Seasonal Fruits Distribution System

BY

Name: A K M Shamsur Rahman

ID: 201-16-488

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computing & Information System

Supervised By

Md. Nasimul Kader

Assistant Professor

Department of CIS

Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

23, SEPTEMBER 2024

APPROVAL

This thesis titled **"Seasonal Fruits Distribution System"**, Submitted by **A.K.M Shamsur Rahman**, ID No: **201-16-488**, to the Department of Computing & Information System, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information System and approved as to its style and contents. The presentation has been held on 23-09-2024.

BOARD OF EXAMINERS

Md Sarwar Hossain Mollah — **Chairman**
Associate Professor and Head
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Md. Nasimul Kader — **Internal Examiner**
Assistant Professor
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Md. Mehedi Hassan — **Internal Examiner**
Lecturer
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Dr. Saifuddin Md. Tareeq — **External Examiner**
Professor
Department of Computer Science and Engineering
University of Dhaka, Dhaka

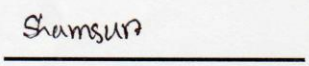
DECLARATION

I, A.K.M Shamsur Rahman, hereby declare that my work titled "Seasonal Fruits Distribution System" has been conducted under the supervision of, **Md. Nasimul Kader, Assistant Professor** Department of Computing & Information System (CIS) of Daffodil International University. This thesis work or any part of this work has not been submitted anywhere for the award of any educational degree like B.Sc., M.Sc., Diploma, or other qualifications.

SUPERVISED BY

Md. Nasimul Kader
Assistant Professor
Department of Computing & Information System
Daffodil International University, Bangladesh

SUBMITTED BY



A.K.M Shamsur Rahman
ID: 201-16-488
Department of Computing & Information System
Daffodil International University, Bangladesh

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty Allah for His divine blessing making us possible to complete the final year project successfully.

We are grateful and wish our profound indebtedness to**, Associate Professor, Department of CIS Daffodil International University, Dhaka.** The Deep knowledge & keen interest of my supervisor in the field of "Web development" helped to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express my heartiest gratitude to **___Head,** Department of CIS, for his kind help to finish my project and also to other faculty members and the staff of the CIS department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

The full results of my plan, which is a "seasonal fruit distribution system," are available here. The steps used to turn the concept into a functional website are covered in full in this article. For system users, one specific characteristic that stands out is the user dashboard. The economy

depends on agriculture; hence diseases of plants must be kept to a minimum. Although it is crucial to identify issues early on, inspecting by hand is labor-intensive, sluggish, and prone to mistakes. It also takes a lot of time. Fruit shape, texture, and color data may be extracted using artificial intelligence, which can help with virus diagnosis. In this project creating a web application for the webpage displays the company's basic details, a fruit list, and information on field leases. Additionally, the user may schedule a meeting in the field lease by choosing a location by acors and they can schedule a meeting to purchase fruits in bulk. In the web app user interface, a dataset of fruits will be used to analyze certain information provided by the user. Following the application of several models, an accuracy outcome will be displayed in the website. Also create a detection system to detect 4 class of data like: ”Rot_Apple”, “Scab_Apple”, “Normal_Apple” & “Blotch_Apple”. Applying CNN model to generate the disease detection through the web app. This project consists two dashboards like: User dashboard & Admin dashboard. The report includes the structure, style of the user interface, and technologies utilized in the building of the web application, covering every step from inception to execution. Python was employed for the backend. MySQL was utilized as the database system and JavaScript and ReactJS for the user interface. All you need to set up our system application is a regular desktop computer and internet connectivity; you don't need any expensive software or computer components.

TABLE OF CONTENTS

List of Tables

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Introduction

Systems are applications that display the connections and exchanges between several apps. Computers' "System" page contains tools for system administration, applications, and programming connections. Although the definition of "system" varies according on context, the general notion remains the same. The "Seasonal fruit distribution system" integrates several distinct technologies to provide a solid base. The several components that make up this architecture apply restrictions to each system that is being examined. There are several systems in each module. Through this web application, it offers a variety of services, including a detection system and a log-in system. Technology has a big impact on the agriculture industry, the companies that surround it, and the growth of farms. Recent technical advancements in agriculture have made it feasible to grow food even in arid regions. The industry with the most need for automation techniques is agriculture. Artificial intelligence (AI) is a big factor in many sectors, and agriculture is only one of them. AI may be used to tackle most typical agricultural problems. It aids in the early detection of some leaf diseases. By using automated apple leaf disease detection technology, farmers may reduce losses while increasing yields. Numerous researchers have worked on this problem. Most of them experimented with various deep learning, image analysis, and machine learning methods to diagnose agricultural ailments. It is economically critical to detect plant diseases early. Modern AI and machine learning techniques are able to detect and diagnose diseases at the earliest stages, which slows the progression of sickness and increases the likelihood of recovery. In many situations, illness symptoms become evident relatively early on, but in other circumstances, they do not show up until it is too late to treat the condition effectively [1]. Among other fruits, apples and cherries are susceptible to powdered mildew, rust, and black rot, which are frequently detected by hand examination or in a lab. Inspection by hands requires a lot of effort and time, is sluggish, and is prone to errors. Infection detection can be aided by using artificial intelligence to obtain fruit color, shape, or texture data. Annual fruit harvests exceed 374 million; 50–55% of these crops were cultivated in Asia in 2011 [2].

Customers' judgments are influenced by fruit look, which differs depending on where it is of sale (hawkers, shops, malls, etc.). The majority of fruits are hand-packed, which can lower quality and increase labor costs. Automated methods need to be quick, function well in low light, satisfy different customer needs, and handle novel fruit kinds. Poor quality fruit might be supplied to customers who would rather use it for pickling and juicing; the bad portions are cut off and the rest is cut into pieces or pressed. Fruits of superior quality fetch a fair price.

We need an automatic device that can identify different kinds and qualities of fruit.

CHAPTER 2

Initial Study

2.1 Project Proposal

Objectives

The main objective of this project is to create an easy-to-use web application that can accurately identify and classify fruits diseases based on specific symptoms or images. By applying AI techniques, we want to provide farmers with a trustworthy and efficient tool that might aid in the early detection of illnesses, reduce the possibility of monetary losses, and improve all types welfare.

- To utilize deep learning on all kinds of photos to forecast the four sorts of diseases associated with apple fruit.
- The aim of the compilation is to forecast fruit ailments.
- To lease any types of lands through this website.
- To deliver the fruits about booking fruits items.

Benefits of the website:

- Any client may schedule a meeting to accept a lease and purchase fruits for the entire sale.
- The entire fruit management system is displayed.
- AI-based fruit disease detection;
- System administration control

2.2 Background of the Project

Deep learning is the most notable development in machine learning. The subject of deep learning looks at how a computer software may pick up information via observation and make judgments based on that understanding. Computer vision, face identification, audio processing and recognition, and many more applications benefit from deep learning techniques. [9] proposed developing a reliable, trustworthy, and efficient system for

recognizing leaf illness with a focus on tomato plants by utilizing open-source algorithms, picture the process of segmentation and clustering to detect disease in tomato plant leaves. According to studies by Durmuş et al. [10], illnesses that are known to affect tomatoes produced in fields or greenhouses. Several illnesses in the leaves of tomato plants were identified using deep learning. The goal of the project was to make the robot perform algorithms based on deep learning in real time. Thus, either manually or automatically, the robot can identify plant diseases as it is moving across the field or within the greenhouse itself. Because deep learning technology is capable of understanding how to analyze images on its own, it can aid in the inspection and supervision of the fruit industry, especially in the early detection of diseased fruits. To stop the spread of disease fruit and preserve biosecurity, deep learning as well as driven artificial intelligence (AI) tools might be included into the statistical analysis process to detect, predict, and alert end users to odd conditions.

2.3 Problem Area

There are a number of possible issues to take into account while utilizing ReactJS to create a webpage for the detection of fruit illnesses in addition to leasing the land and scheduling the quantity of fruits. By taking care of these possible problems as soon as you build a Python detection website, you may enhance the customer experience and lower the possibility of major problems after launch. It appears that the primary objective of this study is to compile and evaluate each of those data sets. The data gathering process had been standardized and cleaned up using a variety of instruments and techniques. We were left waiting a long time to view the finished output because of the massive size about the data sets and the realization that they comprised many layers from different historical periods. There are multiple datasets available on this topic. I find it difficult to choose the best answers fast since we didn't complete the investigation, as every project demands a lot of effort from me. Preserving this massive quantity of data and preparing it for categorization presented another challenge. The challenge included applying deep learning models to categorize the constituent parts of diseases associated with apples in a web application.

2.4 Possible Solution

- User-friendly and uncomplicated elements on the website, along with a respectable user experience;
- Design that can adjust to several screen widths.
- I will demonstrate how research on apple fruit illness forms the basis for the study of other apple fruit disorders.
- Gaining a better understanding of how to use deep learning (DL) to identify diseased cartilage from images of sick apples;
- Comparing my findings with previous research using old picture data;
- Figuring out which CNN model employing DL is best suited for diagnosing different fruit illnesses based on available data.
- User can easily lease the land also booking fruits with the various typ
-

3-Lesson

3.1

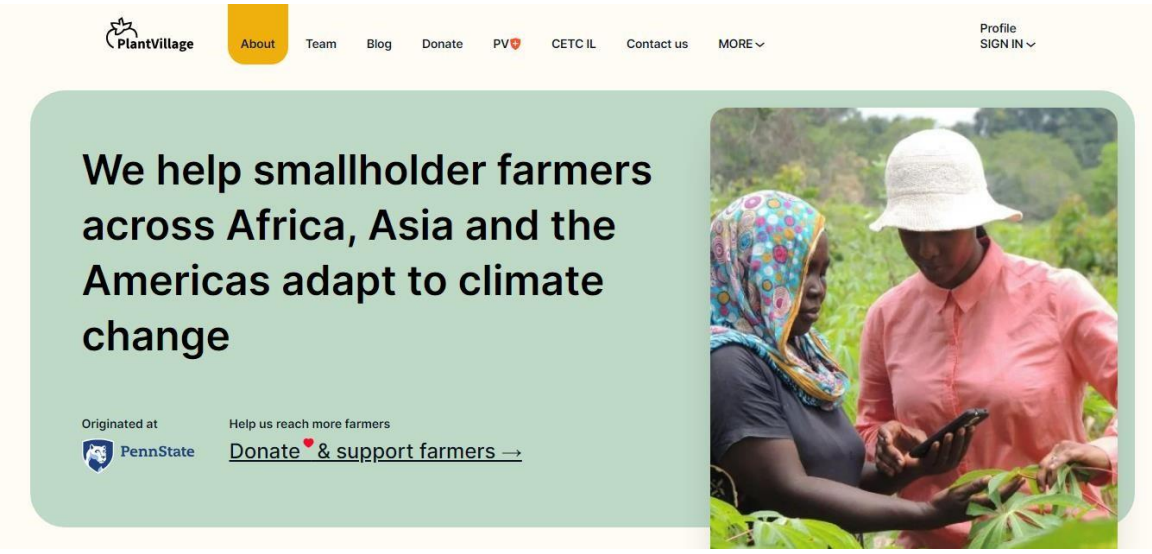
Deeper understanding of present issues and creative solutions may be obtained by examining the problem areas of fruit distribution and fruit disease detection systems using the prism of published research. The development of AI greatly benefits fruit distribution as well as fruit disease identification. Overcoming the particular issues noted in the latest research is essential to the implementation's success. Improving planning and logistics for fruit delivery can result in less waste and more effective operations. The accuracy and promptness of illness detection diagnostics may be improved by utilizing computer vision, enhancement of data, and environmental influences. It is probable that continuous development and research will enhance these structures further, making them more resilient, precise, and economical. Participants in the fruit sector may improve their operational management and lessen distribution and disease control difficulties by keeping up with these innovations.

3.2

When talking about AI-based solutions for detecting chicken diseases in online applications, it's important to highlight the many methods that have been investigated and the useful outcomes that may be obtained from them. The integration of artificial intelligence (AI) with online applications for the diagnosis of chicken diseases may greatly improve the precision and effectiveness of disease control, as demonstrated by published publications and case studies. Behavioral monitoring, predictive analytics, image-based diagnostics, and interaction with veterinary services are some of the solutions. The acceptance and efficacy of these technologies are further supported by an emphasis on user-focused design, confidentiality of information, and security. By putting these suggestions into practice, one may eventually increase the production and overall health of chickens by improving management techniques and early illness identification.

3.3 Comparison of leading solutions

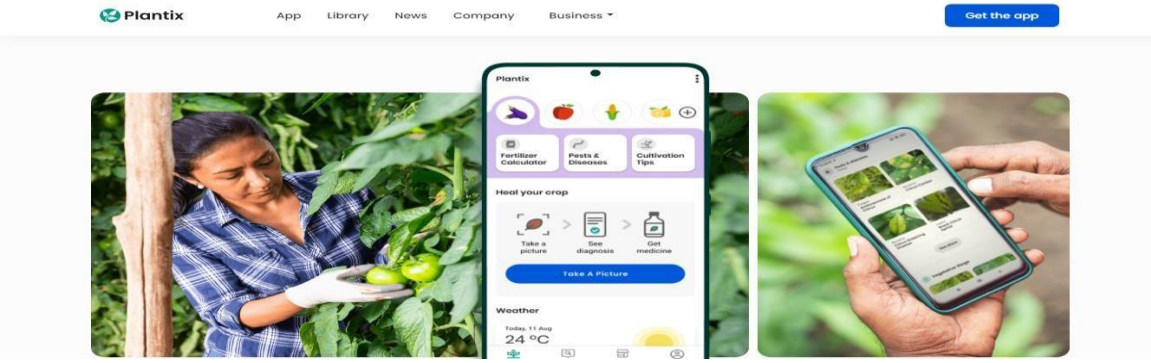
PlantVillage



Overview: PlantVillage analyzes photos taken by cellphones or drones to identify plant illnesses using artificial intelligence (AI) and computer vision. The deep learning models on the platform are trained using an extensive collection of plant photos.

Important characteristics:

- Real-time illness diagnosis via a mobile app.
- Vast information pertaining to plant illnesses.
- Support for a large number of illnesses and crops.
- Gives suggestions for managing the illness. **Plantix**



Overview: The Plantix smartphone & web app diagnoses plant problems and offers useful information by utilizing AI and machine learning. It offers therapy recommendations by fusing professional expertise with image recognition.

Principal Elements:

- Identifying diseases in real time using photos from smartphones.
- Offers advice on prevention measures and treatments.
- It has a platform for exchanging information throughout the community.
- Provides integration with experts in agriculture for supplementary assistance.

3.4 Recommended Approach

Table 1: Modules descriptions

Actuators	Functions
User	• User Dashboard • Sign In & Sign Up • Details about company • Fruit list and booking no of items. • Land list and lease the land with no of items. • Fruit disease Detection system.

Admin	• Admin Dashboard • View & Edit fruits and land item • View no of fruits booking items • View no of land lease

CHAPTER 4

Methodology

4.1 What to Use

For this project, a web application that shows a fruit list, the company's basic information, and specifics on field leasing is being created for the webpage. There are two dashboards in this project: an admin dashboard and a user dashboard. The user also has the option to arrange a meeting in the field lease by selecting a place via acors and to arrange a meeting for the bulk purchase of fruits. Additionally, develop an algorithm for detecting to identify four classes of data, such as "Blotch Apple," "Normal Apple," "Scab Apple," and "Rot Apple." use the CNN model to produce illness diagnosis via the web application. It is acknowledged that the SDLC life cycle paradigm—a thorough framework for expansion, design, etc.—is the best approach. I know a lot about different kinds of SDLC models. Software development employs a number of paradigms, including the Big Bang, the Spiral, waterfall, Agile, Continuous, and Adaptive System Architecture models. Every model offers a framework to direct the creation and implementation of the platform for fruit diseases components. This project's specific requirements for the stages of the SDLC in order to create an effective development process that aligns with the goals of disease detection using fruit disease photos.

4.2 Why to use

The development process started with the creation of the system architecture. This included figuring out the components and how they interacted. The architecture of the network infrastructure prioritized security, reliability, and scalability. It meant separating the back-end processes of the product, such as database administration, from the user's interface. The design also contained the required security mechanisms to ensure secure transactions and protect computer data. The agile methodology must be applied to all software projects. Feature-driven improvement, scrum, crystal, the dynamic system development approach, and kanban are among the agile methodology names that I am familiar with. But for my project, I used the DSDM methodology. There are several benefits to using the DSDM technique. The dynamic system method of development is used in iterative development, allowing for flexible requirement adjustments. When timely delivery is required, this strategy performs effectively. Delivering working software projects on a regular basis is a requirement of the Dynamic System Development Method. By reducing time to market and improving reporting on market movements, this agility approach allows the organization to employ important resources sooner. By integrating risk management techniques into its operations, this agile methodology assists team members in identifying potential threats early in the venture's lifecycle. Proactive risk management reduces the likelihood of project failure. For all of these reasons, I used the DSDM agile methodology while approaching my project.

4.3 Section of methodology

Numerous methods or strategies may be employed to decide how to evaluate the data that was used for this inquiry. In this study, a multi-step technique comprising model development, extension and improvement, data collection, and production was used.

Pre-Project Phase:

- **Feasibility Study**: This phase comprises evaluating the project concept's functional, financial, and infrastructure feasibility. It means striking a balance between the project's potential benefits, expenses, and dangers.

- **Conditions Collecting**: All of the requirements for the course were subsequently compiled and documented. Considering the constraints, demands, and business needs is necessary to determine the project's scope.

- **Planning:** Creating a strategic plan that outlines the objectives, timeline, required materials, and deliverables of the project is a part of planning. Establishing roles and responsibilities, identifying the project's stakeholders, and creating a collaborative and risk mitigation strategy are all crucial.

Project Lifecycle Phase:

- **Gathering Data:** Using Kaggle, I gathered and examined web statistics data to produce a reliable collection of my own. Due to the challenges of finding and acquiring data for the many apple fruits disease conditions, the four phases' images of the various illnesses, and other factors, a sizable, thorough dataset is not easily accessible in this industry.
- **Data preparing:** Following its acquisition in its data format from several industry vendors, every category of data was managed independently. Many data sets can have errors, especially ones that contain noise. In a technical sense, I use the chosen data set to process the information first before moving on with the next stage.
- **Data Preprocessing:** The results expanded and narrowed as each class was assessed. For it to function, I had to add details and change the size. Because I was worried about excessive fitting, I limited the number of alterations that I implemented to the largest and most acceptable ones.
- **Model Selection:** Once you've made your choice, use the available data to train and evaluate the model to improve accuracy. CNN models are used by DL. Several iterations of the idea were tried with my technology in order to identify the ideal setup for accurate data analysis.
- **Evaluation of Performance:** Every result is explained in this section. Following the initial testing and training, these strategies left us lacking sufficient reliability for our subsequent two courses. In order to identify different apple fruit diseases, they

created a web-based tool and created graphics for the f1 measurements, recall, performance, and confusion matrix.

- **Design:** The software design is constructed in this step by utilizing the requirements that have been collected. Designing databases, interfaces, and architectures are only a few of the complex and high-level tasks it involves.
- **Development**: In this step, the software is coded using its architectural criteria as a guide. To construct a functional software product, a developer develops the source code, runs unit tests, and assembles components.
- **Testing**: This process aims to verify the software's functionality and quality by running it throughout its paces. It includes a broad spectrum of testing techniques, including system, integration, unit, and user legitimacy testing.
- **Deployment**: After being accepted and going through a thorough testing process, the program is implemented. The program must be installed, configured, and properly set up in the proper environment.

Post-Project Phase:

- **Maintenance:** The program moves into the ongoing upkeep phase following deployment. To guarantee that the program keeps functioning and adapts to changing requirements, this phase entails continuous maintenance, bug patches, and upgrades.
- **Evaluation:** By comparing the project's actual results with its intended aims, one may assess the efficacy of the endeavor. It assists in identifying areas that need to be changed and lessons learnt for projects that come after.
- **Closure:** This marks the official end of the project. It include completing the project's paperwork, monitoring the objects it produced, and doing an evaluation or examination of the project.

These parts offer an organized method of managing software development projects and assist in achieving effective results, from early planning to post-deployment assistance.

4.4 Implementations plans

As soon as an issue is identified and fixed, the new technology has to be turned on. This section selects the settings, protocols, and release requirements. After testing and implementation, the revised system is put into use if everything goes as planned. The data collection must be submitted after all further steps have been completed in order to guarantee correctness. To make the assignment easier to perform, I divided it into its most important components. I have to abide by these guidelines to ensure that my work is completed accurately.

- Collections of datasets.
- Actions taken before processing an image • Predicting class pictures for four fruits illnesses.
- CNN model have been used.
- Using CNN to categorize photos of apple fruits diseases
- Evaluate the accuracy and outcomes.

CHAPTER 5

5.1 Plan

I assembled all of my datasets openly using programs like Kaggle. I decided on a dataset that made sense for the various fruit circumstances. After that, I could go to work on data preparation. Before I tried the idea, I started playing around with the code. I evaluated the accuracy of the four algorithms that were used. I choose the method that would prove most useful to me after taking accuracy into account. This method of diagnosing fruit illnesses has shown to be quite reliable, based on knowledge of conditions in apple fruits. A thorough examination of all pertinent mathematical and logical concepts and methods led to the development of a set of fundamental requirements. These prerequisites must be met for every single photo classification attempt using the create web app tool. Before any project can begin development, its potential, budget, timetable, risk management, and communication server protocols must all be determined. Planning is necessary before work begins in order to reduce risks that might jeopardize the developer's ability to complete the project. Establishing goals and objectives, managing risks, adhering to deadlines, and other things are all examples of components of project planning. Software project plans frequently include time boxes as common tools for project planning. These technologies play a crucial role in the methodical organization of tasks linked to intuitive disease detection in apple fruit.

5.1.1 Management plan

Describe the project team's duties and responsibilities as well as the project management procedure. To guarantee a successful partnership, establish the reporting and communication routes. Determine the steps of the issue-resolution process that involve decision-making and escalation processes.

Table 2: Management Planning

No	Task Name	Duration	Start Date	End Date
1	Introduction	5	1 August 2023	5 August 2023
2	Initial Study	4	10 August 2023	13 August 2023
3	Literature Review	4	17 August 2023	20 August 2023
4	Methodology	7	23 August 2023	29 August 2023
5	Planning	30	3 September 2023	2 October 2023
6	Feasibility	25	5 October 2023	28 October 2023
7	Foundation	18	1 November 2023	18 November 2023
8	Exploration	24	6 January 2024	30 January 2024
9	Engineering	30	1 February 2024	1 March 2024
10	Deployment	18	4 March 2024	21 March 2024
11	Testing	10	23 March 2024	1 April 2024

12	Implementation	65	21 April 2024	26 June 2024
13	Critical Appraisal and Evaluation	10	28 June 2024	7 July 2024
14	Lessons Learning	6	8 July 2024	13 July 2024
15	Conclusion	1	14 July 2024	14 July 2024
	Total	257 days		

5.1.2 Resource Allocation

Determine every resource related to the project, such as staff, hardware, and software. Considering the project schedule and workload, decide how best to allocate the resources. Assign duties and obligations to team members, and make sure they possess the required knowledge and abilities.

Table 3: Resource Allocation

No	Task Name	Duration	Resource
1	Introduction	5	End User
2	Initial Study	4	Analyst
3	Literature Review	4	Analyst
4	Methodology	7	Analyst
5	Planning	30	Analyst, Designer, Developer
6	Feasibility	25	Analyst
7	Foundation	18	Designer
8	Exploration	24	Designer , Developer
9	Engineering	30	Developer
10	Deployment	18	Analyst, Developer

11	Testing	1o	Analyst, Developer, Tester, Users
12	Implementation	65	Analyst, Developer
13	Critical Appraisal and Evaluation	10	Analyst, Tester and Developer
14	Lessons Learning	6	Analyst, Users
15	Conclusion	1	Analyst
	Total	257 days	

5.1.3 Time Boxing

Table : 4 Time boxing

Time-Box	Task Name	Duration	Resource
TB1	Introduction	5	End Users, Analyst
	Initial Study	4	Analyst
	Literature Review	4	Analyst
TB2	Methodology	7	Analyst
	Planning	30	Analyst, Designer, Developer
	Feasibility	25	Analyst
TB3	Foundation	18	Designer

TB4	Exploration	24	Designer, Developer
	Engineering	30	Developer
TB5	Deployment	18	Analyst, Developer
	Testing	10	Analyst, Developer, Tester, Users
TB6	Implementation	65	Analyst, Developer
TB7	Critical Appraisal and Evaluation	10	Analyst, Tester and Developer
	Lessons Learning	6	Analyst, Users
TB8	Conclusion	1	Analyst
	Total	257 days	

CHAPTER 6

Feasibility

An engineering, organizational, legal, and financial feasibility study establishes the possibility that all pertinent aspects will be considered for a project to be completed effectively. In system development, operationally practicability is a measure of how well a system matures, utilizes the scope defined during potential definition, and satisfies requirements set during the project or requirement assessment level. A online application that we have built will enable the stepby-step diagnosis of certain fruit diseases through the use of deep learning. A web-based program that functions as an apple fruit disease detector for apples is at the core of the suggested concept.

6.1.2 Technical feasibility

Hardware	Software	Database
Dell Laptop, Wi-Fi, Router, Cable, Android Phone	Android Studio, Google Chrome Browser, Windows, MS Word, VS code	MySQL

6.1.3 Technology

Client slide	Server side
Html, CSS, JavaScript, React JS	Python

6.2

The idea by project managers to weigh the advantages and disadvantages of various project paths, taking into account relationships, activities, business requirements, and capital. A cost-benefit analysis helps me choose the best route of action to achieve my objective for the smallest amount of funds out of all the possibilities available.

Project Name: Seasonal Fruit distribution System

Table 5: Cost Benefit

Equipment	1 Periods	2 Periods	3 Periods	4 Periods	Total
Web Based Application	20000				20000
Domain & Hosting		10000	10000	10000	30000
Software	1000				1000
Internet	2000	2000	2000	2000	8000
Model Training	5000				5000
Development		5000			5000
Maintenance	10000	10000	10000	10000	4000
Total					73,000 BDT.

6.3 DSDM Dynamic System Development Method (DSDM)

Rather of being a particular application development tool or technology,. It heavily emphasizes the use of iterative development methods, regular software delivery, and cooperation between development teams and business stakeholders. It's crucial to remember that DSDM requires the use of specific tools or technologies, such as JavaScript, React JA, HTML, and CSS with Python.

These are popular web development tools that are definitely useful for DSDM initiatives.

CHAPTER 7

Foundation

7.1 Some potential approaches

7.1.1 Interview

The scheduling of interviews is crucial in every endeavor. I am able to obtain data on chicken sickness, pay attention to users' needs, and engage with qualified people through interviews in order to assist users in using the fruit disease detection tool to solve using apple images of 4 classes of apple fruit illness. I can learn a great deal about identifying fruit illnesses by doing interviews with people. I might learn about all the needs and any barriers to communication around distributing resources by participating in interviews.

7.1.2 Observation

The suggested project, that's a website that was developed to identify fruit illness, allows users to complete a range of paperwork prior to and following their diseases are detected on a single platform. We have developed a web-based program that makes it possible to swiftly and effectively identify fruit disease and manage every facet associated with its detection system in a thorough, methodical manner.

7.1.3 Data Collection

I have acquired 512 images in all. The Kaggle sources provided the internet data that was gathered. The four categories in the dataset, "Rot_Apple," "Scab_Apple," "Normal_Apple," and "Blotch_Apple," are named for the types of fruit ailments of which they are typical. Half of the data are utilized for test and validation, while the remaining 20% are labeled as train.

Fig:1 Sample of dataset

7.1.4 Data Processing

Each image in the collection I used was produced by merging the various online field data on height and width that Kaggle had collected. Since every image for my model must meet strict quality standards, I modified the script to produce an image with 224 by 224 pixels. Additionally,

I prefixed each image in my model with "jpg" before processing it. After data augmentation, I divided and edited the pictures to prepare them for classification. Consequently, I trained the framework using each dataset's split counterpart.

- Pictures with predefined code-based sizes.
- JPG will be created from the various file kinds.
- Eliminate any inaccurate photos.
- Delete any unnecessary photos.

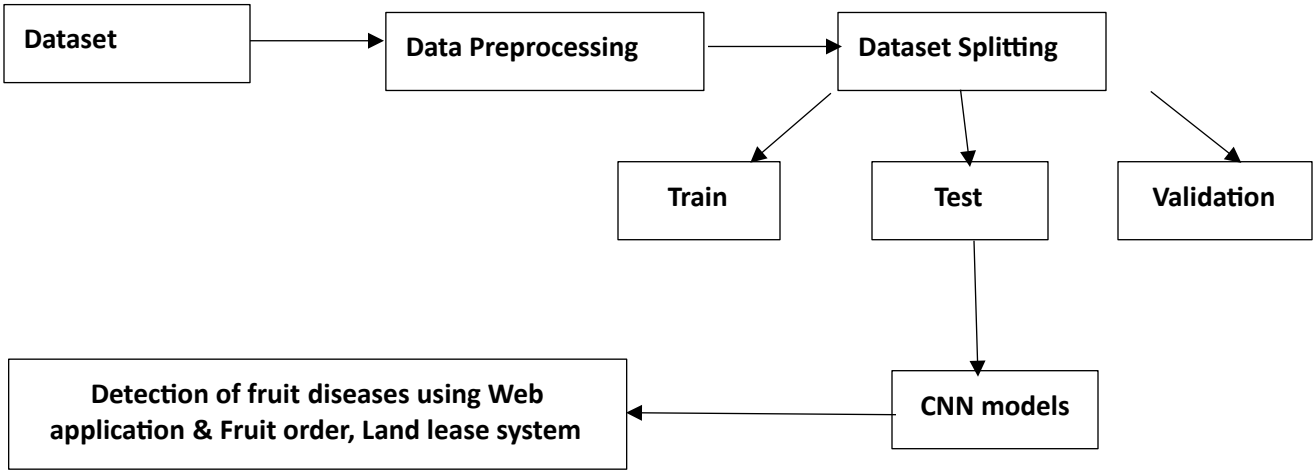


Fig 1: The model that is suggested for the entire study endeavor.

7.2

A number of and their solutions come up in numerous important sectors while developing and implementing an AI-powered web application for diagnosing illnesses in apples. To create a practical and effective application, these problems must be identified and resolved. Many challenges need to be overcome in order to create an AI-powered web tool for diagnosing fruit diseases. These challenges include those related to the confidentiality of information, modelling abstraction, fast processing, solidarity, confidentiality of data, model servicing, cost, concerns about ethics, and user education. To solve these specific problems and ensure that the app is credible, effective, and simple to use, careful planning, continual improvement, and support are required.

7.3 Possible solution

A based on artificial intelligence web app for fruit disease detection may employ a combination of practical, technological, and strategic approaches to address its unique problems. Four different class kinds have been employed to identify utilizing the apple fruits of the subsequent deep learning models. It is necessary to address several concerns in order to use AI to solve the difficulties with a web app that detects fruit illness. App developers may increase the app's effectiveness, reliability, and user satisfaction by putting these

modifications into practice. This will enhance disease prevention and lead to the detection of healthier fruit varieties.

7.4 Overall Requirement List

- Functional Requirements
- Non-Functional Requirements.

7.4.1 Functional Requirements

7.4.1.1 User

- User Dashboard
- Sign In & Sign Up
- Fruit list and booking no of items.
- Land list and lease the land with no of items.
- Fruit disease Detection system.

7.4.1.2 Admin

- Admin Dashboard
- View & Edit fruits and land item
- View no of fruits booking items
- View no of land lease

7.4.2 Non-Functional Requirements

7.4.2.1 Security

All users in the system have their own accounts, to which only authorized users with a password are allowed access. The passwords are encrypted using JavaScript and Python.

Fig : User interfaces

7.4.2.2 Performance

It's easy to update and maintain documents. Once good accuracy was gathered for the prediction of a particular fruit disease, performance improved.

7.4.2.3 Availability

Users just need a PC with a connection to the web to utilize the system whenever they want and from wherever. Web browsers that are compatible with the system include Internet Explorer, Mozilla, Opera, and Chrome.

7.4.2.4 User Friendly

- The technology has an engaging user interface and is easy to use.
- When a large number of people are accessing the website at once.

7.5 Which technology to be implemented

The program I'm working on is entirely web based. I'm developing my project with JavaScript, React JS, Html, CSS, Python.

HTML: Web pages may be created using markup languages like HTML. HTML files may be seen and understood by web browsers. All websites are built using HTML components. enables the usage of HTML elements and graphics, letting you create engaging content. In [2], it may also produce chapters, lists, quotes, titles, and links.

CSS: To further personalize the material on our webpage, we may utilize CSS to alter the fonts, colors, and layouts. This improves the cohesion and aesthetic attractiveness of our website. The website seems friendlier as a consequence. Inside [2]

JavaScript: JavaScript is currently one of the most popular programming languages. We also employ JavaScript as a web development language. In doing so, a layer of common web technologies is developed. Within [3]

Bootstrap 4: As of right now, Bootstrap is version 4.0. Using Bootstrap 4, you can create responsive websites with all the necessary HTML, CSS, and JavaScript components. I consequently granted users accounts on my website.[4]

React JS: React.js is a feature-rich JavaScript framework designed especially for designing graphical user interfaces (UIs) for web applications. React.js is widely utilized because to its efficiency, enhanced performance, adaptability as an aspect, and friendly for developers features. Cutting-edge, dynamic internet apps and interfaces for users may be made using it increasingly as additional capabilities and features are added.[5]

A studied subject is an area of study that is being examined and examined right now in order to make sense of concepts for creating models, establishing and completing objectives, collecting information, managing, instructing, and enhancing performance. I go through the tools and methods I use to measure things. NumPy makes possible Python programming, software from Microsoft, and other tools such as Scikit-learn. Google Co Labs' facilities are only utilized for instruction and assessment. Python-based data analysis and sophisticated DL algorithms might be created by Google Colab developers.

Libraries:

- **Matplotlib**: One of Matplotlib's visualization features is Py-plot graphing, which is a set of methods. Plotting lines inside a plot and creating boundaries are only two of its numerous applications. Another is shapes.
- **NumPy:** One popular way to work with vectors in Python is to use the NumPy package. The fundamentals of algebraic geometry, matrix operations, and the Fourier transform are covered in this section. The NumPy package offers Python tools and resources for working with matrices of various sizes. NumPy facilitates the deliberate and exact construction of arrays. NumPy is a Python package used for numerical operations. There's also reference to "a wide variety of varieties Python".
- **H5 Py:** A native data wrapper for Python HDF5 is provided via the h5py module. HDF5's support for NumPy makes it simple to manage and store large amounts of numerical data.
- **OS:** Artists can use a range of tools available in the OS section of the Python programming language to interact with the software that powers certain parts of the computers they work on.

- **TensorFlow:** A free programming language Python math package that makes it easier to create self-learning techniques and deep neural network systems.

7.6 Recommendation and justifications

Making strategic recommendations to overcome various challenges is necessary when implementing an AI-powered online application for fruit disease detection. By employing edge computing, developing via transfer, supplementing data, and focused on users design, the software's productivity and friendliness are enhanced. Integration with existing systems, strong data protection measures, regular model upgrades, comprehensive training, and moral issues all contribute to a successful deployment. By evaluating cost-effectiveness and exploring financing possibilities, the app's development and long-term sustainability are further encouraged, improving the management of poultry health and operational efficiency. Using data specific to fruits, pre-trained algorithms on large image datasets may be refined to improve the accuracy of sickness identification. Through the use of pre-existing algorithms that have developed general properties, transfer learning reduces the amount of data and training time needed.

I used MySQL for the database and Python, React JS, and JavaScript. "Seasonal fruit distribution system" develops an allencompassing framework by merging several technologies. The limits described are applied by each of the parts of this architecture to each system that is being studied. Each module has a number of systems. Establish systems that will retrain models automatically with new data to guarantee correctness. To find and fix issues as soon as possible, keep an eye on and evaluate the model's performance.

CHAPTER 8

Exploration

8.1 Use case

This part analyzes both non-functional and functional needs using visuals and use-case data.

User:

Following their login, the user can carry out the following tasks:

- User Dashboard

- Sign In & Sign Up
- Details about company

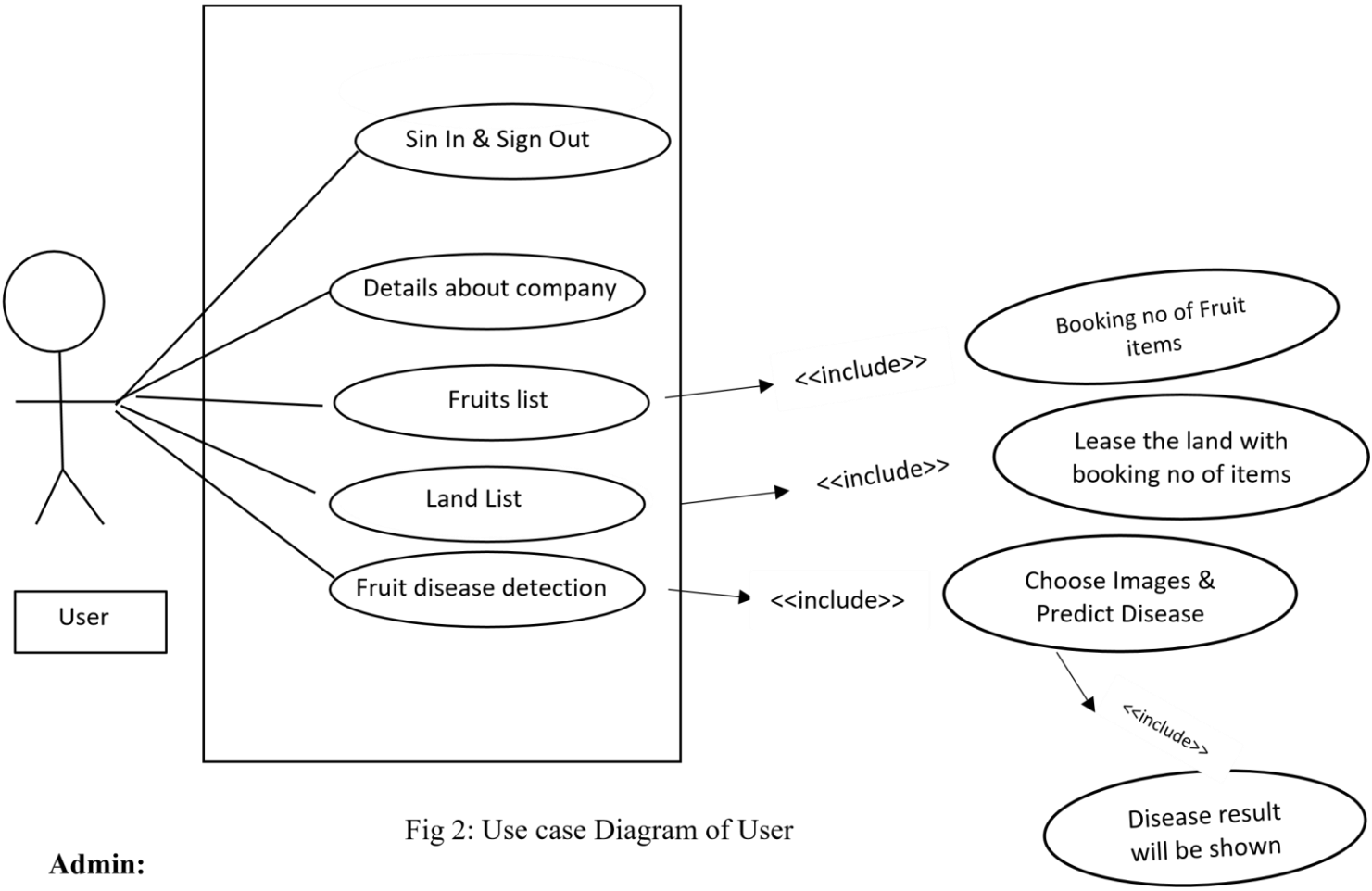


Fig 2: Use case Diagram of User

Admin:

- Fruit list and booking no of items.
- Land list and lease the land with booking no of items.
- Fruit disease Detection system.

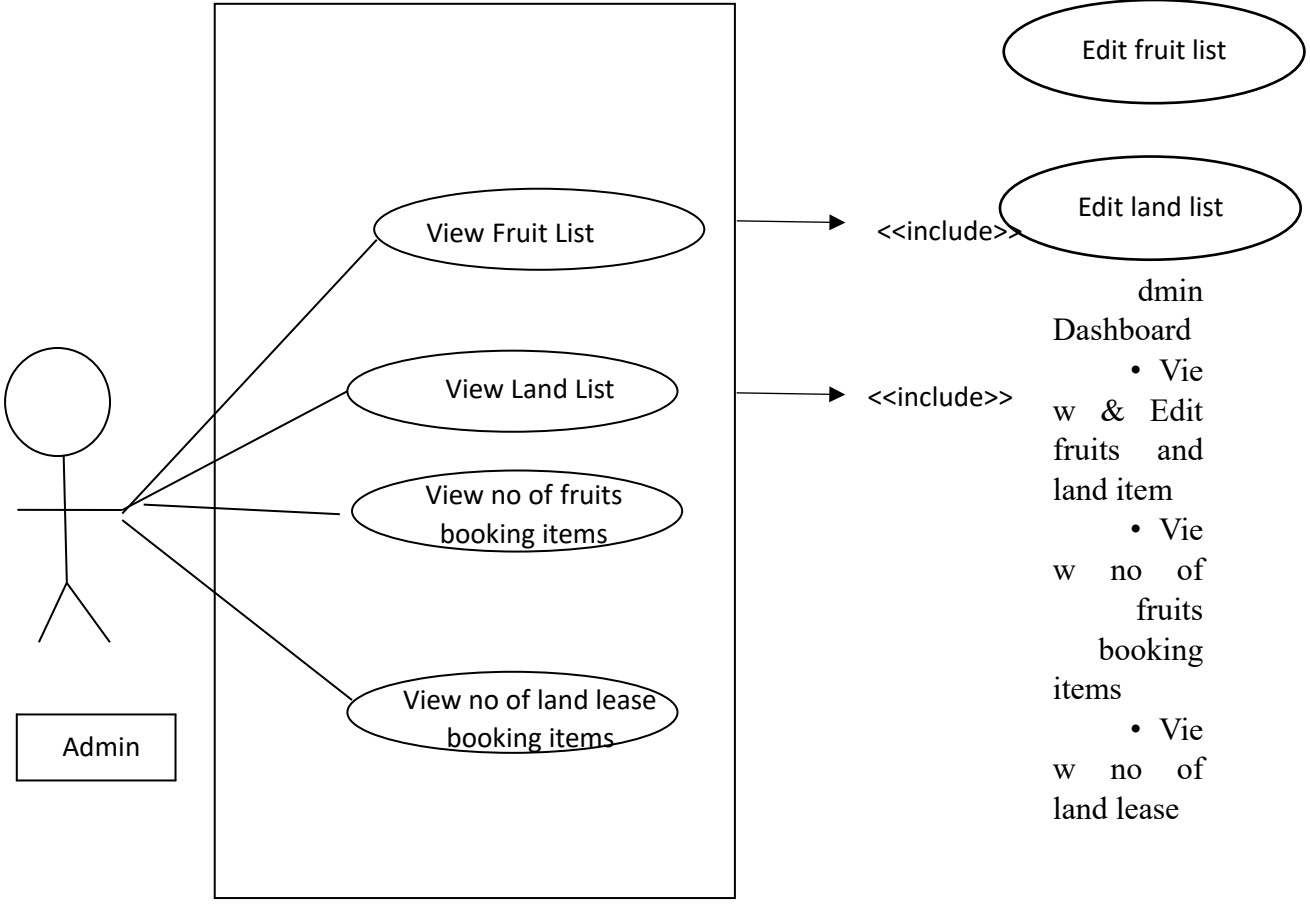


Fig 3: Admin

8.2

Explain the system's dynamic features. It resembles a flowchart that illustrates the connections between several jobs. You might use the activity to explain the operation of the system. As a result, control is shared across all the operations. The entire activity diagram for each module has this appearance:

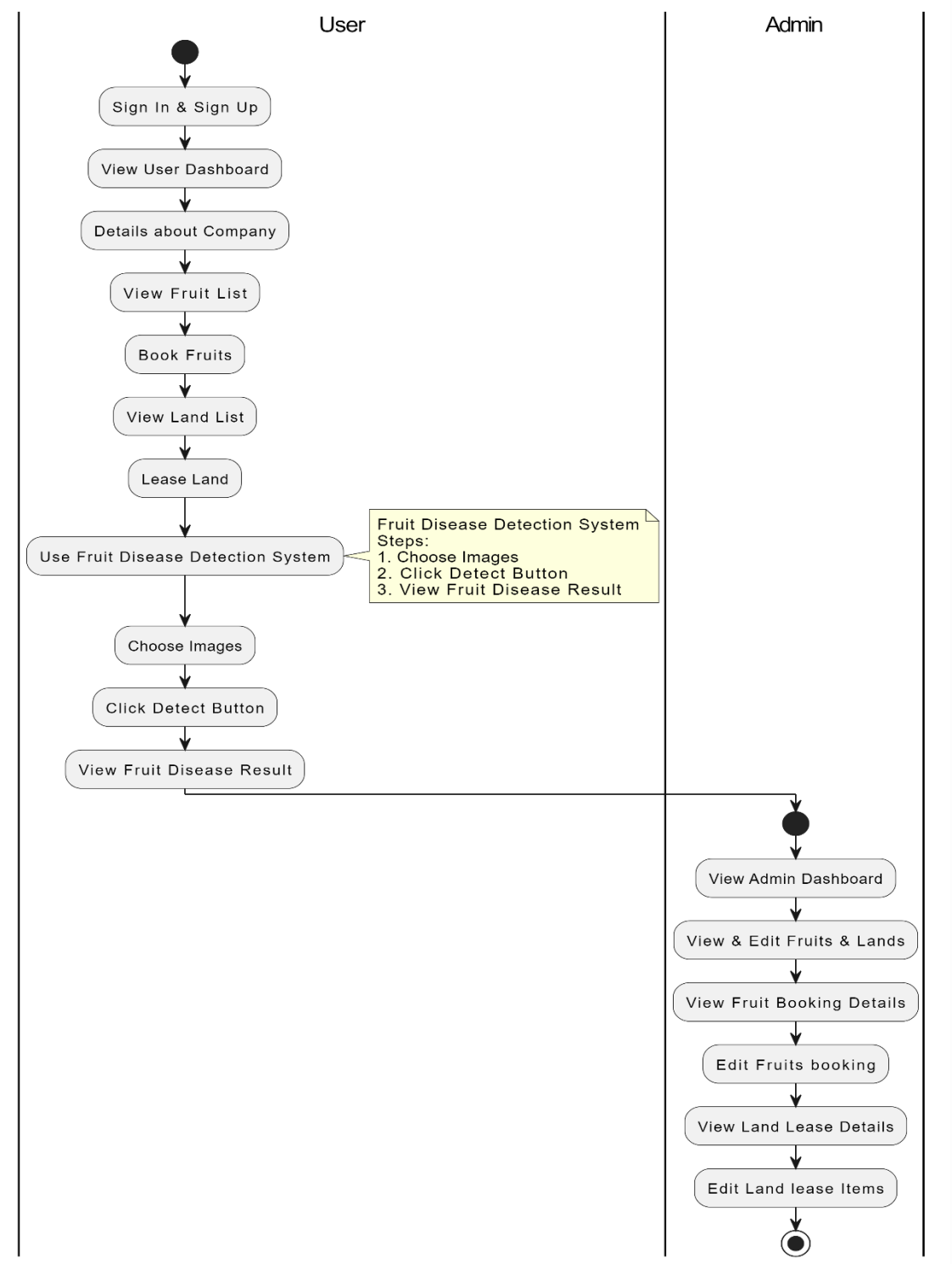


Fig 4: Activity Diagram.

8.3 Requirement catalogue

Functional requirements:

- FR1: User can Sign In & Sign Up for access the web app
- FR2: User can see Fruit list and booking no of items.
- FR3: User can see Land list and lease the land with booking no of items.
- FR4: Fruit disease Detection system using DL.
- FR5: Admin can edit & add fruit and land list and store in database.

Non-Functional Requirements:

- NFR1: Maintaining and updating records is simple.
- NFR2: Users just need a PC with an Internet connection to utilize the system whenever they want and from wherever. Web browsers that are compatible with the system include Internet Explorer, Mozilla, Opera, and Chrome.
- NFR3: The system has an engaging user interface and is easy to use.
- NFR4: View business information and contact data.
- NFR5: User and admin can view the fruits and lands info.

User Interface Requirements:

- UIR1: A user-friendly interface with intuitive navigation that makes it simple to access various features and functionalities.
- UIR2: Responsive design allows for compatibility with various screen and device sizes.
- UIR3: Icons offer visual signals to help users comprehend and operate the system.

Security and Privacy Requirements:

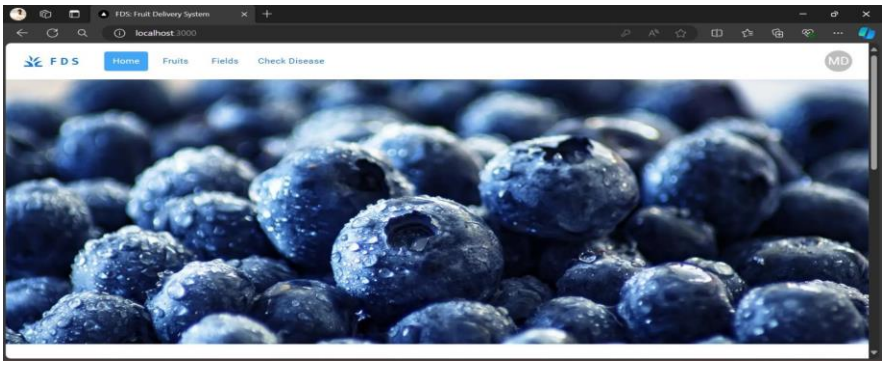
- SR1: User accounts and data are protected by secure permission and authentication mechanisms.
- SR2: Protect all data from various fruits, properties, and user accounts.

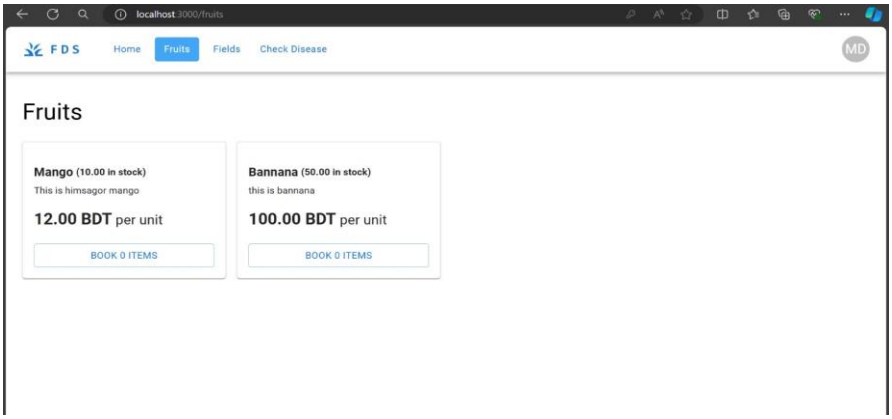
8.4 Prioritized Requirement List (PRL)

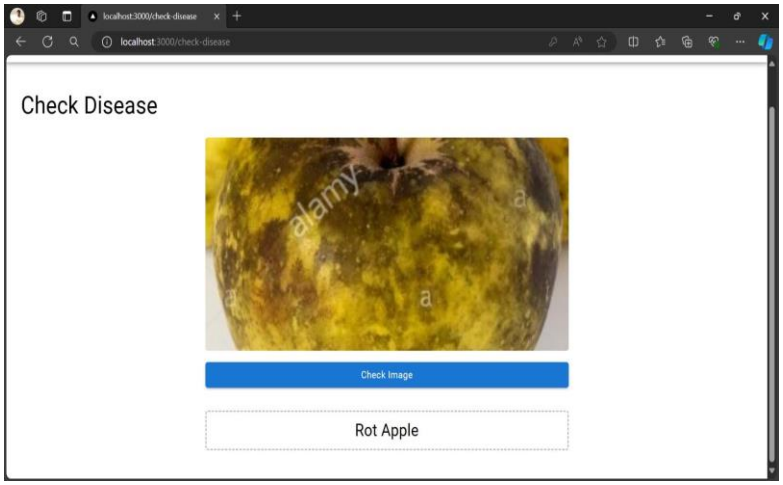
Table 6: Prioritized requirement list

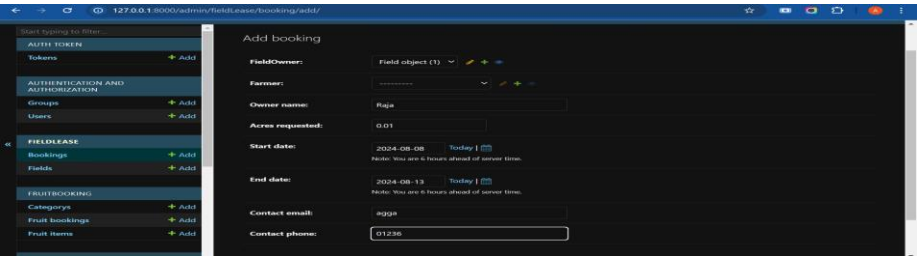
Requirement ID	Requirement Description	Priority	Dependencies	Status	Validation Criteria
RQ1	User login functionality	High			User can login successfully
RQ2	Fruits booking	High	RQ1		User can see Fruit list and booking no of items.
RQ3	Land lease booking	High	RQ1		User can see Land list and lease the land with booking no of items.
RQ4	Detection of fruits diseases	High	RQ1		Detect 4 classes of apple fruits disease.
RQ5	Admin panel	High			Admin can edit & add fruit and land list.

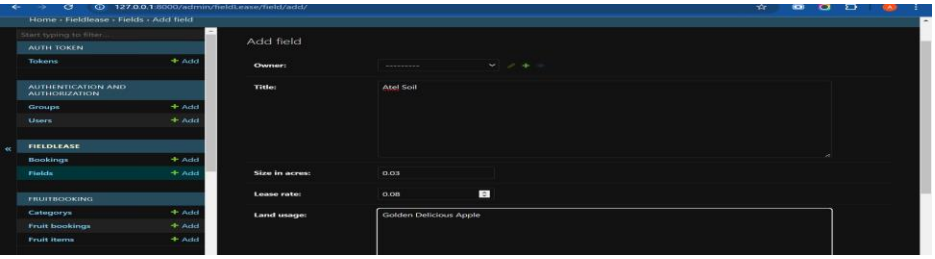
8.5 Prototype of new system

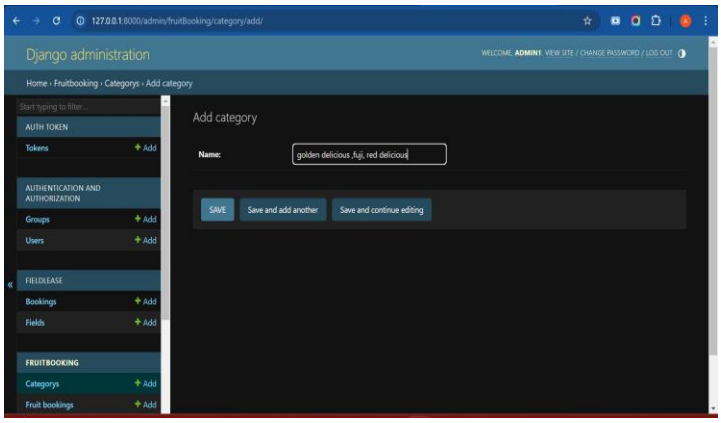


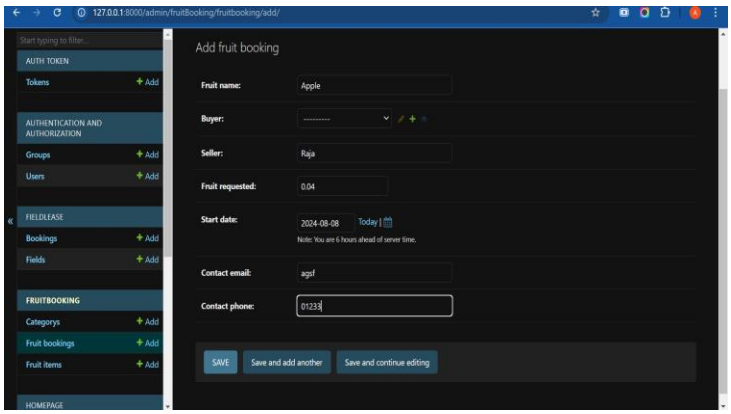



localhost:3000/check-disease
Check Disease
Check Image
Rot Apple

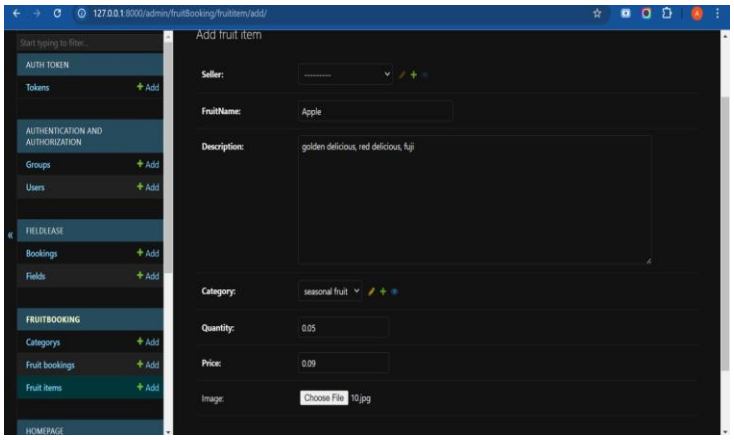

Add booking
FieldOwner:
Farmer:
Owner name:
Raja
Acres requested:
0.01
Start date:
2024-08-08
End date:
2024-08-13
Contact email:
egga
Contact phone:
01236

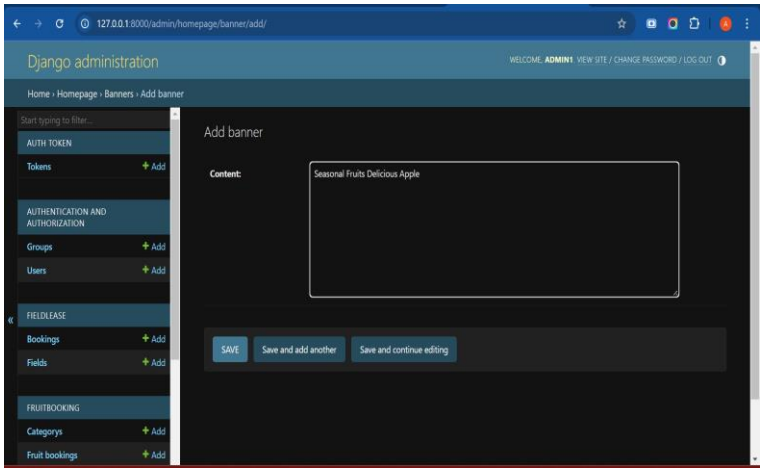

Add field
Owner:
Title:
Atel Soil
Size in acres:
0.03
Lease rate:
0.08
Land usage:
Golden Delicious Apple

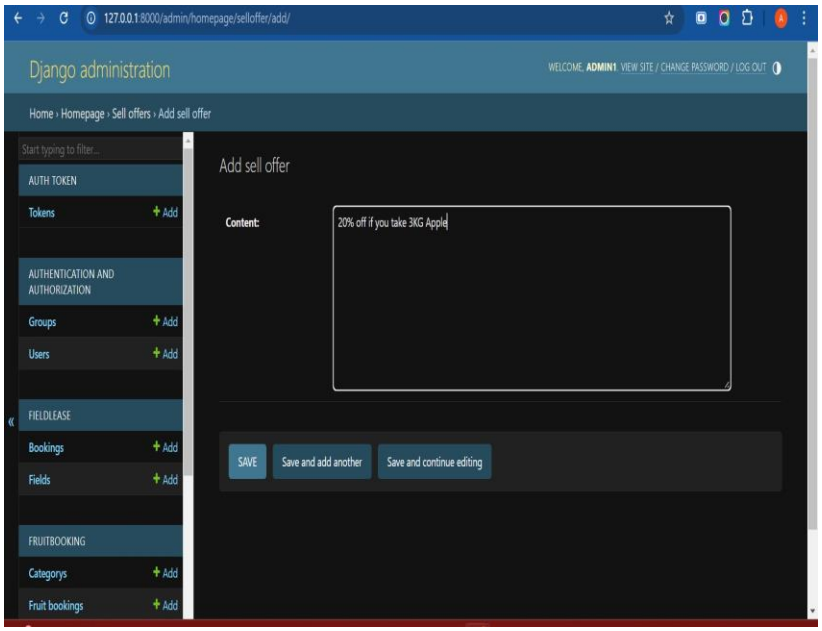

127.0.0.1:8000/admin/fruitBooking/category/add/
Django administration
WELCOME, ADMIN1. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Fruitbooking › Categorys › Add category
Start typing to filter...
AUTH TOKEN
Tokens + Add
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
FIELDLEASE
Bookings + Add
Fields + Add
FRUITBOOKING
Categorys + Add
Fruit bookings + Add
Add category
Name:
golden delicious,fuji, red delicious
SAVE
Save and add another
Save and continue editing

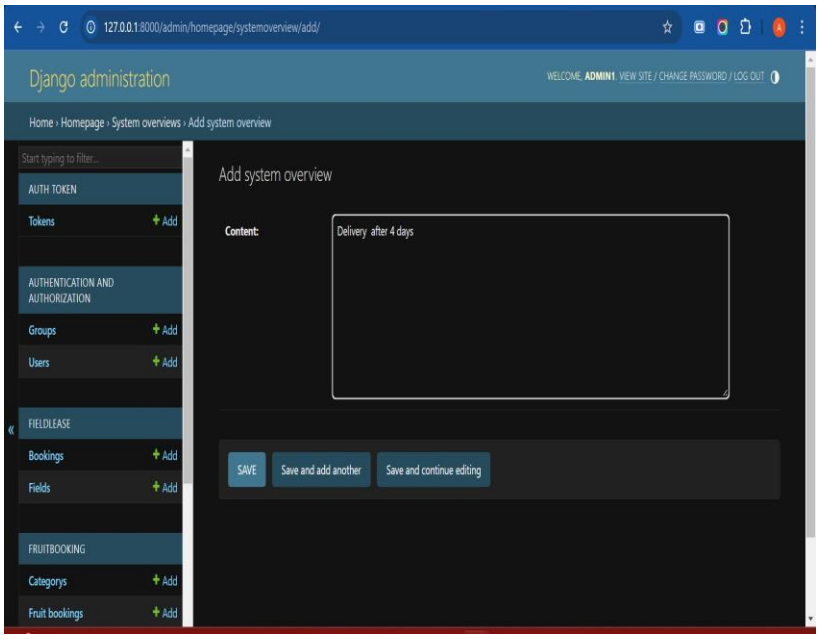

127.0.0.1:8000/admin/fruitBooking/fruitbooking/add/
Start typing to filter...
AUTH TOKEN
Tokens + Add
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
FIELDLEASE
Bookings + Add
Fields + Add
FRUITBOOKING
Categorys + Add
Fruit bookings + Add
Fruit items + Add
HOMEPAGE
Add fruit booking
Fruit name:
Apple
Buyer:

Seller:
Raja
Fruit requested:
0.04
Start date:
2024-08-08
Today |
Note: You are 6 hours ahead of server time.
Contact email:
agsf
Contact phone:
01233
SAVE
Save and add another
Save and continue editing


127.0.0.1:8000/admin/fruitBooking/fruitItem/add/
Add fruit item
Seller:
FruitName:
Apple
Description:
golden delicious, red delicious, fuji
Category:
seasonal fruit
Quantity:
0.05
Price:
0.09
Image:
Choose File
10.jpg


127.0.0.1:8000/admin/homepage/banner/add/
Django administration
Home › Homepage › Banners › Add banner
Add banner
Content:
Seasonal Fruits Delicious Apple
SAVE
Save and add another
Save and continue editing


127.0.0.1:8000/admin/homepage/selloffer/add/
Django administration
WELCOME, ADMIN1. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Homepage › Sell offers › Add sell offer
Start typing to filter...
AUTH TOKEN
Tokens + Add
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
FIELDLEASE
Bookings + Add
Fields + Add
FRUITBOOKING
Categorys + Add
Fruit bookings + Add
Add sell offer
Content:
20% off if you take 3KG Apple
SAVE
Save and add another
Save and continue editing


127.0.0.1:8000/admin/homepage/systemoverview/add/
Django administration
WELCOME, ADMIN1. VIEW SITE / CHANGE PASSWORD / LOG OUT
Home › Homepage › System overviews › Add system overview
Start typing to filter...
AUTH TOKEN
Tokens + Add
AUTHENTICATION AND AUTHORIZATION
Groups + Add
Users + Add
FIELDLEASE
Bookings + Add
Fields + Add
FRUITBOOKING
Categorys + Add
Fruit bookings + Add
Add system overview
Content:
Delivery after 4 days
SAVE
Save and add another
Save and continue editing

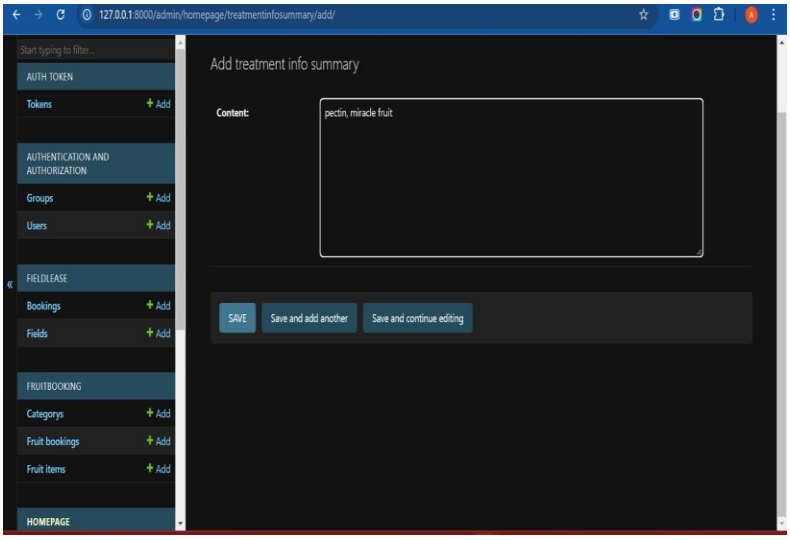


Fig 5: Interfaces for dashboard

CHAPTER 9

Engineering

9.1 Class Diagram

A class was created in order to show the original material for the interclass links. In this case, the class describes an object's variables and actions, either as a single programming specification or as a separate entity inside a program.

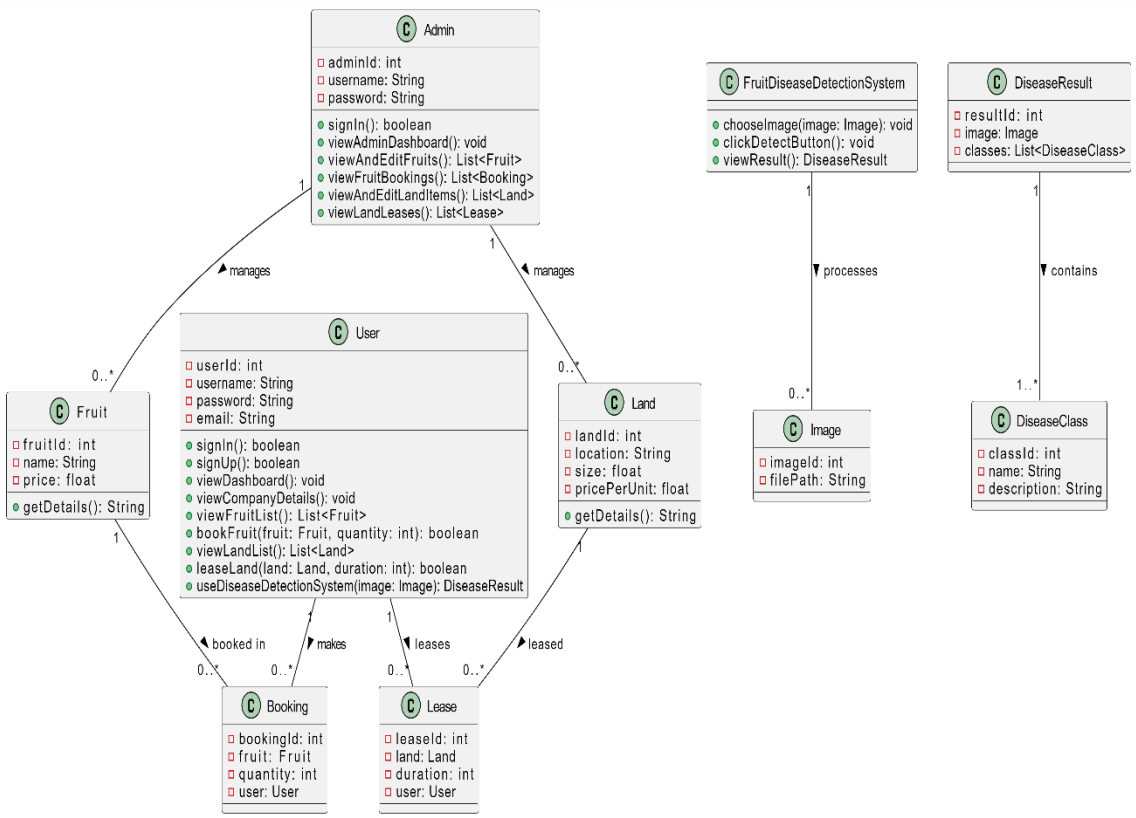


Fig 6: Class Diagram

9.2 ER diagram

The institutional communication system, sometimes referred to as the model of the ER, ER Diagram, or ERD, is an example of a structural application used in design. Two main kinds of data that the ERD conveys and displays in various ways are the main elements of the limited system and the connections among these different entities. We have finished the entityrelationship diagram for the user module.

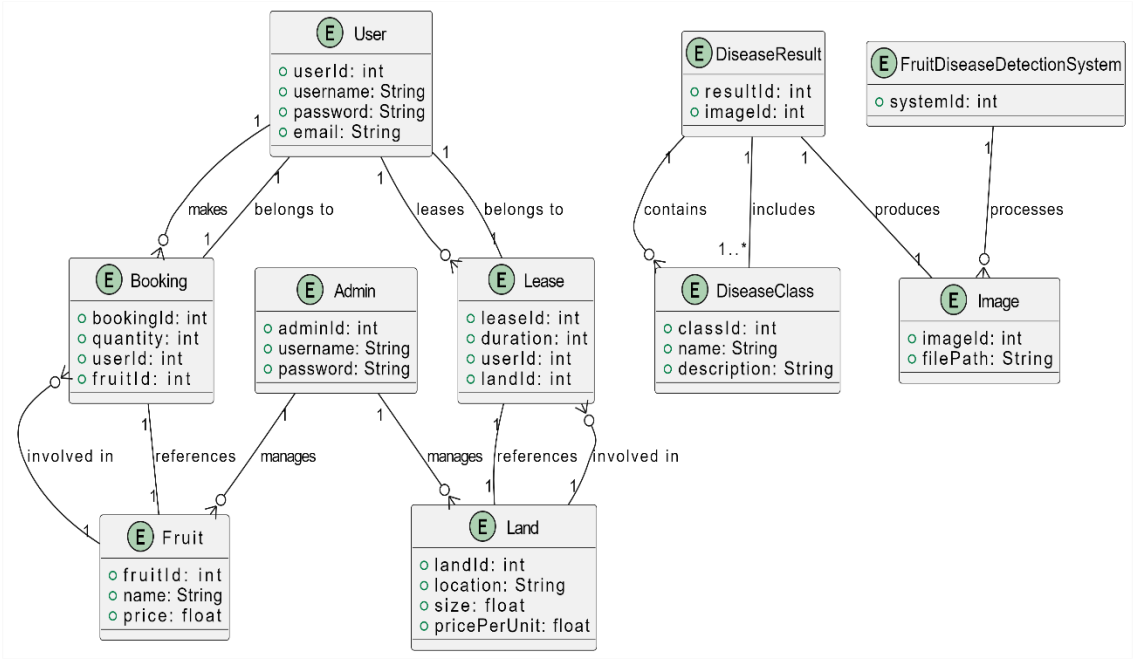


Fig 7: ER Diagram.

9.3 Sequence Diagram

The user and admin dashboard is the only module that is the subject of this discussion.

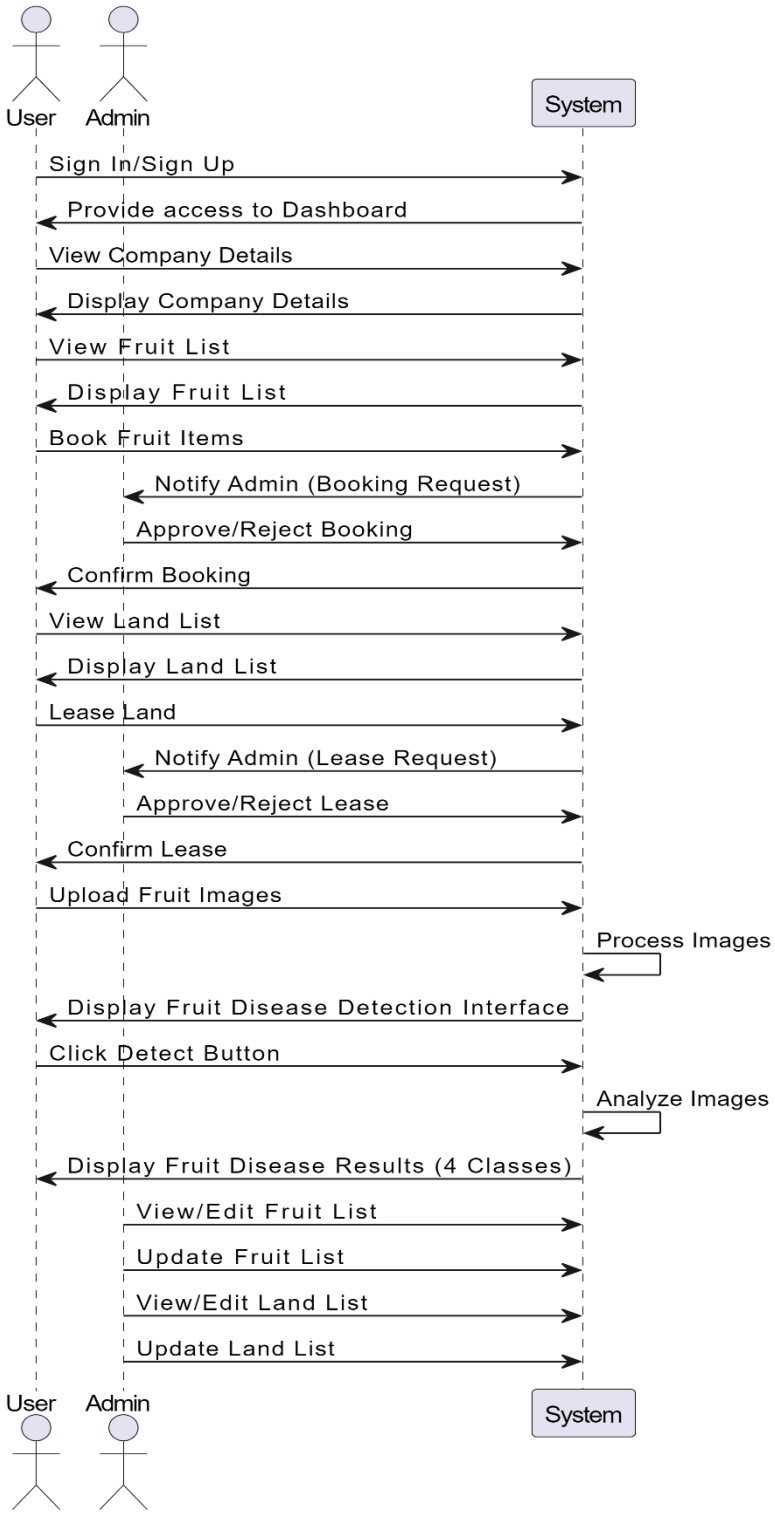
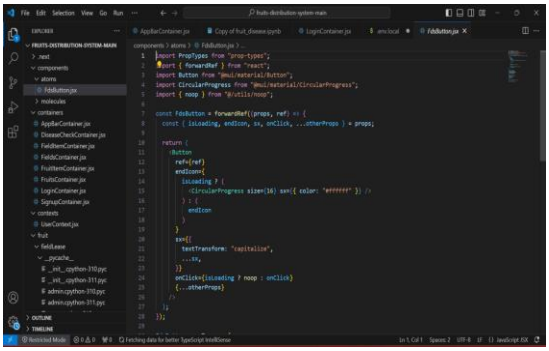


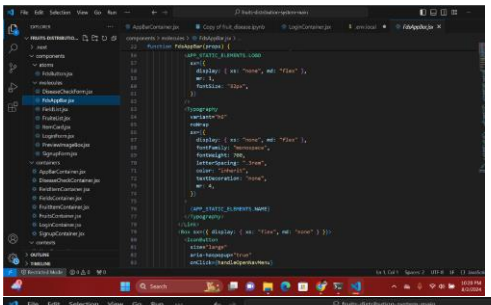
Fig 8: Sequence Diagram

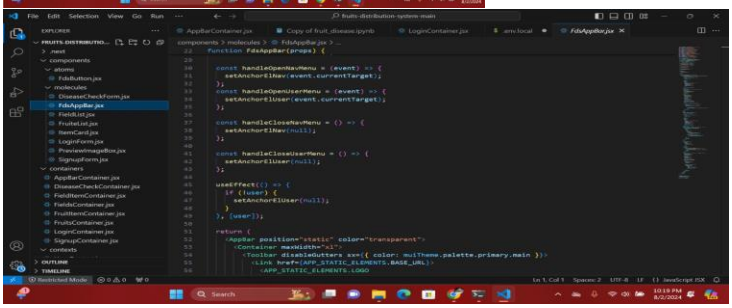
CHAPTER 10

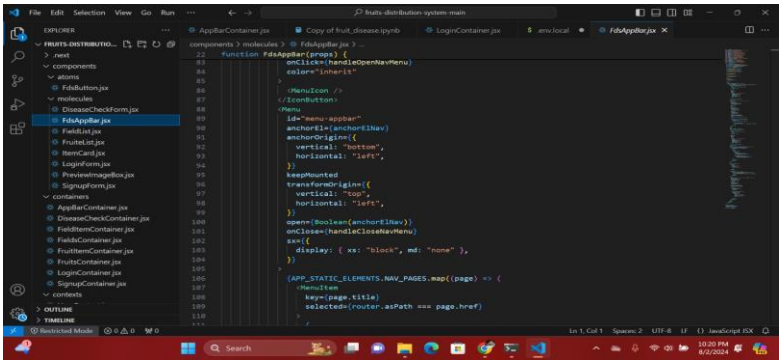
Development

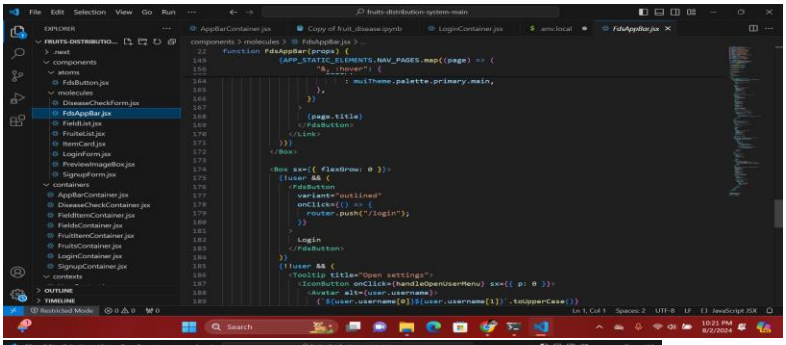
10.1 Core Module Samples

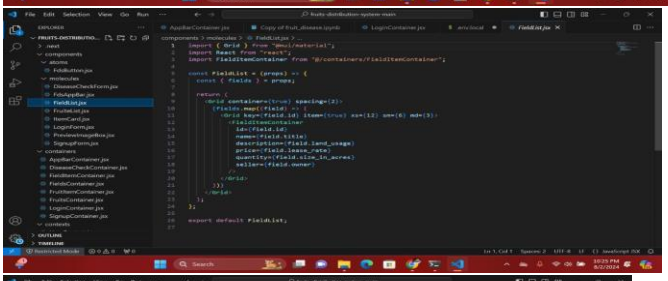


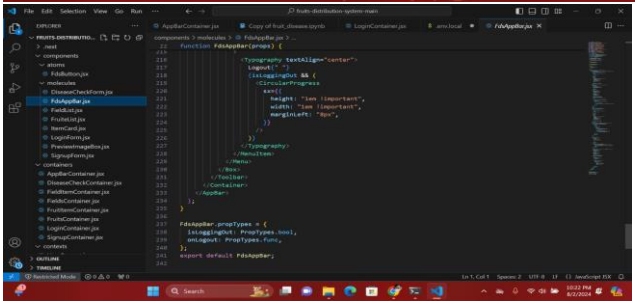


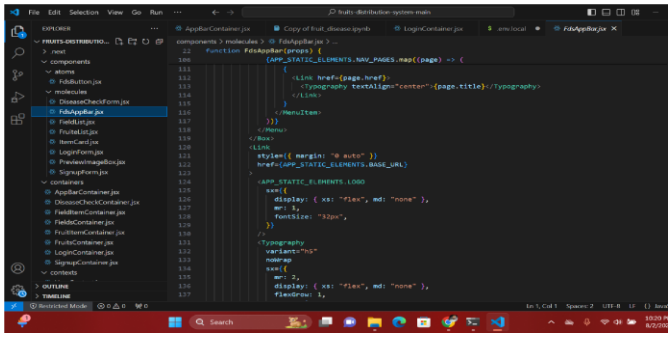


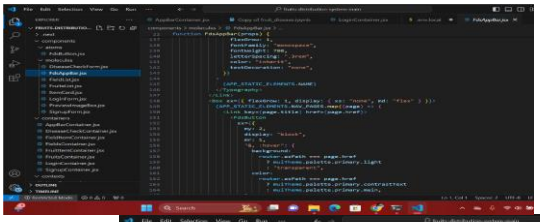


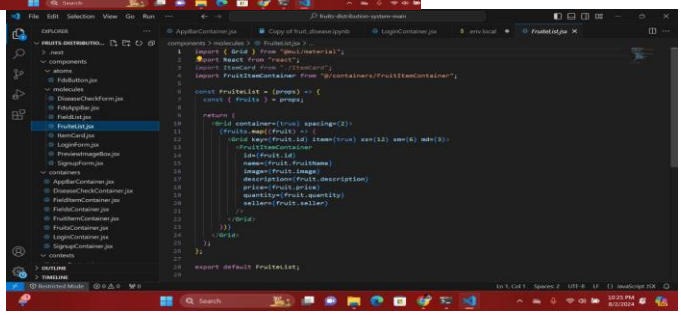


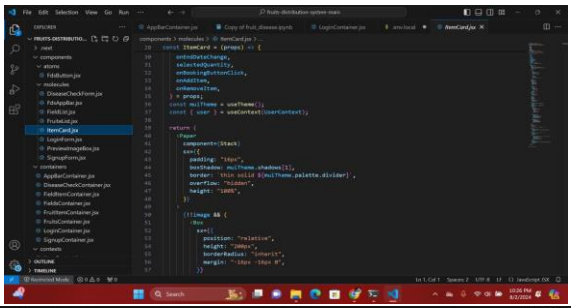


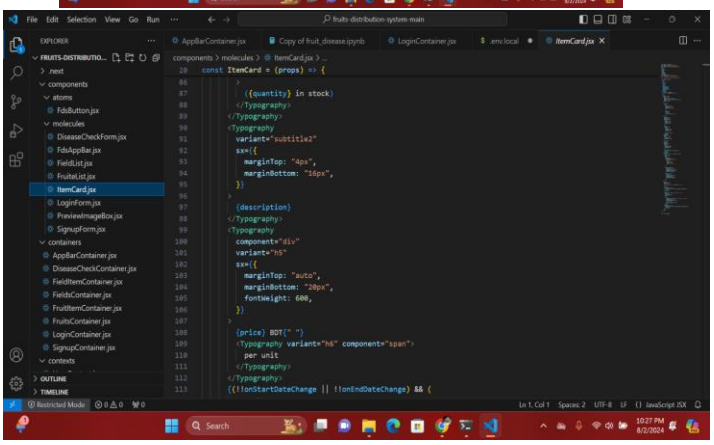


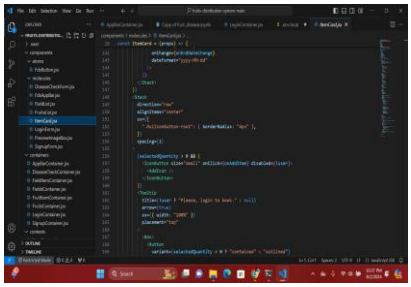


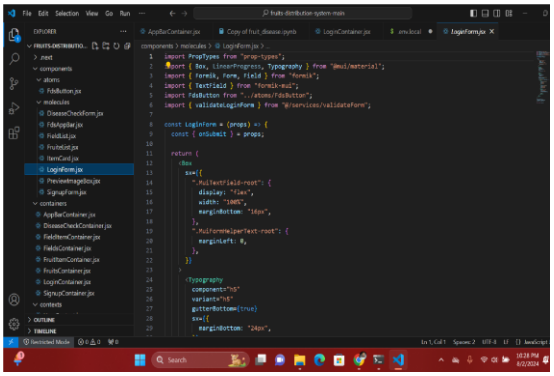


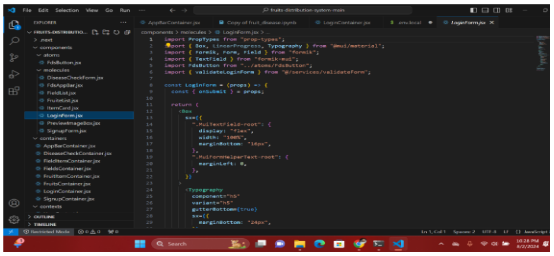


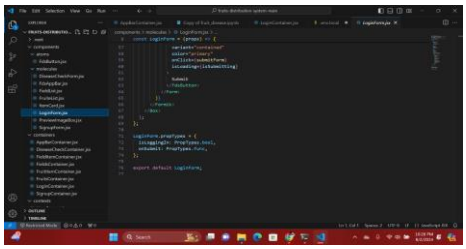


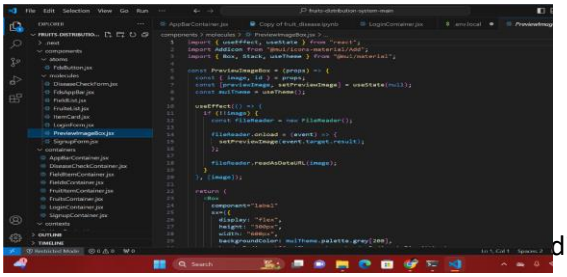


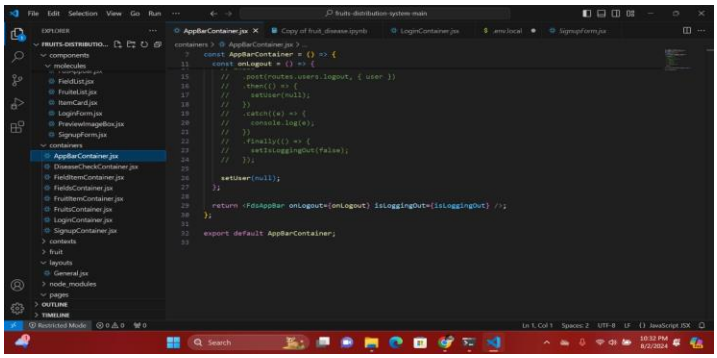


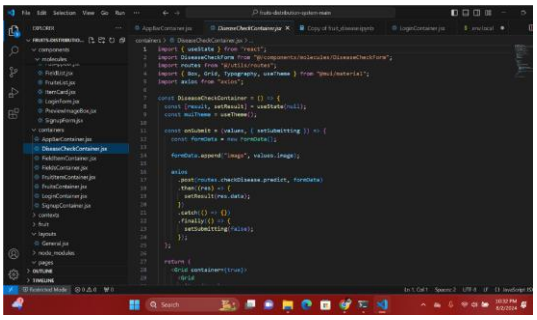


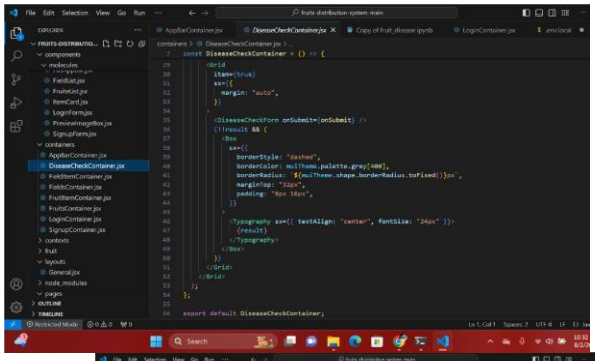


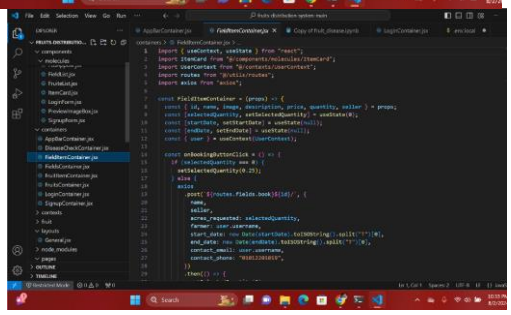


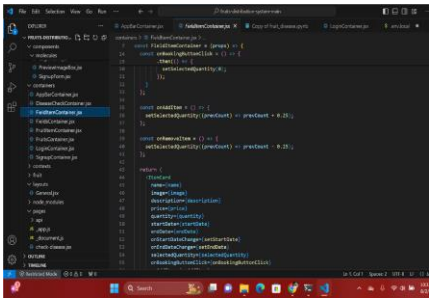


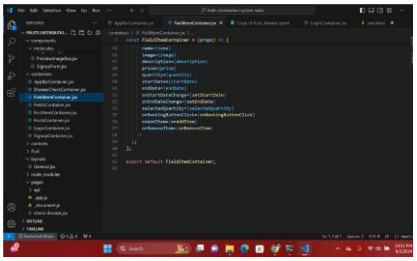


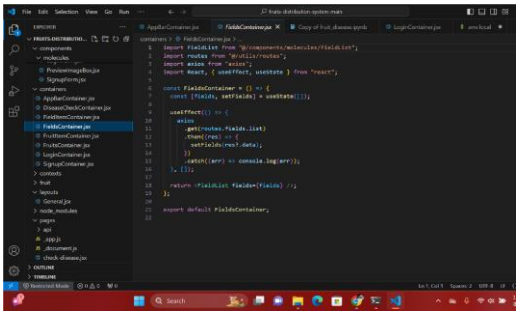


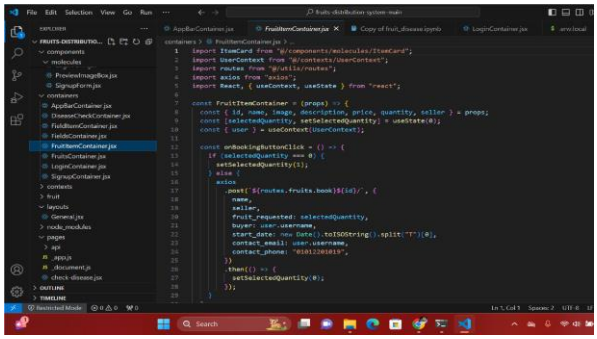


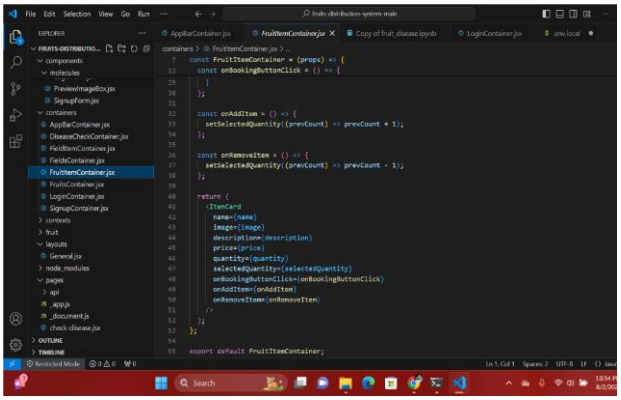


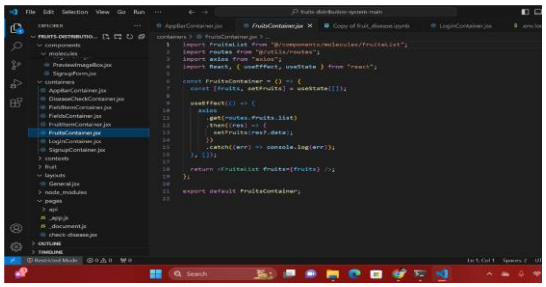


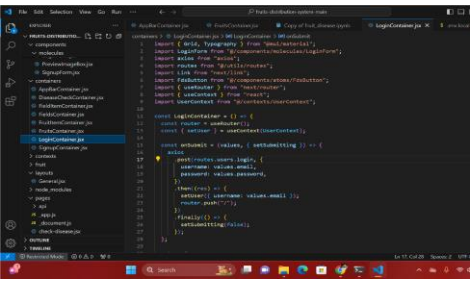


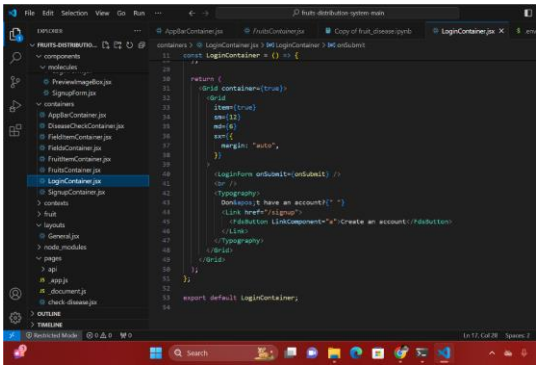


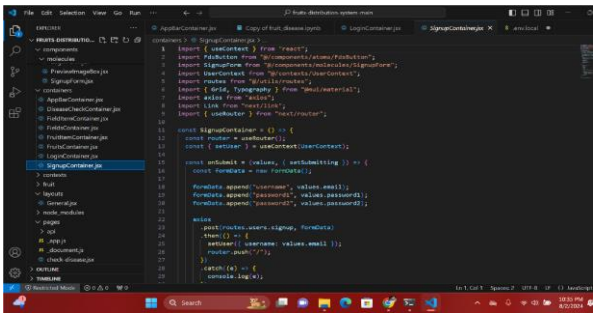


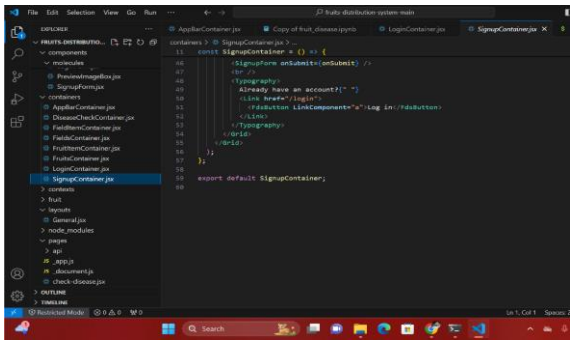


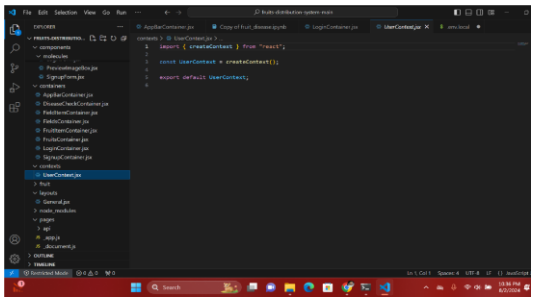


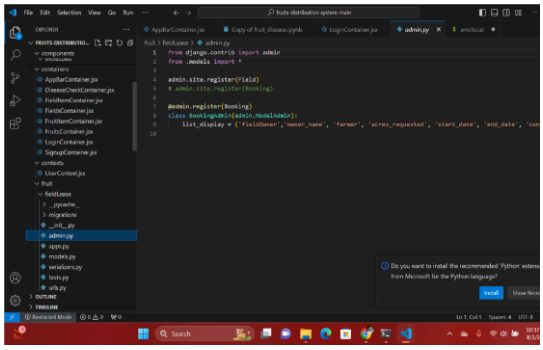


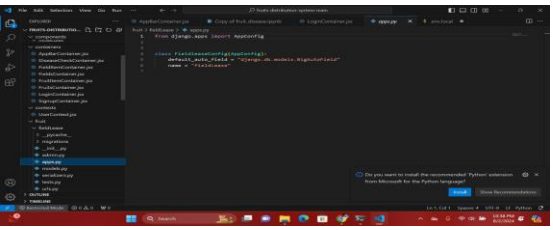


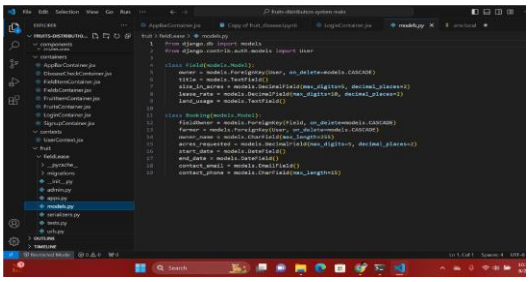


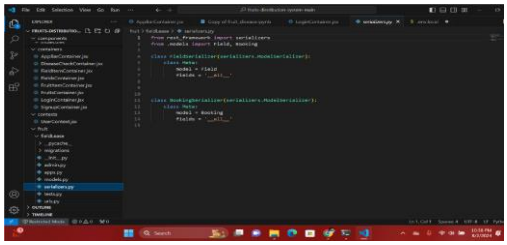


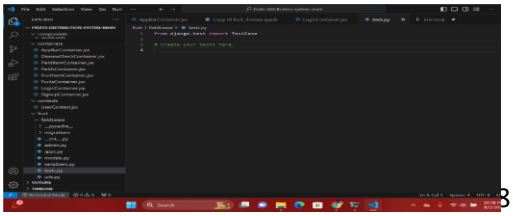


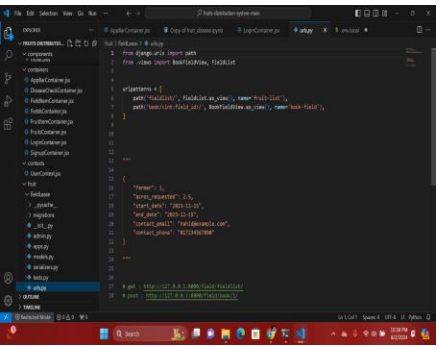


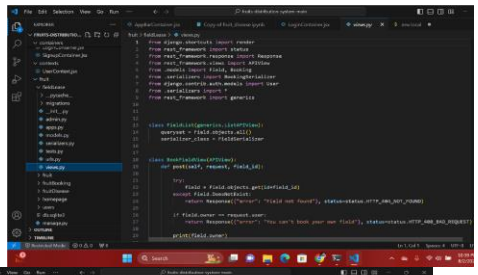


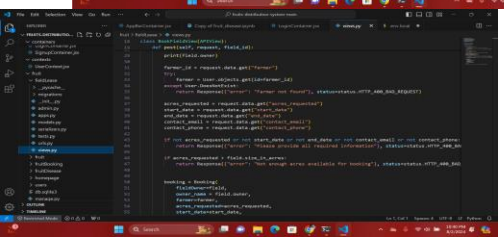


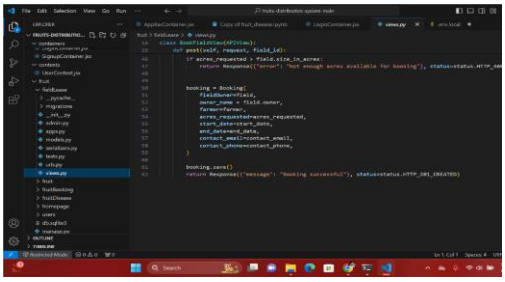


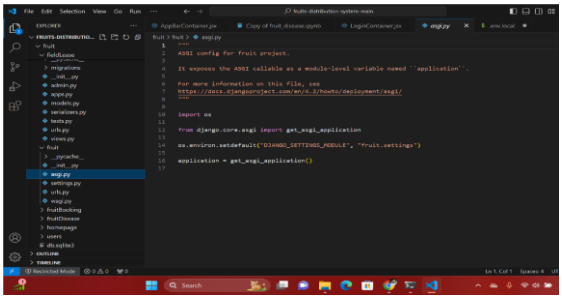


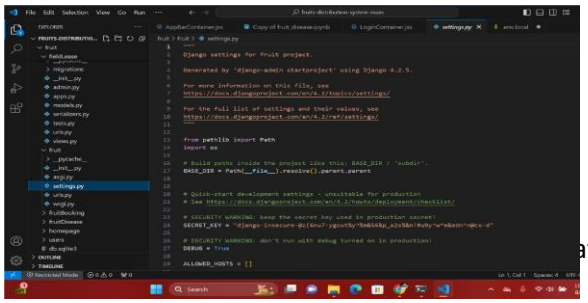


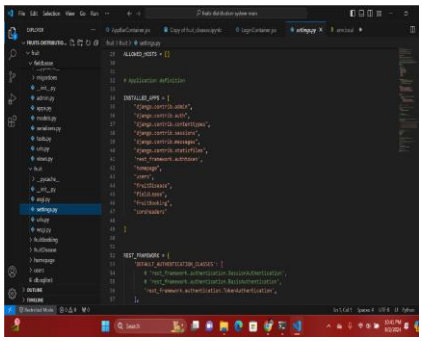


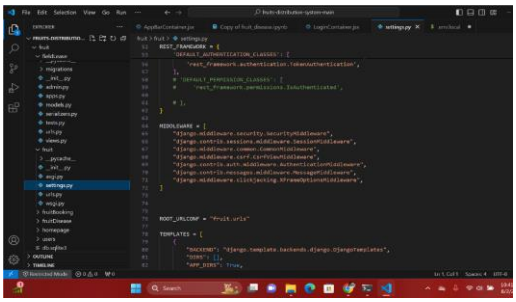


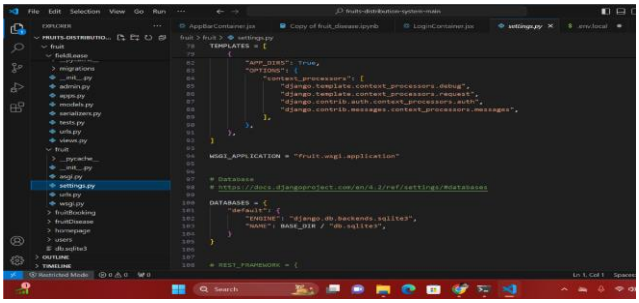


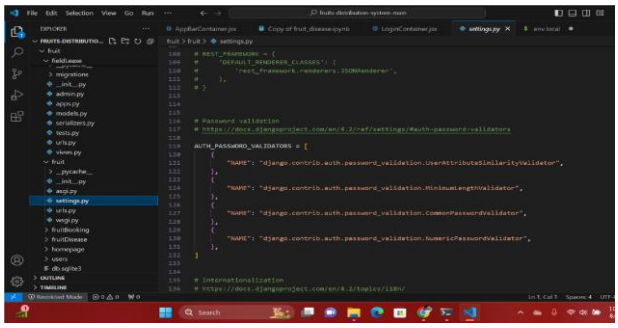


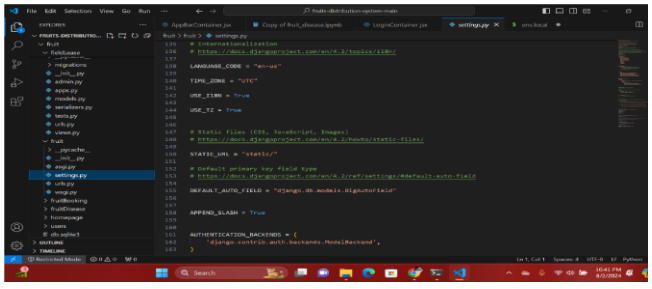


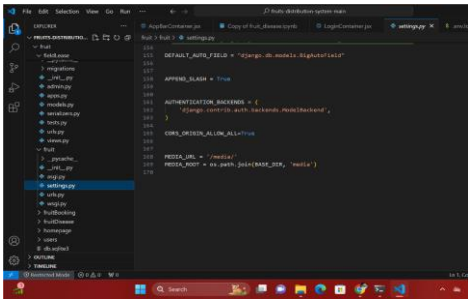


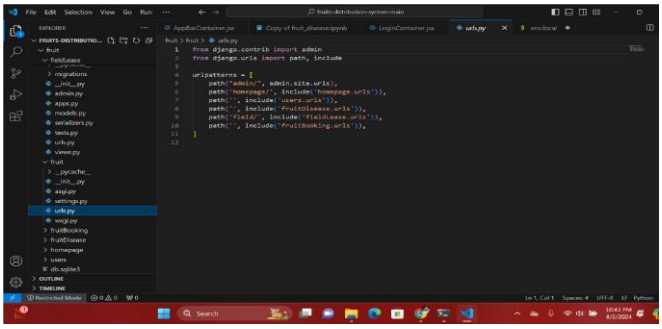


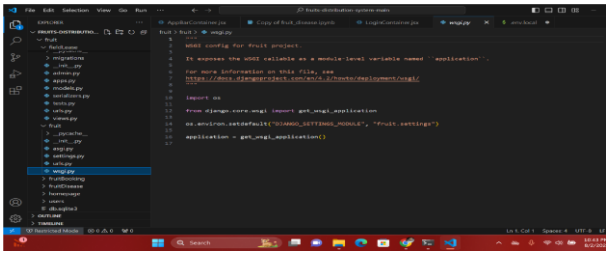


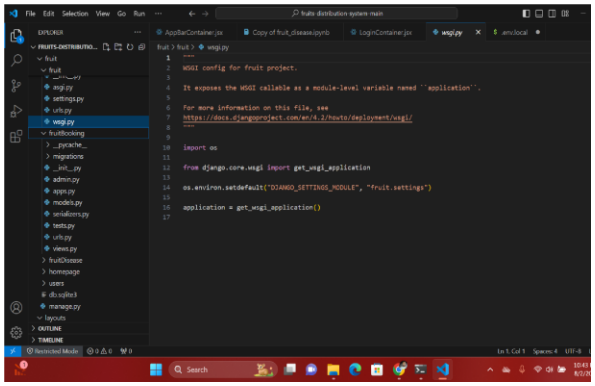


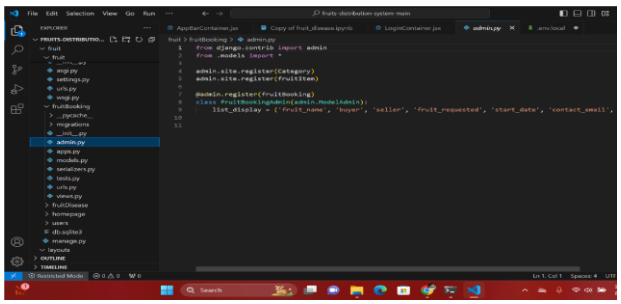


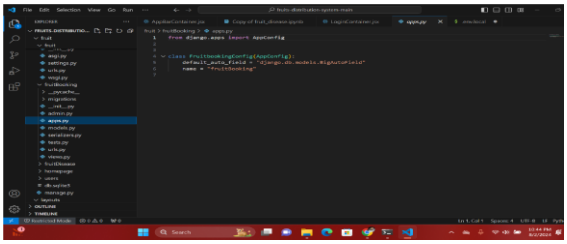


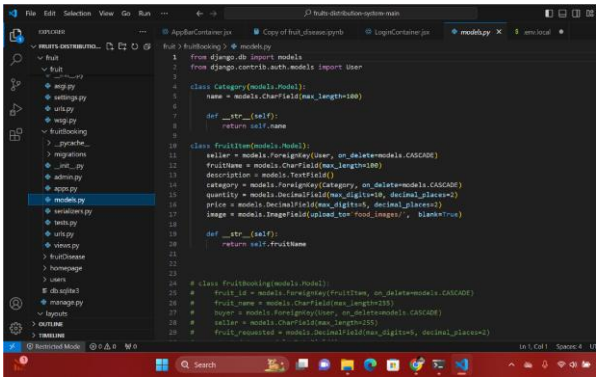


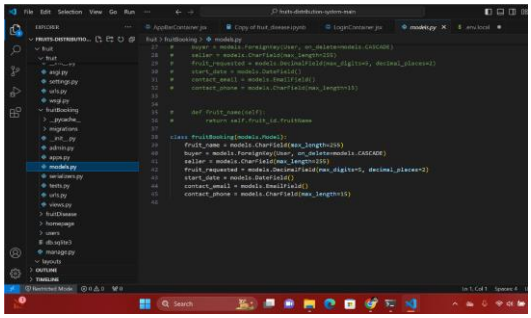


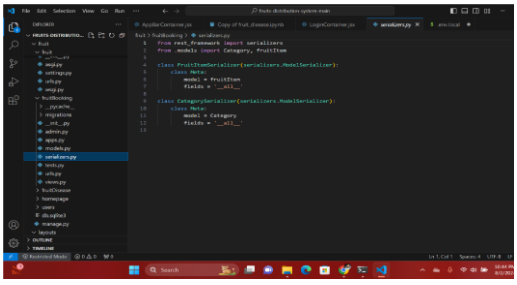


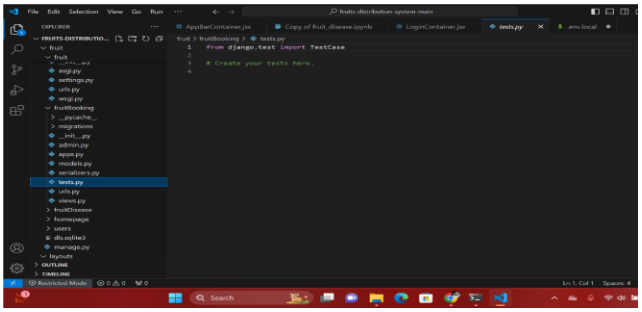


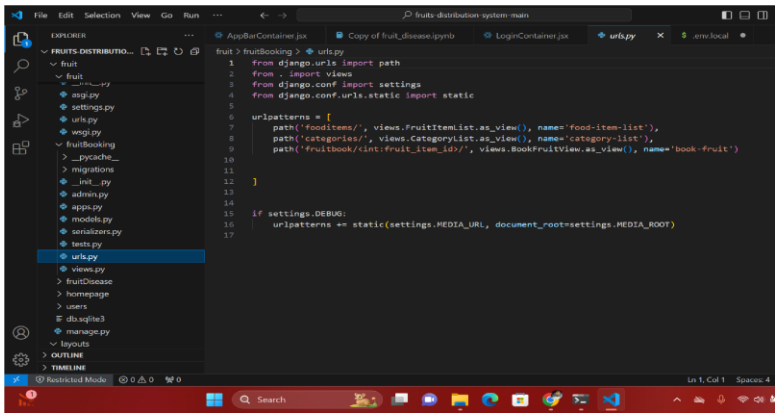


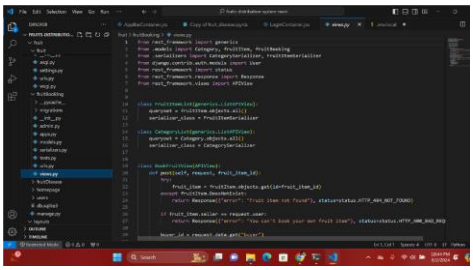


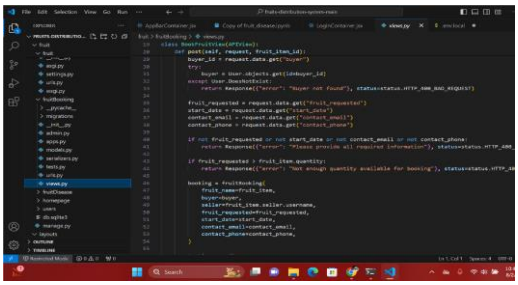


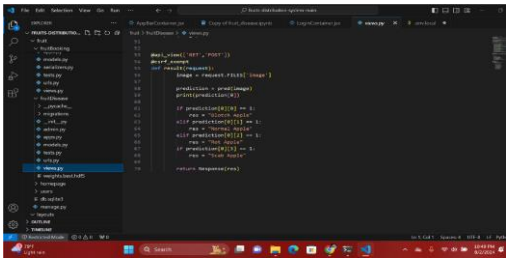


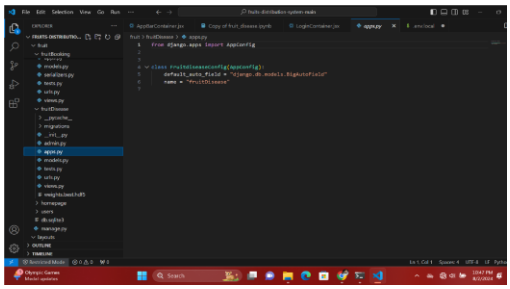


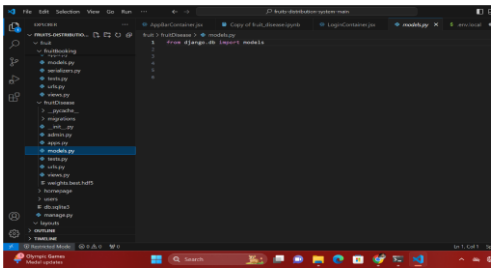


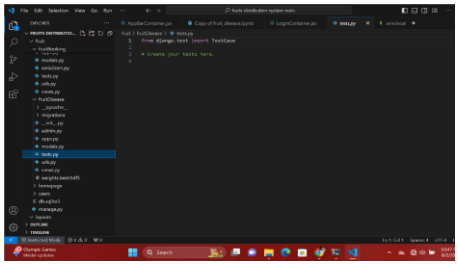


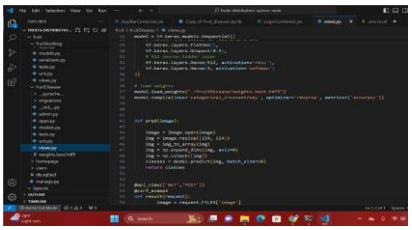


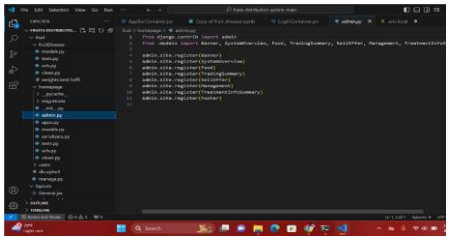


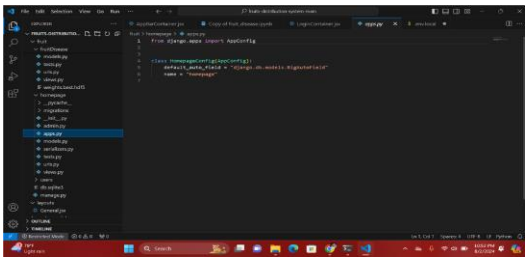


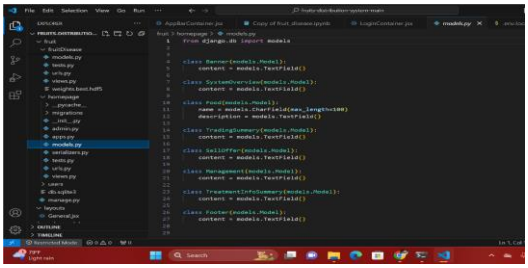


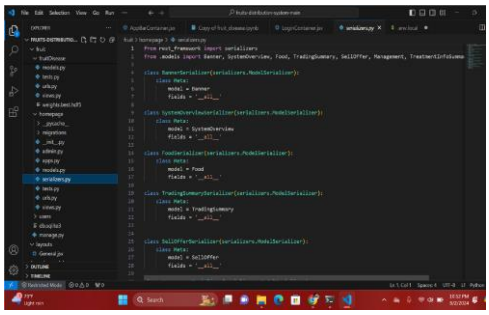


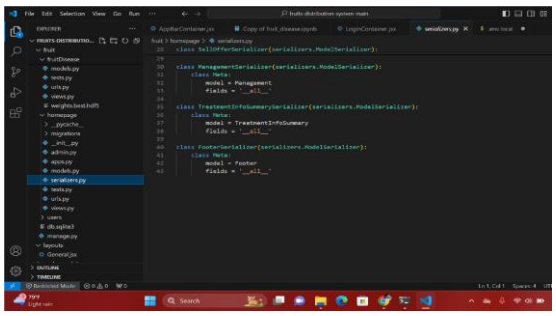


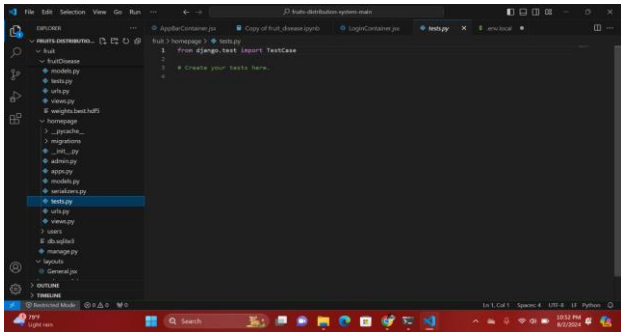


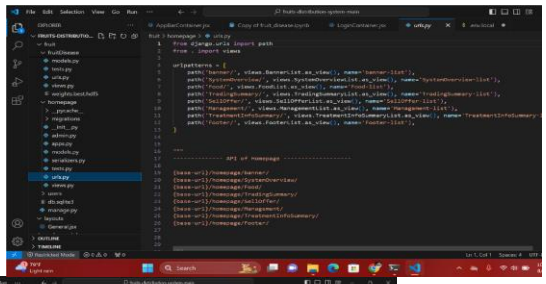


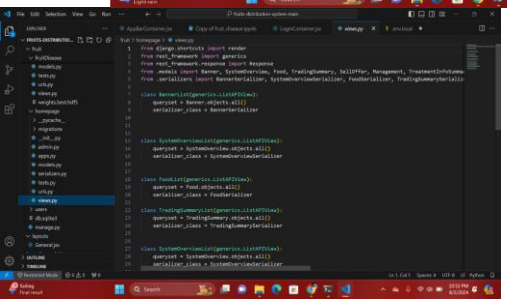


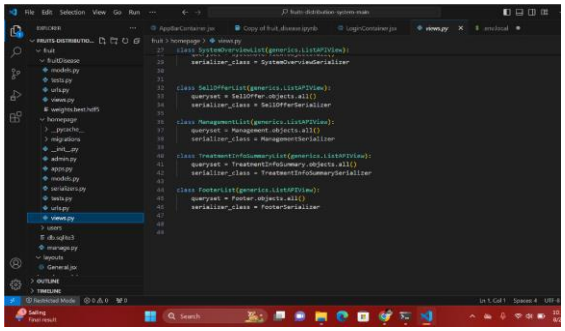


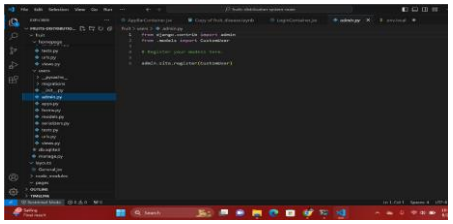


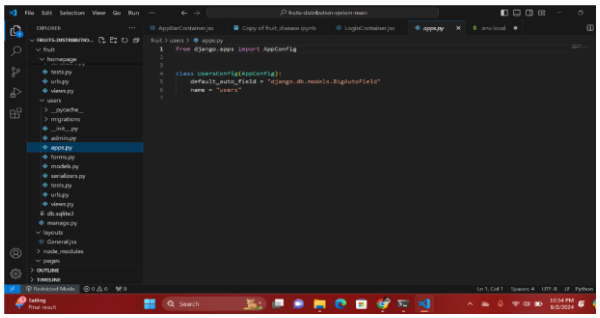


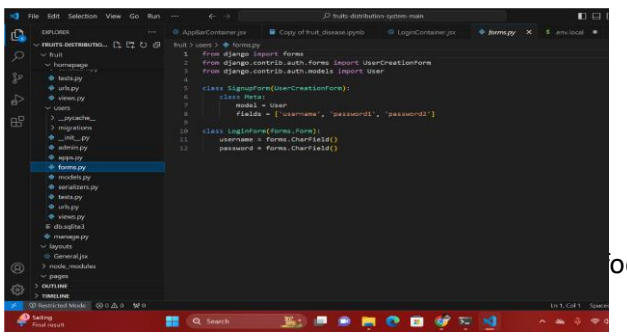


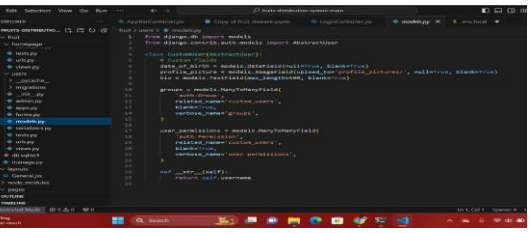


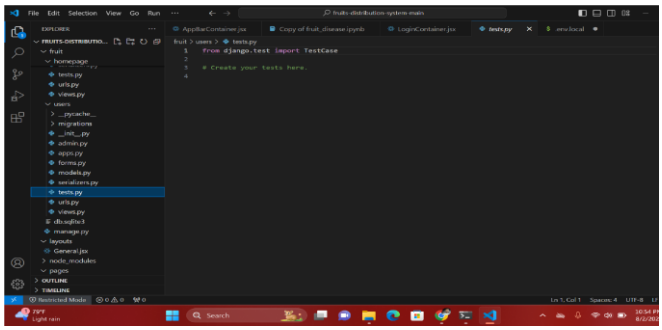


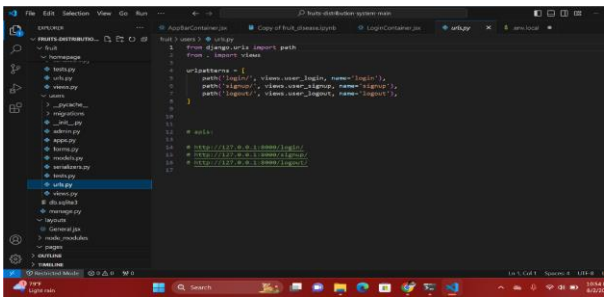


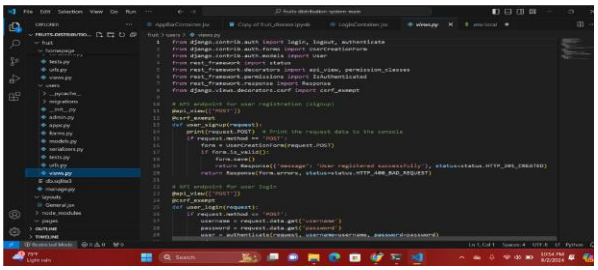


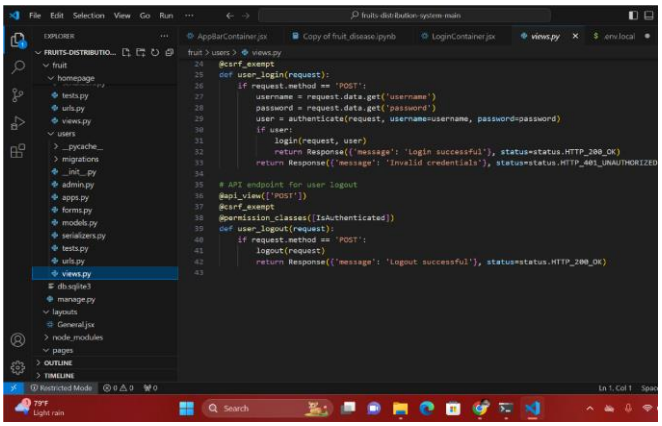


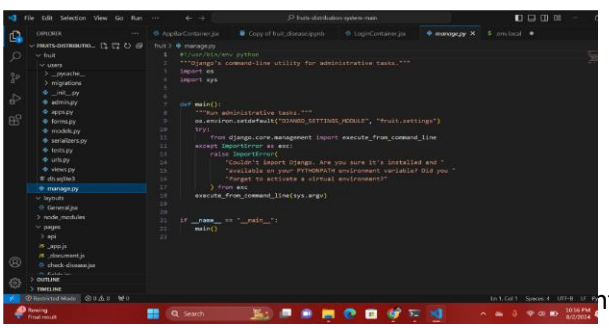


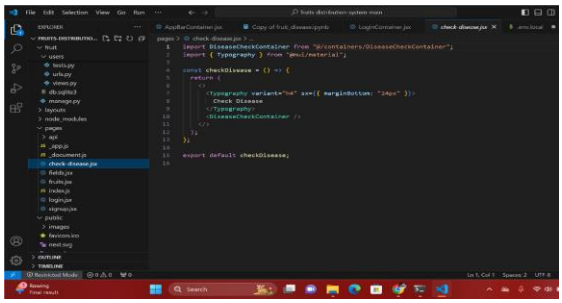


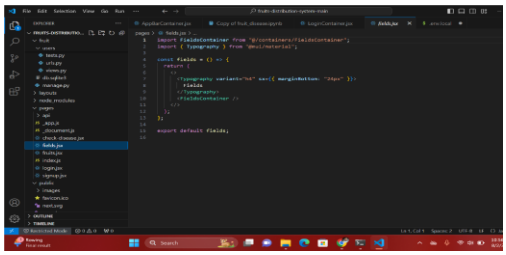












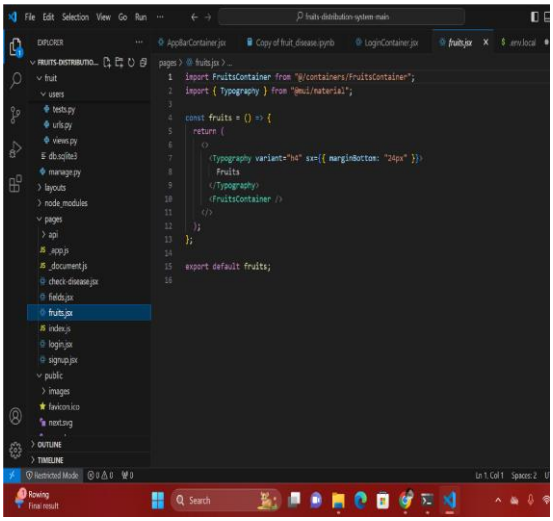


Figure : 9

10.2 Probability problem break down

To solve the probability problems associated with an AI-powered web tool for apple fruit disease detection, breaking the problem down into smaller, more manageable components is essential. Probability issues in these sorts of systems usually have to do with making decisions in the face of ambiguity, trusting the correctness of the model, and predicting sickness. By breaking down probability problems associated with an online fruit disease detection application, we can address important challenges such model accuracy,

dependability score, lifespan, handling rare diseases, and error rate. Solving these problems means implementing robust criteria for evaluation, calibration techniques, performance monitoring, and adaptation methods to ensure that the model remains valuable and reliable in diagnosing fruit illnesses.

- **Data security and protection**: Making ensuring that material that has been scanned, especially source code and configuration files, is managed appropriately to prevent unauthorized access or disclosure. utilizing strong encryption methods to safeguard data both in transit and at rest, particularly when sending scan results or storing them in databases.
- **Debugging and Testing:** When creating online or mobile apps, identifying and fixing problems, such as front-end and back-end flaws, is essential. Automation testing is utilized in all JavaScript, React JS, Python and MySQL developing processes to help identify and correct errors.
- **Quality Control and Testing:** To find and fix any potential issues or flaws, thorough testing and quality control methods are used throughout the development process. Testing was done on user acceptability, effectiveness, and usability. Finding and fixing vulnerabilities was one of the objectives of security testing. The modifications that were required to guarantee a stable, high-caliber system were discussed with administrators and other relevant parties.

10.3 Prioritization while developing

Prioritization	**Requirements and Explanation**
Core Functionality	Two primary components comprise the basic functionality of an AI fruit distribution and illness detection system: the Fruit Distribution System and the Fruit illness Detection System. Within the system, each has distinct roles and responsibilities to carry out. The Fruit Disease Identification System requires uploading photographs, utilizes artificial intelligence to detect illnesses, and provides useful feedback. On the other hand, the Fruit Distribution System concentrates on inventory management, order processing. Collectively, they establish a holistic approach to fruit distribution management and fruit quality assurance via sophisticated disease detection.

UX	We should prioritize usability, clarity, and efficiency in the UX design of the Fruit Distribution System and Fruit Disease Detection System. Purchasing and managing fruits using a simple interface for users as well as administrators is the main goal of the fruit distribution system. Providing concise, useful data from image processing in a simple .
Security and Data Management	Strong data encryption, safe authentication, and long-lasting user access restrictions. To continually enhance the usability of the online application, evaluate it with actual users and get their comments. Data integrity and user protection need to be given careful consideration. Maintaining data security and safeguarding admin user information requires careful consideration of appropriate access controls, secure authentication of users, and strong data encryption.
Optimization Performance	Performance enhancement, such as accelerating efficient caching techniques and raising the network's coding productivity, improved the effectiveness of the web-based application.
Integration with DL models with accuracy	Using CNN models of Deep Learning to obtain the most accurate prediction, which may aid in creating a web application for data classification into four classifications.
Quality Assurance and Testing	I'm currently working on a web application. For this project, I have a ton of code huge write. The code has a few errors. Automated testing was used by me to solve this problem. Through automated testing, the project's quality is verified. Thus, before the scheduled release, search for and discover errors.
User Feedback and Continuous Improvement	User input is essential to any program since it enhances overall software functioning and helps the design team grasp new ideas. Correcting mistakes and improving the program's functionality when feedback is given is essential.

CHAPTER 11
Testing

Project Name	Seasonal Fruit Distribution system		
Name of product	Seasonal Fruit Distribution system Web App		
Product description	Seasonal Fruit Distribution system Web App		
Project description	JavaScript, React JS, Python, MySQL.		
Project duration	Project Type	Testing/ Verification	
	Start date	End date	

11.1

Enumerate objectives constraints Before the test strategy is put into action, identify the key players and obtain their permission. Give a clear indication of the acceptance criteria for each testing stage.

11.2 Unit Testing

To conduct unit tests, it is recommended to integrate the structure as an entire unit and test every module separately. Unit testing keeps the software's architecture, which is the smallest part of each module, at the center of verification efforts. Module testing is another term for this. Each system module is examined independently. Verify that this method is compatible with every browser.

11.3 Validation Testing

Sometimes referred to as assurance of software quality, software testing uses validation and verification procedures to make sure a system meets requirements and performs as intended.

11.4 Integration Testing

Integration testing addresses the issues around the two issues pertaining to program generation and inspection. A set of high-order assessments are performed after software integration. The main objective of this testing technique is to use unit-tested components to build a program structure that complies with design criteria.

11.5 TEST CASES

Table 7: Test Case

Case Id	CASE NAME	Expected Result	Actual Result	Result (Pass/Fail)
1	User Name	Incorrect or empty names will result in a notice being shown.	Incorrect or empty names will result in a notice being shown.	Pass
2	Password	A notification will appear if the password is invalid or empty.	A notification will appear if the password is invalid or empty.	Pass
3	For Sign Up	A notification will appear when email, name and password in invalid or empty.	A notification appears when field is invalid or empty.	Pass
4	Booking Fruits Items	Add booking fruits item and show in admin dashboard	Fruits booking Details shows in admin dashboard	Pass
5	Booking Land lease Items	Add booking land lease item and show in admin dashboard	Land lease booking Details shows in admin dashboard	Pass

6	Admin panel	Fruits & land details can add or edit in admin dashboard	Admin can add & edit all details	Pass
7	Choose diseases images	Click button of choose images then give the access to select of computer images.	After clicking button of choose images then give the access to select of computer images.	Pass
8	Detection Fruits diseases.	Detection 4 classes of disease using apple diseases images.	Detection 4 classes of disease using apple fruits images accurately.	Pass

12-Lesson

12.1

User	Train	Time	Comment
Users or Clients			

12.2 Big Bang Implementation

Rather of using a trial or parallel phase of implementation beforehand, the Big Bang strategy for execution launches the entire system at once. All web app users—especially the admin—must get comprehensive instructions in order to arrange the detection of fruit infections. Before exploring the concept, I started playing with the code. I evaluated CNN models which were used for accuracy. When all was said and done, I assessed the correctness of the procedure. I considered the accuracy and decided which would be most useful for my purposes. Research on the illnesses that affect apples has demonstrated how accurate this method is in identifying fruit diseases.

12.3 Scaling

AI fruit distribution and disease detection system scaling requires a number of guidelines and solutions to manage growing workloads, more datasets, and increasing complexity. For adaptable architecture, make use of load balancing, micro service architectures, and cloud services. Use effective data transmission lines, caching, and horizontal scalability to handle massive datasets and heavy traffic. To efficiently handle AI jobs, make use of distributed instruction, cutting-edge computing, and algorithm optimization strategies. Utilize CDNs and caching to maximize performance; use access restrictions and encryption to guarantee security and compliance. Create intuitive user interfaces with strong feedback systems to manage large user counts and enhance system usefulness. You may make sure that the systems for fruit distribution and fruit disease identification scale well to meet increasing needs and complexity by paying attention to these factors.

12.3.1 Design of scaling

When developing a scalable AI web application for the detection of apple fruit illness, several architectural layers and considerations need to be made. The scalability of an AI-powered fruit disease detection web application depends on creating a robust framework that can handle heavy loads, maintain accuracy, and provide exceptional speed.

12.3.2 Testing Performance

Each activity performed by the scanner, including analysis, susceptibility detection, disclosure, and both static and dynamic evaluation, requires the creation of a little software. Instruct the development and architecture teams on how to design systems that are efficient and scalable. This means understanding components like caching, horizontal scalability, database efficiency, and performance monitoring.

12.4 Experiment Result

Based on images of diseased apples, algorithms anticipated the discovery of several fruit diseases. As such, I used a variety of strategies. Before determining the best plan of strategy for the trial, I thought through and evaluated a number of choices. Attempting to raise the

standard of my creations, I experimented with several methods. utilized the public datasets from Kaggle for the four fruit illnesses that were the subject of the lesson. I took advantage of the existing dictionaries, content categorization techniques, and Python tools that were already available. In order to apply deep learning towards the task of categorizing images of fruit diseases into four groups and creating a web-based program to test whether or not distinct apple fruit circumstances can be identified from photos of diseased fruits, this study uses Python DL sections for identification.

12.4.1 Descriptive Analysis

My classification strategies had an impact on the outcomes I obtained. I have used CNN deep learning techniques to identify the exact locations of fruit illnesses in photos. I employ CNN's deep learning techniques, which have shown promise in accurately diagnosing illnesses in apples. The same dataset, which included both mine online dataset and data from Kagle sources, was shared by every one of the models. After completing the dataset process, I used Mat-lab's pre-made libraries to evaluate the approaches' viability. I also predicted fruit illnesses using web prototypes.

Fig: Classification report & Confusion matrix

13.1

A fruit distribution and disease detection system implemented in AI can accomplish a number of important goals. These goals include boosting productivity, enhancing user experience, protecting data integrity, and using AI to get useful insights. Simplify order processing and inventory management. Improve user interface and provide customized purchasing experiences. Real-time findings and excellent detection accuracy can be attained. Provide actionable information and an intuitive user interface. Achieving these goals will improve overall user happiness and operational efficiency, guarantee that both systems run efficiently, and offer insightful information.

13.1.1

Rates of accomplishment for any objective are determined by setting clear, measurable objectives and regularly assessing performance with respect to these goals. Reaching these objectives means that users can be sure that the AI-powered web software for diagnosing fruit illnesses is accurate, scalable, secure, and trustworthy.

13.1.2 How much better could have been done

When developing changes to the AI-powered web-based program for diagnosing fruit illnesses, it's crucial to consider the model's accuracy, speed of instant execution, adaptability, convenience, data administration, flexibility, affordability, and regulatory compliance. If these improvements are implemented, the app's effectiveness will increase and its impact on illness identification will grow. Along with managing larger loads, providing a better user interface, and upholding strict safety and compliance standards, it will also perform better.

13.1.3 Why it could not be done

While artificial intelligence (AI) has great promise for enhancing the diagnosis of illnesses in apples, many barriers may hinder its practical use. These encompass limitations on data accessibility and quality, processing capacity, technical complexity, compatibility with existing systems, regulatory compliance, and budgetary constraints. To tackle these problems, a multifaceted approach that includes improved data collection, complex model building, and effective stakeholder interaction is required. Continuous innovation and adaptation are crucial to overcoming these challenges and making the most of AI in this industry.

13.1.4 Which objectives have been missed

A web-based application that use artificial intelligence (AI) to diagnose illnesses in apples failed to meet its objectives due to a variety of factors, including issues with user interface and integration, issues with data, limitations in technology, and a lack of resources. In order to address these gaps, a great deal of effort needs to be done to ensure scalability, optimize user interface design, improve real-time processing, increase model accuracy, and maintain strict guidelines for data protection, performance tracking, and regulatory compliance. By identifying and resolving these problems, the AI-based sickness detection system may operate more efficiently and effectively.

13.1.5 Why these objectives have missed

Missed goals for an AI fruit distribution and illness detection system are frequently the consequence of many difficulties and traps that might occur in the phases of design, implementation, and scale. Recognizing potential reasons for these goals' failure might aid in pinpointing areas in need of development. Improving system architecture, making significant infrastructure investments, guaranteeing user-friendly interfaces, and upholding strict security and compliance requirements are all necessary to address these problems. Including user feedback and upgrading models and processes on a regular basis can also assist to better achieve the goals.

13.1.6 What could have been done to complete those objectives

There are a number of proactive tactics and methods that may be used to effectively achieve the goals for the AI fruit distribution and fruit disease detection systems. Make optimal use of model complexity and large datasets. Make the UI simpler and offer practical advice. Make it easier for systems to share data and integrate seamlessly.

By taking care of these issues, the systems may successfully accomplish their goals and raise user happiness and operational efficiency.

13.2

If goals of an AI fruit distribution and illness detection system remain unfulfilled or unobserved, there may be underlying problems with the system's conception, execution, or administration. Limited real-time processing and poor detection accuracy. Lack of useful information and a complex UI insufficient handling of data and technology. No integration of user feedback or routine model changes. Ineffective data exchange and poor system integration

13.2.1 Including software and documentation

Regular inspection, bug patches, and feedback from customers would have all been part of the procedure for developing software in order to guarantee the efficacy and usability of the program. The evidence, which included technical guidelines, servicing instructions, and user manuals, had to be comprehensive in order to guarantee easy understanding and future updates.

CHAPTER 14

14.1

To assure the project's success, a pre-project stage for a fruit distributing and fruit disease detection system using AI entails careful planning, a feasibility evaluation, and early setup. Before real development starts, this phase is essential for determining needs, risks, and resources. Clearly state the objectives and limitations of the project. Evaluate viability in terms of operations, economy, technology, and law. Determine the requirements—both functional and non-functional—of the stakeholders. Determine, evaluate, and prepare for any hazards. Make a budget, schedule, and resource plan. Create early system architectures, choose technology, and develop prototypes. Get approval from stakeholders, compile documentation, and get ready for the project to begin. You lay a strong basis for the effective creation and use of automated AI fruit distribution and illness detection systems by taking care of all of those pre-project tasks.

14.2 Review

Regular reviews are essential to assess project progress, identify deviations from the plan, and ensure that everything is proceeding on schedule. I may evaluate if the web application has succeeded in achieving its goals and determine how any adjustments or improvements are required by using these evaluations. Frequent feedback from participants and readers can help identify areas that require development and promote informed choice-making.

14.3 Lessons Learned

Before making this web application, I did a comprehensive analysis. Prior to beginning work on this application, I researched system design and analysis. The most challenging stage of system development is this one. Precise programming cannot be produced without analysis. I then learned MySQL, JavaScript, React JS, and Python. Gaining understanding is difficult. Even with their extensive usage, JavaScript and MySQL can be difficult to understand and operate in their most basic forms. From the moment as I start learning, I recognize my weaknesses so that I can be given consideration for further responsibilities.

14.4 Problem Faced

When developing an AI-powered online application for the detection of fruit illnesses, it requires expertise to get beyond challenging roadblocks related to data quality, model accuracy, quick processing, growth, user experience, regulatory compliance, and cost. To get over these challenges, a methodical approach will be required, one that includes improving data quality, ensuring high availability, optimizing computing resources, and designing userfriendly interfaces. By addressing these problems, the program's stability and effectiveness might be significantly improved.

14.5 Problems That are solutions

These are difficult software challenges. Even if I make web apps, nevertheless. While putting together the project, I ran into a few common problems, but nothing particularly challenging. However, coming up with concepts that might actually be effective could be difficult. Apple fruit disease detection system websites may enhance customer happiness, increase operational efficiency, and establish credibility in the cutthroat agriculture sector by proactively dealing with these issues. Consistent observation of user input and performance indicators will help guide future enhancements and adjustments.

CHAPTER 15

Conclusion

15.1 Summary of the project

The whole results of my project, which is titled "Seasonal fruits distribution system," may be seen here. This article goes over the process in detail of turning the idea into a working website. For system users, there are two modules like: User & Admin. The project is to create an integrated system that uses AI to improve procedures for detecting fruit diseases as well as fruit delivery. Better fruit quality and productivity will result from the system's automated inventory control, accurate and early disease diagnosis, and efficient fruit distribution management. Some fruit illnesses may be early to identify thanks to deep learning technology. This website will require user input in order to identify fruit ailments. As a result, it will have two main components. Website creation is the first, and apple fruit disease detection is the second, which makes use of deep learning.

15.2 Goal of the project

The primary aim of this endeavor is developing a user-friendly online application that has the ability to precisely diagnose and categorize fruit illnesses by using certain photographs or symptoms. Our goal is to use AI technology to give farmers a reliable and effective tool that might help with early disease identification, lower the risk of financial losses, and enhance the welfare of agriculture. With a focus on precision, user comfort, scalability, and conformity, the effort aims to enhance the productivity of agricultural operations and the overall health of fruit disease development. It also aims to provide farmers and veterinarians with a helpful resource.

15.3

project has achieved great success. The principal actions are:

- User can Sign In & Sign Up for access the web app
- Details about company
- User can see Fruit list and booking of items.
- User can see Land list and lease the land with booking no of items.

- Fruit disease Detection system using DL.
- Admin can edit & add fruit and land list and store in database.

15.4 Documentation

The paperwork most likely contained the following stages, assignments, and plans:

- **Prior-project details:** Preliminary requirements collecting, viability evaluations, and project conceptions are examples of records from earlier projects that may be included.
- **Project Plan:** A written document outlining the objectives, constraints, timetable, required resources, and risk-reduction strategies for a project.
- **Technical Specifications:** Technical requirements are a detailed collection of rules that relate to the characteristics, capabilities, and design of a web-based application that is used in a system for managing travel or tourists.
- **User Documentation:** Offers guidance and recommendations on how to utilize the software effectively to all parties involved, including the person in charge who looks for vacation packages or locations.
- **Inspection and Assurance of Quality Instruction**: Document test methodologies, scenarios, and outcomes to ensure the program meets quality criteria.
- **Deployment as well as care plans**: Documentation outlining the necessary steps is provided, along with plans for the implementation, frequent consumption, and upgrading of apps.

15.5 Value of the project

With implications for several stakeholders and facets of the fruit sector, the fruit distribution and fruit disease detection system in AI is valuable in many ways. In the end, the project will result in a more productive, adaptable, and sustainable fruit business by utilizing AI in these two crucial areas and offering substantial benefits in the functional, economic, and social domains.

15.6 My Experience

I may provide my skills in project management, design, or software development to the web application project for the tourism management system. My background has equipped me with the skills necessary to work with stakeholders, efficiently manage project timelines, get beyond technical challenges, and implement solutions designed specifically to anticipate fruit diseases of all kinds. I have improved my problem-solving, collaboration, and documentation abilities to further my career.

References

[1] M.M. Islam, M.A.A. Adil, M.A. Talukder, M.K.U. Ahamed, M.A. Uddin, M.K. Hasan, S. Sharmin, M.M. Rahman, and S.K. Debnath, "DeepCrop: Deep learning-based crop disease prediction with web application", 2023. *Journal of Agriculture and Food Research*, *14*, p.100764.Duckett, J., 2001. *Web Design with HTML, CSS, JavaScript and jQuery*. 1st ed.

[2] W3schools.com. 2021. *JavaScript Tutorial*. [Online] Available at: <https://www.w3schools.com/js/default.asp> [Accessed 17 December 2021].

[4] W3schools.com. 2021. *What is Bootstrap*. [Online] Available at: <https://www.w3schools.com/whatis/whatis_bootstrap.asp> [Accessed 17 December 2021].

[5] Learn about React Js, available at <<https://youtu.be/4UZrsTqkcW4>>, Last accessed on 20-01-2023 9:48 PM.

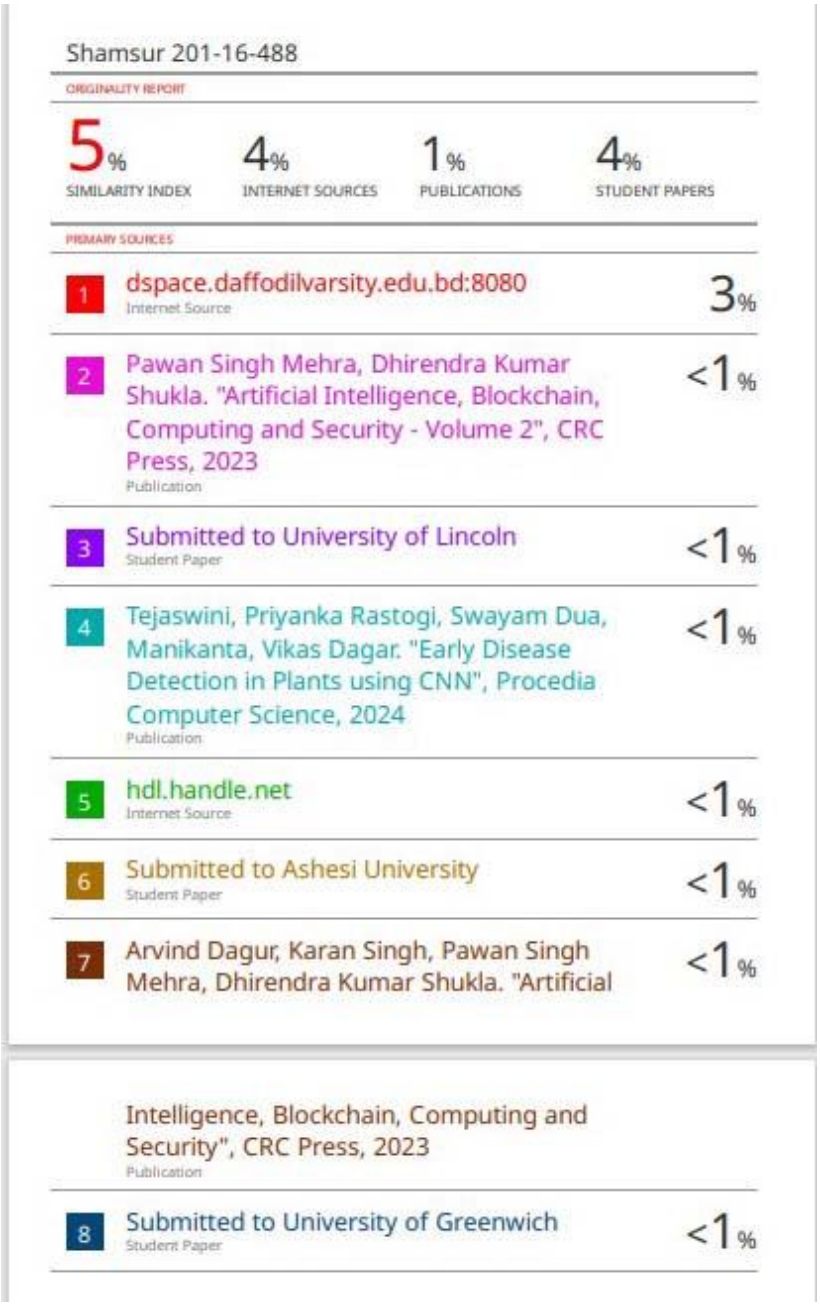
[6] Learn about Node Js, available at <<https://nodejs.org/en/>>, Last accessed on 15-01-2023 5:00 AM.

[7] Learn about HTML and CSS, available at <<https://youtu.be/-8ORfgUa8ow>>, Last accessed on 04-012023 4:23 AM.

[8] Learn about JavaScript, available at <<https://youtu.be/2Ji-clqUYnA>>, Last accessed on 21-012023 5:08 AM.

[9] S. Ashok, G. Kishore, V. Rajesh, S. Suchitra, S.G. Sophia and B. Pavithra, "Tomato leaf disease detection using deep learning techniques", 2020, June. In *2020 5th International Conference on Communication and Electronics Systems (ICCES)* (pp. 979-983). IEEE.

[10] H. Durmuş, E.O. Güneş, and M. Kırcı, "Disease detection on the leaves of the tomato plants by using deep learning", 2017, August. In *2017 6th International conference on agro-geoinformatics* (pp. 1-5). IEEE.



Shamsur 201-16-488

ORIGINALITY REPORT

5%	4%	1%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	3%
2	Pawan Singh Mehra, Dhirendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security - Volume 2", CRC Press, 2023 Publication	<1%
3	Submitted to University of Lincoln Student Paper	<1%
4	Tejaswini, Priyanka Rastogi, Swayam Dua, Manikanta, Vikas Dagar. "Early Disease Detection in Plants using CNN", Procedia Computer Science, 2024 Publication	<1%
5	hdl.handle.net Internet Source	<1%
6	Submitted to Ashesi University Student Paper	<1%
7	Arvind Dagur, Karan Singh, Pawan Singh Mehra, Dhirendra Kumar Shukla. "Artificial Intelligence, Blockchain, Computing and Security", CRC Press, 2023 Publication	<1%
8	Submitted to University of Greenwich Student Paper	<1%