# Chapter 01 – Implementation

## Chapter Overview

In this chapter the technological and the functional aspect of the system is defined. And the functionality of our system “Market Value Prediction System” is explained in detail and it’s explained in detail for the better understanding. A clear description of the technologies used and how these technologies are used in the web application is included.

Although we planned to develop a mobile application with the web application finally we decided to develop only the web application considering the users using our system. As this system is mostly used by the farmers and considering their financial status and educational level and the complexity in using a mobile application also we came into a conclusion of developing the web application with our fullest effort of the group members and providing the best output for the users of the system and providing them a very simple and a user-friendly system which can be used with a minimum amount of data and with a minimum expense when using our application. So we were able to implement a web based application having a higher accuracy level in the prediction and also with a simple and an easy interface which allows the farmers to use it without any difficulty and which allows the non-educated to use our system with ease without any difficulty. So we implemented it in the best and simplest way for it to be hundred percent user-friendly for the users.

## Overview of the Prototype

* The front end consists of mainly three sections. One for the data science component and one for the farmers section and the other for the sections of the buyer. Farmer will get access to the data table updated by the buyer side which is consisting of attributes Seller, Vegetable, Maximum Accepted stock and Available Stock, Price per (1KG). Where the farmer will be able to see the maximum stock and by consideration of the available stock the farmer will be able to request to supply goods.
* Buyer can accept the requests sent by the farmers who are the sellers and after accepting a notification will be sent to the farmers to notify them to supply the vegetables as requested.
* Front end is developed using React JS with the integration of AdminITE and material UI to get a better output for a simple and a user-friendly output for the farmer.

## Technology Selections

Mainly we focused on creating our project as a web application and developed the Frontend and back end for our web application mainly based on React Js. The table below shows the core technologies used in our application.

|  |  |
| --- | --- |
| Component | Technology Used |
| Web application (Front end) | React js(with intergration of AdminITE), Material UI, |
| Web application (Back end database) | Firebase Authentication, Firebase Realtime database. |
| Data Science Component | Phython 3.8, Flask API, Rain forest, Jupiter notebook, Google Collab |

Table 1 - Technologies Used

Table 2.3 - Technologies Used

With the development of technology the field of prediction systems are relatively developed and improved. An entire dissertation can be written about the development and software process we’re using within this project. This section provides a concise summary of the development process, highlighting the most salient points and parts in our project.

Our system is developed with a Feature Driven Methodology which will be similar to the waterfall methodology which is used mostly in the industry. We used this method to get the main benefits of the structured approach which is offered by the Feature Driven Development Methodology which could be combines with the use of the prototype used in the evolutionary development model. We used this method in getting the main benefits of the structured approach which is offered by the Feature Driven Methodology which will be combined with an approach of prototype used in the evolutionary development model.

We are using ReactJs with the integration of AdminITE for the simplicity of the application and for it to be more user friendly as this is mostly used by farmers and as they’re not much familiar with technology . Our backend of the application is used with Firebase real-time authentication as it should be updated real time as the stocks and prices of the products are updated with the buyers and sellers in our real-time database. Firebase authentication is used for the login of the application and it’s mostly used as two sections for the farmer and the seller/buyer. And by logging into the system the farmer will be able see the table of content where the farmer will be able to see the sellers/buyers with the vegetable prices.

Technology selections for AgroX are detailed below along with the language selection and libraries/frameworks selection for our data science component.

## Implementation of the Data Science Component

Data Science component is implemented where the farmer can get a predicted real-time price of a selected vegetable and the farmer or the user will need to enter the Vegetable type and the required time to get a prediction of a vegetable at that time according to the dataset we implemented in our data science component. Rain forest is used in implementing the machine learning model and the reason why we can’t build back propagation neural network is due to the insufficient amount of data available. In order to build the model we’ve used Jupiter notebook for the data science component.

**Language selection**

After considering several programming languages available for the data science component, Python was selected as the main programming language for the implementation of this project due to the following factors,

* Flexible - It is an open-source language therefore its most suitable for developers who need to prearrange applications and sites.
* Easy to learn and understand - Python's straightforwardness and meaningfulness make Python an ideal instrument for starting software engineers. Due to its simplicity, its users invest more energy playing with it and less time managing code.
* More support material availability and contributors - Due to the increased popularity of python among developers hence more clients will contribute data on their client experience, and that implies more help material is accessible at no expense.

**Libraries/Frameworks selection**

**Pandas**

It is an open-source package in python which is widely used by professionals for machine learning and data analysis related tasks. In this project, panda allows data importing from excel file format and view data in a data frame.

**Pickle**

The pickle module is used to serialise and deserialise the python objects. In the serialisation process where the object is converted byte stream that can be saved on a hard drive or sent over the internet. In the deserialization, the process byte stream can be retrieved and converted back to the python object.

**Flask**

Flask is a microframework is developed for a web application and it is written in python. It allows you to submit data and obtain an answer in the form of a prediction. Even though there are several web frameworks our team has decided to choose Flask due to its simplicity and ease of work. For this project, the flask uses Werkzeug for creating software objects for functions like request, response, and usage.

**Numpy**

It is a Python library that adds support for large, multi-dimensional arrays and matrices, as well as a large lot of high mathematical functions to manage these arrays.

**Random Forest regression**

Every decision tree has a large variance, but when we combine all of them in parallel, the resultant variance is low because each decision tree is perfectly trained on that specific sample data, and therefore the output is dependent on several decision trees rather than one. In our project, we have implemented Random Forest Regression therefore the average of all the outputs is the final output.

## Implementation of the backend component

Our backend has two parts in this prototype. In first part there can be see authentication, crud operation that store data and display data, and the other part is rain forest algorithm is used to price prediction. As in the first part, we mentioned about authentication. We had used firebase authentication to identify the login users to this prototype and only the authorized users can log in to this system an proceed. And we use firebase real-time database for the crup operation that we use in the prototype to enter sale details and others. To connect the firebase database to our prototype, first we created a firebase web database and added the auto generated firebase API key to the prototype files.

As we are needing a real-time database as in we’re needing a database where the seller can update the tables within the time frame and also the tables are updated when the farmers supply items and the stocks are updated we thought of using Firebase real-time database which is having a strong user based security and also which provides a Firebase Authentication and as it can be used to allow access based on the user identity or with pattern which matches the data. And also we implanted a system where the farmers will be able to send a request if they like to supply vegetables to a required buyer and where the buyer will be able to accept the request within the backend if the buyer is is interested in the request of the farmer. And if the request is accepted by the buyer a notification will be sent to the farmer for the confirmation and also which confirms that the supply request is accepted. We have implemented a section to check the predicted price of a vegetable at a specific time that user gives. That part we implemented by connecting rain forest algorithm with our firebase real-time database.

## Implementation of the frontend component

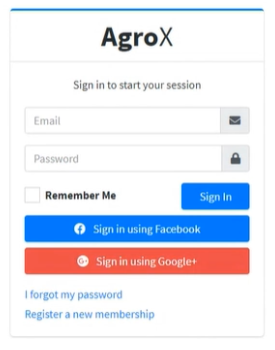
This prototype technology contained react js(with the integration of AdminITE Bootstrap) and Material UI. In this prototype frontend, there are four parts,

* Log In

This login implemented using react js and material UI. This login use to log in for two users. Farmer and Seller can log in separately to each home page.

* Sign Up

Sign Up page also implemented using react js and material UI. Farmer and Seller can be created an account separately in this Sign-Up.



* Seller Home with a table

This part was implemented using the integration of the AdminITE Bootstrap template and react js. The table should contain Buyer Name, Vegetable, max stock, available Stock, and pricePerKg. Sellers should be able to add orders via a model. Sellers are able to delete or update their orders. You can embed two buttons for every table row. One for delete and one for the update.

* Farmer Home with a table

This farmer home was also implemented using the integration of the AdminITE Bootstrap template and react js and The table should contain Buyer Name, Vegetable, max stock, availableStock, and pricePer(1Kg). If the user is a farmer above two buttons should hide and display a button called Accept Order.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Seller | Vegetable | Maximum Stock Accepted | Available Stock | Price per (1Kg) |
|  |  |  |  |  |

Table 3 - Table used in Farmer side Interface

## 1.7 Deployments CI-CD Pipeline

## 1.8 Chapter Summery

The above chapter explains the implementation process in advance and a detailed description of the technologies, the interface and about the system is described. Although at first we planned in implementing two sections as a mobile application and web application we limited it only to a web application by considering the educational and financial status of the users considering the expenses and the data used for the usage of the applications. And designed the web application with a simple and an easy user interface for the users to get a better use of the application and for the users to use the application without any prior understanding or an idea about the application. Data science component is also described separately with the implementation and the user interface of the application is explained. Simply the project is explained and the functionalities are described in detail. Testing criteria’s will be explained in the next chapter.

# Chapter 02 – Testing

## 2.1 Chapter Overview

From the previous chapter the implementation phase of the Market Value prediction system is discussed. This chapter will be focused on the testing phase of our Market Value Prediction System. After the goals and objectives of testing are discussed, the testing criteria will be discussed in detail. Testing of Functional and Non Functional requirements will be discussed after that to make sure that it will meet all the requirements for the implementation report and to ensure it meets all the required standards. And this chapter includes testing information for the system's functions as well as the overall system's performance test results. Description about the limitations faced during the testing process with an evaluation of the testing results are concluded in this chapter. Finally, this chapter will include a thorough overview of the shortcomings encountered during the testing process, as well as an assessment of the testing results.

## 2.2 Goals and objectives of testing

We carried out the testing phase to verify the prototype functions which are implemented according to the requirements which are identified during the requirement elicitation phase.

The objectives of the testing phase for the implemented prototype is shown below.

* To identify the errors and bugs which are possible in the implemented application and to ensure that they are clearly fixed before the final product is deployed.
* To improve and enhance the application we designed based on the test results.
* To test and verify the functional requirements of the web application as proposed.
* To test and verify the non-functional requirements of the web application as proposed.

## 

## 2.3 Testing Criteria

The project's testing will be carried out using the test plan template, which will primarily concentrate on functional and non-functional testing. Test cases for the system's core functional and non-functional specifications were defined to verify the application's test coverage. These test cases will define a set of criteria that will be used to determine whether the Price prediction framework is functioning properly.

## 2.4 Testing functional requirements

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test case no. | Feature tested | Test case description | Test case condition | Expected result | Actual result | Status |
| 1.0 | FR1 | The farmer can see the predicted price | We take test data into consideration.  Vegetable Type =Beans  Time = 12.45p.m | Price per Kilogram=110 | Price per Kilogram=93.5 | Pass  (85%) |
| 2.0 | FR2 | The farmer can see available amount of stock | Buyer has maximum stock of 20Kg of beans. He has already 10kg beans in his stock .The remaining stock will display to the farmer via the app. | Available stock=10 | Available stock=10 | Pass  (100%) |
| 3.0 | FR3 | User selects to sign up | Sign Up the web application | Sign Up from appears | Sign Up from appears | Pass  (100%) |
| 4.0 | FR4 | User submit correct data to signup | input username and password and signup | Display registration successful message | Display registration successful message | Pass  (100%) |
| 5.0 | FR5 | User submit incorrect data to signup | Submit incorrect data to user name and password | Display error message | Display error message | Pass  (100%) |
| 6.0 | FR6 | User enters valid data to login | To login, the user must have valid information. | Displays login successful message | Displays login successful message | Pass  (100%) |
| 7.0 | FR7 | User enters invalid data to login | When attempting to log in, the user provides incorrect information. | Display error message | Display error message | Pass  (100%) |
| 8.0 | FR8 | User looks for available stock | The user searches for stock that is currently available. | Display available stock | Display available stock | Pass  (100%) |

## 2.5 Testing non-functional requirements

## 2.6 Unit testing

During the development stage, unit tests were performed, and any bugs discovered during the testing stage were addressed.

## 2.7 Performance testing

Performance testing for the web application is covered in depth in this section, which includes Load Testing, Stress Testing, Spike Testing, and Soak Testing. These experiments were carried out using Microsoft Azure's built-in testing tools for the site backend. The client's success and response times will be addressed in the final section.

## 2.8 Usability testing

Usability testing determines how user-friendly the system is and how well the user interface design and experience are managed. The web application for price prediction was created using user interface principles that prioritized simplicity. It has a one-of-a-kind logo that was created from the ground up, as well as a completely responsive web interface that works on all browsers and computers.

## 2.9 Compatibility testing

Compatibility ensures that the device functions properly in all browsers. Google Chrome, Safari, Mozilla Firefox, and Microsoft Edge were used to test the price prediction web application. All of the features worked flawlessly in all browsers.

## 2.10 Chapter Summary

The test results for the system's functions were originally contained in this chapter.This chapter went through the specifics of price prediction system testing phase, including the system's goals and objectives, testing the system's functional and non-functional specifications, and the weaknesses encountered during the testing process. The system's critical assessments are discussed in the following sections.

This chapter initially contained the test results for the functions of the system. Next it showed  
the performance testing results for the client side and server side of the web application simply. It showed a described functionality of our application and how testing’s are done for a better and an accurate deployment and a quality output for our web application.

# Chapter 3 – Evaluation

## 4.1 Chapter Overview

This chapter elaborate the author evaluation subsequently the expert overview and the customer  
feedback. In the first part, it describes the author experiences, the lessons they learnt throughout  
the project, the project methodology they used and the challenges they faced. In the second  
half, it discusses about the Expertise overview and their critical analysis regarding the project  
context and their suggestions for further improvements. Finally, it describes the user experience  
of the product and their feedback.

## 4.2 Evaluation Methods

## Quantitative Evaluation

## Qualitative Evaluation

## 4.2 Author Evaluation

As planned we were willing to provide a best application which is simple and easier to use and we wanted to make our application usable even for non-educated and low income citizens. We wanted to make our application and usage more cost effective for the user so even we considered about the data usage while using because for the citizens who are using our application even a single cent will be precious. So due to that reason by interviewing several personalities in the sector we came into an conclusion of developing only a Web application for our project because if we implement a mobile application also it will cost more for the user for data when downloading the app and even when using the app compared to the web application. So although we planned to develop a mobile application also we decided to implement only a web application by the interviews we took with the users.

After evaluating the prototype, we move forward for the second phase of development. There  
we completed the backend solution and improved UI and UX. At the end of the second phase  
of development all the functions were working smoothly as we planned.

And finally we were able to implement a simple and a user friendly interface for any user and we were able to provide the users an economical and a user-friendly application which is a huge achievement for us and for the entire agricultural industry. We are happy to present that now the farmers will not have to face in unfair situations when their hard work is brought into a price. So now we are able to give the predicted price for each vegetable before farmers take their harvest to the markets to sell which is an unexplainable achievement for our group and also for the whole agricultural sector.

## 4.4 Limitations of the research

* Limitations of the availability of sufficient data

Due to the limitations of the availability of sufficient data, we couldn’t able to develop the BPNN neural network. Therefore, we built three regressions models to predict the price. Lasso regression model, Multi linear model, and the random forest model. Out of them, the random forest model gives a better accuracy than the other models. So, we created the random forest model to predict the price.

* Supported Language is English

The web-based application is currently only supports the English language, although Sinhala and Tamil languages support can be added in the future. This will especially targets for the farmers.

## 4.5 Future enhancements

* **Including the use with all three languages. (Sinhala, English, Tamil)**

Our system is currently implemented with English language for the ease of the developers and due to the lack of time to implement, so we’re planning to implement the application in all three languages (Sinhala, Tamil and English) for the better readability of the user and ease of use.

* **Expanding the Functions of the Application**

We developed our application in the most simple manner to make it easier for the users to use but after some time and after the users of the application increases and after the users become familiar with the application we are planning to expand the core functions and optional functions by adding more optional functions such as tools to Analyze the sales and supplies and to improve the analytics of the business.

* **Developing the interface with theme facility.**

We’re planning to implement a theme function to the application where the users will have access to edit the interface according to their requirement for better readability and ease of use. Where the users will be able to change font sizes and background colors accordingly.

## 4.6 Extra work (Competitions, research papers, etc)

## 4.7 Concluding remarks

At the start of the project we set ourselves objectives to be achieved in the course of the 6  
months. We are happy that at the end of the six months we have completed all the objectives  
that we intended to complete and finally we are ready to deploy an important application which will be really important for the farmers and the fair shop owners and which will be a really interesting and an important in the future.