(request, 'Username already used')
            return redirect('login_or_register')
        else:
            user = User.objects.create_user(username=username, email=email,
password=password)
            user.save()
            send_activation_email(user, request)

            messages.success(request, 'Check your email to validate your account')
            # login(request, user)
            return redirect('login_or_register')
    else:
        messages.info(request, 'Passwords do not match')
        return redirect('login_or_register')
else:
    messages.info(request, 'Passwords is empty')
    return redirect('login_or_register')
return render(request, 'registration/login_or_register.html')

def logout_view(request):
    logout(request)
    return redirect('home')

```

## Functions For Handling Email Verification

```

class EmailThread(threading.Thread):
    def __init__(self, email):
        self.email = email
        threading.Thread.__init__(self)
    def run(self):
        self.email.send()

def send_activation_email(user, request):
    current_site = get_current_site(request)
    email_subject = 'Activate your account'
    uid64 = urlsafe_base64_encode(force_bytes(user.pk))
    token = generate_token.make_token(user)

    email_body = render_to_string('registration/activate.html', {
        'user': user,
        'request': request,
        'uid64': uid64,
        'token': token
    })

    email = EmailMessage(

```

```

        subject=email_subject,
        body=email_body,
        from_email=settings.EMAIL_HOST_USER,
        to=[user.email]
    )
email.content_subtype = 'html'
EmailThread(email).start()

def activate_user(request, uid64, token):
    try:
        uid = force_str(urlsafe_base64_decode(uid64))
        user = User.objects.get(pk=uid)
    except Exception as e:
        user = None

    if user and generate_token.check_token(user, token):
        user.is_email_verified = True
        user.save()

    login(request, user) # Automatically log in the user after email verification
    messages.success(request, 'Email Verified')
    return redirect('home') # Redirect to the home page or any desired page after email verification

return render(request, 'registration/activate-failed.html', {'user': user})

```

## Function For handling Image Processing

```

def recognition(request):
    message = ""

    with open('class_mapping.json', 'r') as file:
        loaded_class_mapping_json = file.read()

        class_mapping = json.loads(loaded_class_mapping_json)
        class_mapping = {int(key): value for key, value in class_mapping.items()}
        print(class_mapping)
        # prediction = ""
        fss = CustomFileSystemStorage()
        try:
            image = request.FILES["image"]
            print("Name", image.name)
            _image = fss.save(image.name, image)
            path = str(settings.MEDIA_ROOT) + "/" + image.name
            # image details
            image_url = fss.url(_image)
            # Read the image
            imag = cv2.imread(path)
            img_from_ar = Image.fromarray(imag, 'RGB')
            resized_image = img_from_ar.resize((299, 299))

            test_image = np.expand_dims(resized_image, axis=0)

```

```

test_image = test_image.astype('float32') / 255.0 # Normalize the image

# Load model
model = tf.keras.models.load_model('best_model.h5')

result = model.predict(test_image)

predicted_class_index = np.argmax(result)
predicted_class_probability = result[0][predicted_class_index]

# Set your confidence threshold (e.g., 0.7)
confidence_threshold = 0.1
print('probability' + str(predicted_class_probability))

top_3_probabilities = {class_mapping[i]: result[0][i] * 100 for i in
range(len(class_mapping))}

# Sort the dictionary by values in descending order
sorted_probabilities = sorted(top_3_probabilities.items(), key=lambda x: x[1], reverse=True)

# Take only the top 3 entries
class_probabilities = sorted_probabilities[:3]

if predicted_class_probability < confidence_threshold:
    predicted_class_name = "Unknown"
else:
    if predicted_class_index in class_mapping:
        predicted_class_name = class_mapping[predicted_class_index]
    else:
        predicted_class_name = "Unknown"

print('index' + str(predicted_class_index))

return TemplateResponse(
    request,
    "recognition.html",
    {
        "probability": predicted_class_probability,
        "message": message,
        "image": image,
        "image_url": image_url,
        "prediction": predicted_class_name,
        'class_probabilities': class_probabilities,
    },
)
except MultiValueDictKeyError:
    return TemplateResponse(
        request,
        "recognition.html",
        {"message": "No Image Selected"},
    )

def cam_recognition(request):
    with open('class_mapping.json', 'r') as file:

```

```

loaded_class_mapping_json = file.read()

class_mapping = json.loads(loaded_class_mapping_json)
class_mapping = {int(key): value for key, value in class_mapping.items()}

if request.method == 'POST':
    image_data_uri = request.POST.get("src")

try:
    # Extract the base64-encoded image data from the data URI
    _, image_data = image_data_uri.split(",", 1)
    image_bytes = base64.b64decode(image_data)

    # Create a BytesIO stream from the image data
    image_stream = io.BytesIO(image_bytes)

    # Load the image with OpenCV
    imag = cv2.imdecode(np.frombuffer(image_stream.read(), np.uint8),
cv2.IMREAD_COLOR)
    img_from_ar = Image.fromarray(imag, 'RGB')
    resized_image = img_from_ar.resize((224, 224))
    test_image = np.expand_dims(resized_image, axis=0)
    test_image = test_image.astype('float32') / 255.0 # Normalize the image

    # Load model
    model = tf.keras.models.load_model('best_model.h5')
    result = model.predict(test_image)

    predicted_class_index = np.argmax(result)
    predicted_class_probability = result[0][predicted_class_index]
    # Set a threshold for class probability

    confidence_threshold = 0.1 # Adjust this threshold as needed

    # Check if any class probability exceeds the threshold
    print('probability' + str(predicted_class_probability))

    top_3_probabilities = {class_mapping[i]: result[0][i] * 100 for i in
range(len(class_mapping))}
    # Sort the dictionary by values in descending order
    sorted_probabilities = sorted(top_3_probabilities.items(), key=lambda x: x[1],
reverse=True)
    # Take only the top 3 entries
    class_probabilities = sorted_probabilities[:3]

    if predicted_class_probability < confidence_threshold:
        predicted_class_name = "Unknown"
    else:
        if predicted_class_index in class_mapping:
            predicted_class_name = class_mapping[predicted_class_index]
        else:
            predicted_class_name = "Unknown"

    print('index' + str(predicted_class_index))

```

```

context = {
    'class_probabilities': class_probabilities,
    'prediction': predicted_class_name,
    'probability': predicted_class_probability,
}

return render(request, "cam-recognition.html", context)

except Exception as e:
    # Handle errors gracefully
    return HttpResponseServerError(f"Error: {str(e)}")

return render(request, 'cam-recognition.html')def recognition(request):
message = ""

with open('class_mapping.json', 'r') as file:
    loaded_class_mapping_json = file.read()

class_mapping = json.loads(loaded_class_mapping_json)
class_mapping = {int(key): value for key, value in class_mapping.items()}
print(class_mapping)
# prediction =
fss = CustomFileSystemStorage()
try:
    image = request.FILES["image"]
    print("Name", image.name)
    _image = fss.save(image.name, image)
    path = str(settings.MEDIA_ROOT) + "/" + image.name
    # image details
    image_url = fss.url(_image)
    # Read the image
    imag = cv2.imread(path)
    img_from_ar = Image.fromarray(imag, 'RGB')
    resized_image = img_from_ar.resize((299, 299))

    test_image = np.expand_dims(resized_image, axis=0)
    test_image = test_image.astype('float32') / 255.0 # Normalize the image

    # Load model
    model = tf.keras.models.load_model('best_model.h5')

    result = model.predict(test_image)

    predicted_class_index = np.argmax(result)
    predicted_class_probability = result[0][predicted_class_index]

    # Set your confidence threshold (e.g., 0.7)
    confidence_threshold = 0.1
    print('probability' + str(predicted_class_probability))

    top_3_probabilities = {class_mapping[i]: result[0][i] * 100 for i in
range(len(class_mapping))}
    # Sort the dictionary by values in descending order
    sorted_probabilities = sorted(top_3_probabilities.items(), key=lambda x: x[1], reverse=True)

```

```

# Take only the top 3 entries
class_probabilities = sorted_probabilities[:3]

if predicted_class_probability < confidence_threshold:
    predicted_class_name = "Unknown"
else:
    if predicted_class_index in class_mapping:
        predicted_class_name = class_mapping[predicted_class_index]
    else:
        predicted_class_name = "Unknown"

print('index' + str(predicted_class_index))

return TemplateResponse(
    request,
    "recognition.html",
    {
        "probability": predicted_class_probability,
        "message": message,
        "image": image,
        "image_url": image_url,
        "prediction": predicted_class_name,
        'class_probabilities': class_probabilities,
    },
)
except MultiValueDictKeyError:
    return TemplateResponse(
        request,
        "recognition.html",
        {"message": "No Image Selected"},
    )

def cam_recognition(request):

    with open('class_mapping.json', 'r') as file:
        loaded_class_mapping_json = file.read()

    class_mapping = json.loads(loaded_class_mapping_json)
    class_mapping = {int(key): value for key, value in class_mapping.items()}

    if request.method == 'POST':
        image_data_uri = request.POST.get("src")

    try:
        # Extract the base64-encoded image data from the data URI
        _, image_data = image_data_uri.split(",", 1)
        image_bytes = base64.b64decode(image_data)

        # Create a BytesIO stream from the image data
        image_stream = io.BytesIO(image_bytes)

        # Load the image with OpenCV
        imag = cv2.imdecode(np.frombuffer(image_stream.read(), np.uint8),
                           cv2.IMREAD_COLOR)

```

```

img_from_ar = Image.fromarray(imag, 'RGB')
resized_image = img_from_ar.resize((224, 224))
test_image = np.expand_dims(resized_image, axis=0)
test_image = test_image.astype('float32') / 255.0 # Normalize the image

# Load model
model = tf.keras.models.load_model('best_model.h5')
result = model.predict(test_image)

predicted_class_index = np.argmax(result)
predicted_class_probability = result[0][predicted_class_index]
# Set a threshold for class probability

confidence_threshold = 0.1 # Adjust this threshold as needed

# Check if any class probability exceeds the threshold
print('probability' + str(predicted_class_probability))

top_3_probabilities = {class_mapping[i]: result[0][i] * 100 for i in
range(len(class_mapping))}
# Sort the dictionary by values in descending order
sorted_probabilities = sorted(top_3_probabilities.items(), key=lambda x: x[1],
reverse=True)
# Take only the top 3 entries
class_probabilities = sorted_probabilities[:3]

if predicted_class_probability < confidence_threshold:
    predicted_class_name = "Unknown"
else:
    if predicted_class_index in class_mapping:
        predicted_class_name = class_mapping[predicted_class_index]
    else:
        predicted_class_name = "Unknown"

print( 'index' + str(predicted_class_index))

context = {
    'class_probabilities': class_probabilities,
    'prediction': predicted_class_name,
    'probability': predicted_class_probability,
}

return render(request, "cam-recognition.html", context)

except Exception as e:
    # Handle errors gracefully
    return HttpResponseServerError(f"Error: {str(e)}")

return render(request, 'cam-recognition.html')

```

## Function for handlin Map Processing And Map Interaction

```
def herbal_map(request):
```

```

herbs = Herb.objects.all()
stores = Store.objects.all()
mapherbs = MapHerb.objects.all()

# Create a map
m = folium.Map(location=[6.918658, 122.077802], zoom_start=13, control_scale=True,
max_zoom=20, min_zoom=2, max_bounds=True)

# OpenTopoMap tile layer with attribution
folium.TileLayer('https://tile.opentopomap.org/{z}/{x}/{y}.png', name='OpenTopoMap',
attr='Map tiles by OpenTopoMap, under CC BY-SA 3.0').add_to(m)

folium.TileLayer('cartodbpositron').add_to(m) # CartoDB Positron
# Esri tile layer with attribution
esri_tile_url =
'http://server.arcgisonline.com/ArcGIS/rest/services/World_Street_Map/MapServer/tile/{z}/{y}/{x}'
esri_tile_attribution = 'Tiles © Esri'

folium.TileLayer(esri_tile_url, name='esri', attr=esri_tile_attribution).add_to(m)

herb_group = folium.FeatureGroup(name='Herbs')
store_group = folium.FeatureGroup(name='Stores')
herb_user_group = folium.FeatureGroup(name='User Uploads')

# Iterate over store objects and create markers
for store in stores:
    lat = store.lat
    long = store.long

    if lat is not None and long is not None:
        popup = folium.Popup(store.name)
        folium.Marker(
            location=[lat, long],
            popup=popup,
        ).add_to(store_group)

# Iterate over herb objects and create markers
for herb in herbs:
    lat = herb.lat
    long = herb.long

    if lat is not None and long is not None:
        popup = folium.Popup(herb.name)
        folium.CircleMarker(
            location=[lat, long],
            radius=20,
            color='green',
            fill=True,
            fill_color='green',
            fill_opacity=0.6,
            popup=popup,
        ).add_to(herb_group)

for herb in mapherbs:

```

```

lat = herb.lat
long = herb.long
name = herb.herb

if lat is not None and long is not None:
    popup = folium.Popup(name)
    folium.CircleMarker(
        location=[lat, long],
        radius=20,
        color='yellow',
        fill=True,
        fill_color='yellow',
        fill_opacity=0.4,
        popup=popup,
    ).add_to(herb_user_group)

# Add feature groups to the map
herb_group.add_to(m)
store_group.add_to(m)
herb_user_group.add_to(m)

# Add layer control to the map
folium.LayerControl().add_to(m)

# Get the map HTML
map_html = m._repr_html_()
context = {
    'map': map_html,
    'herbs': herbs,
    'store': store,
    'mapherbs': mapherbs,
}
}

return render(request, 'herbal-map.html', context)

def herbal_map_inter(request, id=None, herb=None):
    herbs = Herb.objects.all()
    stores = Store.objects.all()
    mapherbs = MapHerb.objects.all()
    herbb = None
    mapherb = None

    try:
        herbb = Herb.objects.get(id=id)
    except Herb.DoesNotExist:
        try:
            mapherb = MapHerb.objects.get(herb=herb)
        except MapHerb.DoesNotExist:
            pass

    form = MapCommentForm()
    if request.method == 'POST':
        if request.user.is_authenticated:
            form = MapCommentForm(request.POST)

```

```

if form.is_valid():
    print('Form is valid') # Add this line for debugging
    mherb = form.save(commit=False)
    mherb.username = request.user
    mherb.map_herb = mapherb
    mherb.save()
    print('Data saved successfully') # Add this line for debugging
    return redirect(request.path + '?submitted=true')
else:
    print('Form is not valid') # Add this line for debugging
    print(form.errors) # Add this line for debugging
else:
    return redirect('login_or_register')

# Create a map
if herbb and herbb.lat is not None and herbb.long is not None:
    m = folium.Map(location=[herbb.lat, herbb.long], zoom_start=16, control_scale=True,
max_zoom=20, min_zoom=2, max_bounds=True)
    elif mapherb and mapherb.lat is not None and mapherb.long is not None:
        m = folium.Map(location=[mapherb.lat, mapherb.long], zoom_start=16, control_scale=True,
max_zoom=20, min_zoom=2, max_bounds=True)
    else:
        m = folium.Map(location=[6.918658, 122.077802], zoom_start=13, control_scale=True,
max_zoom=20, min_zoom=2, max_bounds=True)
        messages.info(request, 'Location is Not Available')

# OpenTopoMap tile layer with attribution
folium.TileLayer('https://tile.opentopomap.org/{z}/{x}/{y}.png', name='OpenTopoMap',
attr='Map tiles by OpenTopoMap, under CC BY-SA 3.0').add_to(m)

folium.TileLayer('cartodbpositron').add_to(m) # CartoDB Positron
# Esri tile layer with attribution
esri_tile_url =
'http://server.arcgisonline.com/ArcGIS/rest/services/World_Street_Map/MapServer/tile/{z}/{y}/{x}'
esri_tile_attribution = 'Tiles © Esri'

folium.TileLayer(esri_tile_url, name='esri', attr=esri_tile_attribution).add_to(m)

herb_group = folium.FeatureGroup(name='Herbs')
store_group = folium.FeatureGroup(name='Stores')
herb_user_group = folium.FeatureGroup(name='User Uploads')

for herbb in herbs:
    lat = herbb.lat
    long = herbb.long
    job_name = herbb.name

    if lat is not None and long is not None:
        marker = folium.CircleMarker(location=[lat,
long],radius=20,color='green',fill=True,fill_color='green',
fill_opacity=0.6, popup=job_name)
        marker.add_to(herb_group)

# Iterate over store objects and create markers

```

```

for store in stores:
    lat = store.lat
    long = store.long

    if lat is not None and long is not None:
        popup = folium.Popup(store.name)
        folium.Marker(
            location=[lat, long],
            popup=popup,
        ).add_to(store_group)

for herbb in mapherbs:
    lat = herbb.lat
    long = herbb.long
    name = herbb.herb

    if lat is not None and long is not None:
        popup = folium.Popup(name)
        folium.CircleMarker(
            location=[lat, long],
            radius=20,
            color='yellow',
            fill=True,
            fill_color='yellow',
            fill_opacity=0.1,
            popup=popup,
        ).add_to(herb_user_group)

# Add feature groups to the map
herb_group.add_to(m)
store_group.add_to(m)
herb_user_group.add_to(m)

# Add layer control to the map
folium.LayerControl().add_to(m)

# Render the map to HTML
context = {
    'map': m._repr_html_(),
    'herbs': herbs,
    'herbb': herbb,
    'mapherbs': mapherbs,
    'mapherb': mapherb,
    'form': form,
}
}

return render(request, 'herbal-map.html', context)

```

### **Function for handling dataset processing and dataset preparation with an addition of automated Processed dataset download**

```

def download_processed_data(request):
    media_root = settings.MEDIA_ROOT
    herbs_path = os.path.join(media_root, "herbs.npy")

```

```

labels_path = os.path.join(media_root, "labels.npy")

# Create a zip file to download multiple files
response = HttpResponse(content_type='application/zip')
response['Content-Disposition'] = 'attachment; filename=processed_data.zip'

with ZipFile(response, 'w') as zip_file:
    zip_file.write(herbs_path, 'herbs.npy')
    zip_file.write(labels_path, 'labels.npy')

return response

@csrf_exempt
@staff_member_required(login_url='login_or_register')
def process_images(request):

    media_root      =      settings.MEDIA_ROOT
    root_directory = os.path.join(media_root, "Datasets")

    try:
        data   =   []
        labels = []

        subdirectories = [d for d in os.listdir(root_directory) if
os.path.isdir(os.path.join(root_directory, d))]

        for label, subdirectory in enumerate(subdirectories):
            subdirectory_path = os.path.join(root_directory, subdirectory)
            image_files = os.listdir(subdirectory_path)

            class_mapping = {}
            for i, subdirectory in enumerate(subdirectories):
                class_mapping[i] = subdirectory

            class_mapping_json = json.dumps(class_mapping, indent=4)
            with open('class_mapping.json', 'w') as file:
                file.write(class_mapping_json)

            for image_file in image_files:
                image_path = os.path.join(subdirectory_path, image_file)
                # Check if the image is readable
                imag = cv2.imread(image_path)
                if imag is None:
                    print(f"Skipping unreadable image: {image_path}")
                    continue

                img_from_ar = Image.fromarray(imag, 'RGB')
                resized_image = img_from_ar.resize((224, 224))
                data.append(np.array(resized_image))
                labels.append(label)

            herbs   =   np.array(data)
            labels = np.array(labels)

            # Save the processed data and labels

```

```

        np.save(os.path.join(media_root, "herbs.npy"), herbs)
        np.save(os.path.join(media_root, "labels.npy"), labels)

    return JsonResponse({'status': 'success', 'message': 'Images processed successfully.'})

except Exception as e:
    return JsonResponse({'status': 'error', 'message': str(e)})


@staff_member_required(login_url=reverse_lazy('login_or_register'))
def dataset_upload(request):
    warning_message = None
    class_name = None
    images = None

    datasets = Datasets.objects.order_by('class_name')

    try:
        if request.method == 'POST':
            class_name = request.POST.get('class_name')
            images = request.FILES.getlist('images')

            if not images:
                print('redirect')
                warning_message = {'message': 'Please Add Images'}
                return JsonResponse(warning_message)
            else:
                if class_name and images is not None:
                    # print('passed')# print(images)# print(class_name)
                    # class_folder = os.path.join(settings.MEDIA_ROOT, class_name)
                    # os.makedirs(class_folder, exist_ok=True)
                    dataset_instance = Datasets(class_name=class_name)
                    dataset_instance.save()

                    for i, image in enumerate(images):
                        try:
                            dataset_instance_images = DatasetImages(class_name=dataset_instance,
                                                       images=image)
                            dataset_instance_images.save()
                        except DuplicateImageError as e:
                            # Handle the duplicate image error, you can redirect or render an error message
                            warning_message = {'message': str(e)}
                            dataset_instance.delete()
                            # dataset_instance_images.delete()
                            return JsonResponse(warning_message)

                    print("Preparation Success")
                    response_data = {'progress': 100, 'message': 'Upload complete'}
                    return JsonResponse(response_data)

                except Exception as e:
                    print(f"Error during dataset upload: {str(e)}")

    return render(request, 'Admin/dataset-upload.html', {'progress_percentage': 0, 'datasets': datasets})

```

## Function for handling model upload and model Training

```
def          model_upload(request):
    json_file_path = 'training_history.json'

    try:
        with open(json_file_path, 'r') as file:
            training_history_data = json.load(file)
    except FileNotFoundError:
        training_history_data = None
    try:
        if request.method == 'POST':
            model_file      = request.FILES.get('model')
            train_accu     = request.POST.get('training_accuracy')
            val_accu       = request.POST.get('validation_accuracy')
            val_loss       = request.POST.get('validation_loss')
            num_classes    = request.POST.get('classes')
            data_length    = request.POST.get('class_length')

            # Validate if model file is provided
            if not model_file:
                return JsonResponse({'message': 'Model file is required.'})

            # Validate numeric values
            try:
                train_accu   = float(train_accu)
                val_accu     = float(val_accu)
                val_loss     = float(val_loss)
                num_classes  = int(num_classes)
                data_length  = int(data_length)
            except (TypeError, ValueError):
                return JsonResponse({'message': 'Invalid numeric value provided.'})

            temp_model_path = f'tmp_model_{model_file.name}'
            # Save the uploaded model temporarily
            with open(temp_model_path, 'wb') as temp_model_file:
                for chunk in model_file.chunks():
                    temp_model_file.write(chunk)
            # Load the model directly from the temporary path
            model = tf.keras.models.load_model(temp_model_path)
            # Save the loaded model as an HDF5 file
            model.save('best_model.h5')
            # Remove the temporary model file
            os.remove(temp_model_path)

            data = {
                'training_accuracy':      train_accu,
                'validation_accuracy':    val_accu,
                'validation_loss':        val_loss,
                'num_classes':           num_classes,
                'data_length':            data_length,
```

```

        }

        with open('training_history.json', 'w') as json_file:
            json.dump(data, json_file)

        response_data = {'progress': 100, 'message': 'Upload Completed'}
        return JsonResponse(response_data)

    except Exception as e:
        print(f"Error: {str(e)}")

    return render(request, 'Admin/model-upload.html', {'training_history_data': training_history_data})

@staff_member_required(login_url=reverse_lazy('login_or_register'))
def model_training(request):

    try:
        if request.method == 'POST':
            label_file = request.FILES.get('label')
            herb_file = request.FILES.get('herb')

            if not label_file or not herb_file:
                # Handle the case where files are missing
                response_data = {'message': 'Please Upload Both Files'}
                return JsonResponse(response_data)

            train_model(herb_file, label_file)

            response_data = {'message': 'Training Completed'}
            return JsonResponse(response_data)

    except Exception as e:
        print(f"Error: {str(e)}")

    return render(request, 'Admin/model-training.html')

import numpy as np
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers, models
from tensorflow.keras.applications import VGG16
import matplotlib.pyplot as plt
import json

# Load your data
def train_model(label_file, herb_file):
    herbs = np.load(label_file)
    labels = np.load(herb_file)

    # Shuffle and preprocess the data
    s = np.arange(herbs.shape[0])
    np.random.shuffle(s)
    herbs = herbs[s]
    labels = labels[s]

```

```

num_classes      = len(np.unique(labels))
data_length = len(herbs)

class_mapping = {i: str(label_name) for i, label_name in enumerate(np.unique(labels))}

# Split the data into training and testing sets
(x_train, x_test) = herbs[(int)(0.1 * data_length):], herbs[:int)(0.1 * data_length)]
x_train = x_train.astype('float32') / 255
x_test = x_test.astype('float32') / 255
train_length = len(x_train)
test_length = len(x_test)

(y_train, y_test) = labels[(int)(0.1 * data_length):], labels[:int)(0.1 * data_length)]

# Define the EarlyStopping callback
early_stopping = keras.callbacks.EarlyStopping(
    monitor='val_accuracy',
    min_delta=0.001,
    patience=1,
    verbose=1,
    restore_best_weights=True,
)

# Add ModelCheckpoint callback for saving the best model
model_checkpoint = keras.callbacks.ModelCheckpoint(
    filepath="best_model.h5",
    monitor="val_accuracy",
    save_best_only=True,
    verbose=1,
)

base_model = VGG16(weights='imagenet', include_top=False, input_shape=(224, 224, 3))

# Freeze the weights of the pre-trained layers
for layer in base_model.layers:
    layer.trainable = False

# Create a new model on top of the ConvNeXtXLarge base
model = models.Sequential()
model.add(base_model)
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(num_classes, activation='softmax'))

# Compile the model
model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])

# Train the model
history = model.fit(x_train, y_train, epochs=100,
                     validation_data=(x_test, y_test), callbacks=[early_stopping, model_checkpoint])

# Unfreeze some layers for fine-tuning

```

```

for layer in base_model.layers[-10:]:
    layer.trainable = True

# Recompile the model after unfreezing some layers
optimizer = tf.keras.optimizers.Adam(learning_rate=1e-5)
model.compile(optimizer=optimizer,
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])

fine_tune_epochs = 1 # You need to define this variable

history_fine_tune = model.fit(x_train, y_train, epochs=fine_tune_epochs,
                               validation_data=(x_test, y_test), callbacks=[early_stopping,
model_checkpoint])

plt.plot(history.history['accuracy'], label='accuracy')
plt.plot(history.history['val_accuracy'], label='val_accuracy')
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.ylim([0.5, 1])
plt.legend(loc='lower right')

test_loss, test_acc = model.evaluate(x_test, y_test, verbose=2)
print(test_acc)

model.save("model_with_transfer_learning.h5")

save_training_history(history, "training_history.json", num_classes, data_length)
save_class_history(class_mapping, "class_history.json")

def save_training_history(history, filename, num_classes, data_length):
    history_dict = {
        "training_accuracy": history.history['accuracy'][-1],
        "validation_accuracy": history.history['val_accuracy'][-1],
        "training_loss": history.history['loss'][-1],
        "validation_loss": history.history['val_loss'][-1],
        "num_classes": num_classes,
        "data_length": data_length,
    }

    with open(filename, 'w') as file:
        json.dump(history_dict, file, indent=4)

def save_class_history(class_mapping, filename):

    class_mapping_str = {str(key): str(value) for key, value in class_mapping.items()}

    history_dict = {
        "class_mapping": class_mapping_str,
    }

```

## APPENDIX C

### TESTING

The Capstone Project proponents created 10 sample registered users. All users tested the search functionality of the system by searching plant names or specific health concerns. By searching any health concerns, the system will display the plants that are related or effective to the symptoms-based search. 3 out of 10 users used the mapping to locate some of the herbal plants.

First Name	Last Name	Type of Search (P=Plant/S=Symptoms/ L=Locate)	Plant Name(s) Searched	Symptoms/Health Concern(s) Searched	Mapping (Plant(s) Located)
Khyniemar	Hassan	P	Kangkong	Overall Health	
Haniemar	Hassan	S	Lagundi	Cough	
Tima	San	S	Ampalaya, Banaba	Blood Sugar Regulation	
Yazmin	Arellano	L			
Dessirey	Barro	P	Mabolo, Gumamela	Anti-diabetic, Blood Pressure	
Carmela	Wadja	L			
Faizer	Habibon	S		Headaches	
Raymond	Pobletin	L			

Azenei	Dakmood	S	Gumamela, Celery	Blood Pressure	
Anie	Acedo	P	Tsaang Gubat	Diarrhea/Stomach Problems	

The following table contains the test result summary of the various test cases done to the different modules within the system.

<b>Test Report</b>		
Executed	Passed	122
		0
		122
Failed		0
(Total) Tests Executed (Passed + Failed)		
In Progress		0
Blocked		0
Completed		100%

Functions	Description	% TCs Executed	% TCs Passed	TCs Pending	Priority	Remarks
Register/Sign Up	Create a new account for the user	100%	100%	0	High	
Sign Up/Registration	Enter a valid name, valid email, matching password and	100%	100%	0	Normal	
SignUp/Registration	Wait to receive the email for account verification	100%	100%	0	Normal	
SignUp/Registration	Verify account after receiving the email.	100%	100%	0	Normal	
Verify the log-in	Enter a valid email in the email field.	100%	100%	0	Normal	
Verify user role	enter a valid name and password	100%	100%	0	Normal	

Verify admin role	Enter valid name and password	100%	100%	0	High	
Verify the log-in	Enter valid email and an invalid password	100%	100%	0	Normal	
Verify the log-in	Enter invalid email and an invalid password	100%	100%	0	Normal	
Verify the log-in	Enter blank email and a password	100%	100%	0	Normal	
Verify the log-in	Enter valid email and a blank password	100%	100%	0	Normal	
Verify the log-in	Enter blank email and a blank password	100%	100%	0	Normal	
Enter password	Verify if password is masked	100%	100%	0	Normal	

Enter an existing email account	Verify forgot password	100%	100%	0	Normal	
Enter a non-existent account	Pop up message for invalid email used	100%	100%	0	Normal	
Enter a valid email	Verify that users will receive a link in their mail for resetting their password	100%	100%	0	Normal	
Logged In Successfully	Displaying Homepage/Back to Site	100%	100%	0	Normal	
Explore Homepage	Click first “More” option or move the mouse to explore	100%	100%	0	Normal	
Herbal Library	Click Herbal Plants on the option above to view more herbal plants	100%	100%	0	Normal	
View Herbal	Choose and enter one of the plants	100%	100%	0	Normal	
View Herbal	See details by clicking one	100%	100%	0	Normal	

View Herbal	Read or learn herbal information	100%	100%	0	Normal	
View Herbal	See Medical Conditions	100%	100%	0	Normal	
View Herbal	See Medicinal Properties	100%	100%	0	Normal	
View Herbal	See Potential Side Effects/Allergic reaction	100%	100%	0	Normal	
View Herbal	See inside the Recently Uploaded other herbal Plants	100%	100%	0	Normal	
View Herbal	See reference and tools	100%	100%	0	Normal	
View Herbal	See Testimonials	100%	100%	0	Normal	
View Herbal	Leaving Comments on testimonial	100%	100%	0	Normal	
View Herbal	Submit the testimonial	100%	100%	0	Normal	

View Herbal	See other comments and feedback	100%	100%	0	Normal	
Add Favorite	From each herbal plant details you can add your favorite plant on the lower right button	100%	100%	0	Normal	
Account	It has User Menu	100%	100%	0	Normal	
User Menu	On user menu displays favorite herbal plants	100%	100%	0	Normal	
Favourites	Displays favorite herbal plants	100%	100%	0	Normal	
Favourites	Each plant displayed form favorite has remove button and view herb to view details	100%	100%	0	Normal	
Favourites	Each plants has displayed date	100%	100%	0	Normal	
Uploads	Uploads displays Title	100%	100%	0	Normal	

Uploads	Uploads displays Latitude and longitude	100%	100%	0	Normal	
Uploads	Uploads displays Timestamp	100%	100%	0	Normal	
Uploads	Uploads displays the Image Attachment	100%	100%	0	Normal	
Action	Uploads consist of two actions	100%	100%	0	Normal	
Actions	Edit and delete button	100%	100%	0	Normal	
Image Search	It is where the image processing process	100%	100%	0	Normal	
Image Search	It has two options, to capture or upload	100%	100%	0	Normal	
Image Capture	The system will process the image through using the device's camera to capture the plant(s)	100%	100%	0	Normal	

Take Photo	Capturing the plant using the device's camera to classify the plant	100%	100%	0	Normal	
Classify	Shows the output of the image processing, the plant's label/name, prediction, and its probability	100%	100%	0	Normal	
See More(Capture)	To view more details about the plant that was being captured	100%	100%	0	Normal	
Image Upload	The system will process only process the image recognition from the existing image that was already stored from any devices	100%	100%	0	Normal	
Choose File	Upload the image from the file device	100%	100%	0	Normal	
Predict	Processing the image to give the output such as: Plant labels, image name, prediction, and Probability.	100%	100%	0	Normal	
See More(Upload)	To view more details about the plant that was being recognized by the system	100%	100%	0	Normal	

Herbal Map	Interactive map to locate herbal plants	100%	100%	0	Normal	
Herbal Map	It has zoom in and out button to view the location conveniently	100%	100%	0	Normal	
Herbal Map	Use the green circle that is being displayed on the map to view location	100%	100%	0	Normal	
View List	List of herbal plants that can be locate on the herbal map	100%	100%	0	Normal	
Add New Herbal Location	Users can upload new herbal plant's location on herbal map	100%	100%	0	Normal	
Add Name	Input herbal name	100%	100%	0	Normal	
Add new location	Users should input new herbal name location after adding the herbal name	100%	100%	0	Normal	
Admin Dashboard	Shows statistics	100%	100%	0	Normal	
Admin User	Admin can manage the users and sees the users list	100%	100%	0	Normal	

Herbal Uploads (Admin user)	Admin can manage the herbal plants that has being uploaded to the system	100%	100%	0	Normal	
User Herbal Comments	Admin can managed comments onherbal map and herbal details	100%	100%	0	Normal	
Admin User	Managed the CCN Model and Dataset compiler	100%	100%	0	Normal	
Interactive Map (Admin)	Admin can view the Map are chart (Herbs/User Uploaded Herbs/Stores)	100%	100%	0	Normal	

The following table contains the test result summary of the various modules within the system.

Test Result						
Executed	Passed					104
	Failed					0
	Total Tests Executed					104
Module	Description	% TCs Executed	% TCs Passed	TCs Pending	Priority	Remarks

Log in	Sign into existing account	100%	100%	0	High	
Registration	Create an account for the new user	100%	100%	0	High	
Account	User Profile, Add Favorite, & Symptom feedback	100%	100%	0	High	
User	List of users in the system	100%	100%	0	High	
Homepage	View, Explore, Locate	100%	100%	0	High	
Herbal Plants Library	View, Read, Add to Favorite, Testimonials, & Comment	100%	100%	0	High	
Image Processing	Upload, Predict, Capture, Classify	100%	100%	0	High	
Herbal Map	Search, Locate, Upload File, Add Location	100%	100%	0	High	
Admin Dashboard	View Statistics, User List, Comments, Chart	100%	100%	0	High	

## Beta Testing Result

Tester Name: Genelyn G. Madjos, Lady Jane G. Morilla, Kean Roe Felipe Mazo			
Module	Description & Function	Pass/Fail	Usability (1-5 rating)
Log in	Sign into existing account	Pass	5

Registration	Create an account for the new user	Pass	5
Account	User Profile, Add Favorites, & Symptom feedback		
	Does it accept user input? (ex. address)	Pass	5
	Does it make the user leave comments after reading herbal plant details?	Pass	5
	Does it have add favorite function?	Pass	5
	Does the complaints field accept both letters and numbers input?	Pass	5
Homepage	View, Explore, Locate		
	Does the homepage shows you the small details about the system?	Pass	5
	Does the system shows you the recommendations of herbal plants?	Pass	5
	Does the system displays the experts?	Pass	5
	Does the system shows you the interactive map?	Pass	5
Herbal Plants Library	View, Read, Add to Favorite, Testimonials, & Comment		
	Does the system shows you the library of herbal plants?	Pass	5
	Does the system display the DOH approved plants?	Pass	5
	Does the system have details of each herbal plants that is displayed on the library ?	Pass	5

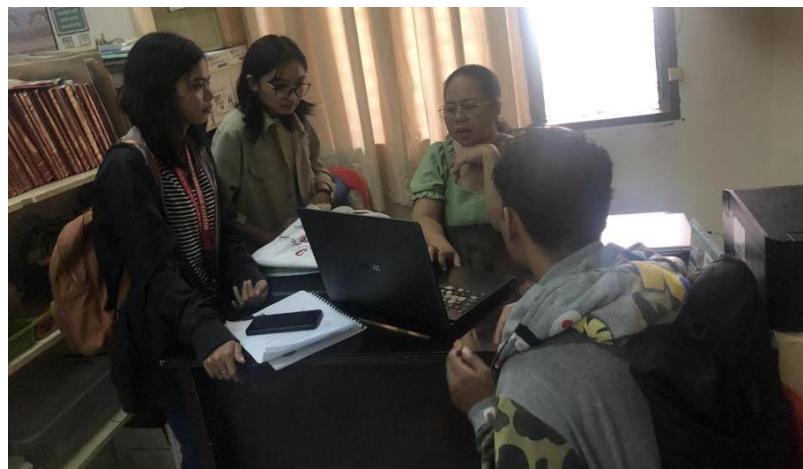
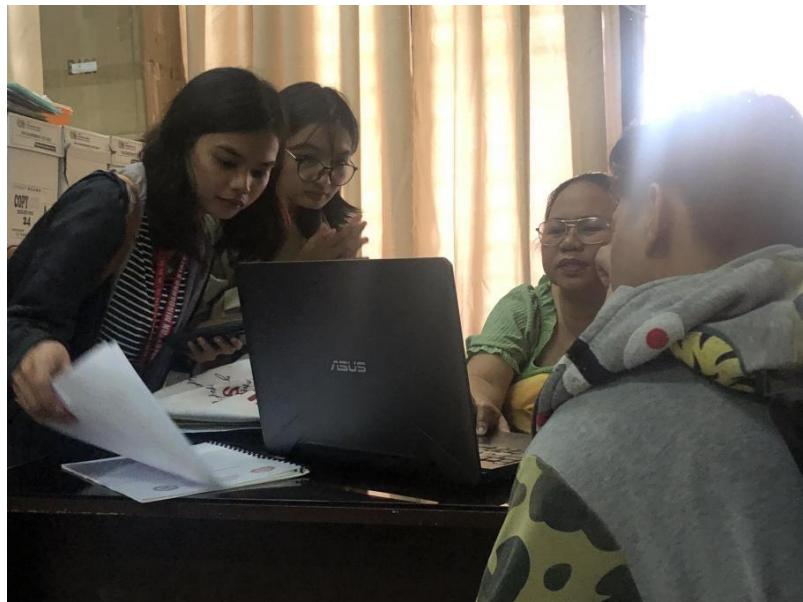
	Does it allow you to read and leave comments on testimonials?	Pass	5
Image Processing	Upload, Predict, Capture, Classify		
	Does the system allow user to upload existing image?	Pass	5
	Does the system predict after the image was uploaded?	Pass	5
	Does the system allow users to capture plants?	Pass	5
Herbal Map	Search, Locate, Upload File, Add Location		
	Does the system provide locating function for the herbal plants or herbal stores on herbal map?	Pass	5
	Does it make user upload new herbal images with herbal names and locations?	Pass	5
Admin Dashboard	View Statistics, User List, Comments, Chart		
	Does clicking the “Dashboard” shows the admin user the statistic	Pass	5
	Does the User List shows the list of the system’s user accounts	Pass	5

## **Photo and Video Documentation**

November 29, 2023



*Consultation meeting with IT Capstone 2 Adviser*



*Live Demo and Beta Testing with the Experts*

## APPENDIX D EVALUATION FORM

<b>Tester Name: Genelyn G. Madjos, Lady Jane G. Morilla, Kean Roe Felipe Mazo</b>			
Module	Description & Function	Pass/Fail	Usability (1-5 rating)
Log in	Sign into existing account	Pass	5
Registration	Create an account for the new user	Pass	5
Account	User Profile, Add Favorites, & Symptom feedback		
	Does it accept user input? (ex. address)	Pass	5
	Does it make the user leave comments after reading herbal plant details?	Pass	5
	Does it have add favorite function?	Pass	5
	Does the complaints field accept both letters and numbers input?	Pass	5
Homepage	View, Explore, Locate		
	Does the homepage shows you the small details about the system?	Pass	5
	Does the system shows you the recommendations	Pass	5

	of herbal plants?		
	Does the system displays the experts?	Pass	5
	Does the system shows you the interactive map?	Pass	5
Herbal Plants Library	View, Read, Add to Favorite, Testimonials, & Comment		
	Does the system shows you the library of herbal plants?	Pass	5
	Does the system display the DOH approved plants?	Pass	5
	Does the system have details of each herbal plants that is displayed on the library ?	Pass	5
	Does it allow you to read and leave comments on testimonials?	Pass	5
Image Processing	Upload, Predict, Capture, Classify		
	Does the system allow user to upload existing image?	Pass	5
	Does the system predict after the image was uploaded?	Pass	5

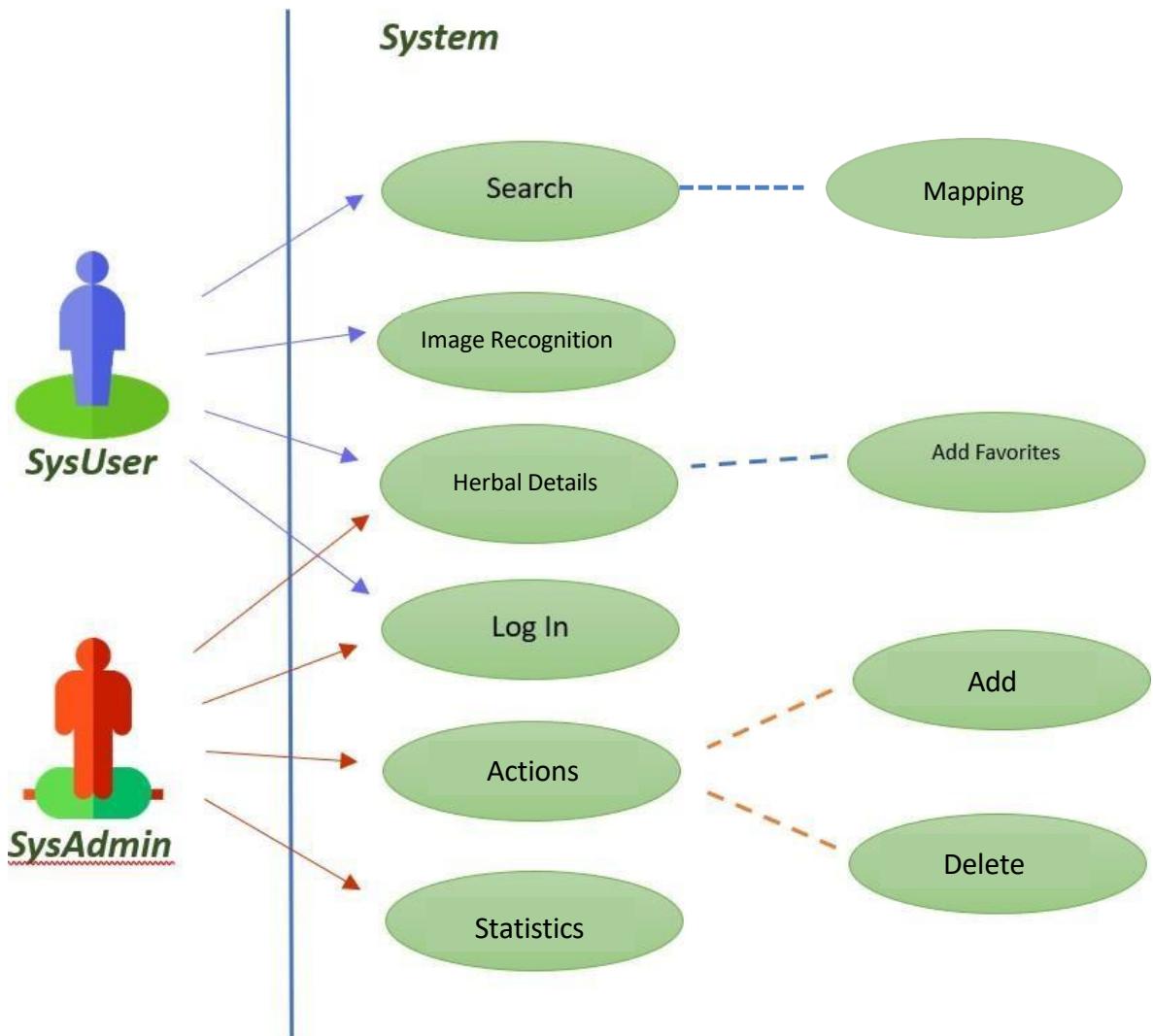
	Does the system allow users to capture plants?	Pass	5
Herbal Map	Search, Locate, Upload File, Add Location		
	Does the system provide locating function for the herbal plants or herbal stores on herbal map?	Pass	5
	Does it make user upload new herbal images with herbal names and locations?	Pass	5
Admin Dashboard	View Statistics, User List, Comments, Chart		
	Does clicking the “Dashboard” shows the admin user the statistic	Pass	5
	Does the User List shows the list of the system’s user accounts	Pass	5

## APPENDIX E GANTT CHART

### CURAHERB

Task ID	Task Description	Task Duration	Start Date	End Date	JUNE	JULY	AUG	SEPT	OCT	NOV
1	Capstone Project Title submissions(individual)	5	June 14, 2023	June 18, 2023						
2	Capstone Chapters 1-4 submission	5	June 21, 2023	June 25, 2023						
3	Capstone Chapters 1-4 revisions	5	June 26, 2023	June 30, 2023						
4	Mock Defense Deliverable	7	July 1, 2023	July 7, 2023						
5	Week 4 Progress Report	5	July 8, 2023	July 12, 2023						
6	Initial Title Defense	8	July 13, 2023	July 17, 2023						
7	Week 8 Progress Report	1	July 18, 2023	July 18, 2023						
8	Submission of Defense Deliverable	2	July 19, 2023	July 20, 2023						
9	Proposal Defense	5	July 21, 2023	July 21, 2023						
10	Finalization of the Design GUI	12	August 9, 2023	August 20, 2023						
11	QR Generator Module	12	August 21, 2023	September 1, 2023						
12	CuraHerb: Intelligent Herbal Plant Recognition and Medication Guide	12	September 2, 2023	September 13, 2023						
13	Listing of Herbal Plants	12	September 14, 2023	September 25, 2023						
14	Finalized Database Idea to input on the system	12	September 26, 2023	October 7, 2023						
15	Scheduling the date to Capture Herbal Plants using mobile phone	10	October 8, 2023	October 17, 2023						
16	Assigned to gather Informations for each Herbal Plants	12	October 18, 2023	October 20, 2023						
17	Distribute Plants to gather more informations	7	November 1, 2023	November 7, 2023						
18	Finalized the list of Herbal Plants	1	November 8, 2023	November 8, 2023						
19	Managed to Capture Herbal Plants	1	November 9, 2023	November 9, 2023						
20	Test Planning	4	November 10, 2023	November 14, 2023						
21	Gather more herbal plants photos	4	November 15, 2023	November 19, 2023						
22	Meeting with Herbalist/Botanist	2	November 20, 2023	November 21, 2023						
23	Herbal approval by Expert	2	November 22, 2023	November 23, 2023						
24	Integration Testing	1	November 24, 2023	November 24, 2023						
25	System Testing	1	November 25, 2023	November 25, 2023						
26	Acceptance Testing	1	November 26, 2023	November 26, 2023						
27	Rebuilding some modules	1	November 27, 2023	November 27, 2023						
28	Deploy the project	1	November 28, 2023	November 28, 2023						
29	Identify Bugs	1	November 29, 2023	November 29, 2023						
30	Implement the modification	1	November 30, 2023	November 30, 2023						

## APPENDIX F USE CASES



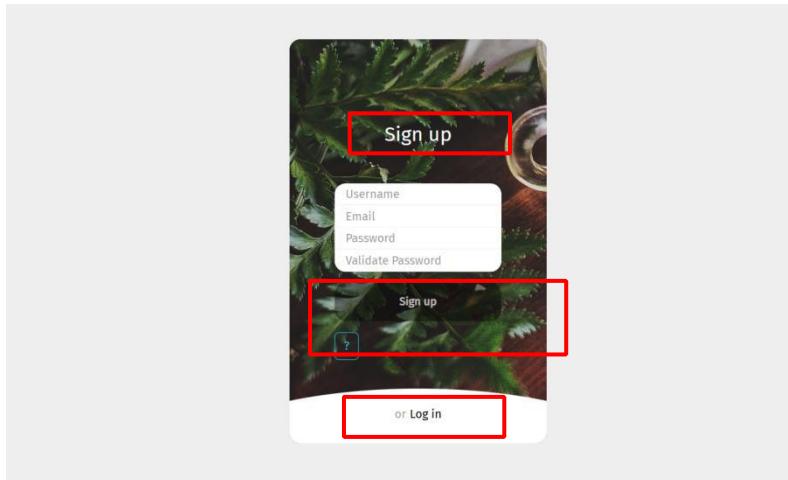
The diagram above shows that the system user can search, read details of the desired herbal plants in herbal library, and add any herbal plants to the list of their favorites. The system user can also use the mapping to locate herbal plants. All the functions except viewing such as herbal details cannot be access by the user if its not going to log in. On the diagram above, after logging in to admin's account, it can now access the the statistics, and can manage actions such as add and delete.

**APPENDIX F**  
**CLIENT ACCEPTANCE FORM**

## **APPENDIX H**

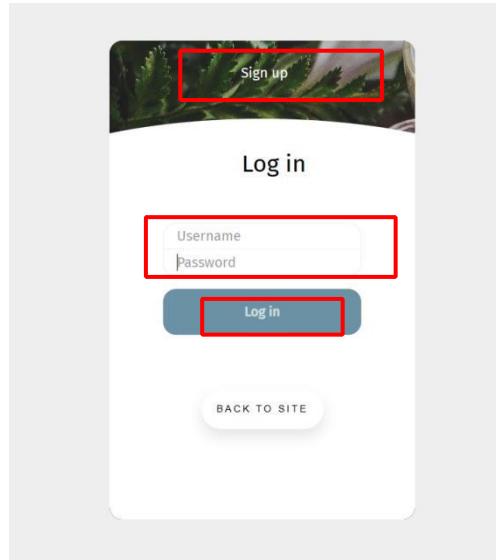
### **USER GUIDE**

1. The system shall allow the citizen users to create an account.



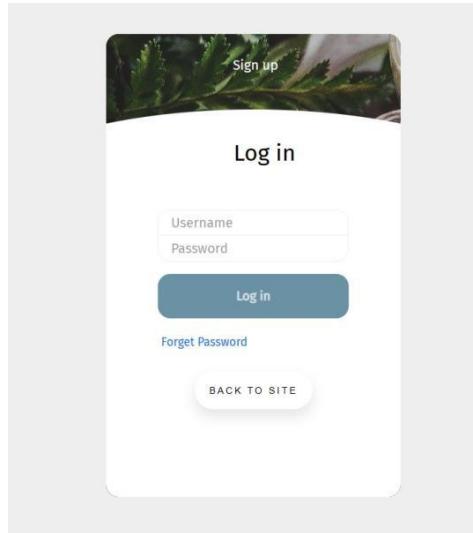
1. The user will see a screen with the login form. For this process let's focus on the register. Click the "Sign Up" in the navigation bar to create an account.
2. Enter the information required on all of the fields and click "Sign Up" to create an account and this will save it in the system.
3. At the bottom of the registration page there is a "Log in" which will lead the user to the login page if they already have an existing account.
4. Enter the information required on all of the fields and click "Sign Up" to create an account and this will save it in the system.
5. At the bottom of the registration page there is a "Log in" which will lead the user to the login page if they already have an existing account.

2. The system shall allow all users to log in.



1. All of the users will log in using their email and password.
2. Click the "Log in" button to proceed. All of the information will be validated by the system if the user left the area blank. An error tool tip will be displayed to inform the user that the email and password fields need to be filled out.
3. At the top of the login page there is a "Sign up here" which will lead the user to the registration page if they don't have an account

3. The system shall allow users to recover their accounts.

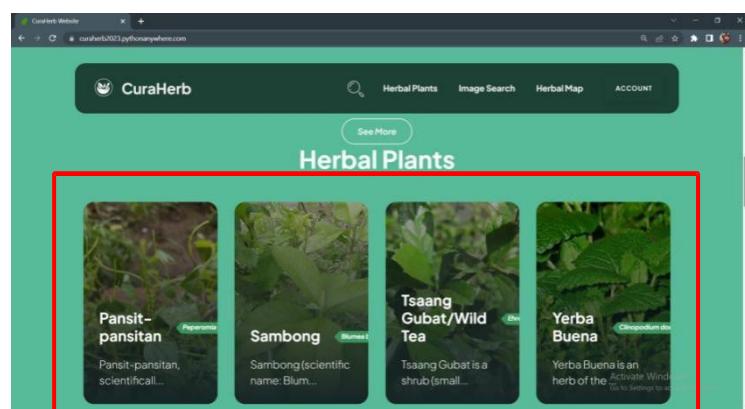
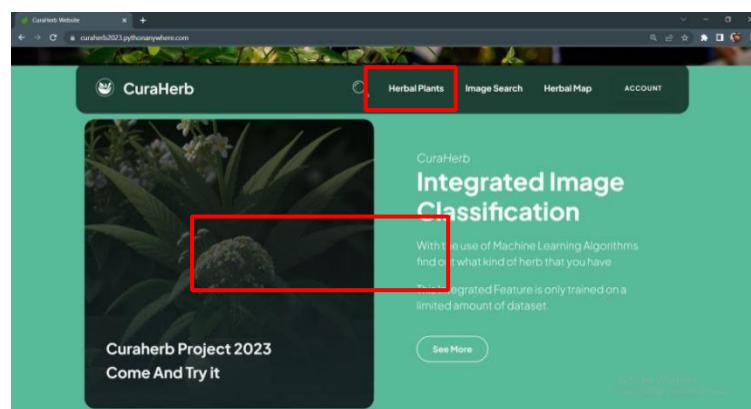


1. Clicking the “forgot password” on the log-in page will allow users to recover their account.

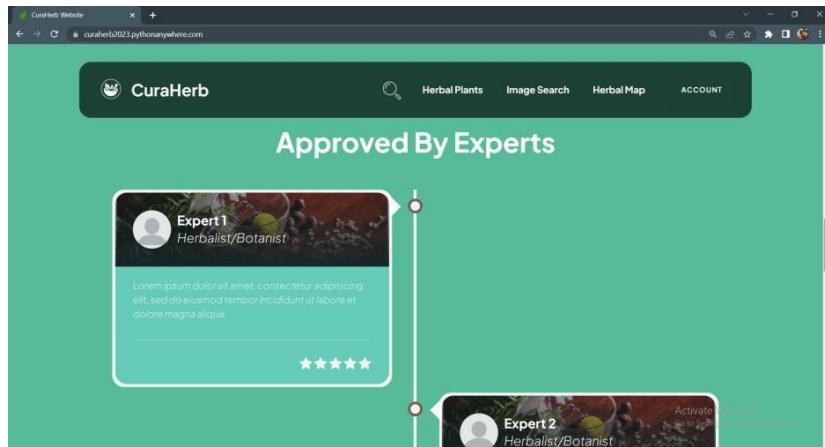
2. Enter your email and click on the “Reset password” button. If the field is left blank, it will produce an error tool tip that says “Please fill out this field”.
3. After clicking the reset button, it will redirect the user to the page where they could input their new preferred password.
4. After successfully confirming the changed password. You can now log in with your email account and new password.

## User (Logged In) - Homepage

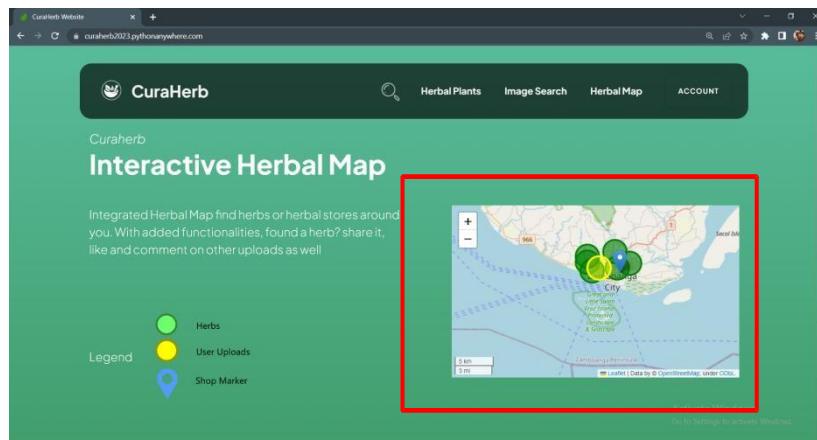
5.



Arriving at the home page, the user will be able to see the Sampung Halamang Gamot approved by the DOH, stored in homepage where the information of each herbal included. There is also a search bar to find your preferred herbal plant.

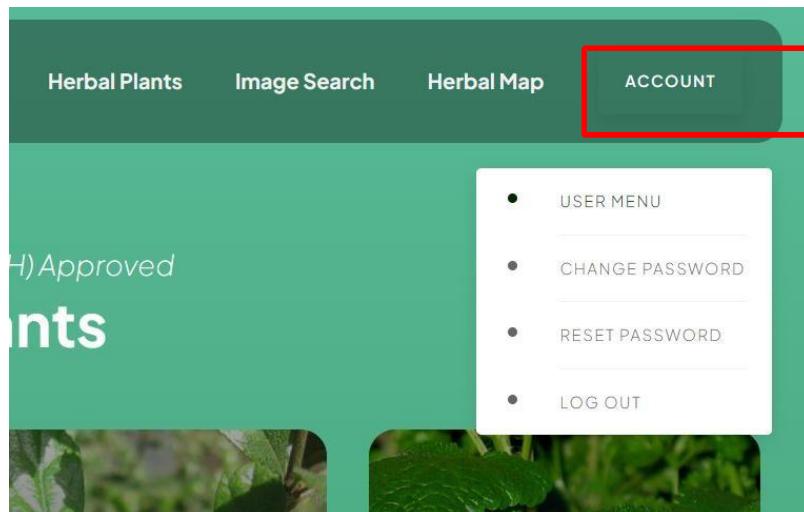


6. Scrolling down and you'll be able to view the profile each of the Herbal expert who help to identify our herbal plants.



7. At the bottom of the homepage , you'll see the Interactive Herbal Map. This Map will help the users to locate near herbal plants and Herbal shop in the city. They can also upload image of the herbal plant they found and pin a location, to help future researcher locate herbal plants.

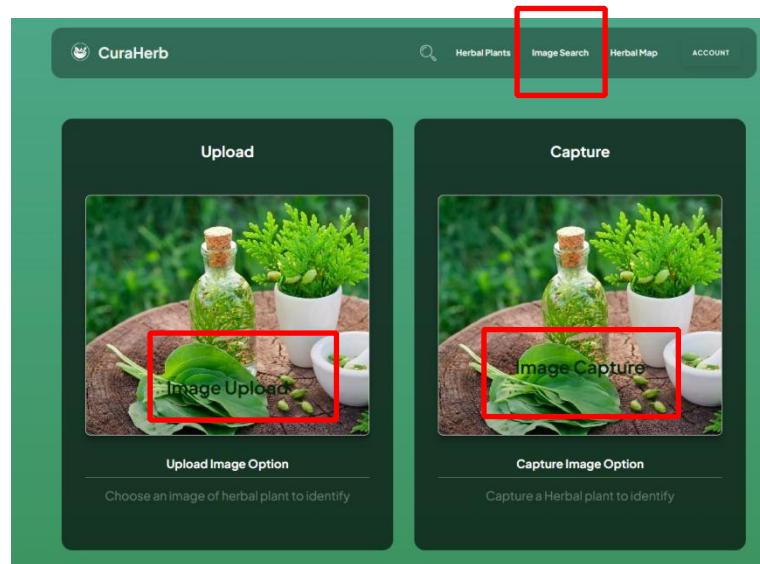
4. The Account button will allow the user change and reset their password.

A screenshot of a "CHANGE PASSWORD" form. The form has three input fields: "OLD PASSWORD", "NEW PASSWORD", and "NEW PASSWORD CONFIRMATION", each with a placeholder text. Below the fields is a blue button labeled "Reset Password". The entire form area is highlighted with a red rectangular box.

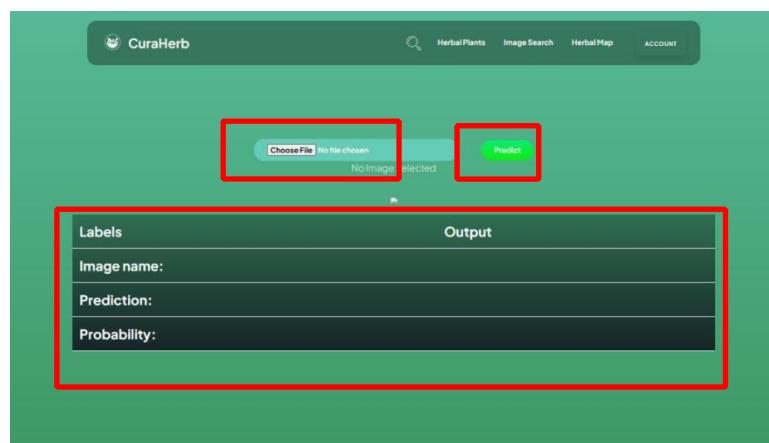
1. Click “Account” on the top-right of the screen to change and reset the user password
2. Fill up the fields and enter your old and new password and click “Reset Password” to update the profile password.

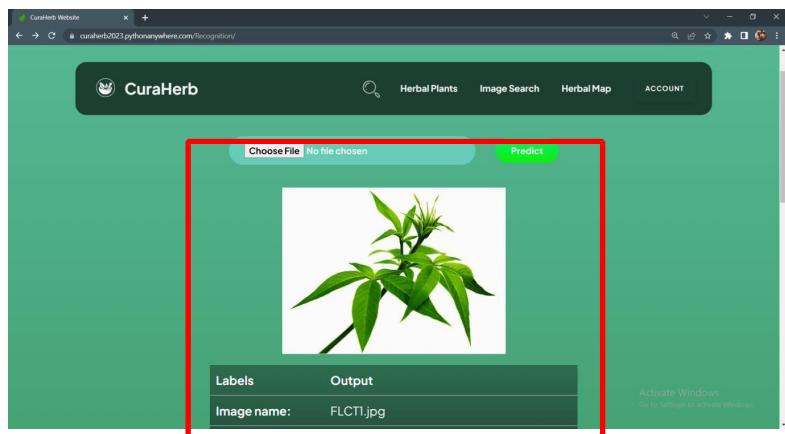
## UPLOAD AND CAPTURE

1. The image search function will allow the user to upload and capture images of herbal plants.

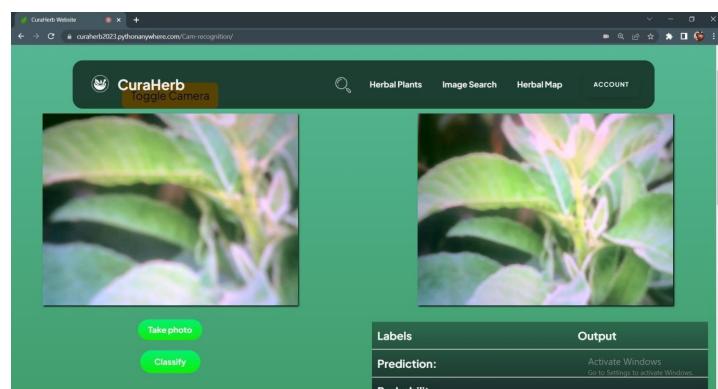
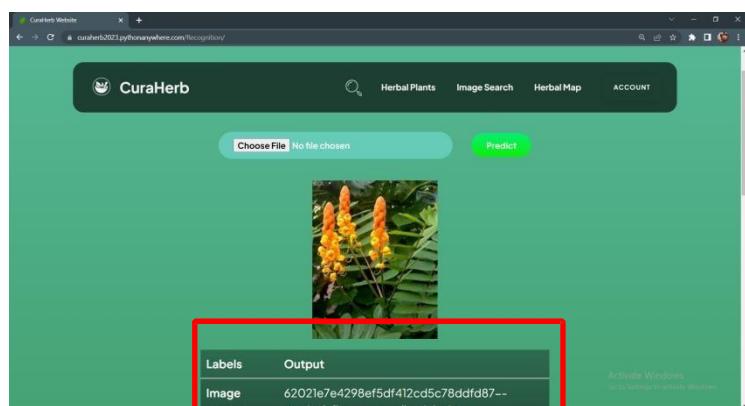


2. When the user choose to upload. It will directly reload to another page where the user can choose file from its browser and click the “Predict” button to see the results.

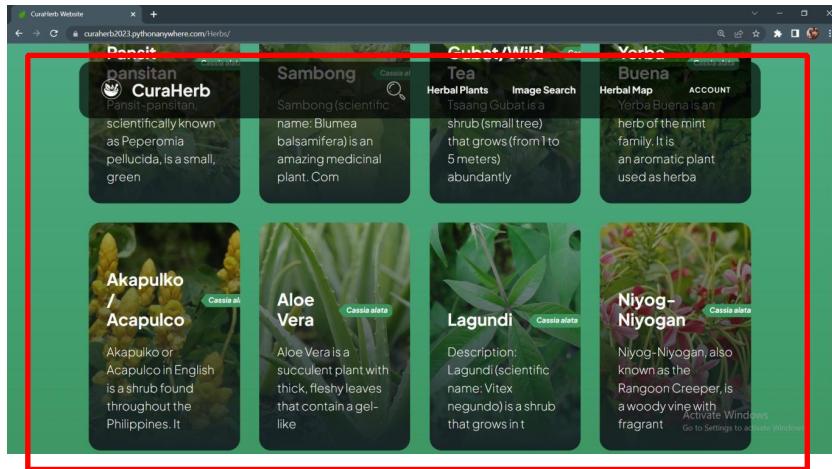




3. After the user upload the picture it will display the results of the herbal plants such as the “Label, Image name

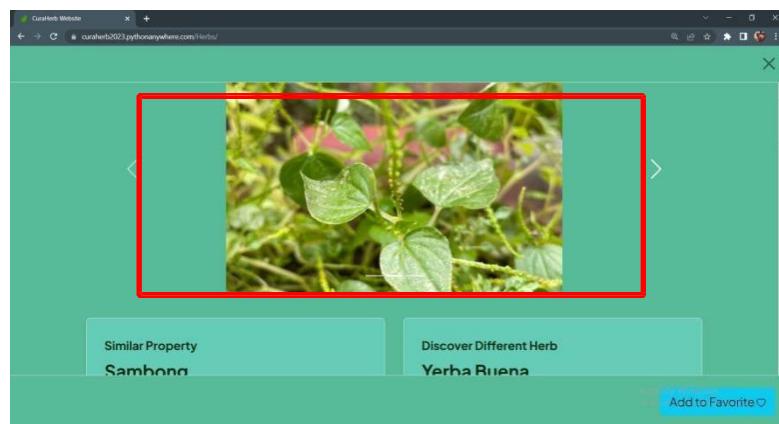


## Visitor/ User (Logged In) - Herbal Plants Library



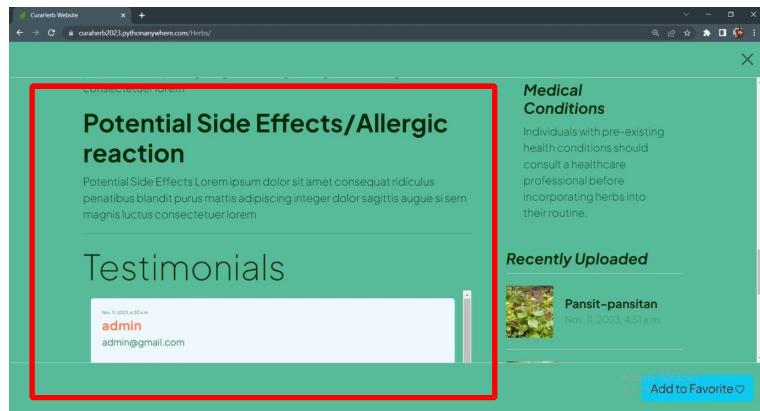
4. Herbal plants Library offers the list of herbal plants available. In this feature users can explore different types of herbal plants with the description, medication guide and uses.

## Visitor/ User (Logged In) - Herbal Plant Details



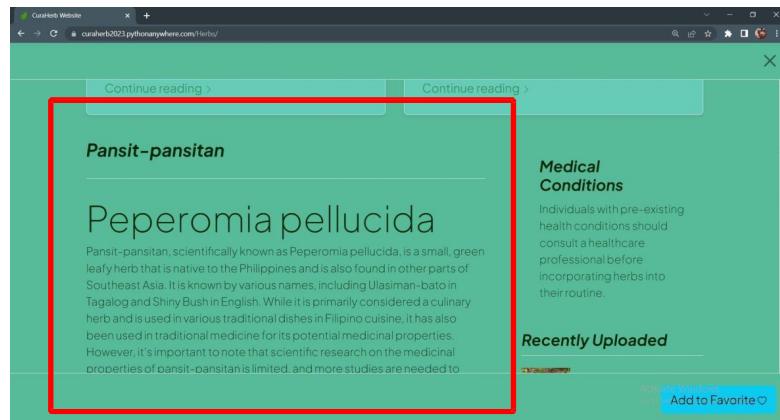
1. When the user click the preferred herbal to view. This will show now all the information of the herbal plants.

## Visitor/ User (Logged In) - Herbal Plant Details & Testimonials



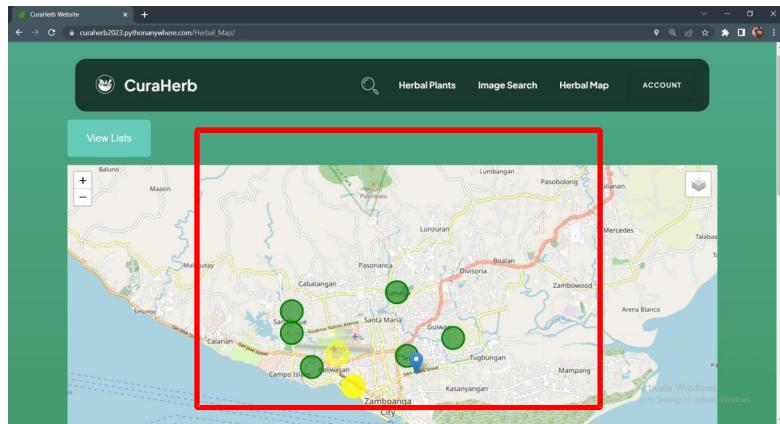
The screenshot shows a web browser window with a green header bar. Below the header, there's a section titled "Potential Side Effects/Allergic reaction" which contains placeholder text. Below this, a section titled "Testimonials" is shown, which also contains placeholder text. To the right of the main content area, there are two columns: "Medical Conditions" and "Recently Uploaded". The "Medical Conditions" column has some descriptive text. The "Recently Uploaded" column shows a thumbnail image of a plant, the name "Pansit-pansitan", and the date "Nov. 11, 2023, 4:51 a.m.". At the bottom right of the main content area, there's a blue button labeled "Add to Favorite".

1. User can also add testimonials about the herbal they used such as how effective it is and possible recommendation they may share to other users in the future.

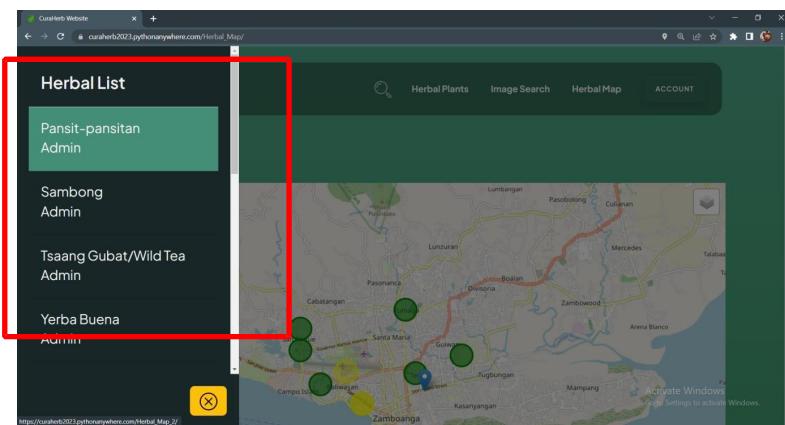


The screenshot shows a web browser window with a green header bar. Below the header, there's a section titled "Pansit-pansitan" which contains placeholder text. Below this, a section titled "Peperomia pellucida" is shown, which also contains placeholder text. To the right of the main content area, there are two columns: "Medical Conditions" and "Recently Uploaded". The "Medical Conditions" column has some descriptive text. The "Recently Uploaded" column shows a thumbnail image of a plant, the name "Pansit-pansitan", and the date "Nov. 11, 2023, 4:51 a.m.". At the bottom right of the main content area, there's a blue button labeled "Add to Favorite".

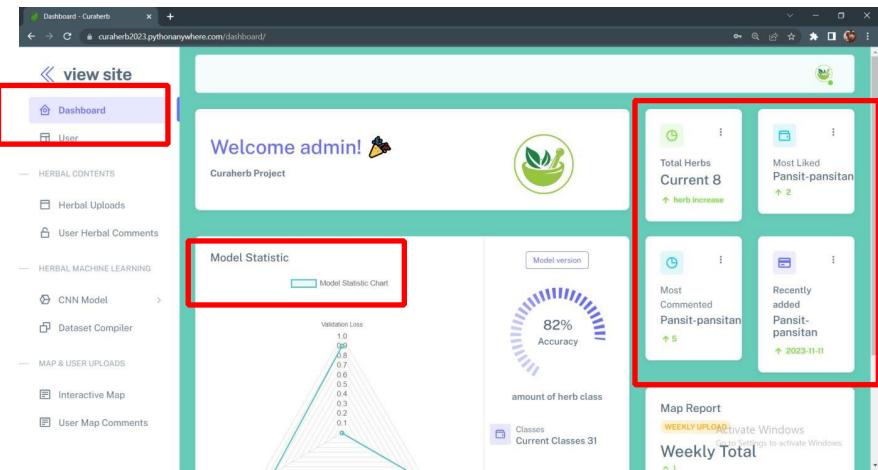
## Visitor/ User (Logged In) - Herbal Map



2. Herbal Map features allow the user to click and view a location where the herbal originally located. This will help the user to find nearest herbal in there place.

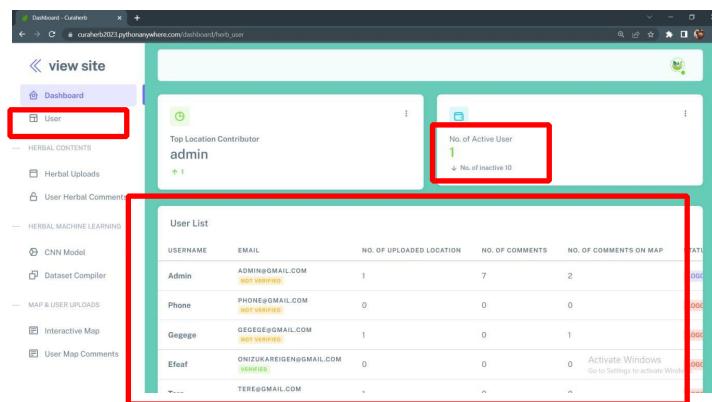


## ADMIN SIDE (DASHBOARD)



1. The admin dashboard will display all the statistics result of the data being collected by the Curaherb System. This include the “model statistics”, “total number of herbals”, “most liked herbal in the system” “most commented”, “Recently added” and a Map Report.

## Admin User - System User (List)



1. The “User” feature under the dashboard main menu is will display all the number of user registered in the system.

## Admin User - Herbal List

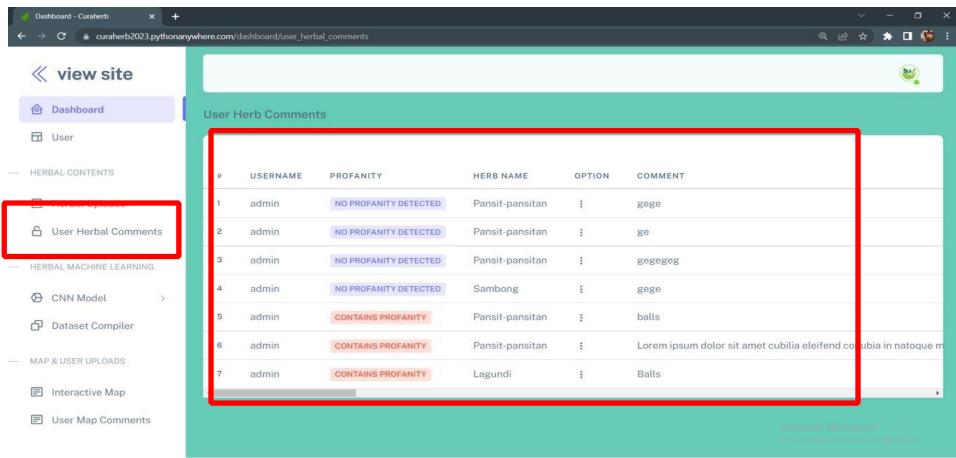
The screenshot shows a web browser window titled "Dashboard - Curaherb". The URL is "curaherb2023.pythonanywhere.com/dashboard/herbal\_upload". The left sidebar has a "view site" section with "Dashboard" selected. Under "HERBAL CONTENTS", "Herbal Uploads" is highlighted with a red box. The main content area is titled "Herb List" and contains a table with the following data:

NAME	NO. OF COMMENTS	NO. OF BOOKMARKS	IMAGE ATTACHMENT	ACTIONS
Pansit-pansitan	5	2	HERBS/54BF69799455302DC7A653C25157D45.JPG	[Edit]
Sambong	1	1	HERBS/49EF60B539FEBF24E6A860ABD1F11191.JIF	[Edit]
Tsaang Gubat/Wild Tea	0	0	HERBS/EHRETIAMICROPHYLLA-WEB.JPG	[Edit]
Yerba Buena	0	0	HERBS/17-REMARKABLE-BENEFITS-OF-YERBA-BUENA.JPG	[Edit]
Akapulko / Acapulco	0	0	HERBS/AKA.JPG	[Edit]
Aloe Vera	0	0	HERBS/ALOE-VERA-PLANT-OUTSIDE.JPG-1522875135.JPG	[Edit]
Lagundi	1	0	HERBS/LAGUNDI2.JPG	[Edit]
Niyog-Niyogan	0	0	HERBS/NIYOG-NIYOGAN.JPG	[Edit]

At the bottom right of the table, there is a link "Activate Windows".

1. Herbal list function allow the admin to see the current list of herbals in the system. It also allow the user to see the current no. of comment, boookmarks.
2. Admin may take an action to activate and deactivate the account of the user if it is no longer using the system.

## Admin User - User Herbal Comments



The screenshot shows a web browser window titled "Dashboard - Curaherb". The URL is "curaherb2023.pythonanywhere.com/dashboard/user\_herbal\_comments". The left sidebar has a "User" section expanded, with "User Herbal Comments" highlighted and enclosed in a red box. The main content area is titled "User Herb Comments" and displays a table with the following data:

#	USERNAME	PROFANITY	HERB NAME	OPTION	COMMENT
1	admin	NO PROFANITY DETECTED	Pansit-pansitan	⋮	gege
2	admin	NO PROFANITY DETECTED	Pansit-pansitan	⋮	ge
3	admin	NO PROFANITY DETECTED	Pansit-pansitan	⋮	gegegeg
4	admin	NO PROFANITY DETECTED	Sambong	⋮	gege
5	admin	CONTAINS PROFANITY	Pansit-pansitan	⋮	balls
6	admin	CONTAINS PROFANITY	Pansit-pansitan	⋮	Lorem ipsum dolor sit amet cubilia eleifend cubilia in natoque m
7	admin	CONTAINS PROFANITY	Lagundi	⋮	Balls

1. The admin side can receive the comments from the user on the herbal they commented. In this feature the admin has the right thing to accept or decline the comment if its include profanity words or false information.
2. Admin may click the “option” button to delete the comment

## Admin User - Model Statistics

This screenshot shows the 'Model Statistics' section of the dashboard. A red box highlights the 'CNN Model' entry in the sidebar under 'HERBAL MACHINE LEARNING'. A green circle labeled '1' is positioned over this entry. Another green circle labeled '2' is positioned above the main content area. A large red box surrounds the 'Model Statistics' card, which displays the following data:

Model Statistics	
3000 Total Data	
82 Accuracy	
31	Total Classes
Accuracy	Accuracy on Training Data
Val Accuracy	Accuracy on Separate Validation Data
0.82	0.82
Classes	Number of Classes Supported
31	31
Data Length	Number of Images Used
3000	3000

To the right of the statistics card, there is a 'FILE INPUT' section with several input fields. A red box highlights the 'Choose File' field, which currently shows 'No file chosen'. A green circle labeled '3' is positioned above this field. A green circle labeled '4' is positioned below it. A red box highlights the 'Upload' button at the bottom of this section. A green circle labeled '5' is positioned below the 'Upload' button.

## Admin User - Model Training

This screenshot shows the 'Training' section of the dashboard. A red box highlights the 'CNN Model' entry in the sidebar under 'HERBAL MACHINE LEARNING'. A green circle labeled '1' is positioned over this entry. A large red box surrounds the 'Training' input fields. The fields include:

- 'Choose File' (No file chosen)
- 'UPLOAD A VALID 'DATASET.NPY' HERB FILE'
- 'Choose File' (No file chosen)

Below these fields is a 'Load Data' button. A green circle labeled '2' is positioned above the first 'Choose File' field. A green circle labeled '3' is positioned above the second 'Choose File' field. A green circle labeled '4' is positioned above the 'Load Data' button.

## Admin User - Dataset Compiler

The screenshot shows a web-based application for dataset compilation. On the left, a sidebar menu includes options like Dashboard, User, Herbal Contents, Herbal Uploads, User Herbal Comments, Herbal Machine Learning, CNN Model, Dataset Compiler, Interactive Map, and User Map Comments. The main area has a teal header bar with a 'view site' link. Below it, there's a 'Selected Images' section and a 'Dataset Preparation' section. The 'Dataset Preparation' section contains fields for 'HERB CLASS NAME' (with a red border and error message 'Please Enter Correct Class Name') and 'MULTIPLE FILES INPUT' (with a red border and message 'Choose Files | No file chosen'). A green 'Prepare Dataset' button is also in this section, also highlighted with a red border. At the bottom, there's a 'Dataset List' section with a 'Compile New Dataset' button.

## Admin User - Interactive Map (Area Chart)

The screenshot shows the 'Interactive Map' section of the application. The sidebar menu is identical to the previous screenshot. The main area features a large red-bordered area containing a radial chart titled 'Map Area Chart'. The chart has three segments: a green 'Herbs' segment, a blue 'Store' segment, and a yellow 'User Uploaded Herbs' segment. A tooltip 'Herbs Dataset: 8' is shown over the green segment. The chart is labeled 'Herbs' at the top, 'User Uploaded Herbs' at the bottom, and 'Store' on the right. A legend at the top right identifies the colors: green for Herbs, blue for Store, and yellow for User Uploaded Herbs. A message 'Activate Windows Go to Settings to activate Windows.' is visible in the bottom right corner.

## Admin User - Interactive Map (Table)

The screenshot shows a web browser window titled "Dashboard - Curaherb". The URL is "curaherb2023.pythonanywhere.com/dashboard/interactive\_map". On the left, there is a sidebar with a "view site" button and a navigation menu. The menu includes "Dashboard", "User", "HERBAL CONTENTS" (with "Herbal Uploads" and "User Herbal Comments" sub-options), "HERBAL MACHINE LEARNING" (with "CNN Model" and "Dataset Compiler" sub-options), "MAP & USER UPLOADS" (with "Interactive Map" and "User Map Comments" sub-options). The main content area is titled "Map Table" and contains a table with columns: NAME, LONGITUDE, LATITUDE, TYPE, and ACTIONS. The table data is as follows:

NAME	LONGITUDE	LATITUDE	TYPE	ACTIONS
Pansit-pansitan	122.096545	6.925598	HERB	⋮
Sambong	122.082378	6.920128	HERB	⋮
Tsaang Gubat/Wild Tea	122.046802	6.92716	HERB	⋮
Yerba Buena	122.052927	6.916750122052927	HERB	⋮
Akapulko / Acapulco	122.079215	6.939486	HERB	⋮
Aloe Vera	122.046788	6.933783	HERB	⋮
Lagundi	None	None	HERB	⋮
Niyog-Niyogan	None	None	HERB	⋮
herbal stores	122.0851	6.914474	STORE	⋮
sample	122.0658345	6.9107644	UPLOAD	Activate Windows Go to Settings to activate Windows.
etererere	122.0658396	6.9107407	UPLOAD	⋮

## Admin User - Map Comment

The screenshot shows a web browser window titled "Dashboard - Curaherb". The URL is "curaherb2023.pythonanywhere.com/dashboard/user\_map\_comments". On the left, there is a sidebar with a "view site" button and a navigation menu. The menu includes "Dashboard", "User", "HERBAL CONTENTS" (with "Herbal Uploads" and "User Herbal Comments" sub-options), "HERBAL MACHINE LEARNING" (with "CNN Model" and "Dataset Compiler" sub-options), "MAP & USER UPLOADS" (with "Interactive Map" and "User Map Comments" sub-options). The main content area is titled "Map - Comments" and contains two tables. The first table, "Map - Comments", has columns: #, USERNAME, NAME, and PROFANITY. The data is as follows:

#	USERNAME	NAME	PROFANITY
1	gegege	sample	NO PROFANITY DETECTED
2	admin	sample	CONTAINS PROFANITY
3	admin	sample	NO PROFANITY DETECTED

The second table, located to the right, has columns: OPTION and COMMENT. The data is as follows:

OPTION	COMMENT
⋮	huehue
⋮	Balls
⋮	ball



# HESAM K. DAWAMI

PROGRAMMER

## Contact

- S.Arabi Siocon Zamboanga del norte
- 09150023326
- <https://curaherb2023.pythonanywhere.com>
- Azenei24@gmail.com

## Skills

### Software Development



### Problem Solving



### Data Analysis



## About Me

I am interested in software development because it is one of the things that I wish to accomplish or achieve so that I could create my own software that will be able to make some people happy and make things easy for some people.

## Education

- Bachelor of Science in Information Technology  
2019-2024  
*Western Mindanao State University*
- Science Technology Engineering Mathematics  
2017-2019  
*Jose Rizal Memorial State University*
- Lower Secondary Level  
2012-2017  
*Jose Rizal Memorial State University*

## Work Experience

### POBLACION. SIOCON ZAMBOANGA DEL NORTE MARCH 1 – 30, 2019

#### Work Immersion

- I did my first work immersion back then and I have learned many things from it that I can use in an actual job and it made me feel what it really is to be in a workplace it gave me a new experience and knowledge useful for me when I finally get to work.



# SHYNIEMAR A. HASSAN

UI/UX DESIGNER



## Contact



Southcom Upper Calarian Z,C



09667109155



<https://curaherb2023.pythonanywhere.com>



shyniemarh@gmail.com



## Skills

### Project Management



### Problem Solving



### Creativity



### Leadership



## About Me

I specialize in crafting user-friendly interfaces that seamlessly blend aesthetics and functionality. My designs prioritize intuitive experiences to resonate with users' needs.



## Education

### Bachelor of Science in Information Technology

2019-2024

**Western Mindanao State University**

### Humanities and Social Sciences

2017-2019

**Pilar College of Zamboanga City**

### Lower Secondary Level

2012-2017

**Zamboanga National High School**

**West**

## Work Experience

### National Beauru of Investigation

2019

#### Project Manager

- I assist individuals in creating NBI accounts on a website while also addressing their queries and providing guidance at the National Bureau of Investigation.

### Western Mindanao State University

2022-2023

#### Project Manager

- Coordinates tasks, communicates updates, and resolves issues to ensure the team delivers the project on time and meets quality standards
- Manage website appearance, as well as the logo that is displayed on the system



# MARGO KATE G. MAPASI

TESTER

## Contact

- Yvanoff Drive, Tumaga, Zamboanga City
- 09976017186
- <https://curaherb2023.pythonywhere.com>
- mapasimargokate@gmail.com

## Skills

### Creativity



### Graphic Design



## About Me

Proficient in data entry software and tools, including Microsoft Excel, Google Sheets and powerpoint.  
Basic Graphic Design using Canva

## Education

- Bachelor of Science in Information Technology  
2019-2024  
*Westen Mindanao State University*
- Science Technology Engineering  
Mathematics  
2017-2019  
*Don Pablo Lorenzo Memorial High School*
- Lower Secondary Level  
2012-2017  
*Zamboanga City High School Main*

## Work Experience

- WORK EXPERIENCE  
City Social Welfare and Development -  
Work Immersion April 2019-May 2020
- Upwork - Market Research Data  
Entry

