Abstract—A website helping farmers to ensure greater profitability by the process of direct farmer to user communication. This service increases business communication and brings transparency to the system. Each kind of user will have different pages for login with different functionalities. This innovative website allows good user to user communication. It allows each user to login and communicate with other user depending on their role. All users may also submit their grievances and complaints to admin using their farmer login on a contact us page and admins will check the page regularly using their login IDs and passwords. Farmers can check weather for the upcoming 5 days. They can also check whether their crop has been infected with any kind of disease which can be detected by using multi class svm classifier.

Keywords—Farmastant, customers, farmers, merchants, shipping, products, goods, chatbot Ai1, online agent AIMUS.

I. INTRODUCTION

The aim of our project is to give assistance to farmers by providing web service that can be used by all kinds of users. Our project helps farmers to ensure greater profitability by the process of direct farmer to user communication. Users can chat with an AI assistant chatbot named Farmastant Ai1. They can even be connected to an agent in case of any doubts not clarified by Ai1 or to post complaints to administrator about products or other users. Farmers can even know about upcoming 5 days weather to plan things in advance. If they suspect any crop is affected by a disease, they can just verify it from our matlab GUI code so that they can buy pesticides or other disinfectants. It allows farmers to communicate with respective dealers and buy goods from them. When dealers publish goods available for them from there portal farmer can view the goods available to buy in their portal. The farmers may also register their complaints using their respective login on a separate complaint page where it can only be seen by dealers. And another portal is customer portal. Farmer can sell his goods directly through this website to the customer without the need of middlemen. Farmer gets suggestion for growing crops with the help of agro api and the readings of past 30 years of areas.

II. OBJECTIVE

a. Separate pages with different functionality for farmers, merchant, customer, transport and contact us in case of queries or complaints.
b. Separate page for checking current and upcoming weather up to 5 days so that they can plan their activities accordingly.
c. Index page where farmers new productswill be posted automatically.
d. Customers can get view all the available products to decide what to buy.
e. An effective interface easier to understand designed following design principles.
f. Separate page for transportation of products/goods from merchants to farmers and farmers to customers.
g. Chat with Virtual Assistant Farmastant Ai1.
h. Chat with online desk help AIMUS at Zendesk.

III. MOTIVATION

In this era of pandemic life is difficult for everyone but is more difficult for farmers. Cultivating crops and selling them is already difficult enough in this technologically advancing world. This pandemic has made it much more difficult due to lack of proper facilities available and curfews decreasing the work time. Customers are facing lot of issues to go out
and buy what they want and selecting is always a hassle. But what if we can find a way to connect agriculture and technology? What if we can share knowledge about advanced farming techniques with just a message request from anywhere in the world? What if customers can take their time deciding what to buy and where to buy without actually being present there? Our motivation is to create a website that is easier even for novice users and help them buy/sell the products without even travelling there physically.

IV. BACKGROUND

The background of our project is a context where pandemic has not ended and the world is devastatingly affected by the pandemic. Hence our project helps people to interact and buy products and take decisions without even actually being there.

Technology has taken deeper roots into all and every part of the world irrespective of the sector. Artificial Intelligence has become more of a habit than a trend.

V. PROJECT SCOPE

- Ensuring greater profitability to farmers through our website.
- This service boosts business communication and brings transparency in the system. Communication happens between agents and customers also.
- Users can be over all the places for communication and dealing with each other as long as shipping services are available in those places.
- Farming tip is given to farmers easily and their complaints can be solved with VA Ai1 Chatbot or Zendesk help agent.

VI. GOALS

a) Separate pages with different functionality for farmers, merchant, customer, transport and contact us in case of queries or complaints.

b) Separate page for checking current and upcoming weather up to 5 days so that they can plan their activities accordingly.

c) Page where farmers can post their new products.

d) Customers can get view all the available products to decide what to buy.

e) An effective interface easier to understand designed following golden rules and heuristic principles.

f) Separate page for transportation of products/goods from merchants to farmers and farmers to customers.

VII. PRODUCT PERSPECTIVE

An Agricultural E-Commerce stores the following information.

- **ProductDetails:**
  It includes the products to be sold or to be bought and its details about it like product code number.

- **Customerdescription:**
  It includes customer details and his order history.

- **Farmerdescription:**
  It includes details both personal and his products and orders. This information may be used for keeping the records of the farmer for any emergency or for any other kind of information.

- **Merchantdescription:**
  It includes merchant code, name, address and phone number. This information may be used for keeping the records of the merchant for any emergency or for any other kind of information.

- **Purchasedescription:**
  It includes customer / farmer details, product code number, maybe date of purchase, details etc.

- **Weather description:**
  It depicts the information of weather at a place.

- **Chatbot description:**
  Users can chat with Ai1 or AIMUS.

VIII. USER CLASSES AND CHARACTERISTICS

Users of the system are classified as customers, farmers, merchants.

- **Purchase a product**
  Farmer purchase a product
  Customer purchase a product
View Orders
Customer able to view products in his purchase history
Farmer able to view products in buy and sell history
Merchant able to view products in sell history.
Based on the type of user, they have following functions:

- **CUSTOMER FUNCTIONS**
  1. Show all products available to buy.
  2. Show all products bought recently in history.
  3. Show all products bought not recently.

- **FARMER FUNCTIONS**
  1. Show all products available to sell.
  2. Show all products available to sell.
  3. Show all products already bought.
  4. Show all products already sold.
  5. Add / Delete products to sell.

- **MERCHANT FUNCTIONS**
  1. Show all products available to sell.
  2. Show all products already sold.
  3. Add / Delete products to sell.

- **ADMINISTRATIVE**
  1. Add / delete Customers, Farmers, Merchants.
  2. Control product display.
  3. Update weather at a particular area.
  4. Update new features if ever required.

IX. OPERATING ENVIRONMENT
- WampServer
- Client/Server system
- OS: Windows.
- Database: MySQL database
- Platform: PHP
- A computer system with minimum of 4gb RAM and i5 gen would be suitable for fast processing.

X. DESIGN - IMPLEMENTATION CONSTRAINTS

This application will run on windows without any issues but it is limited to windows. If the admin is using mac then he has to use MAMP as the solution stack for localhost. This application uses Mysql database so if you want to change database then platform might require changing too based on database being used. For example, Java for JDBC. All three of us as a group will be held responsible for the maintenance of software delivered.

XI. USER DOCUMENTATION

Users will find a contact us page to contact us for any queries and they will get help as soon as possible. It will be online help and no user manuals will be available. We might try to come up with a tutorial. Hence it would be all digital.

XII. ASSUMPTIONS AND DEPENDENCIES

There are a lot of assumed facts and dependencies that would affect the requirements stated in SRS. If we face any issues with WampServer then we might use XamppServer and hence the database would be changed to MariaDB. Similarly, if Mac OS is used then MAMPServer will be used so everything changes.

XIII. EXTERNAL INTERFACE REQUIREMENTS

1. User Interfaces
- Front-end software: PHP
- PHP is used to host web application over a server and integrate with database.
- Back-end software: MySQL
- Mysql is used as the database for its ease of use.

2. Hardware Interfaces
- Windows system.
- System with a minimum of 4GB RAM and i3 gen for performance.
- A web browser which supports CGI, HTML & Javascript.
- Any browser that supports these should be used. General browsers like Chrome, Edge are recommended.

3. Software Interfaces
- Following are the softwares used for the Farmastant Web application.

<table>
<thead>
<tr>
<th>Software</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS</td>
<td>We have chosen Windows OS for its best support and user-friendliness.</td>
</tr>
<tr>
<td>Database</td>
<td>To save the details we have chosen MySQL database.</td>
</tr>
<tr>
<td>SSL</td>
<td>To implement the project, we have chosen PHP language for its more interactive support.</td>
</tr>
<tr>
<td>Server</td>
<td>WampServer</td>
</tr>
</tbody>
</table>

4. Communications Interfaces

This project supports all types of web browsers. We are using simple electronic forms for the login, purchasing, registration, etc.. There are no communication security or encryption issues.

XIV. SYSTEM FEATURES

a) Farmer Purchase

4.1.1 Description and Priority

This feature is for farmer to purchase product. This feature has high priority. Benefit rating is high, penalty is also high, cost would be moderate and has a moderate risk factor 5.

4.1.2 Stimulus/Response Sequences

- Search for required products sold by merchant.
Displays a list of all available products and purchase required product.

4.1.3 Functional Requirements

1FR-1: Merchant should be able to add/delete data
1FR-2: Every time farmer searches for products, he should have access to updated list.

b) Customer Purchase

4.2.1 Description and Priority
This feature is for customer to purchase product. This feature has high priority. Benefit rating is high, penalty is also high, cost will be moderate and has a moderate risk factor.

4.2.2 Stimulus/Response Sequences

- Search for required products sold by farmer.
- Displays a detailed list of available products and purchase required product.

4.2.3 Functional Requirements

2FR-1: Merchant should be able to add/delete data
2FR-2: Every time farmer searches for products, he should have access to updated list.

XV. Other Non-functional Requirements

a. 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. It would also be wise to store all the required files in a drive such as Google Drive or OneDrive.

b. Security Requirements
The system needs to be secured. Hence, software like McAfee Security should be sufficient. Make sure that these software are trustworthy since all these are third party software.

c. Software Quality Attributes

- AVAILABILITY: The products should be available on the specified date and specified time as many users are doing purchases. If not available either users need to be informed of availability status or they should be excluded from search.
- CORRECTNESS: The products description should match its real life specifications. Users should have the right to get what they purchased and it should match the specs.
- MAINTAINABILITY: The administrators and sellers should maintain correct details of products. In case of error or unavailability, proper management is important.
- USABILITY: The products should satisfy a maximum number of user needs. It is better to have products that satisfy a vast section of users.

d. Business Rules
Merchants sell products to farmers. Farmers sell products to customers. There is no strict classification in real life as all are users but in this web application all are considered different. Same person has to create different accounts for merchant, customer, farmer.
ARCHITECTURE

WEBSITE GUI
XVI. SCHEDULE, TASKS AND MILESTONES

<table>
<thead>
<tr>
<th>View</th>
<th>Name</th>
<th>Duration</th>
<th>Free Slack</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Start project</td>
<td>0 days</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 9:00 AM</td>
</tr>
<tr>
<td>1</td>
<td>Team and project selection</td>
<td>4 days</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 10:00 AM</td>
</tr>
<tr>
<td>2</td>
<td>Requirement Analysis</td>
<td>15 days</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 11:00 AM</td>
</tr>
<tr>
<td>3</td>
<td>Research</td>
<td>5 days</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 11:00 AM</td>
</tr>
<tr>
<td>4</td>
<td>Brainstorming</td>
<td>1 day</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 11:00 AM</td>
</tr>
<tr>
<td>5</td>
<td>Feature list</td>
<td>2 days</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 11:00 AM</td>
</tr>
<tr>
<td>6</td>
<td>Budget analysis</td>
<td>3 days</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 11:00 AM</td>
</tr>
<tr>
<td>7</td>
<td>GUI features list</td>
<td>4 days</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 11:00 AM</td>
</tr>
<tr>
<td>8</td>
<td>Design</td>
<td>32 days</td>
<td>0 days</td>
<td>2/3/21 8:00 AM</td>
<td>4/4/21 11:00 AM</td>
</tr>
<tr>
<td>9</td>
<td>Feature Prioritization</td>
<td>2 days</td>
<td>0 days</td>
<td>3/3/21 8:00 AM</td>
<td>3/3/21 11:00 AM</td>
</tr>
<tr>
<td>10</td>
<td>Architecture definition</td>
<td>11 days</td>
<td>0 days</td>
<td>4/3/21 8:00 AM</td>
<td>4/3/21 11:00 AM</td>
</tr>
<tr>
<td>11</td>
<td>Specification development</td>
<td>9 days</td>
<td>0 days</td>
<td>3/2/21 8:00 AM</td>
<td>3/2/21 11:00 AM</td>
</tr>
<tr>
<td>12</td>
<td>GUI Design</td>
<td>10 days</td>
<td>0 days</td>
<td>4/4/21 8:00 AM</td>
<td>4/4/21 11:00 AM</td>
</tr>
<tr>
<td>13</td>
<td>Implementation</td>
<td>20 days</td>
<td>0 days</td>
<td>15/4/21 8:00 AM</td>
<td>5/5/21 11:00 AM</td>
</tr>
<tr>
<td>14</td>
<td>Coding module 1</td>
<td>15 days</td>
<td>3 days</td>
<td>15/4/21 8:00 AM</td>
<td>5/5/21 11:00 AM</td>
</tr>
<tr>
<td>15</td>
<td>Coding module 2</td>
<td>18 days</td>
<td>0 days</td>
<td>15/4/21 8:00 AM</td>
<td>5/5/21 11:00 AM</td>
</tr>
<tr>
<td>16</td>
<td>Coding module 3</td>
<td>13 days</td>
<td>0 days</td>
<td>15/4/21 8:00 AM</td>
<td>5/5/21 11:00 AM</td>
</tr>
<tr>
<td>17</td>
<td>Coding GUI</td>
<td>13 days</td>
<td>0 days</td>
<td>15/4/21 8:00 AM</td>
<td>5/5/21 11:00 AM</td>
</tr>
<tr>
<td>18</td>
<td>Documenting</td>
<td>2 days</td>
<td>0 days</td>
<td>4/3/21 8:00 AM</td>
<td>4/3/21 11:00 AM</td>
</tr>
<tr>
<td>19</td>
<td>Testing</td>
<td>5 days</td>
<td>0 days</td>
<td>13/5/21 8:00 AM</td>
<td>13/5/21 11:00 AM</td>
</tr>
<tr>
<td>20</td>
<td>Unit testing</td>
<td>3 days</td>
<td>0 days</td>
<td>13/5/21 8:00 AM</td>
<td>13/5/21 11:00 AM</td>
</tr>
<tr>
<td>21</td>
<td>Integration testing</td>
<td>5 days</td>
<td>0 days</td>
<td>13/5/21 8:00 AM</td>
<td>13/5/21 11:00 AM</td>
</tr>
<tr>
<td>22</td>
<td>System testing</td>
<td>5 days</td>
<td>0 days</td>
<td>13/5/21 8:00 AM</td>
<td>13/5/21 11:00 AM</td>
</tr>
<tr>
<td>23</td>
<td>Acceptance testing</td>
<td>3 days</td>
<td>0 days</td>
<td>13/5/21 8:00 AM</td>
<td>13/5/21 11:00 AM</td>
</tr>
<tr>
<td>24</td>
<td>Presentation</td>
<td>4 days</td>
<td>0 days</td>
<td>20/5/21 8:00 AM</td>
<td>20/5/21 11:00 AM</td>
</tr>
<tr>
<td>25</td>
<td>Team presentation</td>
<td>1 day</td>
<td>0 days</td>
<td>20/5/21 8:00 AM</td>
<td>20/5/21 11:00 AM</td>
</tr>
</tbody>
</table>

![Gantt Chart Image]
XVII. Project Demonstration
   a) Home Page

   ![Home Page Image]

   b) Register page:

   ![Register Page Image]
Login Page – Similar for all

Customer Page

Pending Deliveries

<table>
<thead>
<tr>
<th>Order ID</th>
<th>Category</th>
<th>Quantity</th>
<th>Price</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>199820202020</td>
<td>Wheat</td>
<td>5</td>
<td>75</td>
<td>PENDING</td>
</tr>
<tr>
<td>101029202020</td>
<td>Wheat</td>
<td>5</td>
<td>70</td>
<td>PENDING</td>
</tr>
</tbody>
</table>

Completed Deliveries

<table>
<thead>
<tr>
<th>Order ID</th>
<th>Category</th>
<th>Quantity</th>
<th>Price</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Disease Detection is a matlab code image processing code which uses multi class svm classifier to classify crops into diseases identified.


This is a view-only link of our crop disease detection using MATLAB. Anyone can view the folder on the internet. But they need permission to edit the file. Below you can find implementation of that. Its accuracy is 96.8%, we couldn’t show here because computing it is time taking process.
My Products

<table>
<thead>
<tr>
<th>Order ID</th>
<th>Category</th>
<th>Description</th>
<th>Quantity Available</th>
<th>Unit Price</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Pumpkin Sweats</td>
<td>Pumpkin Sweats - 500 per kg</td>
<td>10</td>
<td>500</td>
<td>Edit</td>
</tr>
<tr>
<td></td>
<td>Seeds</td>
<td>Rs. 50 per 10 packets</td>
<td>20</td>
<td>50</td>
<td>Edit</td>
</tr>
<tr>
<td></td>
<td>Sweats</td>
<td>Rs. 60 per 10 packets of each available 100 packets but should purchase in multiples of 10</td>
<td>10</td>
<td>60</td>
<td>Edit</td>
</tr>
<tr>
<td></td>
<td>Seeds</td>
<td>Rs. 60 per 10 packets</td>
<td>20</td>
<td>50</td>
<td>Edit</td>
</tr>
</tbody>
</table>

Add Product

[Form fields for product details]

Add Product
<table>
<thead>
<tr>
<th>Vehicle</th>
<th>Description</th>
<th>Quantity Available</th>
<th>Shift Base Price</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tractor</td>
<td></td>
<td>10</td>
<td>5000</td>
<td>Edit, Delete</td>
</tr>
<tr>
<td>Container</td>
<td></td>
<td>10</td>
<td>50000</td>
<td>Edit, Delete</td>
</tr>
<tr>
<td>Container</td>
<td></td>
<td>15</td>
<td>50000</td>
<td>Edit, Delete</td>
</tr>
</tbody>
</table>
Above is our online agent AIMUS, an agent of Zendesk. Below is our own IBM Waston Virtual Assistant named Farmastant Ai1.
XVIII. Results and Discussion

Thus, our website FARMASTANT successfully helps in:

i) User Registration and successful login of all users.

ii) Customer purchasing desired products.

iii) Farmer checking, adding products and delivering orders.

iv) Weather Forecast for next 5 days.

v) Crop disease detection.

vi) Merchant selling goods to farmer.

vii) Shipping products/goods to customer/farmer respectively.

viii) Getting to know about Farmastant and its creators us.

ix) Contact us through a form submission.

x) Chat with our IBM Watson Assistant Farmastant Ai1.

xi) Get connected to our desk agent AIMUS at Zendesk.

XIX. SUMMARY

Our project FARMASTANT is an agricultural E-Commerce website which helps all users to buy/sell agricultural related products depending on their role in the context of farming and marketing. The actors are Customers, Farmers, Merchants, Shipping services and Customer Care Agents. We provided unique features like Weather Report, Crop Disease Detection, chat with our Zendesk agent AIMUS and AI trained Virtual Assistant Chatbot named Farmastant Ai1. Customers can buy required products and just checkout with address and farmers will order transportation from shipping services to deliver customer orders to their address. Farmers can buy required farming merchandize from merchants and get them delivered to their door step by placing order to shipping services. Merchants sell farming merchandize to farmers. Shipping services provide transportation for all to deliver products/goods to customers/farmers respectively. Users can chat with our Virtual Assistant Ai1. In case of queries they can submit a form or they can talk to our online desk agent AIMUS at Zendesk.

XX. References


