A Mobile Application Based System for Tomato Pest and Disease Detection

Robert Ngeruro Kamau

073652

A Dissertation Submitted in Partial Fulfilment of the Requirements for the Degree of Master of Science in Mobile Telecommunication and Innovation at Strathmore University

School of Computing and Engineering Sciences

Strathmore University

Nairobi, Kenya

June, 2025

Declaration and Approval

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree at this or any other University. To the best of my knowledge and belief, the dissertation contains no material previously published or written by another person except where due reference has been made in the dissertation itself.

Robert Ngeruro Kamau.

Signature:

Date: 22nd May, 2025.

Approval

The dissertation of Robert Ngeruro Kamau was reviewed and approved for examination by the following:

Dr. Victor Rop,
Lecturer, School of Computing & Engineering Sciences,
Strathmore University

Dr. Julius Butime,
Dean, School of Computing & Engineering Sciences,
Strathmore University

Prof. Bernard Shibwabo,
Director of Graduate Studies,
Strathmore University

Abstract

Precision farming is an approach to farming that uses technology and data to reduce farming costs, improve crop yields, and optimize resource utilization. It relies on a range of tools and technologies to collect, process, and analyse data to derive actionable information that is used to make informed decisions about planting, disease and pest control, and harvesting. Tools used in data collection include Internet of Things (IoT) devices, Unmanned Aerial Vehicles (UAVs), Geographic Information Systems (GIS), and weather stations. Data analytics, GIS mapping and machine learning are used to process, analyse, and make sense of the data. Several factors limit smallholder farmers from adopting precision farming solutions. These include high cost of implementation, limited digital infrastructure such as the Internet, lack of awareness and limited knowledge in utilising these solutions.

This research aimed to bridge this gap by developing a system that uses machine learning to detect tomato pests and diseases and suggest treatment and prevention measures through the mobile phone. A review of the factors influencing agricultural productivity among smallholder farmers, and existing solutions that attempt to mitigate these factors was carried out. A requirement analysis was carried out to establish the viability of developing a mobile application-based system for tomato disease detection. Subsequently, this dissertation developed a deep learning Convolutional Neural Network (CNN) model trained using 16,880 images of tomato diseases. The trained model achieved a validation accuracy of 98.22%. A mobile application was developed, and the machine learning model embedded into it. Prototyping software development methodology was adopted to develop the system. The system was tested at several stages, to ensure that it met the set requirements for performance and functional requirements. The system was able to detect tomato diseases and provide the user with disease treatment and prevention recommendations.

Keywords: precision farming, machine learning, convolutional neural network, smallholder farming.

Table of Contents

List of Figures

List of Tables

List of Abbreviations/Acronyms

AI	-	Artificial Intelligence
API	-	Application Programming Interface
APK	-	Android Package Kit
CNN	-	Convolutional Neural Network
CUDA	-	Compute Unified Device Architecture
DFD	-	Data Flow Diagram
DL	-	Deep Learning
FAO	-	Food and Agriculture Organization of the United Nations
GCP	-	Google Cloud Platform
GIS	-	Geographic Information System
GPU	-	Graphics Processing Unit
GSMA	-	GSM (Global System for Mobile communications) Association
ICT	-	Information and Communications Technology
IFAD	-	International Fund for Agricultural Development
IoT	-	Internet of Things
IVR	-	Interactive Voice Response
JSON	-	JavaScript Object Notation
KALRO	-	Kenya Agricultural and Livestock Research Organization
ML	-	Machine Learning
OOD	-	Object-Oriented Design
SMS	-	Short Message Service
XML	-	Extensible Markup Language

UAV - Unmanned Aerial Vehicle

USSD - Unstructured Supplementary Service Data

Definition of Terms

Google Colaboratory Also popularly known as Google Colab, is a hosted Jupyter Notebooks service well suited for machine learning. It provides computing resources, including GPUs (Naik, 2023).

JPEG This is a standard format for storing image files, created by the Joint Photographic Experts Group (JPEG) (Hanna, 2023).

Keras A high-level API for building and training deep learning models (Duerr & Sick, 2020)

MobileNetV3 A lightweight convolutional neural network architecture, developed by Google, specifically for mobile and embedded vision applications (Elsayed Abd Elaziz, et al., 2021).

Acknowledgement

First and foremost, I would like to give thanks to God for the blessings and inspiration through my journey pursuing this master's programme. I would like to appreciate my supervisor Dr. Victor Rop for the guidance, support and honest critique throughout the dissertation work. I also wish to thank Dr. Vitalis Ozianyi, who offered a helping hand in the initial stages of conceptualizing the research. I appreciate my family and friends, for the patience and words of affirmation.

Chapter 1: Introduction

1.1 Background of the Study

Improving agricultural productivity is the most effective strategy for poverty alleviation and economic development. Therefore, advancing working agricultural systems is an important step to alleviating poverty (Anik et al., 2020). Unfortunately, agriculture faces numerous challenges, including unpredictable weather conditions, environmental degradation, pests and diseases. These challenges are apparent in developing countries such as in sub-Saharan Africa, where due to unreliable infrastructure and limited access to information, overreliance on rainfed farming, poor farming practices such as misuse (overuse or underuse) of fertilizers and pesticides, and planting crops not suited to a particular environment, are observed. According to a report by the Alliance for a Green Revolution in Africa (AGRA) (2017), close to 70% of farmers in sub-Saharan Africa are smallholder farmers who whilst ill-equipped to cushion themselves from losses resulting from these unpredictable conditions, are also the most vulnerable as farming is often their primary source of income.

There are three major ways in which technology is being utilized to improve food production in Kenya. Firstly, there are several research initiatives aimed at promoting sustainability in agriculture, such as the Kenya Agricultural and Livestock Research Organization (KALRO), who lead and coordinate research in crops and biotechnology, leading to development of drought-resistant crops, pesticides, and access to research information (KALRO, 2016). KARLO has developed the Kenya Agricultural Observatory Platform (KAOP) that offers location-specific advisory services based on weather information. There are as well initiatives by the Ministry of Agriculture such as the National Agricultural and Rural Inclusive Growth Project (NARIGP) aimed at strengthening community-level institutions and production value chains, and the Small Scale Irrigation and Value Addition Project (SIVAP) aimed at enhancing irrigation development, and soil and water conservation (Agriculture Sector Development Support Programme, 2019).

Secondly, precision farming technologies utilize tools such as IoT sensors, UAVs, remote sensing, GIS, cloud computing and machine learning to collect raw data and perform data analytics to gain localized and relevant insights that are used to accurately map crop, weather and soil characteristics to provide farmers with individualized crop, pest and disease management practices. These solutions are being utilized in research institutions, large scale maize and wheat farms in North and Central Rift Valley, tea plantations in the Rift Valley and Central regions, and coffee plantations in Central Kenya. These solutions are however beyond the reach of the smallholder farmer.

Thirdly, mobile technology is utilized to improve agricultural productivity by providing information regarding weather, pests and diseases, price forecasts, and market availability. Mobile application-based services such as Digifarm, Digicow, and M-Shamba focus on providing agricultural advisory services to smallholder farmers via SMS, USSD and android mobile application (FAO, 2023). Mobile technology can be integrated with digital technologies such as precision farming and machine learning to avail invaluable insights and advisory at a low cost. This has the potential to positively impact the economic status of smallholder farmers, through improved agricultural practices that lead to reduced input costs and increased crop yields.

1.2 Statement of the Problem

Smallholder farmers in Kenya rely on agricultural extension services such as field days, information desks, agricultural exhibitions, and farm visits, for the detection and management of tomato pests and diseases. The Ministry of Agriculture, Livestock, Fisheries and Cooperatives notes that these agricultural extension services are impacted by reduced financing and extension manpower (Ministry of Agriculture, Livestock, Fisheries and Cooperatives, 2022). Farmers also use mobile-based agriculture information services. Even though these mobile-based agriculture information services provide useful information about tomato pests and diseases, the information they provide lacks localization since they lack the capacity to provide feedback based on the farmer’s precise input.

Precision farming technologies are used to provide data that is processed to provide accurate insights on tomato pests and diseases. Unfortunately, these precision farming technologies are beyond the reach of most smallholder farmers. By utilising machine learning to identify patterns in diseased images of tomato pests and diseases, a localized solution in which inference is performed within a mobile application in the farmer’s phone can be developed to provide tailor-made and relevant insights on tomato pests and diseases amongst smallholder farmers in Kenya. This dissertation seeks to develop a solution that trains a deep learning model and integrates it into a mobile application, that can be used by smallholder tomato farmers to provide image inputs of diseased tomato crops from their own farms, and get feedback on the disease diagnosis, and disease treatment and prevention recommendations.

1.3 Main Objective of the Research

The primary objective of this dissertation was to develop a mobile application-based system that utilizes machine learning for pattern identification in tomato disease data to provide smallholder farmers with important insights into treatment and prevention of tomato disease, with the aim of improving their agricultural productivity.

1.4 Specific Objectives of the Research

i. To evaluate existing application of mobile technologies in agriculture.
ii. To review digital technologies that can be applied to improve agricultural productivity.
iii. To design and develop a mobile application-based image recognition system to detect tomato pests and diseases and recommend treatment and prevention measures for smallholder farmers.
iv. To test the functionality of the developed system.

1.5 Research Questions

This research sought to address the following questions:

i. What are the existing mobile technology applications in agriculture?
ii. What are the digital technologies that can be applied to improve agricultural productivity?
iii. How can a mobile application-based image recognition system that detects tomato pests and diseases and recommends treatment and prevention measures for smallholder farmers be designed and developed?
iv. How will the developed system be tested?

1.6 Significance of the Study

Agriculture is an important sector of the economy that is at the cornerstone of assuring food security. Smallholder farmers constitute the biggest portion of farmers in Kenya and therefore play a significant role in promoting attainment of food security. However, they face challenges in leveraging technology and innovation to bring predictability and improved efficiency to the production process. This research sought to investigate how creation and use of intelligent systems that utilize data from precision farming technologies to provide farmers with real-time, relevant information and recommendations, can help the farmers improve their tomato production. This has the potential to have a positive economic impact on the smallholder tomato farmers through improved crop yields and reduced input costs. Optimized farming practices also help reduce impact on the environment as relevant insights empower the farmer to optimize use of pesticides. The research can also act to influence policy direction by the government and other organizations that are in this space, on how precision farming and artificial intelligence can be made more accessible to smallholder farmers.

1.7 Scope of the Study

The output of this research is expected to be applicable to smallholder farms anywhere in Kenya, for diverse crops, since in most cases, small-scale farmers practice mixed farming. The mobile application however limited itself to tomato farming amongst smallholder farms in Kajiado county of Kenya. The application of precision farming solutions is also wide and varied, and as such it will not be possible to cover the subject in its entirety in this paper. Therefore, the dissertation will limit itself to detection and control of tomato pests and diseases

1.8 Limitations of the Study

The dissertation was limited by time constraints, thus informing the scope of the study. Implementing the data collection tools was also limited by the cost involved, and availability of infrastructure and skilled personnel for data collection. Additionally, successfully training a machine learning model requires a lot of data. Therefore, the researcher relied on publicly available image datasets to train the deep learning image classification model. Since the mobile application utilises the smartphone's camera capabilities as an input towards tomato pest and disease detection, the study was limited by smartphone ownership by the target farmers.

Chapter 2: Literature Review

2.1 Introduction

This chapter provided a review of existing use of mobile technologies to enhance agricultural productivity among smallholder farmers. To contextualize this review, the factors influencing agricultural productivity among smallholder farmers in Kenya were evaluated. The chapter explored existing literature to bring out the characteristics of precision farming and machine learning that can be utilized to improve agricultural productivity. Therefore, an overview of the technologies that make precision farming and machine learning possible was done, and their usefulness in the Kenyan context explored.

2.2 Factors Influencing Agricultural Productivity Among Smallholder Farmers in Kenya

The Alliance for a Green Revolution in Africa (2017) has defined smallholder farmers as farmers who manage fragmented farms that are on average of a size less than two hectares. They are characterized by reliance on rainfed farming, a focus on farming for both subsistence and to sell their produce, and employing family labour, with little or no machinery (Bartol, 2023). Small-scale farmers hold more than 70% of all arable land in sub-Saharan Africa and are responsible for producing more than 80% of the food consumed in the continent (Alliance for a Green Revolution in Africa, 2017). They therefore play a critical role in ensuring food sustainability. The terms smallholder farmer and small-scale farmer are synonymous and therefore used interchangeably throughout this paper.

Hlatshwayo et al. (2023) categorized the factors influencing agricultural productivity among small-scale farmers into social economic, climate change, crop and farm management practices, access to advisory services, and access to infrastructure and technology.

2.2.1 Social Economic Factors

A study by Tatis et al. (2022) showed that smallholder farmers are characterised by low education levels, low incomes, poor access to credit and difficulty in accessing agricultural services due to them being primarily located in rural areas. There is also a low participation in producers associations among smallholder farmers (Anang & Asante, 2020). The studies showed a correlation between these factors and the farmers' choice of farming practices, meaning the smallholder farmer is least likely to adopt better farming practices.

2.2.2 Climate Change

Variability in temperature, unpredictable rainfall and extreme weather events have a negative impact on crop yields and increased incidence of pests and diseases. Cairns et al. (2021), in a study among maize farmers in Kenya, concluded that climate change was a factor in the decline of maize production. A study by Harvey et al. (2018) also found that crop production in 87% and 66% respectively, of maize and coffee farmers studied was negatively impacted by climate change. Similarly, 32% of the smallholder farmers experienced food insecurity due to adverse weather occurrences. Smallholder farmers are the most vulnerable to climate change due to their limited ability to adopt; they rely on rain-fed agriculture, cultivate on fragmented pieces of land, have limited technical knowhow, limited access to financial services and crop insurance. Smallholder farmers also lack access to climate prediction facilities, which makes them less adaptable to climate change (Harvey, et al., 2018).

2.2.3 Land and Crop Management Practices

Whereas experimental agricultural fields within research organizations show a maize tonnage of up to nine tonnes per hectare, most smallholder maize yields are on average less than two tonnes per hectare (Birch, 2018). The biggest factor in this variation in the crop yields is the land and crop management practices adopted by the farmers. Poor land management practices among smallholder farmers contribute to a deterioration in soil fertility. This includes the lack of or inadequate use of fertilizers. Increased soil erosion and nutrient degradation can also be caused by the smallholder farmers' tendency to cultivate their lands every season to keep up with their financial demands, and thus not allowing for fallow periods that are important to restore soil fertility (Jouzi, et al., 2017).

Other crop management practices such as proper seedbed preparation, proper crop spacing, pest and disease control, and irrigation use and control, are shown to improve crop growth, development and yields (Walia, 2021). A baseline survey by the Agriculture Sector Development Support Programme revealed that only 40% of surveyed smallholder households practiced some form of sustainable crop and land management (Agriculture Sector Development Support Programme, 2019). Government involvement through sensitization programmes, providing affordable access to input markets such as fertilizers and pesticides, and improved agricultural extension services are key to encouraging adoption of land and crop management practices among smallholder farmers.

2.2.4 Access to Agricultural Advisory Services

A study carried out by Gatheru et al. (2021) estimated that the proportion of farmers who accessed agricultural advisory services varied between 13% and 26%, among the surveyed respondents. The

Government is the main provider of the agricultural extension services, with a proportion of between 50% and 90% of the agricultural advisory being provided by public extension officers.

The government-led public agricultural extension system is a training and visitation system, consisting of village-level extension workers, who are regularly trained through seminars organized with research station scientists, and who in turn schedule visits to designated farmers who act as points of contact, with the belief that the knowledge shared will transfer from these points of contact, to the rest of the farmers (Bonilla et al., 2024). This system faces challenges due to financial unsustainability, declining numbers of agricultural extension officers, affecting the number of farmers that can be reached, and inadequate infrastructure to support agricultural extension.

A more demand-driven approach that follows the informational needs of the farmers to determine the most optimal means of provision of extension services is necessary. Emerging alternative approaches include private extension systems, public-private partnerships, and adoption of information and communication technologies in the provision of these services.

A study by Phiri et al. (2019) ranked the kind of information farmers sought. As illustrated in Table 2.1 information on crop husbandry, pest and disease control, market for farmers' produce and weather/climate ranked highly as the most sought kind of information.

Table 2.1: Information Needs of Farmers (Phiri et al., 2019)

Information Needs	**Respondents**	**Percentage (%)**
Crop Husbandry	149	78
Pest and Disease Control	105	55
Market for Their Produce	99	52
Weather/Climate	96	50
Farm Mechanisation	84	44
Animal Husbandry	82	43
Access to credit	58	30
Income Generation	24	12
Agro Technology	17	9

Smallholder farmers also access information from traditional sources such as friends, family and neighbours, who may not have sufficient, accurate or reliable information. This leaves them to rely on anecdotal methods to decision-making, leading to poor crop yields, increased costs from wastage of resources such as water and fertilizer, and destruction of the environment resulting from the overapplication of fertilizer and pesticides. The number of respondents who sought information from more reliable sources such agricultural extension officers and electronic media such as radio, television, mobile phone-based and web-based agricultural information sources was low (Phiri et al., 2019).

2.2.5 Access to Infrastructure and Technology

Tatis et al. (2022) observed that access to transport and communication infrastructure can enhance the agricultural output and economic growth of rural farmers. This is because in addition to easing access to input and output markets, improved access to transport and communication infrastructure also leads to better access to agricultural extension services, as well as improved quality of the information they access (Anang & Asante, 2020). With proper adoption strategies that encourage farmers to use their mobile phone as more than just a tool for personal communication, ICTs can be employed to boost dissemination of information. Digital channels can be adopted in place of extension workers, or act as a complement to the activities of the extension workers (Bonilla et al., 2024). Alowolodu and Gabriel (2021) examined how smartphones can help farmers explore markets for their produce and tools for production with an aim at economic development. The studies noted the challenges that farmers face and thus proposed mobile application to help them manoeuvre these challenges.

2.3 Agricultural Technology Mobile Applications

Mobile agriculture, also known as M-Agriculture, is an umbrella term used to refer to the provision of agricultural services and information through mobile devices such as the mobile phone, tablet devices, or any other portable communication device (Kumar & Chandrasekaran, 2019). The mobile technologies used in these devices include mobile applications, web-based applications, SMS, USSD and IVR services. Mobile technology offers advantages because of its distinct features of mobility, internet access, and support for features such as SMS and USSD. Studies have been carried out with evidence showing improved productivity resulting from use of mobile technologies in delivering agricultural extension services. Mwita et al. (2020), through a survey of dairy farmers in Bomet, Nyandarua and Uasin Gishu counties in Kenya showed that the use of iCow services increased their milk production, milk income and household income by 13%, 29%

and 22%, respectively. Similarly, Haruna et al. (2018) indicated that mobile phone use by farmers improved their farm productivity by at least 261 kilograms per hectare in a production season.

Based on the need they address, applications of mobile technologies in agriculture can be categorized into market services, financial inclusion and risk management, and agricultural extension and advisory services (Kumar & Chandrasekaran, 2019). This dissertation focuses on how a mobile-based system can be developed that detects tomato pests and diseases and provides insights into treatment and prevention of the disease. This solution falls under the agricultural extension and advisory services category. The literature identified three tools for provision of agricultural extension and advisory services: PlantwisePlus, Digicult and Kenya Agricultural Observatory Platform. The research appraised their strengths and weaknesses to identify frugal innovation gaps. This helped to inform the design and implementation of the mobile application-based image recognition system.

2.3.1 PlantwisePlus

PlantwisePlus is an agricultural advisory programme by The Centre for Agriculture and Bioscience International (CABI) that enables early detection of pests allowing for prompt action in pest containment, prevention from spreading to new areas, and farmer advisory services. PlantwisePlus applies the plant clinics model where farmers bring samples of the diseased plants with them, and specially trained extension officers, known as plant doctors, diagnose the disease and offer advisory (Migiro & Otieno, 2020). The advantage that PlantwisePlus has is that in addition to the plant doctors, who are trained on how to diagnose plant health problems from visual symptoms, CABI owns a rich knowledge bank on invasive pest species, specific to Kenya, among twelve other countries in Africa, containing information and factsheets about pest diagnosis and management. CABI has also created digital decision support tools to empower the plant doctors and other agricultural extension stakeholders. However, the primary means of interaction with the smallholder farmers remains through the plant clinics. With the rich knowledge bank, there exists an opportunity to realize efficiency in the work by introducing mobile applications as the primary means of interaction with the small-scale farmers and subsequently focus human effort towards enriching the knowledge bank further.

2.3.2 Digicult

Digicult is a pest surveillance digital platform that provides digital training resources and crowdsourcing, to train farmers through their mobile devices on detection and prevention of fall armyworm (Awuor & Otanga, 2019). Digicult provides a monitoring module that performs data analysis and image processing of information collected from the farmers for early detection of fall armyworm outbreaks and provide treatment options and mechanisms to control the invasion of the pest. The system supports crowdsourcing which enables farmers without access to smartphones to be made aware of potential fall armyworm outbreaks and get educated via its SMS and USSD platform. This solution concentrates itself on only the fall armyworm. It is therefore a platform that can be improved upon, by including other popular crop pests and diseases.

2.3.3 KAOP App

The Kenya Agricultural Observatory Platform (KAOP) is a digital platform that integrates climatic data from a range of meteorological models including satellite imagery to generate location-specific weather advisory (Akshatha & Dhulipala, 2024). The platform is a collaboration between KALRO and the Kenya Meteorological Department. The KAOP weather platform provides weather forecast of up to fourteen days. Additional platforms: Agronomic advisory and Marketing information, have been integrated to provide advisory on good agricultural practices and location-specific market information respectively. The solution is provided through a web-based application, a mobile application and an USSD service.

This researcher observes that there exists a lot of research data on agriculture, domiciled in research institutions, non-governmental organizations, the ministry of agriculture, as well as other publicly available sources. KAOP, being a research institution (KALRO) backed platform, is a good launching pad for a common collaboration point, to enhance its datasets using data from these other institutions, individual contributors as well as publicly available sources, to provide smallholder farmers with additional services such as pest and disease detection advisory, in addition to the weather and agronomic insights it presently provides. The insights and advisory are delivered to smallholder farmers in a more cost effective and timely manner using mobile technologies. By applying data-driven farming practices, farmers can make informed decisions about their crop management, informational and advisory needs. This leads to higher crop yields and enhances sustainability of agriculture among smallholder farmers.

2.4 Precision Farming Technologies

Precision farming uses data collection technologies including Internet of Things (IoT), Unmanned Aerial Vehicles (UAV), and Geographic Information Systems (GIS). A communication network is used to transfer data to a cloud-based platform. Data Analytics (DA) and Machine Learning (ML) are the methods used to process, analyse, and make sense of the collected data. Mobile technologies are used to dispense the useful information to the end-user, in this case the small-scale farmer. This section delves into these technologies and how they fit into the precision farming ecosystem.

2.4.1 Internet of Things

The Internet of Things (IoT) is a network of physical objects that are embedded with sensors, software and internet connectivity, enabling them to collect data and transfer to other systems (Thirisha, et al., 2023). IoT is the most predominantly used in-field monitoring technology. IoT for agriculture uses on-site sensors to collect data on various farming attributes, including soil pH, temperature within the environment, and soil moisture (Elijah et al., 2018). The data collected over time becomes pertinent for farmers to make decisions and optimize results, resulting in high operational efficiency and increased crop yields (Elijah et al., 2018). IoT devices integrate with technologies such as cloud computing and the mobile communication ecosystem, making IoT a critical enabler of precision farming in smallholder farms.

Examples have been illustrated of how IoT has been used to help farmers improve their crop management. Singh et al. (2024) demonstrated how introducing intelligence in greenhouses by use of IoT tools leads to better crop yields, compared to normal greenhouses. Equally, Thirisha et al. (2023) utilised an IoT system to collect data about the environment and, using a cloud-based platform, analysed and processed the data to generate useful insights to farmers to improve their crop yields and reduce water wastage.

IoT sensors collect data in real time. Therefore, one of the challenges is the need for the devices to have internet connectivity and always stay powered (Mendes, et al., 2020). Other challenges include the reliability, accuracy and security of the data collected. Interference can occur owing to the fact that IoT devices mostly use the unlicensed spectrum, inadequate security can lead to loss of data and breach of privacy, and the IoT devices are deployed in outdoor environments where they are exposed to hazards such as harsh weather conditions and animals (Elijah et al., 2018).

To make IoT more accessible, technology companies like Microsoft have developed smartphone sensors that can replace standalone sensors (Vasisht, et al., 2017). Sensor integration into smartphones and other tools developed by Microsoft's Azure FarmBeats promise to deliver a platform with negligible downtime, support for sensors with capacity for a wide range of requirements, cloud connectivity and freshness of data (Vasisht, et al., 2017).

Use of IoT technology can result in ethical issues of privacy and security. These ethical risks arise since malicious people can use the data to manipulate the farmers into purchasing specific products when the data is in the wrong actors. Breach of privacy issues occurs since IoT devices use big data by availing real-time agricultural solutions to enable effective decision-making (Elijah et al., 2018). The notable privacy problems imply the need to invest in security to protect data that may be used to affect food security negatively. Regulatory and governance frameworks should also be established to mitigate data privacy and security challenges.

2.4.2 Unmanned Aerial Vehicles

Unmanned Aerial Vehicles (UAV) are defined as aircraft without a human operator onboard but instead fly autonomously or are remotely controlled (Norhashim, et al., 2023). They are also referred to as drones; therefore, the names Drone and UAV can be used interchangeably. According to Insider Intelligence (2021), UAV technology has in the recent past found application in disaster rescue operations, law enforcement and surveillance, storm tracking, package delivery, video photography, geographic mapping of physically inaccessible spaces, and precision farming. Their popularity is driven by their relative low cost, ease of use and control, operation in hard-to-reach areas, and flexibility in use for a wide range of tasks, compared to other aerial methods such as satellites and manned aircraft (Amirgaliyev, et al., 2023).

Drones are the most used remote data collection technology for precision farming. They carry payloads such as sensors and photography equipment. This makes them useful in applications including monitoring the growth of vegetation, weed mapping and management, and can be integrated with smart sensors (Amirgaliyev, et al., 2023). Some authors such as Klauser and Pauschinger (2021) have even proposed that precision farming will be the most invested area in UAVs.

Some significant drawbacks to the successful adoption of UAVs include inadequate trained and licenced personnel to implement UAVs at the farm level, regulation and prohibitive cost of entry (Howe, 2023). The benefits of precision farming, however, outweigh the drawbacks. In China, for example, the adoption of drones for farming is progressive, with over 50,000 drones in use (Qingqing, 2020). Another challenge to the implementation of UAVs lies in their regulation. According to Ayamga et al. (2021), regulation of the use of UAVs is necessary due to their impact on safety, privacy, data protection. Countries in Sub-Saharan Africa are divided into those that have UAV regulation policies in place, those that do not have regulation laws, and some that are grappling with the legal framework; Kenya falls into the latter category, since its initial drone regulation was withdrawn due to a contention on the taxation (Ayamga et al., 2021). Even with this setback, it goes to show the dedication of the authorities in seeing that proper regulation is put in place and enforced.

2.4.3 Geographic Information Systems (GIS)

A Geographic Information System (GIS) is a computer system that captures, stores, analyses, and displays data about the earth's geography (USGS, 2023). GIS uses information that includes location in it to map data; this can be expressed as a zip code, address, or, more frequently, latitude and longitude coordinates.

GIS can analyse different kinds of data, such as landscape, rivers, schools, roads, electric power lines, human population etc., and show how the data relates to each other. Therefore, GIS is used in a wide area of application, in fields such as healthcare, education, urban planning, environmental management, and agriculture. All the collected data will need to be mapped onto a cartographic database. Data collected using the various primary collection methods (IoT sensors, remote sensors, imagery from satellites or drones), and demographic data is uploaded into GIS; this is referred to as data capture (USGS, 2023).

Useful insights from precision farming are location specific. This therefore makes GIS a very important tool. Some experimental works, such as Dunaieva et al. (2019), noted the various benefits of using GIS. The authors created a system to assess the state of winter crops by region by including various variables like type of crops, water, and soil attributes. The system provided collection, storage, and data processing, which would be crucial for Crimea, the location where the study was based. Equally, GIS was used to evaluate land suitability in Southern Iran for wheat production (Tashayo et al., 2020). From the analysis of the Iran study, 25% and 38% of the of the area studied was highly and moderately suitable or not suitable for wheat farming, respectively, whereas 27% and 8% of the area was either marginally or suitable for wheat farming. Based on these results, GIS can be crucial for site-specific soil management, land-use planning, and protection of the environment by performing GIS examinations.

Even with its significant benefits, there are a few considerations for its successful application in precision farming. First, there is a need for high resolution spatiotemporal data and skilled personnel to collect GIS data. GIS technology can be complex to use. Additionally, GIS software and hardware can be expensive and therefore prohibitive to most farmers. The data collected from GIS can sometimes be erroneous, making it unusable (Mathenge et al., 2022). Therefore, when developing GIS infrastructure, it is important to integrate the different standards to ensure the data is stored with similar spatial units.

2.4.4 Cloud Computing

The National Institute of Standards and Technology defines cloud computing as the delivery of a shared pool of computing resources over the internet, on demand. These computing services include servers (compute and memory resources), storage, networks, software, applications, analytics, and intelligence (Mell & Grance, 2011). The term cloud arose to depict the fact that these services are provided over the internet (a series of interconnected networking equipment), which is diagrammatically often represented as a "cloud" (Comer, 2021).

Data uploaded into cloud computing platforms is processed into various databases that store information on the farm and crop primary attributes such as soil moisture levels, pH levels, climate, extreme weather events, fertilizer, pests and diseases. The ability of cloud computing to provide intelligence and analytics is critical to delivering the benefits of precision farming to smallholder farmers.

Providers of cloud services such as Amazon Web Services and Microsoft Azure allow us to upload data from the primary collection tools and process the data into custom insights that can be delivered to the farmers. Choosing the right cloud computing platform can be a challenge. Krisnawijaya et al. (2023) propose a multi-criteria decision analysis approach as a systematic way to select the most feasible cloud computing platform for precision farming applications.

Complete ecosystems exist which integrate IoT devices to cloud platforms and analytics solutions, to provide data capture, data visualization, public cloud computing, and data analytics in a unified platform. Such solution providers include Ubidots, ThingWorx and Phytech (Sharma et al., 2023). The public cloud service providers also incorporate machine learning modules such as Amazon SageMaker, Google Cloud Vertex AI and Microsoft Azure Machine Learning (Panwar, 2023).

2.5 Machine Learning in Agriculture

Sheikh et al. (2023) define artificial intelligence (AI) as the capability of computing systems to exhibit capabilities that are characteristic of human intelligence. AI has gained momentum in recent years, thanks to the advancement in the techniques that it employs to empower computational systems to exhibit human-like behaviour. These techniques are illustrated in Figure 2.1.

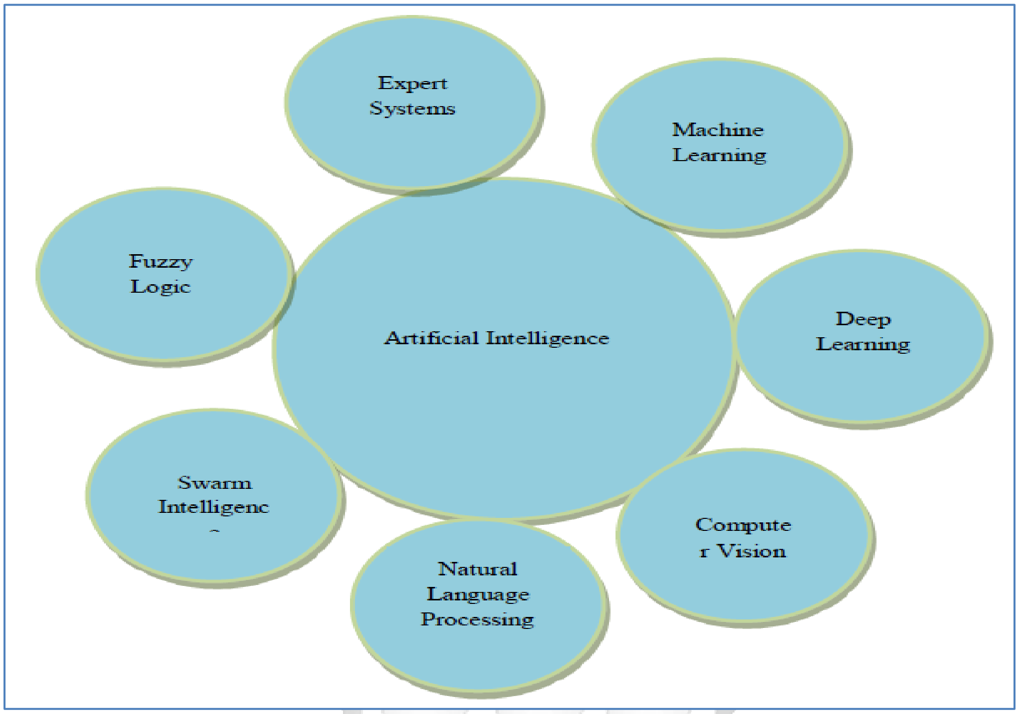


Figure 2.1: Techniques That Power Artificial Intelligence (Sharma, et al., 2020)

One of these, Machine Learning (ML), is the backbone of modern AI. Machine Learning uses advanced algorithms to detect patterns in large data sets, allowing machines to learn and adapt (Zhuhadar & Lytras, 2023). Availability of large datasets, and improved computing power capable of processing the large datasets has opened the avenue for the opportunity to explore and apply more intricate subsets of machine learning, such as deep learning (DL) (see Figure 2.2), whose algorithms can be trained from both structured and unstructured data, making Machine Learning applicable in diverse fields including image processing for pattern identification, automation of industrial processes and natural language processing for customer service applications (Zhuhadar & Lytras, 2023).

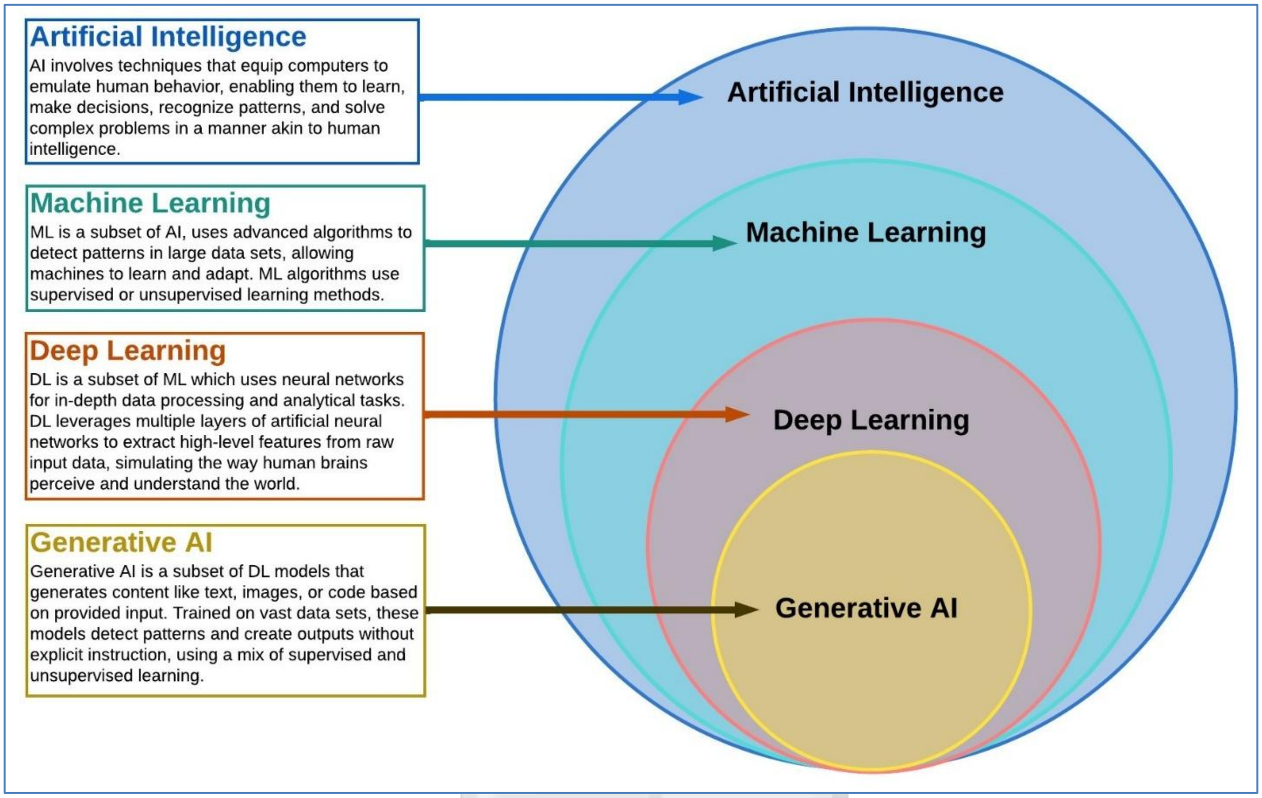


Figure 2.2: Machine Learning, Deep Learning and Generative AI (Zhuhadar & Lytras, 2023).

Data from precision farming tools such as UAVs is being used to predict and manage crop pests and diseases by applying machine learning methodologies to the data. Sharma et al. (2020) opined that images captured by UAVs can be processed using DL algorithms for disease and weed control. Oppenheim and Shani (2017) used photos collected from potato tubers to train a CNN model and evaluate its performance in classifying potato diseases. Similarly, Sumbasivam and Opiyo (2020) used a CNN-based ML model to detect and classify cassava diseases. Adopting machine learning methods to detection, elimination and prevention of crop pests and diseases can result in optimal use of pesticides, therefore minimizing environmental damage due to overuse, and saving costs for the farmer. According to Lee and Yun (2023), with conventional methods of controlling pests and diseases, only 0.1% of the used pesticides eliminate the target pests, with the rest being disbursed into the environment, causing pollution. Deep learning models can be used for prediction and early detection of disease, therefore minimizing the need for pesticide use (Lee & Yun, 2023).

2.6 Conceptual Framework

The literature review plays a key role in conceptualizing the study. This chapter reviewed the factors that influence agricultural productivity among smallholder farmers. Existing agricultural research initiatives, and use of mobile technology in addressing agricultural advisory and information needs of farmers, were studied, their usefulness analysed, and opportunities for improvement identified. Key precision farming technologies that are essential to availing relevant data on crops, weather, weeds, pests and diseases were introduced. Cloud-based platforms and machine learning techniques, which are useful in processing the raw data into useful insights, were also reviewed. The conceptual framework provides the link between the literature study and the design and implementation objectives. The system comprises a machine learning module that is trained using images captured of crop profiles at different levels of pest and disease infestation, as well as healthy crops. From this training, the machine learning module can judge the type of infestation, given an input by the system's end user. The machine learning module is embedded into a mobile application. Using the mobile application, the end-user can upload images of their infested crops, and the system makes a diagnosis of the infestations on their crop fields and gives advisory on preventative and remedial actions. The conceptual model is presented in Figure 2.3.

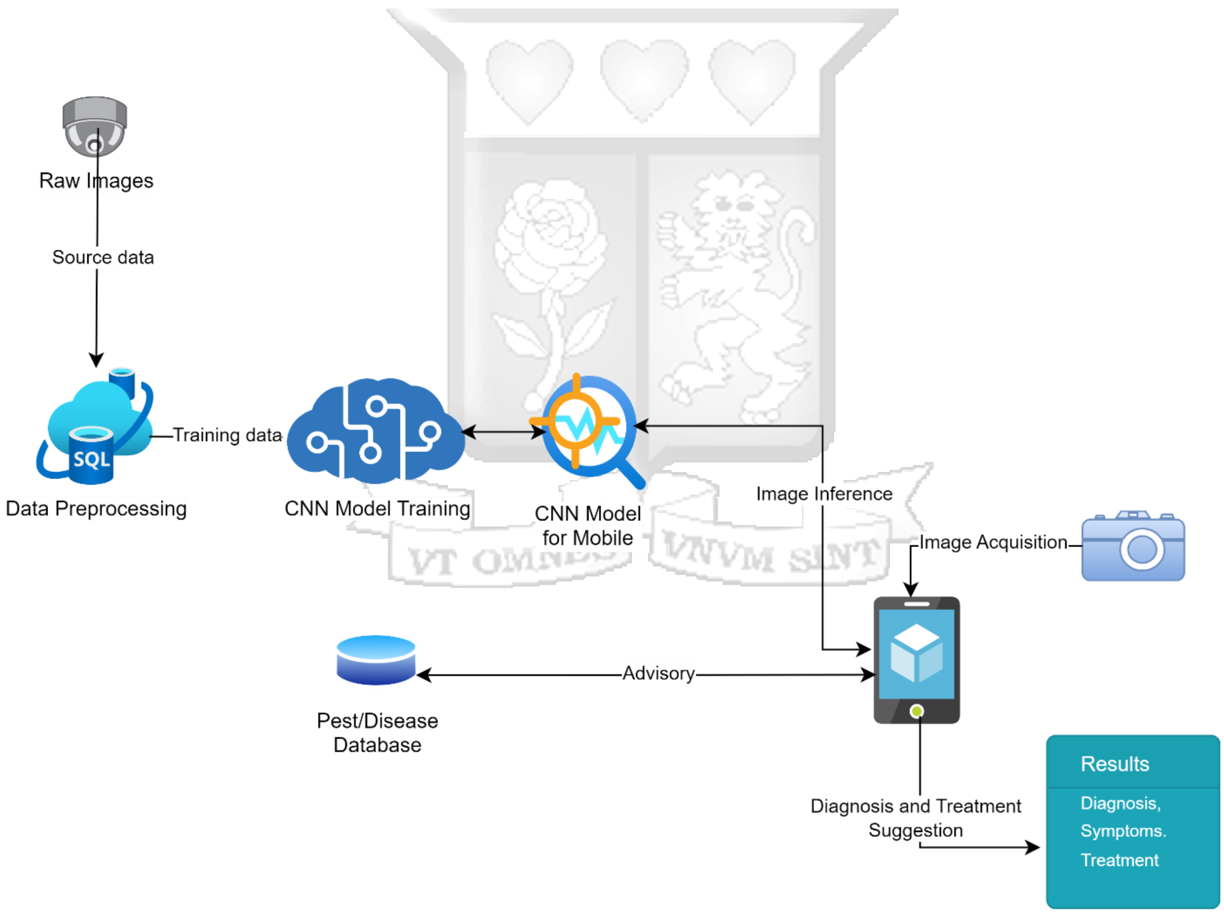


Figure 2.3: Conceptual Model of the Tomato Pest and Disease Detection System

Chapter 3: Research Methodology

3.1 Introduction

This chapter discusses how the research methodology that was employed to address the research questions. It describes the methods used to conduct the research and the software development methodology that was adopted to build the system. The approaches applied in system analysis, system design, implementation and testing are explored.

3.2 Research Design

The research used an experimental research design. Reviewing existing literature and online questionnaires were the main data collection tools applied. The literature review was used to understand existing agricultural technology solutions, their strengths and weaknesses, and gain a deeper understanding of the digital technologies that can be applied to improve agricultural productivity. The digital technologies explored were precision farming technologies and machine learning. The online questionnaires were used to collect data from the target population. Subsequently, a mobile-based image recognition system was designed, that would help address farmers' need for a solution to mitigate the food production challenges. The system was tested against the use cases derived for this research, to establish its efficacy.

3.3 Software Development Methodology

Prototyping software development methodology was adopted to develop the system. The prototype methodology performs the system analysis, design and implementation phases concurrently, with all the three phases being performed repeatedly until the system is completed (Dennis et al., 2020).

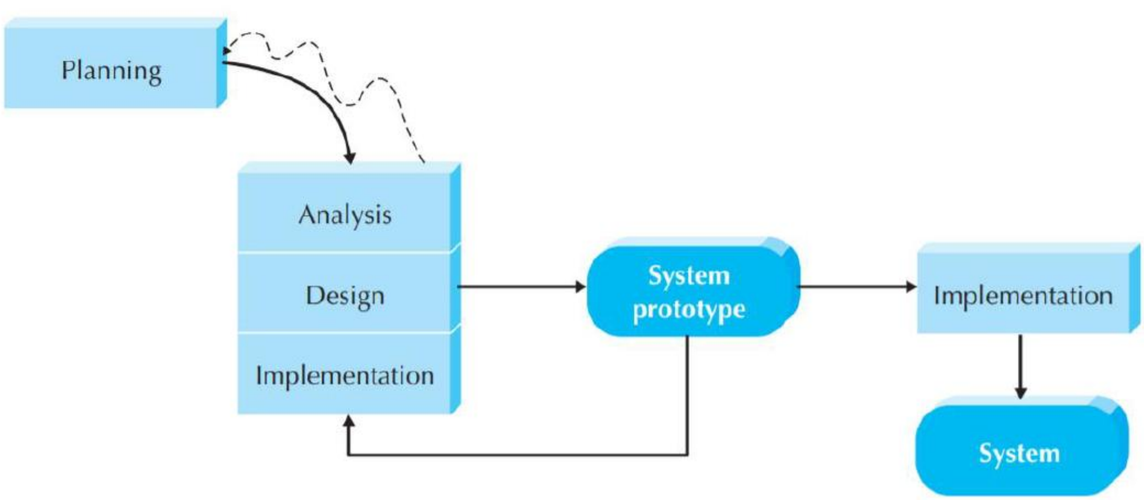


Figure 3.1: Prototyping System Development Lifecycle (Dennis et al., 2020)

Figure 3.1 illustrates the prototyping system development lifecycle. The basic prototyping approach is as follows; Once analysis and design are performed, work begins on a system prototype that provides a minimum viable product. The prototype is evaluated by the stakeholders, and the feedback is used to reanalyse, redesign and reimplement a second prototype. This cycle continues until all the stakeholders agree that the system provides enough functionality to be approved for release (Dennis et al., 2020).

3.4 System Analysis

System analysis involved gathering user and CNN model training data requirements, as well the hardware and software requirements for the convolutional neural network and the mobile application.

3.4.1 User Requirements Determination

A survey was carried out to determine the farmers' need for the solution. The survey was delivered through a Google Forms questionnaire, as presented in Appendix C. The target population was tomato farmers in Kajiado county, estimated to be over 2000 farmers. Guided by Naing, et al. (2022), the researcher employed simple random sampling, with a confidence level of 95%, an expected prevalence of 50%, and a confidence interval of $\pm 16\%$, to achieve a desired sample size of 38 farmers.

3.4.2 System Requirements Definition

Training deep neural networks can take days or weeks, since it requires large amounts of computational power. To minimize the computational resources and time required to train the model, the research employed transfer learning. Google Cloud Platform (GCP) was utilized to provide computational resources. The TensorFlow machine learning framework was employed to create the CNN model. This was implemented in Google Colaboratory (Google Colab). Android studio was used to develop the mobile application.

3.5 System Design

The researcher adopted an object-oriented approach to developing the system. In object-oriented approaches, a problem is decomposed into objects that contain both data and processes, thus balancing the emphasis between process and data (Dennis et al., 2020). Unified Modelling Language (UML) is a standardized modelling language for object-oriented software development.

It is use-case driven, architecture-centric, and emphasises iterative and incremental development (Liang & Jin, 2020).

The research follows the UML principles. Use case diagrams were used to illustrate how the different actors interact with the system. Use cases are fundamentally simple since they focus on one process at a time. Sequence diagrams were used to illustrate the interaction between objects in the system. Class diagrams were used to describe the structure of the system, showing the class attributes, methods and the relationship between the class objects. An Entity relationship diagram (ERD) was used to design the database. Draw.io is the diagramming software that was used to generate the various system models.

3.6 System Implementation

3.6.1 CNN Model Training

A convolutional neural network is a specialized type of deep learning algorithm that is most suitable for image recognition and classification. The CNN model was developed using the following steps: data acquisition, data pre-processing, and model training.

To achieve a large enough dataset for effective training of the CNN model, image data of tomato crops at different stages of infestation with disease as well as healthy tomato crops was obtained from public sources of images such as the Kaggle data science community. The images were classified into six directories, representing five different disease classifications as well as a class of healthy crop. The images were cropped and resized to a standard size 224 x 224-pixel square images of JPEG format. Data pre-processing techniques including data normalization, pooling, fine-tuning and dropout were applied to improve the diversity of the data.

Training CNNs usually requires large amounts of data. Using a pre-trained model is referred to as transfer learning. Transfer learning can train deep neural networks using comparatively less data and require less computing power. A Keras model for classifying five tomato diseases was built using a MobileNetV3 pre-trained model for image feature extraction that has been trained on a much larger image dataset in ImageNet. MobileNetV3 was chosen, since it is specifically optimized for use in mobile and embedded devices. 20% of the data was used to validate the trained model. The trained deep learning neural network was saved as a Keras model.

3.6.2 Mobile Application Implementation

The mobile system was implemented using the Android Studio Integrated Development Environment (IDE). Following the prototyping methodology, the prototype was iteratively developed. The trained image classification model was converted to a TensorFlow Lite version and deployed to the mobile application. Firebase Realtime Database was used for persisting data locally in the mobile application, and MySQL for the online database. The CodeIgniter PHP framework was used to develop the administrative backend.

3.7 System Testing and Validation

Tests were carried out to validate the functionality of the system. These tests were divided into functional and non-functional tests. Functional testing was done against the system functional requirements, which were well articulated in the use cases. Therefore, the functional testing followed the use cases outlined. Non-functional testing was carried out to validate the non-functional requirements of the system. These tests included unit testing, compatibility testing, and usability testing. Unit testing was used to test the individual object classes, to determine whether they meet the requirement specifications. Compatibility testing was used to ensure correct interworking of the different parts of the system, and on different android versions. Usability testing covered the end-user's experience when interacting with the system.

3.8 Ethical Considerations

Ethical considerations were observed in undertaking this study. The primary data utilized to train the image classification deep learning model was obtained from the Kaggle machine learning and data science community and is publicly available data, under a Creative Commons CC0 1.0 licence. Adequate attribution to the data author was done: This is done in section 5.3.1. Utmost care was taken to ensure that the information collected from the target population was used solely for the purpose of this research. A participant information sheet was shared with the survey participants, to make them aware of the purpose of the study, and how the information collected from them would be used and protected. It also included a consent form, that allowed the participant to, or not to, voluntarily participate in the survey, by signing the form. All previous works were appropriately cited and the authors acknowledged. Ethical approval to carry out the research was obtained from the Strathmore University Institutional Scientific and Ethics Review Committee (SU-ISERC). A research license was also obtained from the National Commission for Science, Technology and Innovation (NACOSTI).

Chapter 4: System Analysis and Design

4.1 Introduction

This chapter presents the analysis and design of the tomato crop pests and disease detection system. This incorporates analysis of functional and non-functional requirements, design of the system components, and the interaction between the different components as well as between the user and the system, in the form of use case, sequence, data flow, and class diagrams. The system architecture was also outlined, in line with the conceptual framework presented in section 2.7.

4.2 System Analysis

System analysis helps in defining the characteristics of the system that meet the needs of the end user. The literature review was used to identify the challenges facing small-scale farmers in Kenya, existing mobile applications for agriculture, their strengths and weaknesses, and how precision farming can be employed to alleviate these challenges. Qualitative research was used to validate the findings in the literature review. This involved questionnaires to farmers. Based on the requirements gathering done, the system functional and non-functional requirements were documented.

4.2.1 Requirements Analysis

A survey was carried out to determine the farmers' need for the solution. The survey was delivered through a Google Forms questionnaire, as presented in Appendix C. A total of 36 respondents participated in this survey. The distribution of the respondents, by gender, age group and education level is as shown in Figure 4.1, Figure 4.2 and Figure 4.3 respectively.

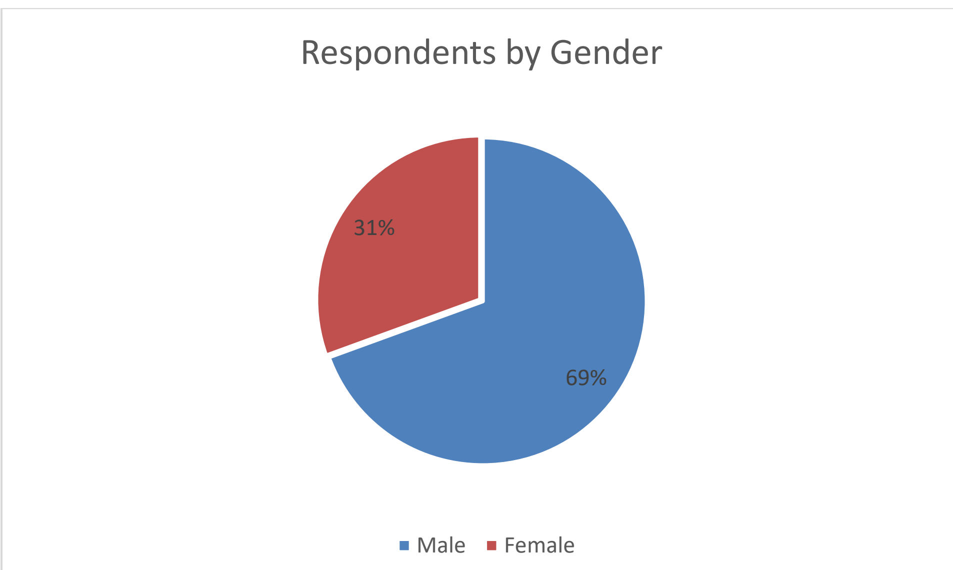


Figure 4.1: Respondents by Gender

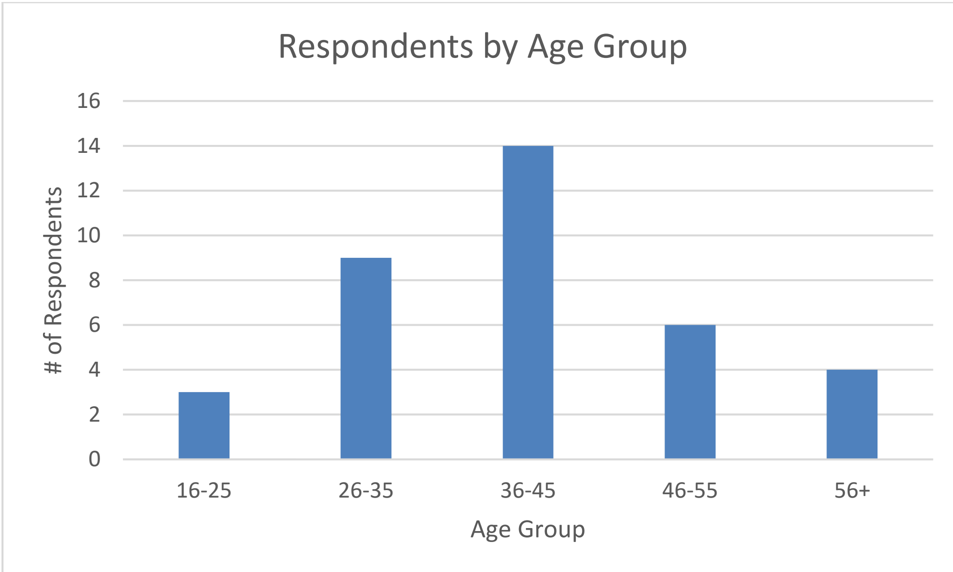


Figure 4.2: Respondents by Age Group

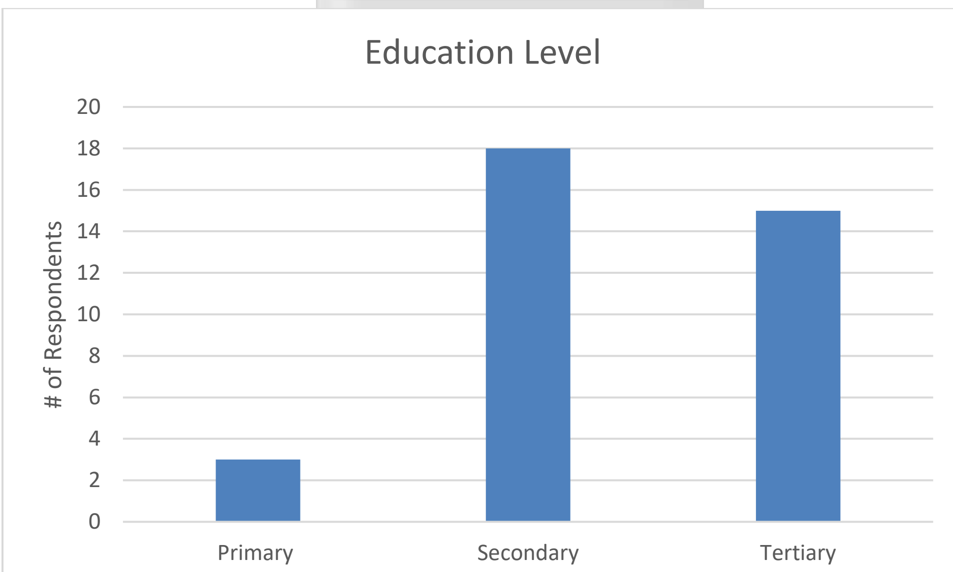


Figure 4.3: Respondents by Education Level

The survey also set out to determine the size of land owned by the farmers. The survey revealed that 66% of the farmers own farm sizes of less than 5 acres. Figure 4.4 summarises the size of land owned by the farmers.

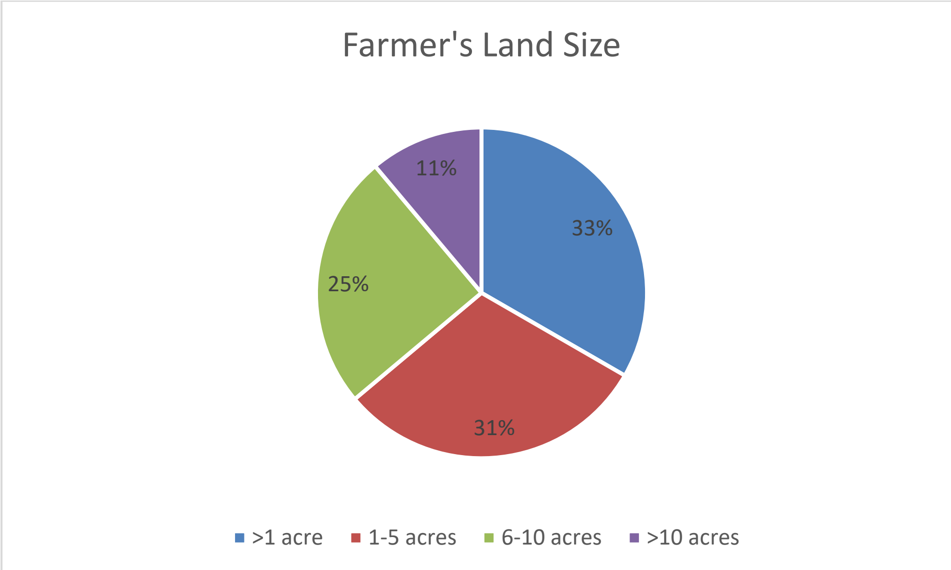


Figure 4.4: Farmer’s Land Size

Most of the respondents interviewed, 32, owned an Android phone. 3 respondents answered that they own a feature phone, and only one respondent answered that they do not own a mobile phone. The survey sought to capture how the farmers use their mobile phones. Table 4.1 illustrates the ways in which the farmers use their mobile device.

Table 4.1: How farmers use their mobile phone

Ways in Which I Use Mobile Phone	**# of Respondents**	**% of Respondents**
Keeping in touch with Family and Friends	36	100%
Accessing social media	16	44%
Make Payments and Receive Money Using Mpesa	30	83%
Conducting Business for my Farm	12	33%
Looking for Buyers and Sellers of my Farm Produce	24	67%
Finding Information About How I can Improve my Farming	26	72%

When asked if they felt the mobile phone was useful in enabling them to improve their crop production, 23 farmers responded that the mobile phone was indeed an enabling tool in improving crop production. All the respondents’ responses were as shown in Figure 4.5.

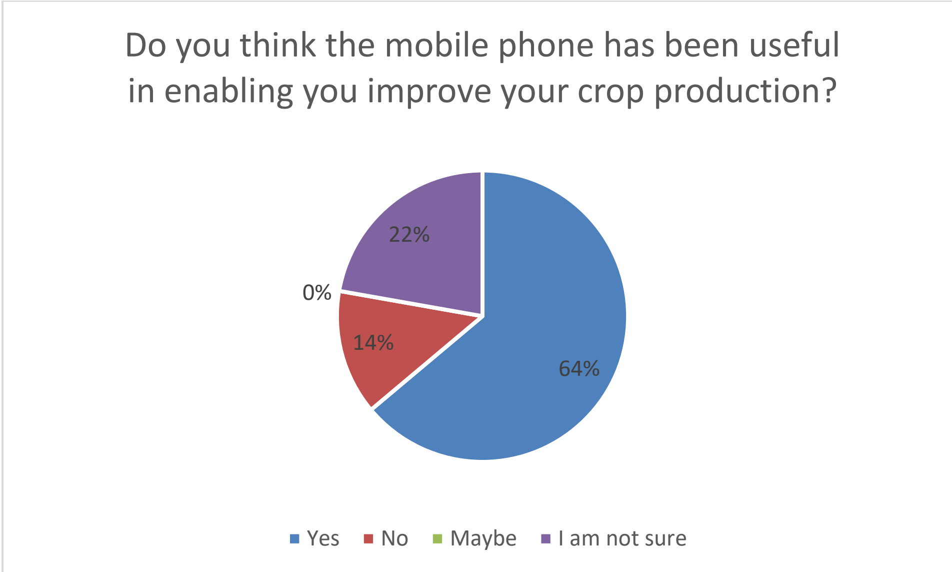


Figure 4.5: Role of the mobile phone in enabling farmers improve their crop production

Finally, the farmers were asked whether they would consider using a mobile-based application that helps them identify and prevent tomato disease. 25 farmers answered that they would consider a mobile application-based crop disease detection system. Figure 4.6 captures the responses.

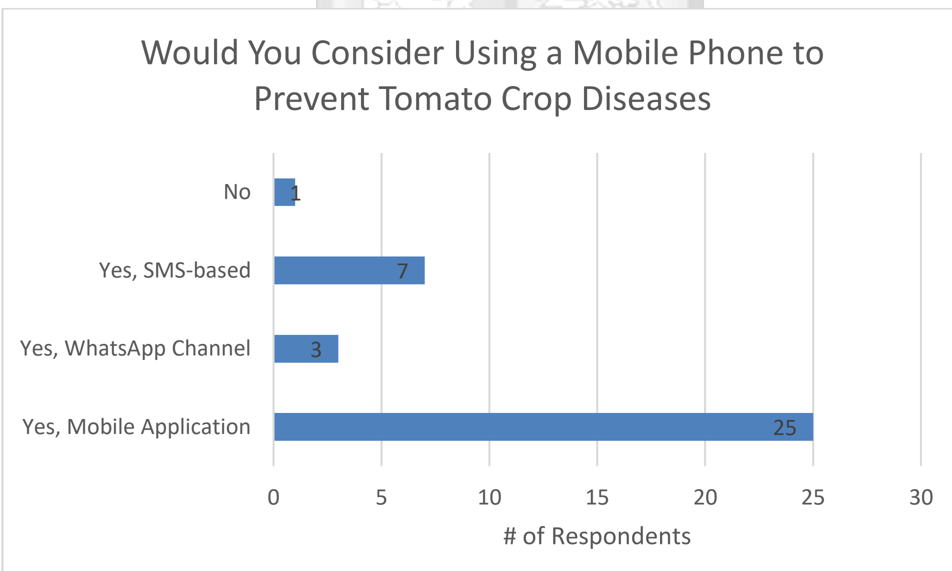


Figure 4.6: Farmers’ consideration for using a mobile-based crop disease detection system

4.2.2 Functional Requirements

Functional requirements define what the system does. They are the features that the system must have to meet the user needs. The functional requirements for this system are:

i. The system should allow the user to take a picture using their mobile device and use this photo as an input for disease classification system. The system should also allow the user to upload a picture from their phone's image gallery.
ii. The system should be able to classify the type of disease affecting a crop, based on the image that is uploaded.
iii. The system should allow the user to register and create a profile.
iv. The system should save the user's classification results, for future reference.
v. The user should be able to view their historical queries.
vi. The system administrator should have full administrative access to the system and should therefore be able to manage users as well as crop disease information.

4.2.3 Non-functional Requirements

Non-functional requirements define how the system works. They are the system's operational capabilities. The non-functional requirements for this system are:

i. The mobile application should have a user-friendly user interface that conforms to common user interface design guidelines.
ii. The system should be able to perform disease classification and return a result within a reasonable time.
iii. The system should be stable, to ensure availability and usability whenever the farmer needs to use it.
iv. The system should be secure, ensuring that the user's data is encrypted.

4.3 System Design

The system analysis guided the research towards utilizing images of tomato crops to train a deep learning system that classifies tomato diseases, and provides insights to a farmer, based on image inputs from the farmer's mobile device. System design seeks to model how the system works. It specifies the system's architecture, interaction between the system components, and data flow in the system. This research adopted the object-oriented design (OOD) approach to develop the system. Unified Modelling Language (UML) was used to visualize the system.

4.3.1 System Architecture

The architecture of the crop pest and disease detection system is shown in Figure 4.7. The system comprises a deep learning image classification model that is trained to identify crop pests and diseases. Raw crop images are pre-processed to create datasets for training the model. Once the model training is completed, it is optimized for application in mobile devices. The optimized model is embedded into the mobile application. The android application forms the user interface to the system. The user uploads a picture of a crop for which they need an analysis to identify the crop pest or disease infesting the crop. An on-board inference is performed, and the user is fed back information relating to the crop pest or disease. The information can optionally be stored locally in a database for future use. Firebase Realtime Database was the preferred on-device database solution, with MySQL being utilized to provide online backup. Firebase Realtime Database is the archetype NoSQL for real-time data synchronisation and offline operation.

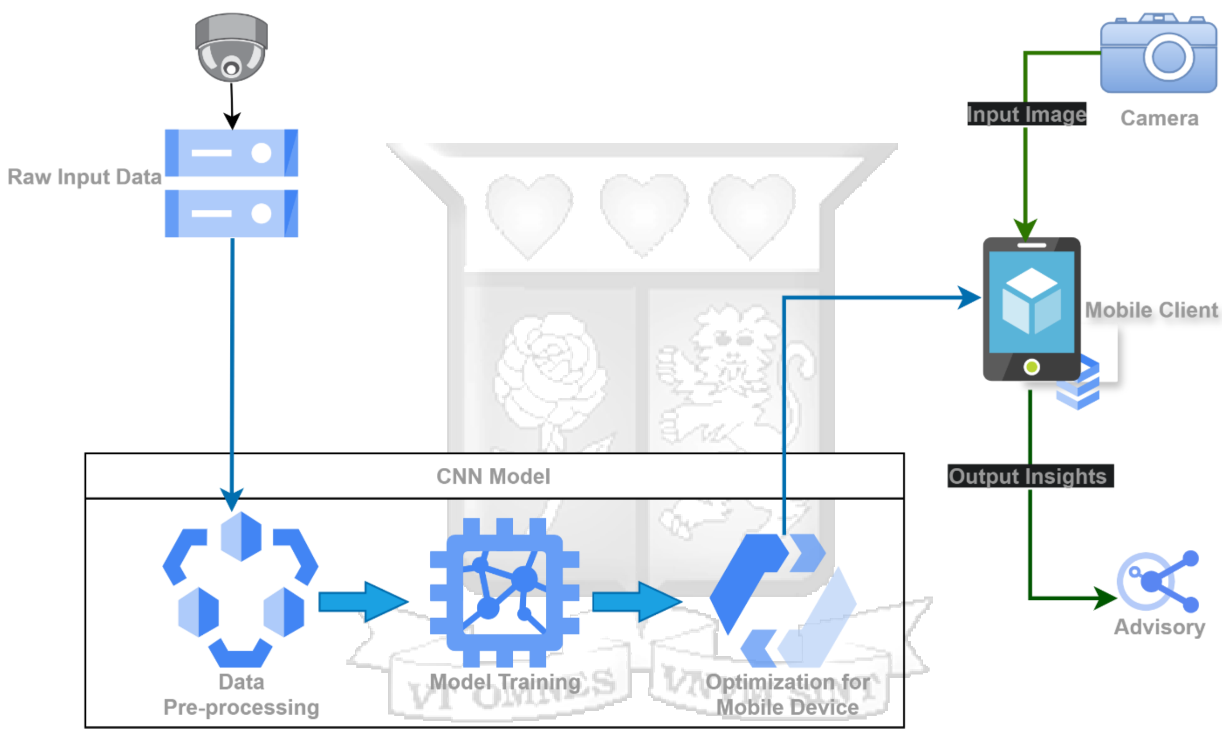


Figure 4.7: System Architecture

4.3.2 Use Case Diagram

The use case diagram illustrates how the system interacts with its environment. The actors identified for this system were the user or farmer, and the system administrator. The users will interact with the system through the mobile application. The system administrator will have access to the system through the web portal backend. Figure 4.8 shows the use case diagram for the crop pest and disease detection system.

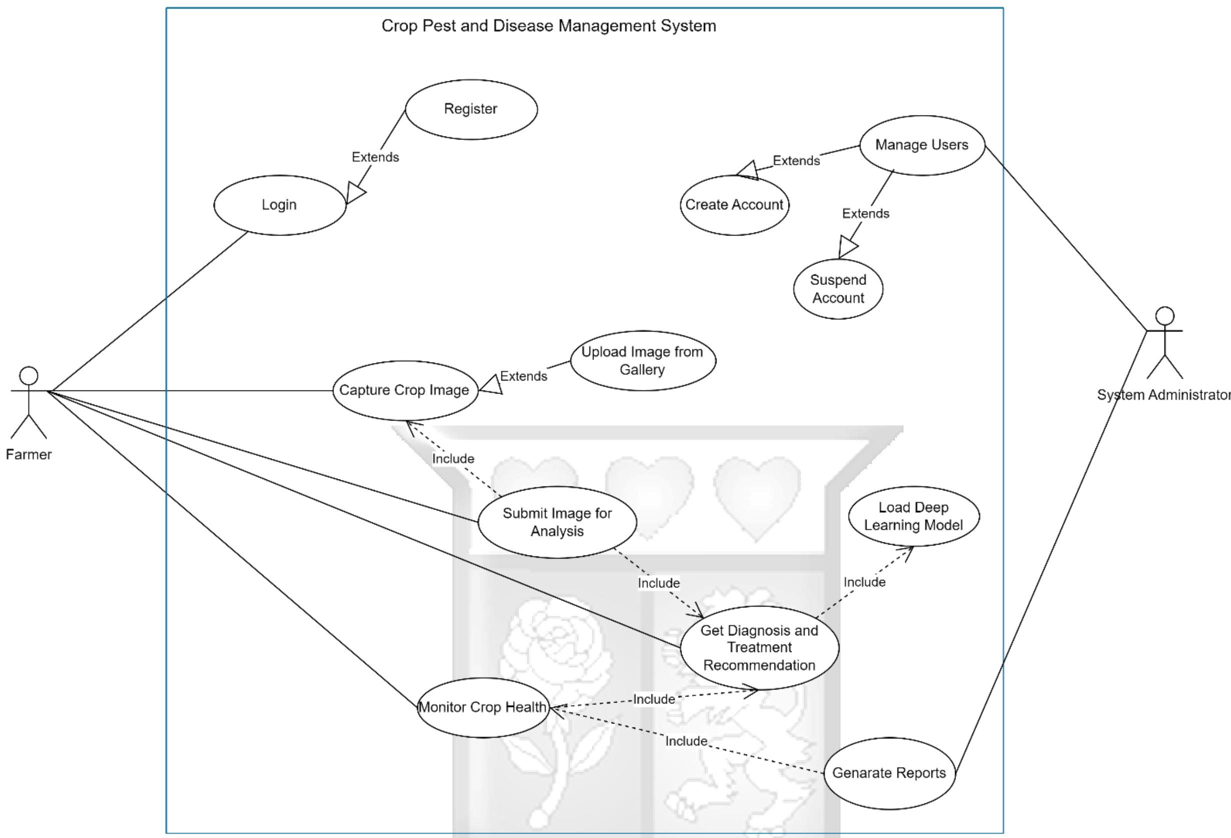


Figure 4.8: System Use Case Diagram

4.3.3 Use Case Descriptions

Table 4.2: Register and Login

Use Case:	Register and Login
Primary Actor:	Farmer/User
Description:	The user will optionally register themselves in the mobile system. The system verifies and saves the registration details. The system allocates a unique ID to the user for identification & tracking purposes. The system verifies and authenticates the user.
Preconditions:	The user has access to the mobile application. To register, the user must not have an existing account/active registration. To login, the farmer must have a registered account.
Post conditions:	New user is added to the system. The user is authenticated and granted access to the system

Table 4.3: Capture Crop Image and Upload Image from Gallery

Use Case Name:	Capture Crop Image/Upload Image from Gallery
Primary Actor:	Farmer
Description:	The mobile application prompts the farmer to take a picture of the crop. The farmer takes a picture of the crop or uploads a photo from their image gallery. The mobile application allows the farmer to preview the picture taken, with an option to retake or save the picture. Also, if the picture is not clear, the mobile application prompts the farmer to retake the picture.
Preconditions:	The farmer has successfully opened the mobile application.
Post conditions:	The image is successfully uploaded into the mobile application, ready for inference.

Table 4.4: Submit Image for Analysis

Use Case Name:	**Submit Image for Analysis**
Primary Actor:	Farmer
Description:	The farmer selects the submit button, to submit image for inference. The mobile application loads the image to the embedded machine learning model.
Preconditions:	The farmer has successfully uploaded the image.
Post conditions:	The image is submitted for inference.

Table 4.5: Get Diagnosis and Treatment Recommendation

Use Case Name:	**Get Diagnosis and Treatment Recommendation**
Primary Actor:	Farmer
Description:	The embedded machine learning model analyses the uploaded image and detects potential disease based on the pre-learned patterns and generates a result. The mobile application displays the result of the identified disease to the farmer. The mobile application provides an option for the farmer to view more information about the disease, such as the severity of the disease, preventive measures and treatment options. The mobile application stores the uploaded image, along with the diagnosis results and recommendations in a database.
Preconditions:	The crop image has been submitted for analysis.
Post conditions:	The farmer gets the disease diagnosis, along with treatment recommendations and preventive measures. If no treatment is recommended, the farmer is advised accordingly.

Table 4.6*:* Monitor Crop Health

Use Case Name:	Monitor Crop Health
Primary Actor:	Farmer
Description:	The farmer navigates to the "Track Crop Health" section of the mobile application. The mobile application displays a list of previous crop images uploaded and the disease diagnoses. The farmer can select a specific item and view, in detail, information about the disease diagnosis and treatment and prevention recommendations.
Preconditions:	Images that the farmer has previously submitted for diagnosis, together with the diagnosis results and recommendations are stored in a database.
Post conditions:	The farmer can view historical data of past diagnoses.

Table 4.7*:* Manage Users

Use Case:	Manage Users
Primary Actor:	System Administrator
Description:	The system administrator navigates to the "Manage Users" section of the web application to manages the user accounts. The system displays the list of active accounts. The system administrator can create user account, edit account details and delete user account.
Preconditions:	The system administrator is authenticated and has access to the system.
Post conditions:	New user is added to the system. User information is updated in the system. A suspended account is no longer available under the active accounts list.

Table 4.8*:* Generate Reports

Use Case Name:	**Generate Reports**
Primary Actor:	System Administrator
Description:	The system administrator can generate a crop health report under the “Generate Reports” tab in the web interface. The report will consist of crop health information, without user specific data.
Preconditions:	Historical disease detection enquiries must exist.
Post conditions:	A crop health report is generated.

4.3.4 Sequence Diagram

The sequence diagram illustrated in Figure 4.9 facilitates visualisation of the message flow between objects, in a sequential order. It derives from the use case diagram illustrated in Figure 4.8. and models the series of events that a use case must execute.

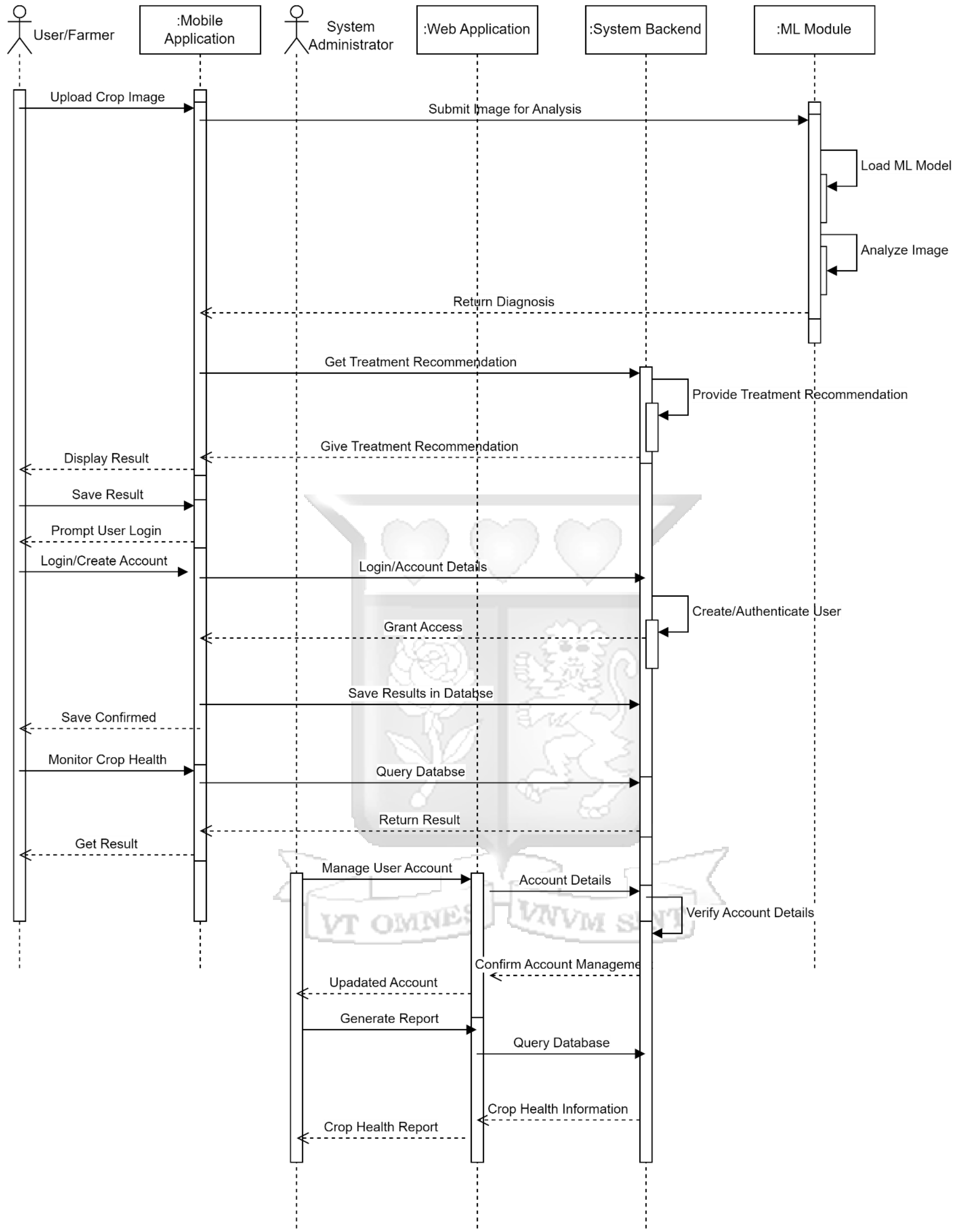


Figure 4.9: Sequence Diagram

4.3.5 Class Diagram

The class diagram in Figure 4.10 describes the structure of the system. The class diagram shows the different classes together with their attributes, methods, and the relationship among objects.

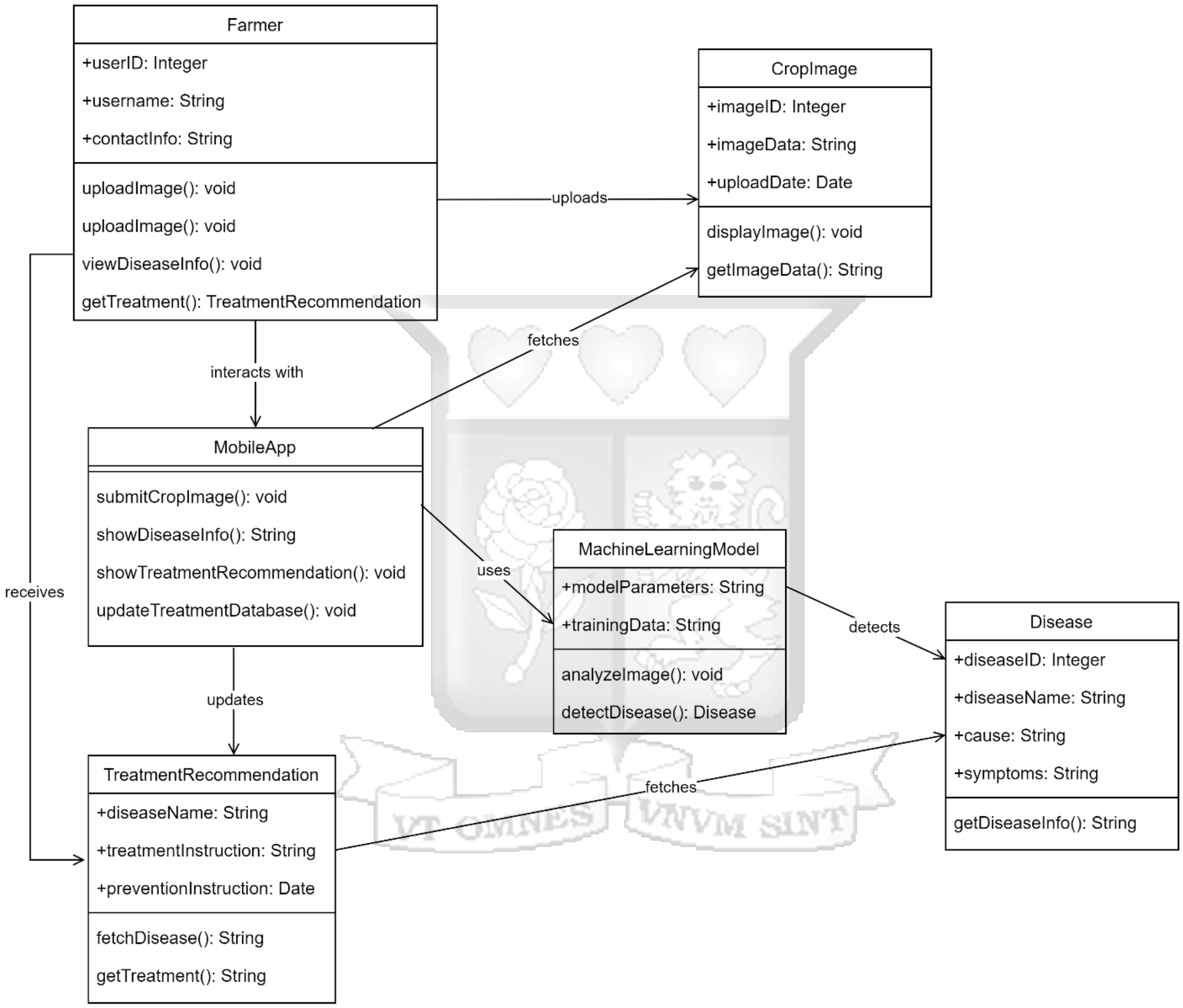


Figure 4.10: Class Diagram

4.3.6 Entity Relationship Diagram

The Entity Relationship Diagram in Figure 4.11 was used to show the database entities and their relationship.

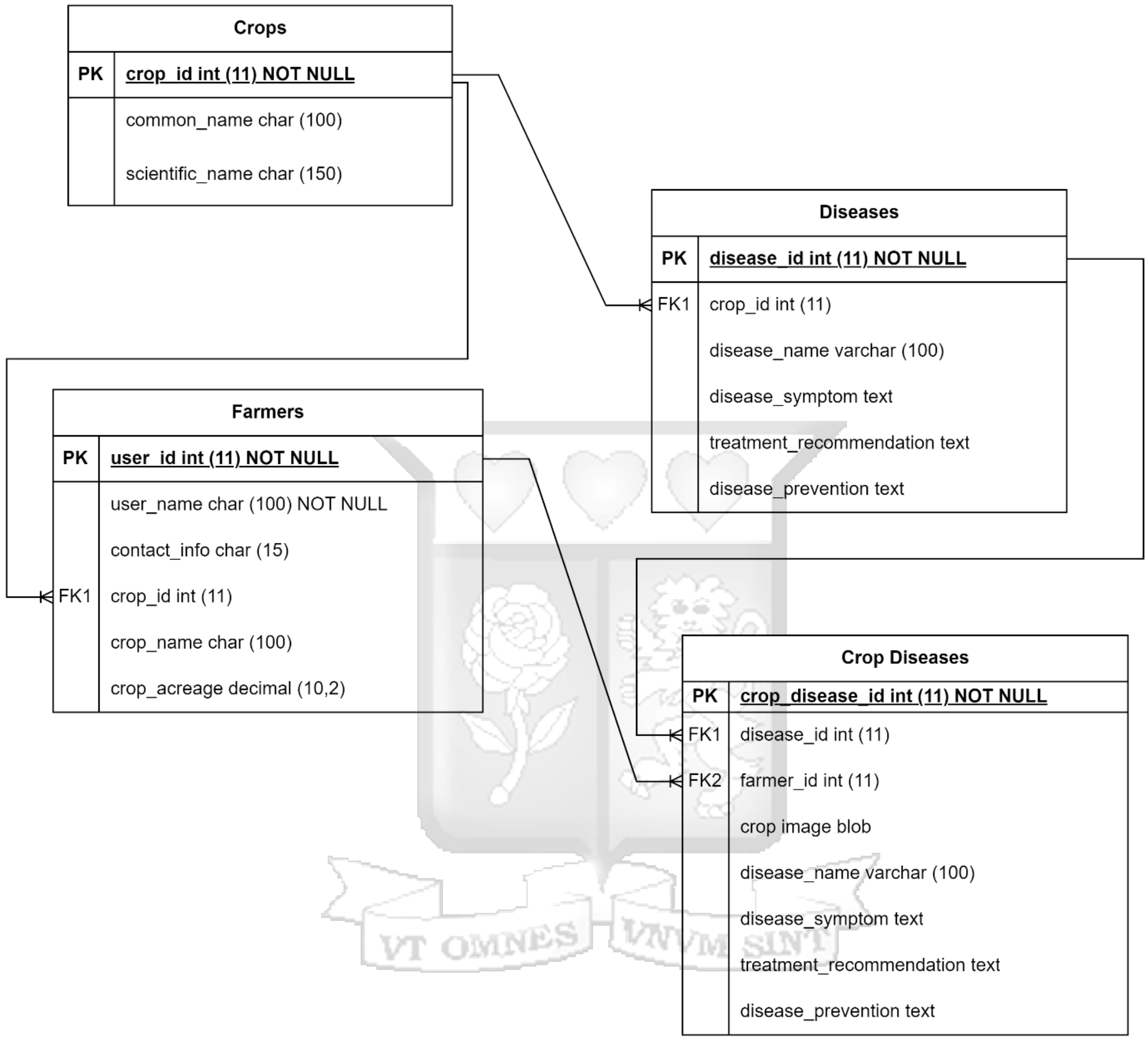


Figure 4.11: Entity Relationship Diagram

4.3.7 Mobile Application Wireframes

The user interface was modelled using wireframes, as illustrated in Figure 4.12. The user interface was modelled to be easy to use and uncluttered. The user is presented with the option to upload an image, either by taking a photo using the mobile device's camera or uploading a photo from the photo gallery. This action does not require the user to be logged in. Once the image is uploaded, the system makes an inference after which a task item is created giving details of the detected disease, its symptoms, treatment and prevention recommendations. To save this result, the user is now required to login into the system. If the user does not have an existing account, they will be prompted to register a new account. Once the user is logged in the inference results are saved automatically under their account. In future, the user can login into their account and retrieve inferences done in the past.

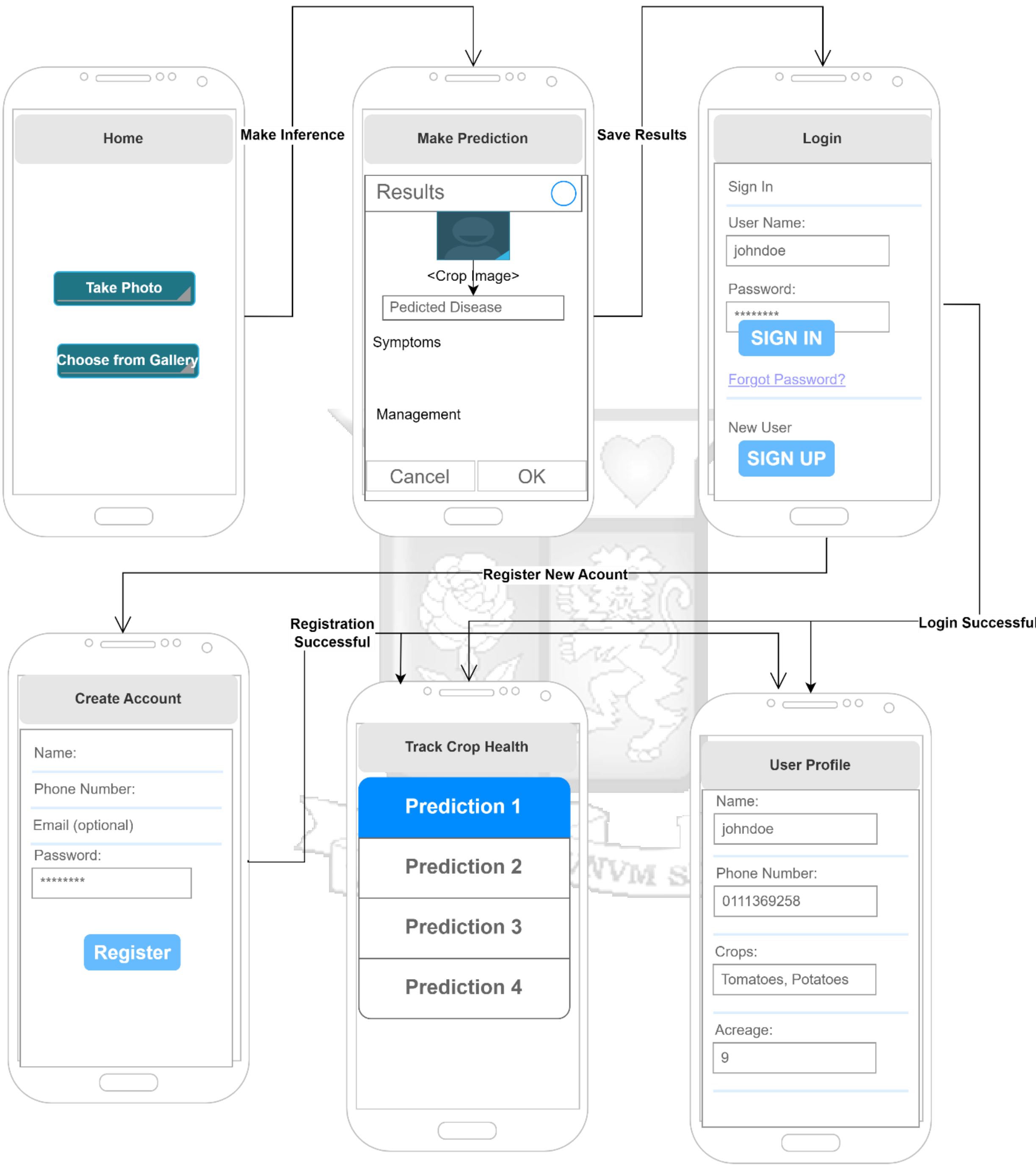


Figure 4.12: Mobile Application Wireframes

Chapter 5: System Implementation and Testing

5.1 Introduction

The system implemented in this research was aimed at detecting crop diseases and prescribing treatment for the disease, through the mobile phone. Following the system design in Chapter 4, this chapter discusses the implementation of the several components that make up the system, as well as testing of the system. The crop disease detection system was developed by retraining a deep learning neural network with an image dataset of classified tomato diseases and embedding the model into an android mobile application. Functional tests were performed on the system to ascertain that the system and its components performed as per expectation. Usability tests were also performed to determine if the system met the user requirements.

5.2 The Development Environment

The development environment encompassed hardware and software components. The deep learning neural network was implemented in Google Colab. Google Collab is a cloud-based Jupyter notebooks for implementing python libraries and the TensorFlow machine learning framework. Computational resources for the CNN model training and testing were provided by the Google Cloud Platform (GCP) compute engine, accessed through the Google Colab environment as well. Mobile application and system backend development was carried out on the researcher's personal computer.

The development environment specifications were as follows:

i. Google Compute Engine backend: Python 3 runtime, NVIDIA T4 GPU: 55GB System RAM, 15BG GPU RAM, 250GB storage
ii. Google Colab 1.2.0
iii. CUDA 12.5
iv. Python 3.11
v. TensorFlow 2.17.1
vi. Keras 3.5.0
vii. Matplotlib 3.10.0
viii. Pandas 2.2.2
ix. Numpy 1.26.4
x. Pillow (PIL Fork) 11.1.0

xi. HP Envy X360: Microsoft Windows 11 Operating System, 11th Gen Intel Core i7-1195G7 CPU: 8GB RAM, 500GB storage.
xii. Android Studio 2024.2.2
xiii. Kotlin 1.9.24
xiv. TensorFlow Lite 2.9.0
xv. Firebase Realtime Database 21.0.0
xvi. XAMPP 3.3.0
xvii. PHP 8.2.12
xviii. MariaDB 10.4.32

5.3 Image Classifier Implementation

5.3.1 Image Dataset Collection

The tomato crop image dataset was obtained from image datasets publicly available in Kaggle, under the Creative Commons CC0 1.0 licence (Source: Asish Motwani, Kaggle). Kaggle is an online community of data scientists that provides tools and resources for collaboration. The image dataset was classified, based on disease type, into early blight, leaf mold, tomato mosaic virus, septoria leaf spot and tomato yellow leaf curl virus diseases as well as healthy crop. The ensuing classified images were placed into the directory structure shown in Figure 5.1.

```
PS D:\ML Dataset\Tomato Crop Disease\train> tree
Folder PATH listing for volume Data
Volume serial number is 9E88-1BD4
D:.
├───early_blight
├───healthy
├───leaf_mold
├───mosaic_virus
├───septoria_leaf_spot
└───yellow_leaf_curl_virus
```

Figure 5.1: Dataset Folder Structure

5.3.2 Image Dataset Pre-processing

The image dataset was split into a training dataset, and a validation dataset that would be used to evaluate the model's performance: The training dataset contained 16,880 images and the validation dataset consisted of 4,507 images: An approximate 80:20 split. 20 images were also set aside for testing the resulting model's performance at disease detection. The image datasets were uploaded to Google Drive, from where the data could be pre-processed and used in Google Colab. Data pre-processing was applied to optimize the performance of the deep neural network during training:

The images were standardized to 224 x 224-pixel JPEG images of RGB colour scale. Image normalization was applied to rescale the image pixel values to a range between -1 and 1, which is the range expected by the MobileNetV3 pre-trained neural network model. Shuffling was applied to the training data to improve the model's generalization during training. Prefetching was also applied to both the training and validation datasets, to improve the training efficiency by loading the next batch of data as training of a current batch is ongoing. Figure 5.2 presents the python script ran to pre-process the training data. Image label names were inferred from the directory structure.

Image Dataset Pre-processing

```
#MobileNetV3 pre-processing
preprocess_input = tf.keras.applications.mobilenet_v3.preprocess_input
```

```
#Training data pre-processing
traindata_path = "/content/drive/MyDrive/tomato_crop_disease/train"

train_data = tf.keras.utils.image_dataset_from_directory(
    traindata_path,
    labels='inferred',
    label_mode='categorical',
    image_size=(224,224),
    batch_size=32,
    shuffle=True,
)
train_data = train_data.map(lambda x, y: (preprocess_input(x), y))
train_data = train_data.prefetch(buffer_size=tf.data.AUTOTUNE)

class_names = sorted(os.listdir(traindata_path))
print("Class Names:", class_names)
```

```
Found 16880 files belonging to 6 classes.
Class Names: ['early_blight', 'healthy', 'leaf_mold', 'mosaic_virus', 'septoria_leaf_spot', 'yellow_leaf_curl_virus']
```

Figure 5.2: Training Image Data Pre-processing

5.3.3 CNN Model Training

Supervised transfer learning was used to train the CNN model. A CNN model is comprised of the convolution layer which extracts patterns and features from the input image, pooling layer which samples down the output of the convolution layer to optimize computational requirements, and fully connected layer that performs the classification. As reviewed in 3.6.1, MobileNetV3 deep learning neural network was chosen as the base model for convolution. Specifically, MobileNetV3-Small was used, as the lightweight neural network adapted for mobile applications. The architecture of the MobileNetV3-Small neural network is illustrated in Figure 5.3.

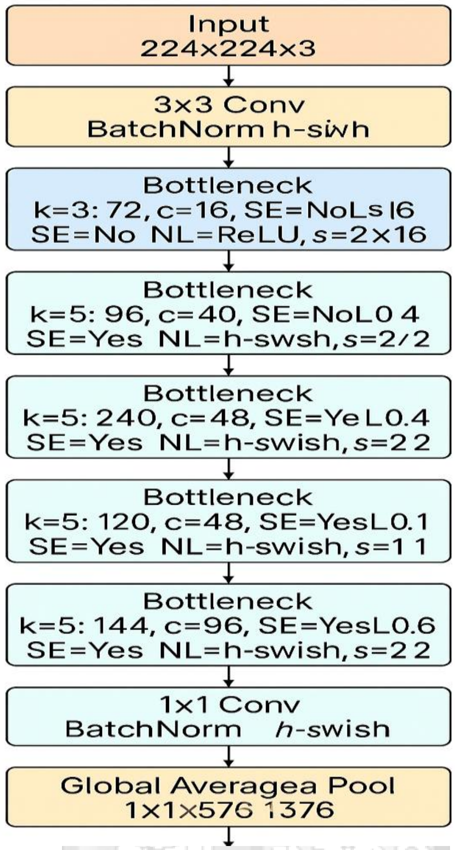


Figure 5.3: MobileNetV3-Small Architecture

Experimentation with several variations of training parameters, to come up with the combination that yields the best result, resulted in the choice of parameters shown in Table 5.1.

Table 5.1: Parameters Used for Model Training

Parameter	**Value**
Activation Function	ReLU
Optimizer	Adam
Learning Rate	0.0001
Loss Function	Categorical cross-entropy
Metrics	Accuracy
Number of Epochs	100

The number of epochs was set to 100, with an EarlyStopping callback function set to monitor the validation loss after every epoch and stop the training when the validation loss stops improving for five consecutive epochs. Subsequently the training completed in 32 epochs. The code executed to train the model is shown in Figure 5.4. The figure also illustrates the training progress, showing that both the training and validation accuracy improved over the epochs, as the training and validation loss decreased.

```
# Train the model
history = model.fit(
    train_data,
    validation_data=val_data,
    epochs=100,
    callbacks = [EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True),checkpoint_callback
    ]
)
```

```
Epoch 1/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 2441s 5s/step - accuracy: 0.4286 - loss: 1.5532 - val_accuracy: 0.8678 - val_loss: 0.4624
Epoch 2/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 63ms/step - accuracy: 0.7990 - loss: 0.5984 - val_accuracy: 0.9139 - val_loss: 0.2731
Epoch 3/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 62ms/step - accuracy: 0.8647 - loss: 0.4059 - val_accuracy: 0.9392 - val_loss: 0.2004
Epoch 4/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 62ms/step - accuracy: 0.8921 - loss: 0.3183 - val_accuracy: 0.9450 - val_loss: 0.1675
Epoch 5/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 63ms/step - accuracy: 0.9103 - loss: 0.2602 - val_accuracy: 0.9523 - val_loss: 0.1444
Epoch 6/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 62ms/step - accuracy: 0.9193 - loss: 0.2281 - val_accuracy: 0.9561 - val_loss: 0.1272
Epoch 7/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 62ms/step - accuracy: 0.9315 - loss: 0.1994 - val_accuracy: 0.9629 - val_loss: 0.1137
Epoch 8/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 62ms/step - accuracy: 0.9411 - loss: 0.1750 - val_accuracy: 0.9678 - val_loss: 0.1049
Epoch 9/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 62ms/step - accuracy: 0.9469 - loss: 0.1567 - val_accuracy: 0.9700 - val_loss: 0.0971
Epoch 10/100
528/528 ━━━━━━━━━━━━━━━━━━━━ 33s 62ms/step - accuracy: 0.9532 - loss: 0.1338 - val_accuracy: 0.9720 - val_loss: 0.0939
Epoch 11/100
```

Figure 5.4: Model Training Command and Output

The complete Python notebook used for model training is presented in Appendix D.

5.4 Mobile Application Implementation

5.4.1 Converting the CNN Model into an Android-Compatible Format

The trained deep learning model was saved as a Keras file. The image class names, together with their corresponding class indices were also saved into a JSON file. The Keras model was converted into a TensorFlow Lite model. The command in Figure 5.5 was executed to perform the conversion. To ensure that a correct prediction was possible for images the model had not been trained on, embedding distance metrics were extracted from the fully connected layer before the final softmax classifier. This enabled prediction to be done based on the distance to the centroids of the image classes, in addition to the standard prediction that utilised probabilities for the different image classes. This was also converted into a TensorFlow Lite model. The TensorFlow Lite models, together with the class indices JSON file were uploaded into the assets folder in Android studio, for deployment in the android application.

```python
#Model conversion
converter = tf.lite.TFLiteConverter.from_keras_model(model)
tflite_model = converter.convert()

with open('/content/drive/My Drive/model_checkpoints/best_model.tflite', 'wb') as f:
  f.write(tflite_model)
```

Figure 5.5: Command to Convert Keras Model to TensorFlow Lite Model

5.4.2 Building the Mobile Application

The mobile application was developed in Android Studio version 2024.2.2. The gradle build file was modified to include all the necessary dependencies. The TensorFlow Lite and JSON image class files generated in section 5.4.1 were copied into the assets folder. ResultActivity Kotlin program was created to read and parse the JSON file and implement the ML model in Android. Activity layout and mobile navigation XML files were built to implement the user interface and navigation respectively. Database classes were implemented in the models module. These were incorporated into the MainActivity Kotlin program. Other functionalities of the mobile application, including image capture, login and registration were also implemented in Kotlin code. The overall layout of the project in Android studio is illustrated in Figure 5.6.

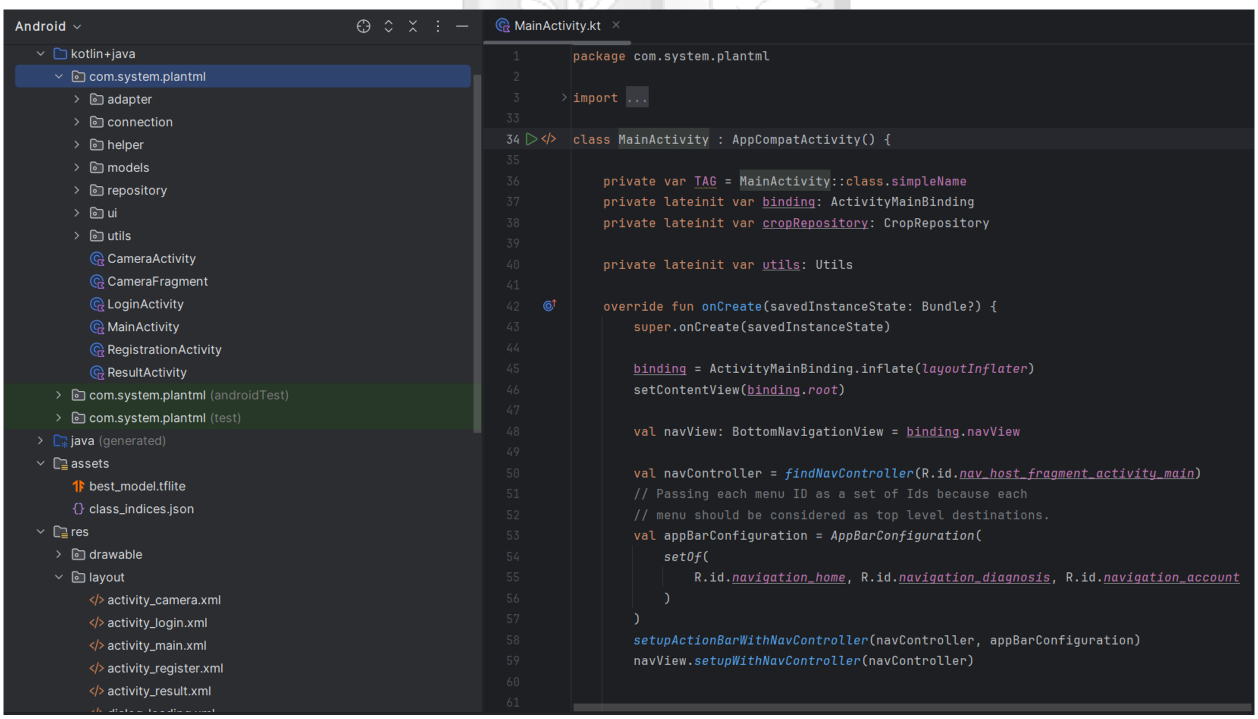


Figure 5.6: Android Mobile Application Development

The resultant mobile application APK was installed in a mobile device for testing. Screenshot samples of the system implementation are included in Appendix E.

5.4.3 Administrative Backend

The landing page on the system backend was the login page for the system administrator. Once logged in, the system administrator is presented with several menus from where they can perform various system administration activities. The backend dashboard is presented in Figure 5.7.

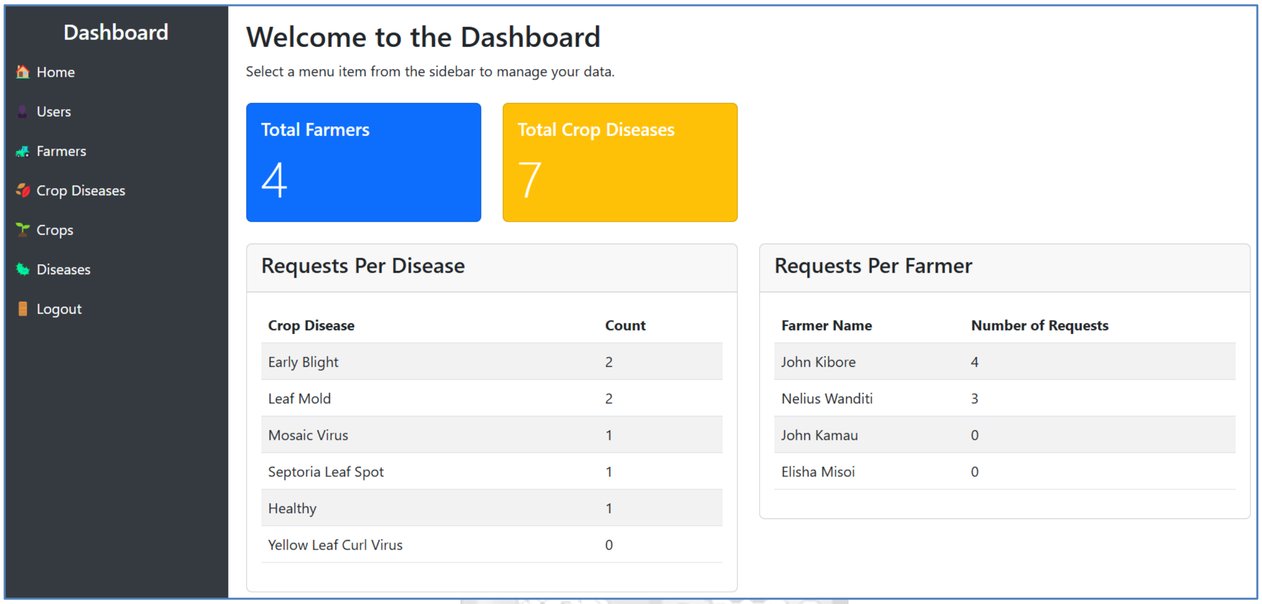


Figure 5.7 System Administration Website

5.5 System Testing

The testing carried out on the system is outlined in this section. Testing carried out included testing to validate the inference capabilities of the deep neural network, functional testing of the mobile application, and non-functional testing.

5.5.1 CNN Model Validation and Testing

The training and validation loss both decreased with subsequent epochs, meaning that the model was learning well. The training and validation loss curves also closely matched each other, indicating that the model was not overfitting. The training and validation loss curves are illustrated in Figure 5.8. The final training and validation loss values achieved were 0.0443, and 0.0651. Zero loss is the ideal value.

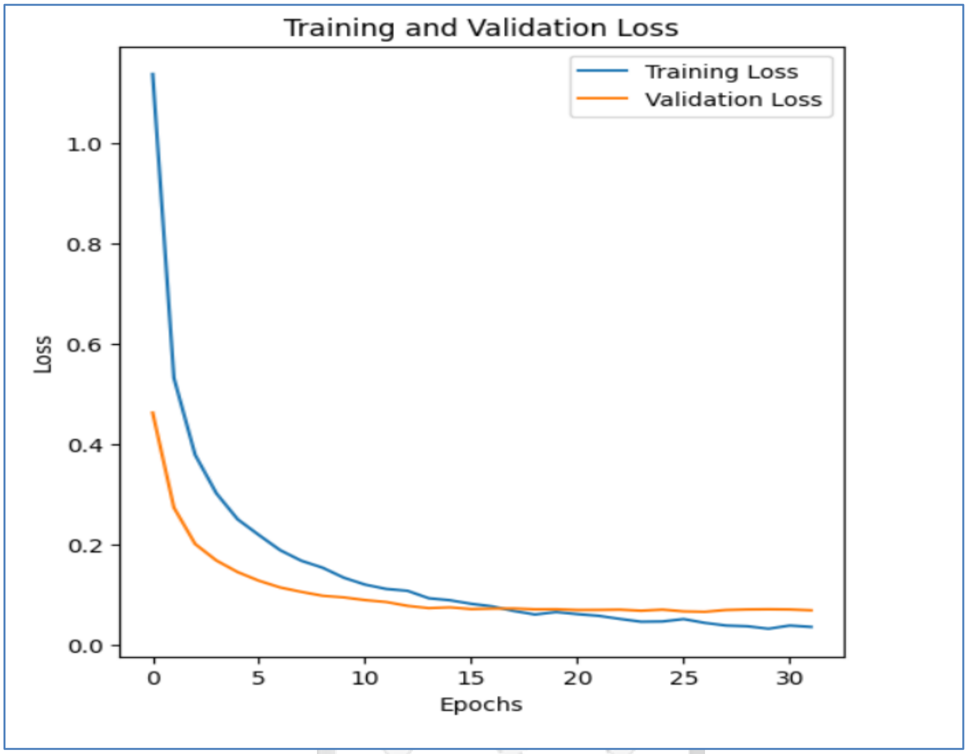


Figure 5.8: Training and Validation Loss

The training and validation accuracy improved with subsequent epochs, implying the model improved how well it classified the training data, and how well the model predicts a diseased tomato crop image based off previously unseen data. The training achieved final training and validation accuracy values of 98.49% and 98.22% respectively. This was a strong indication that the model would perform well in real-world applications. Figure 5.9 represents the training and validation accuracy curves.

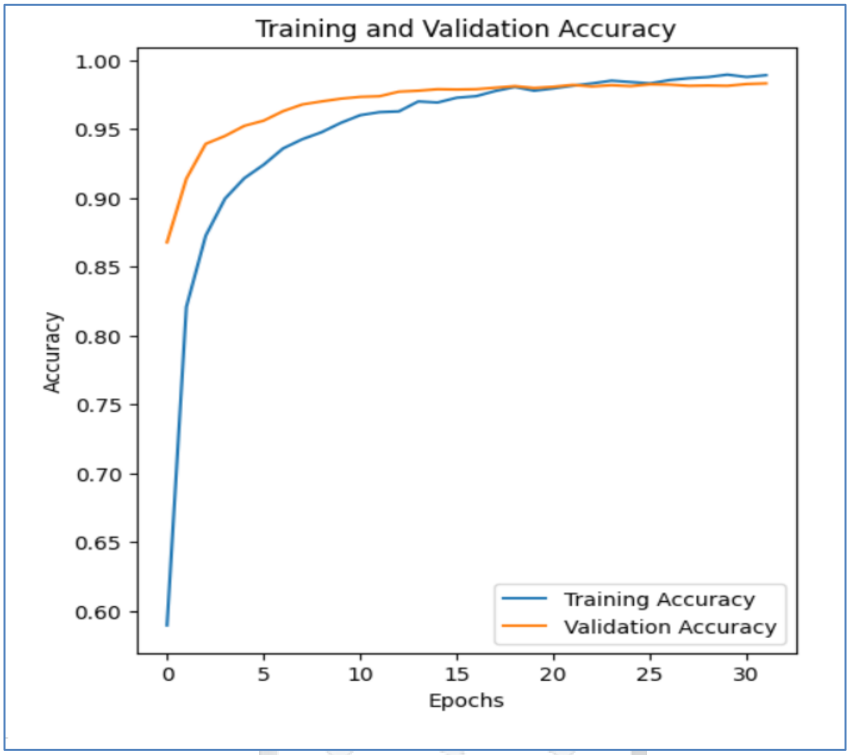


Figure 5.9: Training and Validation Accuracy

A python program was run that extracted distance metrics from the embedding model and computed centroids from random samples of the training dataset. The program randomly picked images from each of the classes in the pool of test images and performed inferences using the standard and embedding TensorFlow Lite models. The pool of test images included a class of images of other crops and objects, for which the model was not trained on. The model predictions with distances to the different classes, and confidence of the predicted class, are illustrated in Figure 5.10. The class with the closest computed distance to centroid is the predicted class.

```
================================================
Actual Class: mosaic_virus
Prediction: mosaic_virus (ID: 3)
Distance to closest centroid: 0.0107
Threshold: 0.3750
Confidence: (99.99%)

Distances to each class centroid:
  Class mosaic_virus (ID: 3): 0.0107
  Class septoria_leaf_spot (ID: 4): 1.4107
  Class leaf_mold (ID: 2): 1.4081
  Class yellow_leaf_curl_virus (ID: 5): 1.4139
  Class early_blight (ID: 0): 1.4050
  Class healthy (ID: 1): 1.4142

================================================
Actual Class: septoria_leaf_spot
Prediction: septoria_leaf_spot (ID: 4)
Distance to closest centroid: 0.0345
Threshold: 0.1028
Confidence: (96.39%)

Distances to each class centroid:
  Class mosaic_virus (ID: 3): 1.4035
  Class septoria_leaf_spot (ID: 4): 0.0345
  Class leaf_mold (ID: 2): 1.4027
  Class yellow_leaf_curl_virus (ID: 5): 1.4140
  Class early_blight (ID: 0): 1.3716
  Class healthy (ID: 1): 1.4137

================================================
Actual Class: unknown
Prediction: Unknown
Distance to closest centroid: 0.0053
Min threshold: 0.0100
Normal threshold: 0.1028
Confidence: (100.00%)

Distances to each class centroid:
  Class mosaic_virus (ID: 3): 1.4035
  Class septoria_leaf_spot (ID: 4): 0.0053
  Class leaf_mold (ID: 2): 1.4034
  Class yellow_leaf_curl_virus (ID: 5): 1.4140
  Class early_blight (ID: 0): 1.3971
  Class healthy (ID: 1): 1.4142
Reason: Distance suspiciously small
```

```
================================================
Actual Class: early_blight
Prediction: early_blight (ID: 0)
Distance to closest centroid: 0.0166
Threshold: 0.3767
Confidence: (99.89%)

Distances to each class centroid:
  Class mosaic_virus (ID: 3): 1.4082
  Class septoria_leaf_spot (ID: 4): 1.4091
  Class leaf_mold (ID: 2): 1.4077
  Class yellow_leaf_curl_virus (ID: 5): 1.4140
  Class early_blight (ID: 0): 0.0166
  Class healthy (ID: 1): 1.4142

================================================
Actual Class: unknown
Prediction: Unknown
Distance to closest centroid: 0.0577
Threshold: 0.0010
Confidence: (93.89%)

Distances to each class centroid:
  Class mosaic_virus (ID: 3): 1.4060
  Class septoria_leaf_spot (ID: 4): 1.4086
  Class leaf_mold (ID: 2): 1.3671
  Class yellow_leaf_curl_virus (ID: 5): 1.4140
  Class early_blight (ID: 0): 1.4026
  Class healthy (ID: 1): 0.0577
Reason: Distance exceeds threshold

================================================
Actual Class: healthy
Prediction: Unknown
Distance to closest centroid: 0.0010
Min threshold: 0.0100
Normal threshold: 0.0010
Confidence: (99.90%)

Distances to each class centroid:
  Class mosaic_virus (ID: 3): 1.4076
  Class septoria_leaf_spot (ID: 4): 1.4094
  Class leaf_mold (ID: 2): 1.4081
  Class yellow_leaf_curl_virus (ID: 5): 1.4140
  Class early_blight (ID: 0): 1.4048
  Class healthy (ID: 1): 0.0010
Reason: Distance suspiciously small
```

Figure 5.10: Model Predictions with Randomly Selected Test Images

5.5.2 Functional Testing of The Mobile Application

Functional tests were done based on use cases to determine the success of the implemented system. Table 5.2, Table 5.3, Table 5.4 and Table 5.5 present the major test cases and their outcome.

Table 5.2: Upload Image Test Case

Test Case Name:	Upload Crop Image
Test Description:	i. The farmer takes a picture of the crop or uploads a photo from their image gallery. ii. The mobile application allows the farmer to preview the picture taken, with an option to retake or save the picture. Also, if the picture is not clear, the mobile application prompts the farmer to retake the picture.
Utilised Use Case(s):	Capture Crop Image/Upload Image from Gallery
Expected Outcome:	The image is successfully uploaded into the mobile application, ready for inference.
Test Pass/Fail:	Pass

Table 5.3: Classify Disease Test Case

Test Case Name:	Classify Disease
Test Description:	i. The embedded machine learning model analyses the uploaded image, detects potential disease based on the pre-learned patterns and generates a result. ii. The mobile application displays the result of the identified disease to the farmer.
Utilised Use Case(s):	Submit Image for Analysis; Get Diagnosis and Treatment Recommendation
Expected Outcome:	The farmer gets the disease diagnosis, along with treatment recommendations and preventive measures. If no treatment is recommended, the farmer is advised accordingly.
Test Pass/Fail:	Pass

Table 5.4: Save Diagnostic Report Test Case

Test Case Name:	Save Diagnostic Report
Test Description:	i. The mobile application prompts the farmer to save the diagnostics report. ii. The farmer is prompted to login into their account. If the farmer does not have an account, they are prompted to create a new account. iii. The farmer stores the uploaded image, along with the diagnosis results and recommendations in a database.
Utilised Use Case(s):	Get Diagnosis and Treatment Recommendation; Register and Login
Expected Outcome:	The farmer is logged into their user account. The crop image is successfully uploaded into the database, alongside the diagnosis results and disease treatment and prevention recommendations.
Test Pass/Fail:	Pass

Table 5.5: Track Historical Diagnoses Test Case

Test Case Name:	Track Historical Diagnoses Test Case
Test Description:	i. The farmer navigates to the "Track Crop Health" section of the mobile application. ii. The mobile application displays a list of previous crop images uploaded and the disease diagnoses. iii. The farmer can select a specific item and view, in detail, information about the disease diagnosis, treatment and prevention recommendations.
Utilised Use Case(s):	Monitor Crop Health
Expected Outcome:	The farmer can access information about past inferences they have done regarding their crops.
Test Pass/Fail:	Pass

The disease prediction feature was tested by uploading images of diseased tomato crop, and observing the predictions returned. Based on the inference made, the mobile application returned to the farmer the name of the diagnosis, symptoms and disease management recommendations. This is illustrated in Figure 5.11.

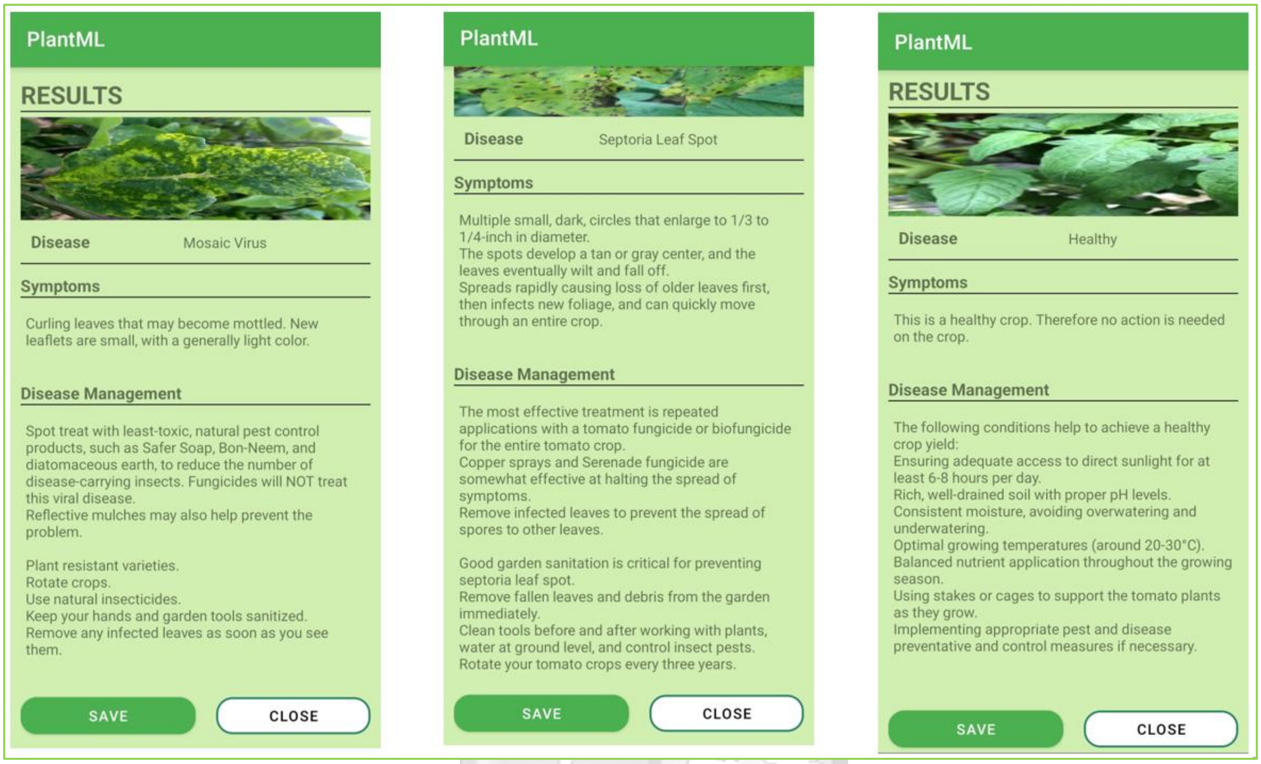


Figure 5.11: Disease Prediction in The Mobile Application

5.5.3 Non-functional Testing

Non-functional testing comprised usability and compatibility tests. The system presented the farmer with the option to either take a new photo using the phone's camera capabilities or upload a picture from the phone's photo gallery. The inference was observed to be fast, with inference times of less than 500 milliseconds. A farmer who opted to save the inference outcome was prompted to login into their account, and in case they did not have a registered account, they were prompted to register one. The different functionalities are illustrated in Appendix E.

The mobile application was tested against android versions beginning with Android API 16. The outcome is illustrated in Table 5.6.

Table 5.6: Android Version Compatibility Test

Android API Level	**Android Version**	**Compatible**
Level 16	Android 4.1	YES
Level 17	Android 4.2	YES
Level 18	Android 4.3	YES
Level 19	Android 4.4	YES
Level 21	Android 5.0	YES
Level 22	Android 5.1	YES
Level 23	Android 6	YES
Level 24	Android 7.0	YES
Level 25	Android 7.1	YES
Level 26	Android 8.0	YES
Level 27	Android 8.1	YES
Level 28	Android 9	YES
Level 29	Android 10	YES
Level 30	Android 11	YES
Level 31	Android 12	YES
Level 33	Android 13	YES

Chapter 6: Discussion Of Results

6.1 Introduction

This chapter discusses the research findings and reviews the solution, in relation to the research objectives.

6.2 Review of the Research Outcome in Relation to the Research Objectives

The primary objective of this dissertation was to develop a crop disease prediction system that would help farmers improve their crop yields by providing disease treatment and prevention recommendations.

6.2.1 Existing Application of Mobile Technologies in Agriculture

The research began by reviewing the contributing factors in reduced crop yields among smallholder farmers. These factors include social economic factors such as low education levels, low income levels, unpredictable climatic conditions, poor land and crop management practices that lead to wastage of inputs, deterioration in soil fertility and poor crop yields, and difficulty in accessing agricultural advisory services due to being in rural setups and agricultural extension being a training and visitation system primarily provided by the government. The research subsequently identified existing initiatives that are aimed at improving food production among smallholder farmers. The study concentrated three tools for provision of agricultural extension and advisory services, that are provided by three different kinds of sector players. The first tool, PlantwisePlus, is provided by a non-governmental research institution, that has a lot of research data and trained experts on crop pests and diseases. The second tool, Digicult is provided by a private small and medium enterprise, providing good design of the mobile part of the system, without necessarily having free access to research data and expertise. The third, KAOP App, is a tool by a governmental agricultural research institution, that is used for agronomic advisory based on whether patterns, and that can be scaled to include crop pests and disease management advisory. These individual strengths and weaknesses of the different kinds of agricultural digital technology solutions helped to identify the gaps that informed the design and implementation of the tomato disease detection system.

6.2.2 Digital Technologies That Can Be Applied to Improve Agricultural Productivity

Precision farming technologies were reviewed as the methods for primary data collection. How the technologies are utilized, their advantages, and the considerations for their success were explored.

IoT was studied as the most predominantly used in-field monitoring technology. UAVs were studied as the most used remote data collection technology for precision farming. They offer cost and ease of use advantages and operation in hard-to-reach areas, compared to other aerial methods such as satellites and manned aircraft. Their ability to carry payloads such as sensors and photography equipment makes them flexible for use in a wide range of tasks including monitoring the growth of vegetation, weed mapping and management, and can be integrated with smart sensors. GIS was studied as the technology that is used to create cartographic database, enabling mapping of location-specific information. Cloud computing was studied as the engine capable of providing a complete ecosystem for provision of the storage, processing and analysis of data collected using the precision farming technologies. Machine learning was reviewed as a means to process and analyse primary data as well as data from research institutions, to derive useful intelligence and insights that are useful in aiding the smallholder farmers towards improving their agricultural productivity.

6.2.3 Designing and Developing a Mobile Application-Based Image Recognition System

Subsequently, the research sought to develop a system that would apply the latest technological advancements to process data not readily available to smallholder farmers, to provide relevant insights and make this available to smallholder farmers. An analysis of data collected from a sample of the target population was carried out to determine the target users' need for this system. The outcome was that seventy percent of the farmers embraced the use of a mobile application-based system to manage and prevent crop pests and diseases. The system's functional and non-functional requirements were also documented. A system design was carried out, and the prototyping methodology applied in developing the system, which incorporated a trained deep learning model for image pattern identification in crop disease data prediction, and a mobile application for interacting with the end user to provide input images to the deep learning model for prediction and provide disease treatment and prevention recommendations.

6.2.4 Testing The Functionality of the Developed System

The system was tested against the requirements set out to ascertain that it met the intended purpose. Validation of the trained deep learning model was carried out against data that it had not previously been exposed to. This was done to ensure that the model would perform inference to acceptable confidence levels. The validation attained an accuracy of 98.22%. The model was further tested on a set of images that was neither part of the dataset it was trained on, nor the dataset it was validated against. To ensure that the model would also correctly predict as unknown, images of objects for which it was never trained on, distance metrics were applied to each prediction. This resulted in accurate predictions for tomato diseases, and prediction as unknown, of several images that were not of any of the tomato diseases the model was trained on. The model was embedded into an android application, and provided prediction based on images taken from the mobile device's camera or uploaded from images the user already had in their photo gallery. It was observed that the inferences done returned correct disease classifications. The mobile application therefore achieved the research objective to develop a system that can be used to alleviate the agricultural productivity challenge of managing crop pests and disease. Detailed functional and non-functional tests carried out on the system validated that the system performed as was intentioned.

6.3 Merits of the Developed System

The most significant advantage of the developed system over existing systems is the ability to provide agricultural insights based on a machine learning model. Without the need for any other human input, the farmer can get a disease diagnosis and perform corrective and preventive actions. By utilising large sets of data that the smallholder farmer would otherwise not have access to, the system alleviates the social economic and access factors that inhibit high productivity in small-scale farming setups. The developed system can help farmers to reduce farming input costs such as cost of pesticides and empower them to reduce impact on the environment through optimum use of pesticides. Use of the system to help prevent tomato diseases has the potential to improve crop yields, positively impacting the economic status of the tomato smallholder farmers.

6.4 Shortfalls of the System

One of the biggest shortfalls of the system is that the classifier would sometimes fail to correctly classify images from objects it was never trained on. Since for every set of data this image is trained on, there exists thousands of image classes it has never been exposed to, there is a probability that the model identifies an image pattern that is similar to the pattern learned from one of the training datasets, leading to a wrongful classification. The primary prediction method for the model is a probability-based method, with a sum of probabilities of one. In this method, an inference will be made of a class among the classes the model is trained on, regardless of what the image provided to the model is. Several techniques were applied to mitigate this challenge, including introducing a confidence threshold that would only return a certain class as the predicted class if the probability for that class exceeded a certain threshold, negative sampling that introduced a dataset of image examples consisting of images of objects the model had not been trained on, as well as applying an entropy-based detection. Using embedding distance metrics was the method that was found to be most effective, since it introduced a means to identify out-of-distribution image samples. With this method, the image classifier was able to classify images of unknown objects, by flagging predictions with suspiciously small distance to the centroid of the closest class. However, the system may still predict as unknown an image of an object of a known class, if the calculated minimum threshold for the known class is too low. This was the case with healthy tomato crop class, for which the minimum calculated threshold was 0.01, and the distance to the class's centroid was also 0.01.

Chapter 7: Conclusions and Recommendations

7.1 Conclusions

A review of existing literature identified the factors that influence agricultural output among smallholder farmers. These include social economic factors such as low education levels, low-income levels and difficulty in accessing agricultural services due to smallholder farmers being primarily located in rural areas. Poor land and crop management practices contribute to the deterioration of soil fertility and lead to poor crop yields per acreage. The farmers have inadequate access to agricultural extension services, the primary source for agricultural extension being a government-led training and visitation system.

The literature review also identified gaps in existing solutions to these agricultural productivity challenges among smallholder farmers. Existing mobile applications were identified to not have a feedback loop whereby the farmer can provide real input from their farm, and get localised insights based on this input. The mobile applications were also found to only augment agricultural extension services and not form part and parcel of it. It was found that most research institutions work with the farmers to collect data and implement solutions, as well as with extension workers, training them to offer best practice advice to farmers, but physical engagement remained to be the primary means of interaction with the farmers and extension workers.

The research sought to address these gaps by developing a system that could utilise such available data to enhance information access and agricultural extension. The dissertation therefore presented a detailed approach to develop a mobile application-based system that integrates deep learning neural networks to classify crop diseases and provide insights to the farmer based on crop disease inputs from the farmers' own farms. The resultant mobile application performed satisfactorily and was able to predict tomato diseases based on an image provided by a user and provide insights and recommendations.

7.2 Recommendations

Based on the outcome for this research, the researcher recommends that institutions which advance agricultural sustainability through research should embrace mobile technologies more. This will enable a lot of research data that is not widely available to smallholder farmers to be available to them. There is also a need for better utilization of data from precision farming tools in the agricultural research institutions and large farms, to enable local solutions have richer datasets for training machine learning models.

7.3 Future Work

Further work can be carried out on the developed system to enhance its functionality as well as expand its application. The following areas of enhancement are recommended:

i. The developed system concentrated on a few tomato disease classes, based on the data that was readily available. Further development work can therefore be carried out to incorporate more tomato diseases.
ii. The system can also be scaled to include other crops that are popular with smallholder farmers. This would require datasets of images of diseased crops for crop considered.
iii. The system can be integrated with research initiatives, and agricultural extension services to intensify their outreach to the smallholder farmers as a target audience.
iv. Further research work should be carried out to improve the capability of image classifiers to perform correct predictions for out of distribution image inputs.

References

Agriculture Sector Development Support Programme. (2019). *National Baseline Survey Report.* Nairobi, Kenya: Ministry of Agriculture, Livestock, Fisheries and Cooperatives.

Akshatha, A., & Dhulipala, R. (2024). *Enhancing Kenya Agricultural Observatory Platform (KAOP) by integrating AICCRA's AgDataHub.* Accelerating Impacts of CGIAR Climate Research for Africa.

Alliance for a Green Revolution in Africa. (2017). *Africa Agriculture Status Report: The Business of Smallholder Agriculture in Sub-Saharan Africa.* Nairobi, Kenya: Alliance for a Green Revolution in Africa (AGRA).

Alowolodu, O., & Gabriel, A. (2021). Cloud-based System for Agro-Extension Services towards Combating Food Insecurity in Africa. *Communications on Applied Electronics, 7,* 26-33. doi:10.5120/cae2021652882

Amirgaliyev, Y., Sokolov, D., Zaitseva, E., Levashenko, V., Symagulov, A., Kuchin, Y., . . . Mukhamediev, R. I. (2023). Coverage Path Planning Optimization of Heterogeneous UAVs Group for Precision Agriculture. *IEEE Access*, *11*, 5789-5803. doi:10.1109/ACCESS.2023.3235207

Anang, B. T., & Asante, B. O. (2020). Farm household access to agricultural services in northern Ghana. *Heliyon, 6*(11). doi:doi.org/10.1016/j.heliyon.2020.e05517

Anik, A. R., Rahman, S., & Sarker, J. R. (2020). Five Decades of Productivity and Efficiency Changes in World Agriculture (1969–2013). *Agriculture*, *10*, 1-2. doi:10.3390/agriculture10060200

Awuor, F. M., & Otanga, S. (2019). Farmer Centered Large Scale e-Surveillance and Control of Crop Pests in Kenya. *Journal of Agricultural Informatics, 1*. doi:10.17700/jai.2019.1.1.500

Ayamga, M., Tekinerdogan, B., & Kassahun, A. (2021). Exploring the Challenges Posed by Regulations for the Use of Drones in Agriculture in the African Context. *Digital Agriculture for Sustainable Food Systems: Implications for Land-Resource Use and Management*, *10*(2). doi:10.3390/land10020164

Bartol, T. (2023). Smallholders and small-scale agriculture: Mapping and visualization of knowledge domains and research trends. *Cogent Social Sciences*, *9*(1). doi:10.1080/23311886.2022.2161778

Birch, I. (2018). *Agricultural productivity in Kenya: barriers and opportunities.* Knowledge, evidence and learning for development (K4D).

Bonilla, J. D., Coombes, A., Romney, D., & Winters, P. C. (2024). Changing the logic in agricultural extension: evidence from a demand-driven extension programme in Kenya. *Journal of Development Effectiveness, 16*(1), 118-141. doi:doi: 10.1080/19439342.2023.2181848

Cairns, J. E., Chamberlin, J., Rutsaert, P., Voss, R. C., Ndhlela, T., & Magorokosho, C. (2021). Challenges for sustainable maize production of smallholder farmers in sub-Saharan Africa. *Journal of Cereal Science, 101*. doi:10.1016/j.jcs.2021.103274

Comer, D. (2021). *The Cloud Computing Book : The Future of Computing Explained.* Boca Raton, Florida: CRC Press.

Dennis, A., Wixom, B., & Tegarden, D. (2020). *Systems Analysis and Design: An Object-Oriented Approach with UML* (6th ed.). Cambridge, MA: John Wiley & Sons, Inc.

Duerr, O., & Sick, B. (2020). *Probabilistic Deep Learning: With Python, Keras and TensorFlow Probability.* Manning. Retrieved from http://ieeexplore.ieee.org/document/10280401

Dunaieva, I., Mirschel, W., Popovych, V., Pashtetsky, V., Golovastova, E., Vecherkov, V., . . . Topaj, A. (2019). GIS Services for Agriculture Monitoring and Forecasting: Development Concept. *EInternational Scientific Conference Energy Management of Municipal Facilities and Sustainable Energy Technologies EMMFT 2018* (pp. 236-246). Cham: Springer International Publishing. doi:10.1007/978-3-030-19868-8_24

Elijah, O., Abdul Rahman, T., Orikumhi, I., Leow, C. Y., & Hindia, M. (2018). An Overview of Internet of Things (IoT) and Data Analytics in Agriculture: Benefits and Challenges. *IEEE Internet of Things Journal, 5*(5). doi:10.1109/JIOT.2018.2844296

Elsayed Abd Elaziz, M., Dahou, A., Alsaleh, N., Elsheikh, A., Saba, A., & Ahmadein, M. (2021, 10 22). Boosting COVID-19 Image Classification Using MobileNetV3 and Aquila Optimizer Algorithm. *Entropy, 23*. doi:10.3390/e23111383

FAO. (2023). *Pilot Digital Villages Initiative in Africa – Findings of country assessments and recommendations for effective implementation.* Accra: FAO. doi:doi.org/10.4060/cc6131en

Gatheru, M., Njarui, D., Gichangi, E., Ndubi, J., Murage, A., & Gichangi, A. (2021). Status and Factors Influencing Access of Extension and Advisory Services on Forage Production in Kenya. *Asian Journal of Agricultural Extension, Economics & Sociology*, 99-113. doi:10.9734/ajaees/2021/v39i330550

Hanna, K. T. (2023, 04 15). JPEG (Joint Photographic Experts Group). Retrieved from https://www.techtarget.com/whatis/definition/JPEG-Joint-Photographic-Experts-Group#:~:text=JPEG%20(pronounced%20JAY%2Dpeg),browsers%20and%20image%20editing%20software.

Haruna, I., Musah, B. A., & Kwame, P. N. (2018). Does the Use of Mobile Phones by Smallholder Maize Farmers Affect Productivity in Ghana? *Journal of African Business, 19*(3), 302-322. doi:10.1080/15228916.2017.1416215

Harvey, C. A., Saborio-Rodríguez, M., Martinez-Rodríguez, M. R., Chain-Guadarrama, A., Vignola, R., & Alpizar, F. (2018). Climate change impacts and adaptation among smallholder farmers in Central America. *Agriculture & Food Security, 7*(1). doi:10.1186/s40066-018-0209-x

Hlatshwayo, S. I., Ngidi, M. S., Ojo, T. O., Modi, A. T., Mabhaudhi, T., & Slotow, R. (2023). The determinants of crop productivity and its effect on food and nutrition security in rural communities of South Africa. *Frontiers in Sustainable Food Systems, 3*. doi:10.3389/fsufs.2023.1091333

Howe, S. (2023). 6 Barriers to Drone Adoption in Precision Agriculture and How to Overcome Them. *Forestry & Agriculture*. Retrieved from https://www.commercialuavnews.com/forestry/6-barriers-to-drone-adoption-in-precision-agriculture-and-how-to-overcome-them

Insider Intelligence. (2021, 01 12). *Drone technology uses and applications for commercial, industrial and military drones in 2021 and the future.* Retrieved from Business Insider: https://www.businessinsider.com/drone-technology-uses-applications?r=US&IR=T

Jouzi, Z., Azadi, H., Taheri, F., Zarafshani, K., Gebrehiwot, K., Van Passel, S., & Lebailly, P. (2017). Organic Farming and Small-Scale Farmers: Main Opportunities and Challenges. *Ecological Economics*, *132*, 144-154. doi:10.1016/j.ecolecon.2016.10.016

KALRO. (2016). *Service Charter.* Nairobi, Kenya: Kenya Agricultural and Livestock Research Organization. Retrieved from https://www.kalro.org/service-charter/

Klauser, F., & Pauschinger, D. (2021). Entrepreneurs of the air: Sprayer drones as mediators of volumetric agriculture. *Journal of Rural Studies, 84*, 55-62. doi:10.1016/j.jrurstud.2021.02.016

Krisnawijaya, N. N., Tekinerdogan, B., Catal, C., & van der Tol, R. (2023). Multi-Criteria decision analysis approach for selecting feasible data analytics platforms for precision farming. *Computers and Electronics in Agriculture, 209*. doi:doi.org/10.1016/j.compag.2023.107869

Kumar, A., & Chandrasekaran, K. (2019). Status of Mobile Agricultural Apps in the Global Mobile Ecosystem. *International Journal of Education and Development using Information and Communication Technology, 15*(3), 63-74.

Lee, S., & Yun, C. M. (2023). A deep learning model for predicting risks of crop pests and diseases from sequential environmental data. *Plant Methods*, *19*(1). doi:10.1186/s13007-023-01122-x

Liang, J., & Jin, L. (2020). Multi-perspective modeling of computer sales system Based on Unified Modeling Language. *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*, (pp. 109-113). doi:10.1109/ITOEC49072.2020.9141934

Mathenge, M., Sonneveld, B. G., & Broerse, J. E. (2022). Application of GIS in Agriculture in Promoting Evidence-Informed Decision Making for Improving Agriculture Sustainability: A Systematic Review. *Sustainability, 14*(16). doi:10.3390/su14169974

Mell, P., & Grance, T. (2011). The NIST Definition of Cloud Computing. *Recommendations of the National Institute of Standards and Technology: Special Publication 800-145*.

Mendes, J., Pinho, T. M., Neves dos Santos, F., Sousa, J. J., Peres, E., Boaventura-Cunha, J., . . . Morais, R. (2020). Smartphone Applications Targeting Precision Agriculture Practices—A Systematic Review. *Agronomy, 10*(6). doi:10.3390/agronomy10060855

Migiro, L., & Otieno, W. (2020). The Role of Plantwise in Improving Detection and Action on Pest Situations. In S. Niassy, S. Ekesi, L. Migiro, & W. Otieno, *Sustainable Management of Invasive Pests in Africa* (pp. 245-251). Springer International Publishing. doi:10.1007/978-3-030-41083-4_19

Ministry of Agriculture, Livestock, Fisheries and Cooperatives. (2022). *Kenya Agricultural Sector Extension Policy.* Nairobi, Kenya: Ministry of Agriculture, Livestock, Fisheries and Cooperatives.

Mwita, E. M., Mburu, J., Elizaphan, R., Oburu, J., Okeyo, M., & S., K. (2020). Impact of ICT Based Extension Services on Dairy Production and Household Welfare: The Case of iCow Service in Kenya. *Journal of Agricultural Science*, 141. doi:10.5539/jas.v12n3p141

Naik, P. (2023). *Conceptualizing Python in Google COLAB: Hands-on Practical Sessions.* Shashwat Publication.

Naing, L., Nordin, R. B., Abdul Rahman, H., & Naing, Y. T. (2022). Sample size calculation for prevalence studies using Scalex and ScalaR calculators. *BMC Medical Research Methodology, 22*(1). doi:10.1186/s12874-022-01694-7

Norhashim, N., Kamal, N. L., Sahwee, Z., Shah, S. A., Yusri, H. S., & Fadzilah, M. N. (2023). Dynamic Evaluation of Electric Propulsion System Performance of Unmanned Aerial Vehicle (UAV). *TENCON 2023 - 2023 IEEE Region 10 Conference (TENCON)* (pp. 1-6). Chiang Mai, Thailand: IEEE. doi:10.1109/TENCON58879.2023.10322550

Oppenheim, D., & Shani, G. (2017). Potato Disease Classification Using Convolution Neural Networks. *Advances in Animal Biosciences, 8*, 244-249. doi:10.1017/S2040470017001376

Panwar, G. S. (2023, 08 29). *Best Machine Learning as a Service Platforms (MLaaS) That You Want to Check as a Data Scientist.* Retrieved from Neptune.ai: https://neptune.ai/blog/best-machine-learning-as-a-service-platforms-mlaas

Phiri, A., Chipeta, G., & Chawinga, W. (2019). Information needs and barriers of rural smallholder farmers in developing countries: A case study of rural smallholder farmers in Malawi. *Information Development, 35*(3), 421-434. Retrieved from https://api.semanticscholar.org/CorpusID:65081429

Qingqing. (2020). *How drones help improve agricultural productivity in poverty-stricken areas.* Retrieved from CGTN News: https://news.cgtn.com/news/2020-10-02/Agricultural-productivity-flies-high-with-drones-in-poverty-hit-areas-UgaQ3MXWhO/index.html

Sambasivam, G., & Opiyo, G. D. (2020). A predictive machine learning application in agriculture: Cassava disease detection and classification with imbalanced dataset using convolutional neural networks. *Egyptian Informatics Journal, 22*(1), 27-34. doi:10.1016/j.eij.2020.02.007

Sharma, A., Jain, A., Gupta, P., & Chowdary, V. (2020). Machine Learning Applications for Precision Agriculture: A Comprehensive Review. doi:10.1109/ACCESS.2020.3048415

Sharma, K., Gupta, S., Sharma, A., & Le, D. (2023). *Evolving Networking Technologies: Developments and Future Directions.* Wiley.

Sheikh, H., Prins, C., & Schrijvers, E. (2023). Artificial Intelligence: Definition and Background. In H. Sheikh, C. Prins, & E. Schrijvers, *Mission AI: The New System Technology* (pp. 15-41). Cham, Switzerland: Springer International Publishing. doi:10.1007/978-3-031-21448-6_2

Singh, N., Sharma, A. K., Sarkar, I., Prabhu, S., & Chadaga, K. (2024). IoT-based greenhouse technologies for enhanced crop production: a comprehensive study of monitoring, control, and communication techniques. *Systems Science & Control Engineering, 12*(1). doi:doi: 10.1080/21642583.2024.2306825

Tashayo, B., Honarbakhsh, A., Azma, A., & Akbari, M. (2020). Combined Fuzzy AHP–GIS for Agricultural Land Suitability Modeling for a Watershed in Southern Iran. *Environmental Management*, *66*(3), 364-376. doi:10.1007/s00267-020-01310-8

Tatis Diaz, R., Pinto Osorio, D., Medina Hernández, E., Moreno Pallares, M., Canales, F. A., Corrales Paternina, A., & Echeverría-González, A. (2022). Socioeconomic determinants that influence the agricultural practices of small farm families in northern Colombia. *Journal of the Saudi Society of Agricultural Sciences, 21*(7), 440-451. doi:10.1016/j.jssas.2021.12.001

Thirisha, R., Sugumar, D., Sugitha, K., J, A. S., V, D., Jose, A. V., & P, T. T. (2023). Precision Agriculture: IoT Based System for Real-Time Monitoring of Paddy Growth. *2023 International Conference on Sustainable Emerging Innovations in Engineering and Technology (ICSEIET)* (pp. 247-251). Ghaziabad, India: IEEE. doi:10.1109/ICSEIET58677.2023.10303483

USGS. (2023). *What is a geographic information system (GIS)?* Retrieved 04 12, 2024, from https://www.usgs.gov/faqs/what-geographic-information-system-gis#:~:text=A%20Geographic%20Information%20System%20(GIS)%20is%20a%20computer%20system%20that,Where%20are%20USGS%20streamgages%20located%3F

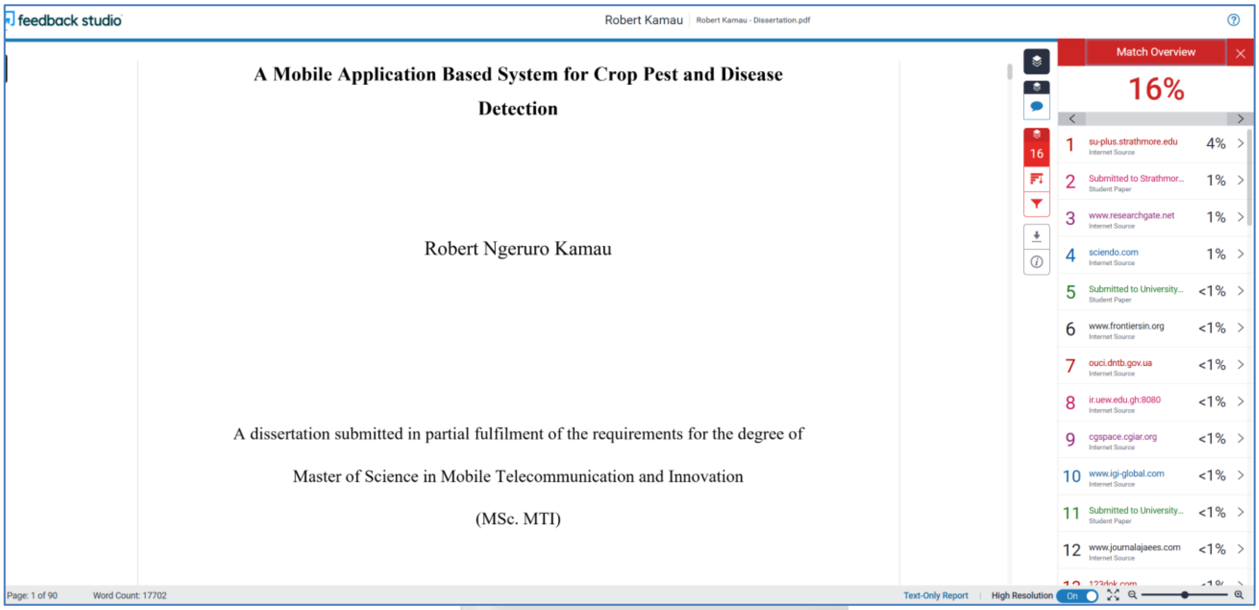
Vasisht, D., Kapetanovic, Z., Won, J., Jin, X., Chandra, R., Sinha, S., . . . Stratman, S. (2017). FarmBeats: An IoT Platform for Data-Driven Agriculture. *14th USENIX Symposium on Networked Systems Design and Implementation* (pp. 515-529). Boston, MA: USENIX Association.

Walia, M. (2021). Basics of Crop Management. *Extension*. Retrieved from https://extension.unr.edu/publication.aspx?PubID=4103

Zhuhadar, L. P., & Lytras, M. (2023). The Application of AutoML Techniques in Diabetes Diagnosis: Current Approaches, Performance, and Future Directions. *Sustainability, 15*. doi:10.3390/su151813484

Appendices

Appendix A: Originality Report



Appendix B: Ethical Clearance Letter

Strathmore
UNIVERSITY

5th February 2025

Mr Kamau Robert,
robert.kamau@strathmore.edu

Dear Mr Kamau,

<u>RE: A Mobile Based System for Crop Pest and Disease Management</u>

This is to inform you that SU-ISERC has reviewed and **approved** your above SU-masters proposal. Your application reference number is **SU-ISERC2533/24.** The approval period is from **5th February 2025 to 4th February 2026.**

This approval is subject to compliance with the following requirements:

i. Only approved documents including (informed consents, study instruments, MTA) will be used.
ii. All changes including (amendments, deviations, and violations) are submitted for review and approval by SU-ISERC.
iii. Death and life-threatening problems and serious adverse events or unexpected adverse events whether related or unrelated to the study must be reported to SU-ISERC within 72 hours of notification.
iv. Any changes anticipated or otherwise that may increase the risks or affected safety or welfare of study participants and others or affect the integrity of the research must be reported to SU-ISERC within 72 hours.
v. Clearance for the export of biological specimens must be obtained from relevant institutions.
vi. Submission of a request for renewal of approval at least 60 days prior to the expiry of the approval period. Attach a comprehensive progress report to support the renewal.
vii. Submission of an executive summary report within 90 days of completion of the study to SU-ISERC.

Before commencing your study, you will be expected to obtain a research license from National Commission for Science, Technology, and Innovation (NACOSTI) https://research-portal.nacosti.go.ke/ and obtain other clearances needed.

Yours sincerely,

Mr Ambrose Rachier,
Chairperson; SU-ISERC

Ole Sangale Rd, Madaraka Estate. PO Box 59857-00200, Nairobi, Kenya. Tel +254 (0)703 034000
Email admissions@strathmore.edu www.strathmore.edu

Appendix C: Questionnaire to Farmers

Dear Respondent,

I am a Masters in Mobile Telecommunication and Innovation student at Strathmore University, conducting research on how mobile technology and machine learning can be used to improve the management of crop pests and diseases. You have been selected to form part of this study, and your response is highly appreciated. Kindly fill in the questionnaire below. The information you give will be used purely for academic purposes.

1. What is your gender?

   Mark only one oval.

   - Male
   - Female

2. What is you age group?

   Mark only one oval.

   - 16-25
   - 26-35
   - 36-45
   - 45-55
   - 56 or older

3. Do you own a mobile phone?

Mark only one oval.

- Yes, I own and Android phone
- Yes, I own an iPhone
- Yes, I own a simple phone, without android or apple operating system
- No, I do not own a mobile phone

4. Do you access the Internet using your phone?

Mark only one oval.

- Yes
- No

5. What is your highest level of education

Mark only one oval.

- Primary
- Secondary
- Tertiary (University Degree or College Diploma)

6. What would you say is the approximate size of your farm (in acres)?

____

7. Which of the following are the ways in which you mostly use your mobile phone?

Tick all that apply.

- [ ] Keeping in touch with family members and friends
- [ ] Accessing social media e.g. WhatsApp, Facebook and Twitter
- [ ] Conducting business for my farm
- [ ] Looking for buyers and sellers of my farm produce
- [ ] Finding information about how I can improve my farming practices and output

8. Do you think the mobile phone has been useful in enabling you improve your crop production?

Mark only one oval.

- ◯ Yes
- ◯ No
- ◯ Maybe
- ◯ I am not sure

9. If you answered yes, what do you think are ways in which the mobile phone has helped you in improving your crop production?

10. Would you consider using a mobile application that helps you identify and eliminate or prevent tomato crop diseases?

Mark only one oval.

- ( ) Yes, I would consider using a mobile application
- ( ) Yes, I would consider using a WhatsApp-based channel
- ( ) Yes, I would consider using an sms-based information service
- ( ) No, I would not consider using any application for crop disease prevention.

Appendix D: Python Code Used for ML Model Development

<u>Image Data Pre-processing:</u>

```
#MobileNetV3 pre-processing
preprocess_input = tf.keras.applications.mobilenet_v3.preprocess_input
```

```
#Training data pre-processing
traindata_path = "/content/drive/MyDrive/tomato_crop_disease/train"

train_data = tf.keras.utils.image_dataset_from_directory(
    traindata_path,
    labels='inferred',
    label_mode='categorical',
    image_size=(224,224),
    batch_size=32,
    shuffle=True,
)
train_data = train_data.map(lambda x, y: (preprocess_input(x), y))
train_data = train_data.prefetch(buffer_size=tf.data.AUTOTUNE)

class_names = sorted(os.listdir(traindata_path))
print("Class Names:", class_names)
```

```
#Validation data pre-processing
valdata_path = "/content/drive/MyDrive/tomato_crop_disease/valid"

val_data = tf.keras.utils.image_dataset_from_directory(
    valdata_path,
    labels='inferred',
    label_mode='categorical',
    image_size=(224,224),
    batch_size=32,
    shuffle=False,
)
val_data = val_data.map(lambda x, y: (preprocess_input(x), y))
val_data = val_data.prefetch(buffer_size=tf.data.AUTOTUNE)
```

Model Training:

```
#Instantiate the MobileNetV3Large model
conv_base = MobileNetV3Large(
    weights='imagenet',
    input_shape=(224, 224, 3),
    include_top=False
)
#Freeze weights of the Base Model
conv_base.trainable = False

#conv_base.summary()
```

```
model = Sequential([
    conv_base,
    GlobalAveragePooling2D(),
    BatchNormalization(),
    Dense(128, activation='relu'),
    Dropout(0.3),
    Dense(64, activation='relu'),
    Dropout(0.3),
    Dense(6, activation='softmax')  # Output layer with softmax for multi-class classification
])

model.summary()
```

```
# Compile the model
model.compile(
    optimizer=Adam(learning_rate=0.0001),
    loss='categorical_crossentropy',
    metrics=['accuracy']
)
```

```python
#Checkpoint after every epoch and Save Model with best validation accuracy
checkpoint_path = '/content/drive/My Drive/model_checkpoints/best_model.keras'
checkpoint_callback = ModelCheckpoint(checkpoint_path, save_best_only=True, monitor='val_accuracy', mode='max')
```

```python
# Train the model
history = model.fit(
    train_data,
    validation_data=val_data,
    epochs=100,
    callbacks = [EarlyStopping(monitor='val_loss', patience=5, restore_best_weights=True),checkpoint_callback
    ]
)
```

Performance Validation:

```
training = model.evaluate(train_data)
evaluation = model.evaluate(val_data)

# Print the evaluation metrics

print("Training Loss:", training[0])
print("Training Accuracy:", training[1])
print("Validation Loss:", evaluation[0])
print("Validation Accuracy:", evaluation[1])
```

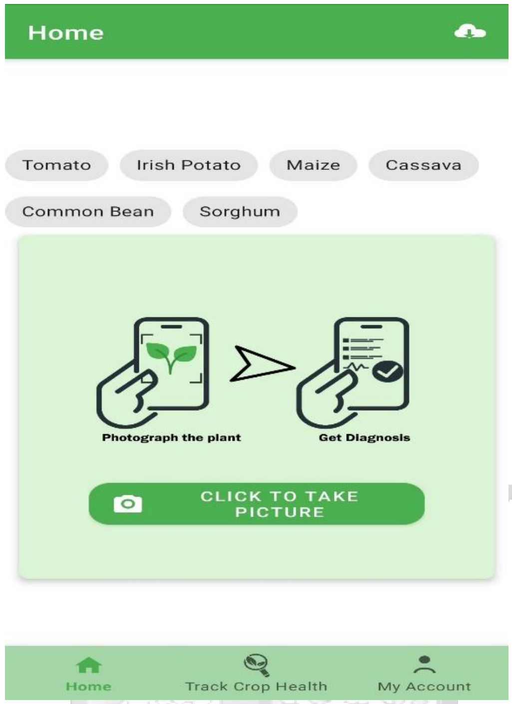
```
acc = history.history["accuracy"]
val_acc = history.history["val_accuracy"]

loss = history.history["loss"]
val_loss = history.history["val_loss"]
```

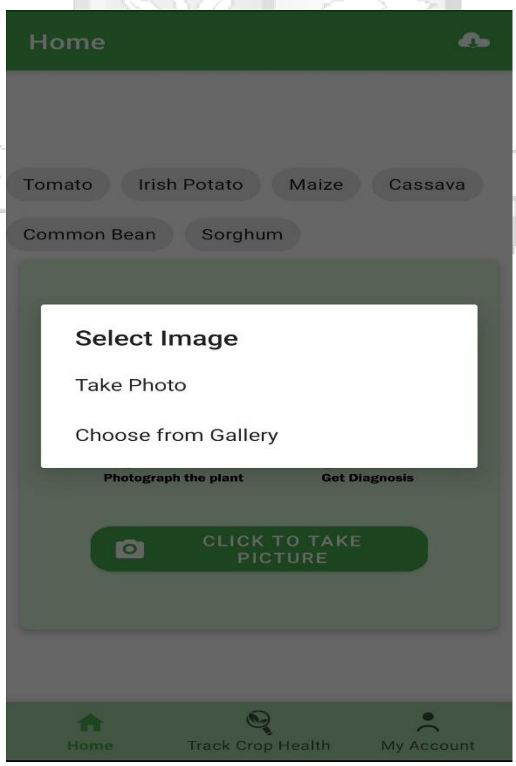
```
plt.figure(figsize=(12,6))
plt.subplot(1,2,1)
plt.plot(range(32),acc, label="Training Accuracy")
plt.plot(range(32),val_acc, label="Validation Accuracy")
plt.xlabel("Epochs")
plt.ylabel("Accuracy")
plt.legend(loc="lower right")
plt.title("Training and Validation Accuracy")

#plt.figure(figsize=(6,6))
plt.subplot(1,2,2)
plt.plot(range(32),loss, label="Training Loss")
plt.plot(range(32),val_loss, label="Validation Loss")
plt.xlabel("Epochs")
plt.ylabel("Loss")
plt.legend(loc="upper right")
plt.title("Training and Validation Loss")
plt.show()
```

Appendix E: System Implementation Screenshots

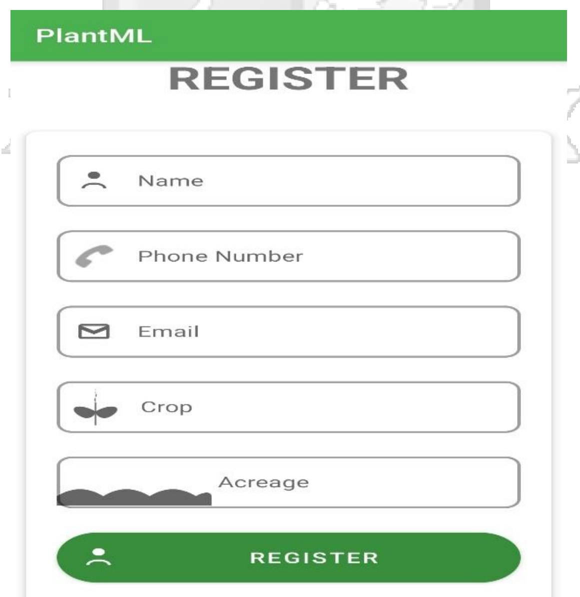


Appendix E.1: Application Landing Page

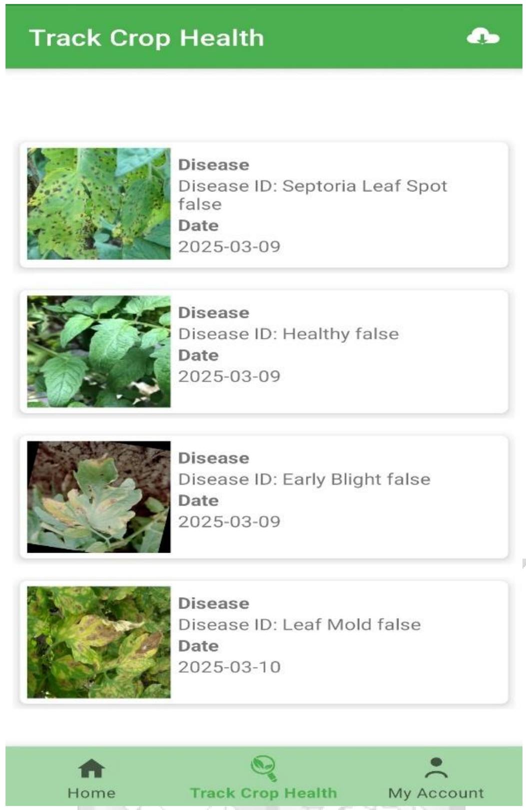


Appendix E.2: Capture Crop Image

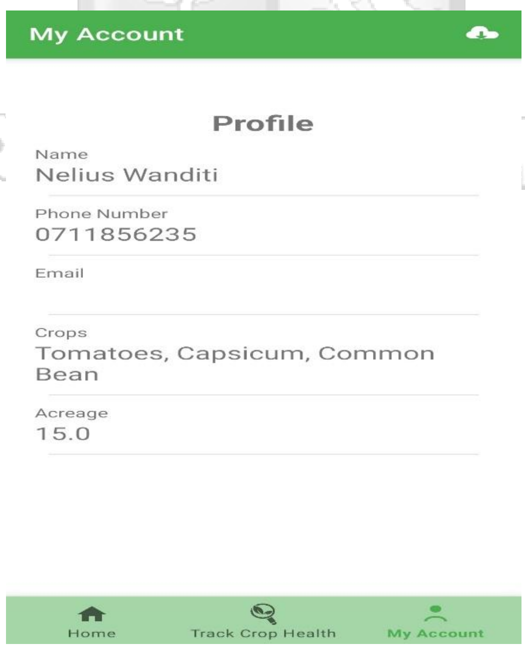
Appendix E.3: Diagnostics Outcome



Appendix E.4: User Registration



Appendix E.5: Monitor Crop Health



Appendix E.6: User Profile