# AI BASED CROP RECOMMENDATION SYSTEM

## 

## INDEX

**1. INTRODUCTION**

1.1 General Introduction

1.2 Project’s Purpose

1.3 Existing Problem

1.4 Proposed Solution

**2. Flow CHARTS**

3.1 Application's Workflow

3.2 Back-End Flow of Application

3.3 Flow of Forecasting Model

**3. ENVIRONMET DETAILS**

4.1 Hardware Details

4.2 Software Details

**4. DEVELOPMENT**

5.1 Environment Setup

5.2 Data Cleaning & Extraction

5.3 Feature Engineering

5.4 Modelling

5.5 Averaging

**5. RESULTS:**

6.1 Findings

**6. CONCLUSION**

7.1 Advantages & Limitations

7.2 Future Scope

**7. APPENDIX & REFERENCES**

**INTRODUCTION**

India being an agricultural country, its economy predominantly depends on agriculture yield growth and agro-industry products. 58 % of the rural households depend on agriculture as their primary means of livelihood which acts as one of the largest contributors to the Gross Domestic Product (GDP).

We know that there is a decrease in crop production and shortage of food across the country which also has been consequence of bad crop selection and thus, leading to increasing farmer suicides.

**Project's Purpose**

Data mining is the practice of examining and deriving purposeful information from the data. Data mining finds its application in various fields like finance, retail, medicine, agriculture etc. Data mining in agriculture is used for analysing the various biotic and abiotic factors. Agriculture in India plays a predominant role in economy and employment. The common problem existing among the Indian farmers are they don't choose the right crop based on their soil requirements. Due to this they face a serious setback in productivity. This problem of the farmers has been addressed through precision agriculture. Precision agriculture is a modern farming technique that uses research data of soil characteristics, soil types, crop yield data collection and suggests the farmers the right crop based on their site-specific parameters. This reduces the wrong choice on a crop and increase in productivity.

**Existing Problem**

Different crops need different type of soils, different types and amounts of nutrients, and different types and amounts of water. The amount of water required by the plant is also dependent on the growing season and the climate where it is grown. Hence, there are lot of factors that should be taken into account while preparing for cultivation on the farm.

Generally, farmers do not have much information regarding which crop to select based on various parameters and mostly they rely on either their gut or sometimes on knowledge gained from some other source which at times may not be reliable.

Hence, as a result of bad crop selection farmers face many issues and most problematic is the financial one. So, it is good to access the parameters before starting with the cultivation part and for that there is a need of an application that can help the farmers in determining that what will be the best crops for their land.

**Proposed Solution**

Thus, this project aims to use Artificial Intelligence & Machine Learning techniques Models to recommend/suggests best crops to grow on the given land based on various parameters.

The Flutter application is deployed on the Oracle cloud, fetches the information from user in real-time when the user logs in. The developed application recommends /suggests the best crops to cultivate on the given land, weather and seasonal parameters by the farmers. The simple and convenient UI provides user the flexibility to switch across various pages of application to access the features.

Because the app updates on regular intervals, automatically, it is termed as "Semi-Driverless". In other words, when the date changes at midnight, the previous date and weather data is removed after creating the backup. New Data is fetched from the weather API, Newly Generated reports are sent to the server without any human intervention.

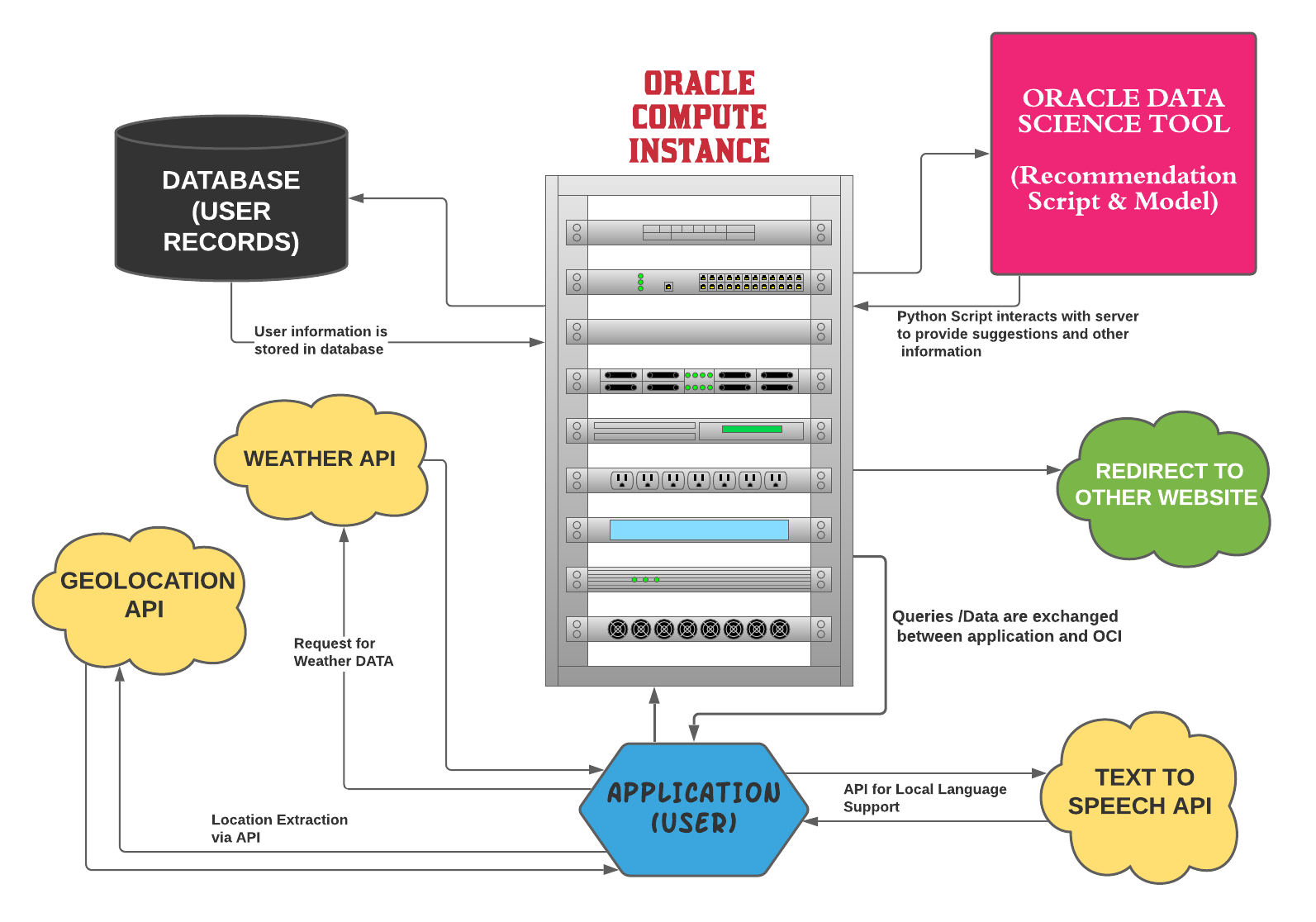
**FLOW CHARTS**

**Application Workflow**

The application workflow provides the basic idea about how the user will be able access various features of the application.

* Before accessing any feature, the user needs to login into the application. If he/she does not possess any account then they are welcomed to create one.
* After the login, the user can make various choices, either they can check the recommendations, or check various sites for information or if they wish to provide feedback based on the result for their own farm then they can fill the form with correct details.
* Inside every choice, the user in turn gets many choices to visualize and observe the recommendations. The details are kept as simple as possible so that it can be easily understood by people from non-technical background as well.

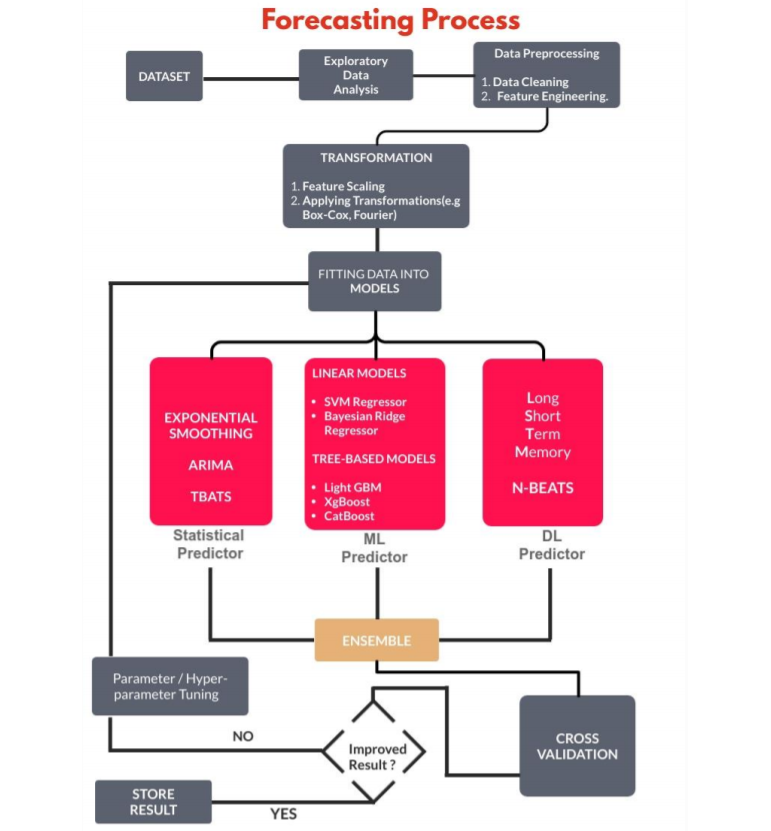
**Back-End Flow of the Application**



* Above figure shows how data travels between resources to obtain the final recommendation for the farmers.
* Once the user opens the application, the geolocation API fetches the location and sends it to script via OCI. The location then is used to fetch other weather and seasonal parameters.
* The input parameters from user are sent to the python script where the recommendations are computed and are then sent back to user.

**Flow of Forecasting Model (for weather parameters)**

The flow demonstrates how (weathers) forecasts are generated from the raw data fetched from the API/Dataset.



**ENVIRONMENT DETAILS**

**Hardware Details:**

1. The Python Script was simulated on a 64-bit windows 10 OS machine with 16 GB RAM having Intel (R) 8th gen i-7, 8685U processor (1.8 Ghz base, 4.8 Ghz turbo boost), 8 cores.
2. The Android Application Script was created on a 64-bit windows 10 OS machine with 8GB RAM having 7th gen i-5, 7200U processor (2.5 Ghz) and 8 cores.
3. The Computational Server for prefinal deployment with recommendation models is an Ubuntu 18.04 machine with 32GB RAM having Intel (R) Xeon(R) E5-2676 v3 processor (2.40 Ghz) and 8 cores.

**Software Details:**

The Python Script was simulated on PyCharm and Google Collaboratory.

**Language:** Python 3.6

**ML Libraries:** CatBoost and Sklearn.

**Data Visualisation Libraries:** Seaborn,Matplotlib.

The android script was simulated on andriod studio**.**

**Flask:** Python framework to create Restful API to deploy the predicting model on server.

**Flutter:** Android Framework by Google for integrating the hybrid app with real-time monitoring of data.

**DEVELOPMENT**

**Account/ Environment Setup:**

We created accounts for the team that is required/will be required during the project for accessing various resources. E.g., OCI free trial accounts were provided by ORACLE for accessing Oracle cloud and other facilities, Android Studio account for creating an android application.

**Data Extraction & Cleaning:**

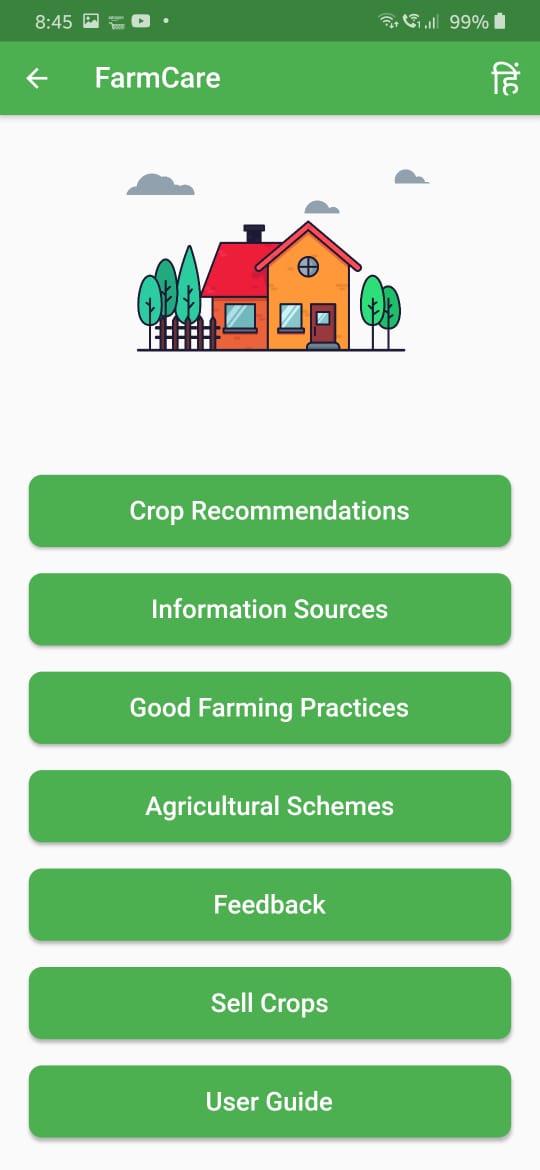
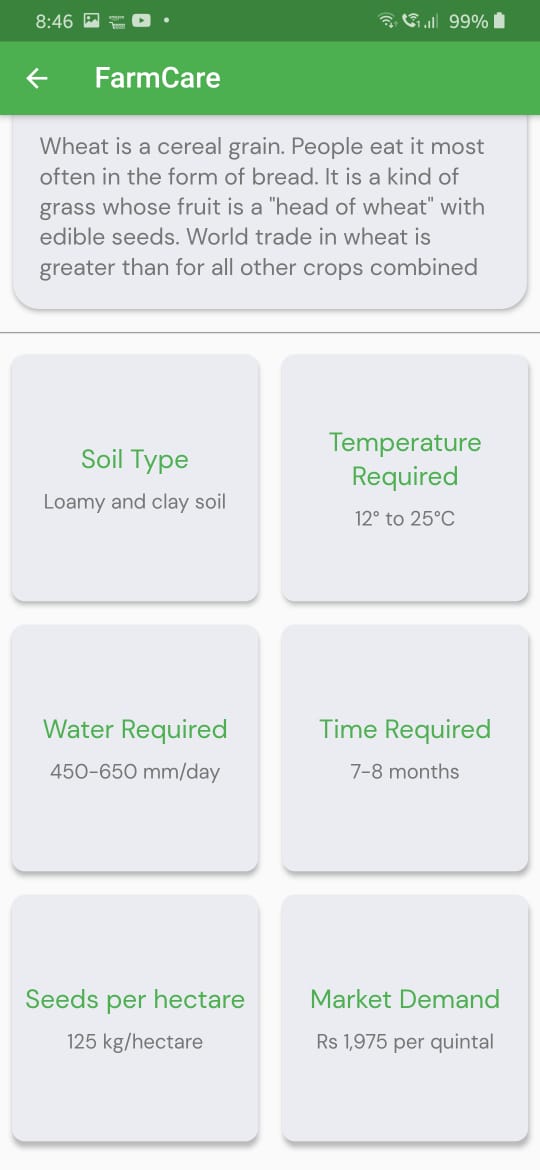
* We first extracted the data from various online resources like government websites, educational institutions, research labs etc. The data thus obtained is used as the training data for the recommendation models.
* The data obtained from API is in JSON format with a dictionary-like structure, so to make data easy to handle and process, we convert it into a data frame using python's PANDAS library. The ***Time*** feature of the data is used as an index. The data frame is then sorted as per the index values.
* We then check if the data contains any null value or not. We observed that, many features have null values. We replace the null values with the mean value of the feature.

**FEATURE ENGINEERING**

In this step we manually generated features that we thought could help the algorithms to understand the patterns in a more effective manner. With the use of relevant features, the complexity of the algorithms reduces. Even if we somehow misjudged the best fit algorithm that is not ideal for the situation, the results would still be accurate.

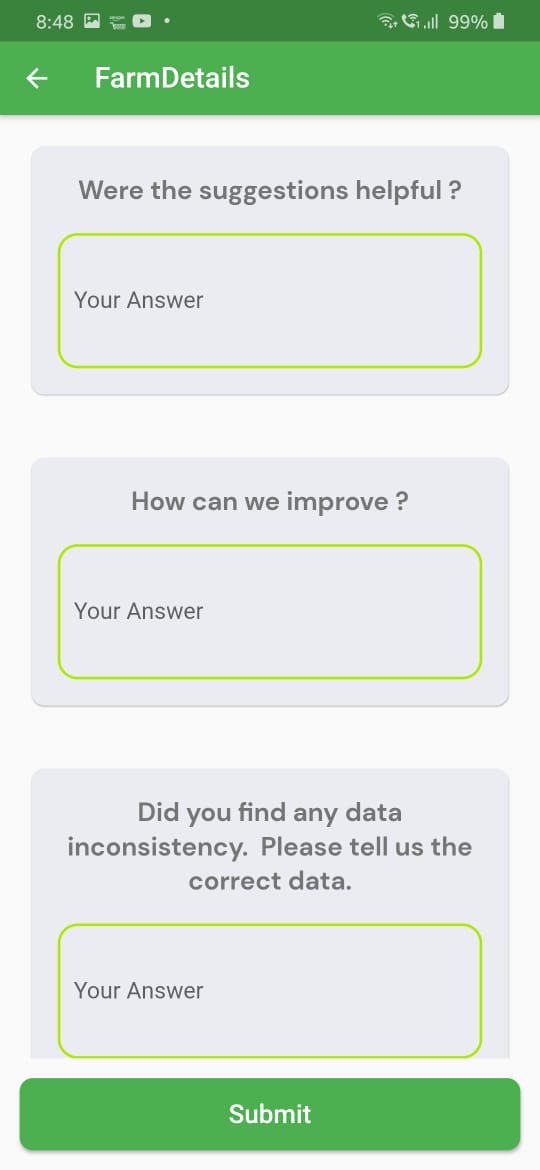
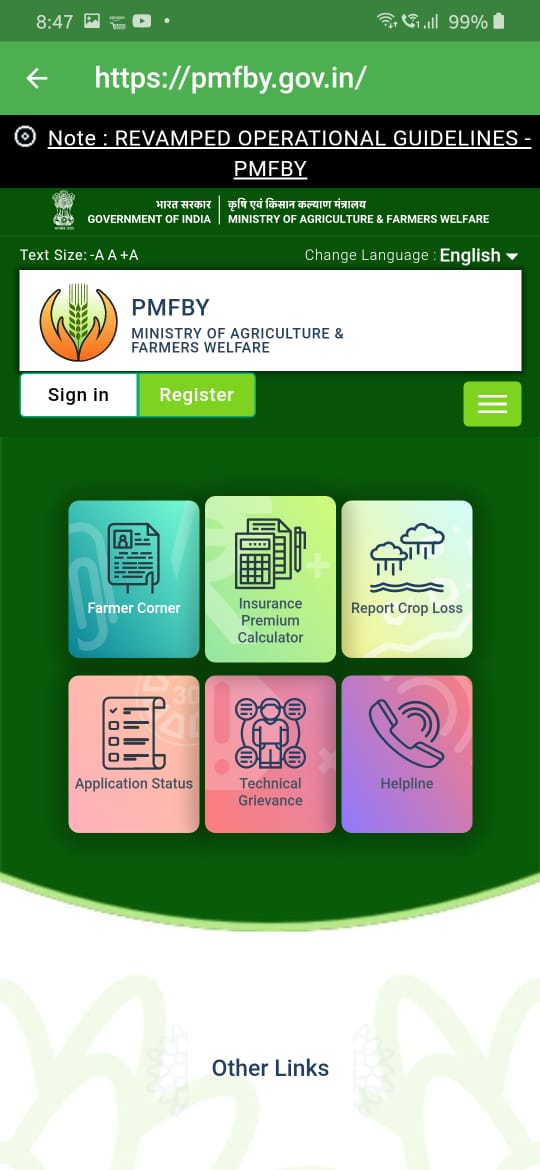
**RESULT**

After training the data on all the models, the result from all the models is combined and is stored in file. The final file is then sent to the application via Oracle Cloud. The app then displays result on the UI.

****

****

**INFORMATION FOR FARMERS**

****

**MORE FEATURES**

**CONCLUSION**

The AI based crop recommendation system application is successfully deployed using Oracle Cloud Instance. The application provides recommendations based on the input provided by the user with the aim to maximize the profit for farmers.

**Advantages**

1. The output of the application is very easy to interpret.
2. The application will not incur any kind of data loss in case of any kind of failure be it hardware or software relate. Because after every computation i.e. whenever new file are generated, the data is safely backed up.
3. The app is "semi-driverless" i.e. it operates on its own and whenever the date changes, new data is fetched, new forecasts are generated, new reports and results are generated, all necessary files are backed up and the final output is sent to the application.

**Limitations**

1. Due to the complex algorithms and methodologies used in the development the application, the code may be hard to debug.
2. Due to real-time fetching some functionality of the application may not perform as desired under slow/weak internet connections.
3. The application depends on various live data sources for training the models so if in any case the API from which the data is being fetched goes down then the application might not be able to produce accurate forecasts.

**FUTURE SCOPE**

1. The e-commerce store where farmers can sell their crops at their own price (in accordance to MSP) is under development and can help in empowerment of people related to the agricultural field.
2. Text -to-Speech for various local language.
3. Addition of powerful models (with fine tuning) that can map non-linear data well can make the recommendations more accurate.
4. Due to computational and other resource related limitations some good performing models were dropped from the final ensemble. Thus, in large computational environment the existing approach can give better results.
5. More detailed analysis of what all crops are suitable for a given land can be done if the other datasets are taken into account.

**BIBLIOGRAPHY**

**Names**: Adhyansh Bhardwaj [1], Anant Jakhmola [2], Kartikay Sawhney [2], Ritik Aggarwal [2]

**College Name**: Maharaja Agrasen Institute of Technology [1], Graphic Era University [2]

**Work Title:** AI Based Crop Recommendation System.

# 

# APPENDIX

1. **Weather BIT API:** <https://www.weatherbit.io/api>
2. **ORACLE CLOUD:** **https://www.oracle.com/in/cloud/**
3. **NIC:** **https://www.nic.ina/**
4. **IEEE:** https://site.ieee.org/indiacouncil/