AI Based Automation Solution to Foster Grading of Vegetables and Fruits to Support Uzhavan Logistic Planning

Submitted by

AANDAL S A (821121104002)
KEERTHANA J (821121104026)
NITHYASHRI B M (821121104040)

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

KINGS COLLEGE OF ENGINEERING, PUNALKULAM

ANNA UNIVERSITY:: CHENNAI 600 025

MAY 2025

ABSTRACT

Grading and sorting play a vital role in agriculture by ensuring product quality and minimizing post harvest losses. This study presents a robust AI-powered system designed to automate various processes, including classification, grading, expiration prediction, nutritional evaluation, and disease detection in fruits and vegetables. The proposed solution integrates advanced deep learning techniques such as Convolutional Neural Networks (CNNs), YOLO (You Only Look Once) for real-time object detection, Artificial Neural Networks (ANNs) for quality assessment, and Long Short-Term Memory (LSTM) networks for predicting shelf life. Image datasets sourced from platforms like Kaggle, Roboflow, and other repositories undergo automated preprocessing using a Personal Image Classifier, ensuring standardized inputs for efficient model training and validation. A customized CNN model is built and fine-tuned through preliminary experimentation. The complete system is deployed through a user-friendly Android application developed using MIT App Inventor. Experimental results demonstrate high classification accuracy, reliable grading performance, and effective prediction of shelf life and disease presence. This flexible and cost-effective solution is well-suited for real-time implementation in modern agricultural environments.

CHAPTER 1
INTRODUCTION

1.1 AI-POWERED PRODUCE ANALYSIS

Agriculture, being one of the most essential yet vulnerable sectors, now faces a pressing need for modern, technology-driven solutions - especially in post-harvest handling - making the demand for intelligent systems more critical than ever. Farmers and vendors often face challenges in manually evaluating produce due to lack of time, training, and consistent standards.This project aims to address these challenges by developing an AI-powered automation system capable of analyzing fruits and vegetables based on visual characteristics, shelf life, and quality parameters.

The proposed system leverages image inputs to perform classification, quality grading, expiry date prediction, and disease detection, while also retrieving nutritional information from verified external sources. All these modules are unified within a mobile platform to provide a seamless, real-time solution that operates with minimal user effort. It eliminates the subjectivity of manual inspection by providing data-driven results using pretrained deep learning models. This kind of automation reduces dependency on expert human intervention and promotes transparency and efficiency in agricultural workflows.

1.2 NEED OF THE SYSTEM

Post-harvest losses account for a significant percentage of food wastage, particularly in developing countries, due to improper sorting, grading, and disease control. These losses are exacerbated by outdated manual practices where human inspection often leads to inconsistency, misjudgment, and delayed responses. Furthermore, many small-scale farmers and market vendors lack access to lab-based testing, equipment, or expert advice, leaving them at a disadvantage during quality assessment and price negotiations.

An intelligent automated system is essential to bridge this gap. It provides objective grading standards, detects early signs of spoilage or disease, and enables better sorting decisions for transport and sale. By offering instant feedback, this system empowers stakeholders across the supply chain - from farms to retail - to make informed decisions. The integration of such a system into mobile devices ensures broader accessibility, especially in rural and semi-digital zones, and supports precision agriculture and digital transformation efforts promoted by governments and agri-tech platforms.

1.3 TECHNOLOGY STACK BEHIND THE SOLUTION

The backbone of this system is a set of optimized deep learning models trained on diverse datasets of fruits and vegetables. Convolutional Neural Networks (CNNs) are used for classification and grading tasks, where surface texture, shape, and color are analyzed to differentiate between types and quality levels of produce. YOLO (You Only Look Once) is employed for object detection, especially for identifying diseases like rot, scab, or black scurf through bounding box localization.

For shelf life prediction, Long Short-Term Memory (LSTM) networks are used to model temporal degradation by analyzing changes in visual features across sequential images. Additionally, rule-based classifiers are incorporated to provide explainable, lightweight classification, particularly for systems with constrained hardware or in offline use cases. The USDA FoodData Central API is integrated to provide verified nutritional values based on the classified produce.

All models are implemented in Python, utilizing frameworks like FastAI, PyTorch, and TensorFlow Lite for training and deployment. The mobile interface is developed using MIT App Inventor, enabling user-friendly access and compatibility with Android devices. This technical stack ensures that the system remains robust, scalable, and responsive across various devices and use environments.

1.4 APPLICATION AREAS AND AGRICULTURAL RELEVANCE

This system has wide-ranging applications in both primary agriculture and allied sectors. For farmers, it acts as a quality control assistant, helping them verify the standard and condition of their harvest before sale or storage. They can classify the type of produce, check for visual defects or diseases, and assess freshness - all with a simple image. This empowers them to fetch fair prices in the market and reduce post- harvest rejections.

In agricultural markets (mandis), the system can be used during auctions to verify grade and type, ensuring standardized pricing and improved buyer trust. Warehouses and cold storage units can use the expiry prediction module to prioritize which produce to ship or consume first. Retailers and Distributors benefit from the disease detection and grading functionalities by reducing the risk of stocking spoiled goods and ensuring customer satisfaction.

Beyond agriculture, the nutritional data retrieval module finds relevance in food labeling, dietary advisory systems, and consumer-focused applications that value transparency and health tracking. The mobile-based architecture of this system makes it highly adaptable for use in both rural farming communities and urban food distribution networks. It aligns with digital agriculture goals and supports smart farming initiatives aimed at minimizing waste and maximizing quality across the supply chain.

CHAPTER 2

LITERATURE SURVEY

2.1 TITLE : Fruit Vision: A Deep Learning-Based Automatic Fruit Grading System

AUTHOR : Ahatsham Hayat, Fernando Morgado-Dias, Tanupriya Choudhury, Thipendra P. Singh, and Ketan Kotecha

PUBLICATION : Published in Open Agriculture.

YEAR 2024

HIGHLIGHTS

FruitVision is a system that leverages deep learning techniques, specifically pre-trained Convolutional Neural Networks (CNNs) such as MobileNetV3, DenseNet, and ResNet, to automate the grading of fruits including apples, bananas, and dates. The model efficiently extracts and processes visual features like color, texture, and shape to assess fruit quality. The study demonstrated the effectiveness of automatic feature extraction in achieving high classification performance, thereby showcasing the potential of CNNs in agricultural automation.

ADVANTAGES

The key advantage of FruitVision lies in its ability to deliver high accuracy in classifying fruits using lightweight and efficient CNN models. These models are particularly suitable for embedded systems, making the solution scalable and effective for visual quality analysis in fruits.

DRAWBACKS

Despite its strengths, the system is limited to fruit grading and does not extend support to vegetables. Furthermore, it lacks functionality for predicting additional quality parameters such as expiry dates or nutritional content. Another notable limitation is the absence of integration with real-time mobile applications, which restricts its practical use in field environments where portability and instant results are essential.

2.2 TITLE : EA-CNN: Enhanced Attention-CNN with Explainable AI for Fruit and Vegetable Classification

AUTHOR : Zeshan Aslam Khan, Muhammad Waqar, Khalid Mehmood Cheema, Ali Abu Bakar Mahmood, Quratul Ain, Naveed Ishtiaq Chaudhary, Abdullah Alshehri, Sultan S.Alshamrani, and Muhammad Asif Zahoor Raja

PUBLICATION : Published in ScienceDirect

YEAR 2024

HIGHLIGHTS

EA-CNN is a classification system that integrates an attention mechanism into the traditional Convolutional Neural Network (CNN) architecture to enhance performance. The model focuses on identifying and emphasizing critical regions within fruit and vegetable images during classification. It also incorporates explainable AI (XAI) techniques, allowing users to visualize attention weights and better understand the basis for each classification.

ADVANTAGES

One of the main advantages of EA-CNN is its ability to provide interpretable predictions through visualization of attention maps. This not only improves classification accuracy but also builds trust and transparency in the decision- making process. Additionally, it supports classification of both fruits and vegetables, offering a wider application scope than many earlier models.

DRAWBACKS

However, the EA-CNN model does not include broader grading features such as disease detection, expiry prediction, or nutritional analysis. Furthermore, due to the added computational complexity of the attention mechanism and explainability layer, the model becomes resource-intensive, making it less suitable for real-time or mobile-based deployments, especially in resource-constrained agricultural environments.

2.3 TITLE : Efficient Fruit Disease Diagnosis on Resource-Constrained Agriculture Devices

AUTHOR : Sadaf Iftikhar, Hasan Ali Khattak, Ahsan Saadat, Zoobia Ameer, and Muhammad Zakarya

PUBLICATION : Published in ScienceDirect

YEAR 2024

HIGHLIGHTS

The study developed a fruit disease diagnosis system tailored for low-power agricultural devices. By using lightweight CNNs and optimization methods like quantization and pruning, the system maintained accuracy while minimizing resource demands, enabling effective disease detection on constrained hardware.

ADVANTAGES

The primary advantage of this approach lies in its efficiency and portability. By tailoring the model for low-resource environments, the solution enables real-time disease detection directly on edge devices, making it highly practical for deployment in rural or field conditions where high-end infrastructure may not be available.

DRAWBACKS

Despite its strengths in disease diagnosis, the system lacks several key features essential for a comprehensive grading solution. It does not include quality scoring, nutritional value estimation, or grading classifications based on factors like ripeness, texture, or color. Additionally, the study does not provide a real-time mobile interface, limiting the user-friendliness and accessibility of the system for non-technical users in agricultural settings.

2.4 TITLE : Real-Time Visual Inspection System for Grading Fruits Using Computer Vision and Deep Learning Techniques

AUTHOR : Nazrul Ismail and Owais A. Malik

PUBLICATION : Published in ScienceDirect **YEAR**

: 2024

HIGHLIGHTS

A real-time visual inspection system designed for fruit grading in industrial production environments. The system integrates computer vision techniques with CNN-based classification models and is implemented alongside a conveyor mechanism for continuous grading of fruits on production lines. It is primarily focused on detecting external defects such as blemishes, bruises, and deformities, enabling high-throughput sorting based on visual quality.

ADVANTAGES

The system performs fast and reliable inspection in real-time and is well suited for automated grading in industrial settings. Its integration with a mechanical conveyor makes it effective for handling large volumes of produce, significantly increasing efficiency and reducing human error during the sorting process.

DRAWBACKS

While the model excels in high-speed detection of external defects, it does not account for predictive quality assessments such as expiry date estimation, nutritional value analysis, or internal defect detection. Moreover, the focus on industrial-grade deployment may limit its accessibility and practicality for small scale or mobile agricultural applications where portability and flexibility are crucial.

2.5 TITLE : Classification and Grading of Multiple Varieties of Apple Fruit

AUTHOR : Anuja Bhargava and Atul Bansal

PUBLICATION : Published by Springer

YEAR 2022

HIGHLIGHTS

A classification and grading system specifically targeting multiple varieties of apples. Their approach combines classical machine learning algorithms, particularly Support Vector Machines (SVMs), with manual extraction of color and texture features. The system was successful in achieving high accuracy in distinguishing between apple types and grades based on these visual attributes, demonstrating the viability of traditional methods in specific, narrowly defined tasks.

ADVANTAGES

The model is effective and accurate for apple-specific grading tasks and is relatively lightweight in terms of computational complexity. Its reliance on classical ML techniques makes it easy to interpret and suitable for controlled environments where only one type of fruit is processed.

DRAWBACKS

However, the system's dependency on manual feature engineering makes it less flexible and difficult to adapt to other types of fruits or vegetables without redesigning the feature extraction process.Furthermore, it lacks scalability and does not include advanced capabilities like nutritional estimation, expiry prediction, or integration with real-time mobile or industrial platforms.

2.6 TITLE : Machine Learning–Based Detection and Sorting of Multiple Vegetables and Fruits

AUTHOR : Anuja Bhargava, Atul Bansal, and Vishal Goyal

PUBLICATION : Published by Springer

YEAR 2021

HIGHLIGHTS

A machine learning-based detection and sorting system capable of handling multiple types of fruits and vegetables. The approach begins with image preprocessing using Gaussian filtering to reduce noise, followed by fuzzy c- means clustering for effective segmentation of produce regions. Once segmented, classification algorithms such as Support Vector Machines (SVM) and Artificial Neural Networks (ANN) are used to categorize the produce based on visual features like shape, size, and color.

ADVANTAGES

This system represents a step forward in making machine learning models more adaptable to real-world agricultural scenarios by supporting multiple categories of produce rather than focusing on a single fruit. The combination of segmentation and classification techniques improves the accuracy of sorting, and the method offers flexibility for classifying a wide variety of items.

DRAWBACKS

Despite its versatility, the model requires substantial computational power due to its multi-stage processing pipeline, which may limit its deployment on low-resource or mobile devices. In addition, it lacks predictive functionalities such as disease detection, expiry prediction, and nutritional analysis, which are increasingly important for comprehensive post-harvest quality management and logistics planning.

2.7 TITLE : Classification and Grading of Multiple Varieties of Apple Fruit

AUTHOR : Anuja Bhargava and Atul Bansal

PUBLICATION : Published by Springer

YEAR 2022

HIGHLIGHTS

The application of deep convolutional neural networks (CNNs) for the classification of fruits, focusing particularly on architectures like Inception V3. The study leveraged transfer learning to improve model performance, allowing the system to benefit from pre-trained weights on large image datasets. This approach significantly reduced training time while still achieving high classification accuracy, especially when working with small or moderately sized fruit image datasets. The model effectively identified and categorized multiple fruit types based on key visual features such as shape, color, and texture.

ADVANTAGES

The approach demonstrated that pre-trained deep learning models can be effectively applied to fruit classification tasks with minimal retraining, making them highly efficient and accessible. The model achieved strong accuracy across various fruit types, validating the effectiveness of deep CNNs in feature extraction and classification.

DRAWBACKS

Despite its classification success, the system is limited in scope—it does not handle grading tasks such as identifying surface defects, assessing quality, or predicting freshness or expiry. Additionally, the solution is not integrated into a mobile or real-time platform, reducing its practical utility for farmers or field-level deployment.

2.8 TITLE : An Effective Pomegranate Fruit Classification Based on CNN-LSTM Deep Learning Models

AUTHOR : M.T. Vasumathi and Kamarasan Mari

PUBLICATION : Published by ResearchGate

YEAR 2021

HIGHLIGHTS

A hybrid deep learning approach using CNN-LSTM models for classifying pomegranate fruits into two categories: normal and abnormal. The CNN component extracts detailed spatial features from fruit images, while the LSTM captures sequential dependencies—either through image slices or time- based transformations—enhancing the classification capability for subtle visual differences.

ADVANTAGES

This combination of convolutional and recurrent networks allows the system to capture both static and dynamic aspects of the input data, resulting in improved accuracy over traditional CNN-only models. It proves effective in detecting anomalies that may not be evident from spatial features alone, showcasing the strength of temporal modeling in agricultural classification tasks.

DRAWBACKS

Despite its high performance, the system is specifically designed for a single fruit type and performs only binary classification, limiting its generalization to other fruits or vegetables. Additionally, the model does not offer grading, disease prediction, or expiry estimation, and it lacks real-time or mobile deployment, reducing its broader applicability in agricultural automation.

2.9 TITLE : A Hybrid Deep Learning-Based Fruit Classification Using Attention Model and Convolution Autoencoder

AUTHOR : Gang Xue, Shifeng Liu, and Yicao Ma

PUBLICATION : Published by Springer

YEAR 2020

HIGHLIGHTS

A hybrid deep learning model that integrates attention mechanisms with a convolutional autoencoder (CAE-ADN) for the classification of various fruit types. The attention modules help focus the model on salient regions within the images, while the autoencoder enables the system to learn compact and robust feature representations. This dual architecture significantly enhances the model's ability to distinguish between different fruits based on subtle visual cues.

ADVANTAGES

The use of an attention mechanism improves the interpretability and accuracy of classification by directing focus to the most relevant parts of the input image. The integration with a convolutional autoencoder further boosts the model's capability by effectively compressing and reconstructing critical features, resulting in strong classification performance across diverse fruit categories.

DRAWBACKS

Despite its promising architecture, the system is limited to classification tasks only. It does not incorporate grading features such as disease detection, expiry date prediction, or nutritional analysis. Furthermore, the study does not discuss real time application or mobile deployment, which restricts its usability in practical, on field scenarios where instant feedback and accessibility are essential.

2.10 TITLE : Automatic Detection and Grading of Multiple Fruits by Machine Learning

AUTHOR : Anuja Bhargava and Atul Bansal

PUBLICATION : Published by Springer

YEAR : 2019

HIGHLIGHTS

A classical machine learning-based pipeline for the automatic detection and grading of various fruits. Their approach utilizes threshold-based segmentation for feature extraction, followed by traditional classifiers to assign grades. The system was tested across multiple fruit types and achieved moderate classification accuracy, mainly through color and texture analysis.

ADVANTAGES

The method is relatively computationally lightweight and can be implemented with limited hardware resources. It also demonstrates the feasibility of automated fruit grading without relying on large datasets or deep learning models, making it suitable for settings with constrained resources or limited technical infrastructure.

DRAWBACKS

The approach is heavily dependent on manual feature engineering, which reduces scalability and adaptability to different fruit types. It lacks integration with advanced deep learning frameworks, real-time capabilities, and predictive features such as disease detection, expiry estimation, or nutritional content analysis. Furthermore, it is not designed for mobile or field-level use, limiting its practical deployment.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

In the current agricultural supply chain, the grading of vegetables and fruits is predominantly performed through manual inspection by human laborers. This process involves workers visually assessing produce based on surface characteristics such as size, color, shape, ripeness, and the presence of any visible defects or damage. This traditional method is typically carried out at collection centers, wholesale markets, or directly at the farm level before packaging and transportation.

Grading is usually done by hand or with the aid of simple mechanical tools such as sieves, rollers, and weighing scales. However, even in facilities where basic sorting belts or conveyors are available, the grading process remains heavily dependent on human input. In many cases, the criteria for classifying produce into different quality grades are not standardized and can vary between workers or locations, which often results in inconsistent outcomes. Moreover, the absence of clear visual guidelines or digital tools means that grading quality can fluctuate depending on the experience, training, and physical condition of the laborers involved.

One of the major limitations of this system is its inefficiency. Manual grading is time-consuming and labor-intensive, leading to bottlenecks during peak harvesting periods when large volumes of produce must be processed quickly. The reliance on human workers also makes the system prone to fatigue-induced errors, inconsistency, and increased costs. Additionally, the lack of integration with digital platforms like Uzhavan prevents real-time tracking of graded produce and delays coordination with packaging and logistics, ultimately leading to post-harvest losses, spoilage, and reduced market value. As a result, farmers often face difficulties in maintaining quality standards and achieving fair pricing for their produce in the market.

3.2 DISADVANTAGES OF THE EXISTING SYSTEM

- The grading process is subjective and inconsistent, as it relies on human judgement, which varies from person to person.
- It is time-consuming and inefficient, slowing down the overall handling and distribution of produce.
- The system is heavily labor-dependent, increasing operational costs and making it difficult to scale.
- Most existing systems lack integration with centralized platforms, making it difficult to connect farmers, vendors, and consumers efficiently.

3.3 PROPOSED SYSTEM

The proposed system introduces an intelligent, automated solution that leverages Artificial Intelligence (AI) and Computer Vision to perform precise, efficient, and scalable grading of vegetables and fruits. This system is designed to replace traditional manual grading methods, ensuring consistency, speed, and seamless integration with digital platforms like Uzhavan. At the core of the system is a high-resolution imaging module, which captures images of produce as they move along a conveyor system. These images are then analyzed in real time by advanced AI algorithms trained using machine learning techniques, such as convolutional neural networks (CNNs). These algorithms are capable of assessing key quality parameters - including size, shape, color, texture, ripeness, and the presence of surface defects - with high accuracy. The model can classify the produce into various predefined grades (such as Grade A, B, or rejected), based on customized quality standards suited to specific market or export requirements.

The grading results are recorded digitally, and the system can be integrated with an automated sorting mechanism that physically separates the produce into corresponding categories. This entire process is streamlined and occurs in real time, enabling high-throughput sorting without human intervention.

It eliminates the need for manual inspection or external hardware, offering quick and reliable results directly through an Android app built using MIT App Inventor. This streamlines quality assessment at the user level without dependency on centralized logistics systems.

Additionally, the system supports scalability, allowing deployment at various levels - individual farms, collection centers, or large-scale warehouses. It reduces dependency on manual labor, minimizes grading errors, lowers operational costs, and significantly reduces post-harvest losses. The data collected can also be used for future analysis, crop planning, and decision-making, further empowering the agricultural ecosystem with intelligent insights.

3.4 ADVANTAGES OF THE PROPOSED SYSTEM

- Improved accuracy and consistency of the system uses AI algorithms to ensure precise and uniform grading, minimizing the inconsistencies found in manual sorting.
- Faster processing of real-time image analysis allows the system to grade large volumes of produce quickly, boosting overall efficiency.
- Reduced labor dependency automation significantly lowers the need for manual labor, reducing costs and human errors.
- The system enables real-time quality assessment by allowing users to capture an image through a mobile application and instantly receive results.
- Accurate and timely grading helps reduce spoilage, preserving the quality and market value of the produce helps to minimized the post –harvest loss.

CHAPTER 4

SYSTEM REQUIREMENTS

4.1 INTRODUCTION

The integration of artificial intelligence and mobile technology has opened new avenues for solving practical problems in agriculture and beyond. This project aims to develop a mobile-based solution that utilizes image classification techniques to assist in the grading and sorting of fruits and vegetables. By leveraging deep learning models trained on diverse image datasets and implementing the solution through an Android application, the system enables real-time analysis and decision making directly from the field.

The primary objective of this project is to create a user-friendly, efficient, and portable tool that can enhance productivity and quality control in agricultural processes. The application is designed to be intuitive for users, capable of processing images on the go, and provides immediate feedback based on model predictions. With the support of tools such as MIT App Inventor and high-performance image classifiers, the project represents a meaningful step towards accessible, tech-driven solutions for modern agriculture.

4.2 HARDWARE REQUIREMENT

Device	:	Android Phone
Camera Resolution	:	8 Megapixels
Battery Capacity	:	3000 mAh

4.3 SOFTWARE REQUIREMENT

Front End	:	MIT App Inventor
Back End	:	Dataset, Personal Image Classifier
Operating System	:	Windows 10
IDE	:	AndroidStudio

The Android application serves as the user-friendly interface for interacting with the image classification system. Developed using MIT App Inventor, the app allows quick and visual development of mobile applications without advanced coding knowledge. Its primary functions include capturing images of agricultural produce using the device's camera, processing the image through an embedded machine learning model, and displaying the classification results in real time.

The app is designed with simplicity in mind, making it accessible to farmers and agricultural workers with minimal technical experience. Users can take a photo, and the app instantly analyzes the image to determine the type and quality of the fruit or vegetable - such as whether it is fresh, ripe, or spoiled. Results are displayed in a clear, easy-to-understand format.

The application is optimized for offline use, allowing real-time operation even in rural areas without internet access. It is lightweight, responsive, and compatible with most Android devices. The focus is on portability, speed, and ease of use, enabling farmers to make quick decisions directly in the field based on accurate AI- driven insights.

4.4 HARDWARE SPECIFICATIONS

The proposed system is designed to operate on standard Android smartphones, making it portable, cost-effective, and easy to deploy in agricultural settings. To ensure smooth functioning of the image classification app, the device should be equipped with a camera of at least 8 megapixels. The device should also have a minimum of 16 GB of internal storage to accommodate the app, temporary image files, and possibly offline model data. Since the application is intended for on-field use, the smartphone should be **lightweight and portable**, with a battery capacity of 3000 mAh **or more** to support extended use without frequent charging. Additional features like good screen visibility under sunlight and durable casing can further enhance usability in real-world agricultural environments.

CHAPTER 5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

The developed system architecture offers a complete AI-powered solution for analyzing fruits and vegetables through image classification. Using a personal image classifier built with CNN, the system can classify produce, grade quality, predict diseases, display nutritional content, and estimate expiry dates—all based on a single image. These models are trained and then integrated into an Android application using MIT App Inventor, allowing users to access all functionalities through a simple mobile interface. The system ensures fast and reliable results without manual effort, making it a practical and efficient tool for real-time agricultural analysis.

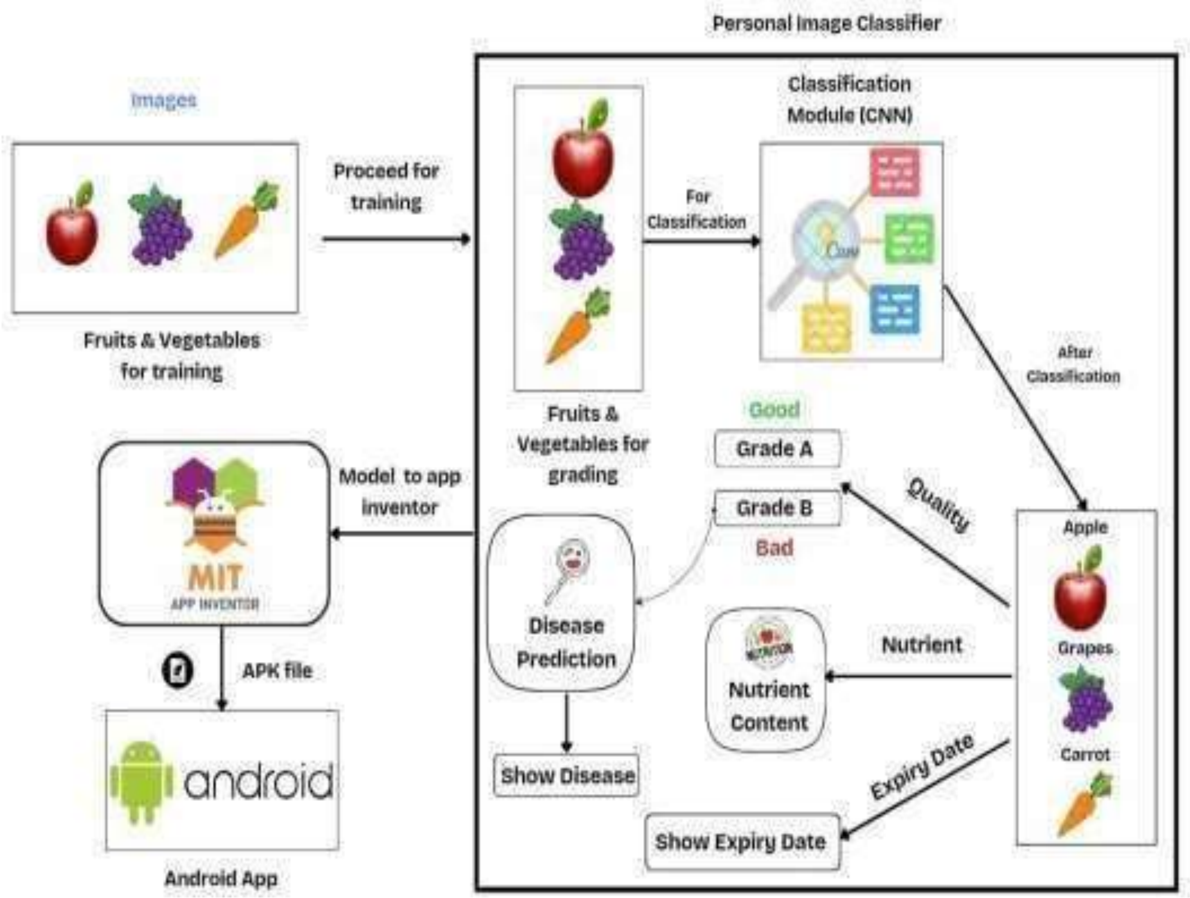


Fig.5.1: Architecture Diagram

5.2 DATA FLOW DIAGRAM

The Data Flow Diagram Level 0 illustrates the overall flow of data in the AI- Based Grading System. The system interacts with two main external entities: **Farmers** and **Vendors/Individual Users**, who serve as primary sources of input. These users provide fruit or vegetable images through a user interface, which are then processed by the **AI-Based Grading System**. Internally, the system routes these inputs to the **CNN Models**, responsible for analyzing image data to predict the category, grade, or condition of the produce. The output generated is returned to the users, providing them with critical insights such as the type of item, quality, and usability. This high level representation captures the essence of user-system-model interaction in a streamlined manner.

DFD 0

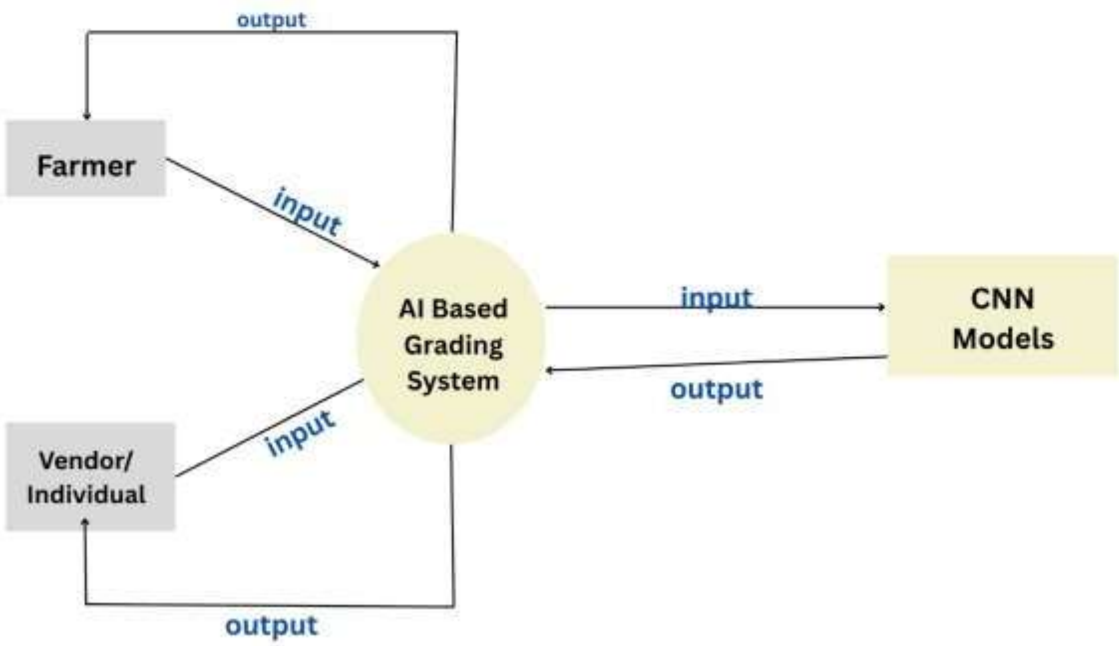


Fig.5.2: Data Flow Diagram – Level 0

The Level 1 DFD expands the internal processes of the system, beginning with image acquisition from users like farmers or vendors. The captured image is first routed to the Classification Module, which identifies the category of the item (e.g., apple, grape, carrot). The image also proceeds to the **Grading Module**, which evaluates visual attributes to assign a grade such as Grade A or B. If the grade is identified as Grade B, indicating potential defects or issues, the image is further analyzed by the Disease Detection Module, which predicts specific diseases affecting the produce. Simultaneously, the classified fruit or vegetable type is passed to two other modules: the Expiry Prediction Module, which estimates shelf life, and the Nutrient Analysis Module, which fetches nutritional information from an integrated source. The final output includes item type, grade, disease status, nutritional values, and remaining shelf life—all delivered seamlessly to the user. This structured data pipeline ensures intelligent, multi-dimensional analysis of agricultural produce through a unified AI system.

DFD 1

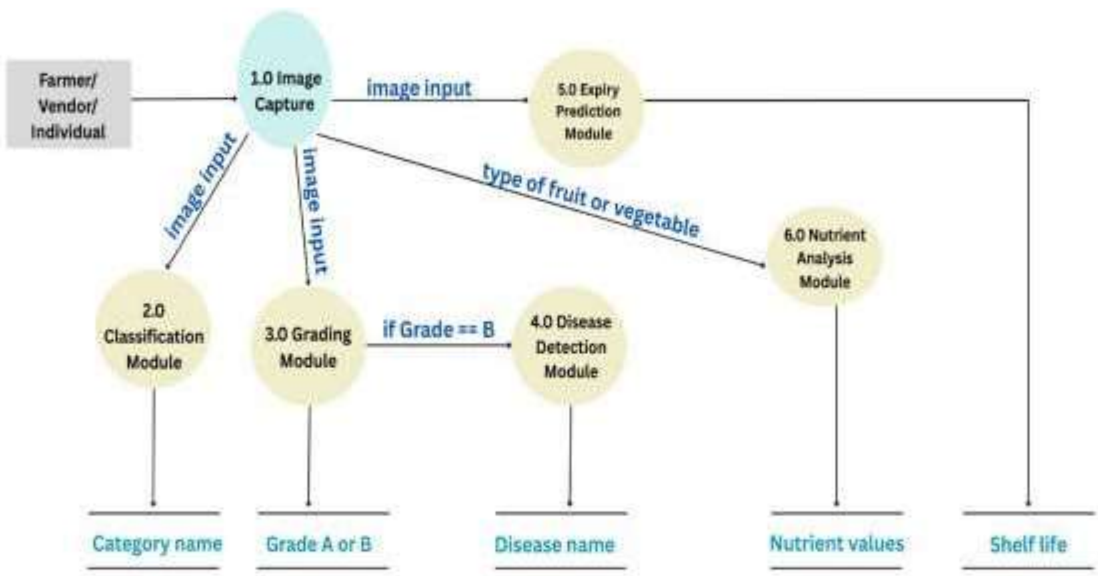


Fig.5.3: Data Flow Diagram – Level 1

5.3 USE CASE DIAGRAM

Through the "Capture Image via WebViewer" use case, the user initiates the produce evaluation process, which activates the AI Engine to begin automated analysis. This triggers the optional "Preprocess Image" use case to enhance image clarity, followed by "Classify Item," where the system identifies the type of fruit or vegetable. Based on this classification, the "Grade Quality (A/B)" use case assesses the produce's freshness. If the grade is B, the system invokes "Detect Disease" to check for visible signs of illness. Parallelly, the AI Engine processes the "Predict Expiry" and "Fetch Nutritional Info" use cases to estimate shelf life and retrieve nutritional data. Finally, all results are compiled and presented through the "Display All Results with Score" use case. The seamless collaboration between the User Interface, AI Engine, and Backend System enables a user-driven, intelligent, and efficient grading workflow, providing actionable insights for improved decision-making in agriculture and trade.

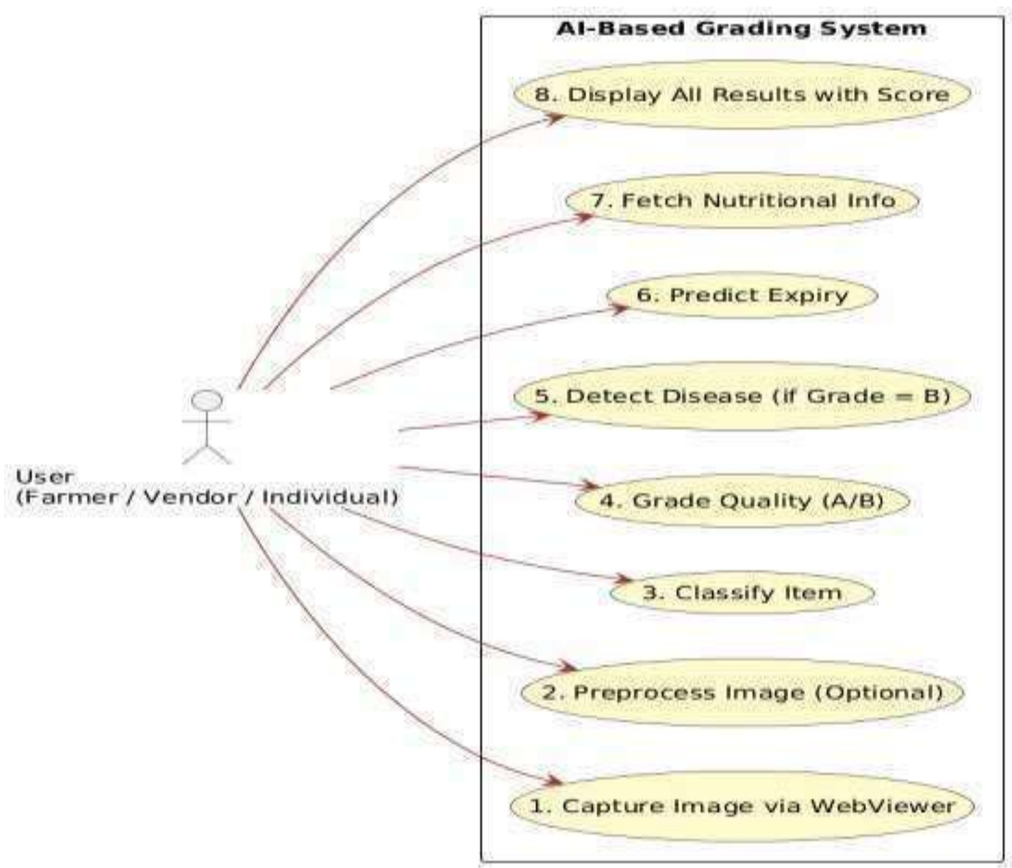


Fig.5.4: Use Case Diagram

5.4 SEQUENCE DIAGRAM

The sequence diagram presents a streamlined image-based workflow within an AI-powered mobile app for analyzing produce. Upon capturing an image, the app engages several specialized modules: classification to identify the produce type, grading to assess quality, and, if quality is low (Grade B), disease detection to check for decay. Expiry prediction and nutrient analysis are also performed independently. All modules interact only with the mobile app, which compiles the outputs into a unified report for the user. This conditional and modular system ensures intelligent, flexible, and efficient post-harvest evaluation.

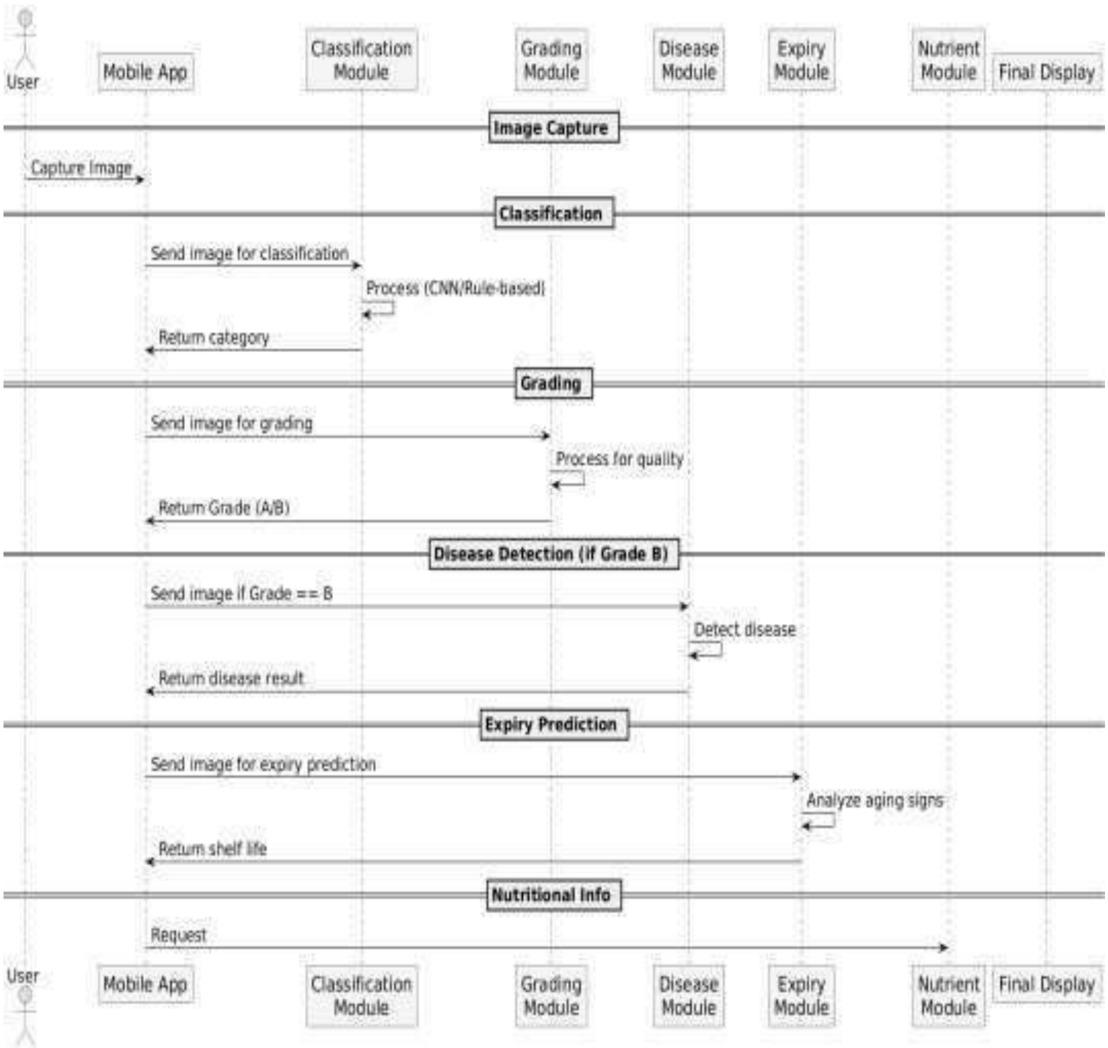


Fig.5.5: Sequence Diagram

5.5 CLASS DIAGRAM

The class diagram illustrates a modular AI-based agricultural automation system where the Mobile app acts as the central controller. It captures images using the Webviewer, optionally preprocesses them, and sends them to specialized modules for independent analysis. These modules - Classification, Grading, Disease, Expiry, and Nutrient analysis - process the input image and return results with associated probability scores to ensure reliable predictions. The Mobile app collects these outputs and displays them to the user. This design promotes clarity, scalability, and efficient decision- making for tasks such as classification, quality grading, disease detection, expiry prediction, and nutrient analysis.

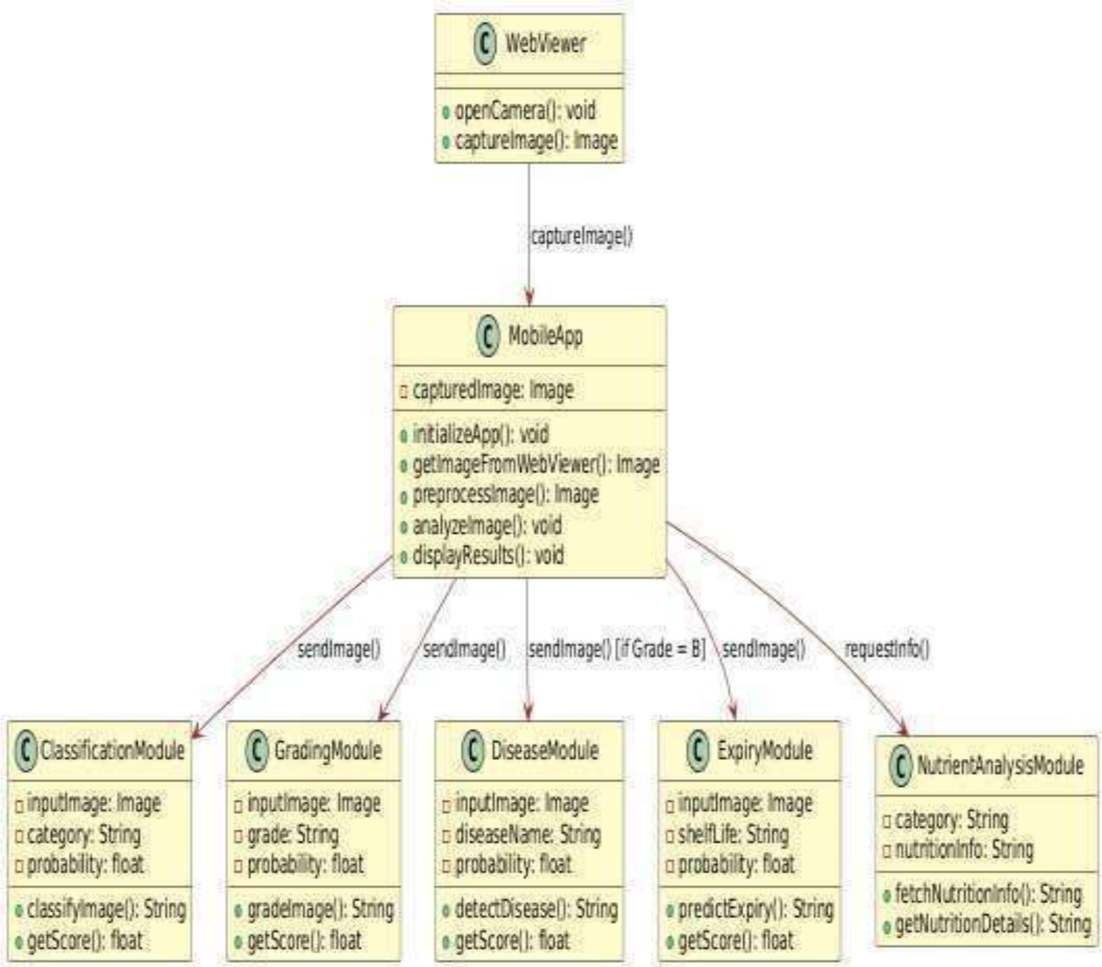


Fig.5.6: Class Diagram

5.6 ACTIVITY DIAGRAM

The AI-based agricultural grading system begins when a user captures an image of produce through a mobile app, initiating the analysis process. The system then processes the image to identify the type and assess the quality of the produce. If the quality is lower, it checks for disease or spoilage. The system also predicts the shelf life and provides nutritional information. Finally, all results are compiled and presented to the user, helping make informed decisions for post-harvest management.

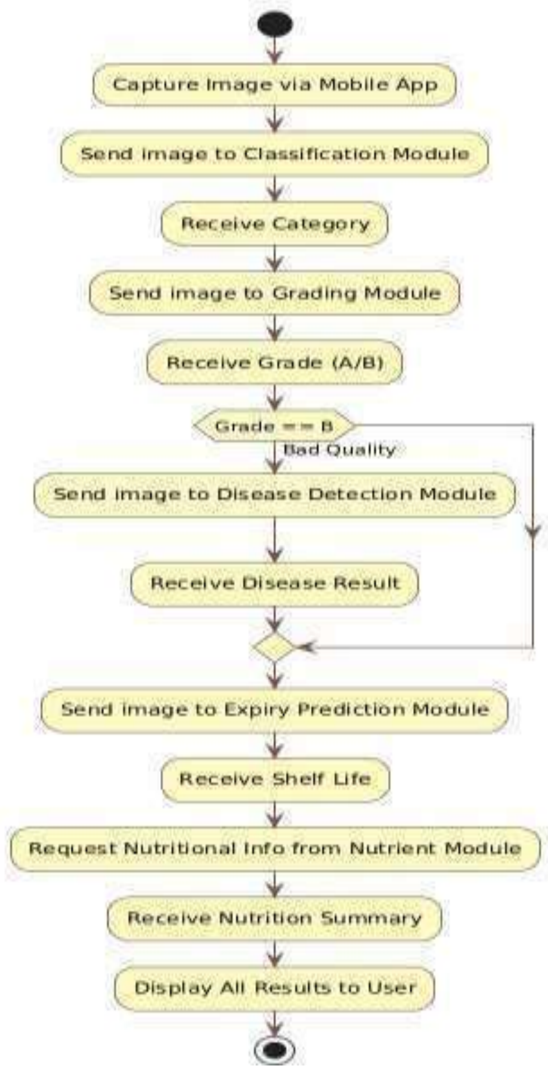


Fig.5.7: Activity Diagram

5.7 ENTITY-RELATIONSHIP DIAGRAM

The Entity-Relationship (ER) diagram represents the high-level structure of the AI-based agricultural automation system, showing how users interact with the mobile application to capture fruit and vegetable images that are processed through multiple intelligent modules. These modules work in coordination to classify, grade, and analyze the input for factors such as disease, shelf life, and nutritional content. The system establishes clear relationships between the user, the application, the image data, and the output results, illustrating how data flows seamlessly across components to deliver accurate and timely insights. This ER model emphasizes the cohesive integration of entities, enabling efficient image- driven analysis and decision-making within a unified mobile framework.

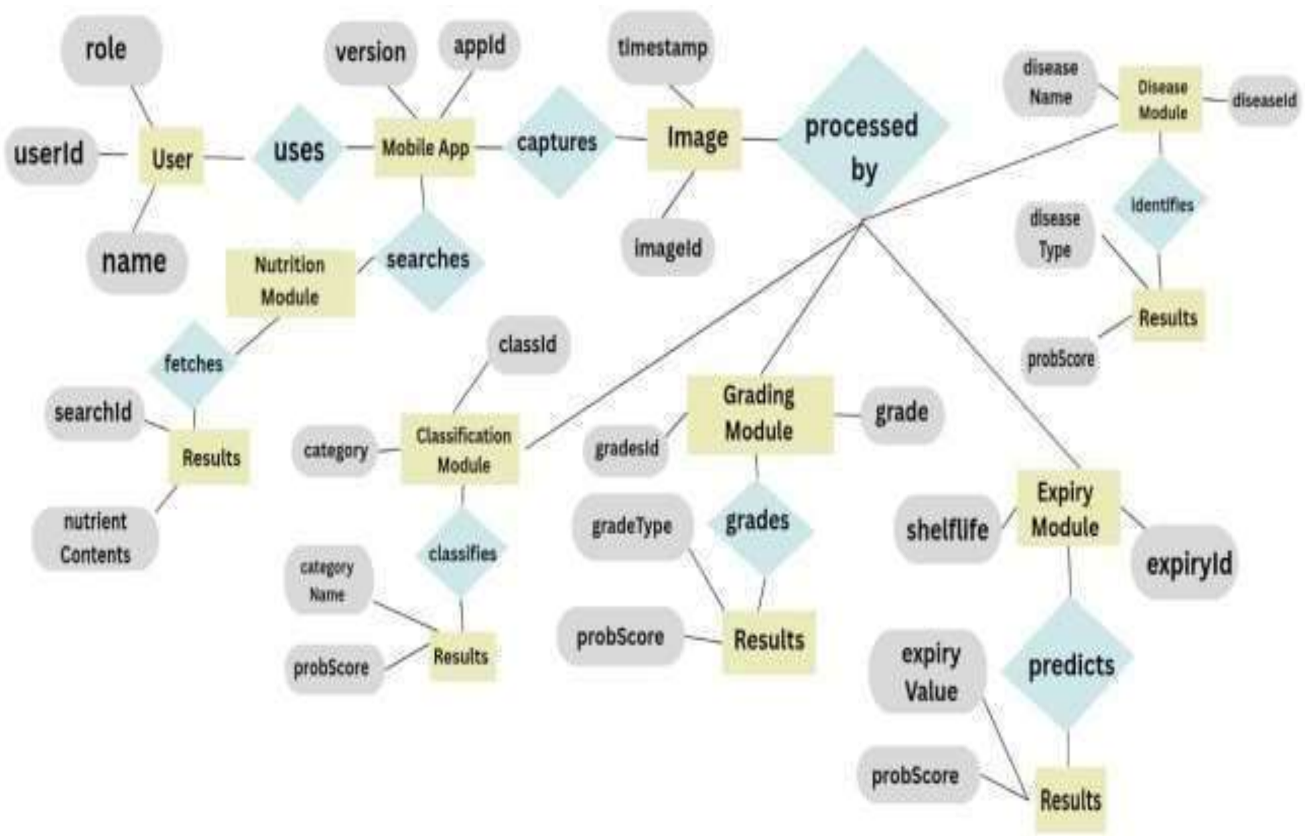


Fig.5.8: Entity Relationship Diagram

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 MODULES

6.1.1. Data Collection

6.1.2. Data Preprocessing

6.1.3. Training the Model for the following :

6.1.3.1.Fruits and Vegetables Classification

6.1.3.2.Grade Prediction

6.1.3.3. Expiry Date Prediction

6.1.3.4.Calculating Nutritional Value

6.1.3.5.Disease Prediction

6.1.4. Integrating with MIT App Inventor

6.1.5.Interpreting Results

6.1.1 DATACOLLECTION

The Data Collection Module is the foundational component of this AI-driven fruit and vegetable grading system, responsible for acquiring and organizing image data essential for training and evaluating machine learning models. In this project, a large scale and diverse dataset was curated to support classification, grading, nutritional evaluation, expiry prediction, and disease detection. Images of selected fruits (apples and grapes) and vegetables (potatoes and carrots) were collected from a combination of online open-source repositories - such as Kaggle and Roboflow - and custom real world captures. The real-world images were taken under varying lighting conditions, angles, and backgrounds to replicate real agricultural environments and ensure model robustness across scenarios. This diversity helps the model generalize better and increases its performance in unpredictable field settings.

Each image in the dataset is annotated with key metadata that includes the name of the fruit or vegetable, its quality grade (such as Grade A or Grade B), and other relevant attributes, such as color consistency, ripeness, firmness, or visible defects. These labels are crucial for supervised learning, allowing models to learn patterns that distinguish between good and bad produce or to identify specific characteristics necessary for accurate classification. For the modules related to fruit and vegetable classification, grade prediction, nutritional value estimation, and expiry date forecasting, a unified dataset was used to maintain consistency in training and testing. However, to enhance the precision of disease detection, a separate and specialized dataset containing images of diseased fruits and vegetables was employed. This separation ensures that the disease prediction module learns from highly relevant visual indicators, such as mold spots, discoloration, or abnormal textures.

The full dataset comprises a total of 82,213 images, carefully partitioned to support model training and evaluation. Of these, 61,488 images form the training set, each featuring a single fruit or vegetable. This subset is used to teach the model how to extract meaningful visual features and associate them with correct labels. The remaining 20,622 images are reserved for testing, ensuring that the model's performance is validated on unseen data. Additionally, a specialized multi-fruit dataset containing 103 images is included to train and evaluate the object detection model's ability to identify multiple fruits or vegetables within a single frame – a necessary capability for real-world deployment scenarios like conveyor-based sorting. In total, the dataset spans 120 classes, representing different combinations of fruits, vegetables, and quality labels.

All images were standardized to a resolution of 100 × 100 pixels to reduce computational complexity while preserving sufficient visual information for accurate analysis. This uniformity in image size allows models to process inputs consistently and improves the training efficiency across all modules. Overall, the Data Collection module ensures the availability of a rich, labeled, and diverse dataset that serves as the backbone for the AI models powering this system. It plays a critical role in enabling the automation of grading, classification, and quality assessment tasks that are traditionally manual, labor-intensive, and error-prone.

6.1.2 DATAPREPROCESSING

The Data Preprocessing Module plays a pivotal role in ensuring the quality and consistency of image data fed into the deep learning models used in this AI-driven fruit and vegetable grading system. Leveraging the FastAI library, built on top of PyTorch, the preprocessing pipeline was designed to be both efficient and scalable. FastAI is particularly well-suited for computer vision applications due to its high- level abstractions that simplify complex tasks such as image normalization, augmentation, and dataset management. In this project, the primary goal of preprocessing was to standardize the input data, enhance variability for better generalization, and streamline the integration with various model training workflows including classification, grading, and expiry prediction.

The preprocessing process began with automated reading and loading of images from structured folders, where the folder names themselves represented class labels, allowing FastAI to automatically infer the correct annotations for each image. All images were resized to a uniform dimension compatible with the input layer requirements of deep learning models, specifically those based on convolutional neural networks like ResNet. This resizing not only reduced computational overhead but also ensured consistency across the training samples. Following resizing, pixel values of the images were normalized—typically to a range between 0 and 1 or according to ImageNet mean and standard deviation values—which helps in stabilizing and accelerating the learning process by ensuring uniform feature scaling.

To further enhance the diversity of the training data, the preprocessing module employed a series of image transformation techniques, collectively referred to as data augmentation. These included random horizontal and vertical flipping, rotations, zooming, and lighting adjustments. Such augmentations introduce slight variations in the dataset without changing the underlying content, thus enabling the models to learn more robust features and reducing the risk of overfitting to training data. This approach simulates real-world variations in angle, lighting, and perspective that may occur in practical deployment environments, thereby making the system more resilient and reliable.

Another significant feature of FastAI utilized during preprocessing was its capability to automatically split the dataset into training and validation sets. This splitting ensures that the model is evaluated on unseen data during training, which helps in early detection of overfitting and provides a realistic measure of model performance. The preprocessing module also maintained the class balance to ensure that no particular class dominated the training process, which could otherwise bias the learning outcomes.

Furthermore, FastAI's seamless integration with pretrained PyTorch models enabled the application of transfer learning, wherein models pre-trained on large benchmark datasets like ImageNet were fine-tuned using the fruit and vegetable dataset. This significantly reduced the required training time while improving the model's accuracy and generalization capabilities, even with limited domain-specific data. Overall, the Data Preprocessing Module established a reliable and automated workflow that ensured high-quality input for all subsequent modules. By combining standardization, augmentation, and intelligent dataset management, this module lays the groundwork for effective and efficient deep learning model training in the system.

6.1.3 TRAINING THE MODEL FOR THE FOLLOWING

6.1.3.1 FRUITS AND VEGETABLES CLASSIFICATION

The Fruits and Vegetables Classification module serves as the foundational step in the AI-based grading system, enabling the system to accurately distinguish between different categories of fruits and vegetables. At the core of this module lies a Convolutional Neural Network (CNN), trained to learn the visual patterns and features unique to each produce type. Before training, each image undergoes a structured preprocessing routine—resizing to a consistent 100×100 pixel dimension, normalizing pixel values for uniformity, and applying a set of augmentation transformations such as flipping, rotating, and lighting variations. These augmentations are essential to enhance the model's robustness, allowing it to perform reliably under diverse real world conditions.

Training is carried out using a supervised learning approach, where a large dataset of manually labeled images—containing annotations like "Apple", "Grapes", "Carrot", and "Potato"—is fed into the CNN model. The network consists of several convolutional and pooling layers that progressively extract and compress spatial features relevant for classification. These features are passed through fully connected layers, and the final output layer uses the softmax activation function to assign a probability score to each category. The class with the highest probability is selected as the prediction. During testing, the model's performance is validated on unseen data to evaluate generalization, consistency, and accuracy. This phase is critical in ensuring the system's ability to classify inputs correctly when deployed in a real-time environment. The ultimate output of this module is a category label, such as "Apple" or "Carrot," which guides subsequent grading and analysis processes.

The rule-based approach uses a set of logical conditions derived from expert knowledge and observable attributes - such as size, color, shape, and texture—to categorize fruits and vegetables before feeding them into the CNN for training. By applying these explicit rules, the system ensures consistency and accuracy in labeling, which is crucial for building a reliable supervised learning model. The detailed conditions and thresholds used for this rule-based classification are outlined in the Table 6.3.1.1, guided the initial annotation process and helped maintain high-quality labeled data for effective model training.

Class	**Dominant Color (HSV Range)**	**Shape Feature**	**Rule Summary**
Apple	Red shades [0-10] or [160-180] hue	Nearly circular (aspect ratio ≈ 1.0)	If red color dominates and shape is round → Apple
Grapes	Purple/Green [35-85] hue (green) or [125-155] (purple)	Small & round clusters	If clustered small circular regions in green/purple → Grapes
Carrot	Orange [10-25] hue	Long vertical (aspect ratio > 2.5)	If orange color and long shape → Carrot
Potato	Brown [10-20] hue (light brown), dull saturation	Oval, irregular	Brownish + oval + dull color (low saturation) → Potato

Table.6.1.3.1: Rule Based Classifier for Classification

6.1.3.2 GRADE PREDICTION

The Grade Prediction module is designed to evaluate the overall visual quality of fruits and vegetables using deep learning techniques, assigning each item to a specific quality grade—typically Grade A for ideal produce and Grade B or C for those with minor defects. This process builds on the foundational classification step and extends the capability of the AI system by incorporating finer quality metrics such as surface defects, texture consistency, and color uniformity. Each fruit and vegetable image used for training is pre-labeled with corresponding grade categories, where Grade A signifies flawless items, and lower grades represent visible blemishes, irregular shapes, or discoloration.

A Convolutional Neural Network (CNN) is employed to extract relevant surface- level features from the input images. These features include texture irregularities, bruising patterns, and color tone deviations—factors critical for visual grading in agricultural contexts. The training process is structured using a supervised learning pipeline, where images with known grade labels guide the learning of feature weight associations. The dataset is split into training and validation subsets to ensure the model generalizes well and avoids overfitting. Evaluation metrics such as precision, recall, and F1-score are used to quantify performance, especially since grading involves subtle visual distinctions that may introduce class imbalances.

During inference, a multi-class classification head processes the extracted features and produces a confidence vector representing the likelihood of each grade. The model then assigns the grade with the highest confidence as the final output. This automated grading ensures consistent, objective, and scalable quality assessment for post-harvest management.

The rule-based approach uses a set of logical conditions derived from expert knowledge and observable attributes – such as, texture contrast, color variance, defect count, shape regularity, size uniformity, mean brightness - to categorize fruits and vegetables before feeding them into the CNN for training.

By applying these explicit rules, the system ensures consistency and accuracy in labeling, which is crucial for building a reliable supervised learning model. The detailed conditions and thresholds used for this rule-based classification are outlined in the Table 6.3.1.2, guided the initial annotation process and helped maintain high-quality labeled data for effective model training.

Grade	Texture Contrast (GLCM)	Color Reference (HSV)	Defect Count	Shape Regularity	Size Uniformity	Mean Brightness
A	< 2.0	< 20	≤ 2	0.9 – 1.1	0.8 – 1.2	100 – 155
B	< 4.0	< 40	≤ 5	0.8 – 1.3	0.7 – 1.3	80 – 180

Table.6.1.3.2: Rule Based Classifier for Grading

6.1.3.3 EXPIRY DATE PREDICTION

The Expiry Date Prediction module plays a critical role in enhancing the post-harvest handling and logistics of fruits and vegetables by estimating the remaining shelf life based on visual changes over time. This is achieved using a deep learning framework that integrates Convolutional Neural Networks (CNNs) for spatial feature extraction and Long Short-Term Memory (LSTM) networks for time-series modeling. The key idea is that certain visual indicators—such as changes in color saturation, emergence of spots, texture fading, or shrinkage can be observed across several days and used to forecast when the produce will no longer be fit for consumption.

To enable this prediction, the dataset is prepared by capturing images of the same fruit or vegetable item over consecutive days. These temporally ordered images are processed using CNN encoders that extract high-level feature vectors representing the current visual state. These sequential features are then fed into LSTM layers, which are adept at modeling temporal dependencies and trends. Through this architecture, the model learns how visual deterioration progresses over time and how it correlates with actual shelf life, which is already known for the training samples.

The training process maps time-sequenced image features to shelf life duration, enabling the model to predict remaining days to expiry during inference. This helps stakeholders make better decisions on storage and logistics, reducing food waste. Table 6.1.3.3 presents rule-based classifiers used in the Expiry Prediction module.

Label	**Color Saturation**	**Texture Sharpness**	**Surface Spots**	**Wrinkle Intensity**	**Glossiness**	**Edge Contrast**
> 5 days	High	Sharp	None	Low	Shiny	Strong
2-3 days	Moderate	Slightly Blurred	Few	Moderate	Dull	Fading
Expired	Low	Blurred	Many	High	Matte	Soft

Table.6.1.3.3: Rule Based Classifier for Expiry Prediction

6.1.3.4 CALCULATING NUTRITIONAL VALUE

The Calculating Nutritional Value module is designed to provide users with an accurate estimate of the nutritional content of various fruits and vegetables based on visual characteristics and classification results. A structured dataset is used, which maps specific food categories (e.g., "Apple Red Delicious," "Carrot Nantes") to their average nutritional values sourced from authoritative databases like the USDA and FSSAI. Once the system classifies the produce type using a CNN-based model, the information is passed to an Artificial Neural Network (ANN). This ANN is trained to refine nutritional estimates by analyzing visual features such as color intensity, surface texture, and size—factors that indicate ripeness and variety.

The ANN adjusts nutritional values accordingly, especially for produce with multiple subtypes or grades. The final output is presented in the form of a nutrition summary card, which includes key macronutrients - calories, carbohydrates, protein, fiber, and sugar - alongside optional vitamin and mineral percentages such as Vitamin A and C. The card can be customized to display values per 100 grams or a user defined weight, offering a practical and user-friendly insight. This module empowers users to assess dietary content quickly and supports healthy food selection and meal planning .

6.1.3.5 DISEASE PREDICTION

The Disease Prediction module plays a crucial role in identifying and diagnosing common diseases in fruits and vegetables through image-based deep learning. It is designed to detect visual symptoms such as lesions, mold, or discoloration that indicate diseases like bacterial spot, common scab, or powdery mildew. The system uses specialized Convolutional Neural Networks (CNNs) or object detection models like YOLO (You Only Look Once), which can localize and classify disease affected regions in real-time.

The process begins when the user captures or uploads an image of a fruit or vegetable through the mobile application. This image undergoes preprocessing steps, including resizing, normalization, and real-time augmentation techniques like flipping, zooming, or brightness adjustment, which simulate variable environmental conditions during image capture.

The model is trained on a curated dataset containing labeled examples of healthy and diseased produce. In the case of object detection using YOLO, disease- affected areas in the training images are annotated using bounding boxes or segmentation masks. These annotations enable the model to not only classify the disease but also accurately localize the infected regions within the image. During inference, the model processes the input image, identifies disease symptoms, and outputs both a classification label (e.g., "Powdery Mildew") and a visual indicator. In YOLO-based deployments, this includes bounding boxes drawn directly on the image to show affected areas. Early and automated disease detection significantly reduces crop loss, minimizes manual inspection, and supports timely intervention, ultimately enhancing agricultural productivity and reducing post-harvest losses. The rules which are specifically used to train the model is mentioned in the Table 6.1.3.4 to define the thresholds.

Disease	Spot Color	Spot Density	Color Deviation	Affected Area	Moisture Gloss
Carrot Rot	Dark brown / black	High (≥ 15)	High (orange → brown/black	>60%	Wet/Glossy
Grape Rot	Gray-brown	Medium (10–12)	High (purple → dull gray)	50–70%	Moist surface
Apple Rot	Black / dark brown	High (>15)	High (red/green → brown/black)	60–80%	Glossy/soft
Apple Scab	Olive-brown	Moderate (8–12)	Moderate (green → olive/brown)	30–50%	Dry
Potato Scurf	Dark brown	Medium (10–14)	Medium (light brown → dark brown)	40–60%	Dry surface

Table.6.1.3.4: Rule Based Classifier for Disease Detection

6.1.4 INTEGRATING WITH MIT APP INVENTOR

The Integration with MIT App Inventor module is a pivotal component of the AI-based fruit and vegetable grading system, providing a user-friendly interface that brings the power of deep learning to the hands of farmers, vendors, and consumers. This module involves the development of an Android mobile application using MIT App Inventor, an intuitive, block-based programming platform that allows rapid app development without requiring advanced coding expertise. The mobile app is designed to allow users to capture images of fruits and vegetables in real time using their smartphone camera and instantly receive insights related to classification, quality grade, shelf life estimation, nutritional value, and disease detection.

The core functionality of the application lies in its seamless interaction with the trained AI models. These models, developed using CNNs, LSTMs, and ANNs in earlier modules, are either deployed locally on the device for fast offline inference or hosted on the cloud for enhanced processing power and scalability. The app sends the captured image to the backend where the appropriate model processes it and returns the results. For instance, once an image of an apple is taken, the classification model identifies the fruit type, the grading model determines whether it is Grade A or B based on surface quality, the expiry module forecasts the number of days it can remain fresh, the nutrition module estimates calorie and vitamin content, and the disease detection model flags any signs of infection.

MIT App Inventor's components such as the Web component (for connecting with online APIs), ImagePicker, Camera, and Notifier are used to design this interactive flow. Trained models can be integrated using TensorFlow Lite or accessed via REST APIs hosted on a cloud server. The application provides real-time visual feedback, including labels, nutrition charts, and even annotated images showing disease affected regions if detected.

6.1.5 INTERPRETING RESULTS

The Interpreting Results module serves as the final and user-facing component of the AI-enabled fruit and vegetable analysis system. It plays a crucial role in transforming complex machine learning outputs into meaningful, actionable insights for end-users. After the trained deep learning models have processed the input images—whether for classification, grading, expiry prediction, nutritional value estimation, or disease detection—the results are displayed within the mobile application in a simple, intuitive, and visually clear format. This ensures that users, regardless of their technical background, can understand and make use of the information provided.

When an image is captured or uploaded through the app, the system initiates a sequence of operations based on the integrated modules. First, the fruit or vegetable type is identified using the classification model. Next, the quality grade (e.g., Grade A or B) is determined by evaluating surface features such as color uniformity, size, and presence of defects. Simultaneously, the expiry prediction module estimates the number of days the item will remain fresh by analyzing time-based changes in color and texture. The nutritional module then presents a data card summarizing estimated values for calories, protein, fiber, sugar, and essential vitamins, calculated per 100 grams or user-defined quantity. Finally, the disease detection module, if applicable, highlights areas on the produce image where symptoms such as spots, mold, or discoloration are found, and labels the suspected disease type (e.g., bacterial blight or powdery mildew).

These results are consolidated into a dashboard-style interface within the mobile application. Icons, color-coded indicators, and concise labels are used to enhance clarity and speed of understanding. For example, green may indicate Grade A produce, while red highlights disease-affected items

The system also offers recommendations such as "Consume within 3 days" or "Possible fungal infection detected—avoid sale," helping users make smarter decisions.

By delivering this detailed yet digestible feedback, the interpreting results module empowers farmers, wholesalers, and consumers to take informed actions in real time whether it's sorting fruits for market, adjusting prices based on quality, or separating spoiled items to prevent disease spread.

CHAPTER 7

TESTING

7.1 SYSTEM TESTING

In this project, system testing refers to the process of testing the entire fruit and vegetable classification in the android application as a complete and integrated system. After all components - such as the camera interface, image preprocessing, machine learning model, and result display are developed and integrated, system testing is conducted to ensure everything works together as intended.

This type of testing verifies that the user can capture an image, the image is processed correctly by the embedded model, and the final classification result is displayed accurately within the app. It is performed without focusing on the internal code or algorithms, treating the app as a "black box." Instead, the testing focuses on inputs (captured images) and outputs (predicted labels), ensuring the app behaves correctly under various usage conditions.

System testing in this project helps confirm that the application meets its functional requirements, is stable, and performs accurately across different devices and environments. It is a crucial step before deployment to ensure that the final product is reliable, user-friendly, and ready for real-world agricultural use.

7.2 TYPES OF SYSTEM TESTING

Types of system testing are applied to ensure that the Android-based image classification application functions properly and meets user requirements. The testing carried out are functional testing, usability testing, performance testing, compatibility testing, integration testing and regression testing.

7.2.1 FUNCTIONAL TESTING

Functional testing focuses on verifying that each function of the android application works according to the defined specifications. In this project, it includes checking whether the camera opens and captures images properly, whether the image is processed by the embedded machine learning model, and whether the classification result is displayed accurately on the screen. It also ensures that buttons, UI components, and image input/output flows work as intended. This testing does not consider how the feature is implemented internally—it simply checks that the expected outputs are achieved from given inputs. Functional testing ensures the basic reliability and correctness of the app's features before moving to more complex tests.

7.2.2 USABILITY TESTING :

Usability testing evaluates how user-friendly and accessible the application is, especially for its target users—farmers and individuals in agriculture who may have limited technical skills. This test focuses on how intuitive the interface is, how easily users can navigate through the app, and how well the design supports task completion (such as capturing and classifying an image). Feedback is often gathered from actual users to identify areas for improvement in design, layout, text clarity, button placement, and overall experience. The goal is to make the app simple, efficient, and comfortable for everyday use in real farming environments.

7.2.3 PERFORMANCE TESTING :

Performance testing assesses how well the application performs under various conditions. For this project, it evaluates the speed of the image classification process, how quickly the camera captures and uploads an image, and how fast the model returns a result. It also checks the app's responsiveness during continuous usage and its ability to handle resource limitations, such as limited memory or battery. Key performance indicators include response time, load handling, and resource consumption (CPU/RAM). This ensures that the application works smoothly on a range of devices, including mid-range Android phones commonly used in rural areas.

7.2.4 COMPATIBILITY TESTING :

Compatibility testing ensures that the app runs reliably on different Android devices, screen sizes, and OS versions. Since Android devices vary in hardware and software, this testing verifies that the app adapts well across multiple environments. It includes testing the UI on different screen resolutions, checking if functions like the camera and storage access work on various devices, and ensuring the model loads properly regardless of the device specifications. This type of testing is essential for broader adoption, especially since the app is meant for rural and agricultural users who may use older or lower-end phones.

7.2.5 INTEGRATION TESTING:

Integration testing examines whether different parts of the system work together as expected. In this project, it involves checking the connection between the Android application and the machine learning model. When a user captures an image, the app must properly send it to the model for processing, and then receive and display the classification output. This type of testing ensures there are no breakdowns in the flow of data and that the transitions between components are smooth. It helps catch issues like incorrect image format, failed model response, or errors in displaying results, which could otherwise go unnoticed in isolated tests.

7.2.6 REGRESSION TESTING:

Regression testing is performed whenever the system undergoes changes - such as updates to the machine learning model, UI improvements, or bug fixes. The purpose is to make sure that these changes don’t unintentionally break or alter existing features that previously worked correctly. For example, after improving model accuracy, regression testing would check whether the image capture still functions correctly, the app doesn't crash, and results are still shown properly. It ensures stability and reliability as the app evolves. This type of testing helps maintain quality over time and is a key part of continuous development.

CHAPTER 8

RESULT AND DISCUSSIONS

The developed system demonstrates the practical success of combining artificial intelligence with real-time applications in the agricultural domain. This project aimed to deliver an end-to-end intelligent solution for analyzing the condition and quality of fruits and vegetables by automating tasks that typically require human observation, expertise, and time. The results affirm that the system functions effectively and meets its intended purpose. Throughout the development, deep learning techniques were integrated into a single mobile-accessible framework.

The use of image-based AI models enabled the system to recognize various visual features such as color, texture, shape irregularities, surface damage, and visual degradation. These features were interpreted intelligently to provide meaningful outcomes like identification, grading, spoilage status, and disease detection. Additionally, non-visual information such as nutritional data was fetched from a trusted external source, ensuring factual reliability and enhancing the informational value of the system. The design ensured that the system required minimal user input. Once an image is captured, the backend processes handle all tasks, from image preprocessing and feature extraction to prediction and display of results.

The system was tested with different types of produce under varying visual conditions and maintained consistent behavior and decision-making accuracy. The ability to generalize across diverse data inputs highlights its robustness and adaptability. Efficiency was a key outcome of the project. The models were optimized for mobile deployment, ensuring quick prediction times without requiring high-end processing power. The Android interface, built using MIT App Inventor, proved effective in integrating the AI models while maintaining usability. From a user experience standpoint, the system provided instant, interpretable feedback without the need for manual grading, lab testing, or expert consultation. The project also underlined the broader need for such a solution. In many scenarios, post-harvest quality checks and sorting are still done manually,

which introduces subjectivity, inconsistency, and delays. This system automates those tasks with data-driven accuracy and consistency. Furthermore, by including functionalities like expiry estimation and disease detection, the system supports better storage decisions, reduces the chance of food spoilage, and can ultimately contribute to minimizing post-harvest losses. Overall, the system demonstrated technical soundness, practical utility, and user-friendliness. It reflects the potential of AI to assist in real-world problems by enhancing speed, reliability, and awareness in produce handling. The integration of different models and tools into one cohesive mobile solution makes this project an effective blueprint for future intelligent agricultural systems. With further expansion of datasets and real-time connectivity, the system can scale to support broader agricultural ecosystems and markets.

SAMPLE SCREENSHOTS

HOME PAGE

Fig 8.1: Home Screen

CLASSIFICATION

Fig 8.2: Classification Module Page

GRADE PREDICTION

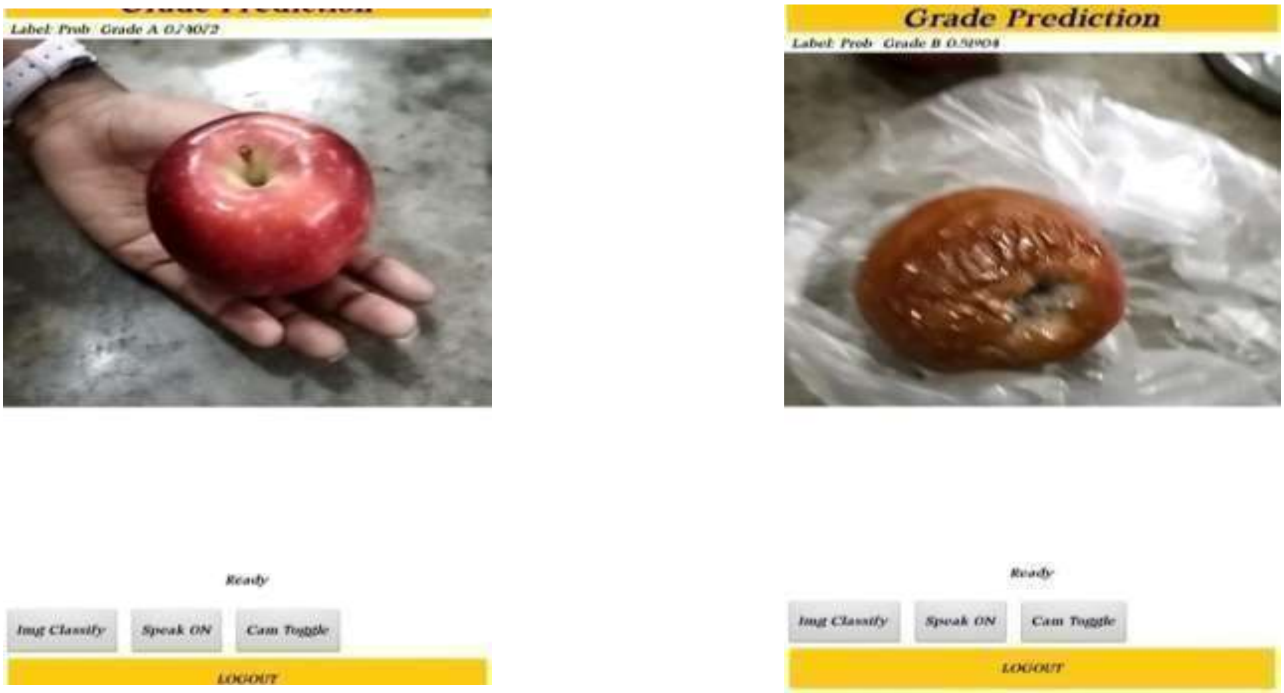


Fig 8.3: Grading Module Page

EXPIRY PREDICTION

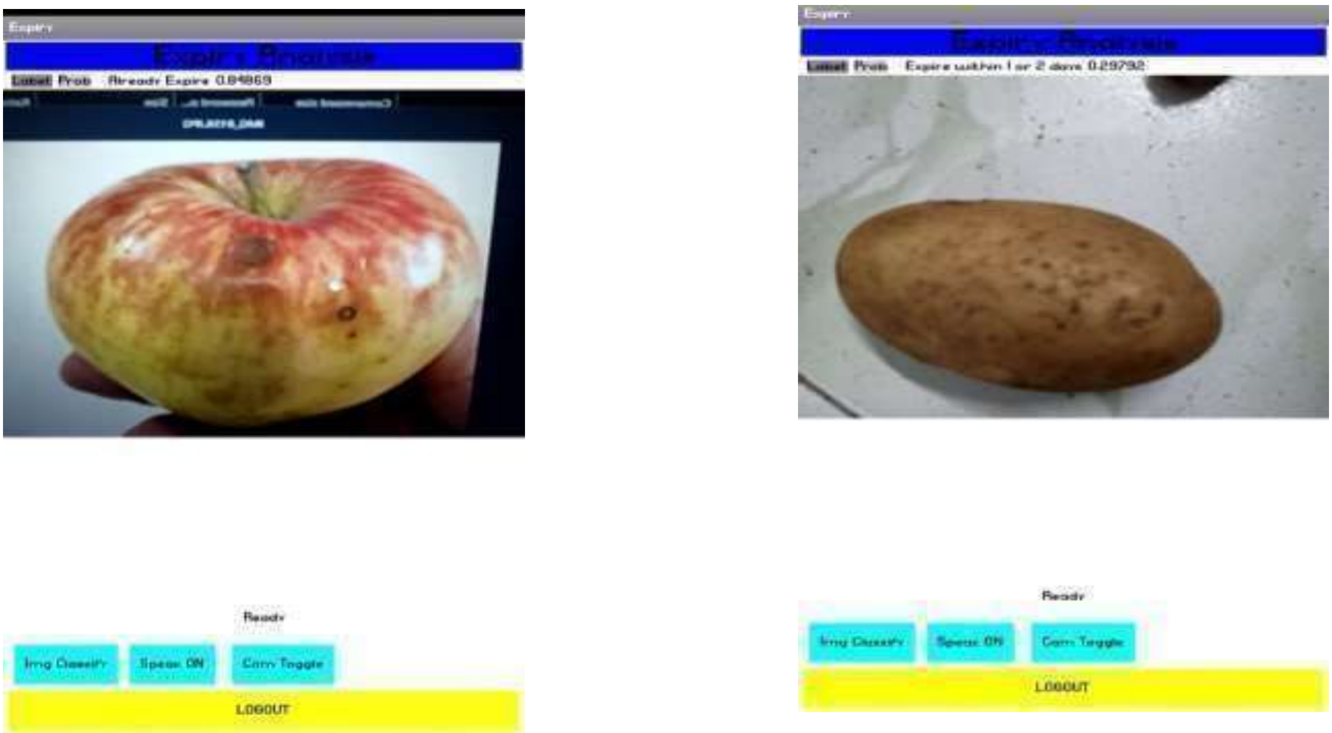


Fig 8.4: Expiry Prediction Module Page

DISEASE PREDICTION

Fig 8.5: Disease Detection Module Page

NUTRIENT ANALYSIS

Fig 8.6: Nutrient Analysis Module Page

CHAPTER 9

APPENDIX

9.1 SOURCE CODE

9.1.1 CNN CODE

CLASSIFICATION

```
from keras.models import load_model
# TensorFlow is required for Keras to work
from PIL import Image, ImageOps
# Install pillow instead of PIL import numpy as np
# Disable scientific notation for clarity
np.set_printoptions(suppress=True)
# Load the model
model    = load_model("keras_Model.h5", compile=False)
# Load the labels
class_names = open("labels.txt", "r").readlines()
# Create the array of the right shape to feed into the keras model data = np.ndarray(shape=(1,
224, 224, 3),
dtype = np.float32) # Replace this with the path to your image
image = Image.open("<IMAGE_PATH>").convert("RGB")
image = ImageOps.fit(image, size, Image.Resampling.LANCZOS)
#  turn  the  image  into  a  numpy  array

image_array = np.asarray(image) # Normalize the image normalized_image_array
(image_array.astype(np.float32) /127.5) - 1
# Load the image into the array
data[0] = normalized_image_array # Predicts the model prediction =
model.predict(data)
index = np.argmax(prediction)
class_name =class_names[index]
```

confidence_score = prediction[0][index] # Print prediction and confidence score

print("Class:", class_name[2:], end="")

print("Confidence Score:", confidence_score)

GRADING, EXPIRY PREDICTION & DISEASE DETECTION

from tensorflow.keras.models import load_model

TensorFlow is required for Keras to work

from PIL import Image, ImageOps # Install pillow instead of PIL

import numpy as np # Disable scientific notation for clarity

np.set_printoptions(suppress=True)

Load the required model – if : Grading Module

model = load_model("grading_Model.h5", compile=False)

Load the labels class_names = open("labels.txt", "r").readlines()

Load the required model – if : Expiry Module

model = load_model("grading_Model.h5", compile=False)

Load the labels class_names = open("labels.txt", "r").readlines()

Load the required model – if : Disease Detection Module

model = load_model("grading_Model.h5", compile=False)

Load the labels class_names = open("labels.txt", "r").readlines()

data = np.ndarray(shape=(1, 224, 224, 3), dtype=np.float32) #
Replace this with the path to your image

image = Image.open("<IMAGE_PATH>").convert("RGB")

image = ImageOps.fit(image, size, Image.Resampling.LANCZOS)

turn the image into a numpy array

image_array = np.asarray(image) # Normalize the image normalized_image_array

= (image_array.astype(np.float32) / 127.5) - 1

```
# Load the image into the array
data[0] = normalized_image_array

# Predicts the model
Prediction =    model.predict(data)

Index =    np.argmax(prediction)
class_name = class_names[index]
confidence_score = prediction[0][index] #
Print prediction and confidence score

print("Class:", class_name[2:], end="")
print("Confidence Score:", confidence_score)
from keras.models import load_model # TensorFlow is required for Keras to work
import cv2 # Install opencv-python
import numpy as np
# Disable scientific notation for clarity
np.set_printoptions(suppress=True)
# Load the model : Grading Model
model = load_model("grading_Model.h5", compile=False)
# Load the model : Expiry Model
model = load_model("expiry_Model.h5", compile=False)
# Load the model : Disease Model
model = load_model("disease_Model.h5", compile=False)
# Load the labels
class_names = open("labels.txt", "r").readlines()
# CAMERA can be 0 or 1 based on default camera of your computer camera =
cv2.VideoCapture(0)
while True:
# Grab the webcamera's image. ret, image = camera.read() #
Resize the raw image into (224-height,224-width) pixels
Image = cv2.resize(image,(224, 224), interpolation=cv2.INTER_AREA) # Show
```

```
the image in a window
cv2.imshow("Webcam Image", image)
# Make the image a numpy array and reshape it to the models input
shape.
image = np.asarray(image, dtype=np.float32).reshape(1, 224, 224, 3)
# Normalize the image array image =
(image / 127.5) - 1
# Predicts the model
Prediction = model.predict(image)  index =
np.argmax(prediction) class_name =
class_names[index] confidence_score =
prediction[0][index] # Print prediction and
confidence score print("Class:",
class_name[2:], end="")
print("Confidence Score:", str(np.round(confidence_score * 100))[:-2], "%")

# Listen to the keyboard for presses. keyboard_input =
cv2.waitKey(1)
if keyboard_input == 27:
break
camera.releas
e()
cv2.destroyAllWindows()
```

9.1.2 ANDROID CODE

```
<?xml version="1.0" encoding="utf-8"?>

<manifest

xmlns:android="http://schemas.android.com/apk/res/android"

android:versionCode="1" android:versionName="1.0"

android:compileSdkVersion="34"

android:compileSdkVersionCodename="14"

package="appinventor.ai_shakthikann.fruits_vs_vegies_1

platformBuildVersionCode="34"

platformBuildVersionName="14">

<uses-permission  android:name="android.permission.INTERNET"  />

<uses-permission android:name="android.permission.CAMERA " />

<uses-sdk android:minSdkVersion="7" android:targetSdkVersion="3 4" />

<application android:theme="@ref/0x7f09000b"

android:label="fruits_vs_veggie"

android:icon="@ref/0x7f03000"

android:name="com.google.appinventor.components.runtime.multidex.MultiDex

Application" android:debuggable="false"

android:networkSecurityConfig="@ref/0x7f060000"

android:roundIcon="@ref/0x7f030000"

android:requestLegacyExternalStorage="true"

android:preserveLegacyExternalStorage="true">

<uses-library android:name="org.apache.http.legacy"

android:required="false" />

<activity android:name=".Screen1" android:exported="true"

android:screenOrientation="1"
```

```
android:configChanges="0xdb0"  android:windowSoftInputMode="0x2">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCH ER" />

</intent-filter>

</activity>

<activity android:name="appinventor.ai_shakthikann.fruits_vs_veggies_1.Disease"

android:exported="true"

android:screenOrientation="-1" android:configChanges="0xdb0"
android:windowSoftInputMode="0x2">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

</intent-filter>

</activity>

<activity

android:name="appinventor.ai_shakthikann.fruits_vs_veggies_1.ME NU"
android:exported="true"

android:screenOrientation="2" android:configChanges="0xdb0"

android:windowSoftInputMode="0x2">

<intent-filter>

<action android:name="android.intent.action.MAIN" />
```

```
</intent-filter>
</activity>
<activity android:name="appinventor.ai_shakthikann.fruits_vs_veggies_1.SPOILAGE"
android:exported="true"
android:screenOrientation="-1" android:configChanges="0xdb0"
android:windowSoftInputMode="0x2">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
</intent-filter>
</activity>
<activity android:name="appinventor.ai_shakthikann.fruits_vs_veggies_1.Grade"
android:exported="true" android:screenOrientation="-1" android:configChanges="0xdb0"
android:windowSoftInputMode="0x2">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
</intent-filter>
</activity>
<activity android:name="appinventor.ai_shakthikann.fruits_vs_veggies_1.Expiry"
android:exported="true" android:screenOrientation="-1"
android:configChanges="0xdb0" android:windowSoftInputMode="0x2">
<intent-filter>
<action android:name="android.intent.action.MAIN" />
</intent-filter>
</activity>
<provider android:name="androidx.core.content.FilePro vider"
android:exported="false"
android:authorities="appinventor.ai_shakthikann.fruits_vs_veggies_1.provider"
android:grantUriPermissions="true">
<meta-data android:name="android.support.FILE_PROVIDER_PATHS"
android:resource="@ref/0x7f060001" />
</provider>
</application>
</manifest>
```

CHAPTER 10

CONCLUSION AND FUTURE ENHANCEMENT

10.1 CONCLUSION

The AI-based automation system developed has successfully demonstrates the capability of deep learning in optimizing post-harvest fruit and vegetable management. The system leverages advanced techniques including CNNs, LSTMs, and ANNs to deliver high accuracy insights from over 82,000 labeled images. FastAI and PyTorch frameworks streamlined preprocessing and training, while the integration with MIT App Inventor enabled real-time mobile-based interaction, making the solution practical and user friendly. By automating the grading and analysis process, the system supports efficient sorting, pricing, and decision making - benefiting farmers, distributors, and consumers alike while minimizing waste and maximizing produce quality.

10.2 FUTURE ENHANCEMENT

To further advance the system, future enhancements may include expanding support for additional fruit and vegetable varieties and incorporating seasonal datasets for improved generalization. Edge computing and federated learning can be explored to make the system operable offline or in remote rural areas. Additionally, integrating real-time sensor data (temperature, humidity, etc.) with vision data could enhance shelf-life and disease predictions. Blockchain-based traceability can provide secure and transparent supply chain records. Lastly, integrating a conveyor belt mechanism tailored for large-scale operations - with features such as modular design, adjustable speed control, automated sorting, and real-time fault detection can significantly enhance efficiency and scalability of the system.

REFERENCES

1. Ahatsham Hayat, Fernando Morgado-Dias, Tanupriya Choudhury, Thipendra P. Singh, and Ketan Kotecha.(2024) 'FruitVision: A deep learning based automatic fruit grading system' - Open Agriculture.

2. Anuja Bhargava and Atul Bansal.(2022) 'Classification and Grading of Multiple Varieties of Apple Fruit' - Springer.

3. Anuja Bhargava, Atul Bansal, and Vishal Goyal.(2021) 'Machine Learning Based Detection and Sorting of Multiple Vegetables and Fruits' - Springer.

4. Anuja Bhargava and Atul Bansal.(2019) 'Automatic Detection and Grading of Multiple Fruits by Machine Learning' - Springer.

5. Gang Xue, Shifeng Liu, and Yicao Ma.(2020) 'A hybrid deep learning-based fruit classification using attention model and convolution encoder' - Springer.

6. Jinu Lilly Joseph, Veena A. Kumar, and Santhosh P. Mathew.(2021) 'Fruit Classification using deep learning' - Springer.

7. M.T. Vasumathi and Kamarasan Mari.(2021) 'An Effective Pomegranate Fruit Classification Based On CNN-LSTM Deep Learning Models' - ResearchGate.

8. Nazrul Ismail and Owais A. Malik.(2024) 'Real-time visual inspection system for grading fruits using computer vision and deep learning techniques' - ScienceDirect.

9. Sadaf Iftikhar, Hasan Ali Khattak, Ahsan Saadat, Zoobia Ameer, and Muhammad Zakarya.(2024) ‘Efficient fruit disease diagnosis on resource constrained agriculture devices’ - ScienceDirect.

10. Zeshan Aslam Khan, Muhammad Waqar, Khalid Mehmood Cheema, Ali Abu Bakar Mahmood, Quratul Ain, Naveed Ishtiaq Chaudhary, Abdullah Alshehri, Sultan S. Alshamrani, and Muhammad Asif Zahoor Raja.(2024) ‘EA-CNN: Enhanced attention-CNN with explainable AI for fruit and vegetable classification’ - ScienceDirect

CONFERENCE CERTIFICATES


CHRIST THE KING
Engineering College
"We form you to transform"
NAAC

INTERNATIONAL CONFERENCE ON INNOVATIVE RESEARCH IN ENGINEERING SCIENCES (ICIRES-2K25)

CERTIFICATE

B.M.Nithyashri, CSE Department HAS PRESENTED A PAPER ON AI BASED AUTOMATION SOLUTION TO FOSTER GRADING OF VEGETABLES AND FRUITS TO SUPPORT UZHAVAN LOGISTIC PLANNING FROM KINGS COLLEGE OF ENGINEERING IN THE SECOND INTERNATIONAL CONFERENCE ON INNOVATIVE RESEARCH IN ENGINEERING SCIENCES AT CHRIST THE KING ENGINEERING COLLEGE ON 23.04.2025.

Dr. S. Dinesh Kirupha
Convenor

Dr. N. R. Gayathiri
Convenor

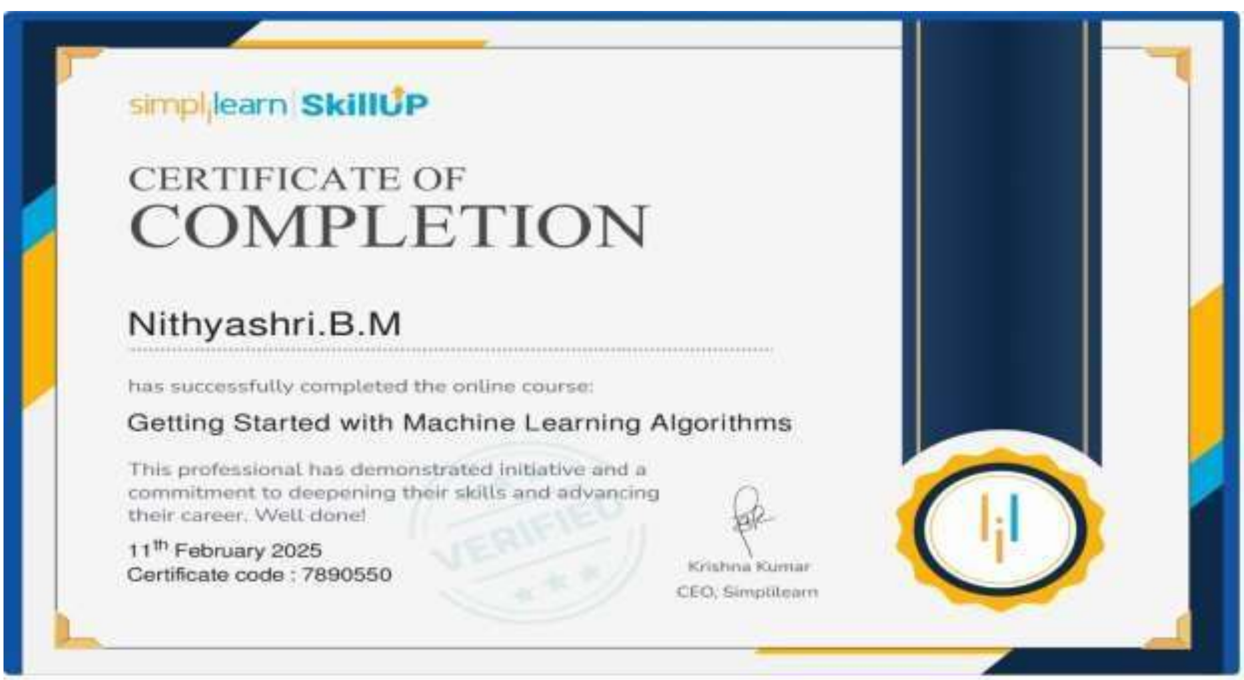
Dr. M. Jeyakumar
Principal

COURSE COMPLETION CERTIFICATES


simplilearn SkillUP
CERTIFICATE OF
COMPLETION
Keerthana J
has successfully completed the online course:
Machine Learning
This professional has demonstrated initiative and a
commitment to deepening their skills and advancing
their career. Well done!
11th February 2025
Certificate code :7890983
VERIFIED
Krishna Kumar
CEO, Simplilearn


simplilearn SkillUP
CERTIFICATE OF
COMPLETION
Keerthana J
has successfully completed the online course:
Getting Started with Machine Learning Algorithms
This professional has demonstrated initiative and a
commitment to deepening their skills and advancing
their career. Well done!
11th February 2025
Certificate code : 7891120
VERIFIED
Krishna Kumar
CEO, Simplilearn


simplilearn SkillUP
CERTIFICATE OF
COMPLETION
Nithyashri.B.M
has successfully completed the online course:
Machine Learning
This professional has demonstrated initiative and a commitment to deepening their skills and advancing their career. Well done!
7th February 2025
Certificate code :7877787
VERIFIED
Krishna Kumar
CEO, Simplilearn


simplilearn SkillUP
CERTIFICATE OF
COMPLETION
Nithyashri.B.M
has successfully completed the online course:
Getting Started with Machine Learning Algorithms
This professional has demonstrated initiative and a commitment to deepening their skills and advancing their career. Well done!
11th February 2025
Certificate code : 7890550
VERIFIED
Krishna Kumar
CEO, Simplilearn

FIELD VISIT CERTIFICATES

Thanjavur Food Museum

FIELD VISIT CERTIFICATE

This is to certify that the following students from **Kings College of Engineering**, namely : **Nithyashri, Aandal, Keerthana** all from the IV year, visited the **Thanjavur Food Museum** on **06/02/2025** as part of their project titled **"AI-Based Automation Solution for Grading Vegetables and Fruits for Uzhavan Logistic Planning".**

During their visit, they demonstrated keen interest in understanding the various aspects of **food processing, preservation, and logistics**. They actively interacted with our museum staff and gathered valuable insights that will aid in the successful completion of their project.

We appreciate their enthusiasm and commitment to innovation and wish them the very best in their future endeavors.

Signature: K. Rohineswara Kumar

Name: K.ROHINESWARA KUMAR

Designation: DIVISIONAL MANAGER, FOOD CORPORATION OF INDIA, DIVISIONAL OFFICE / THANJAVUR

Date: 06/02/2025

Department of Horticulture and Plantation Crops, Thanjavur

FIELD VISIT CERTIFICATE

This is to certify that the following students from **Kings College of Engineering**, namely : **Nithyashri, Aandal, Keerthana** all from the IV year, visited the **Department of Horticulture and Plantation Crops**, Thanjavur District on 06/02/2025 as part of their project titled **"AI-Based Automation Solution for Grading Vegetables and Fruits for Uzhavan Logistic Planning"**.

During their visit, they demonstrated keen interest in understanding the various aspects of **horticulture, plantation crops, and post-harvest management.** They actively interacted with our department staff and gathered valuable insights that will aid in the successful completion of their project.

We appreciate their enthusiasm and commitment to innovation and wish them the very best in their future endeavors.

Signature: Deputy Director of Horticulture THANJAVUR - 613 001

Name : A. VENKATARAMAN

Designation : Deputy Director of Horticulture THANJAVUR - 613 001

Date : 06/02/2025