Chapter 1

INTRODUCTION

1.1 Introduction

Agriculture is the farmer system where they can plan, monitor and analyse the activity of the farmers production system. It manages farmer operation with one system and organizes data in one place. It helps smart farmers become even smarter. This creates in partnership with growers and buyers. It inspires farmer to produce and buyers to consume fresh goods.

Agriculture System will make better connection among Farmers and Buyers ensure quality food. Standardize and increase efficiency of Farm Buddy process.

1.2 Objectives of the Project

The specific objectives of the project include:

❖ To provide qualitative foods to the buyers.

❖ Implementing an automated/online Agriculture system.

❖ To inspire farmer to produce quality goods and supply to the buyers.

❖ Eco-friendly farming system.

1. 3 Scope of the Project

It is focused on studying the existing system of Agriculture in and to make sure that the peoples are getting quality fresh goods. This is also will produce:

• Less effort and less labour intensive, as the primary cost and focus primary on creating, managing, and running a secure quality food supply.

• Increasing number of buyers as individuals will find it easier and more convenient to buy goods.

• Easy management

1.4 Methodology/Procedure

For the development of project, the designing of database was done on PHPMYADMIN, back end was coded in basic PHP and for frontend we used the same basic PHP codes. Software methodologies are concerned with the process of creating software – not so much the technical side but the organizational aspects. Several software development approaches have been used since the origin of information technology.

1.5 Project Framework

A framework is a standardized set of concepts, practices, and criteria for dealing with a common type of problem, which can be used as a reference to help us approach and resolve new problems of a similar nature. The aim of framework is to provide a common structure so that developers don't have to redo it from scratch and can reuse the code provided. In this way, frameworks allows us to cut out much of the work and save a lot of time.

1.6 Data and Information

Data collection plays an important role in a projects succession and also it plays an inevitable role in the timely completion of the project. The data in the project includes contact information of the clients and their respective feedbacks/complaints which is stored in a database. To assure safety, only the admin has proper access to the information provided by the clients.

1.7 Primary Source of Data

Primary data are the first-hand data. The necessary information was collected from day-to-day observation, problems, instructions of supervisor. Queries and personal discussion with the staff of the organization.

• Observation of working environment.

1.8 Source of Data

The Secondary sources of data were collected in order to achieve the real and fact data as far as available.

The major sources of secondary data are as follows:

• Annual reports of the concerned organization

• Related websites.

Chapter 2

SOFTWARE REQUIREMENTS

2.1 Software Used:

Operating system: Windows 98, XP, 7,8 or 10 or Linux

Languages (Front end) : HTML,PHP and Bootstrap

(Back end): SQL

IDE: Notepad++, Xampp and MYSQL

2.2 Software Description:

2.2.1 XAMPP(PhpMyAdmin)

PhpMyAdmin can manage a whole MySQL server as well as a single database. To accomplish the later you'll need a properly set up MySQL user who can read/write only the desired database. It's up to you to look up the appropriate part in the MySQL manual.

➢ browse and drop databases, tables, views, columns and indexes

➢ create, copy, drop, rename and alter databases, tables, columns and indexes

➢ maintenance server, databases and tables, with proposals on server configuration

➢ execute, edit and bookmark any SQL-statement, even batch-queries

➢ load text files into tables

➢ create1 and read dumps of tables

➢ export1 data to various formats: CSV, XML, PDF, ISO/IEC 26300 – Open Document Text and Spreadsheet, Word, and LATEX formats.

➢ import data and MySQL structures from Open Document spreadsheets, as well as XML, CSV, and SQL files

➢ administer multiple servers

➢ manage MySQL users and privileges

➢ check referential integrity in MyISAM tables

➢ using Query-by-example (QBE), create complex queries automatically connecting required tables

➢ create PDF graphics of your Database layout

➢ search globally in a database or a subset of it

➢ transform stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link

➢ track changes on databases, tables and views

➢ support InnoDB tables and foreign keys

➢ support MySQL, the improved MySQL extension

➢ create, edit, call, export and drop stored procedures and functions

➢ create, edit, export and drop events and triggers.

➢ communicate in

➢ synchronize two databases residing on the same as well as remote servers.

2.2.2 PHP

➢ You need PHP 5.2.0 or newer, with session support, the Standard PHP Library (SPL) extension and JSON support.

➢ To support uploading of ZIP files, you need the PHP zip extension.

➢ For proper support of multibyte strings (e.g. UTF-8, which is currently the default), you should install the mbstring and ctype extensions.

➢ You need GD2 support in PHP to display inline thumbnails of JPEGs ("image/jpeg: inline") with their original aspect ratio.

➢ When using the "cookie" authenticated method, the mcrypt extension is strongly suggested for most users and is required for 64–bit machines. Not using mcrypt will cause phpMyAdmin to load pages significantly.

2.3 Problem Analysis

It is related with the accessing the detailed information of a user and a candidate. So, I have initiated this project with simple requirements regarding the user and candidate information. Some of the problems for designing and developing this project are discussed below:

2.3.1 Design and Development Problem

• Problem in running XAMPP.

• To debug the error during the development.

• To show a relationship between entity.

• Minor error with database table.

2.4 Operational Feasibility

The system is operational feasible as the system can be operate by normal users with basic computer skills without any additional trainings. We have developed this system with the willingness and ability to create, manage and operate the system which is easy for the end users to operate it.

CHAPTER 3

DESIGN

3.1 Use Case Diagram

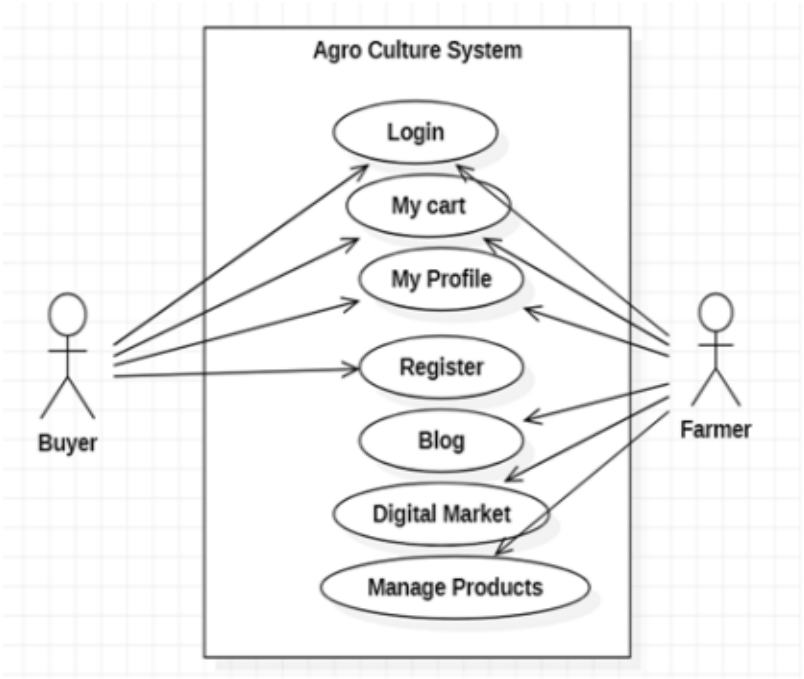


Fig3.2 Use Case Diagram

Above figure represents Use Case Diagram of the project and is a useful technique for identifying, clarifying, and organizing system requirements. It describes how a user uses a system to accomplish a particular goal. Use cases help ensure that the correct system is developed by capturing the requirements from the user's point of view.

3.2 ER DIAGRAM

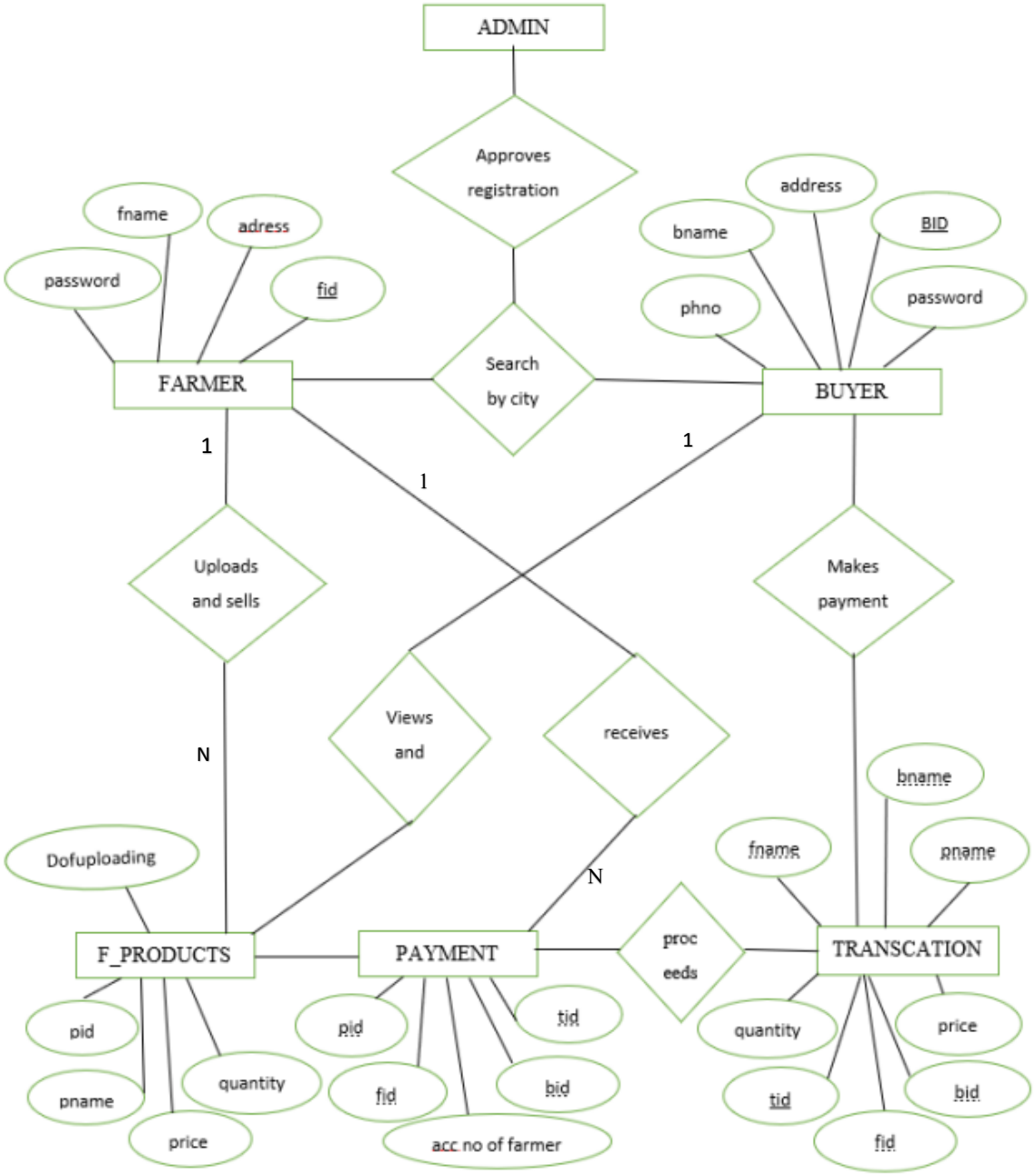
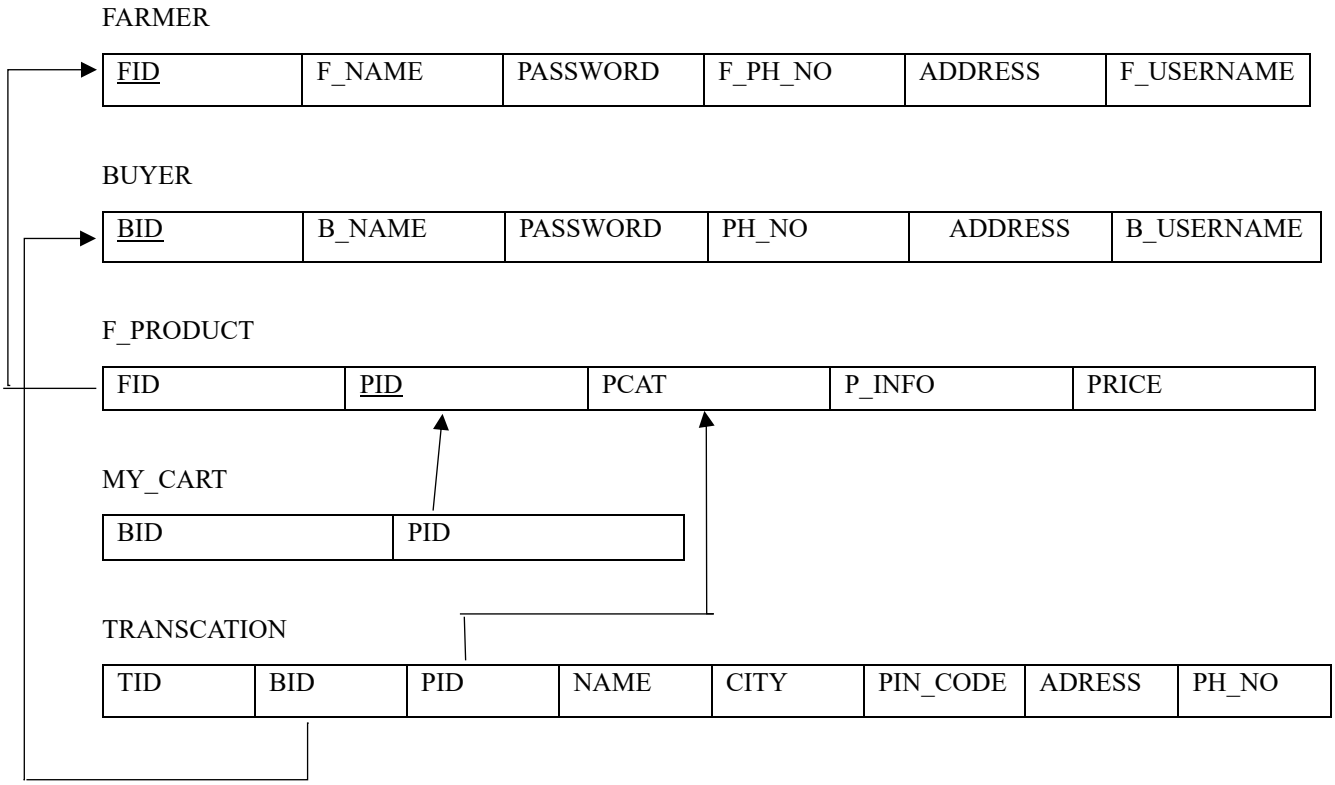


Figure 3.2 ER DIAGRAM OF AGRICULTURE MANAGEMENT SYSTEM

3.3 SCHEMA DIAGRAM



3.3 SCHEMA DIAGRAM

3.4 TABLES DESCRIPTION

The "BUYER" table provides all information about Buyer.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	bid	int(100)			No	None		AUTO_INCREMENT	Change Drop More
2	bname	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
3	busername	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
4	bpassword	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
5	bhash	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
6	bemail	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
7	bmobile	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
8	baddress	text	latin1_swedish_ci		No	None			Change Drop More
9	bactive	int(100)			No	0			Change Drop More

TABLE NO 1 OF BUYER

The “FARMER” table provides all information about Farmer.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	fid	int(255)			No	None		AUTO_INCREMENT	Change Drop More
2	fname	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
3	fusername	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
4	fpassword	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
5	fhash	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
6	femail	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
7	fmobile	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
8	faddress	text	latin1_swedish_ci		No	None			Change Drop More
9	factive	int(255)			No	0			Change Drop More
10	frating	int(11)			No	0			Change Drop More
11	picExt	varchar(255)	latin1_swedish_ci		No	png			Change Drop More
12	picStatus	int(10)			No	0			Change Drop More

TABLE NO 2 OF FARMER

The “F_PRODUCT” table provides all information about f_product.

+ Options

	fid	pid	product	pcat	pinfo	price	pimage	picStatus
Edit Copy Delete	3	27	Mango	Fruit	Mango raseela	500	Mango3.jpeg	1
Edit Copy Delete	3	28	Ladyfinger	Vegetable	Its veggie	1000	Ladyfinger3.jpg	1
Edit Copy Delete	3	29	Bajra	Grains	bajre di rti	400	Bajra3.jpg	1
Edit Copy Delete	3	30	Banana	Fruit	Jalgaon banana	400	Banana3.jpg	1
Edit Copy Delete	4	31	Strawberry	Fruit		100	Strawberry4.png	1
Edit Copy Delete	4	32	sweet lime	Fruit		80	sweet lime4.png	1

TABLE NO 3 OF F_PRODUCT

The “MY_CART” Table provides all information about my_cart.

TABLE NO 4 OF MY_CART

The “TRANSACTION” Table provides all information about transaction.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	tid	int(10)			No	None		AUTO_INCREMENT	Change Drop More
2	bid	int(10)			No	None			Change Drop More
3	pid	int(10)			No	None			Change Drop More
4	name	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
5	city	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
6	mobile	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
7	email	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
8	pincode	varchar(255)	latin1_swedish_ci		No	None			Change Drop More
9	addr	varchar(255)	latin1_swedish_ci		No	None			Change Drop More

TABLE NO 5 OF TRANSACTION

The “F_PRODUCT” Table provides all the information about the farmer products

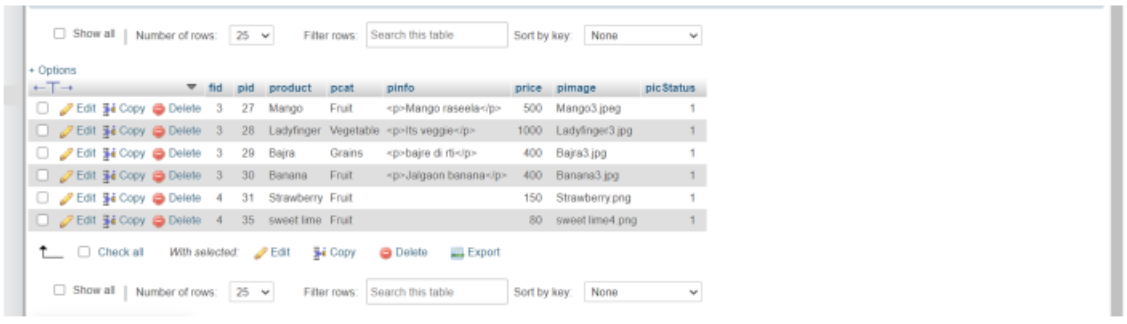


TABLE NO 6 OF F_PRODUCT

Chapter 4

DATABASE TECHNIQUE AND RESULTS

4.1 TRIGGER:

There is a basic trigger in this project.

If we entered the mobile number more than 10 digits or less than 10 digits,

then the message Invalid mobile number will pop up.

```
$sql = "SELECT * FROM members WHERE Username='$user'";
$result = mysqli_query($conn, $sql);
$num_rows = mysqli_num_rows($result);


if($num_rows == 0)
{
    $_SESSION['message'] = "Invalid User Credentials!";
    header("location: error.php");
}
else
{
    $User = $result->fetch_assoc();
```

4.2 PROCEDURE:

The stored procedure in this project will fetch details of all the buyers.

```
<%
        String sql = "SELECT `bid`, `bname`, `bmail`, `b_phno`, `b_address`,
`b_username`,`b_password`, WHERE b_name = '"+b_name+"'";
                    ResultSet rs = dao.getData(sql);
%>
```

4.3 NORMALIZATION:

The complete tables of the database in the project is normalized , obeying all the rules of normalization

1NF:

1NF disallows relations within relations or relations as attribute values within tuples. The only attribute values permitted by 1NF are single atomic (or indivisible) values.

2NF:

A functional dependency X → Y is a full functional dependency if removal of any attribute A from X means that the dependency does not hold any more; that is, for any attribute A ε X, (X – {A}) does not functionally determine Y

3NF:

Transitive functional dependency

A functional dependency X→Y in a relation schema R is a transitive dependency if there exists a set of attribute Z that are neither a primary nor a subset of any key of R(candidate key) and both X à Z and Y à Z holds

Definition: A relation schema R is in third normal form (3NF) if it is in 2NF and no non-prime attribute A in R is transitively dependent on the primary key.

1NF: In the above table there are no multi valued attributes. Thus, the functional dependency FD1 and relation satisfies 1NF.

2NF: There are no partial dependencies found in the above defined functional dependencies. Thus, we can say that relation satisfies 2NF.

3NF: There are no transitive dependencies found in the above defined functional dependencies. Thus, we can say that relation satisfies 3NF.

4.4 TESTING:

4.4.1Testing

Testing is evaluation of the software against requirements gathered from users and system specifications. Testing identifies important defects, flaws, or an error in the application code that must be fixed .It also assesses the feature of a system. Testing assesses the quality of the product.

4.4.2 Unit Testing

Unit testing refers to the testing certain functions and areas of the code. It gives the ability to verify that all the functions work as expected.

4.4.3 Integration Testing

Integration testing is basically a logical extension of unit testing. In simple words, two tested units are combined into a component and the interface between them is tested. It identifies problems that occur when different units are combined. The different modules of this project have undergone integration testing while being merged.

4.4.4 System Testing

System testing tests the behavior of whole system as defined by the scope of the development project. It might include tests based on risks as well as requirement specifications, business process, use cases or other high level descriptions of system behavior, interactions with the operating systems and system resources. It is most often the final test performed to verify that the system meets the specification and its objectives. System testing has been performed at the completion of each feature and is still taking place to make improvements on the existing system.

CHAPTER 5

SNAPSHOTS

5.1 snapshots of project

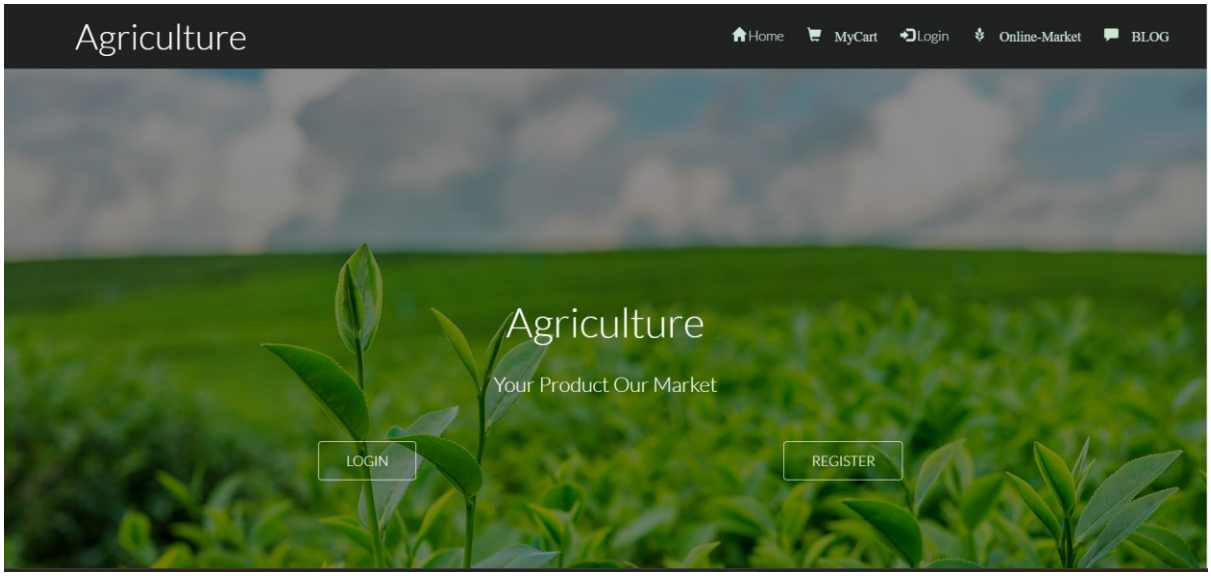


Figure.5.1 Home page

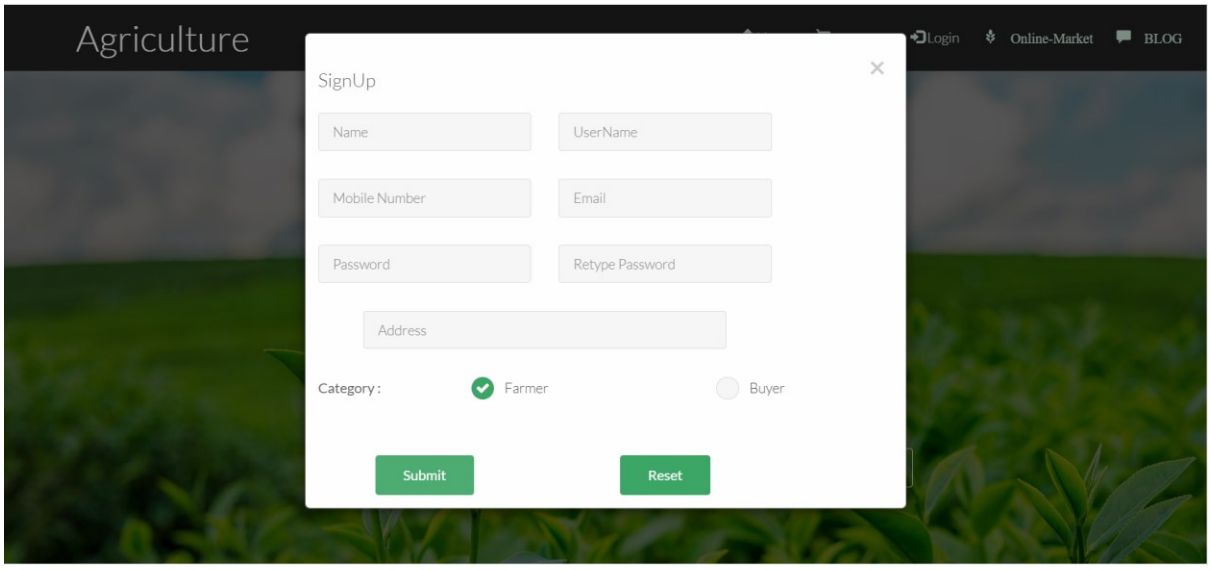


Figure 5.2 Farmer/Buyer registration page

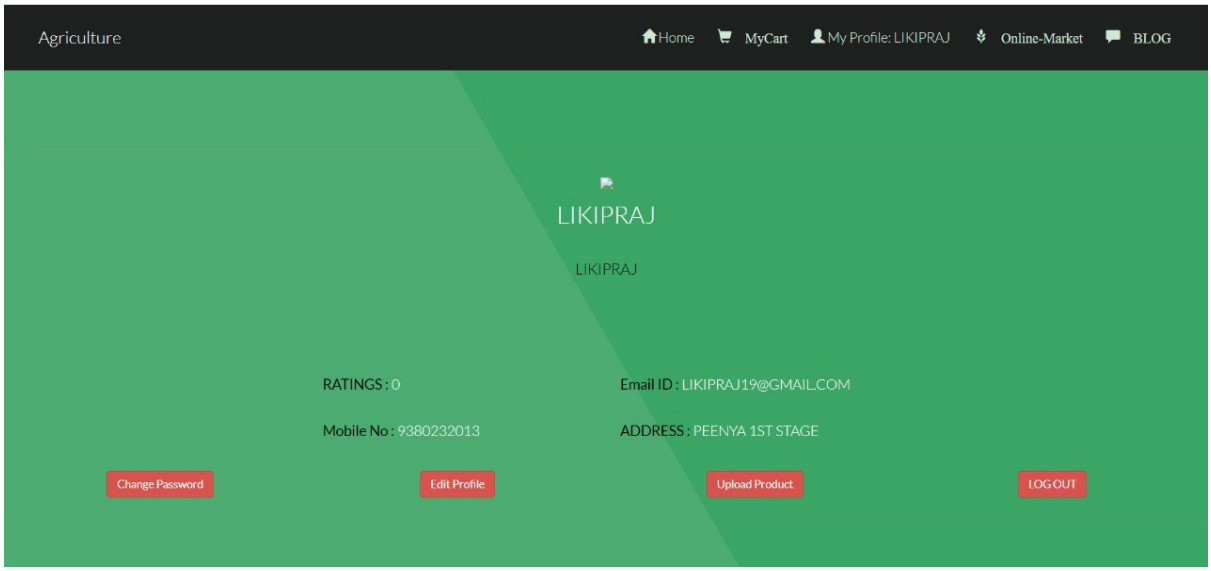


Figure.5.3 Farmer/Buyer Profile page

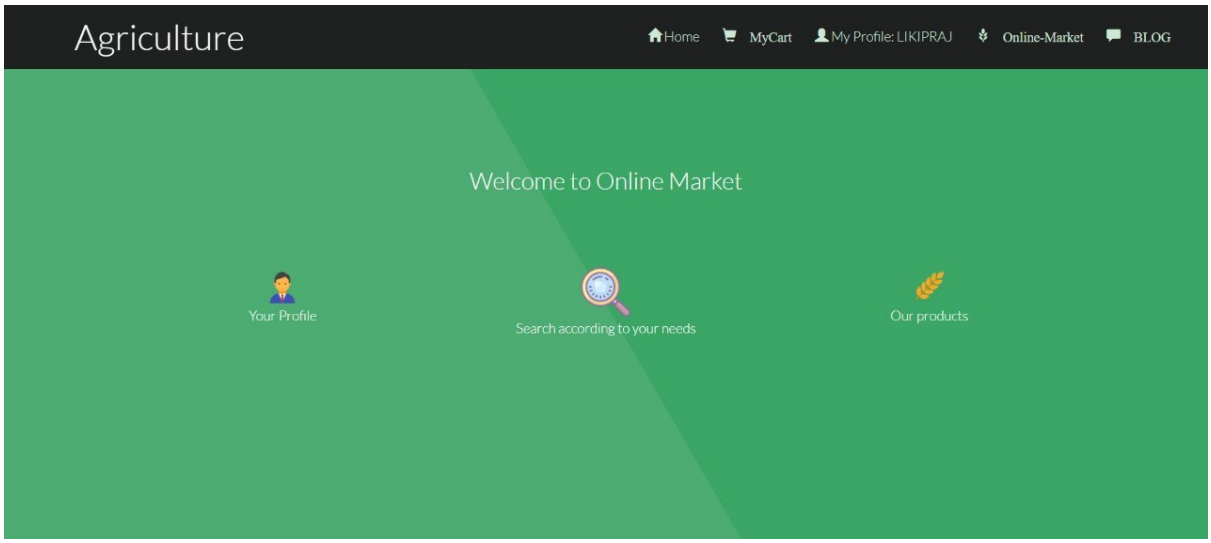


Figure.5.4 F-product page

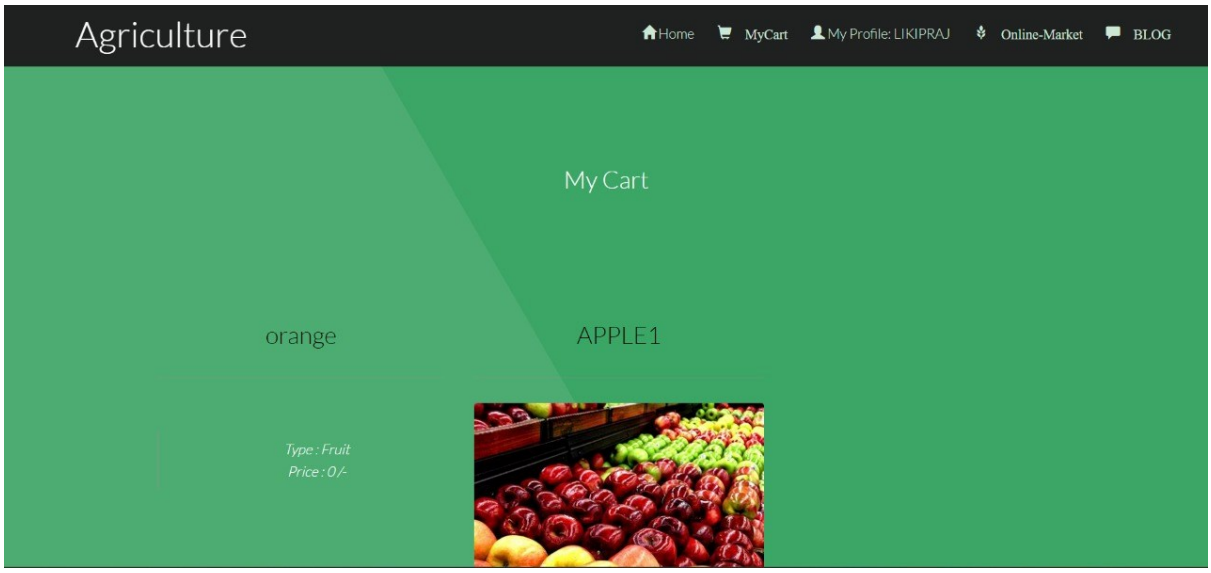


Figure.5.5 Product Purchase Page

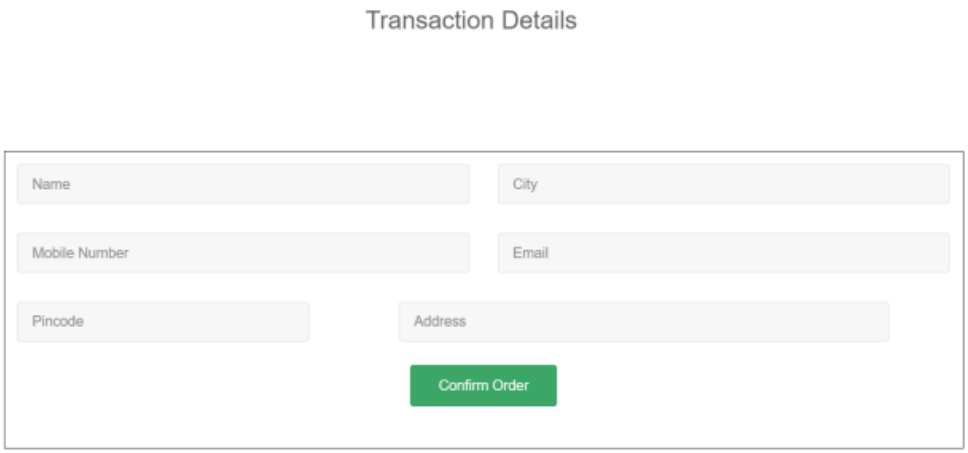


Figure.5.6 Transaction Details Page

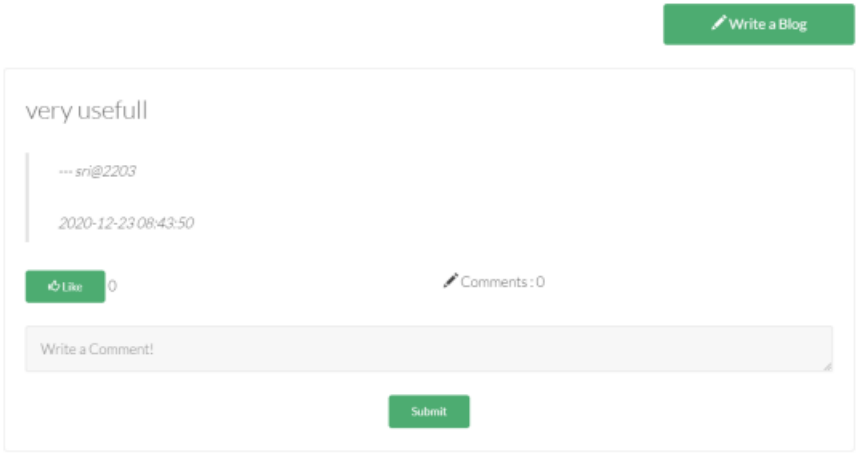


Figure.5.7 Blog page

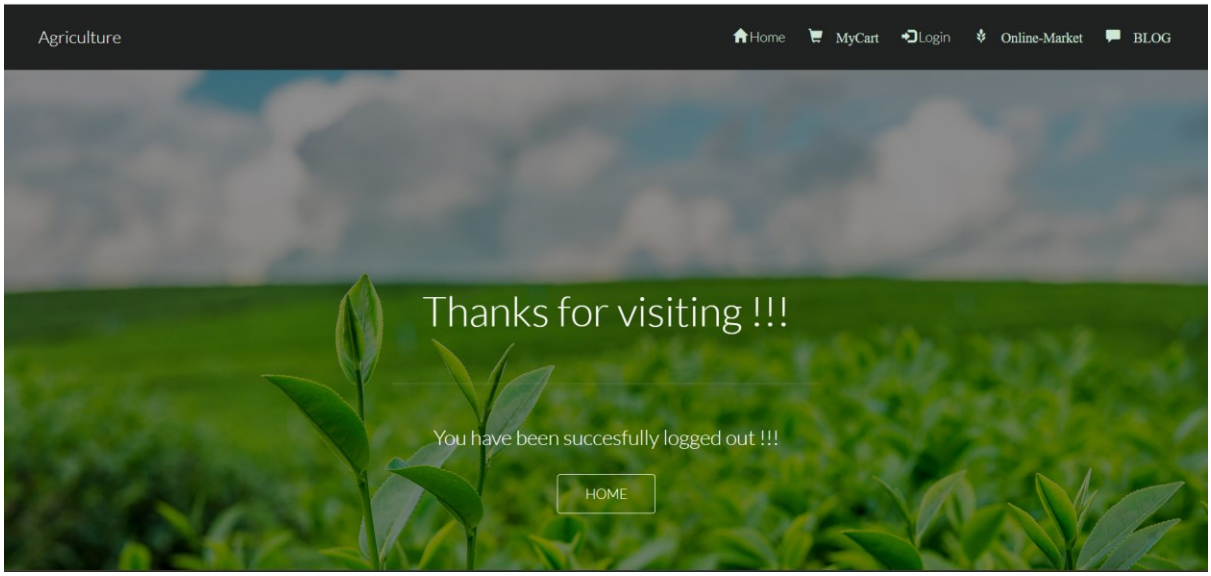


Figure.5.8 Member log out page

CONCLUSION

Agriculture System will make better connection among Farmers and Buyers ensure quality food. Standardize and increase efficiency of Agriculture process. We drive agicculture transactions through our digital platform. In combination with our server partnership network . Agri market place accommodates online payments between buyers and farmers, product quality check options and end to end logistic services. Agri marketplace Accommodates direct transactions between buyers and farmers. all the intermediate supply chain stages are covered by agri mp and our partnership network.

REFERENCES

- www.w3schools.com
- www.tutorialspoint.com
- https://www.w3schools.com/php/default.asp
- https://www.sitepoint.com/php/
- https://www.php.net/
- https://www.javatpoint/php.in/
- https://www.mysql.com/
- http://www.mysqltutorial.org
- http://www.javatpoint/mysql.org
- https://www.apachefriends.org/download.htm