**University of Mumbai**

**Kisaan Bandhu - A Farmer Grain**

**Assistant Application**

Submitted in partial fulfillment of requirements

For the degree of

**Bachelors in Technology**

by

**Meet Bhanushali - 1711005**

**Govinda Patel - 1711038**

**Parth Sheth - 1711055**

**Shailesh Upadhyay - 1711061**

Guide

**Prof. Rajni Pamnani**



**Department of Computer Engineering**

**K. J. Somaiya College of Engineering, Mumbai-77**

**(Autonomous College Affiliated to University of Mumbai)**

**Batch 2017 -2021**

**Certificate**

This is to certify that the dissertation report entitled **Kisaan Bandhu** - **A Farmer Grain Assistant Application** submitted by

1. Meet Bhanushali
2. Govinda Patel
3. Parth Sheth
4. Shailesh Upadhyay

at the end of semester VIII of LY B. Tech under Guidance of **Prof Rajni Pamnani,** is a bona fide record for partial fulfillment of requirements for the degree of Bachelors in Technology in Computer Engineering of University of Mumbai

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Guide Head of the Department

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Principal

Date:

Place: Mumbai-77

**Certificate of Approval of Examiners**

We certify that this dissertation report entitled **Kisaan Bandhu** - **A Farmer Grain Assistant Application** is bona fide record of project work done by

1. Meet Bhanushali
2. Govinda Patel
3. Parth Sheth
4. Shailesh Upadhyay

during semester VII.

This project work is submitted at the end of semester VII in partial fulfillment of requirements for the degree of Bachelors in Technology in Computer Engineering of University of Mumbai.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Internal Examiner

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

External Examiner

Date:

Place: Mumbai-77

**DECLARATION**

We declare that this written report submission represents the work done based on our and / or others’ ideas with adequately cited and referenced the original source. We also declare that we have adhered to all principles of intellectual property, academic honesty and integrity as we have not misinterpreted or fabricated or falsified any idea/data/fact/source/original work/ matter in my submission.

We understand that any violation of the above will be cause for disciplinary action by the college and may evoke the penal action from the sources which have not been properly cited or from whom proper permission is not sought.

|  |  |
| --- | --- |
| **Signature of the Student**  **\_\_\_1711005 - Meet Bhanushali\_\_\_\_\_\_\_**  **Roll No.** | **Signature of the Student**  **\_\_1711055 -Parth Sheth\_\_**  **Roll No.** |
| **Signature of the Student**  **\_\_\_1711038 - Govinda patel\_\_\_\_\_\_\_**  **Roll No.** | **Signature of the Student**  **\_\_\_\_1711061 -Shailesh Upadhyay\_\_\_\_**  **Roll No.** |

**Date:**

**Place: Mumbai-77**

## 

## Dedicated to

***My family and friends***

**Abstract**

As we know that India’s GDP has a major share of the agricultural tasks and activities (around 14-16%), it becomes very important to improvise in the agricultural sector and maintain its sustainability in the fast growing market. But due to lack of advancements in this sector there is a transition seen in the farmers count in which many of them have opted to stop their farming work and moved on to some other sector. Hence we have chosen this project as it is a step towards the overall agricultural sector development by helping the farmers gains what they deserve.

Currently the scenario is such that the farmers sell their products to the dealers which then sell these products to the final buyers or some more small retailers and dealers. Due to this extra addition of the dealers in the cycle the final price of the product sold to the common man is higher than the expectation of the public and actual price which the product has. So our application will enable the farmer to directly reach the buyers hence cutting out the extra cost due the increase in the cycle due to the profit margins of the middle men. Also the entire profit made by selling the product is received by the farmer.

This is an application which is mainly used by farmers after crop season who need some source of platform to directly reach the buyers by eliminating the dealers in between the cycle which take the major share of the profit and finally the share which the farmer has for his part is not enough.

***Key words*:** GDP , Market , Production chain cycle

**Table of Contents**

[1. Introduction 14](#_Toc64925206)

[1.1. Background 14](#_Toc64925207)

[1.2. Problem Definition 14](#_Toc64925208)

[1.3. Motivation 14](#_Toc64925209)

[1.4. Scope of Project 15](#_Toc64925210)

[1.5. Stakeholders 15](#_Toc64925211)

[2. Literature Survey 16](#_Toc64925212)

[2.1. Literature Survey 16](#_Toc64925213)

[3. Overall Description 19](#_Toc64925214)

[3.1. Product Perspective 19](#_Toc64925215)

[3.2. Product Functions 20](#_Toc64925216)

[3.3. Operating Environment 20](#_Toc64925217)

[3.4. Design and Implementation Constraints 20](#_Toc64925218)

[3.5. Assumptions and Dependencies 21](#_Toc64925219)

[4. Software Requirements Document 22](#_Toc64925220)

[4.1. User Interfaces 22](#_Toc64925221)

[4.2. Hardware Interfaces 24](#_Toc64925222)

[4.3. Software Interfaces 24](#_Toc64925223)

[4.4. Communications Interfaces 24](#_Toc64925224)

[4.5. System Features 25](#_Toc64925225)

[4.6. Other Nonfunctional Requirements 26](#_Toc64925226)

[4.6.1. Performance Requirements 26](#_Toc64925227)

[4.6.2. Safety Requirements 26](#_Toc64925228)

[4.6.3. Security Requirements 26](#_Toc64925229)

[4.6.4. Software Quality Attributes 26](#_Toc64925230)

[1. Reliability 26](#_Toc64925231)

[2. Maintainability 27](#_Toc64925232)

[3. Availability 27](#_Toc64925233)

[4. Portability 27](#_Toc64925234)

[5. Project Design 28](#_Toc64925235)

[5.1. Proposed System Block Diagram. 28](#_Toc64925236)

[5.2. Proposed Methodology 28](#_Toc64925237)

[5.2.1. Sell and Buy of the product 28](#_Toc64925238)

[5.2.2. Contract based sales 29](#_Toc64925239)

[5.3. UML Diagrams 29](#_Toc64925240)

[5.3.1. Class Diagram 29](#_Toc64925241)

[5.3.2. Use Case Diagrams 30](#_Toc64925242)

[5.4. Identified Risks 31](#_Toc64925243)

[5.4.1. Risk : 31](#_Toc64925244)

[5.4.2. Issues : 31](#_Toc64925245)

[5.4.3. Challenges : 31](#_Toc64925246)

[5.4.4. Constraints : 32](#_Toc64925247)

[6. Implementation Details 33](#_Toc64925248)

[6.1. Algorithms and Techniques Used 33](#_Toc64925249)

[6.1.1. Convolution Neural Network (CNN model) 33](#_Toc64925250)

[6.1.2. Routing Optimization Algorithm – Travelling Salesman Problem 33](#_Toc64925251)

[6.1.3. Web Scrapping 34](#_Toc64925252)

[6.1.4. Clustering Algorithm 35](#_Toc64925253)

[6.1.5. HTTP Protocol 35](#_Toc64925254)

[6.2. Tools Used 36](#_Toc64925255)

[6.2.1. Flutter 36](#_Toc64925256)

[6.2.2. Python 36](#_Toc64925257)

[6.2.3. API 36](#_Toc64925258)

[6.2.4. Node.js 36](#_Toc64925259)

[6.2.5. MongoDB 37](#_Toc64925260)

[6.3. Google Maps(Geo-Locator) 37](#_Toc64925261)

[6.4. Operating Environment 38](#_Toc64925262)

[6.4.1. Operating System : Windows/MAC/Linux and Android 38](#_Toc64925263)

[6.4.2. Database : MongoDB(NoSQL) 38](#_Toc64925264)

[6.4.3. Platform : Flutter, Node.js, Python 38](#_Toc64925265)

[6.4.4. Software : postman, emulator, git, VS Code, Android Studio 38](#_Toc64925266)

[6.5. Hardware Requirements : 38](#_Toc64925267)

[6.6. Implementation Steps and modules 38](#_Toc64925268)

[6.7. Some of the temporary implementation screenshots until November end. 39](#_Toc64925269)

[7. Conclusion 45](#_Toc64925270)

[7.1. Conclusion 45](#_Toc64925271)

[7.2. Future Scope 46](#_Toc64925272)

[8. Bibliography 46](#_Toc64925273)

# 

# Introduction

This chapter sheds light on the introduction of the project. The brief idea about the project, the techniques and tools required to carry out the project. The scope, and the general outlook of the project is discussed here.

## Background

**Kisaan Bandhu** will enable the farmer to directly reach the buyers such as bulk buyers and supermarkets hence cutting out the extra cost due the increase in the cycle due to the profit margins of the Dealers. Also the entire profit made by selling the product is received by the farmer. This application will be mainly used by farmers after crop season who need some source of platform to directly reach the buyers by eliminating this system will also enable the farmers to directly receive some orders by large industries which will be contract based deals which will be managed by our application.

## Problem Definition

Kisaan Bandhu is an application to empower the farmers by helping them reach the buyer through our application. The application will have a major role of connecting the end users via the application which will basically work as connector application for buying and selling the products which the farmers will directly put over the application for sale. Also we will implement a contract based feature wherein the farmers and the client company can have a production contract between themselves for which this application will be a mediator.

## Motivation

As we know that India’s GDP has a major share of the agricultural tasks and activities (around 14-16%), it becomes very important to improvise in the agricultural sector and maintain its sustainability in the fast growing market. But due to lack of advancements in this sector there is a transition seen in the farmers count in which many of them have opted to stop their farming work and move on to some other sector. Hence we have chosen this project as it is a step towards the overall agricultural sector development by helping the farmers to gain what they deserve. Kisaan Bandhu will be mainly focused to provide a platform for the farmers to sell their products and get the maximum share of their produce.

**Kisaan Bandhu** will enable the farmer to directly reach the buyers such as bulk buyers and supermarkets hence cutting out the extra cost due the increase in the cycle due to the profit margins of the Dealers. Also the entire profit made by selling the product is received by the farmer. This application will be mainly used by farmers after crop season who need immediate sell of crops and transport to storages this system will also enable the buyers to directly place some bids on all agricultural products. Contract based deals which will be managed by our application.

## Scope of Project

1. First of all what we will try to achieve is to remove the unnecessary cycle in between the demand supply chain. We will try to create such a system where there will be direct buyers such as supermarts or a bulk buyer where we will directly supply the production from the farmers. Our cycle will try include maximum areas of production as we will also include transport services.
2. The second task which will try to implement is to create a contract based production which will be provided by the companies. In this type what we are trying to achieve is that we will take certain contracts from the companies If the farmer decides to accept the contract then the company will provide the materials required for proper production. The company will also provide the details of the materials they will provide to the farmers for the proper production of the raw material.

## Stakeholders

* Decision-makers: People with a higher authority, who control the flow of the project, in

terms of functionality, requirements. In this case, the Principal, Head of Department and

Project Mentor and team members.

* Influencers: People who may influence the decision of the project, only by a certain

margin. In this case, it would be the team members.

* End users: The final user of the software, who will use the project, the design has to be

done accordingly. In this case a disabled patient.

# Literature Survey

This chapter discusses about the various papers, articles referenced in preparation for undertaking this project. These items serve as a benchmark to enable this project to be undertaken.

## Literature Survey

The exploitation of the farmer and agricultural sector has been round the corner since the Pre - Independence times. **THE AGRICULTURAL PRODUCE (GRADING AND MARKING) ACT, 1937** marked the initial exploitation phase of the agricultural sector by the British rulers. This act Allowed the central Government of the foregoing ruling governance to –

1. fixing grade designations to indicate the quality of any scheduled article.
2. defining the quality indicated by every grade designation.
3. specifying grade designation marks to represent particular grade designations.
4. authorising a person or a body of persons to mark articles with a grade designation mark.
5. Any officer of the Central Government or a State Government can enter any premises at any reasonable time and make necessary inspection of the agricultural produce.
6. An officer authorised under sub-section (*1*) of Section 3A may seize and detain any agricultural produce. Etc..

All of this provisions made in the ACT were exploiting the agricultural sector of India. This continued on and on even after Independence and still the agricultural sector is the most backward economical sector of the country.

This has led to the increase in the Mortality rate of the Farmers. According to a survey by the **National Crime Records Bureau,** between **1995 and 2014** more than **3,00,000** farmers have committed suicides in India. This is roughly equivalent to a staggering figure of one farmer suicide every 30 minutes. Only between **11772 suicides in** **2013 and 12360 suicides in 2014** the percentage increase in the number of suicides is **5%.** The main reason for the suicides is the overwhelming debts and the decreasing income due to exploitation. A majority of cotton producers of Andhra Pradesh committed suicides with a pending debt of around ₹1,00,000.

But among all these ill treatment before and after Independence, a ray of hope for the farmers and the agricultural sector was the development of **AGRICULTURAL PRODUCE MARKETING COMMITTEE (APMC)** .

One of the major causes of low income of the Indian farmers is the difficulty in marketing their crops. Due to the small size and scattered nature of agricultural holdings, the productivity per acre is low.So he has to rely on a number of middlemen (intermediaries) for the disposal of “his crops at cheap prices.

Marketing in agricultural produce is considered as an integral part of agriculture as agriculturist is made more eager to invest in this sector. It helps in reducing the charges for marketing service by eliminating the chain of middlemen and to have the minimum difference between the buying and the selling prices. Marketing becomes an important instrument in improving the income of the individual producers of all categories, apart from meeting the requirements of the customers.

The main directions in which marketing was modernized were:

* Institutionalizing of agricultural marketing by facilitating the formation of

Cooperative marketing societies

* Regulation of markets for various agricultural products designed to minimize or eliminate unfair trade practices
* Direct involvement of the State in the marketing of certain agricultural products.

The major functions of the APMC are: grant, renew, refuse, suspend or cancel license;

provide the necessary facilities; regulate and supervise the auctions; maintain and manage the

markets; regulate the sales, promote and organize grading and standardization of the

agricultural produce and ware housing facilities in the market area.

Recently in June,2020, a major step was taken in this direction by introducing the

**Farmers (Empowerment and Protection) Agreement on Price Assurance and Farm Services Act, 2020.**

This act empowered the farmers as it:

* Recognised farmers as an individual engaged in the production of farming produce by self or by hired labour or otherwise, and includes the Farmer Producer Organisation(a group of farmers registered under law).
* Allowed farmers to sign legal agreements with sponsors prior to the production of any farming produce of a predetermined quality.
* Made it legal and compulsion for the sponsors to bear the risk of output and any farming needs and pay the farmer for his work regardless of any loss or profit which the sponsor bears.

Another act in the same time period in June,2020 named as

**Farmers’ Produce Trade and Commerce (Promotion and Facilitation) Act, 2020**

also boasted the agricultural sector by the following points:

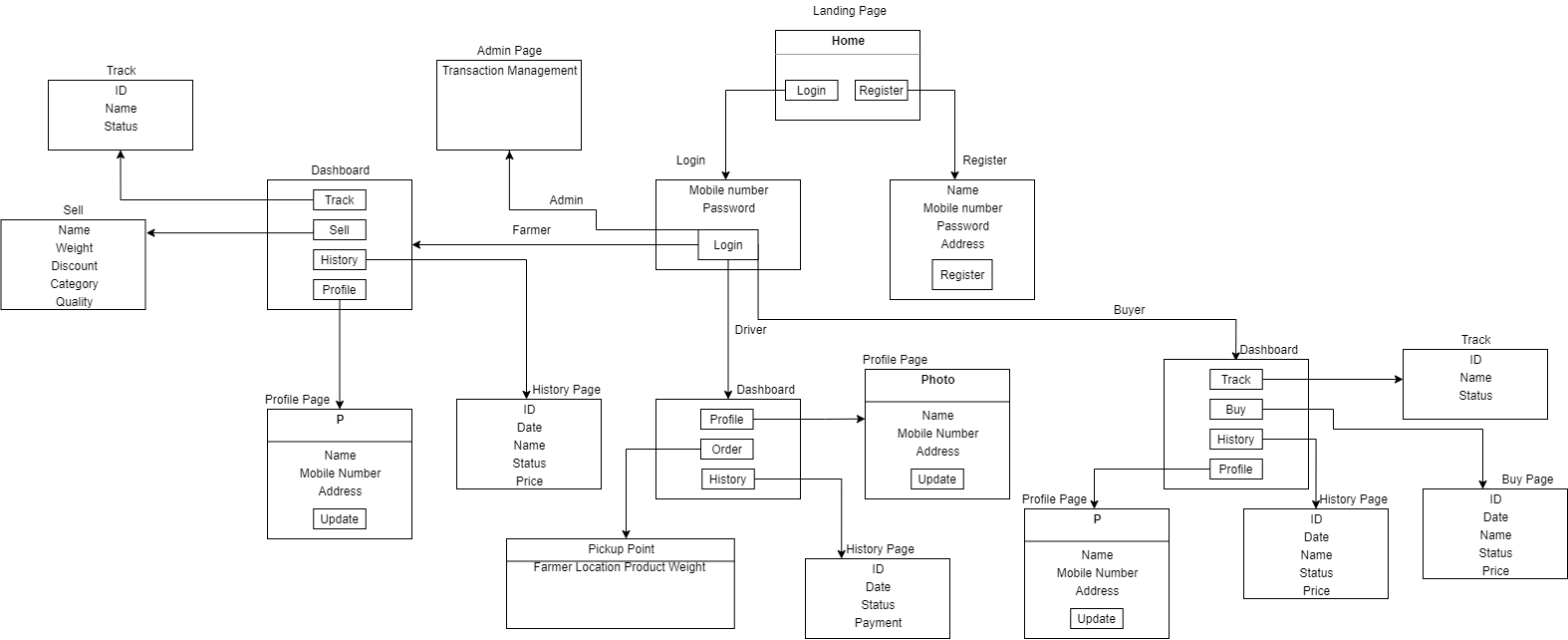
* It allowed farmers to have electronic trading and transaction platform where online buying and selling of farmer’s produce is allowed.
  1. **Some Research Papers Referred**
  + THE AGRICULTURAL PRODUCE (GRADING AND MARKING) ACT, 19
  + FARMER SUICIDES:TRANSITIONAL COSTS FOR INDIAN SOCIETY , AS A RESULT OF POLICY CHANGES OF 1990.
  + THE FARMERS’ PRODUCE TRADE AND COMMERCE (PROMOTION AND FACILITATION) ORDINANCE, 2020 - NO. 10 OF 2020
  + THE FARMERS’ PRODUCE TRADE AND COMMERCE (PROMOTION AND FACILITATION) ACT, 2020 - NO. 21 OF 2020
  + AGRICULTURALMARKETING IN INDIA ROLE OF AGRICULTURAL PRODUCEMARKETING COMMITTEE (APMC)
  + THE FARMERS (EMPOWERMENT AND PROTECTION) AGREEMENT ON PRICE ASSURANCE AND FARM SERVICES ACT, 2020 - NO. 20 OF 2020
  + FARMER SUICIDES IN INDIA TRENDS ACROSS MAJOR STATES, 1995–2011

# Overall Description

This chapter discusses about the about the overall description of the project which also includes diagrams such as the workflow diagram and some product functionalities.

## Product Perspective

As we know that India’s GDP has a major share of the agricultural tasks and activities (around 14-16%), it becomes very important to improvise in the agricultural sector and maintain its sustainability in the fast growing market. But due to lack of advancements in this sector there is a transition seen in the farmers count in which many of them have opted to stop their farming work and move on to some other sector. Hence we have chosen this project as it is a step towards the overall agricultural sector development by helping the farmers to gain what they deserve. Kisaan Bandhu will be mainly focussed to provide a platform for the farmers to sell their products and get the maximum share of their produce.



## Product Functions

**•User**: Farmer

**•Functions**:Farmers are one of the end user who can post the products over the application.

* Sell products
* Track history
* Track current orders
* Profile

**•User** : Buyer

•**Functions** :Buyer is one of the end user who can buy the products over the application.

* Sell products
* Track history
* Track current orders
* Profile

**•User** : Driver (Transport service)

•**Functions** :

* Track history
* Track current orders
* Profile

## Operating Environment

Operating Environment for Kisaan Bandhu is listed as below :

* Operating System : Windows/MAC/Linux and Android
* Database : MongoDB(NoSQL)
* Platform : Flutter, Nodejs, Python
* Software : postman, emulator, git, VSCode, Android Studio

Hardware Requirements :

* Processor : Pentium 4 and higher
* RAM : 1GB +
* Storage : 1GB +

## Design and Implementation Constraints

* The most important factors determining whether customers return to an application are ease of use and the presence of user-friendly features. Usability testing is important for finding problems and improvements in an application. Methods for evaluating usability include heuristic evaluation, cognitive walkthrough, and user testing. Each technique has its own characteristics and emphasizes different aspects of the user experience.
* **Pages details:**
* Home/Landing Page
* Login Page
* Registration Page
* Profile Page
* Individual Dashboard Page
* Selling/Buying Product Page
* History Page

## Assumptions and Dependencies

* Google Maps (Internet connectivity)
* Working API
* Location Services Enabled device

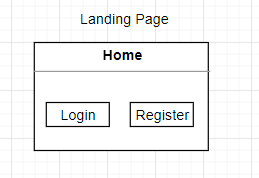
# Software Requirements Document

In this chapter, the different requirements of the system are discussed. It contains sections such as system functions, Interface requirements etc.

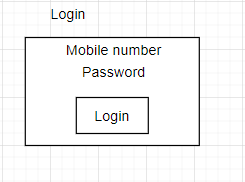
**The link for the SRS Document is -** [**SRS**](https://drive.google.com/file/d/13F3_I7X0WRGk_PJE3Bxmf7xQrKy69cWY/view?usp=sharing)

## User Interfaces

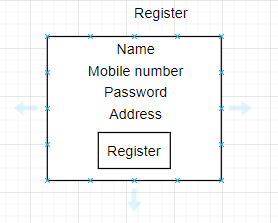
* Home/Landing Page



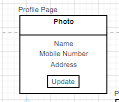
* Login Page



* Registration Page



* Profile Page

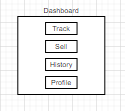


* Individual Dashboard Page

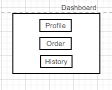
Buyer Dashboard



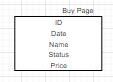
Farmer Dashboard

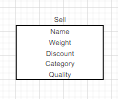


Driver Dashboard

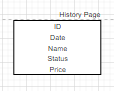


* Selling/Buying Product Page





* History Page



## Hardware Interfaces

**Server Side:**

**The API will be hosted on a web hosting server.**

**Client side:**

1. **Android mobile**
2. **Proper Internet Connectivity**
3. **GPS (Location) enabled service**

## Software Interfaces

**Server Side:**

1. Web server will accept all the requests from the client and respond accordingly.Database will be hosted centrally using a single storage system.

**Client Side:**

Android OS having version 5 and above.

## Communications Interfaces

**The HTTP or HTTPS protocol(s) will be used to facilitate communication between the client and server.**

## System Features

This section provides a requirement overview of the system.

* + 1. SF1

**Title -** User Registration

**Description -** User can register in the application through mobile where the user must provide the required information.

**Dependency** - None

**Result** - User register successfully

* + 1. SF2

**Title -** User Login

**Description -** User can Login the application through mobile where the user must provide the required information to successfully login.

**Dependency** - SF1.

**Result** - User Login successfully

* + 1. SF3

**Title -** Sell product ( by Farmer)

**Description -** Farmers can sell the product via the application by listing their products over the application for sale.

**Dependency** - SF2

**Result** - Successfully listed the product for sale

* + 1. SF4

**Title -** User Profile

**Description -** User can view and edit his/her profile in the application **Dependency** - SF2

**Result** - User profile changed/viewed successfully

* + 1. SF5

**Title -** Track orders

**Description -** User can track the current orders which are placed and can get all the details regarding the status of the order.

**Dependency** - SF2 & SF3

**Result** - User will be able to track their orders successfully

* + 1. SF6

**Title -** Track History

**Description -** Users can track their previous orders which are sold or bought and already completed and can get the status of those particular orders.

**Dependency** - SF2 & SF3

**Result** - User will be able to track their history successfully

* + 1. SF7

**Title -** Buy Product ( Buyers)

**Description -** Users can buy the product via the application which are listed in the application for sale.

**Dependency** - SF2

**Result** - User will be able to buy products successfully

* + 1. SF8

**Title -** track pickup orders (Drivers)

**Description -** The transport service driver will be able to track the pickup points from where the farmers will sell their products.

**Dependency** - SF2 & SF3

**Result** - Driver will able to track the pickup route successfully

## Other Nonfunctional Requirements

### Performance Requirements

* This system is for a smaller area initially and hence it should handle at least 200 users.
* The downtime for the application must be less than 5%.

### Safety Requirements

* If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure.

### Security Requirements

* System will use secured database
* Avoiding misuse of personal data of users such as GPS location.
* System will have different types of users and every user has access constraints.

### Software Quality Attributes

#### Reliability

The system provides storage of all databases on redundant computers with automatic switchover. The reliability of the overall program depends on the reliability of the separate components. The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes.

#### Maintainability

NoSQL is used for maintaining the database and the server takes care of the API’s. In case of a failure, a re-initialization of the program is recommended as soon as possible.

#### Availability

The system should be available at all times, meaning the user can access the application, only restricted by the down time of the server on which the system runs.

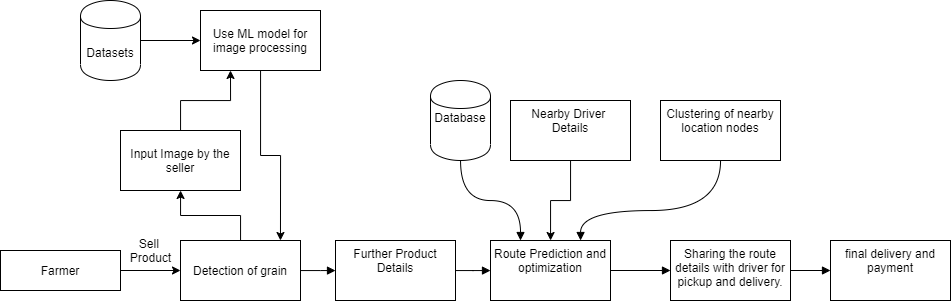
#### Portability

The technologies used are already system independent. The application is Android based. So The end-user part is fully portable and any system using any android version 5 and above should be able to use the features of the system.

# Project Design

In this chapter, different project design topics such as UML diagrams and block diagrams are included.

# Proposed System Block Diagram.



**The link for above diagram is -** [**Diagram**](https://drive.google.com/file/d/10VYaogZik0hg6GIQbZC0FACHftAq6gvX/view?usp=sharing)

## Proposed Methodology

### Sell and Buy of the product

* + - 1. **Activity 1 - Farmer posts the product on the application**
    1. **Authentication of the farmer via the Login into the application.**
    2. **Posting of the product on the application**
    3. **Filling the details of the product.**
    4. **Final submission of the product for sale on the application**
    5. **Tracking of the application for the product in the sale cycle.**
    6. **Final status of the product**
  1. **Activity 2 - Transportation of the product by the Driver**

1. **Checking the availability of Driver for pickup**
2. **Assigning list of pickup locations for driver.**

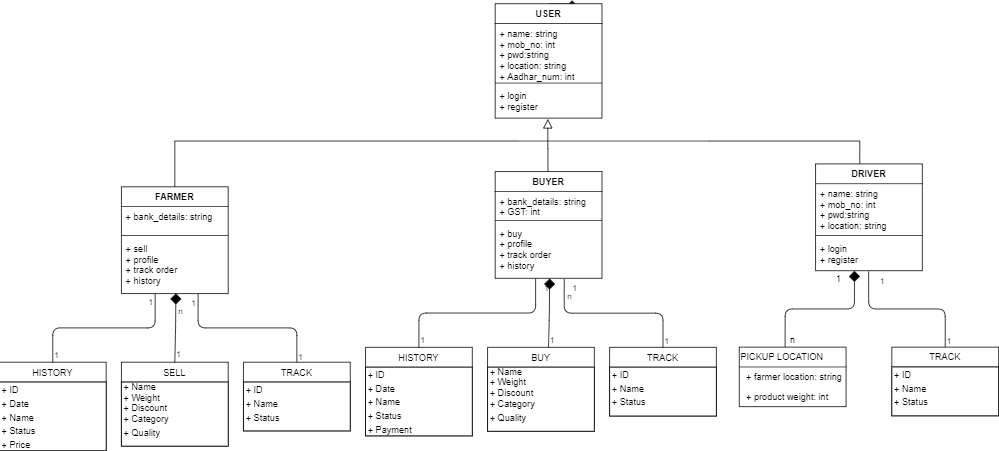
* **Deciding group of pickup location using clustering analysis based on pin code Using Modified Travelling Salesman Problem to predict the optimized route for pickup and distribution.**
  1. **Activity 3 - Buying the product** 
     1. **Buyer places the bid for the product on the application**
     2. **Using greedy approach for allotment amount of product**
     3. **Accepting the payment for the product using the payment gateway.**

### Contract based sales

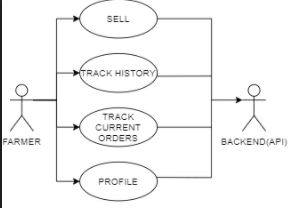
* 1. **Activity 1 - Client** 
     1. **Details for the product to be grown**
     2. **Quantity of the required product to be grown**
     3. **Notifying farmers for the availability of a contract**
  2. **Activity 2 - Assigning of the contract to the farmers by conditional mapping**

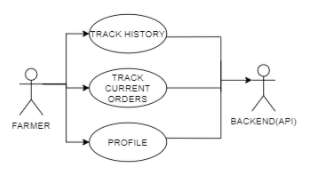
## UML Diagrams

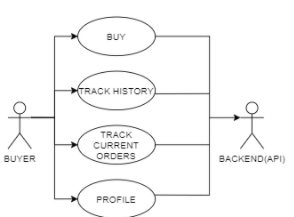
### Class Diagram

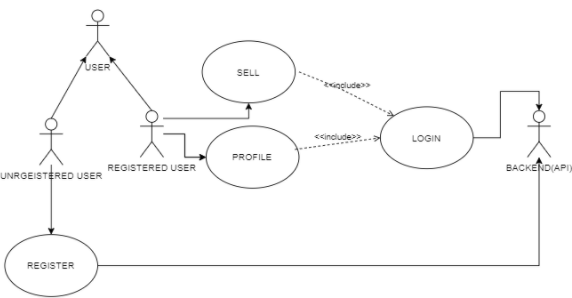


### Use Case Diagrams

****

****

****

****

## Identified Risks

### Risk :

* Sensitive and personal information must be only accessed by authorised personnel.

For example while buying some product over application care should be taken such that confidential data of other users in the system isn’t revealed. Hence we have to devise a mechanism to ensure this as confidentiality is most important in a database system.

### Issues :

* Farming has become a secondary profession looked as a symbol of primitiveness and people practicing it leaving it at a very alarming rate, soon this project will become obsolete as none will be left to use it.
* After amendment of the agriculture act 2020 protest may impose heavy taxes on deals outside of APMCS to make balance between deals in APMCS and deals outside it. That will make this project financially infeasible.
  + Since in future, as we have mentioned in future scope, we will try to predict the future demand of the seasonal produce and try to channelize their production to avoid price crash, we may face some environmental and societal hindrance from those who do not try change for the benefit of others as well but would rather stay with the primitive methods and activities.

### Challenges :

* The main challenge is single/multiple JOINS in the database which may or may not give high accuracy.
* A challenge which we will face is to gain a good hold in the current market situation where people tend to generally avoid using some new application if it is not friendly and easy to use.
* Use of Travelling salesman problem for routing will not be able to give the most efficient route if the nodes are of extreme values.
* Another obstruction which we will face would be by the current dealers as we tend to remove this dealer cycle and hence they will try to hinder our smooth implementation.

### Constraints :

* Currently supports only English language.
* Another technical constraint of our application will be that it will be majorly using GPS (Location services) for best working, so the users of our application must be having some source to update their location (if not by using GPS).

# Implementation Details

This chapter gives information of all the implementation details of the project such as the technologies used etc.

## Algorithms and Techniques Used

### Convolution Neural Network (CNN model)

A **Convolutional Neural Network (ConvNet/CNN)** is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.

The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlap to cover the entire visual area.

We will be using **CNN**  module for image classification during the initial selling procedure.

We will be giving an image input to our model which we have trained and the model will classify it according the type of product it is and will proceed further for filling in the other details of the product.

### Routing Optimization Algorithm – Travelling Salesman Problem

The **travelling salesman problem**(also called the **traveling salesperson problem**or **TSP**) asks the following question: "Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city?" It is an [NP-hard](https://en.wikipedia.org/wiki/NP-hardness) problem in [combinatorial optimization](https://en.wikipedia.org/wiki/Combinatorial_optimization), important in [theoretical computer science](https://en.wikipedia.org/wiki/Theoretical_computer_science) and [operations research](https://en.wikipedia.org/wiki/Operations_research).

The [travelling purchaser problem](https://en.wikipedia.org/wiki/Traveling_purchaser_problem) and the [vehicle routing problem](https://en.wikipedia.org/wiki/Vehicle_routing_problem) are both generalizations of TSP.

In the [theory of computational complexity](https://en.wikipedia.org/wiki/Computational_complexity_theory), the decision version of the TSP (where given a length *L*, the task is to decide whether the graph has a tour of at most *L*) belongs to the class of [NP-complete](https://en.wikipedia.org/wiki/NP-completeness) problems. Thus, it is possible that the [worst-case](https://en.wikipedia.org/wiki/Best,_worst_and_average_case) [running time](https://en.wikipedia.org/wiki/Time_complexity) for any algorithm for the TSP increases superpolynomially (but no more than [exponentially](https://en.wikipedia.org/wiki/Exponential_time_hypothesis)) with the number of cities.

The problem was first formulated in 1930 and is one of the most intensively studied problems in optimization. It is used as a [benchmark](https://en.wikipedia.org/wiki/Benchmark_(computing)) for many optimization methods. Even though the problem is computationally difficult, many [heuristics](https://en.wikipedia.org/wiki/Heuristic) and [exact algorithms](https://en.wikipedia.org/wiki/Exact_algorithm) are known, so that some instances with tens of thousands of cities can be solved completely and even problems with millions of cities can be approximated within a small fraction of 1%.

We’ll be using this algorithm to find the most optimised path for the transport vehicle which would include the cycle from the pickup to the final delivery of the product. This will be our main crux of the application as it will reduce the transportation cost to a great extent.

### Web Scrapping

**Web scraping** is [data scraping](https://en.wikipedia.org/wiki/Data_scraping) used for [extracting data](https://en.wikipedia.org/wiki/Data_extraction) from [websites](https://en.wikipedia.org/wiki/Website). Web scraping software may access the [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web) directly using the [Hypertext Transfer Protocol](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol), or through a web browser. While web scraping can be done manually by a software user, the term typically refers to automated processes implemented using a [bot](https://en.wikipedia.org/wiki/Internet_bot) or [web crawler](https://en.wikipedia.org/wiki/Web_crawler). It is a form of copying, in which specific data is gathered and copied from the web, typically into a central local [database](https://en.wikipedia.org/wiki/Database) or spreadsheet, for later [retrieval](https://en.wikipedia.org/wiki/Data_retrieval) or [analysis](https://en.wikipedia.org/wiki/Data_analysis).

Web scraping a web page involves fetching it and extracting from it. Fetching is the downloading of a page (which a browser does when a user views a page). Therefore, web crawling is a main component of web scraping, to fetch pages for later processing. Once fetched, then extraction can take place. The content of a page may be [parsed](https://en.wikipedia.org/wiki/Parsing), searched, reformatted, its data copied into a spreadsheet, and so on. Web scrapers typically take something out of a page, to make use of it for another purpose somewhere else. An example would be to find and copy names and phone numbers, or companies and their URLs, to a list (contact scraping)

### Clustering Algorithm

**Cluster analysis** or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense) to each other than to those in other groups (clusters). It is a main task of exploratory [data mining](https://en.wikipedia.org/wiki/Data_mining), and a common technique for [statistical](https://en.wikipedia.org/wiki/Statistics) [data analysis](https://en.wikipedia.org/wiki/Data_analysis), used in many fields, including [pattern recognition](https://en.wikipedia.org/wiki/Pattern_recognition), [image analysis](https://en.wikipedia.org/wiki/Image_analysis), [information retrieval](https://en.wikipedia.org/wiki/Information_retrieval), [bioinformatics](https://en.wikipedia.org/wiki/Bioinformatics), [data compression](https://en.wikipedia.org/wiki/Data_compression), [computer graphics](https://en.wikipedia.org/wiki/Computer_graphics) and [machine learning](https://en.wikipedia.org/wiki/Machine_learning).

Cluster analysis itself is not one specific [algorithm](https://en.wikipedia.org/wiki/Algorithm), but the general task to be solved. It can be achieved by various algorithms that differ significantly in their understanding of what constitutes a cluster and how to efficiently find them. Popular notions of clusters include groups with small [distances](https://en.wikipedia.org/wiki/Distance_function) between cluster members, dense areas of the data space, intervals or particular [statistical distributions](https://en.wikipedia.org/wiki/Statistical_distribution). Clustering can therefore be formulated as a [multi-objective optimization](https://en.wikipedia.org/wiki/Multi-objective_optimization) problem.

We will be using **clustering** for clustering the nodes which we are going to give as input to the **TSP** for further processing. This step is our data reduction step in which the outlier nodes are very well handled and hence we can get common nodes of pickup and delivery.

### HTTP Protocol

HTTP functions as a [request–response](https://en.wikipedia.org/wiki/Request%E2%80%93response) protocol in the client–server computing model. A [web browser](https://en.wikipedia.org/wiki/Web_browser), for example, may be the *client* and an application running on a computer [hosting](https://en.wikipedia.org/wiki/Host_(network)) a [website](https://en.wikipedia.org/wiki/Website) may be the *server*. The client submits an HTTP *request* message to the server. The server,which provides *resources* such as [HTML](https://en.wikipedia.org/wiki/HTML) files and other content, or performs other functions on behalf of the client, returns a *response* message to the client. The response contains completion status information about the request and may also contain requested content in its message body.

A web browser is an example of a [*user agent*](https://en.wikipedia.org/wiki/User_agent) (UA). Other types of user agent include the indexing software used by search providers ([web crawlers](https://en.wikipedia.org/wiki/Web_crawler)), [voice browsers](https://en.wikipedia.org/wiki/Voice_browser), [mobile apps](https://en.wikipedia.org/wiki/Mobile_app), and other [software](https://en.wikipedia.org/wiki/Software) that accesses, consumes, or displays web content.

We will be using this protocol for API request handling which is a major part of our backend implementation.

## Tools Used

### Flutter

We will be implementing our application in flutter framework. The main implementation tool of our application will be flutter. As flutter is used to develop applications which can run over IOS or android as well, we have chosen this as our basic framework. Flutter is an open-source UI software development kit created by Google. It is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia, and the web from a single codebase. The first version of Flutter was known as codename "Sky" and ran on the Android operating system.

### Python

Python will be mainly used for implementation of backend code for API and other functionalities which we are providing. Python is an interpreted, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aims to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

### API

API will be mainly used in the backend implementation for pages such as login, registration, buying or selling the product, getting the location of the user, deciding the most optimized route for the delivery and pickup of goods etc. API is an application programming interface is a computing interface that defines interactions between multiple software intermediaries. It defines the kinds of calls or requests that can be made, how to make them, the data formats that should be used, the conventions to follow, etc.

### Node.js

Node.js will be mainly used for collecting the responses from our application via the API and other inputs. Here the main work of Node.js will be to accept server side scripts and responses which the application will send for action.Node.js is an [open-source](https://en.wikipedia.org/wiki/Open-source_software), [cross-platform](https://en.wikipedia.org/wiki/Cross-platform), [back-end](https://en.wikipedia.org/wiki/Front_end_and_back_end), [JavaScript](https://en.wikipedia.org/wiki/JavaScript) [runtime environment](https://en.wikipedia.org/wiki/Runtime_environment) that executes JavaScript code outside a [web browser](https://en.wikipedia.org/wiki/Web_browser). Node.js lets developers use JavaScript to write command line tools and for [server-side scripting](https://en.wikipedia.org/wiki/Server-side_scripting)—running scripts server-side to produce [dynamic web page](https://en.wikipedia.org/wiki/Dynamic_web_page) content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm,[[6]](https://en.wikipedia.org/wiki/Node.js" \l "cite_note-6) unifying [web-application](https://en.wikipedia.org/wiki/Web_application) development around a single programming language, rather than different languages for server- and client-side scripts.

### MongoDB

We have chosen MongoDB as our database where all the database management tasks such as transactions and storing will be done. It will be hosted online so that it can be accessible from anywhere.MongoDB is a [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [document-oriented database](https://en.wikipedia.org/wiki/Document-oriented_database) program. Classified as a [NoSQL](https://en.wikipedia.org/wiki/NoSQL) database program, MongoDB uses [JSON](https://en.wikipedia.org/wiki/JSON)-like documents with optional [schemas](https://en.wikipedia.org/wiki/Database_schema). MongoDB is developed by [MongoDB Inc.](https://en.wikipedia.org/wiki/MongoDB_Inc.) and licensed under the Server Side Public License (SSPL). MongoDB’s document model is simple for developers to learn and use, while still providing all the capabilities needed to meet the most complex requirements at any scale. We provide drivers for 10+ languages, and the community has built dozens more.

## Google Maps(Geo-Locator)

Google maps will be mainly used for location services and collection of the location os the user for mainly determining the pickup point, optimized route for pickup and delivery, delivery point. Google Maps is a web mapping service developed by Google. It offers satellite imagery, aerial photography, street maps, 360° interactive panoramic views of streets, real-time traffic conditions, and route planning for traveling by foot, car, bicycle, air and public transportation.A Flutter geolocation plugin which provides easy access to platform specific location services ([Fused Location Provider Client](https://developers.google.com/android/reference/com/google/android/gms/location/FusedLocationProviderClient) or if not available the [Location Manager](https://developer.android.com/reference/android/location/LocationManager) on Android and [CL Location Manager](https://developer.apple.com/documentation/corelocation/cllocationmanager) on iOS).

Features :

* Get the last known location;
* Get the current location of the device;
* Get continuous location updates;
* Check if location services are enabled on the device;
* Calculate the distance (in meters) between two geo-coordinates;
* Calculate the bearing between two geo-coordinates;

## Operating Environment

### Operating System : Windows/MAC/Linux and Android

### Database : MongoDB(NoSQL)

### Platform : Flutter, Node.js, Python

### Software : postman, emulator, git, VS Code, Android Studio

## Hardware Requirements :

* + 1. Processor : Pentium 4 and higher
    2. RAM : 1GB +
    3. Storage : 1GB +

## Implementation Steps and modules

* + 1. **Farmer Module**

This module mainly contains information about the farmer side of our application user base. There are mainly two parts which are important:

* **CNN Model** – This implementation will be used to predict the initial product name by predicting its image using **Machine learning algorithm**. The application will take in an image of the product and our algorithm will give the predicted name of the product.
* **Selling product and Tracking –** The farmer has the feature to list their product on the application which will be visible to selected buyers for buying. The initial step of selling the product is the above **CNN model image prediction.**

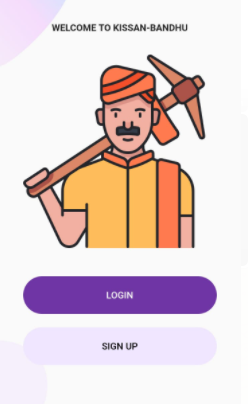
The tracking of the product will show the farmer the status of the product which is already sold.

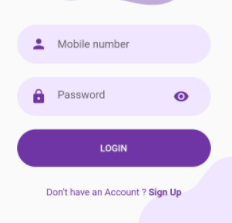
* + 1. **Buyer Module**

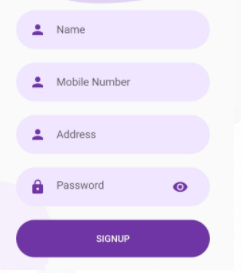
This module mainly contains features pertaining to the buyer side of our user base. The main feature which it will implement is :

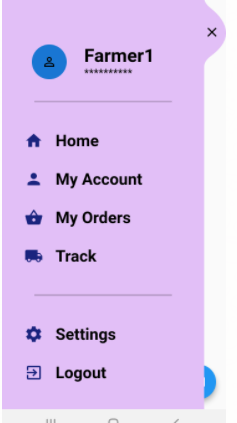
* **Buy Product**  - This implementation feature will be used by the user to buy the products which are listed on the application for sale by the farmer. It will mainly contain buyers such as individual buyer and market owners.
* **Track Product** – This will be used by buyer to track the status of his/her purchased product.
  + 1. **Driver Module**

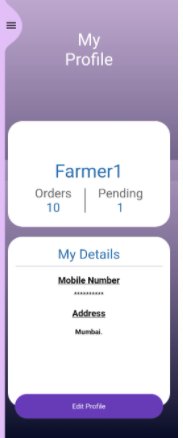
## Some of the temporary implementation screenshots until November end.

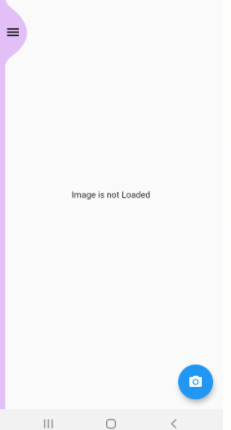
****

****

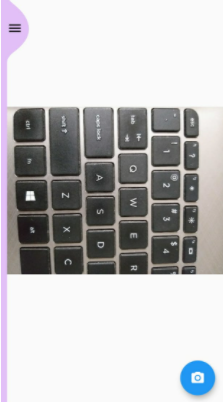
****

****

****

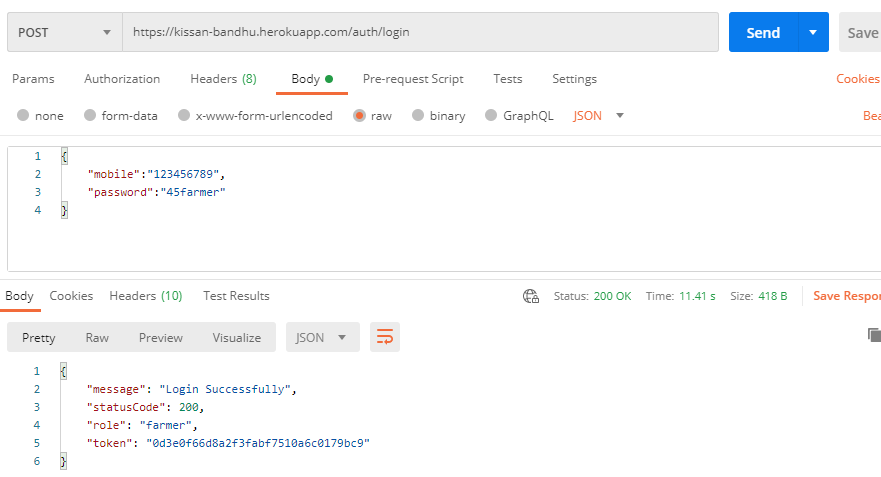
****

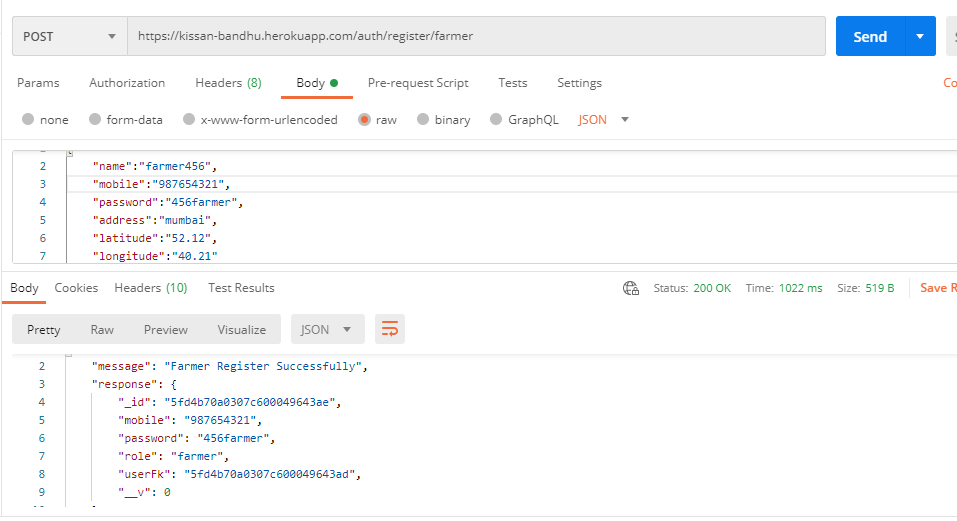
****

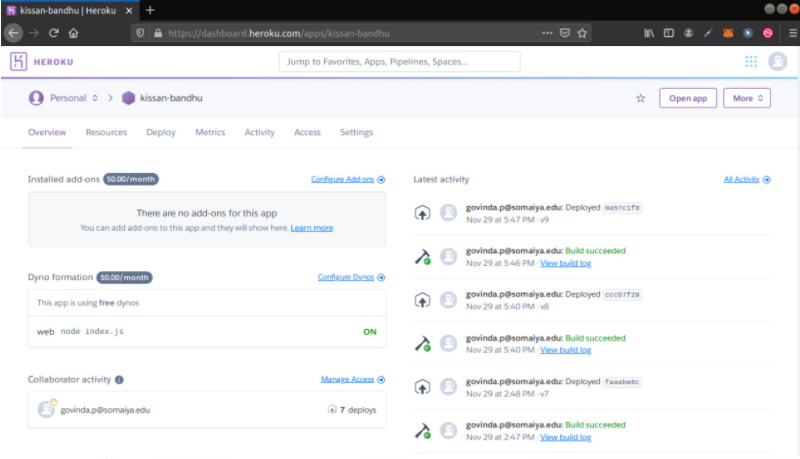
****

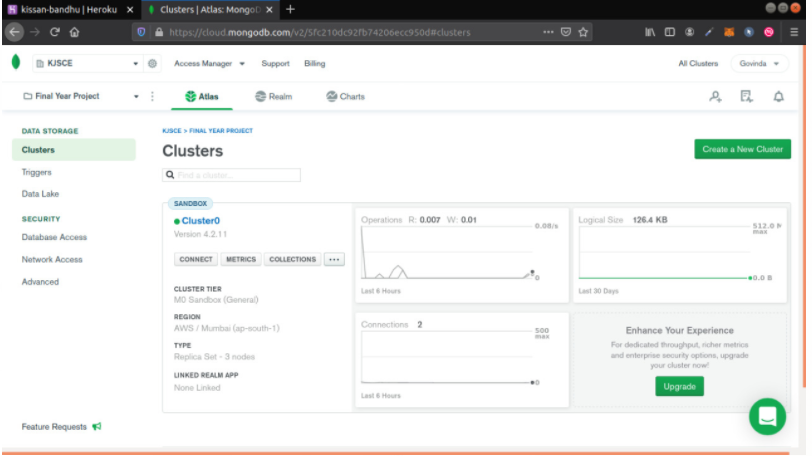
****

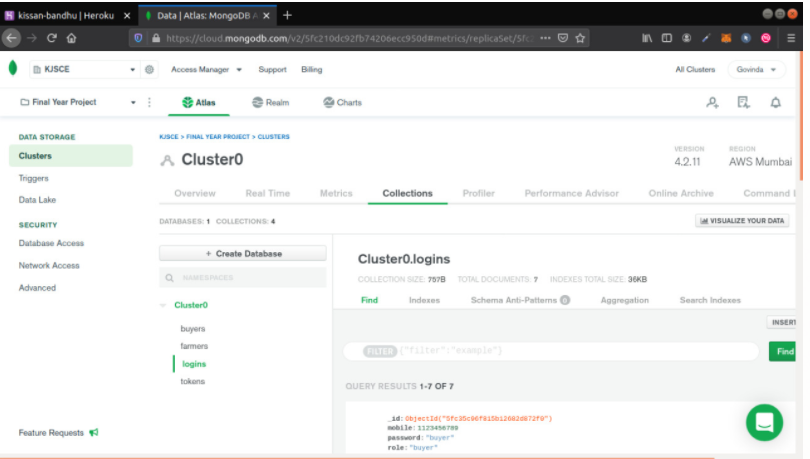
**Geolocator Latitudes and longitudes**

****

****

****

****

****

# Conclusion

This chapter gives the conclusion of the project , apart from this we also discuss the future scope of this project.

## Conclusion

This project aims to reinvent the production and transactional wheel of buying and selling the agricultural products. Currently the project will try to reach the maximum area of the user base but due to some usability constraints will lead to restriction of the area where the product will make impact. It will be easy for the use by any layman as the application will try to avoid any complicated functionalities. Although the current prototype only allows limited functions, we hope the future will increase the number of functionalities and applications that can be used by the user. We hope to be an integral part of this research that will change lives and revolutionize the extent of independency the members of our agricultural society will achieve with this technology.

## Future Scope

1. Creating a network and chain of the farmers to make efficient production cycle.
2. Creating a network of producers and consumers while maintaining supply chain among them.
3. Better transparency of transaction using Block chain technology.
4. Predicting the future demand of a particular product in and hence making shift in storage of the product by utilizing the empty rounds making the transportation cycles efficient.
5. Predicting future demand of seasonal produces to avoid price crash in products during non-demanding seasons while maintaining profitable margin.

# Bibliography

* <https://flutter.dev/docs>
* <https://docs.mongodb.com/>
* <https://expresses.com/en/5x/API.html>
* THE FARMERS’ PRODUCE TRADE AND COMMERCE(PROMOTION AND FACILITATION) ORDINANCE, 2020
* Role of APMCs in Agricultural Marketing in India- A Study
* Farmers Suicides in India research paper
* Farmer Suicides in India - Trends across Major States, 1995–2011
* THE FARMERS’ PRODUCE TRADE AND COMMERCE (PROMOTION AND FACILITATION) ACT, 2020
* THE FARMERS (EMPOWERMENT AND PROTECTION) AGREEMENT ON PRICE ASSURANCE AND FARM SERVICES ACT, 2020
* THE AGRICULTURAL PRODUCE (GRADING AND MARKING) ACT, 1937