Potato leaf Early blight and Late blight disease detection system

Submitted by

Sahbaj Sarder
ID : 191-35-431
Department Of software Engineering
Daffodil International University

Supervised by

Ms. Nusrat Tasnim
Lecturer
Faculty of Science and technology
Department of software Engineering
Daffodil International University

Fall- 2022

A project (DS 431) submitted to fulfill the requirements for getting the degree of B.Sc. in Software Engineering

APPROVAL OF PROJECT

APPROVAL

This Project titled on "**Potato Leaf Early Blight & Late Blight Disease Detection**", submitted by **Sahbaj Sarder (ID: 191-35-431)** to theDepartment of Software Engineering, Daffodil International University has been accepted as

satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS

Dr. Imran Mahmud
Head and Associate Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Chairman

Afsana Begum
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1

Dr. Md. Fazle Elahe
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2

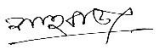
Mohammad Abu Yousuf, PhD
Professor
Institute of Information Technology
Jahangirnogor University

External Examiner

DECLARATION

By signing this document, I certify that this project was completed by me and supervised by Ms. Nusrat Tasnim, Lecturer, Department of Software Engineering, Daffodil International University. I further declare that this project is entirely original and that no portion of it has ever been presented anywhere for the award of any degree.

Submitted By:



Sahbaj Sarder

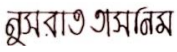
ID: 191-35-431

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University

Certified By:



Ms. Nusrat Tasnim

Lecturer

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University

ACKNOWLEDGEMENT

All praise is due to the All-Powerful Allah, who has always been good to my family and me, has supported me on this trip, and has given me the ability to finish this endeavor.

Additionally, I want to express my gratitude to Ms. Nusrat Tasnim Lecture, my honorary supervisor at the Daffodil International University's Department of Software Engineering, for always being there for me and guiding me in the correct direction. Without her cooperation, assistance, and support, I'm not sure if I could have completed this job.

I want to express my gratitude to my friends for always being willing to share their knowledge and for helping me learn by talking about our class assignments. Finally, I want to express my gratitude to my parents for their unwavering support and encouragement. They are my real role models.

ABSTRACT

Every year, potato farmers lose a lot of money, which leads to many illnesses that harm potato plants. The two most common Blight diseases are Early Blight and Late Blight. Farmers can save a lot of waste and save financial losses if they identify this illness early and administer the appropriate treatment. Early blight is caused by fungi, while late blight is caused by certain microorganisms. It's critical that you correctly categorize the type of disease on that potato plant because there are certain differences between the treatments for early and late blight. Convolutional Neural Network: Deep Learning will be used in the background to diagnose plant illnesses.

Table of Contents

LIST OF FIGURES

Chapter 1
Introduction

1.1 Project overview

Every year, potato farmers lose a lot of money, which leads to many illnesses that harm potato plants. The two most common Blight diseases are Early Blight and Late Blight. Farmers can save a lot of waste and save financial losses if they identify this illness early and administer the appropriate treatment. Early blight is caused by fungi, while late blight is caused by certain microorganisms. It's critical that you correctly categorize the type of disease on that potato plant because there are certain differences between the treatments for early and late blight. Convolutional Neural Network: Deep Learning will be used in the background to detect plant illnesses.

Here, we'll create a comprehensive deep learning project for the agricultural industry. Using a straightforward and conventional convolutional neural network architecture, we will develop a straightforward picture classification model that will classify potato leaf disease. We'll begin by gathering data, create the model, and then use Tensorflow to create a web application and publish it to Flask.

1.2 Project Purpose

From my own observations, I've observed farmers suffering significant losses in the potato industry. My family's primary source of income has always been farming. Both my uncles and my grandfather were farmers. Due to these two diseases, they suffer numerous losses each year. Before it's too late, early and late blight must be distinguished because they are typically difficult to recognize. This project was built with the intention of detecting or identifying the various diseases that might affect potato leaves. because a convolutional neural network can quickly classify them but our unaided sight cannot. If I tell you that the error of some pretrained neural network topologies is around 3%, you won't believe me. Being even lower than the top 5% mistake in human vision. The top five human errors in large-scale photos have been estimated to be 5.1%, which is higher than that of pre trained networks.

1.2.1 Benefits & Beneficiaries

- Detecting disease before it’s too late
- Farmers can rely on this model
- Easy to easy so that anyone can use it
- No more losses in potato agriculture

1.3 Stakeholders

A project stakeholder is "a person, group, or organization that will be influenced or affected by in a decision, activity, and outcome of the project," according to project management. As a stakeholder in this system, I have an administrator and a regular user. Discussion of stakeholders is below.

1.3.1 Admin

The system's primary authority is the admin. The system will be kept up by the admin. All system usage and activity will be under the administrative supervision.

1.3.2 User

The primary users of this system will be those who wish to use it. They can save time, take images, and then view the finished product.

1.4 Objectives

The main goals of the Potato leaf early blight and late blight disease detection system is to detect disease before it’s too late to be cured. The system will help to detect these disease using CNN and will help the farmers to protect their crops.

CHAPTER 2
REQUIREMENT ENGINEERING

2.1 Functional Requirements

A functional requirement outlines a system's function and the services that must be provided by the system. The system's major part is described by a functional requirement. Everything that a system or piece of software must accomplish, including its features and capabilities, is outlined. When precise circumstances are satisfied, it will define specific system behavior or function. The stakeholders must understand them clearly.

Listed below are the project's functional requirements:

2.1.1 Upload Image

FR1	Upload Image
Description	User must be able to upload photos to check whether the leaf is affected or healthy

Stakeholders	User

2.1.2 Show Result

FR2	Show result
Description	System should show the prediction
Stakeholders	User

2.1.3 Classification

FR3	Classification
Description	The system must be able to classify the images into 3 classes
Stakeholders	System

2.1.4 Image Extraction

FR4	Image Extraction
Description	The system must be able to Extract the photos to get the job done
Stakeholders	System

2.1.5 Saving trained model

FR5	Saving trained model
Description	System must be able to save a trained model and use it.
Stakeholders	System

2.1 Non-Functional Requirement

The system's performance and quality attributes are determined by non-functional needs. A standard set that is used to evaluate the specific operation of the system is presented by non-functional requirements.

The project's non-functional need is as follows:

2.1.1 Performance

NFR1	Performance
Description	When User search to perform a particular job then the outcomes must be appearing.
Stakeholders	User

2.1.2 Capacity

NFR2	Capacity
Description	System must be able to perform more than one predictions for various users
Stakeholders	Users

2.1.3 Reliability

NFR3	Reliability
Description	The system must be able to satisfy the functional requirements and fit them. The system needs to be updated frequently and desperately.
Stakeholders	Admin

2.1.4 Security

NFR4	Security
Description	All data needs to be protected from outside attack. Encryption Protection is one great solution. Authentication of every request should been soured.
Stakeholders	Admin

2.1.5 Maintainability

NFR5	Maintainability
Description	The system's admin may easily maintain every profile and update it with PacifiCare's information.
Stakeholders	Admin

2.1.6 Availability

NFR6	Availability
Description	The system should be available24 hours of a day(24x7)
Stakeholders	Admin

CHAPTER 3

SYSTEM ANALYSIS, DESIGN & SPECIFICATION

3.1 Development Model

Since this project is ongoing, I went with the agile model. I need to build my system with an open mind. Therefore, updating the system won't affect the rest of my system in any way. Every stage of this system's development is verified for efficiency. I must test every aspect of the project to ensure the system runs well. So I go with the agile model. With the use of this model, I can test the system as it is being developed, identify issues, and fix them. The agile methodology facilitates rapid development, frequent testing, easy system updates, and standardized product quality.

Figure3.1:Agile Model

3.2 Use Case Diagram

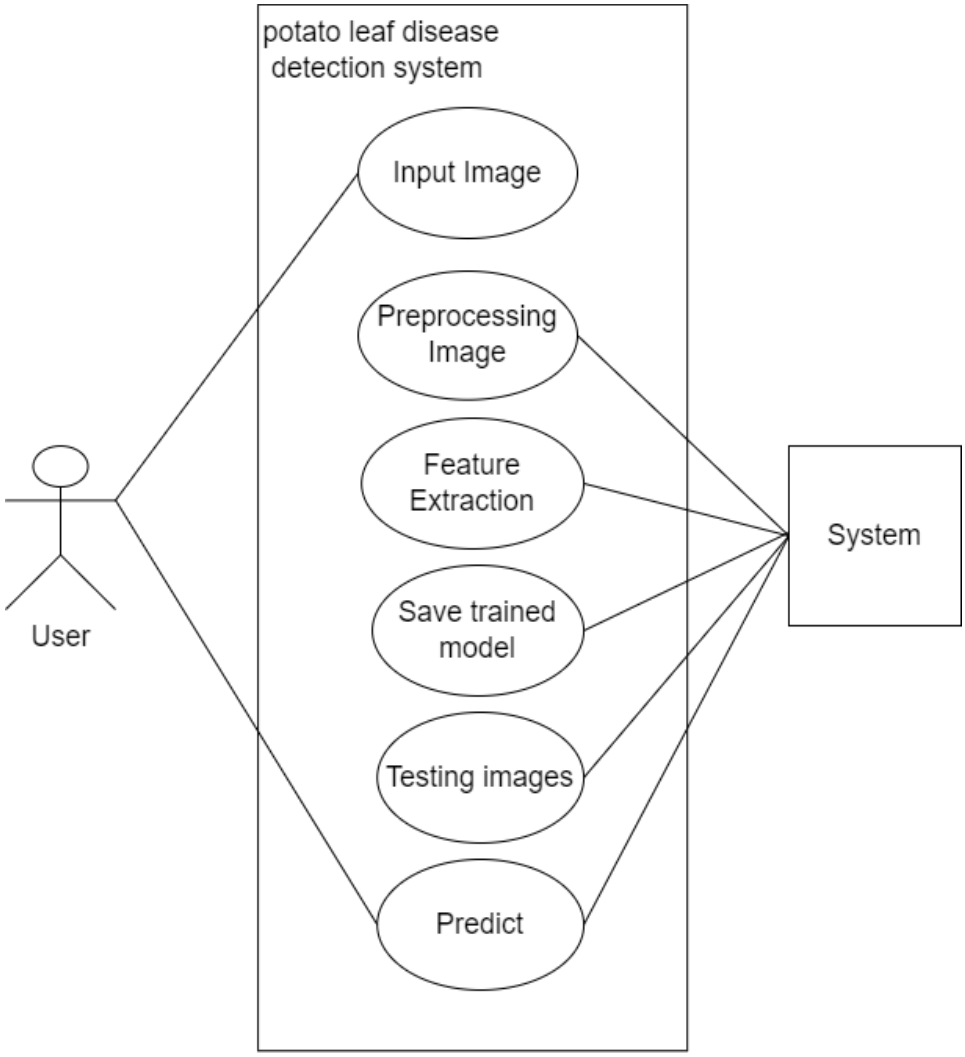


Figure3.2: Use Case Diagram

3.3.1 Input Image

Use Case ID	UC-01		
Use Case Name	Input Image		
Goal	Uploading Image to the system		
Preconditions	Use must have photos of the leaf		
Primary Actor Secondary Actor	User, system		
Trigger	Button		
Description/Main success scenario	Step	Action	
	1	To upload photos	
	2	To get predictions	
Post Condition	User may get result		
Alternative Flow	Click photos		

3.3.2 Preprocessing Image

Use Case ID	UC-02		
Use Case Name	Preprocessing Image		
Goal	To process the training image		
Preconditions	User must get good enough photos		
Primary Actor Secondary Actor	System		
Trigger	Button		
Description/Main success scenario	Step	Action	
	1	To process the training photos	
	2	To create a cnn model	
Post Condition	System can create model		
Alternative Flow	N/A		

3.3.3Feature Extraction

Use Case ID	UC-03		
Use Case Name	Feature Extractions		
Goal	Extracting the feature of the photos		
Preconditions	n/a		
Primary Actor **Secondary Actor**	System		
Trigger	n/a		
Description/Main success scenario	Step	Action	
	1	Photos to be extracted	
	2		
Post Condition	System must be able to extract photos		
Alternative Flow	N/A		

3.3.4Save trained model

Use Case ID	UC-04		
Use Case Name	Model saving		
Goal	Model must be saved		
Preconditions	Model must be trained		
Primary Actor **Secondary Actor**	system		
Trigger	Button		
Description/Main success scenario	Step	Action	
	1	Model trained and saved	
	2	Model must predict	
Post Condition	User may get correct predictions		
Alternative Flow	N/A		

3.3.5 Testing

Use Case ID	UC-03		
Use Case Name	Testing		
Goal	Test the images to get result		
Preconditions	System must get images to test		
Primary Actor **Secondary Actor**	system		
Trigger	N/A		
Description/Main success scenario	Step	Action	
	1	test	
	2		
Post Condition	System can test the images		
Alternative Flow	N/A		

3.3.6 Predict

Use Case ID	UC-04		
Use Case Name	Predict		
Goal	Predict the images to get result		
Preconditions	Images must be trained in model and saved in database		
Primary Actor **Secondary Actor**	User ,system		
Trigger	Button		
Description/Main success scenario	Step	Action	
	1	Predict the actual condition of images	
	2		
Post Condition	Users and system may be able to see the predictions		
Alternative Flow	N/A		

3.4 Activity Diagram

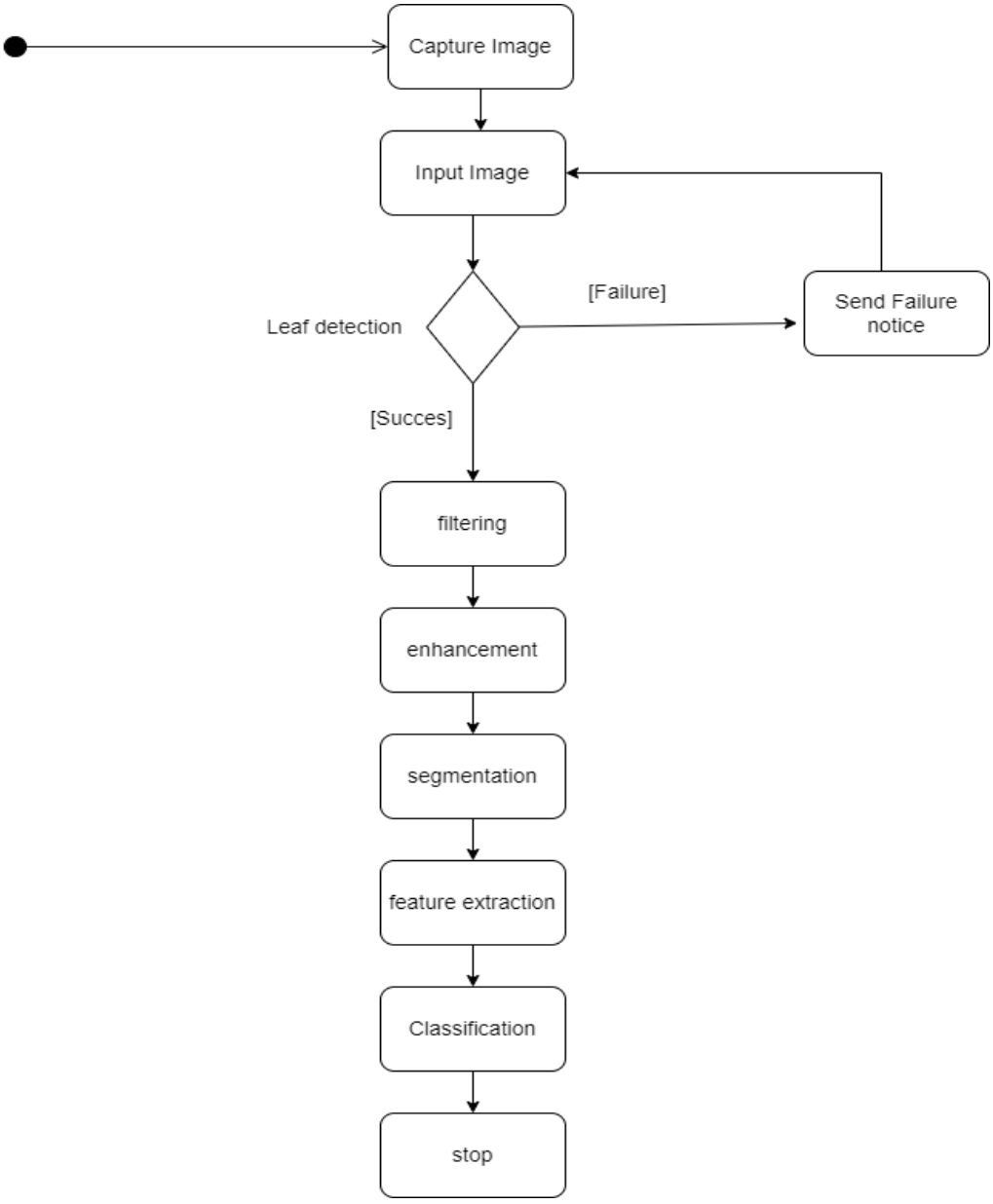
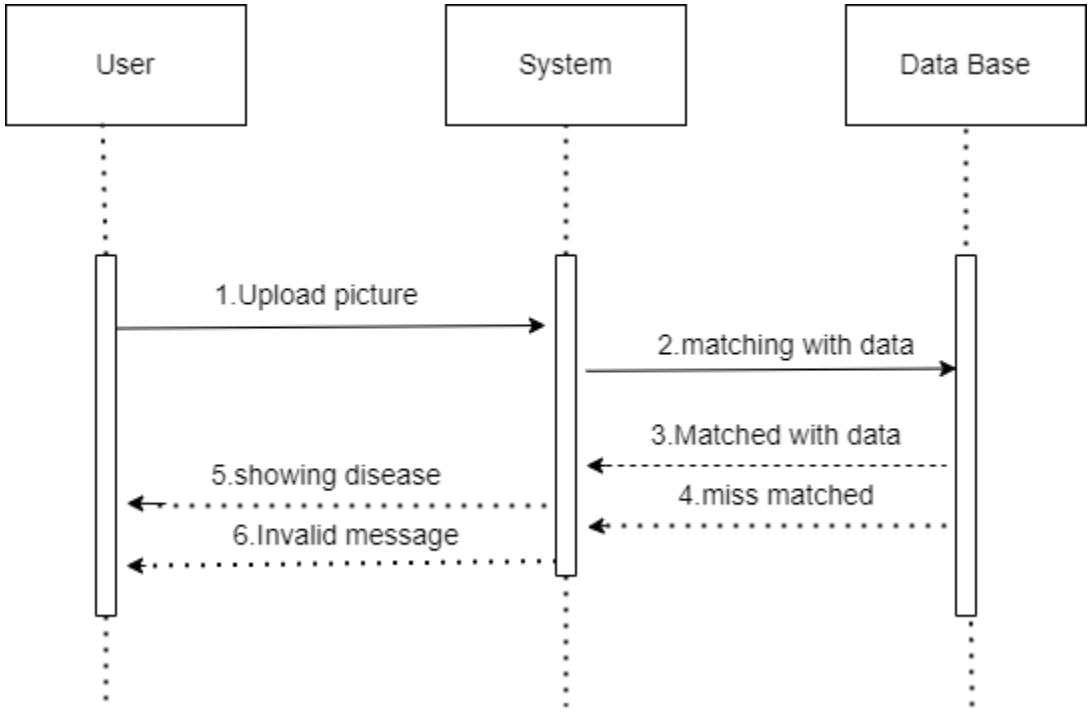


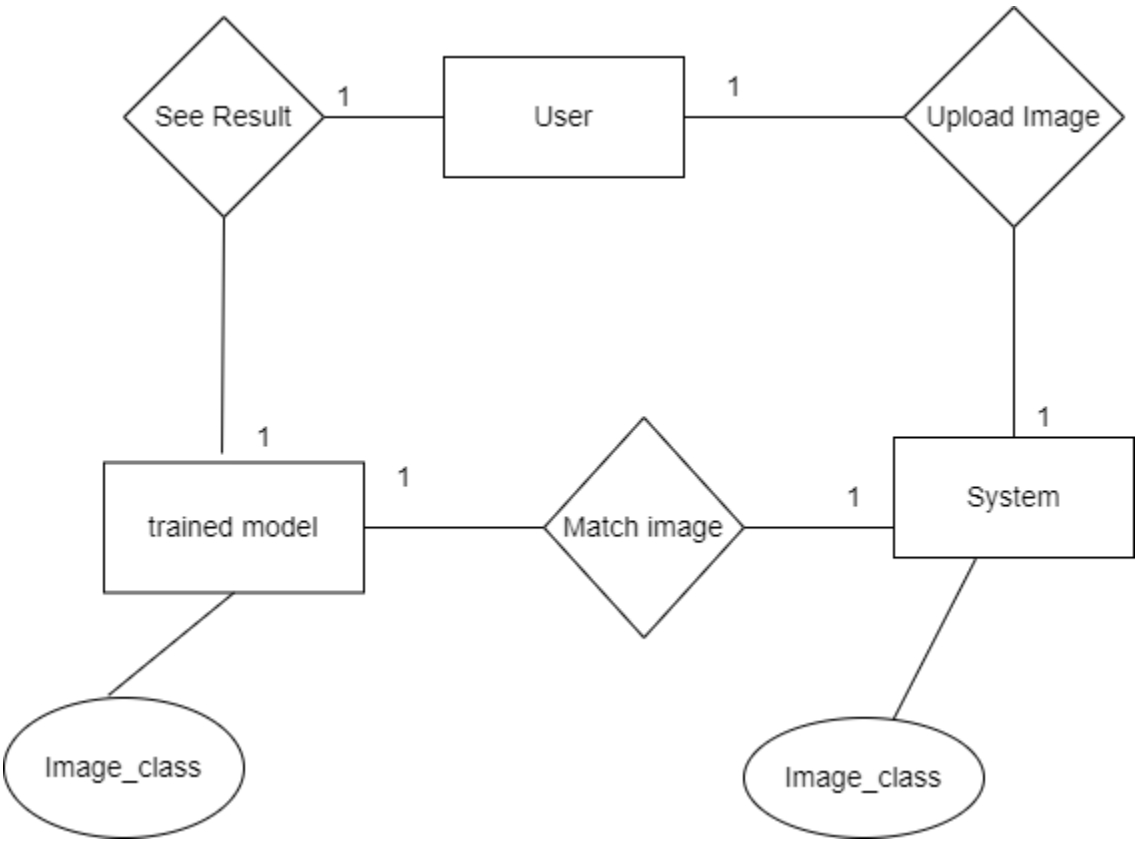
Figure-3.3: Activity Diagram

3.5 Sequence Diagram



Figure–3.4: Sequence Diagram

3.6 Entity Diagram



Figure–3.5: Entity Diagram

CHAPTER 4
SYSTEM TESTING

4.1 Feature Testing

Any application required to be periodically tested to detect fresh defects or issues, and the system needed to be updated frequently. In order to add new features or remove existing features or issues, feature testing is required.

4.1.1 Tested Feature

Feature	Priority	Description
Prediction	1	The user must get accurate prediction
Home	2	Home button must work
Upload photo	1	User must be able to upload photos
result	2	User must be able to see the result
refresh	1	Users must be able to refresh the page

Here, 1= High Priority, 2= Medium Priority, 3= Low Priority

4.2 Testing Strategies

4.2.1 Test Approach

I want to employ two distinct testing methods to guarantee that my project system is of a high quality. two types of testing: black box and white box

They mainly concentrated on structural and functional testing.

1. The system's functioning is tested via black box testing, which is the first step. Based on input and output, functionality is evaluated.

2. White box testing: White box testing is used to examine a system's structure, architecture, and workings.

4.2.2 Pass/Fail Criteria:

There are two types of testing criteria: Pass and Fail. Based on which output is correct and which output is incorrect, the Pass/Fail criteria are created.

- If the system crash it will be a failure.
- Only passing 100% in every criteria would be considered as pass.

4.2.3 Testing Schedule

Test Phase	Time
Testing plan create	1week
Unit testing	During developing time
Component testing	During developing time
Integration testing	1week
Testing user interface	1week
Load testing	1week
Performance testing	1week
Accessibility testing	1week

Table4.1: Testing Schedule

4.2.4 Traceability Matrix

Project Manager			**Business analyst Lead**	
QA leader			**Target implementation date**	
TM	**Functionality Activity**	**Requirement Description**	**Test Case Reference**	**Comments**
TM-01	Functional	Home	TESTCASE 4.4.1	
TM-02	Functional	About	TESTCASE 4.4.2	
TM-03	Functional	Capture image	TESTCASE 4.4.3	
TM-04	Functional	Upload image	TESTCASE 4.4.4	
TM-05	Functional	Show result	TESTCASE 4.4.5	

Table4.2: Traceability Matrix

4.3 Testing Environment

Testing environment is made with hardware and software, so that tester may execute what Tests mean. There are few testings' region for testing environment which I used for my project testing.

- Test data
- Web Server
- Database Server
- Frontend running environment
- Back end running environment
- Network
- Browser

4.4 Test Cases

4.4.1 Capture Image

Serial	Uploaded photo	Expected Result	Pass /Fail	Actual Result	Comment
1	mango.jpg	error	Fail	Pass	
2	Leaf.png	error	Fail	Pass	
3	Leaf.jpg	successful	Pass	Pass	

4.4.2 Show result

Serial	Results	Expected result	Actual Result	Comment
1	Early blight	Early blight	Early blight	
2	healthy	healthy	Late blight	
3	Late blight	Late blight	Early blight	
4	Healthy	healthy	Healthy	

4.4.3 Clear image

TestCase:03			**Test Case Name:** clear image		
System:			**Sub-System:**		
Designed by: Sahbaj Sarder			**Designed date:**		
Executed by:			**Executed date:**		
Short Description: user maybe able to clear images					
Pre-condition: • User must have uploaded the photo to clear it					
Serial	**Action**	**Expected Result**	**Pass/ Fail**	**Actual Result**	**Comment**
1	Click on clear button	Clear the result box	Pass	Pass	
Post-Condition: User can clear the previous photo					

4.4.4 Upload Image

TestCase:04	**Test Case Name:** upload image
System:	**Sub-System:**
Designed by: sahbaj Sarder	**Designed date:**
Executed by:	**Executed date:**

Short Description: Uploading photos

Pre-condition:

- User must have photos to upload

Serial	Action	Expected Result	Pass/ Fail	Actual Result	Comment
1	Upload photo	Uploaded photo	Pass	Pass	

Post-Condition: User can predict the condition of the image

4.4.5 Home button

TestCase:04			**Test Case Name:** Home button		
System:			**Sub-System:**		
Designed by: Sahbaj Sarder			**Designed date:**		
Executed by:			**Executed date:**		
Short Description: Home Button					
Pre-condition: • User must have gone to the about section to come back to home					
Serial	**Action**	**Expected Result**	**Pass/ Fail**	**Actual Result**	**Comment**
1	Home	Home view	Pass	Pass	
Post-Condition: User can view the home					

4.4.6 about

TestCase:05			**Test Case Name: about**		
System:			**Sub-System:**		
Designed by: Sahbaj Sarder			**Designed date:**		
Executed by:			**Executed date:**		
Short Description: user can see the about section					
Pre-condition: • user must be at home					
Serial	**Action**	**Expected Result**	**Pass/ Fail**	**Actual Result**	**Comment**
1	about	About section	Pass	Pass	
Post-Condition: about section					

CHAPTER 5
USER MANUAL

Upload photo

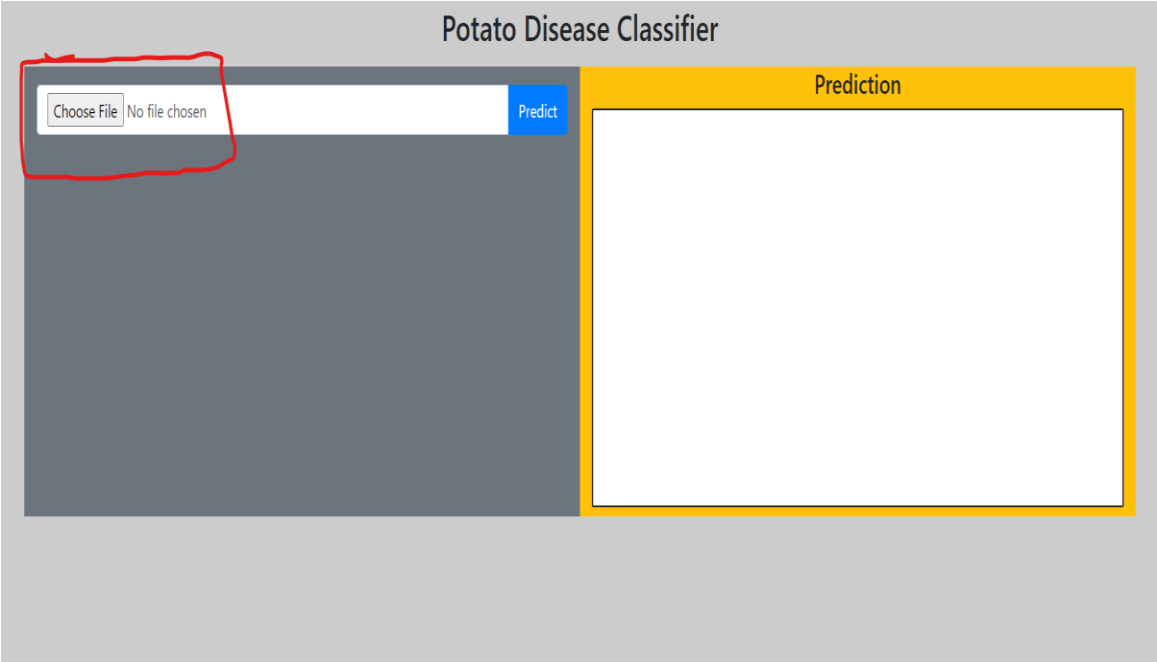


Figure5.1: upload photo

predict

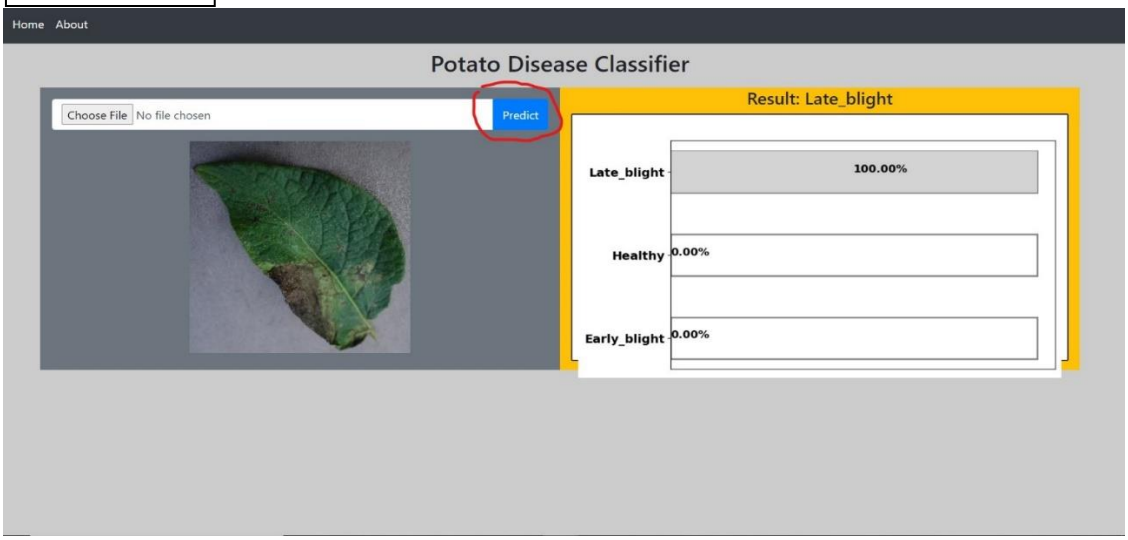


Figure5.2: predict

home

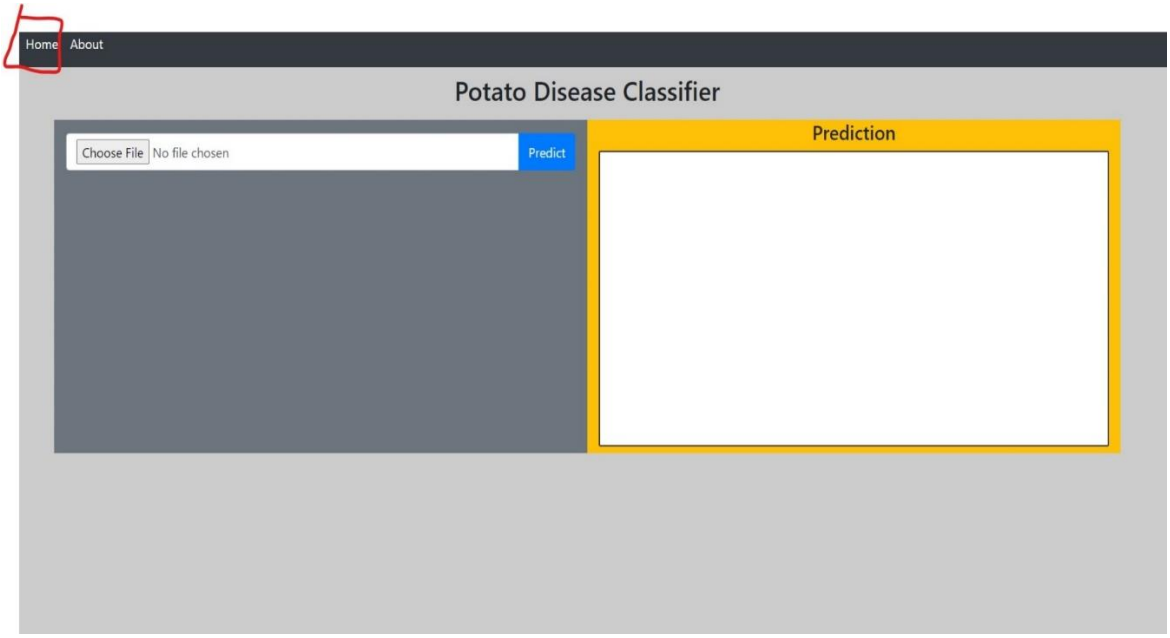


Figure5.3: home

about

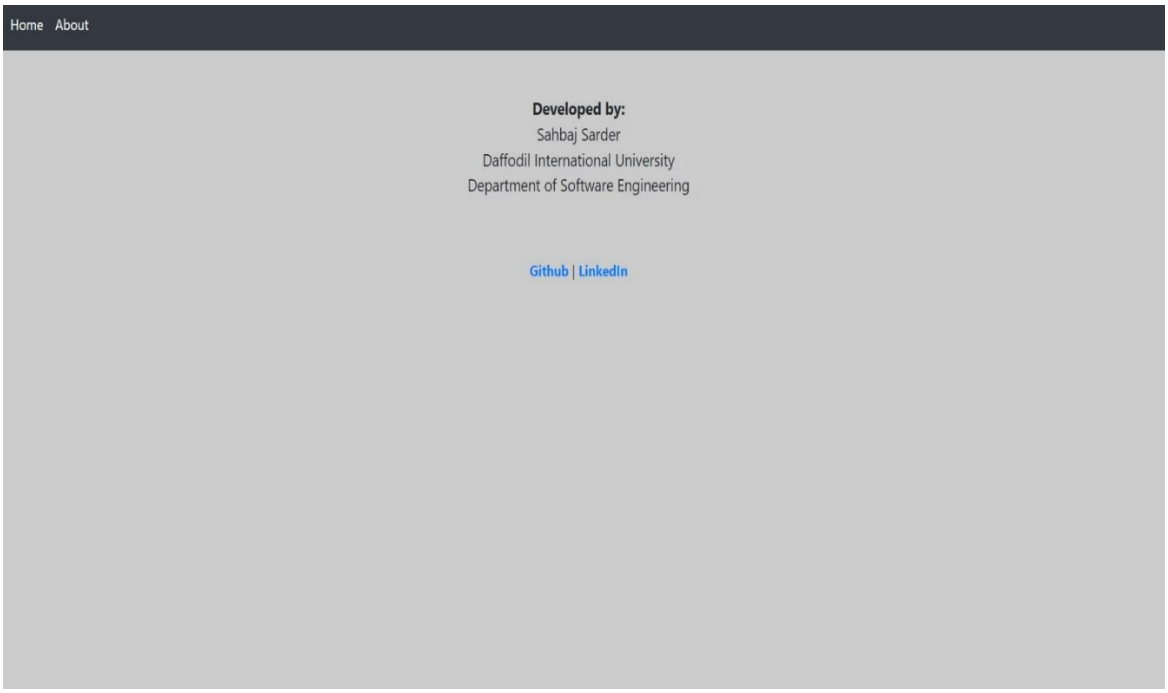


Figure5.4: about

Show result

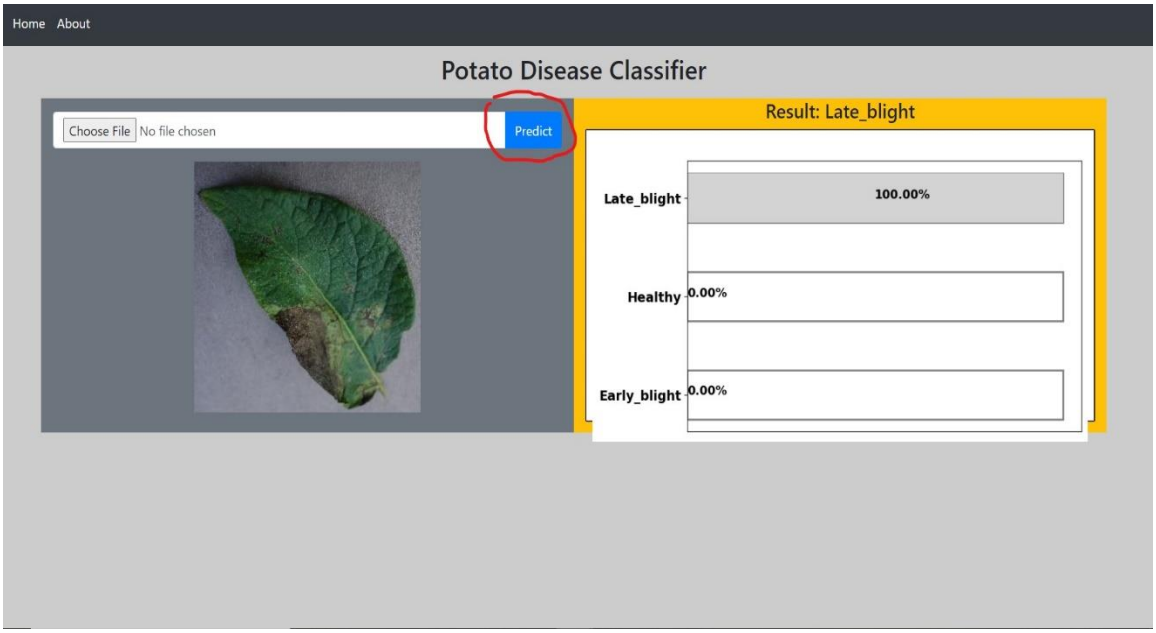


Figure5.5show result

CHAPTER 6

CONCLUSON

6.1 Project Summery

Potato leaf early blight and late blight disease detection system is a project which I have been dreamed of building from the very beginning of my B.SC studentship. My family is surrounded by farmers and the main earning source of my family has been agriculture. I have seen my family struggling and facing losses in potato field for these two disease. I wanted to help the farmers with my knowledge.

This project will help the farmers to detect the diseases with CNN. Farmers can take photos of their potato plant's leaf and detect if their crop is affected with early blight or late blight disease. I have used flask, tensorflow, matplotlib, numpy for this project. This project will use deep learning to detect the condition of the leaf.

6.2 Limitation

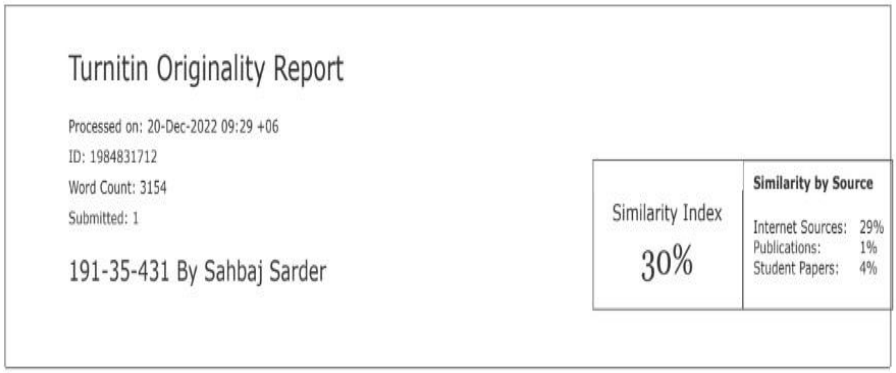
- Not fully responsive
- It is a web-based system only
- This system is not a certified system
- Not fully secure

6.3 Obstacles and Achievement

One must overcome several difficulties before making progress along the grand path. I actually believe I accomplished it with assistance from my buddies, supervisor and co-supervisor, as well as extensive online research and Google for answers. I feel competent enough to complete this assignment on my own..

6.4 Future Scope

- Mobile application can be developed
- A certified system can be developed.



Turnitin Originality Report

Processed on: 20-Dec-2022 09:29 +06
ID: 1984831712
Word Count: 3154
Submitted: 1

191-35-431 By Sahbaj Sarder

Similarity Index	Similarity by Source
30%	Internet Sources: 29% Publications: 1% Student Papers: 4%

15% match (Internet from 26-Oct-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/8501/181-35-2298%20%2820%25%29.pdf?isAllowed=y&sequence=1

3% match (Internet from 26-Oct-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/8363/181-35-2350%20%2828%25%29.pdf?isAllowed=y&sequence=1

3% match (Internet from 21-Nov-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/8400/181-35-2309.pdf?isAllowed=y&sequence=1

1% match (Internet from 21-Nov-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/8406/181-35-2374.pdf?isAllowed=y&sequence=1

1% match (Internet from 21-Nov-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/8242/181-35-2452%20%2830%25%29%20clearance.pdf?isAllowed=y&sequence=1

1% match (Internet from 20-Nov-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/4052/P14413%20%289_%29.pdf?isAllowed=y&sequence=1

1% match (Internet from 26-Oct-2022)
http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/8265/173-35-263%20%2813%25%29.pdf?isAllowed=y&sequence=1

Student Dashboard