TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

A

PROJECT REPORT

ON

AGRO-TECH

SUBMITTED BY:

AKASH PANTHI(PUL075BCT010)

AMRIT ARYAL(PUL075BCT011)

BIGYAN KOIRALA(PUL075BCT019)

BIPIN KHANAL (PUL075BCT022)

SUBMITTED TO:

DEPARTMENT OF ELECTRONICS & COMPUTER ENGINEERING

May 1, 2023

Page of Approval

TRIBHUVAN UNIVERSIY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS
DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

The undersigned certifies that they have read and recommended to the Institute of Engineering for acceptance of a project report entitled **"Agro-Tech"** submitted by **Akash Panthi**, **Amrit Aryal**, **Bigyan Koirala**, **Bipin Khanal** in partial fulfillment of the requirements for the Bachelor's degree in Electronics & Computer Engineering.

.....
Supervisor
Ganesh Gautam
Assistant Professor
Department of Electronics and Computer Engineering,
Pulchowk Campus, IOE, TU.

.....
Internal examiner
.....
Assistant Professor
Department of Electronics and Computer Engineering,
Pulchowk Campus, IOE, TU.

.....
External examiner
.....
Assistant Professor
Department of Electronics and Computer Engineering,
Pulchowk Campus, IOE, TU.

Date of approval:

Copyright

Acknowledgments

We would like to extend our sincere appreciation to all the individuals who have supported us in the completion of this project. Firstly, we express our gratitude to our project supervisor Ganesh Gautam sir, who provided us with valuable guidance, motivation, and support throughout the project. We are grateful for their time and effort in reviewing and providing constructive feedback on our project proposal, mid-term report, and final report.

We would also like to extend our heartfelt thanks to the farmers and retailers who participated in our project and provided us with the necessary data, feedback, and insights into the challenges faced by the agricultural sector in Nepal. Their willingness to share their experiences and knowledge with us was invaluable in shaping our project's direction and ensuring its relevance to the needs of the agricultural community.

Finally, we would like to thank the University for providing us with the opportunity to undertake this project and develop our skills and knowledge in the field of agriculture and technology. The University's resources, facilities, and support have been instrumental in the successful completion of our project.

In conclusion, we express our heartfelt gratitude to all those who have contributed to the successful completion of this project, and we hope that our project will contribute towards addressing the challenges faced by the agricultural sector in Nepal and promoting efficient and sustainable agricultural supply chains.

Abstract

The mid-term report for our project titled "Agro-Tech" provides an update on our progress and outlines the remaining tasks. Our platform aims to connect retailers and farmers for the efficient buying and selling of vegetables, with a focus on reducing post-harvest loss and ensuring food safety and freshness.We have made significant strides in developing the platform, including creating a user-friendly interface for retailers and farmers, and establishing partnerships with local farmers. We have also conducted market research and identified key areas for improvement, such as increasing the variety of vegetables available and expanding our reach to more retailers and farmers.Moving forward, we plan to focus on increasing the number of farmers on our platform and improving the logistical system for transporting vegetables from the farm to the retailer. We will also continue to prioritize food safety and freshness through the use of cooling infrastructure and adherence to global logistical standards. Overall, we are confident that our project has the potential to make a positive impact on the agro-based industry in Nepal, contributing to the overall development of the economy, reducing poverty and unemployment, and promoting equality.

Keywords: *retailers, farmers, vegetables, post-harvest loss, food safety, freshness, user-friendly interface, payment system, partnerships, market research, logistical system, cooling infrastructure, global logistical standards, impact, agro-based industry*

Contents

List of Figures

List of Abbreviations

TU	Tribhuvan University
DB	Database
JS	Javascript
SMART	Smallholder Market Access and Resilience Programme
UNDP	United Nations Development Programme
IFAD	International Fund for Agricultural Development
GDP	Gross Domestic Product
FtMA	Farm to Market Alliance
SQL	Structured Query Language
SSPL	Server Side Public License
OS	Operating System
UI	User Interface
API	Application Programming Interface

1. Introduction

Agriculture is a vital sector in Nepal's economy, employing over two-thirds of the workforce and contributing to more than one-third of the country's GDP. Despite its importance, the sector faces various challenges such as post-harvest losses, limited access to markets, and lack of proper infrastructure. To address these challenges, there has been an increased interest in using technology to improve the efficiency of agro-based industries. In this context, our project "Agro-Tech" aims to develop a platform that connects retailers and farmers for the efficient buying and selling of vegetables.

1.1 Background

Agriculture is a significant sector in Nepal, employing a significant proportion of the population and contributing to the country's economic growth. However, agro-based industries in Nepal face challenges related to post-harvest losses, lack of proper infrastructure, and limited access to markets. These challenges limit the potential of the agriculture sector to contribute to the country's overall development.

In recent years, there has been an increased interest in the potential of technology to address these challenges and improve the efficiency of agro-based industries. The use of technology can help to reduce post-harvest losses by improving the logistics of transportation and storage, providing better access to markets, and promoting food safety and freshness. In this context, our project, "Agro-Tech" aims to develop a platform that connects retailers and farmers for the efficient buying and selling of vegetables. The platform will incorporate key features such as a user-friendly interface, a payment system, and a logistical system that ensures the freshness and safety of produce. By doing so, we hope to contribute to the growth of the agro-based industry in Nepal and support the country's overall economic development.

1.2 Problem statements

Around 68% of our country's population directly depends on agriculture for livelihood. Still, the production is not enough to supply food for the whole country. As a result, wastage is the main problem that is hindering feeding the country. The main cause of the wastage is the lack of a proper market for the farmers. Sometimes they are compelled to sell the products even at a very low-cost price due to which the proper returns for them are very low. To solve the problem of a market for the farmers and to serve them with a reasonable price for

their products, we are proposing a platform linking between these entities namely farmers and purchasers (retailers). This project mainly promotes the value of farmer products and ensures quality products to the retailers and customers at a reasonable price. This is possible by removing the multiple third parties' involvement between the farmers and customers.One of the significant challenges faced by the agro-based industry in Nepal is post-harvest losses. Due to the lack of proper storage and transportation facilities, a significant portion of agricultural produce is lost before it reaches the consumers. Additionally, small-scale farmers often have limited access to markets, which makes it difficult for them to sell their produce at fair prices. This leads to low incomes and limited opportunities for investment in their agricultural activities.

- Farmers experience price risk, information asymmetry about demand, distribution inefficiency, and receive late payments.
- Retailers face problems like higher costs, low quality and unhygienic produce, high price volatility, and the everyday hassle of going to the market.
- The traditional Supply Chain is highly inefficient, unorganized, and has a high rate of food wastage.

1.3 Objectives

The main objectives of this project are:

- To develop a platform that connects retailers and farmers for the efficient buying and selling of vegetables.
- To reduce post-harvest losses by providing better access to markets, improving the logistics of transportation and storage, and promoting food safety and freshness.
- To support small-scale farmers by providing them with fair prices for their produce, enabling them to increase their incomes and invest in their agricultural activities.

2. Literature Review

Agriculture is one of the most important sectors of the Nepalese economy, providing employment opportunities for a significant portion of the population and contributing to economic development. However, the sector is facing numerous challenges, including limited access to markets, lack of infrastructure, and low productivity levels.

Agro-based industries, which utilize local raw materials, have the potential to significantly contribute to the overall development of the economy and increase gainful employment.Recent studies have highlighted the importance of efficient and sustainable agricultural supply chains in promoting economic growth and reducing poverty. The use of technology and innovation has been identified as a key factor in enhancing supply chain efficiency and reducing post-harvest losses. The development of platforms that facilitate direct buying and selling between farmers and retailers has also been recognized as a promising solution to the challenges faced by the agricultural sector.

In Nepal, there have been several initiatives aimed at improving the agricultural supply chain and promoting agro-based industries. For example, the government of Nepal has implemented policies to promote the use of local raw materials, establish agricultural cooperatives, and improve infrastructure. However, challenges such as limited access to markets, high transportation costs, and lack of cold chain facilities continue to impede the growth of the sector.Recent studies have highlighted the potential of technology in addressing these challenges. The use of e-commerce platforms and mobile applications has been shown to improve market access and reduce transaction costs. These platforms also provide farmers with access to price information and facilitate direct interaction with buyers, reducing the need for intermediaries.

2.1 Related work

Several initiatives and projects have been implemented in Nepal and other countries to address the challenges faced by the agricultural sector and promote efficient and sustainable agricultural supply chains. Some of the related works are:

Kisan Network - Kisan Network is an Indian start-up that provides a mobile-based platform for farmers to connect with buyers, access market information, and manage logistics. The platform allows farmers to set their own prices and enables direct selling to buyers, reducing the need for intermediaries.

eKheti - eKheti is a Nepalese mobile application that provides farmers with access to market information, crop management advice, and weather forecasts. The app also facilitates direct buying and selling between farmers and consumers.

AgroZ - AgroZ is a platform that connects farmers directly with retailers and food processors, allowing for efficient and sustainable supply chains. The platform provides farmers with access to price information and enables direct interaction with buyers, reducing the need for intermediaries.

The Smallholder Market Access and Resilience Programme (SMART) - The SMART programme is a joint initiative by the United Nations Development Programme (UNDP) and the International Fund for Agricultural Development (IFAD) aimed at improving smallholder farmers' access to markets in Nepal. The programme provides technical assistance, market linkages, and financial support to smallholder farmers.

Farm to Market Alliance (FtMA) - FtMA is a public-private partnership that aims to improve smallholder farmers' access to markets in Africa. The initiative provides technical assistance, market linkages, and financial support to smallholder farmers, with a focus on developing efficient and sustainable agricultural supply chains.

These initiatives and projects demonstrate the potential of technology and innovation in promoting efficient and sustainable agricultural supply chains and improving smallholder farmers' livelihoods. Our project "Agro-Tech" aims to contribute to this effort by creating a platform for efficient and sustainable buying and selling of vegetables between farmers and retailers in Nepal.

3. General Theory

3.1 Random Forest Algorithm

Random Forest is a supervised machine learning algorithm used for classification and regression tasks. It is an ensemble learning method that combines multiple decision trees to make more accurate predictions than a single decision tree.

The algorithm works by creating a forest of decision trees, where each tree is trained on a random subset of the input data and a random subset of the input features. During training, each decision tree independently makes a prediction, and the final prediction is determined by aggregating the predictions of all the trees. This process is called "bagging" or "bootstrap aggregating," and it helps to reduce the variance of the model and prevent overfitting.

Random Forest also employs a technique called "feature bagging," where only a random subset of features is considered for each decision tree. This helps to reduce the correlation between trees and improves the accuracy of the model.

In classification tasks, Random Forest uses the mode of the predicted classes from all the trees as the final prediction. In regression tasks, it uses the average of the predicted values from all the trees as the final prediction.

Decision Tree

In a decision tree within a Random Forest, the data is split at each node based on the feature that provides the greatest information gain, which is calculated by measuring the reduction in entropy or Gini impurity. The splitting process continues recursively until a stopping condition is met, such as reaching a maximum tree depth or the minimum number of samples required to split a node.

Key Benefits

- Reduced risk of overfitting: Decision trees run the risk of overfitting as they tend to tightly fit all the samples within training data.
- Provides flexibility: Since random forest can handle both regression and classification tasks with a high degree of accuracy, it is a popular method among data scientists.
- Easy to determine feature importance: Random forest makes it easy to evaluate variable importance, or contribution, to the model.

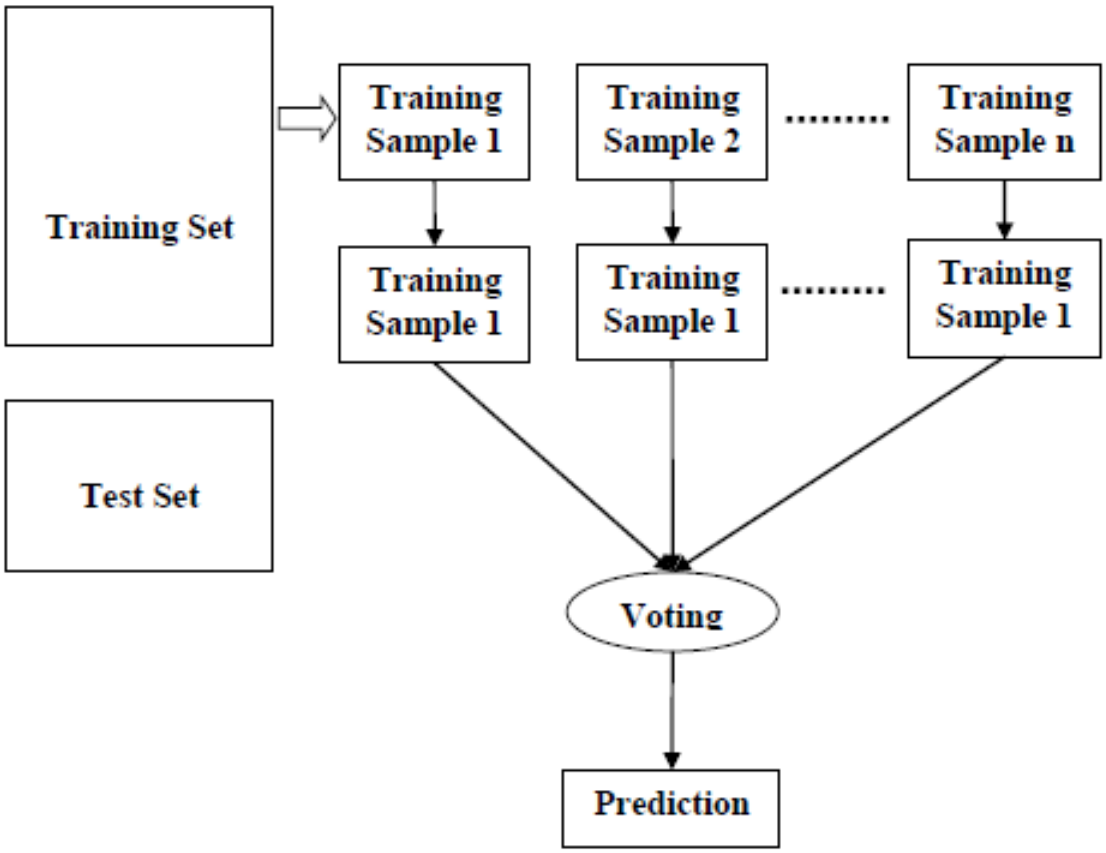

Figure 3.1: Random Forest Algorithm

Key Challenges

- Time-consuming process: Since random forest algorithms can handle large data sets, they can provide more accurate predictions, but can be slow to process data as they are computing data for each individual decision tree.
- Requires more resources: Since random forests process larger data sets, they'll require more resources to store that data.

3.2 MongoDB

MongoDB is an open-source document-oriented database that is designed to store a large scale of data and also allows you to work with that data very efficiently. It is categorized under the NoSQL (Not only SQL) database because the storage and retrieval of data in the MongoDB are not in the form of tables.

The MongoDB database is developed and managed by MongoDB.Inc under SSPL(Server Side Public License) and initially released in February 2009. It also provides official driver support for all the popular languages like C, C++, C, and .Net, Go, Java, Node.js, Perl, PHP, Python, Motor, Ruby, Scala, Swift, Mongoid. So, that you can create an application using any of these languages. Nowadays there are so many companies that used MongoDB like Facebook, Nokia, eBay, Adobe, Google, etc. to store their large amount of data.

How it works?

Now, we will see how actually thing happens behind the scene. As we know that MongoDB is a database server and the data is stored in these databases. Or in other words, MongoDB environment gives you a server that you can start and then create multiple databases on it using MongoDB. Because of its NoSQL database, the data is stored in the collections and documents. Hence the database, collection, and documents are related to each other as shown in figure below.

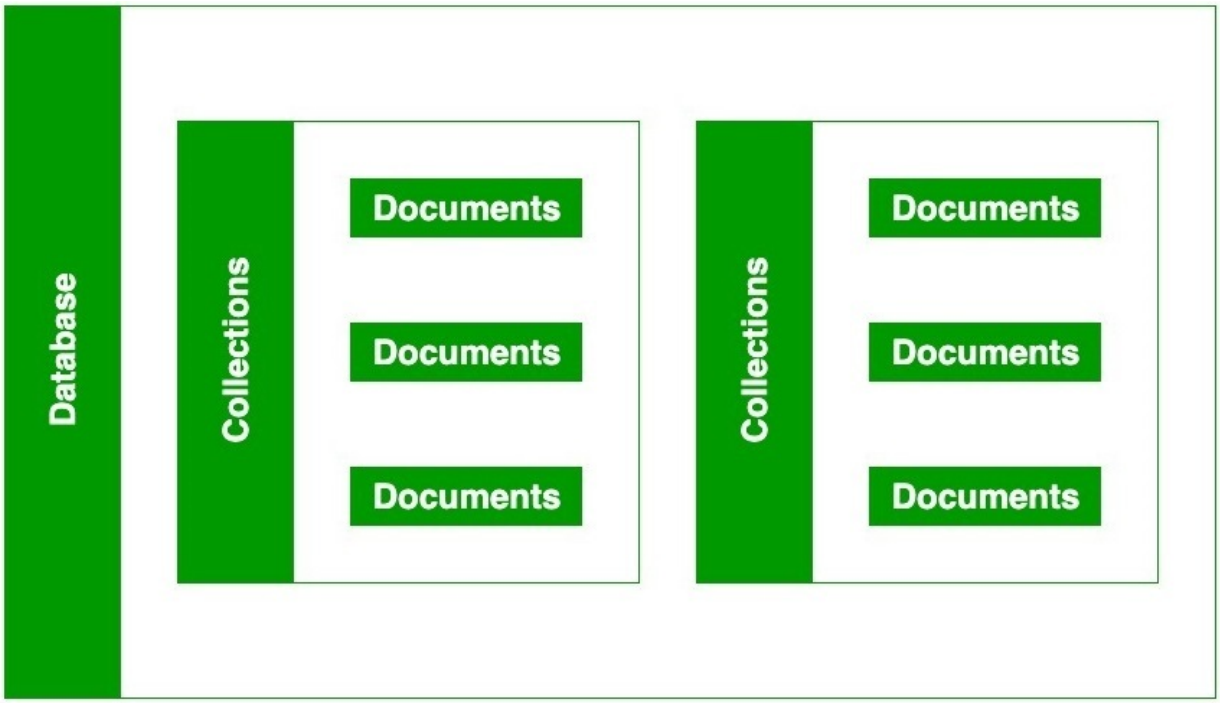

Figure 3.2: MongoDB Structure

Inside of the collection we have documents. These documents contain the data we want to store in the MongoDB database and a single collection can contain multiple documents and you are schema-less means it is not necessary that one document is similar to another. The documents are created using the fields. Fields are key-value pairs in the documents, it is just like columns in the relation database.

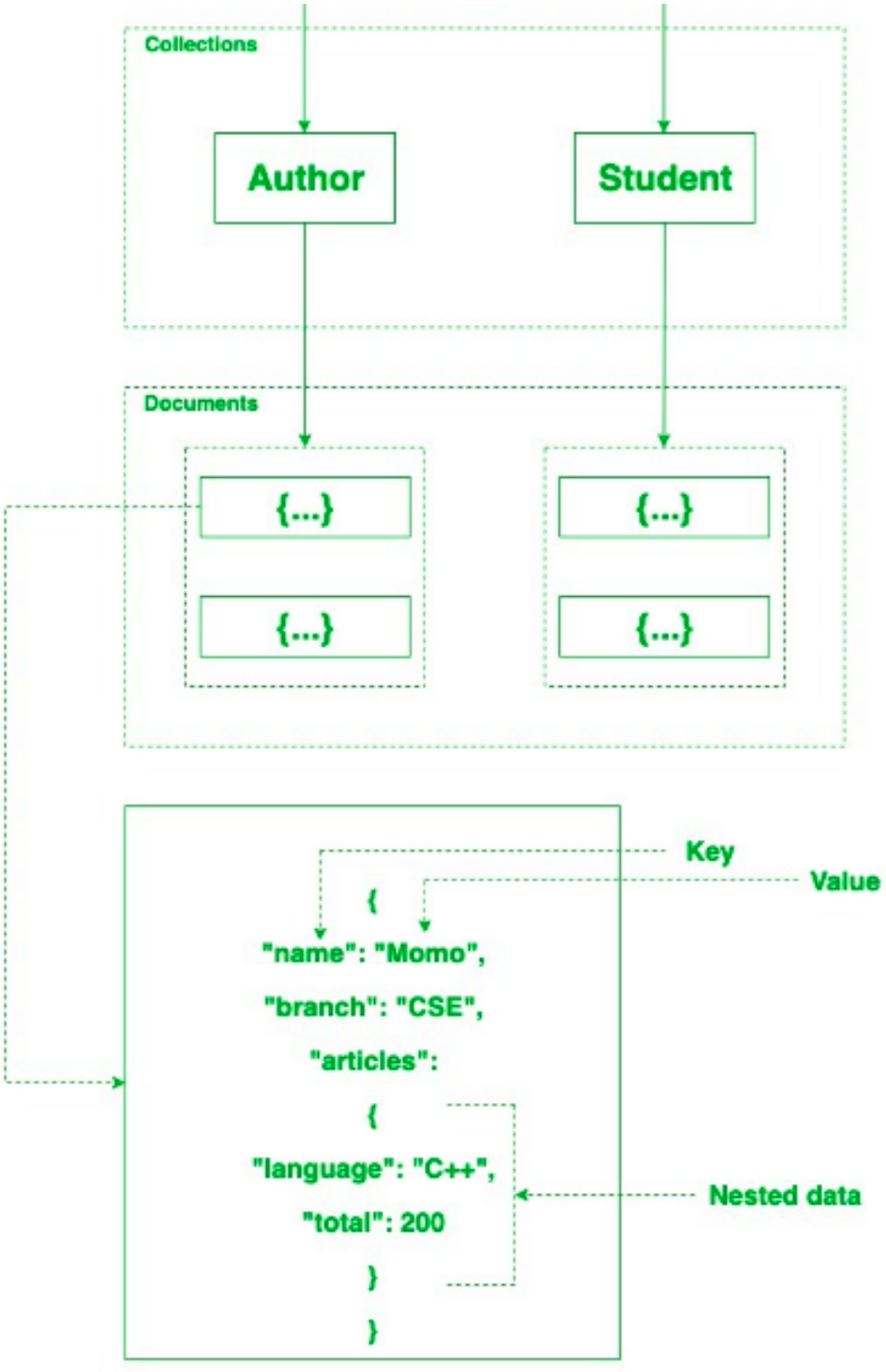

Figure 3.3: Database Sample

3.3 MVC Design Pattern

MVC, short for Model, View, and Controller, is a methodology or architectural pattern used for efficiently relating the user interface to underlying data models and organizing to relate the application code. MVC is primarily used to separate an application into three main components: Model, View, and Controller.

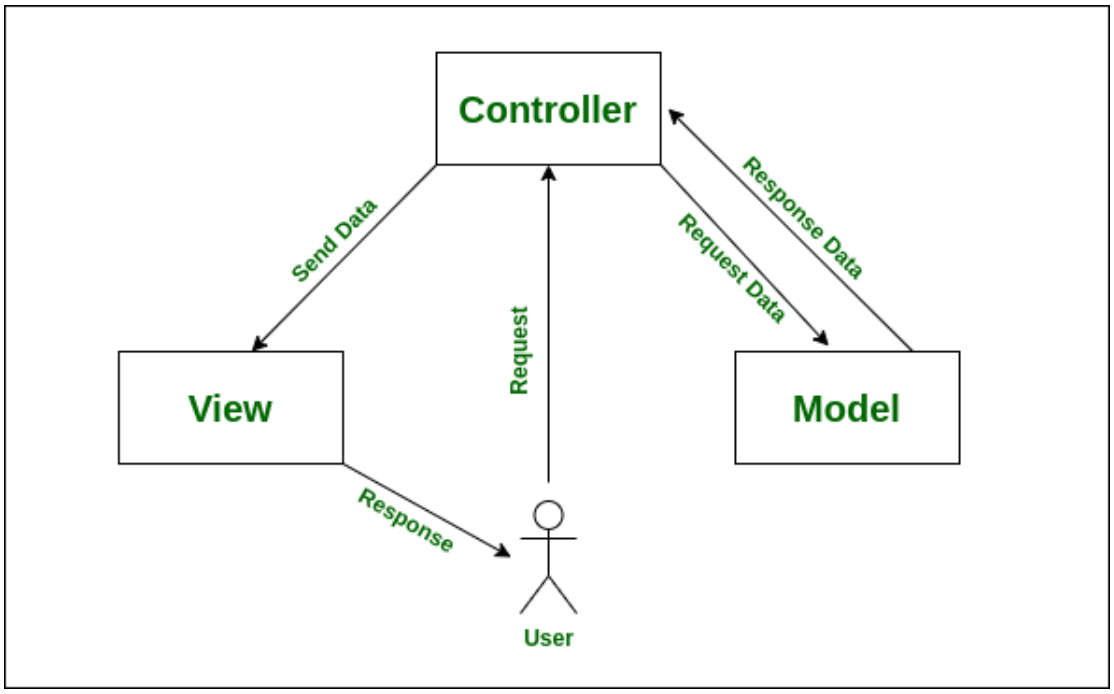

Figure 3.4: MVC Design

3.3.1 Model

This level is considered the lowest level when compared with the View and Controller. It primarily represents the data to the user and defines the storage of all the application's data objects. It also contains the application logic.

3.3.2 Views

This level is majorly associated with the User Interface(UI) and it is used to provide the visual representation of the MVC model. In simpler terms, this level deals with displaying the actual output to the user. It also handles the communication between the user (inputs, requests, etc.) and the controller.

3.3.3 Controller

This level takes care of the request handler. The controller completes the cycle of taking the user output, converting it into desired messages.

3.4 REST API

Representational State Transfer (REST) is an architectural style that defines a set of constraints to be used for creating web services. REST API is a way of accessing web services in a simple and flexible way without having any processing.
REST technology is generally preferred to the more robust Simple Object Access Protocol (SOAP) technology because REST uses less bandwidth, simple and flexible making it more suitable for internet usage. It's used to fetch or give some information from a web service. All communication done via REST API uses only HTTP request.

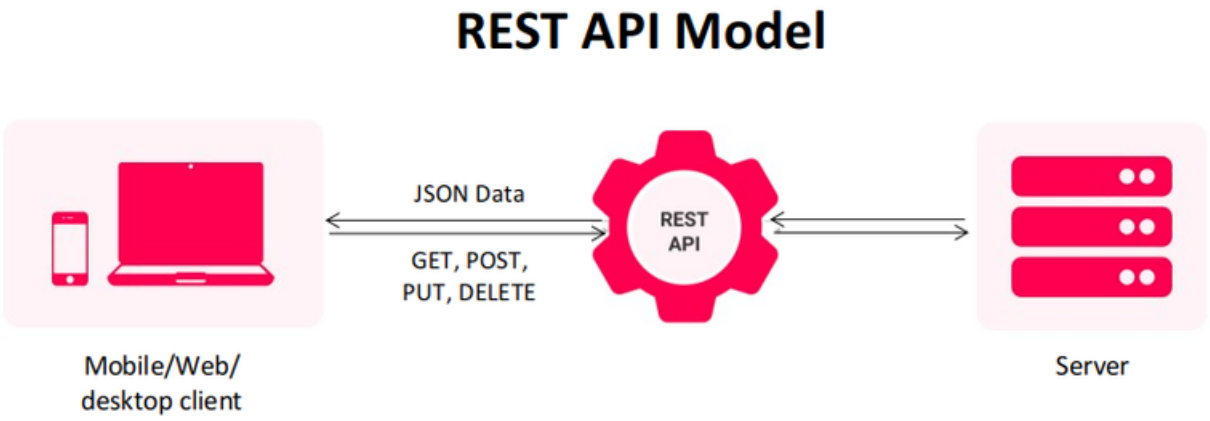

Figure 3.5: REST API

3.4.1 Working

A request is sent from client to server in the form of a web URL as HTTP GET or POST or PUT or DELETE request. After that, a response comes back from the server in the form of a resource which can be anything like HTML, XML, Image, or JSON. But now JSON is the most popular format being used in Web Services.
Characteristics of a REST API:

- Client-Server Architecture
- Statelessness
- Cacheability
- Code-on-Demand

3.5 JSON Web Token

JSON Web Token is an open standard for securely transferring data within parties using a JSON object. JWT is used for stateless authentication mechanisms for users and providers, this means maintaining session is on the client-side instead of storing sessions on the server. JWTs can be signed using a secret or a public/private key pair.

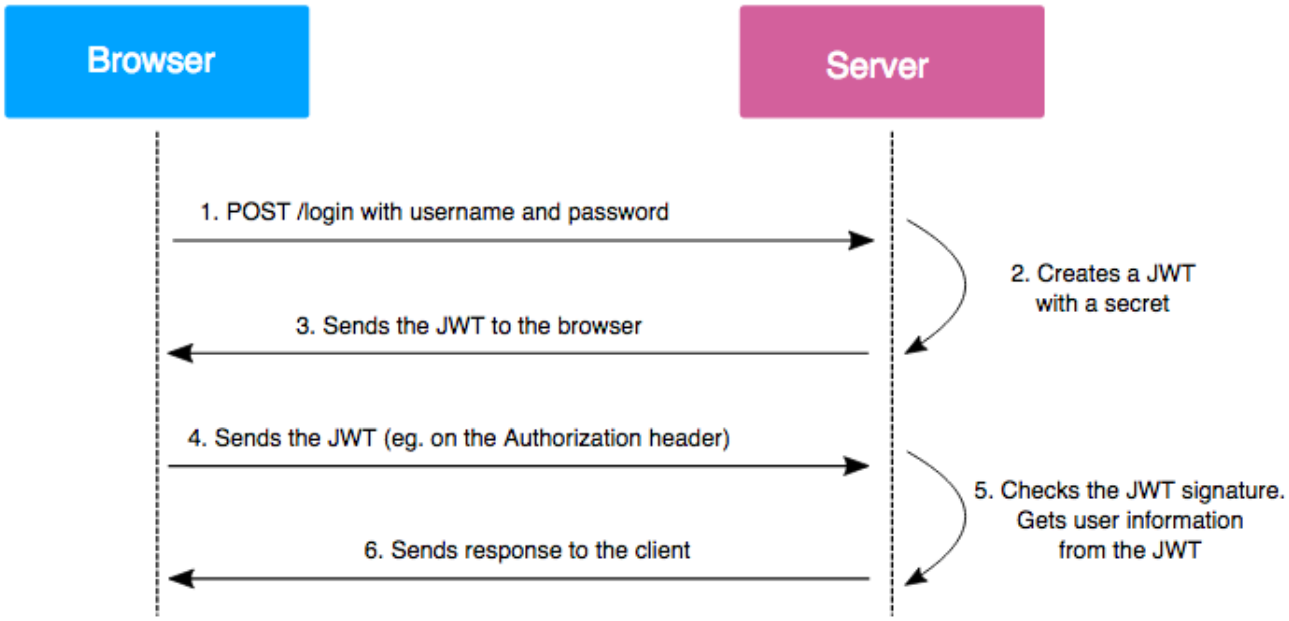

Figure 3.6: JSON Web Token

Node.js authentication with JWT has several advantages over the traditional authentication process, primarily the scalability of stateless applications. And since it's becoming popular among such heavyweights as Facebook and Google, it's adoption across the industry likely will continue to grow.
Other advantages include:

- Simple verification through a JSON Web Token
- Can be used as an authentication service
- Provides more trustworthiness than cookies or sessions

4. Technologies Used

Different components of hardware and software are combined together to build this system. These parts work together in sync to complete a task that satisfies the platform's goal of offering users dependable service uptime and persistence performance whenever required. These components includes

4.1 Web Server

For anyone on the internet to access the system, it needs to be served in a computer which can manage, allow connections to the application and handle the requests. This computer which has a wide range of responsibility to make the application accessible is referred as the web server. Web Servers can be both with linux OS and a windows OS. But, the more preferrable option is to go with the linux OS because it is more managable and grants the administrator more grasp to the underlying features of the system. For now, it has been served through local hosting for handling requests for application.

4.2 Database

The database is an essential component for any web application. It is responsible for storing and managing the data that the application uses. For Agro-tech, a NoSQL database called MongoDB (Atlas) is used. MongoDB (Atlas) is a powerful database that can handle large amounts of data and many connections simultaneously. The hardware and other requirements for the database depend on the expected number of connections and the amount of data to be stored.

4.3 Expert system

The prediction of price is responsible for simulating the price of different items available in our dataset. Expert systems are designed to solve complex problems in a specific domain by reasoning through the knowledge base using a set of predefined rules. It uses random forest algorithm to generate expected/estimated price. The server is built using Python and the Flask web framework.

4.4 Frontend

4.4.1 React

A web application serves as the frontend of the Agro-tech. Popular javascript library ReactJS has been utilized for the frontend part. React is a popular open-source JavaScript library for building user interfaces (UIs) in web applications. It uses a declarative approach to programming, which means that developers describe how the UI should look and React handles the details of updating the UI in response to changes in data or user interactions.

4.4.2 Tailwind CSS

Tailwind CSS is a utility-first CSS framework that allows developers to quickly and easily create responsive user interfaces. Unlike other CSS frameworks that provide pre-designed components, Tailwind CSS provides a set of pre-defined utility classes that developers can use to quickly style their HTML elements. These classes can be combined to create complex styles and layouts without writing custom CSS code. Tailwind CSS is highly customizable and can be configured to fit the needs of any project. It's designed to work well with modern front-end tools like React, Vue, and Angular.

4.4.3 Figma

Figma is a web-based design tool used for creating user interfaces, prototypes, and design systems. It allows designers and teams to collaborate and work together in real-time, making it an ideal tool for remote teams. With Figma, we can design, prototype, and share your work all in one place. It offers a variety of design tools, such as vector editing, prototyping, and commenting, that make the design process more efficient and effective.

4.5 Backend

As our application has price prediction system integrated, microservice architecture is used where each service typically has its own database and communicates with other services through well-defined APIs.

Microservices, often referred to as Microservices architecture, is an architectural approach that involves dividing large applications into smaller, functional units capable of functioning and communicating independently.

Two popular frameworks viz Flask (Python framework) Express.js (Nodejs framework) are used as the backend of this application because python and javascript being ideal to work with.

4.5.1 NodeJS

Node.js is an open-source and cross-platform JavaScript runtime environment. It is a popular tool for almost any kind of project. Node.js runs the V8 JavaScript engine, the core of Google Chrome, outside of the browser. This allows Node.js to be very performant.
A Node.js app runs in a single process, without creating a new thread for every request. Node.js provides a set of asynchronous I/O primitives in its standard library that prevent JavaScript code from blocking and generally, libraries in Node.js are written using non-blocking paradigms, making blocking behavior the exception rather than the norm.

4.5.2 Flask

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions.

5. Methodology

5.1 System Block Diagram

A client, a server, and a database layer make up the three layers of the Agro-Tech. A three-tier architecture offers layer-level flexibility to integrate substantial and reliant elements into the system while being appropriate for horizontal scalability. The figure below illustrates the general design of the Agro-Tech system.

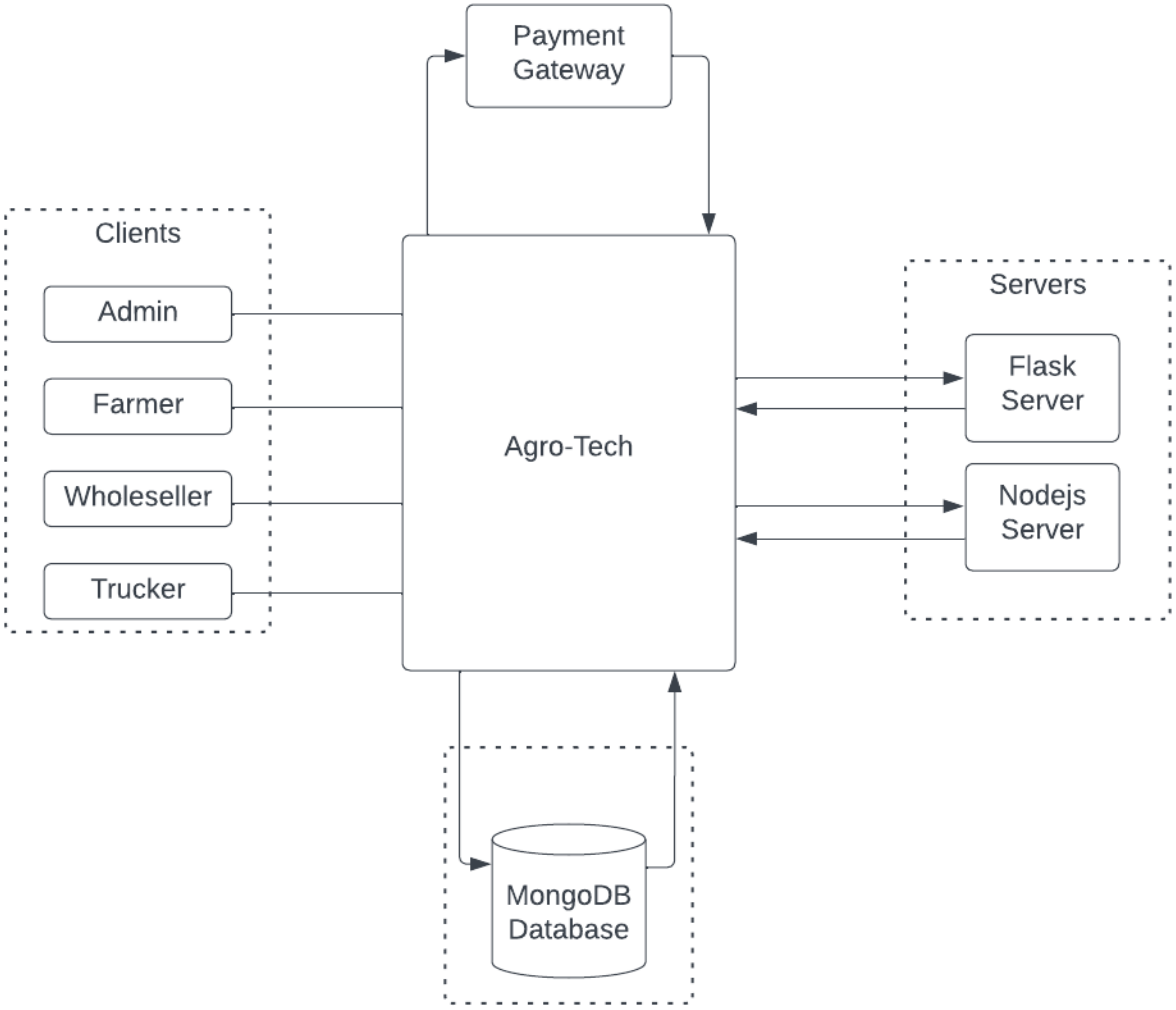

Figure 5.1: System Block Diagram

We have divided the our systems into different parts: In essence, the system can be broken down into four categories of subsystems. The following is a description of the subsystems:

Client

Users interact with the system through a client, which is a graphical user interface. Users can access an interface based on their jobs related to the agro-tech client system. Farmer, wholesaler, trucker, or Agro-tech administrator are just roles as of now. Each interaction can be described as a view of the client system. Any user attempting to access the system must first receive server authentication. They must first register on the system with the necessary information in order to do this. Users can now access their dashboard shown as a view by using those account credentials.

1. **Farmer View:** As the name suggests, the farmer interactions refer to as farmer view. Agro-tech allows a farmer to add their products with required fields along with the status of their products i.e. whether their products are being bought or in processing or being delivered. They can enter their products' price within the mentioned range of price which is calculated using an expert system from a huge datasets.

2. **Whole seller view:** Agro-tech allows a customer to order a service, view its details, pay for the order and provide review and rating to the products and the corresponding farmer. Buyers can also search for the required product based on the category.

3. **Trucker View:** For the selected products that are under processing and are ready to be sold, the admin assign truckers, based on their location and quantity of products, to deliver the products to the collection center.

4. **Admin View:** An admin is able to manage all kinds of users including farmers and whole sellers, and also the services provided . Admin can verify newly registered users and can allow selected users to work in the platform. Admin also provides the transaction management payment mechanism. Admin is also responsible for managing the underlying data and making manipulations whenever necessary in the system.

Server

The server is the most integral part of the Agro-tech System. It is responsible for handling all the data processing and client requests. In the Agro-tech system, the server is responsible for Price Estimation, Tasks execution and Handle Database.

1. **Node.js Server:** Node.js is a popular open-source, cross-platform, server-side JavaScript runtime environment that allows developers to build fast and scalable web applications. Node.js also provides several popular frameworks, such as Express.js, that simplify the process of creating a server and defining routes. These frameworks include built-in support for middleware, which are functions that can be used to modify incoming requests or outgoing responses.

2. **Flask Server:** Flask is a popular open-source web framework for building web applications using the Python programming language. Flask provides a lightweight and flexible approach to building web applications that is easy to learn and use. Since our system is based on microservice architecture, Flask server is used to get the predicted price which is calculated using Random Forest algorithm training a huge datasets.

Database Agro-tech uses MongoDB in order to store data. MongoDB is a non-relational database management system which is efficient for handling more database connections and provides flexibility options during scaling. MongoDB Atlas is a fully-managed cloud database that handles all the complexity of deploying, managing, and healing deployments on the cloud service provider.

5.2 Description of Working Principle

5.2.1 Managing Purchasing Order

Purchase handling is the basic feature that needs to be managed properly. When the users (customers and retailers) make the purchase request, it is then handled by the system to track the location of delivery.

5.2.2 Product Monitoring

All the products that are updated by the farmers are visible to the users and purchases can be made at any time. These quantities of products are monitored by the system. Since the purchase and sell chain is not a linear activity. So it is necessary to monitor the system state and update accordingly.

5.2.3 Tracing Transaction

Once the purchasing and delivery are done, the most important part of the system is maintaining the proper transaction and keeping its record. The transaction should be handled by the system autonomously.

5.2.4 Farmer

Farmers in this system have their own profiles where they can update their available products. They can set the desired amount but it should be within the range provided by the system. Farmer can change the details of the products like quantity, price etc. but in the case of price, it should be within the range suggested by the system at that instant time.

5.2.5 Retailer

Retailers and customers are the buyers of the system. They can view the available products through our system or search for the desired products. They can ensure the quality of products by analyzing the rating of farmers although s/he may not directly get to know the farmers.

5.2.6 Data Collection

We collected data from various digital platforms like Hamropatro and online newspapers' historical data. The data associated with the price of items will be extracted through web scraping. Moreover, various web scraping tools like Scrapy 6 will be fruitful in extracting data from news portals.

5.2.7 Data Processing

We did some processing on the type of items, price of items and time it comes to the market for selling and buying. Various attributes of data are filtered. Additionally, the data are filtered for different types of items separately.

5.3 Data Collection

Once the data has been collected, it is saved as a CSV file for further processing and analysis. This file is then used for data cleaning, preprocessing, and modeling. It's important to keep the original data in its raw form, so that any preprocessing or cleaning steps can be traced back to the original data.

The CSV file contains data on different fields like max price, min price, date, items name, etc. However, this data is not yet ready to be used in a machine learning model, as it contains missing values and inconsistencies. Here are some of the preprocessing steps that might be taken to prepare this data for a machine learning model:

Encoding categorical variables: In the system, the "item" column contains name of the products. These variables could be labelled with specific value to make them usable in a machine learning model.

Scaling numerical variables: Similarly, the columns contain numerical and string variables. These variables are parsed to specific data types to ensure that they have same type and similar range of values.

After these preprocessing steps have been taken, the data would be ready to be used in a machine learning model.

	A	B	C	D	E	F	G
1	SN	Commodity	Date	Unit	Minimum	Maximum	Average
2	0	Tomato Big(Nepali)	2013-06-16	Kg	35.0	40.0	37.5
3	1	Tomato Small(Local)	2013-06-16	Kg	26.0	32.0	29.0
4	2	Potato Red	2013-06-16	Kg	20.0	21.0	20.5
5	3	Potato White	2013-06-16	Kg	15.0	16.0	15.5
6	4	Onion Dry (Indian)	2013-06-16	Kg	28.0	30.0	29.0
7	5	Carrot(Local)	2013-06-16	Kg	30.0	35.0	32.5
8	6	Cabbage(Local)	2013-06-16	Kg	6.0	10.0	8.0
9	7	Cauli Local	2013-06-16	Kg	30.0	35.0	32.5
10	8	Raddish Red	2013-06-16	Kg	35.0	40.0	37.5
11	9	Raddish White(Local)	2013-06-16	Kg	25.0	30.0	27.5
12	10	Brinjal Long	2013-06-16	Kg	16.0	18.0	17.0
13	11	Brinjal Round	2013-06-16	Kg	20.0	22.0	21.0
14	12	Cow pea(Long)	2013-06-16	Kg	20.0	25.0	22.5
15	13	Green Peas	2013-06-16	Kg	55.0	60.0	57.5
16	14	French Bean(Local)	2013-06-16	Kg	25.0	30.0	27.5
17	15	Soyabean Green	2013-06-16	Kg	60.0	70.0	65.0
18	16	Bitter Gourd	2013-06-16	Kg	14.0	16.0	15.0
19	17	Bottle Gourd	2013-06-16	Kg	15.0	20.0	17.5
20	18	Pointed Gourd(Local)	2013-06-16	Kg	30.0	35.0	32.5
21	19	Snake Gourd	2013-06-16	Kg	25.0	30.0	27.5
22	20	Smooth Gourd	2013-06-16	Kg	20.0	25.0	22.5
23	21	Sponge Gourd	2013-06-16	Kg	45.0	50.0	47.5
24	22	Pumpkin	2013-06-16	Kg	18.0	20.0	19.0

Figure 5.2: Sample of collected data

	A	B	C
1	0		
2	Tomato	Big(Nepali)	
3	Tomato	Small(Local)	
4	Potato	Red	
5	Potato	White	
6	Onion	Dry	(Indian)
7	Carrot(Local)		
8	Cabbage(Local)		
9	Cauli	Local	
10	Raddish	Red	
11	Raddish	White(Local)	
12	Brinjal	Long	
13	Brinjal	Round	
14	Cow	pea(Long)	
15	Green	Peas	
16	French	Bean(Local)	
17	Soyabean	Green	
18	Bitter	Gourd	
19	Bottle	Gourd	
20	Pointed	Gourd(Local)	
21	Snake	Gourd	
22	Smooth	Gourd	
23	Sponge	Gourd	
24	Pumpkin		
25	Squash(Long)		
26	Turnip		
27	Okara		
28	Christophine		
29	Brd	Leaf	Mustard
30	Spinach	Leaf	
31	Cress	Leaf	

Figure 5.3: Items listing and categorization

5.4 Expert System

Price prediction, implemented using machine learning algorithms, analyze historical data and identify patterns and trends in market conditions. The system can then use this analysis to predict prices for products, which can be used by Agro-Tech to optimize pricing strategies. To build an effective price prediction system, we collected data via web-scraping and store product information. This data is then used to train the machine learning algorithms to accurately predict prices for different products and at different times. The system is customized to take into account various factors that can affect pricing, such as time of production, product availability, and marketing status. By analyzing these factors, the system can adjust prices in real-time to optimize sales and maximize revenue. Finally, Agro-Tech use the price prediction system to test and refine different pricing strategies, and to continuously improve its pricing algorithms over time. By using data-driven insights to inform pricing decisions, the website can stay competitive and increase its bottom line.

Web Scraping : Web scraping is the process of extracting data from websites using automated software tools. It is a powerful technique for extracting data from websites. This process involves accessing the website's HTML code, parsing the data, and extracting the relevant information.

Beautiful Soup is a Python library that is commonly used for web scraping. It provides a powerful and flexible toolset for web scraping, and it is widely used in the data science and machine learning communities. It involves using tools and techniques to parse the HTML and CSS of web pages and extract the relevant data. Here's how it works:

- **Request the web page**: To begin web scraping with Beautiful Soup, you first need to request the web page that you want to scrape. This can be done using Python's requests library.
- **Parse the HTML**: Once you have the web page, you need to parse its HTML. Beautiful Soup is a library that can parse HTML and XML documents, and it can also handle poorly formed HTML.
- **Find the relevant data**: Once the HTML is parsed, we can use Beautiful Soup's searching and filtering capabilities to find the relevant data that you want to extract. This involves using various functions provided by Beautiful Soup.
- **Extract the data**: Once the relevant elements are found, we can extract the data that we want from them. This can involve getting the text inside an element, getting the value of an attribute, or even getting the URL of a link.
- **Store the data**: Finally, once we have extracted the data, we can store it in a format that is useful for our system. This involve saving the data to a CSV file, a database, or even using it directly in a machine learning model.

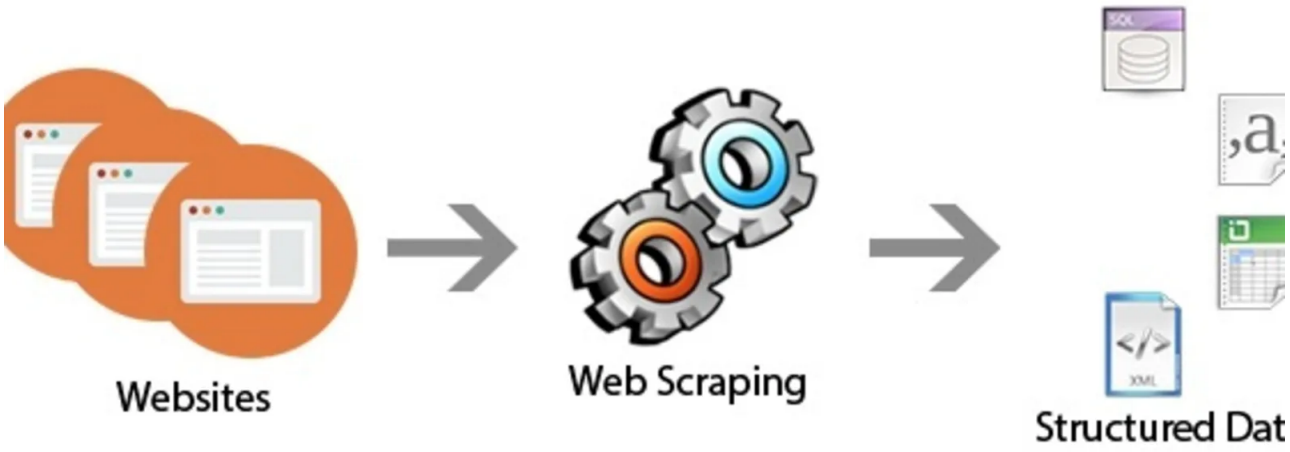

Figure 5.4: Web Scraping

For these analyses, Random Forest algorithm is used to predict price system. Here are the steps involved in using Random Forest algorithm as a price prediction system:
Data Collection: The first step is to collect historical pricing data, including information on product features, demand, and competitor pricing. The data is comprehensive and cover a wide range of products.

Data Preprocessing: The collected data is then preprocessed by handling missing values, outliers, and normalizing the data. This step is important to ensure that the data is accurate and reliable.

Feature Engineering: The next step is to select the most relevant features that affect the price of the product. Feature engineering includes factors such as product category, production time, seasonality, and pricing.

Model Training: The Random Forest algorithm is then trained on the preprocessed data using the selected features. The algorithm constructs multiple decision trees, each trained on a different subset of the data. This helps to reduce overfitting and increase the accuracy of the model.

Model Testing: The trained model is then tested on a separate dataset to evaluate its accuracy. The accuracy of the model can be measured using metrics such as mean absolute error, mean squared error or root mean squared error.

Model Deployment: Once the model has been trained and tested, it is then deployed as a price prediction system. The system can take inputs such as product features, demand, and competitor pricing and provide an accurate price prediction.

Regular Updates: The price prediction system is regularly updated with new data to ensure accurate predictions. The model can be retrained on the new data to improve its accuracy and reliability.

Random Forest algorithm is used as a price prediction system by collecting historical pricing data, pre-processing and selecting relevant features, training and testing the model, and deploying it as a prediction system.
In conclusion, price prediction as an expert system in Agro-tech is a powerful tool for optimizing pricing strategies and improving sales performance. By leveraging machine learning algorithms and analyzing historical data, Agro-tech can make more accurate predictions about future prices, which can lead to better pricing decisions and improved customer satisfaction.

6. System design

6.1 System Diagram

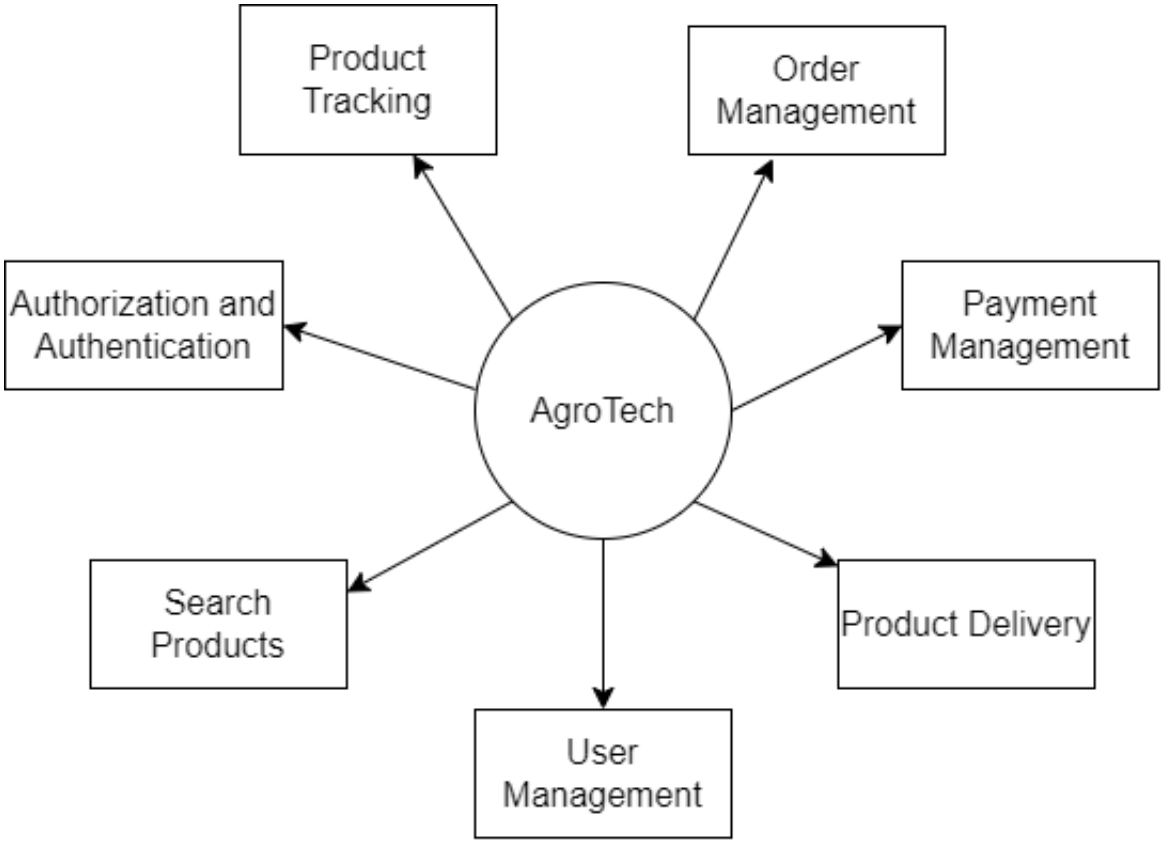

Figure 6.1: System Diagram

The project is divided into multiple subsystems, which provide individual features independent of others. All systems included in this project are product tracking, payment management, user management authentication and authorization order management search products and product delivery.

6.2 Use Case Diagram

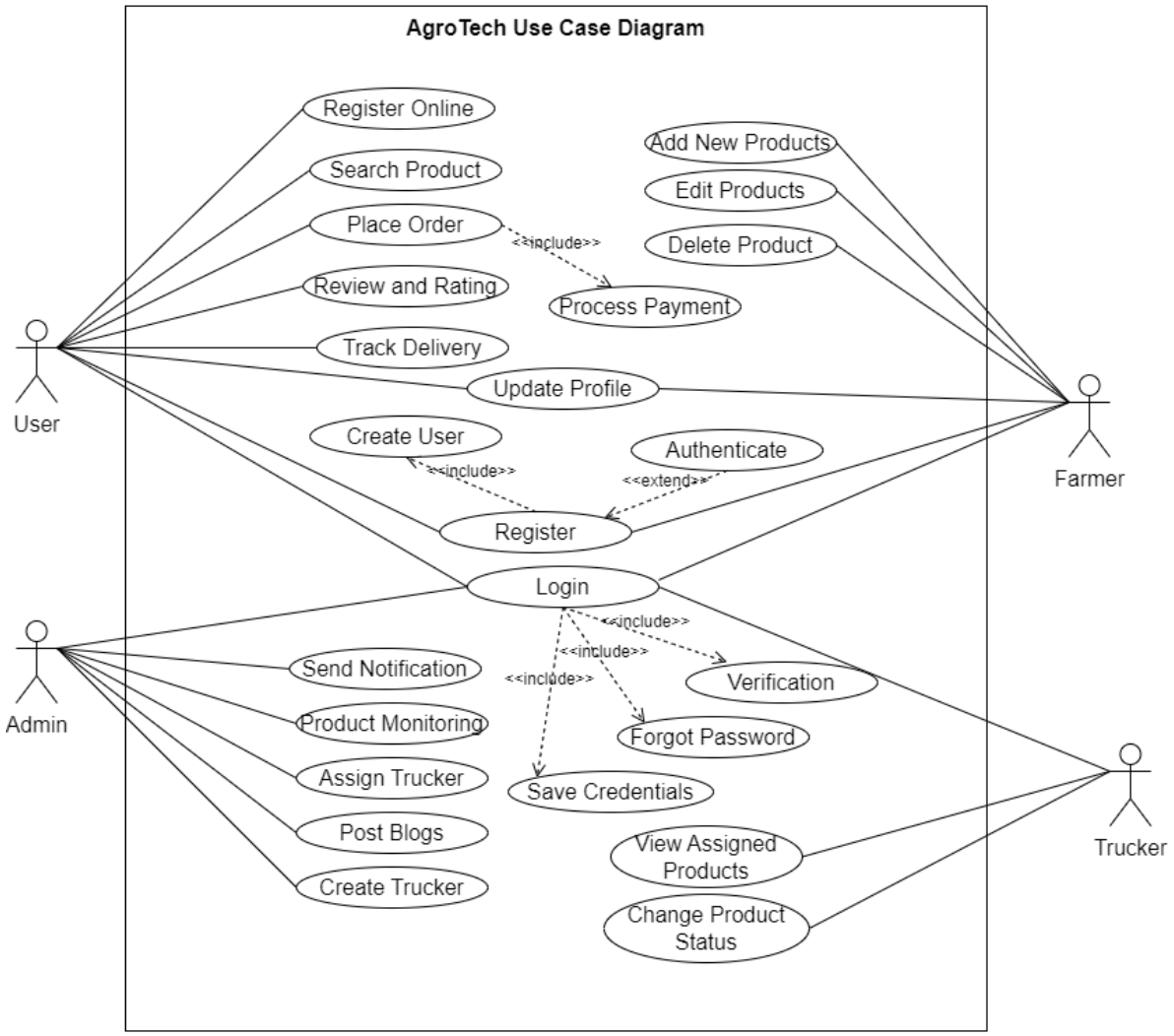

Figure 6.2: Use Case Diagram

6.3 Class Diagram

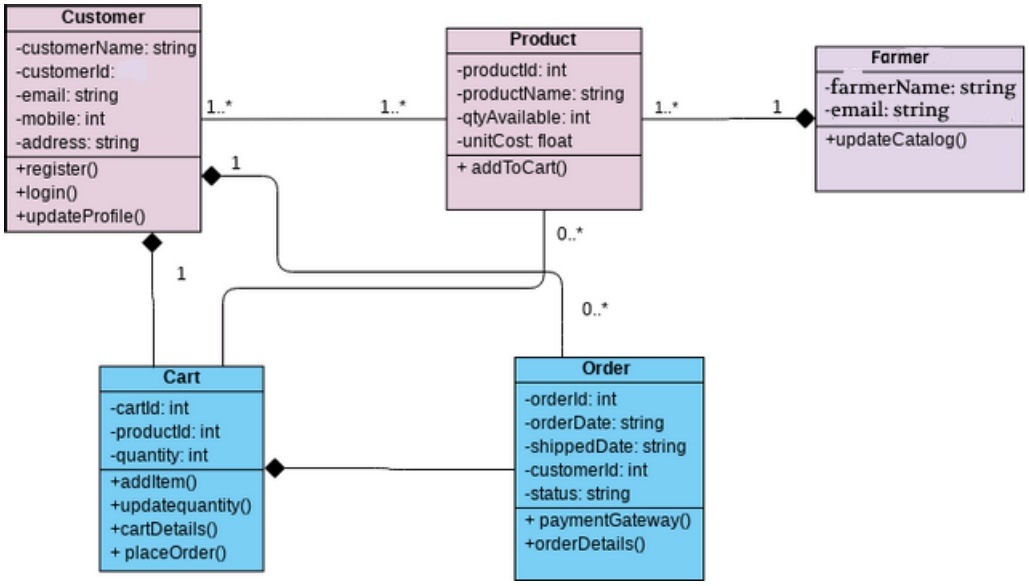

Figure 6.3: Class Diagram

6.4 Entity Relationship Diagram

The relationship between the different entities of Agro-Tech system is shown in the entity diagram below:

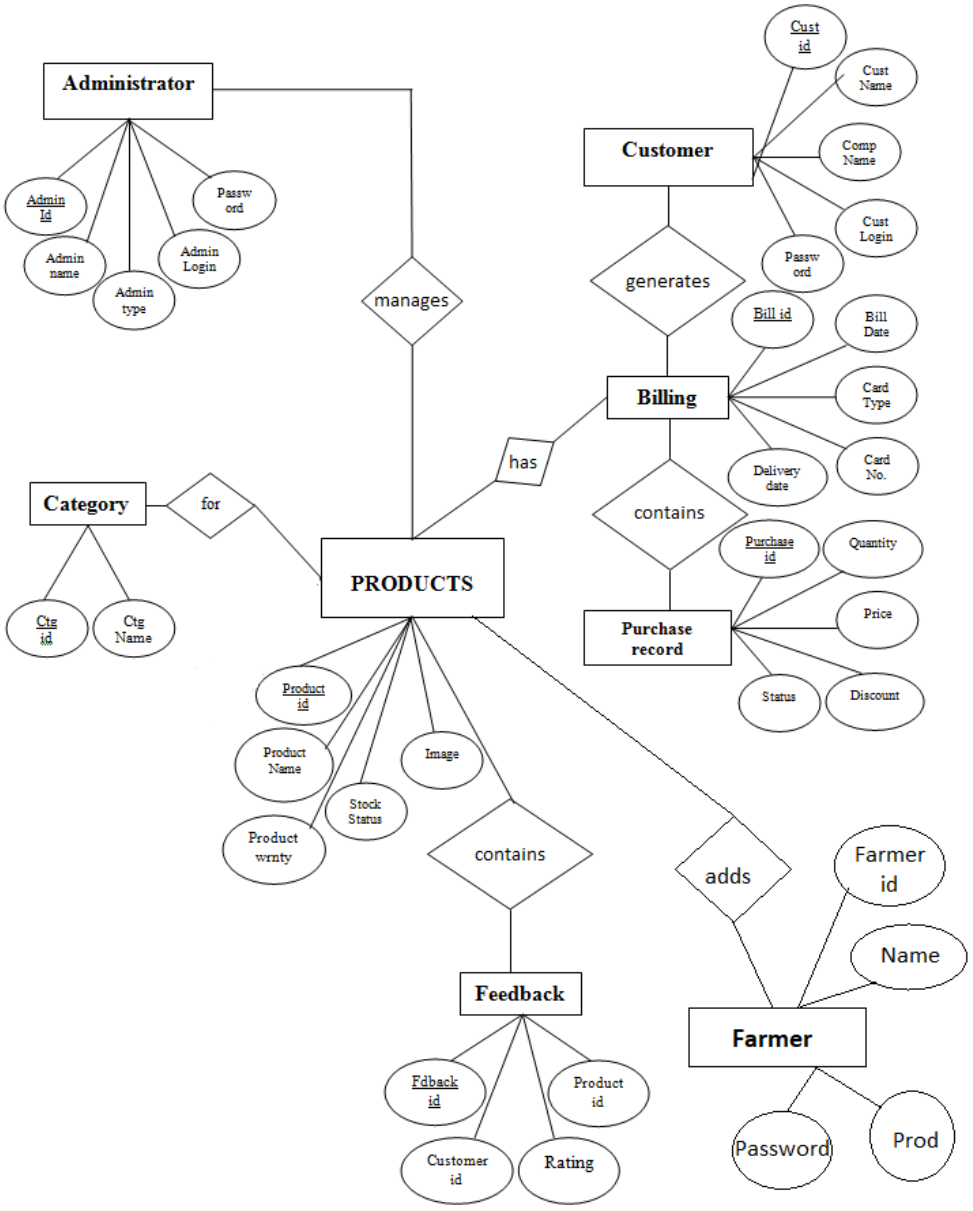

Figure 6.4: Entity Relationship Diagram

The ER diagram for Agro-tech demonstrates how the company uses database design to accommodate the data utilized in various service and management operations (ERD). The database relationships between the system's entities are made clear by this. By identifying entities, properties, and interactions, shows how a database is logically structured. This ER Diagram shows the connections between all System items graphically. The relationships are

defined as:

- **Adds:** The farmer can add products that can be readily displayed in the Products section.
- **Manages:** An admin manages all the tasks like managing and editing products.
- **Contains:** The products contain feedback with fields like rating and review which can be added by the buyer.
- **Generates:** The invoice/bill is generated after the products are bought and made payment.
- **Has:** The bill contains the details of the products being bought.
- **For:** Each product is assigned to a particular category when they're added by the farmer.

The entities that are the part of Agro-Tech system and coordinate with each other for a meaningful operation are defined as:

- **Administrator:** An Admin can view and manage all the Orders, Services, Invoices, and Payments of products in the system.
- **Farmer:** A farmer adds a product and its detail.
- **Category:** The products are displayed in their respective category.
- **Products:** The products are the entities that appeared in every entity and relationship. It is added by only the farmer and admin role.
- **Customer:** Customers or simply whole sellers are the entities that buy the added products.
- **Billing:** Billing is another important entity containing purchase records
- **Purchase Record:** Purchase record saves the status of bought products.
- **Feedback:** It is another important entity that is added by the customer after the products are delivered.

6.5 Sequence Diagram

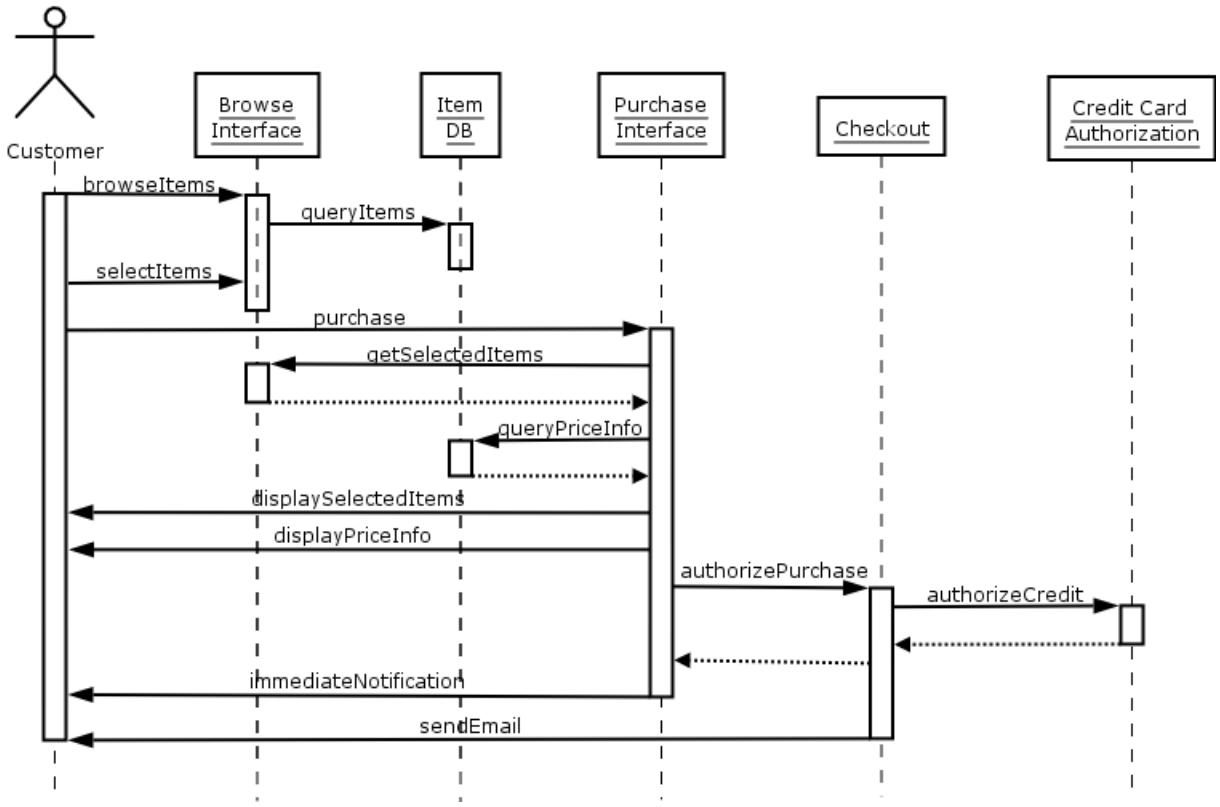

Figure 6.5: Sequence Diagram

The flow in our system begins with the farmer and/or whole seller. When the whole seller requests for a products or place his orders, the buying process is initiated. First of all, from the list of the available products/items that are open to sell are displayed in the site which was being uploaded by farmer. The farmer uploaded his available items with the required information fields where the price field can be entered within a certain range which was returned by our expert system. Admin assign truckers based on the location to bring/ deliver products to the respective whole seller from the respective farmer. The status of the products are tracked and is displayed for each roles who are concerned with that product.

7. Results & Discussion

Our project "Agro-Tech" has made significant progress towards creating a platform for efficient buying and selling of vegetables between farmers and retailers. Our platform aims to reduce post-harvest loss an d ensure food safety and freshness, while also promoting the use of local raw materials.

We have successfully developed the platform's user interface, which is user-friendly for both farmers and retailers. The interface allows farmers to list their available produce and for retailers to browse and purchase the vegetables. We have also implemented a payment system that enables secure and efficient transactions.

Through our partnerships with local farmers, we have been able to offer a range of high-quality vegetables to retailers. This has helped us attract more retailers to our platform, increasing the potential for sales and profits for farmers.

Our market research has helped us identify areas for improvement, including expanding the range of vegetables offered on the platform and increasing our outreach to more retailers and farmers. We are currently working to address these issues, and we remain committed to improving our platform's functionality and outreach.

Furthermore, our platform prioritizes food safety and freshness, utilizing cooling infrastructure and adhering to global logistical standards for the handling of fresh produce. This ensures that the vegetables reach retailers and consumers in the best possible condition.

8. Conclusions

In conclusion, our project "Agro-Tech" aims to create a platform for efficient buying and selling of vegetables between farmers and retailers. We have made significant progress towards this goal, including developing the platform's user interface, implementing a payment system, and establishing partnerships with local farmers.

Our project promotes the use of local raw materials, reduces post-harvest loss, and ensures food safety and freshness. We are committed to expanding our outreach, improving the range of vegetables offered, and enhancing our platform's functionality to benefit farmers, retailers, and the local community.

We believe that our project has the potential to positively impact Nepal's agro-based industry, contributing to economic development, reducing poverty and unemployment, and promoting equality. By utilizing cooling infrastructure and adhering to global logistical standards, we prioritize food safety and freshness, which is essential for both mental and physical well-being.

Moving forward, we will continue to work towards our goal of creating a platform that benefits all stakeholders. We are confident that our project will contribute to the growth of Nepal's agro-based industry, and we remain committed to delivering a high-quality platform that improves the quality of life for the local community.

9. Limitations and Future enhancement

Although our project "Agro-Tech" has made significant progress towards creating a platform for efficient buying and selling of vegetables between farmers and retailers, there are still some limitations that need to be addressed.One limitation is the limited range of vegetables currently offered on the platform. We plan to address this by expanding our partnerships with more farmers, increasing the variety of vegetables offered, and exploring the possibility of adding fruits and other agricultural products.Another limitation is the limited outreach to retailers and farmers in remote areas. We plan to address this by increasing our marketing efforts and partnering with local organizations and cooperatives to reach a wider audience.In terms of future enhancements, we plan to incorporate features such as real-time inventory tracking, predictive demand analysis, and automated order fulfillment to further improve the platform's functionality and efficiency.

Additionally, we can integrate blockchain technology to improve transparency and traceability, allowing consumers to easily verify the origin and quality of the produce they purchase.Overall, we remain committed to improving our platform and addressing its limitations to better serve farmers, retailers, and the local community. We believe that our project has the potential to positively impact Nepal's agro-based industry and contribute to economic development, and we look forward to continuing to work towards our goal of creating a more efficient and sustainable agricultural supply chain.

References

[1] Arumugam N, Fatimah MA, Chew EFC, Zainalabidin M (2010) Supply chain analysis of fresh fruits and vegetables. Agric Econ Czech 56(9):435–442

[2] Kaikini, Anjan S, Mithila. (2019). Challenges of Agro-based Industries: A Study in Sagar Taluk.

[3] Karnataka's new app to help farmers sell produce directly to retail chains. The Economic Times. (n.d.). Retrieved July 14, 2022, from https://economictimes.indiatimes.com/tech/technology/karnatakas-new-app-to-help-farmers-sell-produce-directly-to-retail-chains/articleshow/83393673.cms

[4] Hastuti EY (2007) The influence of agribusiness system applied to vegetable farmers income improvement in Boyolali in the regency. Master Thesis, Diponegoro University, Indonesia

[5] Revolutionizing the fresh produce supply chain. Ninjacart. (n.d.). Retrieved July 14, 2022, from https://ninjacart.in/

[6] Agritech: Digital farming is revolutionizing agriculture. (2021). PWC. Retrieved from https://www.pwc.com/gx/en/industries/agribusiness/publications/agritech.html

[7] How Agtech Is Revolutionizing Farming. (2021). Forbes. Retrieved from https://www.forbes.com/sites/bonniekfoster/2021/01/28/how-agtech-is-revolutionizing-farming/?sh=1540a11f69e2

[8] The Future of Agriculture: How IoT and AI Can Revolutionize Farming. (2021). IoT for All. Retrieved from https://www.iotforall.com/future-of-agriculture-how-iot-and-ai-can-revolutionize-farming/

[9] Leveraging IoT to Transform Agriculture: The AgTech Landscape. (2021). Medium. Retrieved from https://medium.com/@VinsysUS/leveraging-iot-to-transform-agriculture-the-agtech-landscape-b79e07c4ba4b

[10] From precision agriculture to precision food: IoT and AI in the food supply chain. (2019). Deloitte. Retrieved from

https://www2.deloitte.com/us/en/insights/industry/retail-distribution/iot-and-ai-in-the-food-supply-chain.html

[11] AI and IoT: The Future of Agriculture. (2021). IoT for All. Retrieved from https://www.iotforall.com/ai-and-iot-the-future-of-agriculture/

[12] Smart farming: The future of agriculture. (2021). IBM. Retrieved from https://www.ibm.com/blogs/internet-of-things/smart-farming-the-future-of-agriculture/

Appendix

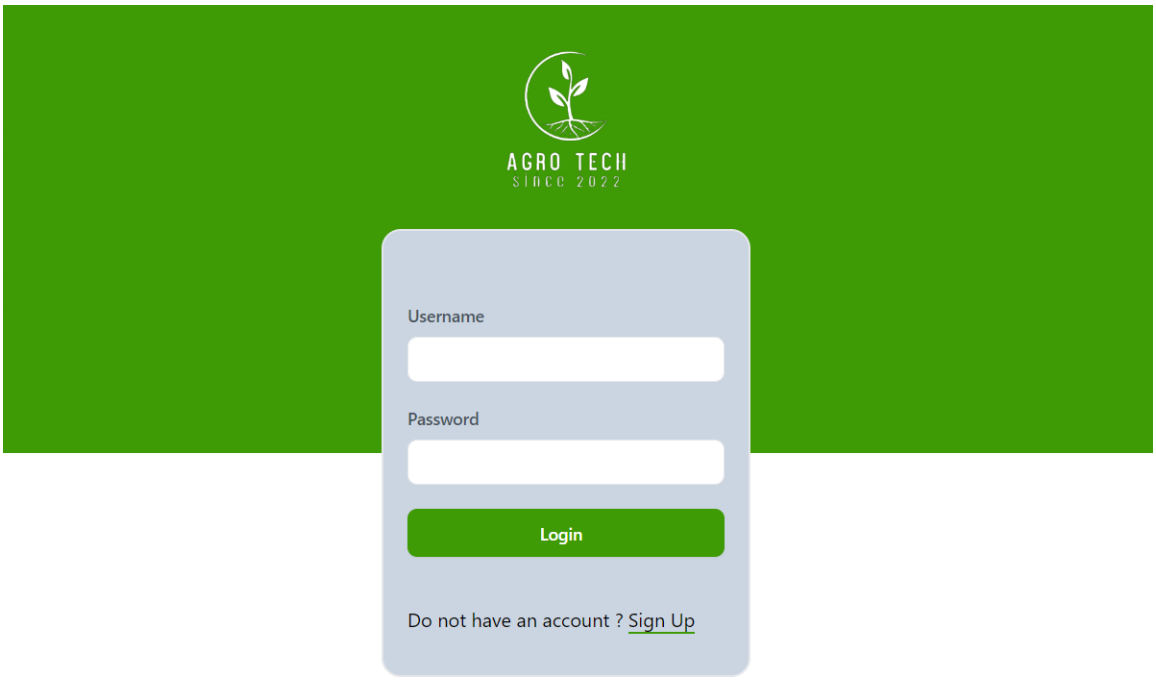

Figure 1: Login Page

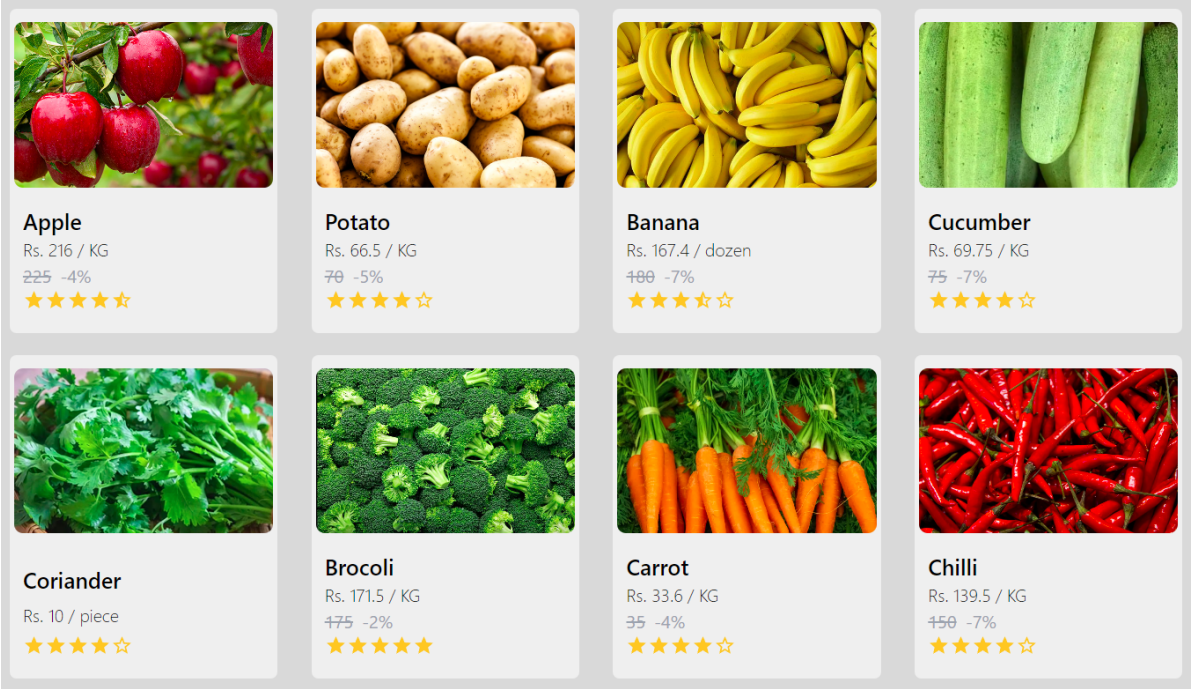

Figure 2: Products

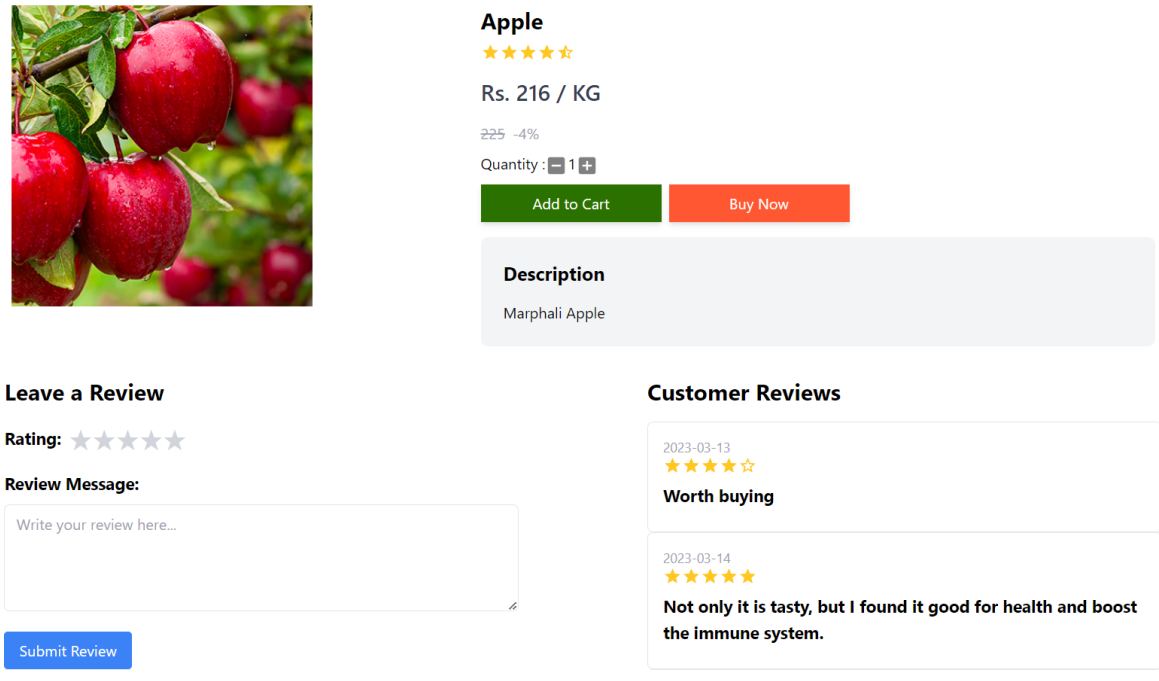

Figure 3: Product Detail

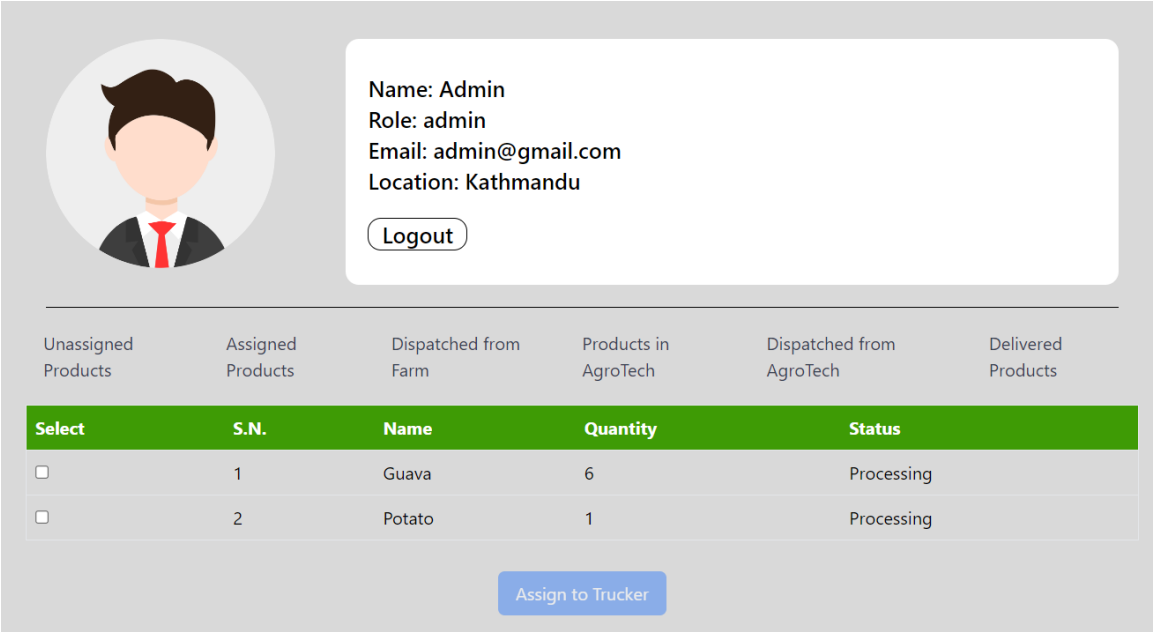

Figure 4: Admin Profile

Select	S.N.	Name	Price	Quantity	Status
☐	1	Banana	180	5	Product in Agrotech
☐	2	Potato	70	1	Trucker Assigned
☐	3	Guava	180	6	Processing
☐	4	Chilli	150	5	Product delivered
☐	5	Coriander	10	3	Product delivered
☐	6	Potato	70	1	Processing

Figure 5: Buy List

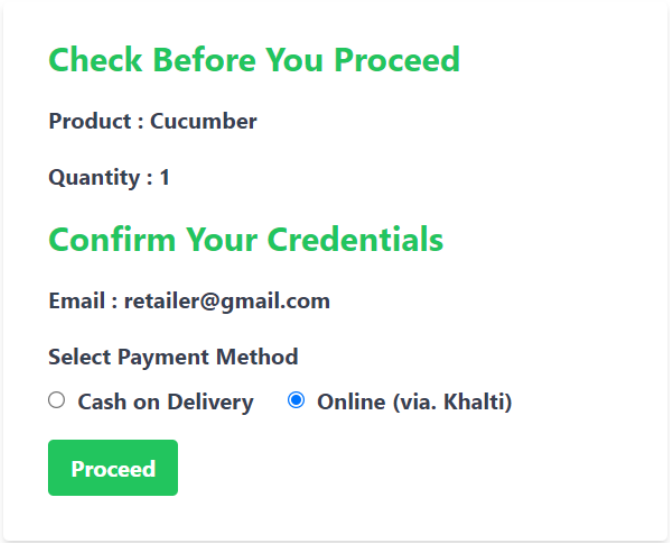

Figure 6: Payment Mode Selection

Figure 7: Blogs