Ecommerce Application on Cloud Foundry

Introduction:

This application is designed for the ecommerce industry. This application is used for shopping the handmade products like flipkart, amazon, etc... In this document, shows how the application works,

Working:

PHP, My SQL are used to working the backend of the application. PHP is used for connecting the database and templates for working together.

Registration Functionality:

Template of registration is shown below

The below php code helps to connect the registration database table and above template.

Code:

<?php

```php
$name=$_POST['name'];

$number=$_POST['number'];

$mail=$_POST['mail'];

$password=$_POST['password'];

$repassword=$_POST['repassword'];

$address=$_POST['address'];


if(!empty($name)||!empty($number)||!empty($mail)||!empty($password)|
|!empty($repassword)||!empty($address))

{

$host="localhost";

$dbusername="root";

$dbpassword="";

$dbname="project";

$conn=new mysqli($host,$dbusername,$dbpassword,$dbname);


if(mysqli_connect_error())

{

die('connect error('.mysqli_connect_error().')'.mysqli_connect_error());

}

else

{

$SELECT="SELECT mail From register where mail= ? Limit 1";
```

```php
$INSERT="INSERT Into
register(name,number,mail,password,repassword,address)values(?,?,?,?,
?,?)";

$stmt=$conn->prepare($SELECT);

$stmt->bind_param("s",$mail);

$stmt->execute();

$stmt->bind_result($mail);

$stmt->store_result();

$rnum=$stmt->num_rows;

if($rnum==0)

{

$stmt->close();

$stmt=$conn->prepare($INSERT);

$stmt->bind_param("sissss",$name,$number,$mail,$password,$repassword,$address);

$stmt->execute();

echo "New record inserted successfully";

}

else{

echo "someone already register using this email";

}

$stmt->close();

$conn->close();

}
```
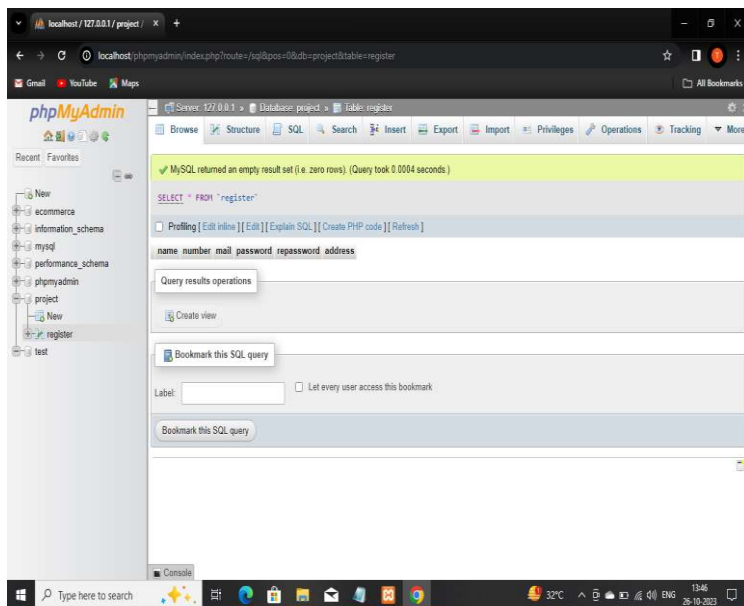
```
}
else{
echo "All field are required";
die();
}
?>
```

After finishing the code,the data entered in the registration form saved in the below registration database table

If the already enterd gmail will enter shows the error message has already registered mail id.

Login Functionality:

The php code for login functionality is mentioned below.This code checks the password whether the password is correct or not for authentication. If it is not logged in otherwise logged in.

Code:

```php
<?php
session_start();
include('db_connection.php');

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $username = $_POST['username'];
    $password = $_POST['password'];

    $stmt = $conn->prepare("SELECT password FROM register WHERE username = ?");
    $stmt->bind_param("s", $username);
    $stmt->execute();
    $stmt->bind_result($hash);

    if ($stmt->fetch() && password_verify($password, $hash)) {
        $_SESSION['username'] = $username; // Store username in the session
        header("Location: home.php"); // Redirect to the home page
        exit();
    } else {
        echo "Login failed.";
    }

    $stmt->close();
}
```

```
$conn->close();
?>
```

LOGIN

Username: [   ]

Password: [   ]

[Log In]

If you haven't account,Please Register here

The above code works for login authentication. If the password and username is correct, redirected to the home page of ecommerce website.

Shopping Cart Functinality:

The shopping cart also the database table to store the product information.The below php code add the product into shopping cart and stores the information on product databse table.

Code:

```php
<?php
session_start();
```

```php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if (!isset($_SESSION['cart'])) {
        $_SESSION['cart'] = array();
    }

    $product_id = $_POST['product_id'];
    $product_name = $_POST['product_name'];
    $product_price = $_POST['product_price'];
    $quantity = $_POST['quantity'];

    $item = array(
        'product_id' => $product_id,
        'product_name' => $product_name,
        'product_price' => $product_price,
        'quantity' => $quantity
    );

    $_SESSION['cart'][] = $item;
}
?>
```

Conclusion:

In this document how the login, registration and shopping cart are working, and source codes are mentioned.