

1NH19CS139_Farmers Coalition

by Rajesh G

Submission date: 21-Jun-2022 02:57PM (UTC+0530)

Submission ID: 1860679386

File name: 1NH19CS139_Rajesh_G.pdf (1.15M)

Word count: 4358

Character count: 22139

Name: Rajesh G

USN:1NH19CS139

SECTION:C

SEM:6th

CHAPTER 1

INTRODUCTION

1.1 PROBLEM DEFINITION

In todays world farmers used to grow the products for such type of products not getting the best and worth value for that so to make the products more valuable . And even they can get the right information with respect to the weather or the products information and they can write the blog about the product to improve their product well and good.

1.1.1 OBJECTIVES

The objectives of the projects include but are not limited to any type of users any one can login those having the product they can take the pic of it and they can upload the product on the website .where they have the good value for it. They can improve their profile with having some of the updates and make it more user friendly and efficient one.

1.1.2 METHODOLOGY TO BE FOLLOWED

The methodology to be followed by the project is to use HTML (Hyper Text Markup Language), CSS (Cascading Style Sheets) for designing the front end of the web application along with JavaScript to add functionalities to the web pages.

1.1.3 EXPECTED OUTCOMES

Farmers Coalition

The expected outcomes are:

- Login page for the Farmers.
- Farmers can access all the active products information at once
- Add to cart and they can order the products.
- They can write the review and blog for a particular product
- They can update the profile and upload the product and they can write something for the product.

CHAPTER 2

FUNDAMENTALS OF WEB DEVELOPMENT

2.1 INTRODUCTION TO WEB

The idea of the internet had existed in some form for at least a half a century before it finally became a common household utility in the 1990s. Conceived in the 1980s, the World Wide Web gained significant traction with the introduction of the Mosaic browser in 1993. In 1989, Tim Berners-Lee outlined his concept of a computer platform that could facilitate collaboration among researchers who are based in different parts of the world. This led to the invention of the Hypertext Markup Language (HTML) in 1990. Strongly based on the Standard Generalized Markup Language (SGML), HTML became the fundamental building block of the World Wide Web, and remains at the core of its coding and infrastructure. The standard enabled coders with the ability to organize web page layouts that could be understood and interacted with over interconnected networks.

As hardware improvements cultivated broader networks and greater bandwidth, web development responded by enabling designers with an array of multimedia to incorporate into the growing and diversifying art of web presentation. Cascading Style Sheets (CSS) afforded web design with new ways to organize and display content. Flash video forged a new and entirely unique style of web art and animation, and video streaming changed the way that people consume motion picture for good. Yet still, with all of these revolutions and progressions in web development – the basic interface and structure of the web page has maintained its integrity and balance of form and function.

There are three types of Web Development:

- **Front-End Development:** Front-end web development is the development of the graphical user interface of a website, through the use of HTML, CSS, and JavaScript, so that users can view and interact with that website.
- **Back-End Development:** Backend developers build code that allows a database and an application to communicate with one another. Backend developers take care and maintain the back-end of a website, including databases, servers, and apps, and they control what you don't see.
- **Full Stack Development:** A full stack web developer is a person who can develop both client and server software. In addition to mastering HTML and CSS, the full stack developer also knows how to use, program a browser (like using JavaScript, jQuery, Angular, or Vue), Program a server (like using PHP, ASP, Python, or Node), Program a database (like using SQL, SQLite, or MongoDB)

2.1.1 HTML

3

HTML (Hyper Text Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behavior (JavaScript).

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web. HTML uses "markup" to annotate text, images, and other content for display in a Web browser. All the tags have to be opened and closed. The elements should be contained inside the opening and the closing of a particular tag.

XHTML stands for Extensible Hyper Text Markup Language. It is a cross between HTML and XML language. XHTML is almost identical to HTML but it is stricter than HTML. XHTML is HTML defined as an XML application. It is supported by all major browsers.

Although XHTML is almost the same as HTML but It is more important to create your code correctly, because XHTML is stricter than HTML in syntax and case sensitivity. XHTML documents are well-formed and parsed using standard XML parsers, unlike HTML, which requires a lenient HTML-specific parser.

Tag	Description
<html> ... </html>	Declares the Web page to be written in HTML
<head> ... </head>	Delimits the page's head
<title> ... </title>	Defines the title (not displayed on the page)
<body> ... </body>	Delimits the page's body
<h _n > ... </h _n >	Delimits a level <i>n</i> heading
 ... 	Set ... in boldface
<i> ... </i>	Set ... in italics
<center> ... </center>	Center ... on the page horizontally
 ... 	Brackets an unordered (bulleted) list
 ... 	Brackets a numbered list
 ... 	Brackets an item in an ordered or numbered list
 	Forces a line break here
<p>	Starts a paragraph
<hr>	Inserts a horizontal rule
	Displays an image here
 ... 	Defines a hyperlink

Figure2.1.1:Basic HTML tags

DHTML stands for Dynamic Hypertext Markup language i.e., Dynamic HTML. Dynamic HTML is not a markup or programming language but it is a term that combines the features of various web development technologies for creating the web pages dynamic and interactive.

2.1.1.1 Components of Dynamic HTML

DHTML consists of the following four components or languages:

- HTML: HTML is a client-side markup language, which is a core component of the DHTML. It defines the structure of a web page with various defined basic elements or tags.
- CSS: CSS stands for Cascading Style Sheet, which allows the web users or developers for controlling the style and layout of the HTML elements on the web pages.

3

- JavaScript: JavaScript is a scripting language which is done on a client-side. The various browser supports JavaScript technology. DHTML uses the JavaScript technology for accessing, controlling, and manipulating the HTML elements. The statements in JavaScript are the commands which tell the browser for performing an action.
- DOM: DOM is the document object model. It is a w3c standard, which is a standard interface of programming for HTML. It is mainly used for defining the objects and properties of all elements in HTML.

2.1.2 CSS

CSS stands for ¹ Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. ¹ CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, which reduces complexity and repetition in the structural content; and enable the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

In CSS, selectors are used to select the particular data, tags for which the particular block of rules are set. The below table gives a quick overview of the types of selectors and the way to access them.

	attribute	Symbol used
1	Using id	use # symbol
2	Using class name	use . symbol
3	Using attribute	tagname[attribute='value']
4	Using multiple attribute	tagname[attribute1='value1'] [attribute2='value2']
5	Contains	* symbol
6	Starts with	^ symbol
7	Ends with	\$ symbol

Figure 2.1.2 Types of CSS Selectors

8

2.1.3 JAVASCRIPT

JavaScript often abbreviated JS, is a programming language that is one of the core technologies of the World Wide Web, alongside HTML and CSS. Over 97% of websites use JavaScript on the client side for web page behavior, often incorporating third-party libraries. All major web browsers have a dedicated JavaScript engine to execute the code on users' devices. JavaScript is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpages in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses js to provide several forms of interactivity and simplicity. Although, JavaScript has no connectivity with Java programming language. The name was suggested and provided in the times when Java was gaining popularity in the market. In addition to web browsers, databases such as CouchDB and MongoDB uses JavaScript as their scripting and query language.

2.1.3.1 FEATURES OF JAVASCRIPT

There are following features of JavaScript:

1. All popular web browsers support JavaScript as they provide built-in execution environments.
2. JavaScript follows the syntax and structure of the C programming language. Thus, it is a structured programming language.
3. JavaScript is a weakly typed language, where certain types are implicitly cast (depending on the operation).
4. JavaScript is an object-oriented programming language that uses prototypes rather than using classes for inheritance.
5. It is a light-weighted and interpreted language.
6. It is a case-sensitive language.
7. JavaScript is supportable in several operating systems including, Windows, macOS, etc.
8. It provides good control to the users over the web browsers.

2.1.4 PHP

² PHP is a general-purpose scripting language geared toward web development. It was originally created by Danish-Canadian programmer Rasmus Lerdorf in 1994. The PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive initialism PHP: Hypertext Preprocessor.

PHP code ¹ is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code – which may be any type of data, such as generated HTML or binary image data – would form the whole or part of an HTTP response. ² Various web template systems, web content management systems, and web frameworks exist which can be employed to orchestrate or facilitate the generation of that response. Additionally, PHP can be used for many programming tasks outside the

web context, such as standalone graphical applications and robotic drone control. PHP code can also be directly executed from the command line.

4 The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on a variety of operating systems and platforms.

2 The PHP language evolved without a written formal specification or standard until 2014, with the original implementation acting as the de facto standard which other implementations aimed to follow. Since 2014, work has gone on to create a formal PHP specification.

2.1.4.1 PHP 7

During 2014 and 2015, a new major PHP version was developed, PHP 7. The numbering of this version involved some debate among internal developers. While the PHP 6 Unicode experiment had never been released, several articles and book titles referenced the PHP 6 name, which might have caused confusion if a new release were to reuse the name. After a vote, the name PHP 7 was chosen.

The foundation of PHP 7 is a PHP branch that was originally dubbed PHP next generation (phpng). It was authored by Dmitry Stogov, Xinchen Hui and Nikita Popov, and aimed to optimize PHP performance by refactoring the Zend Engine while retaining near-complete language compatibility. By 14 July 2014, WordPress-based benchmarks, which served as the main benchmark suite for the phpng project, showed an almost 100% increase in performance. Changes from phpng make it easier to improve performance in future versions, as more compact data structures and other changes are seen as better suited for a successful migration to a just-in-time (JIT) compiler. Because of the significant changes, the reworked Zend Engine was called Zend Engine 3, succeeding Zend Engine 2 used in PHP 5.

Because of the major internal changes in phpng, it must receive a new major version number of PHP, rather than a minor PHP 5 release, according to PHP's release process.

Major versions of PHP are allowed to break backward-compatibility of code and therefore PHP 7 presented an opportunity for other improvements beyond phpng that require backward-compatibility breaks. In particular, it involved the following changes:

- Many fatal or recoverable-level legacy PHP error mechanisms were replaced with modern object-oriented exceptions.
- The syntax for variable dereferencing was reworked to be internally more consistent and complete, allowing the use of the operators `->`, `[]`, `()`, `{}`, and `::`, with arbitrary meaningful left-side expressions.
- Support for legacy PHP 4-style constructor methods was deprecated.
- The behavior of the `foreach` statement was changed to be more predictable.
- Constructors for the few classes built-in to PHP which returned null upon failure were changed to throw an exception instead, for consistency.
- Several unmaintained or deprecated server application programming interfaces (SAPIs) and extensions were removed from the PHP core, most notably the legacy MySQL extension.
- The behavior of the `list()` operator was changed to remove support for strings.
- Support was removed for legacy ASP-style delimiters `<%` and `%>` and `<script language="php"> ... </script>`.
- An oversight allowing a `switch` statement to have multiple default clauses was fixed.
- Support for hexadecimal number support in some implicit conversions from strings to number types was removed.
- The left-shift and right-shift operators were changed to behave more consistently across platforms.
- Conversions between floating-point numbers and integers were changed (e.g., infinity changed to convert to zero) and implemented more consistently across platforms.
- PHP 7 also included new language features. Most notably, it introduced return type declarations for functions which complement the existing parameter type

declarations, and support for the scalar types (integer, float, string, and Boolean) in parameter and return type declarations.

W3Techs reports that, as of January 2022, "PHP is used by 78.1% of all the websites whose server-side programming language we know." PHP version 7.4 is the most used version. Support for version 7.3 was dropped on 6 December 2021.

2.2 FUNDAMENTALS OF DBMS

DBMS is database management system. Databases are the collection of data in order to store and retrieve data. The database consists of data which can be a numeric, alphabetic and also alphanumeric form. Analysing data is a key feature of database management system that is DBMS. ⁹ DBMS allows the definition, creation, querying, update, and administration of databases. Language supported and widely used for querying and accessing the database is SQL.

Some of the common terminologies of DBMS are

- Tuple: The rows in the database are often known as tuples.
- Table: Table is a collection of tuples and related information along with a key to distinguish the data. Although a table can have duplication of data tuples.
- Schema: Schema is the structure of the relation or a table.
- Data redundancy: Data redundancy ensures there are no multiple occurrences of same data hence avoids data duplication.
- Keys: Keys in a table are used to identify the unique attribute of the table.
- Entity: An entity can be any real life physical or non-physical object which has some attributes.
- Attributes: Attributes are the names of the columns or the defining features of the entity.

A DBMS follows 3-schema architecture. The three-schema architecture divides the database into three-level used to create a separation between the physical database and

the user application. In simple terms, this architecture hides the details of physical storage from the user.

A 3-schema architecture consists of 3 level namely:

- External Schema: This is the highest level of database abstraction. It includes a number of external schemas or user views. This level provides different views of the same database for a specific user or a group of users. An external view provides a powerful and flexible security mechanism by hiding the parts of the database from a particular user. This schema is useful for user with no knowledge on database or its workings. It's a layer which can be easily understood by anyone.
- Conceptual Schema: This level describes the structure of the whole database. It acts as a middle layer between the physical storage and user view. It explains what data to be stored in the database, what the data types are, and what relationship exists among those data. There is only one conceptual schema per database. This schema is useful for programmers that use the database regularly.
- Internal Schema: This is the lowest level of database abstraction. It describes how the data is stored in the database and provides the methods to access data from the database. It allows viewing the physical representation of the database on the computer system. This level is useful for people who manage the database and know its internal working.

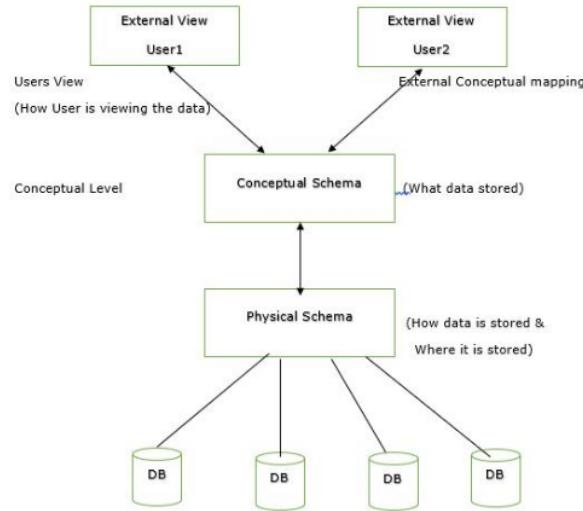


Fig 2.2.13-schema architecture

2.2.1 FUNDAMENTALS OF SQL

SQL (Structured Query Language) is a standardized programming language that's used to manage relational databases and perform various operations on the data in them. The uses of SQL include modifying database table and index structures; adding, updating and deleting rows of data; and retrieving subsets of information from within a database for transaction processing and analytics applications. Queries and other SQL operations take the form of commands written as statements. SQL accepts only structured data in a table form also called as RDBMS (Relational Database Management System). SQL also follows ACID properties where ACID is an acronym for (Atomicity, Consistency, Isolation and Durability). There also exists another type of query language called NoSQL which takes non-relational data also and provides much more flexibility than SQL. The queries in SQL are written in the form of commands.

In DBMS the SQL query language has DML, DDL, DCL, and TCL.

- DML is data manipulation language and is used for selecting, retrieving, storing, modifying, deleting, inserting and updating entries in the database. SELECT, UPDATE, INSERT, DELETE are some of the DML query statements

Farmers Coalition

eg: SELECT *; this statement will select all the values and tuple from the database and display them as an output of this query

- DDL is data definition language and is useful for defining the schema and structure of the database. Commands like DROP, CREATE, ALTER, TRUNCATE, COMMENT, and RENAME are used.

eg: DROP *table name*; this statement will delete the values as well as the structure of the database.

- DCL is data control language and is useful for granting and revoking rights to and from a user. The command like GRANT and REVOKE are used.

eg: GRANT SELECT to *username*; this statement will grant or allow the user to select the data from the database.

- TCL is transaction control language and is useful for managing the transaction in the database. Commands like COMMIT, ROLLBACK, SAVEPOINT and SET TRANSACTION are used.

10

2.2.2 ENTITY-RELATIONSHIP (ER) MODEL

The Entity-Relationship (ER) Model is an attractive high level conceptual data model. It has an entity which may be an object with a physical existence like a particular car, house, person or employee or it may be an object with a conceptual existence like an organization, a profession, or a university course. Each entity has attributes—the definite properties that characterize it. For example, a student entity may be described by the student's name, age, address, USN etc.

5

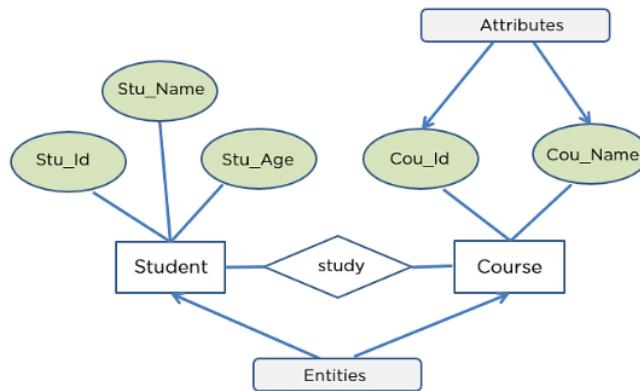


Figure 2.2.2.1 A Sample ER Model for Student Database

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 HARDWARE REQUIREMENTS

1. Processor: Intel core i3 10th gen

2. RAM: 2GB or more.

OS: Windows 10

3.2 SOFTWARE REQUIREMENTS

1. Any text editor
2. Web Browser
3. MySQL / Sqlite3 database

CHAPTER 4

DESIGN

4.1 DESIGN GOALS

The main goal of the project is to get the information of the farmers through the login and they should get good price whatever they have grown so we are creating the so for that here we are using the javascript and bootstrap for the framework to be make easy and responsive front-end part. Since it supports all type of web browsers so we are using the bootstrap and it even establishes the consistency throughout the project.

Farmers can login to the website where if they have account already they can login and they can upload their product and can gain the information with the help of the reviews.

4.2 ALGORITHM

Step 1: Create the home page displaying the login and register forms.

Step 2: Create login page, signup page for farmers and buyers.

Step 3: After login they are able to view some of the options such as change password.

Step 4: After they login they are able to upload the products of the one who has.

Step 5: Once registered, the farmers are not able to register again.

Step 6: They can write the review for the particular product and they can add to cart of that product.

Step 7: After adding to the cart they are able to order that particular product.

4.3 DATABASE DESIGN

The below diagram shows the tables used in creating the project.

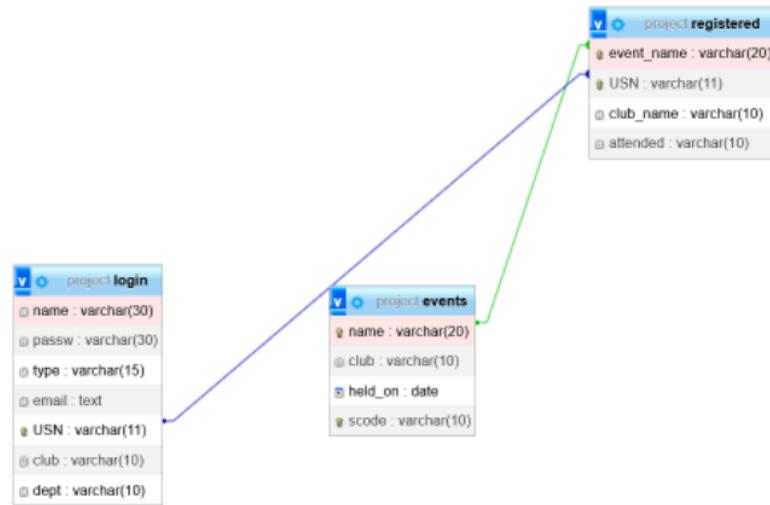


Figure 4.3.1 Design of the database “project”

CHAPTER 5

IMPLEMENTATION

5.1 MODULE 1 FUNCTIONALITY

The first and foremost module is the database connection which is done in this project using mysqli. The database is created in XAMPP Server using phpMyAdmin tool. Then the following code is used to connect to the database and run the queries as required. In this project Object Oriented approach has been used to connect to database.

```
File Edit Format View Help
<?php

$serverName = "localhost";
$userName = "root";
$password = "";
$dbName = "agroculture";

$conn = mysqli_connect($serverName, $userName, $password, $dbName);
if (!$conn)
{
    die("Connection failed: " . mysqli_connect_error());
}

?>
```

Fig 5.1.1 Database Connection using PHP

5.2 MODULE 2 FUNCTIONALITY

Once the database is successfully connected, we can then create a login form for the user to login. Bootstrap, HTML have been used in this project to create the login page and an external stylesheet has been incorporated for the login page. Session has been created upon successful login containing the details of the logged in user which will be useful throughout the project for various purposes. The below code snippet is used for login.

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

if($num_rows == 0)
{
    $_SESSION['message'] = "Invalid User Credentialss!";
    header("location: error.php");
}
```

Figure 5.1.2 Session and Bootstrap

```
function dataFilter($data)
{
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
```

Figure 5.1.3 Setting Session Variables

```
if($num_rows == 0)
{
    $_SESSION['message'] = "Invalid User Credentialss!";
    header("location: error.php");
}

else
{
    $User = $result->fetch_assoc();

    if (password_verify($_POST['pass'], $User['bpassword']))
    {
        $_SESSION['id'] = $User['bid'];
        $_SESSION['Hash'] = $User['bhash'];
        $_SESSION['Password'] = $User['bpassword'];
        $_SESSION['Email'] = $User['bemail'];
        $_SESSION['Name'] = $User['bname'];
        $_SESSION['Username'] = $User['busername'];
        $_SESSION['Mobile'] = $User['bmobile'];
        $_SESSION['Addn'] = $User['baddress'];
        $_SESSION['Active'] = $User['bactive'];
        $_SESSION['logged_in'] = true;
        $_SESSION['Category'] = 0;
    }
}
```

Figure 5.1.4 Login Page Code Snippet

5.3 MODULE 3 FUNCTIONALITY

Before logging in, a new user has to sign-up using the signup option provided by the login page .Each type of account will get different pages and accessibilities of content. The below code is responsible for signing a new user.

```
if ($_SERVER["REQUEST_METHOD"] == "POST")
{
    $name = dataFilter($_POST['name']);
    $mobile = dataFilter($_POST['mobile']);
    $user = dataFilter($_POST['uname']);
    $email = dataFilter($_POST['email']);
    $pass = dataFilter(password_hash($_POST['pass'], PASSWORD_BCRYPT));
    $hash = dataFilter( md5( rand(0,1000) ) );
    $category = dataFilter($_POST['category']);
    $addr = dataFilter($_POST['addr']);

    $_SESSION['Email'] = $email;
    $_SESSION['Name'] = $name;
    $_SESSION['Password'] = $pass;
    $_SESSION['Username'] = $user;
    $_SESSION['Mobile'] = $mobile;
    $_SESSION['Category'] = $category;
    $_SESSION['Hash'] = $hash;
    $_SESSION['Addr'] = $addr;
    $_SESSION['Rating'] = 0;
```

Figure 5.3.1 Bootstrap import for Styling

```
if (mysqli_query($conn, $sql))
{
    $_SESSION['Active'] = 0;
    $_SESSION['logged_in'] = true;

    $_SESSION['picStatus'] = 0;
    $_SESSION['picExt'] = png;

    $sql = "SELECT * FROM farmer WHERE fusername='$user'";
    $result = mysqli_query($conn, $sql);
    $User = $result->fetch_assoc();
    $_SESSION['id'] = $User['fid'];

    if($_SESSION['picStatus'] == 0)
    {
        $_SESSION['picId'] = 0;
        $_SESSION['picName'] = "profile0.png";
    }
    else
    {
        $_SESSION['picId'] = $_SESSION['id'];
        $_SESSION['picName'] = "profile".$_SESSION['picId'].".".$_SESSION['picExt'];
    }
```

Figure 5.3.2 Front-End for signup page

Farmers Coalition

```
$sql = "SELECT * FROM buyer WHERE bemail='$_POST[email]';  
$result = mysqli_query($conn, "SELECT * FROM buyer WHERE bemail='$_POST[email]'") or die(mysqli_error());  
if ($result->num_rows > 0 )  
{  
    $_SESSION['message'] = "User with this email already exists!";  
    //echo $_SESSION['message'];  
    header("location: error.php");  
}  
else  
{  
    $sql = "INSERT INTO buyer (bname, busername, bpassword, bhash, bmobile, bemail, baddress)  
            VALUES ('$_POST[name]','$_POST[username]','$_POST[password]','$_POST[hash]','$_POST[mobile]','$_POST[email]','$_POST[address]');  
    if (mysqli_query($conn, $sql))  
    {  
        $_SESSION['Active'] = 0;  
        $_SESSION['logged_in'] = true;  
        $sql = "SELECT * FROM buyer WHERE busername='$_POST[username]';  
        $result = mysqli_query($conn, $sql);  
        $User = $result->fetch_assoc();  
        $_SESSION['id'] = $User['bid'];  
        $_SESSION['message'] =  
            "Confirmation link has been sent to $email, please verify  
            your account by clicking on the link in the message!";  
        $to      = $email;  
        $subject = "Account Verification ( ArtCircle.com )";  
        $message_body = "  
Hello '$user.',  
";  
    }  
}
```

Figure 5.3.3 Back-End for signup page

5.4 MODULE 4 FUNCTIONALITY

Once logged in, the user will be displayed the current ongoing events and the status of their registration in the home page, if the user is a farmers, additional details like events of the club the farmer is also displayed. The below code snippets show the home page for both types of logins.

Farmers Coalition

```
if(isset($_SESSION['logged_in']) AND $_SESSION['logged_in'] == 1)
{
    $loginProfile = "My Profile: ". $_SESSION['username'];
    $logo = "glyphicon glyphicon-user";
    if($_SESSION['Category']!= 1)
    {
        $link = "Login/profile.php";
    }
    else
    {
        $link = "profileView.php";
    }
}
else
{
    $loginProfile = "Login";
    $link = "index.php";
    $logo = "glyphicon glyphicon-log-in";
}

DOCTYPE html>
<header id="header">
    <h1><a href="index.php">FARMERS COALITION</a></h1>
    <nav id="nav">
        <ul>
            <li><a href="index.php"><span class="glyphicon glyphicon-home"></span> Home</a></li>
            <li><a href="myCart.php"><span class="glyphicon glyphicon-shopping-cart"></span> MyCart</a></li>
            <li><a href="#"><span class=""><?php echo $link; ?></span><?php echo " . $loginProfile; ?></a></li>
            <li><a href="market.php"><span class="glyphicon glyphicon-grain"></span> Digital Market</a></li>
            <li><a href="blogview.php"><span class="glyphicon glyphicon-comment"></span> BLOG</a></li>
        </ul>
    </nav>
</header>
```

Recta

Figure 5.4.1 Home page for Farmers

5.5 MODULE 5 FUNCTIONALITY

The farmers can then register/create new forms. The prominent feature of the project is that it allows farmers with special feature that they can verify the email with the verification whenever they login.

```
if ($_SERVER["REQUEST_METHOD"] == "POST")
{
    $productType = $_POST['type'];
    $productName = dataFilter($_POST['pname']);
    $productInfo = $_POST['pinfo'];
    $productPrice = dataFilter($_POST['price']);
    $fid = $_SESSION['id'];

    $sql = "INSERT INTO fproduct (fid, product, pcat, pinfo, price)
            VALUES ('$fid', '$productName', '$productType', '$productInfo', '$productPrice')";
    $result = mysqli_query($conn, $sql);
    if(!$result)
    {
        $_SESSION['message'] = "Unable to upload Product !!!";
        header("Location: Login/error.php");
    }
    else
    {
        $_SESSION['message'] = "successfull !!!";
    }

    $pic = $_FILES['productPic'];
    $picName = $pic['name'];
    $picTmpName = $pic['tmp_name'];
    $picSize = $pic['size'];
    $picError = $pic['error'];
    $picType = $pic['type'];
    $picExt = explode('.', $picName);
    $picActualExt = strtolower(end($picExt));
    $allowed = array('jpg','jpeg','png');

    if(in_array($picActualExt, $allowed))
    {
        $targetDir = "uploads/";
        $targetFile = $targetDir . basename($picTmpName);
        $uploadOk = 1;
        if(file_exists($targetFile))
        {
            $uploadOk = 0;
        }
        if($uploadOk == 0)
        {
            $_SESSION['message'] = "Sorry, your file was not uploaded.";
        }
        else
        {
            if(move_uploaded_file($picTmpName, $targetFile))
            {
                $_SESSION['message'] = "The file ". $picName . " has been uploaded successfully";
            }
            else
            {
                $_SESSION['message'] = "Sorry, there was an error uploading your file.";
            }
        }
    }
}
```

Figure 5.5.1 backend part of upload product

CHAPTER 6

RESULTS

8.1 Home page

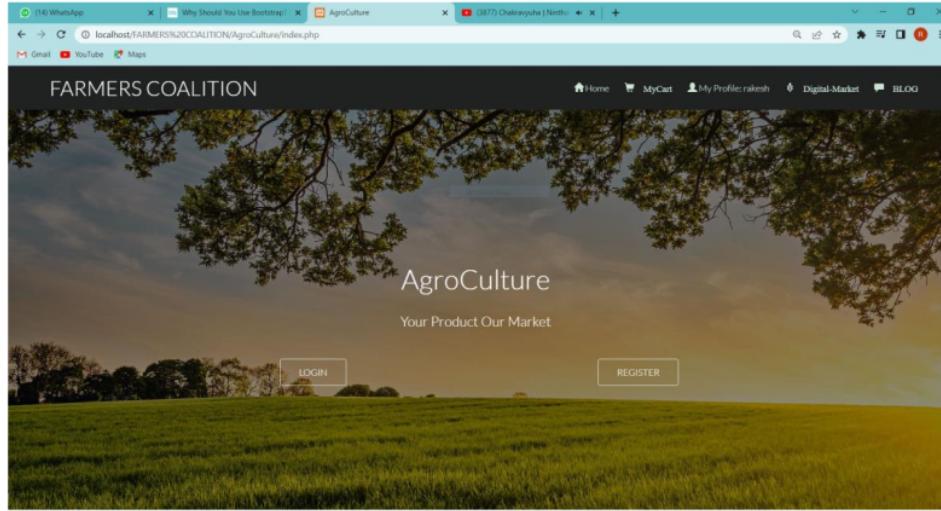


Figure 6.1: Screen Shot of Login Page

Farmers Coalition

8.2 SIGNUP PAGE

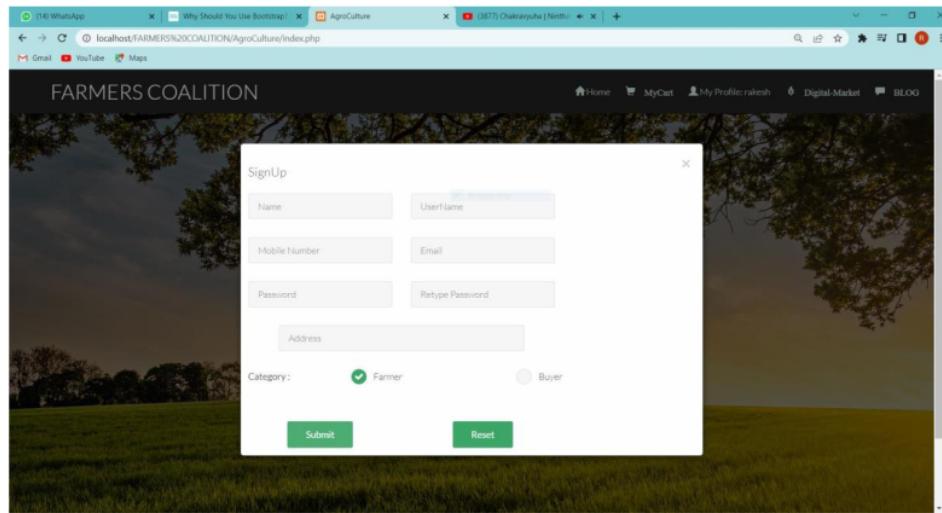


Figure 8.1: New User Registration Screen Shot

6.3 Login page

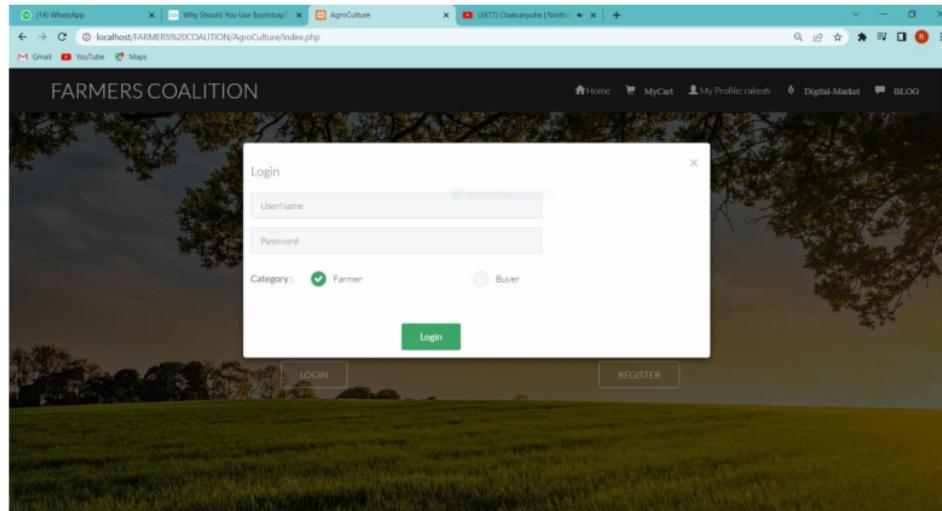


Figure 6.3.1 Farmers login page here can be seen

6.4 ATTENDANCE MANAGER

The user has to enter correct security code in the table to mark attendance. The security pin to be provided by the club members upon successful completion of the event.

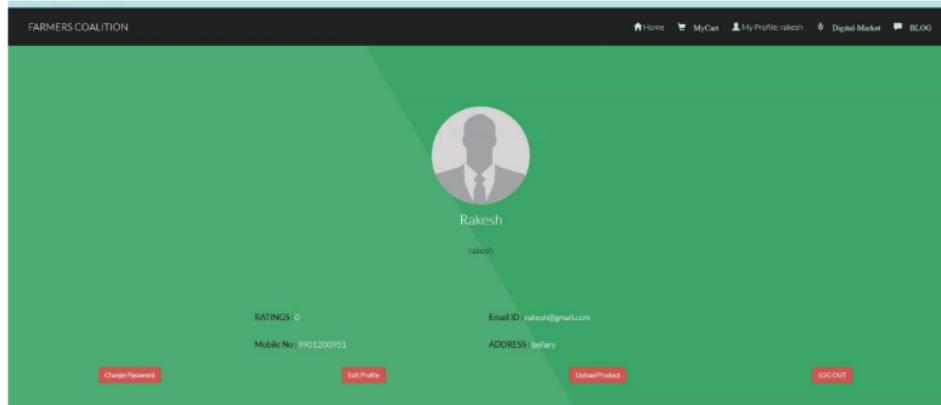


Figure 6.4.1 Wrong attendance security code

6.5 INSERT AND DELETE EVENTS

The event insertion and deletion can be done only by the club-members.

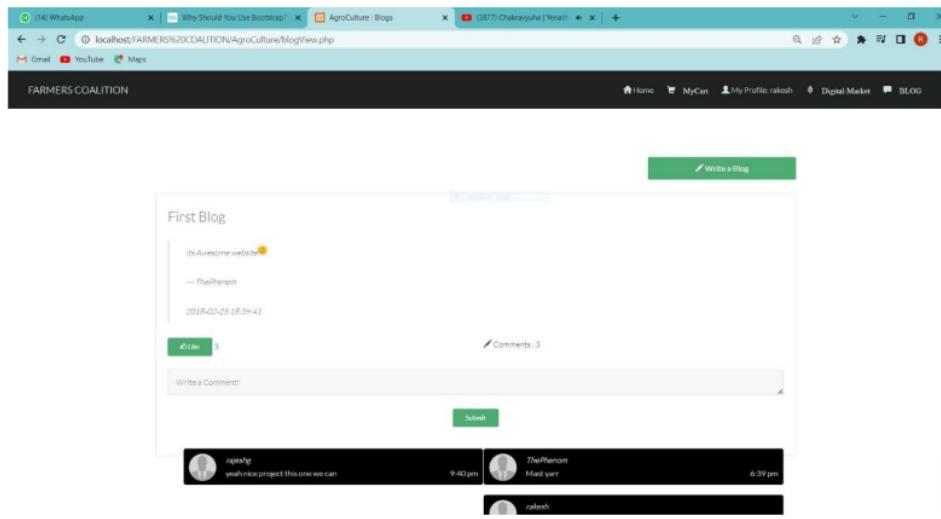


Figure 6.5.1 writing the blog here

CHAPTER 7

CONCLUSION

The mini project has successfully accomplished the goals it had set out in the objectives and design sections of this report.

There is a login page for the farmers, both can view the current ongoing market products. Both login types can register for the events in a single button click. All the data whatever the users login all data stored in the database. Session details are maintained throughout the entire session to help in providing the right pages to the right type of login. Database is used to maintain the durability of the data generated by the users. The project can be improved upon in the future by creating a page for the farmers to access the previous products. Even the admin option can be kept in the project so that they can view the all login users in the website.

1NH19CS139_Farmers Coalition

ORIGINALITY REPORT



PRIMARY SOURCES

- | | | |
|---|---|-----|
| 1 | Khondoker Aminuzzaman, Md. Junayed Miah, Md. Anisur Rahman, Mohammad Moniruzzaman Khan. "Development of Online Home Sharing Web Application", 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), 2021 Publication | 3% |
| 2 | Devi Devapal. "Smart Agro Farm Solar Powered Soil and Weather Monitoring System for Farmers", Materials Today: Proceedings, 2020 Publication | 3% |
| 3 | Eriya, Ade Rahma Yuly, Ryan Naufal Adila, Fitria Nugrahani, Yoyok Sabar Waluyo, Jehad A.H Hammad. "Working Space Virtual Office Prototype in Pandemic Era", 2020 3rd International Conference on Computer and Informatics Engineering (IC2IE), 2020 Publication | 1 % |
| 4 | Mariam Mohamed, Donald J. Ferguson, Adith Venugopal, Mohammad Khursheed Alam, | 1 % |

Laith Makki, Nikhilesh R. Vaid. "An artificial intelligence based referral application to optimize orthodontic referrals in a public oral healthcare system", Seminars in Orthodontics, 2021

Publication

5 Frederick E. Petry. "Fuzzy Databases", Springer Science and Business Media LLC, 1996 1 %

Publication

6 B Mounica, M Akshatha, Anupam Kumar. "A Survey of Real-time Health Care Tracking System for Post Covid Patients", 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), 2022 1 %

Publication

7 Jiang Hong, Zhichong Wang. "Research on big data audit based on financial shared service model", Annals of Operations Research, 2021 1 %

Publication

8 Dipta Voumick, Prince Deb, Sourav Sutradhar, Mohammad Moniruzzaman Khan. "Development of Online Based Smart House Renting Web Application", Journal of Software Engineering and Applications, 2021 <1 %

Publication

9 John Warsinkse. "Software Development Security", Wiley, 2019 <1 %

- 10 R. Norberto Fernández, Marek Rusinkiewicz. "A conceptual design of a soil database for a geographical information system", International journal of geographical information systems, 1993 <1 %

Publication

- 11 Paritosh Tripathi, Vineet Kumar Singh, Muneeesh Chandra Trivedi. "Smart vehicle management using cost effective approach", Materials Today: Proceedings, 2021 <1 %

Publication

Exclude quotes Off

Exclude bibliography On

Exclude matches Off