Final Mini Project - II AI&DS_organized.docx

WORD COUNT 2881
TIME SUBMITTED 08-JUN-2022 09:53PM
PAPER ID 87024843

ABSTRACT

In the Indian economy is heavily reliant on agricultural productivity. Another disadvantage is farmers' inability to choose the optimal crop for their land using traditional methods. Pest infestations also lower the quality and productivity of crop plants.

On a large farm, manually detecting pests in leaves takes a long time and effort. A deep learning technique is used in the proposed system to detect pests on leaves. A crop suggestion model is constructed using a machine learning algorithm that takes into account all aspects such as seeding season, terrain, and geographical region. The suggested model also uses Rest Api to predict weather and alert the famers.

CHAPTER 1

INTRODUCTION

Farmers are unaware of the change in climate due to global warming and could not afford modern techniques for identification of pest attacks in crops. Farmers require prompt diagnosis of insect infestations and crop suggestions to increase agricultural yield. If proper approach is not taken of the pest attacks, it may cause diverse effects in the quality and productivity of the crops. Images of contaminated leaf portions are captured and preprocessed in preparation for training the neural network model to forecast the illness. Convolutional neural networks are capable of handling massive volumes of data.

1.1 Overview of the Project

This project describes a way for the users to identify the pests in their crop using leaf disease detection. It also alert the farmers about the upcoming weather predictions and they could plan their work accordingly. The project also suggest the types of crops that would be suitable for the farmers according to the the geographical location of the user and also with help of the users soil conditions. On total the application assists the farmers in their daily life and reduce their burdens to a level.

CHAPTER 3

PROJECT DESCRIPTION

3.1 PROBLEM DEFINITION

A framework for Crop Suggestion aims to predict the suitable crop based on the favorable conditions as most of the farmers face huge yield loss due to improper seedling of crops and poor managing steps. Crop suggestions are made by regularly evaluating the geologic location's physical, environmental, and financial aspects. The correct diagnosis of crop diseases is amongst the most important parts of smart agriculture. Crops quality and quantity may decrease if predators are not managed effectively. Most landowners are uninformed of all crop pests and have no idea how to approach professionals for advice. Even agronomists frequently fail to correctly diagnose crop disease. Because disease symptoms often occur on different portions of the crops, photographs of those affected parts could be used for identification. As a result, automatic disease diagnosis from photos is beneficial in assisting experts since it is more precise and requires less work and attention and all of these features are mobile app based designed.

3.2 OBJECTIVES

The main steps performed through this system are sketched as follows:

1. Machine learning, a subfield of AI technology that allows software to learn automatically without being explicitly programmed by a human, is used to build the crop recommendation system. The program's performance will thereafter be raised even without human involvement.

2. Convolutional Neural Networks are used to identify pests. It excels in processing large volumes of information in a short time. The pre-trained CNN model's fine tuning is used for analysing image information of pest-infected leaf as well as other parts.

3.3 BLOCK DIAGRAM

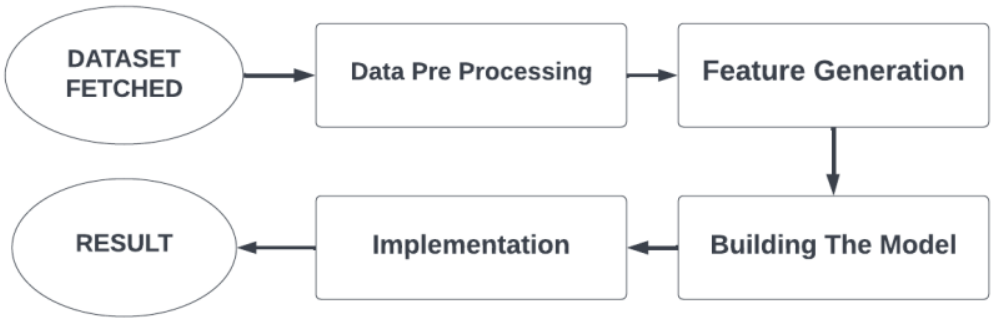

Fig 3.3.1 Flow Diagram for module

3.4 WORKING PRINCIPAL

There are four modules in the project. They are

- Data Pre-Processing
- Feature Generation
- Building the model
- Implementation

3.4.1 DATA PREPROCESSING

Data preprocessing is the way of preparing raw information for use with a learning algorithm. It is the first and most important stage in developing a machine learning model.

Data preprocessing in crop suggestion module:

Suggestion involves replacing the null, 0 values for yield by -1 to get a precise overall prediction. It includes cleaning data by eliminating anomalies or erroneous data and dealing with missing values. Database values are in character unit format. To transfer input, it must be transformed to whole numbers. In addition to minimizing the quantity of data into the crop rotation model, the nutrients necessary in the soil are categorized. If the soil's nutrient composition is below what plants need, the damaged plant will be removed; in this instance, the time taken to train the model is substantially decreased.

Data preprocessing in pest detection:

It involves changing the input format of the given image to size of 256x256 pixels. In the training image database, several random modifications are used, including randomized orientation, zooming, trimming, and browsing. Augmentation is used to prevent overpopulation and to improve the standard model.

Data preprocessing in weather module:

By processing post data, simplified speculation models can be developed using machine-leading APIs, such as the Open Weather API. Users can submit training data on indoor, outdoor and energy requirements, to create machine learning models to predict indoor temperatures and energy demand based on newly distributed outdoor temperatures.

3.4.2 FEATURE GENERATION

Feature extraction is a dimensionality reduction procedure that reduces an initial collection of original data to more controllable groupings for process.

The enormous number of parameters in these massive datasets demands a significant amount of computational resources to process.

3.4.2.1 Transfer Linear Technique

Transfer machine - learning approach in which the information obtained from training the model of one difficulty is used to develop a model of another difficulty. This approach is used to identify pests in leaf.

Deep learning extracts characteristics from the first few layers, something that we can then employ in our model by removing parts of the final levels and substituting our layers. The idea of transfer learning was cleverly manipulated. The Soft - max function is used to add dense layers at the conclusion of the model after precise CNN model tuning. Dataset was used to generate the model, which was then loaded with pre-trained weights.

3.4.3 BUILDING MODEL

Algorithms are the driving force behind machine learning, as they are the ones that turn a set of data into a model.The suggestion and pest detection involving various machine learning algorithms to obtain the best result and accuracy .

The following are the different algorithms which are used

- Random Forest Algorithm
- Decision Tree
- Logistic Regression
- Keras

3.4.3.1 RANDOM FOREST ALGORITHM

The random forest algorithm are concerned with controlled classification. The two most important features are the trees and the forest. The larger the trees, the better the accuracy.

$$ni_j = w_j C_j - w_{left(j)} C_{left(j)} - w_{right(j)} C_{right(j)}$$

W sub(j) – No. of weighted training sample in y axis.C sub(j) – The loss value of training sample of y axis.

ni_j – the obtained value by using left child node and right child node int tree

3.4.3.2 DECISION TREE

The decision tree involves the splitting up of dataset into different nodes and leaves with the specified features. The different nodes end up in leaves which gives us the result

Cost functions used for classification:

$$G = \text{sum}(pk * (1 - pk))$$

The Gini score(G) gives us good a split in the training data.

pk is proportion of same class inputs (x-axis (54 col)).

3.4.3.3 LOGISTIC REGRESSION

The method of modelling the probability of a distinct result given an input variable is known as logistic regression. The most frequent logistic regression models have a binary outcome, which might be true or false, yes or no, and so forth.

$$P = \frac{1}{1 + e^{-(a+bX)}}$$

P is the probability

e is the base of the natural logarithm

a and b are the parameters of the model

3.4.3.4 KERAS

Keras is a cutting-edge TensorFlow API with a user-friendly, high-productivity interface for addressing machine learning problems, with an emphasis on deep contemporary learning. Provides crucial information and building blocks for developing and delivering high-repetition-rate machine learning solutions.

3.4.4 DEPLOYMENT

To make the trained model available to farmers, we would need an simple mobile application that has an simple UI that farmers could use efficiently

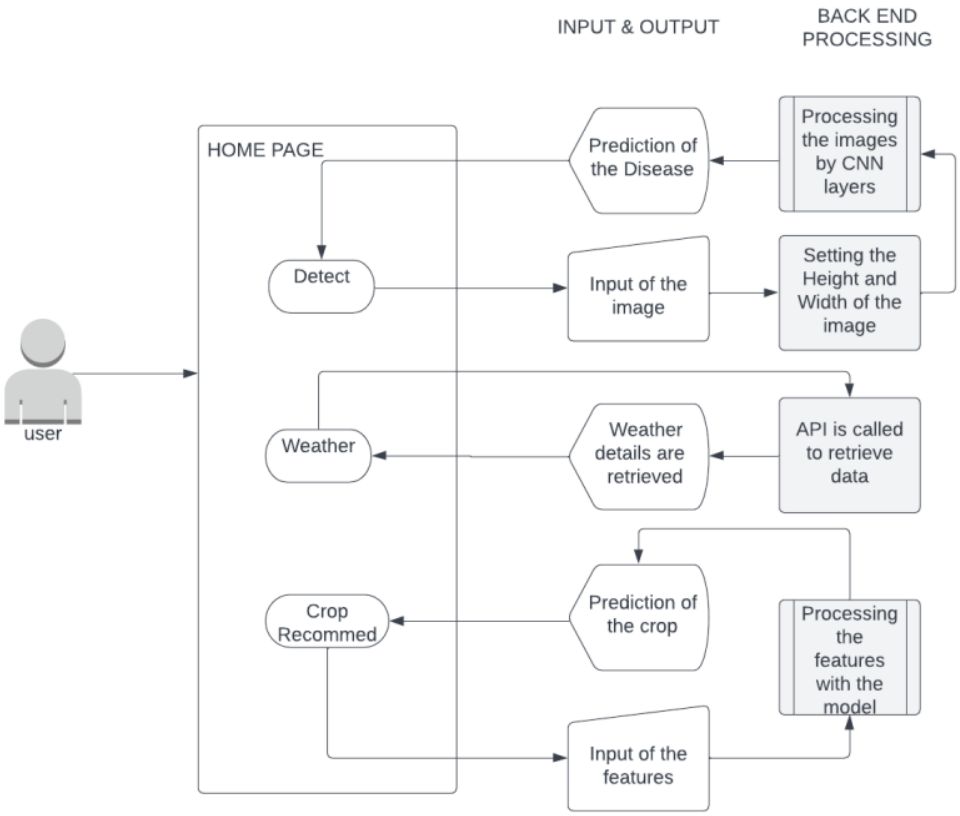

Fig. 3.4.4.1 Deployment flow diagram

3.4.4.1 HOME PAGE :

- In this module, Home Page for the application was implemented.
- There are three cards in the card view, each direct to a specified module
    1. Leaf Disease Detection
    2. Weather
    3. Crop Recommendation
- The news scroll view contains news which is customized for the individual user (to be implemented in next phase).

3.4.4.2 LEAF DISEASE DETECTION :

- The pest detection module is implemented with the pest detection model we created using the Machine learning algorithms.
- The model is converted into a Tflite model hence it could be integrated with the application's user interface
- We could upload the picture of the lead either from the gallery or take a picture using the camera
- The Disease would be detected after uploading or taking the image of the leaf
- The User Interface would display the
    1. Disease Detected
    2. Confidence

3.4.4.3 WEATHER :

- The application uses the REST API' from openweathermap.org to retrieve the weather forecast details.
- An 'app_id' is created and it must be passed down with each API request to retrieve the current forecast of the user's current location.
- The application would request for the location access from the user and get the location details like latitude and longitude.
- The user interface would display the
    1. Current Weather of the location
    2. Present day's Hourly weather cast
    3. 7 days forecast

3.4.4.4 CROP RECOMMENDATION :

- The Crop Recommendation model is implemented with the crop recommendation model we created using the machine learning algorithms.
- The model is converted into a Tflite model hence it could be integrated with the application's user interface
- The User Interface would ask the following details to suggest the crop
    1. Nitrogen
    2. Potassium
    3. Phosphorous
    4. Temperature
    5. Humidity

- The user interface would display the recommended crop for the user with the given details

CHAPTER 4

RESULTS & DISCUSSON

The Application assists the user to detect the disease in their crop plants by taking or uploading a picture of the infected plant's leaf. It also gives the weather details for the user's present location and a weeks forecast is also given. It also suggest the crops, that would be suitable for the farmers land by analyzing the nitrogen, potassium, phosphorous, humidity, temperature values.

ALGORITHMS	ACCURACY
DECISION TREE	98.40%
LOGISTIC REGRESSION	94.54%
RANDOM FOREST	99.54%
CNN	89.25%

Fig 4.1 : Accuracy rates of algorithms

4.2 ACCURACY OBTAINED

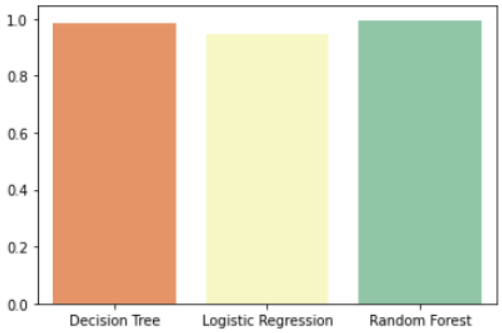

Fig. 4.2.1 Accuracy Graph for algorithm

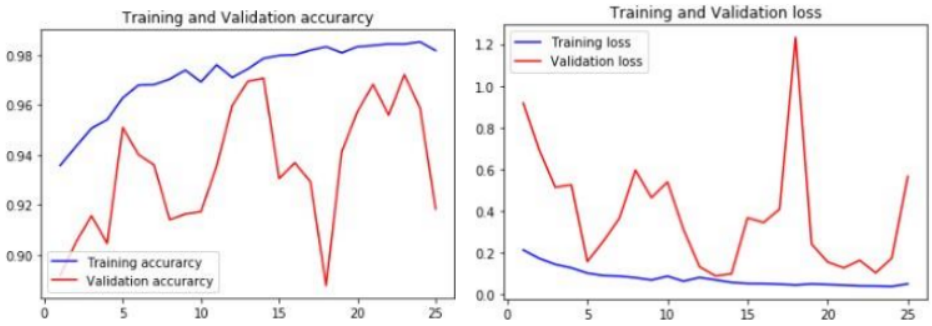

Fig 4.2.2 Accuracy plot for keras model

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION:

The rapid development in technology is a persistent process. This application provides various advantages to the farmers which integrated to work in increasing the global food productivity. It involves various machine learning and neural network algorithms that in co-ordination work for detection of pests affecting the plants and clustering classifier in turn helps to match the various factors for best outcome of the given condition. The weather recording has a huge impact for the good outcome from the crops and it is implemented via API calls from the open weather site.

5.2 FUTURE SCOPE:

Based upon the research, recommendations proposed for the further study are as follows:

1. The leaf disease detection model is trained for about 20 plants in the current project and many more plants dataset could be added and the model could be tuned well.
2. Can modify the weather module to make notification or alert on climate change.
3. The crop recommendation datasets could also be improved by adding aditional datasets of plants.

APPENDICES

APPENDIX 1

SOFTWARE DESCRIPTION

PYTHON 3.7

i. Python is a high level programming language which is interpreter based. Python is dynamically typed and garbage-collected.It supports structured procedural, object-oriented, and functional programming paradigms.

FLUTTER FRAMEWORK

i. The Flutter framework consists of both a software development kit (SDK) and their widget-based UI library. This library consists of various reusable UI elements, such as sliders, buttons, and text inputs

ii. Developers building mobile applications with the Flutter framework will do so using a programming language called Dart. With a syntax like JavaScript, Dart is a typed object programming language that focuses on front-end development.

- **DART**

i. Dart is a client-optimized language for developing fast apps on any platform. Its goal is to offer the most productive programming language for multi-platform development, paired with a flexible execution runtime platform for app frameworks.

ii. Dart provides the language and runtimes that power Flutter apps, but Dart also supports many core developer tasks like formatting, analyzing, and testing code.

iii. Dart offers sound null safety, meaning that values can't be null unless you say they can be. With sound null safety, Dart can protect you from null exceptions at runtime through static code analysis. Unlike many other null-safe languages, when Dart determines that a variable is non-nullable, that variable is always non-nullable. If you inspect your running code in the debugger, you'll see that non-nullability is retained at runtime

VISUAL STUDIO FRAMEWORK

i. Microsoft's Visual Studio Code is a freely downloadable software tool. VS Code is compatible with Windows, Linux, and macOS. VS Code supports a wide range of scripting languages, including Java, C++, and Python, as well as CSS, Go, and Docker files.

ii. You may add on and even create new extensions for VS Code, such as code linters, debugging tools, and cloud and website development capabilities. In comparison to other text editors, the VS Code UI allows for a great deal of interactivity.

APPENDIX 2

SOURCE CODE

HOME PAGE :

```
import 'package:farmeasy/pages/home_page/home_page.dart';
import 'package:flutter/material.dart';
import 'package:flutter_svg/svg.dart';
import 'package:farmeasy/utilities/colors.dart';
import 'package:farmeasy/utilities/constant.dart';
import 'package:farmeasy/weatherprediction/weather.dart';
import 'package:farmeasy/plantsuggestion/suggest.dart';
import 'package:farmeasy/utilities/style.dart';

class Home extends StatefulWidget {
  @override
  _HomeState createState() => _HomeState();
}

class _HomeState extends State<Home> {
  int selectedCard = 0;
  @override
  void initState() {
    super.initState();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
```

```
backgroundColor: AppColors.white,
appBar: AppBar(
  backgroundColor: Colors.transparent,
  centerTitle: false,
  elevation: 0,
  actions: [
  IconButton(
     onPressed: null,
     icon: SvgPicture.asset('assets/menu.svg'),
    ),
    SizedBox(
     width: 20,
    )
  ],
),
body: SingleChildScrollView(
  child: Column(
    children: [
     Padding(
       padding: const EdgeInsets.only(top: 25),
       child: Column(
         crossAxisAlignment: CrossAxisAlignment.start,
         children: [
           Padding(
             padding: const EdgeInsets.only(left: 20),
             child: PrimaryText(
               text: 'Farm',
               size: 22,
             ),
```

```
),
Padding(
  padding: const EdgeInsets.only(left: 20),
  child: PrimaryText(
    text: 'Easy',
    height: 1.1,
    size: 42,
    fontWeight: FontWeight.w600,
  ),
),
SizedBox(height: 10),
Row(
  crossAxisAlignment: CrossAxisAlignment.center,
  children: [
    SizedBox(width: 20),
    Icon(
      Icons.search,
      color: AppColors.secondary,
      size: 25,
    ),
    SizedBox(width: 10),
    Expanded(
        child: TextField(
      decoration: InputDecoration(
        enabledBorder: UnderlineInputBorder(
            borderSide: BorderSide(
                width: 2, color: AppColors.lighterGray)),
        hintText: 'Search..',
        hintStyle: TextStyle(
```

```
            color: AppColors.lightGray,
            fontSize: 20,
            fontWeight: FontWeight.w500),
      ),
    )),
    SizedBox(width: 20),
  ],
),
SizedBox(height: 25),
Padding(
  padding: const EdgeInsets.only(left: 20),
  child: PrimaryText(
      text: 'Categories',
      fontWeight: FontWeight.w700,
      size: 22),
),
SizedBox(
  height: 240,
  child: ListView.builder(
    scrollDirection: Axis.horizontal,
    itemCount: cardview.length,
    itemBuilder: (context, index) => Padding(
      padding: EdgeInsets.only(left: index == 0 ? 25 : 0),
      child:  CategoryCard(cardview[index]['imagePath'],
          cardview[index]['name'], index),
    ),
  ),
),
Padding(
```

```
            padding: const EdgeInsets.only(left: 20, top: 10),
            child: PrimaryText(
                text: 'News', fontWeight: FontWeight.w700, size: 22),
          ),
          Container(
            width: MediaQuery.of(context).size.width,
            height: 90,
            child: Row(
              children: [
              Padding(
                  padding: EdgeInsets.symmetric(horizontal: 10),
                  child: SizedBox(
                    height: 110,
                    width: 110,
                    child: Image.asset("assets/news1.jpeg"),
                  )),
              Expanded(
                  child: Padding(
                padding: EdgeInsets.symmetric(horizontal: 10),
                child: Text(
                    "India wheat farmers count cost of 40C heat that evokes deserts
of Rajasthan",
                    style: TextStyle(
                        fontWeight: FontWeight.bold, fontSize: 15)),
              ))
            ],
          ),
          Container(
```

```
                width:  MediaQuery.of(context).size.width,
                height: 90,
                child: Row(
                  children: [
                  Padding(
                      padding: EdgeInsets.symmetric(horizontal: 10),
                      child: SizedBox(
                        height: 110,
                        width: 110,
                        child: Image.asset("assets/news2.jpeg"),
                      )),
                    Expanded(
                      child: Padding(
                    padding: EdgeInsets.symmetric(horizontal: 10),
                    child: Text(
                        "Punjab Farmers came prepared for Singhu-type haul,AAP
bored down in just 1 day ",
                        style: TextStyle(
                            fontWeight: FontWeight.bold, fontSize: 15)),
                    ))
                  ],
                ),
              ),
              Container(
                width: MediaQuery.of(context).size.width,
                height: 90,
                child: Row(
                  children: [
                  Padding(
```

```
                padding: EdgeInsets.symmetric(horizontal: 10),
                child: SizedBox(
                  height: 110,
                  width: 110,
                  child: Image.asset("assets/news3.jpeg"),
                )),
              Expanded(
                child: Padding(
                padding: EdgeInsets.symmetric(horizontal: 10),
                child: Text(
                  "TN introduces mobile processing units in tomato hub Dharmapuri",
                  style: TextStyle(
                      fontWeight: FontWeight.bold, fontSize: 15)),
              ))
            ],
          ),
        ),
        Container(
          width: MediaQuery.of(context).size.width,
          height: 90,
          child: Row(
            children: [
            Padding(
                padding: EdgeInsets.symmetric(horizontal: 10),
                child: SizedBox(
                  height: 110,
                  width: 110,
                  child: Image.asset("assets/news4.jpeg"),
```

```
                    )),
                Expanded(
                  child: Padding(
                  padding: EdgeInsets.symmetric(horizontal: 10),
                  child: Text(
                    "Tamil Nadu to Man All Villages With Drone-Technology in
Three Years, Trains Pilots To Aid Agriculture, Mining and More ",
                    style: TextStyle(
                      fontWeight: FontWeight.bold, fontSize: 15)),
                ))
              ],
            ),
          ),
          Container(
            width: MediaQuery.of(context).size.width,
            height: 90,
            child: Row(
              children: [
              Padding(
                  padding: EdgeInsets.symmetric(horizontal: 10),
                  child: SizedBox(
                    height: 110,
                    width: 110,
                    child: Image.asset("assets/news5.jpeg"),
                  )),
                Expanded(
                  child: Padding(
                  padding: EdgeInsets.symmetric(horizontal: 10),
                  child: Text(
```

```
                    "Free power a pipe dream for poor farmers in Tamil Nadu",
                    style: TextStyle(
                        fontWeight: FontWeight.bold, fontSize: 15),
                  ),
                ))
              ],
            ),
          )
        ],
      ),
    )
  );
}


Widget CategoryCard(String imagePath, String name, int index) {
 return GestureDetector(
   onTap: () => {
    if (imagePath == 'assets/plantdetect.svg')
     {
      Navigator.push(
        context,
        MaterialPageRoute(builder: (context) => HomePage()),
      )
     }
    else if (imagePath == 'assets/sun.svg')
```

```
    {
      Navigator.push(
          context, MaterialPageRoute(builder: (context) => weather()))
    }
  else if (imagePath == 'assets/coke.svg')
    {
      Navigator.push(
          context, MaterialPageRoute(builder: (context) => suggest()))
    },

  setState(
    () => {
      print(index),
      selectedCard = index,
    },
  ),
},
child: Container(
  margin: EdgeInsets.only(right: 20, top: 20, bottom: 20),
  padding: EdgeInsets.symmetric(vertical: 25, horizontal: 20),
  decoration: BoxDecoration(
      borderRadius: BorderRadius.circular(20),
      color:
          selectedCard == index ? AppColors.primary : AppColors.white,
      boxShadow: [
        BoxShadow(
          color: AppColors.lighterGray,
          blurRadius: 15,
        )
```

```
          ]),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.spaceBetween,
          children: [
            SvgPicture.asset(imagePath, width: 40),
            PrimaryText(text: name, fontWeight: FontWeight.w800, size: 16),
            RawMaterialButton(
                onPressed: null,
                fillColor: selectedCard == index
                    ? AppColors.white
                    : AppColors.tertiary,
                shape: CircleBorder(),
                child: Icon(Icons.chevron_right_rounded,
                    size: 20,
                    color: selectedCard == index
                        ? AppColors.black
                        : AppColors.white))
          ],
        ),
      ),
    );
  }
}
```

APPENDIX 3

SCREENSHOTS

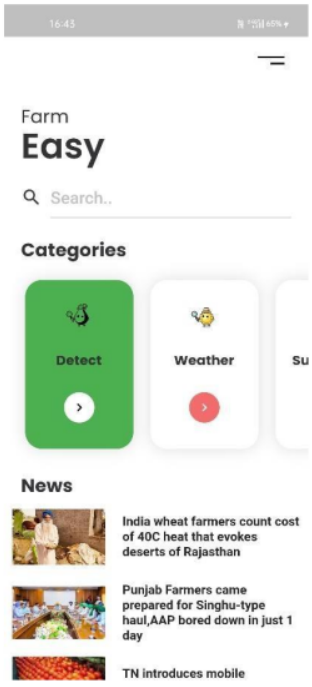

Fig A3.1 Home Page

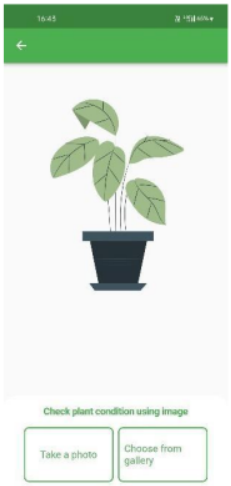

Fig A3.2 Plant Leaf disease detection

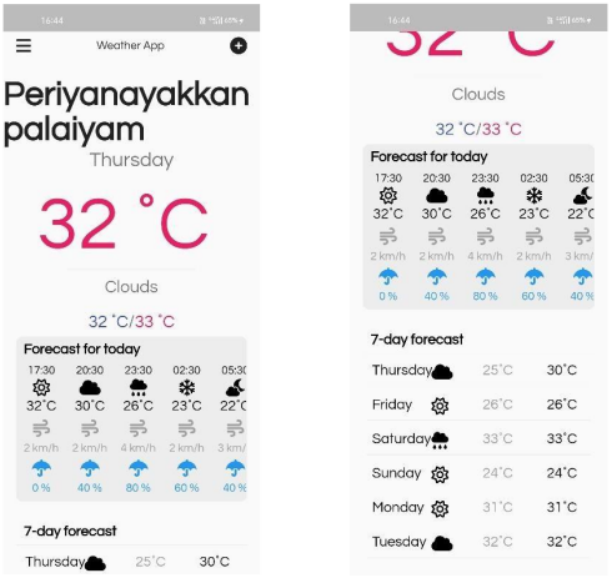

Fig. A3.3 Weather Module

Final Mini Project - II AI&DS_organized.docx

ORIGINALITY REPORT

20%
SIMILARITY INDEX

PRIMARY SOURCES

1	dart.dev Internet	75 words — 2%
2	www.coursehero.com Internet	68 words — 2%
3	uis.brage.unit.no Internet	58 words — 2%
4	Marco L. Napoli. "Beginning Flutter®", Wiley, 2019 Crossref	50 words — 2%
5	fluttertutorial.in Internet	50 words — 2%
6	gitlab.sliit.lk Internet	50 words — 2%
7	www.ijert.org Internet	43 words — 1%
8	git.0x1a8510f2.space Internet	34 words — 1%
9	Kybernetes, Volume 43, Issue 5 (2014-09-16) Publications	21 words — 1%
10	github.com Internet	

		21 words — 1%
11	stackoverflow.com Internet	20 words — 1%
12	www.werindia.com Internet	20 words — 1%
13	tools.ages.pucrs.br Internet	19 words — 1%
14	opensource.com Internet	18 words — 1%
15	Www.indiatoday.in Internet	14 words — < 1%
16	rssrabbit.com Internet	12 words — < 1%
17	timesofindia.indiatimes.com Internet	11 words — < 1%
18	ilmucoding.com Internet	10 words — < 1%
19	rucore.libraries.rutgers.edu Internet	10 words — < 1%
20	docplayer.net Internet	9 words — < 1%

EXCLUDE QUOTES OFF EXCLUDE SOURCES OFF

EXCLUDE BIBLIOGRAPHY OFF

EXCLUDE MATCHES OFF