DBMS LAB

Web Project

AGroculture

INTRODUCTION

Discover the future of farming with Agro Culture a cuttingedge platform designed to revolutionize the way farmers do business. Say goodbye to the days of manual recordkeeping and scattered information. Agro Culture brings together all the elements of a successful farming operation into one centralized system. With this innovative technology, farmers can plan, monitor, and analyze their production with ease. Whether you're a seasoned pro or just starting out, Agro Culture empowers you to make smarter decisions and boost your productivity.

But Agro Culture isn't just about making life easier for farmers. It's about building a community of collaboration and transparency. By partnering with growers and buyers, Agro Culture inspires farmers to produce the freshest, highest quality goods, and makes it easier for buyers to access those goods. It's time to take your farming operation to the next level with Agro Culture.

Objectives of the Project

The specific objectives of the project include:

- To provide qualitative foods to the buyers.
- Implementing an automated/online agro culture system.
- To inspire farmer to produce quality goods and supply to the buyers.
- Ecofriendly farming system.

Scope of the Project

- It is focused on studying the existing system of agro culture in and to make sure that the peoples are getting quality fresh goods.

This is also will produce:

- Less effort and less labor 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.

BACKGROUND

Fronted technologies and Frameworks

1.HTML5

2.CSS/Bootstrap

Backend technologies

1.PHP

2.SQL

HTML (Hypertext Markup Language):

- The standard markup language used for creating web pages and web applications.
- Used to structure content on the web, such as text, images, and links.

CSS (Cascading Style Sheets):

- A stylesheet language used to define the look and feel of a website.
- Used to apply styles, such as font, color, and spacing, to HTML content.

Bootstrap:

- A free frontend framework for faster and easier web development.
- Uses HTML, CSS, and JavaScript to create responsive and mobilefirst web pages.

PHP (Hypertext Preprocessor):

- A serverside scripting language used to develop dynamic web pages and applications.
- Used to process usergenerated data and connect to databases, such as SQL.

SQL (Structured Query Language):

- A standard language used to manage relational databases.
- Used to create, read, update, and delete data stored in databases.

METHODOLOGY

For the successful development of our project, the database design was meticulously executed using PHPMYADMIN, while the backend was coded using the robust PHP programming language. To ensure a seamless user experience, the frontend was also developed using PHP codes.

We utilized a systematic approach to software development by employing various software methodologies. These methodologies focus on the process of creating software, from the organizational and project management aspects, rather than just the technical side. By utilizing proven software development approaches, we were able to deliver a highquality and reliable end product that meets the needs of our stakeholders

TOOLS USED

1. Xampp A popular, free and opensource web server solution that includes Apache, PHP and MySQL.
2. Apache An opensource Java Servlet Container that serves as the application server.
3. MySQL Server A highperformance database management system that handles large databases efficiently.
4. Sublime Text 3.1.1 A sophisticated text editor for code, markup and prose with a sleek user interface and exceptional performance.
5. Web Browsers Google Chrome, Mozilla Firefox, Opera and Internet Explorer were used for testing and debugging purposes.

FUNCTIONAL REQUIREMENTS

1. Farmer Registration: Farmers should be able to create their own account by providing their contact information, location, and details about their farm and the products they supply.
2. Product Management: Farmers should be able to add, edit, and delete information about the fruits and vegetables they supply, including the type, quantity, and price.
3. Profile Management: Farmers should be able to update and manage their personal and farm information, such as contact information, farm description, and product details.
4. Product Display: Farmers should be able to showcase their products to potential buyers, with detailed information such as the type of product, quantity, and price.
5. User Login: Farmers should be able to log in to their account to access their profile and product information.
6. Search Functionality: Buyers should be able to search for farmers and their products based on location, type of product, and other relevant criteria.

CODE

SQL

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";

SET time_zone = "+00:00";

Database: `agroculture`
```

Table structure for table `blogdata`

```
CREATE TABLE `blogdata` (
  `blogId` int(10) NOT NULL,
  `blogUser` varchar(256) NOT NULL,
  `blogTitle` varchar(256) NOT NULL,
  `blogContent` longtext NOT NULL,
  `blogTime` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `likes` int(10) NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table structure for table `blogfeedback`

```
 CREATE TABLE `blogfeedback` (
  `blogId` int(10) NOT NULL,
  `comment` varchar(256) NOT NULL,
  `commentUser` varchar(256) NOT NULL,
  `commentPic` varchar(256) NOT NULL DEFAULT 'profile0.png',
  `commentTime` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Table structure for table `farmer`

```
CREATE TABLE `farmer` (
  `fid` int(255) NOT NULL,
  `fname` varchar(255) NOT NULL,
  `fusername` varchar(255) NOT NULL,
  `fpassword` varchar(255) NOT NULL,
  `fhash` varchar(255) NOT NULL,
  `femail` varchar(255) NOT NULL,
  `fmobile` varchar(255) NOT NULL,
  `faddress` text NOT NULL,
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Dumping data for table `farmer`

```
INSERT INTO `farmer` (`fid`, `fname`, `fusername`, `fpassword`, `fhash`, `femail`, `fmobile`, `faddress` ) VALUES
(1 'Chandran , 'chandu', 'Chandu123@341, '61b4a64be663682e8cb037d9719ad8cd', 'chandran123@gmail.com', '9090898978','Prasanth Bhavanam');
```

Table structure for table `fproduct`

```
CREATE TABLE `fproduct` (
  `fid` int(255) NOT NULL,
  `pid` int(255) NOT NULL,
```

```
  `product` varchar(255) NOT NULL,
  `pcat` varchar(255) NOT NULL,
  `pinfo` varchar(255) NOT NULL,
  `price` float NOT NULL,
  `pimage` varchar(255) NOT NULL DEFAULT 'blank.png',
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Dumping data for table `fproduct`

```
INSERT INTO `fproduct` (`fid`, `pid`, `product`, `pcat`, `pinfo`, `price`, `pimage`, `picStatus`) VALUES
(3, 27, 'Mango', 'Fruit', '<p>Mango</p>\r\n', 500, 'Mango3.jpeg' ),
(3, 28, 'Ladyfinger', 'Vegetable', '<p>Its veggie</p>\r\n', 1000, 'Ladyfinger3.jpg'),
(3, 29, 'Bajra', 'Grains', '<p>bajrei</p>\r\n', 400, 'Bajra3.jpg' ),
(3, 30, 'Banana', 'Fruit', '<p>Mysore  banana</p>\r\n', 400, 'Banana3.jpg');
```

Table structure for table `review`

```
CREATE TABLE `review` (
  `pid` int(10) NOT NULL,
  `name` varchar(255) NOT NULL,
  `rating` int(10) NOT NULL,
  `comment` text NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Indexes for table `blogdata`

```
ALTER TABLE `blogdata`
  ADD PRIMARY KEY (`blogId`);
```

Indexes for table `farmer`

```
ALTER TABLE `farmer`
  ADD PRIMARY KEY (`fid`),
  ADD UNIQUE KEY `fid` (`fid`);
```

Indexes for table `fproduct`

```
ALTER TABLE `fproduct`
  ADD PRIMARY KEY (`pid`);
```

AUTO_INCREMENT for dumped tables

AUTO_INCREMENT for table `blogdata`

```
 ALTER TABLE `blogdata`
   MODIFY `blogId` int(10) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=20;
```

AUTO_INCREMENT for table `farmer`

```
ALTER TABLE `farmer`
  MODIFY `fid` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;
```

AUTO_INCREMENT for table `fproduct`

```
 ALTER TABLE `fproduct`
   MODIFY `pid` int(255) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=31;
```

Table structure for table `likedata`

```
CREATE TABLE `likedata` (
  `blogId` int(10) NOT NULL,
  `blogUserId` int(10) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Dumping data for table `likedata`

```
INSERT INTO `likedata` (`blogId`, `blogUserId`) VALUES
(19, 3);
```

Indexes for table `likedata`

```
ALTER TABLE `likedata`
  ADD KEY `blogId` (`blogId`),
  ADD KEY `blogUserId` (`blogUserId`);
```

Constraints for table `likedata`

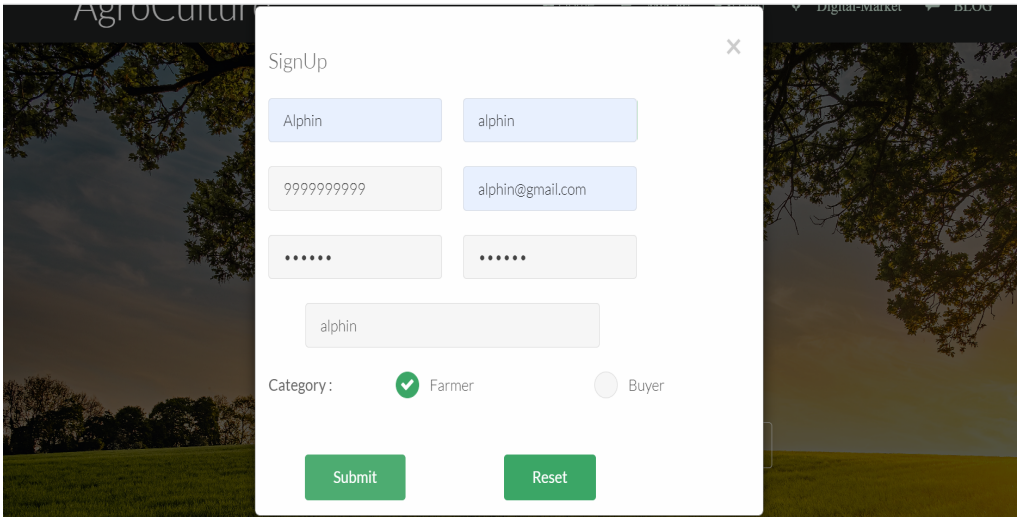
```
ALTER TABLE `likedata`
  ADD CONSTRAINT `likedata_ibfk_1` FOREIGN KEY (`blogId`) REFERENCES `blogdata` (`blogId`);

ALTER TABLE `fproduct`
  ADD CONSTRAINT `farmerid` FOREIGN KEY (`fid`) REFERENCES `farmer` (`fid`);
```

PHP CONNECTION TO DATABASE

```
<?php
   $serverName = "localhost";
   $userName = "root";
   $password = "";
   $dbName = "agroculture";
   $conn = mysqli_connect($serverName, $userName, $password, $dbName);
   if (!$conn)
   {
      die("Connection failed: " . mysqli_connect_error());
   }
?>
```

RESULT



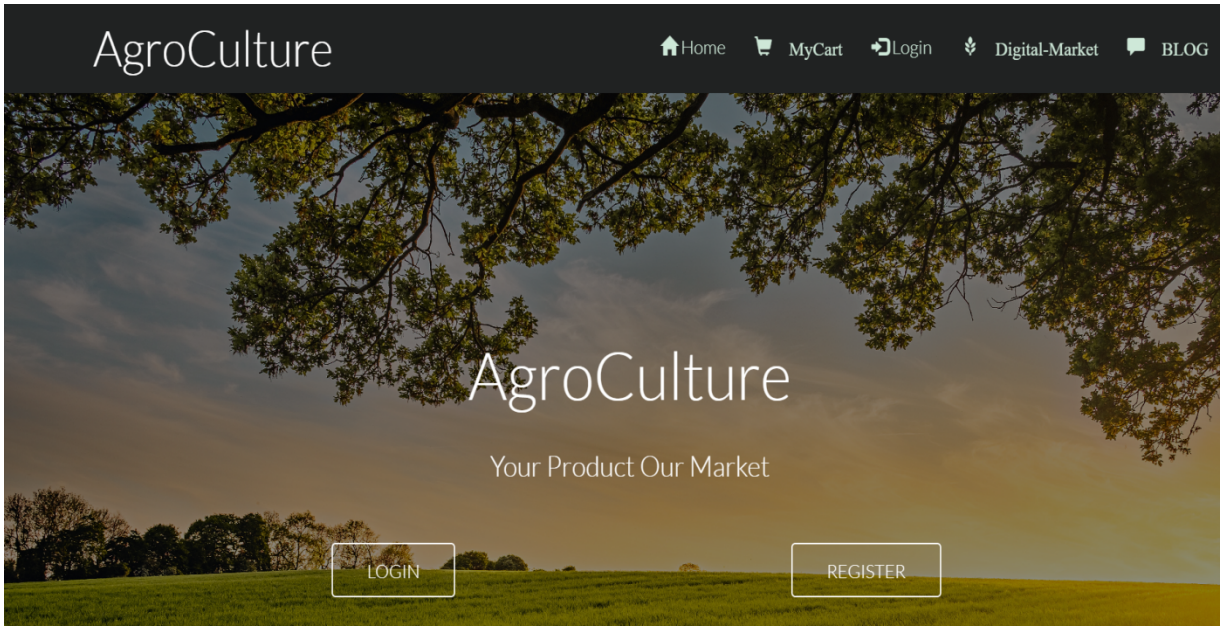


Fig **Home Page**

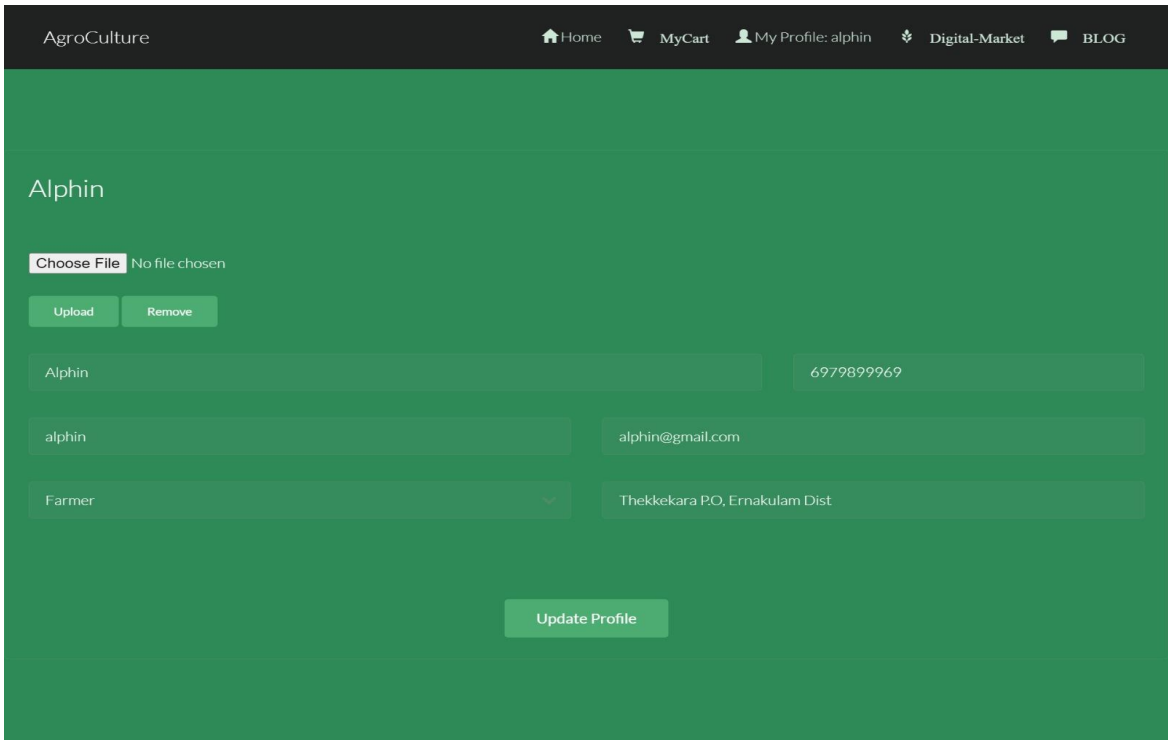


Fig Farmer Profile

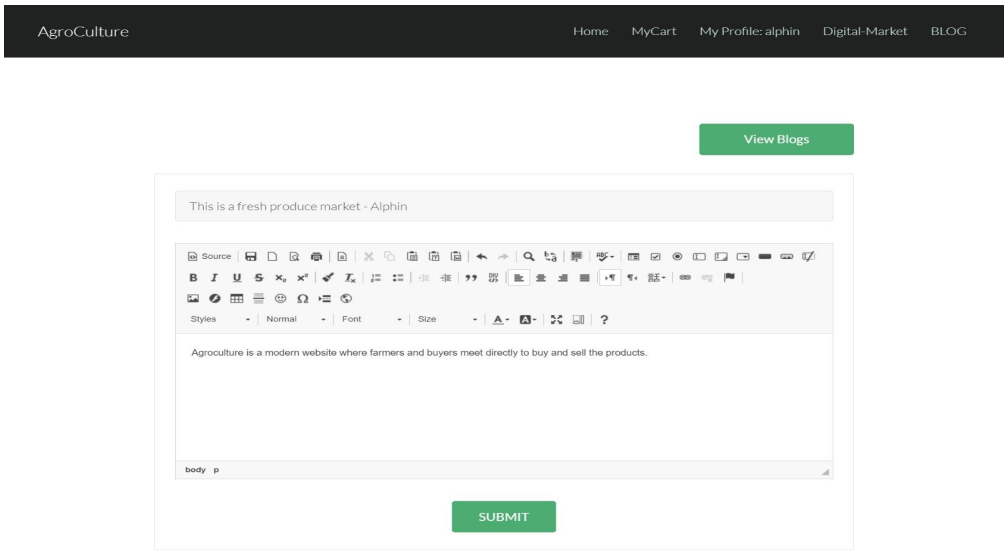


Fig product review

Fig Product Inclusion

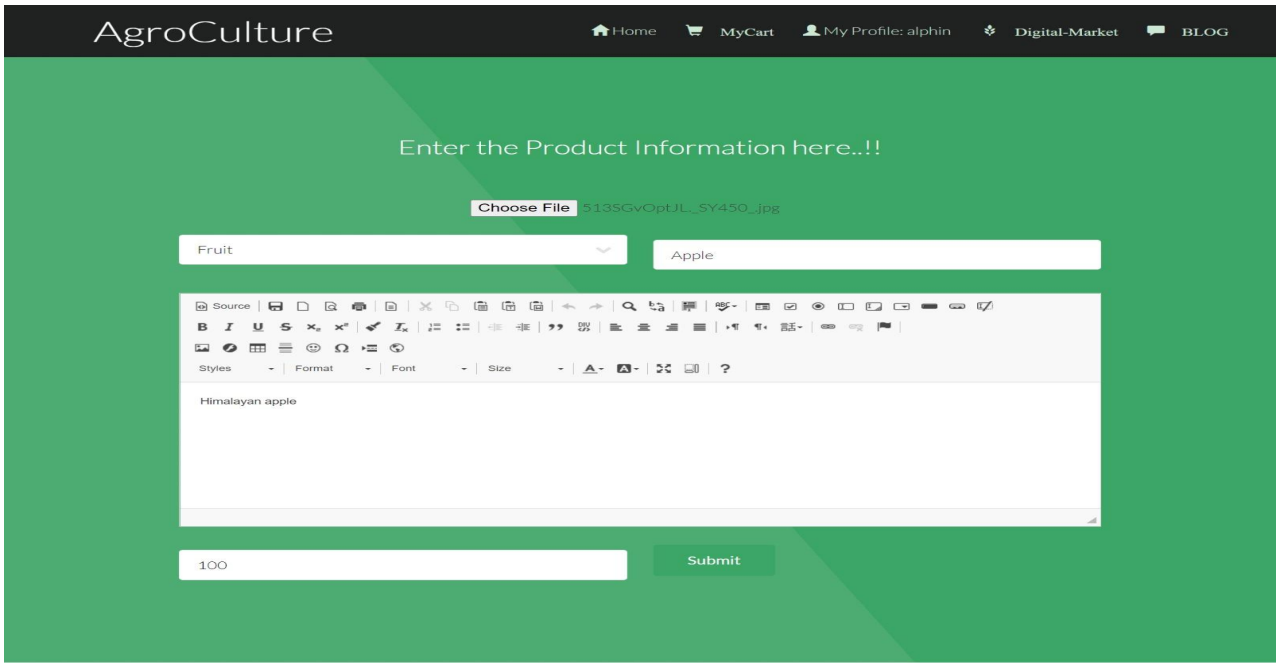
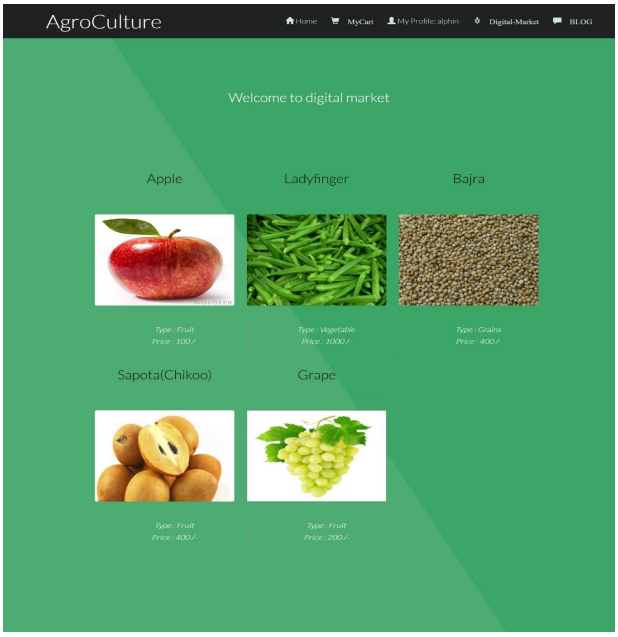
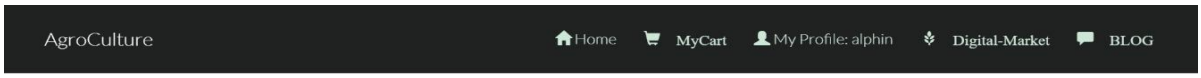


Fig Product Display





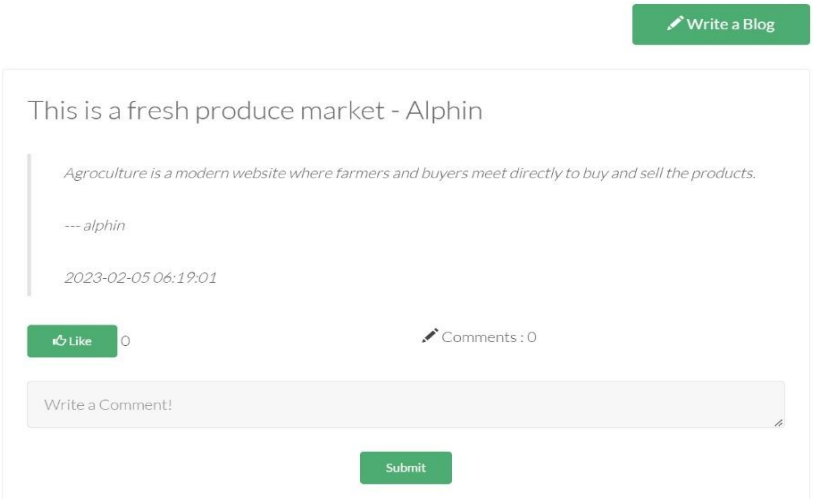
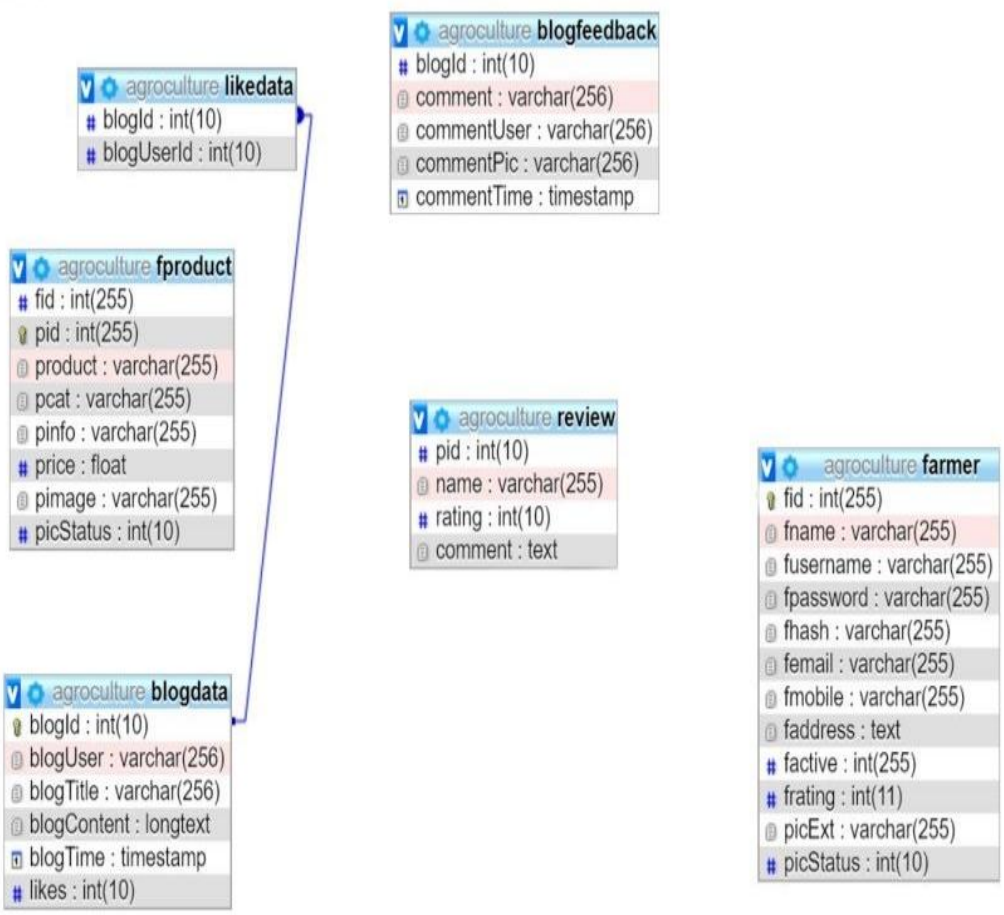


fig:blog

ER Diagram


agroculture likedata
blogId : int(10)
blogUserId : int(10)
agroculture blogfeedback
blogId : int(10)
comment : varchar(256)
commentUser : varchar(256)
commentPic : varchar(256)
commentTime : timestamp
agroculture fproduct
fid : int(255)
pid : int(255)
product : varchar(255)
pcat : varchar(255)
pinfo : varchar(255)
price : float
pimage : varchar(255)
picStatus : int(10)
agroculture review
pid : int(10)
name : varchar(255)
rating : int(10)
comment : text
agroculture farmer
fid : int(255)
fname : varchar(255)
fusername : varchar(255)
fpassword : varchar(255)
fhash : varchar(255)
femail : varchar(255)
fmobile : varchar(255)
faddress : text
factive : int(255)
frating : int(11)
picExt : varchar(255)
picStatus : int(10)
agroculture blogdata
blogId : int(10)
blogUser : varchar(256)
blogTitle : varchar(256)
blogContent : longtext
blogTime : timestamp
likes : int(10)

CONCLUSION

In conclusion, Agro Culture is a farmer system designed to help farmers plan, monitor, and analyze their production activities. With this system, farmers can manage their operations and organize their data in one place. By connecting farmers , Agro Culture aims to increase efficiency and standardize the agro culture process, ensuring quality food for consumers. The website features functionalities such as farmer registration, product management, profile management, product display, user login, search functionality, order management and a user-friendly interface. Agro Culture is a promising solution for farmers , providing a platform for sustainable agriculture and supporting the growth of the agriculture industry.