SPARE PARTS MANAGEMENT SYSTEM

A PROJECT REPORT

Submitted by

Mukilan T
20CSR129

Nagul KS
20CSR131

Naveen AM
20CSR136

Prakatesh B
20CSR150

Rahul V
20CSR161

Nitesh Kumar T
20CSl258

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KONGU ENGINEERING COLLEGE
(Autonomous)

PERUNDURAI ERODE – 638 060

NOVEMBER 2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KONGU ENGINEERING COLLEGE
(Autonomous)

PERUNDURAI ERODE – 638060
NOVEMBER 2022

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **SPARE PARTS MANAGEMENT** is a bonafide record of project work done by **MUKILAN T (Register No.:20CSR129), NAGUL K S (Register No.:20CSR131), NAVEEN AM (Register No.:20CSR136), PRAKATESH B (Register No.:20CSR150), RAHUL V (Register No.:20CSR161), NITHESH KUMAR T (Register No.:20CSL258)** for 20CSC51 –Agile Methodologies during the year 2022 - 2023.

Staff-in-charge **HEAD OF THE DEPARTMENT**

(Signature with seal)

Date:

EXAMINER 2 **EXAMINER 1**

ABSTRACT

As the name specifies “SPARE PARTS MANAGEMENT SYSTEM” is developed for helping the company show its products online using a webpage. The customers can visit the webpage and order the products online. This is a webpage for displaying the products of the shop that are available in their shop.

As seen above we used the Spare parts management system to order the spare parts that are available for all automobile vehicles. Hence the shop is located in a single location they can able to get the orders through online and able to deliver the spare parts through parcel services that are available. The customers are verified directly by the admin or the owner, so they can’t able to get any spam orders. Which will help them to save he time. The admin can able to modify the stock details of their shop.

The company requested us with the following requirements startup for the website, about page for displaying company details, contact page for displaying contact details individual login for admin and customer. In admin login, user login, product page for managing product details, query page for displaying queries and order details. . We have done this project using HTML, CSS and BOOTSTRAP as front end and PHP and SQL as my backend.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 MOTIVATION

The objective of the project is to develop a system that store and display the spare parts that are available in a shop. The purpose is to design a system using which one can perform all operations related to viewing, ordering and booking. Moreover, product value recent learning about the html and react JS programming languages as well as seeing how powerful and dynamic they are when it comes to web designing and applications.

Apart from helping computer science students understand the concepts of web-application designing, it would be very easy to incorporate the idea of using programming techniques from the available visuals to understand how a piece of code appears on a user interface. The languages used to build this application are HTML,php and because I found them to be extremely useful while working on the technologies at my workplace

1.2 SYSTEM STUDY

The proposed system is a web based application and maintains a centralized repository of all related information. The system allows one to easily access the relevant information and make necessary travel arrangements. Users can view and book the products that are available in the shop through online.

1.3 OBJECTIVES

The objective of the project is to develop a webpage that help the customers to order the item through online which will help them to save the time instead of going to the shop and buying things. The owner can also make use of the website to manage products and customers through virtually. The customers can order or keep the product in the cart which will help to buy them later.

1.4 SCOPE

A small automobile spare parts shop sells the spare parts for vehicles of several models. Also each part is typically manufactured by several small industries. To stream line the sales and supply ordering, the shop owner has asked us to develop the following motor part shop software.

The motor part shop deals with large no. of motor parts of various manufacturers and various vehicle types. Some of the motor parts are very small and some are very large. The owner maintains different parts in wall mounted and numbered racks.

The shop owner maintains as few inventory for each item as reasonable, to reduce inventory overheads after being inspired by "just in time (JIT) philosophy".

The system can be accessed by:

- Customers (normal users)
- Shop owner(with unlimited access)

Both the user and shop owner are authorised before using the software

CHAPTER 2

GENERAL DESCRIPTION

2.1 CHARACTERISTICS OF USERS AND ADMIN

CHARACTERISTICS OF USER

The ‘operational or generic user interface’ helps the end users of the system in transactions through the existing data and required services. The operational user interface also helps the ordinary users in managing their own information in a customized manner as per the included flexibilities.

CHARACTERISTICS OF ADMIN

In admin login, the admin can manage bookings, pakage details, queries, etc. In query, if the user raises a query admin can give a possible solution for that query.

2.2 DESIGN AND IMPLEMENTATION CONSTRAINTS

Product is developed using HTML , CSS and JS. The backend database for this SQL server. The product is accomplished with login facility so that specific function is available to specific user.

CHAPTER 3

REQUIREMENTS SPECIFICATION

3.1 FUNCTIONAL REQUIREMENTS

Number of Modules

After careful analysis the system has been identified to have the following modules:

Administrator module

User module

Guest user

ADMINISTRATOR MODULE:

This module provides administrator related functionality. Administrator manages all information and has access rights to add, delete, edit and view the data related to places, travels, routes, bookings, Enquiries etc.

- Products—Admin will add the products and manage the product lists(Add, Update, delete).
- Users- Admin view all Information of all users.
- Orders- Admin will responsible for manage orders. Admin can confirm, verify and cancel the products ordered by customers.

- Manage issues/ Complaints—Admin can take action on any issue /complaint raised by cusomer and Put remark.
- Manage Enquiries—admin can manage all enquiries raised by customers.
- Manage pages- Admin can edit the info of all pages that are display on the website,
- Dashboard- Here admin can view all count of orders, issues , inquiries and Users.
- Change password--- Admin can change own password.

USER MODULE:

- Signup- User can register yourself for ordering items.
- Sign-in- Here user can login with valid username and password.
- Forgot-Password-User can recover his/her own password.
- My Profile- user can update own profile.
- Cart - user can add items to the cart to order them together or they to store them which will be used to buy them later in offline
- Change Password-User can own Password.
- Write-use-Here user can raise any issue related to booking. Cancelation etc.

GUEST MODULE:

- Guest user can visit the website and view the all content of website. Guest user can also Enquiry.

3.2 NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements describe how a system must behave and establish constraints of its functionality. This type of requirement is also known as the system's quality attributes. It lists the requirements of a particular software system including functional, performance and security requirements. It is portable and highly secured. Non-functional requirements define, how the server is reliable and withstand the cause. It describes aspects of the system that are concerned with how the system provides the non-functional requirements. They are:

SECURITY:

We are going to develop a secured database for the organization .There are different categories of users. Depending upon the category of user the access rights are decided

PERFORMANCE AND RESPONSE TIME:

The system has high performance rate when executing user's input and will be able to provide feedback or response within a short time span usually 50 seconds for highly complicated task and 20 to 25 seconds for less complicated task.

ERROR HANDLING:

In this website Error will be considerably minimized and an appropriate error message that guides the user to recover from an error will be provided. Validation of user's input is highly essential. Also, the standard time taken to recover from an error will be 15 to 20 seconds.

AVAILABILITY:

This system should always be available for access at 24 hours, 7 days a week. Also, in the occurrence of any major system malfunctioning, the system should be available in 1 to 2 working days, so that the business process is not severely affected.

3.3 USER INTERFACE

User interface defines how a person is handling an application. The component of user interface is a presentation layer, logical layer. For user, individual login will be provided with signup page. Web page will be there to display the packages and a user setting page to maintain the users setting and the payment page to make payment in online.

CHAPTER 4

DETAILED DESIGN

4.1 HARDWARE AND SOFTWARE SPECIFICATION

HARDWARE REQUIREMENTS

Client side:

- OS: Windows 7 or above MAC or UNIX for better performance.
- Processor: Pentium III or 2.0 GHz or higher.
- RAM: 1GB or more.

Server Side:

- OS: Windows 9x/XP, Windows ME.
- Processor: Pentium 3.0 GHz or Higher.
- RAM: 2 GB or more.
- Hard Drive: 10GB or more.

SOFTWARE REQUIREMENTS

- DATABASE: SQL
- MS Visual Studio Code and extensions for coding purposes.
- GitHub for team collaboration.

TECHNOLOGIES USED FRONT END:

HTML

The HyperText Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

TECHNOLOGIES USED FOR BACKEND

SQL

Structured Query Language, abbreviated as SQL, is a domain-specific language used in programming and designed for managing data held in a relational database management system (RDBMS), or for stream processing in a relational data stream management system (RDSMS). It is particularly useful in handling structured data, i.e. data incorporating relations among entities and variables.

PHP

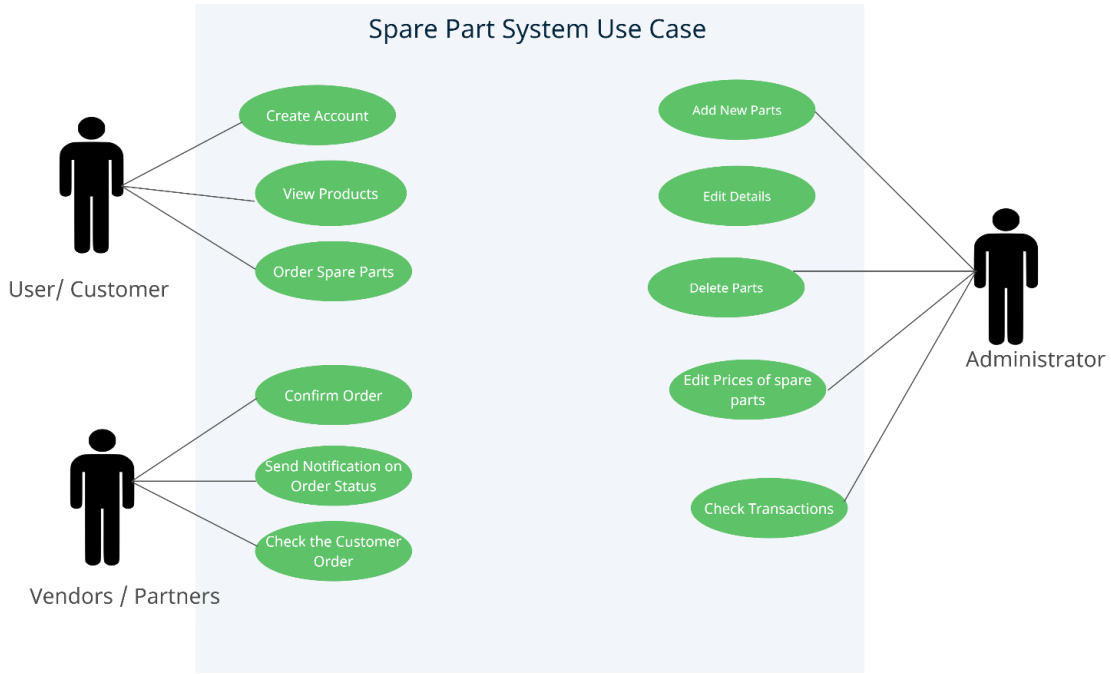
PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI)

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. 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.

4.2 USE CASE DIAGRAM

The functionality of a system can be described in a number of different use cases, each of which represents a specific flow of events in a system. It is a graph of actors, a set of use-cases enclosed in a boundary, communication, associations between the actors and the use-cases, and generalization among the use-cases.
The use cases used in this system are

- **Login:** Used for both people and admin.
- **Add products:** Used to add the product that is available in shop.
- **Details:** People can choose the complaints for which they're registering.
- **View complaint:** Complaint status can be viewed by the people.
- **Update status:** The status of the products will be updated by the admin.
- **Create admin password:** Admin can create a new admin user or change the password if necessary.



ACTORS

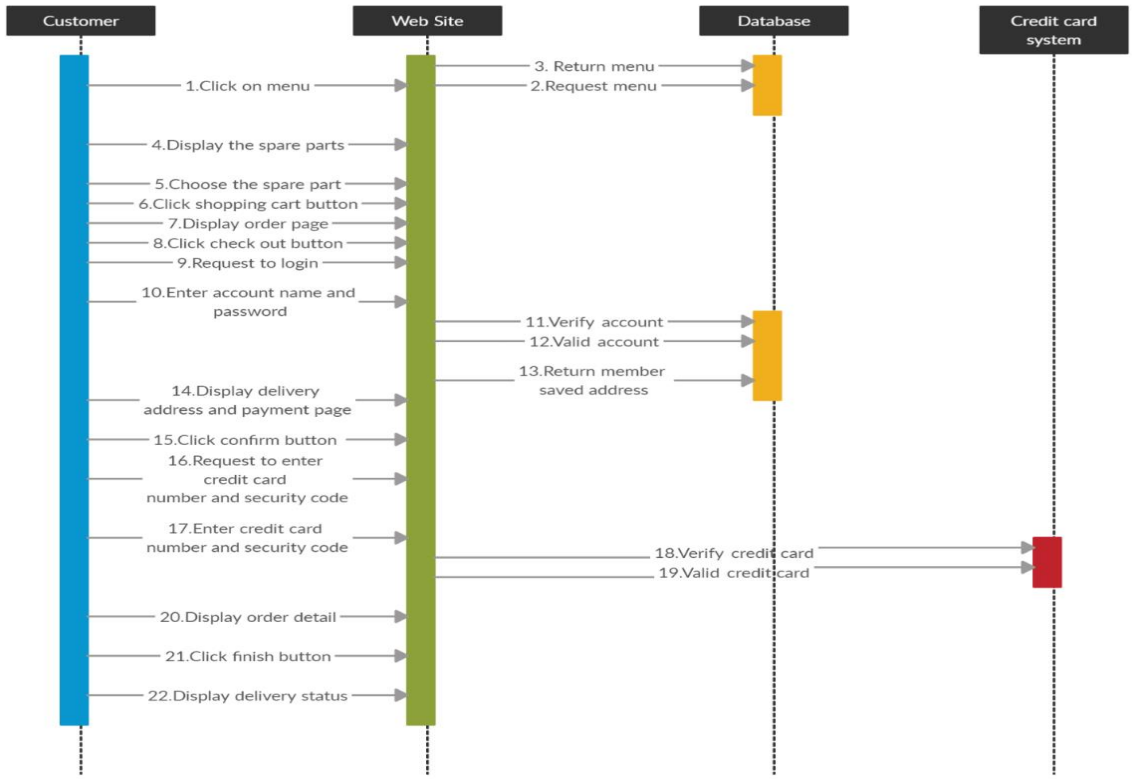
The actors used in this system are

Customer **:** User can book the products that are available in the shop.

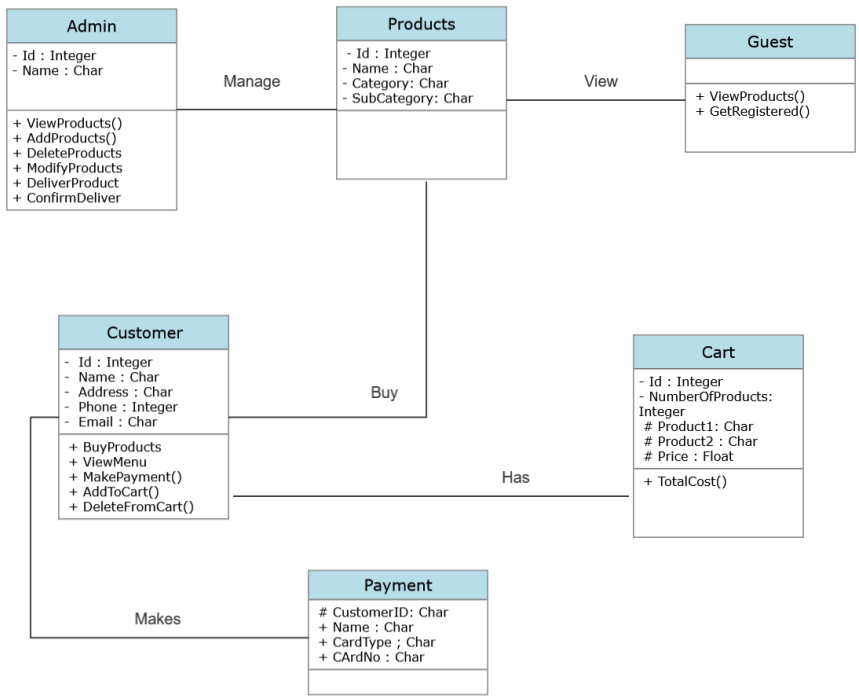
Admin **:** Admin view all Information of all users.

4.3 ACTIVITY DIAGRAM

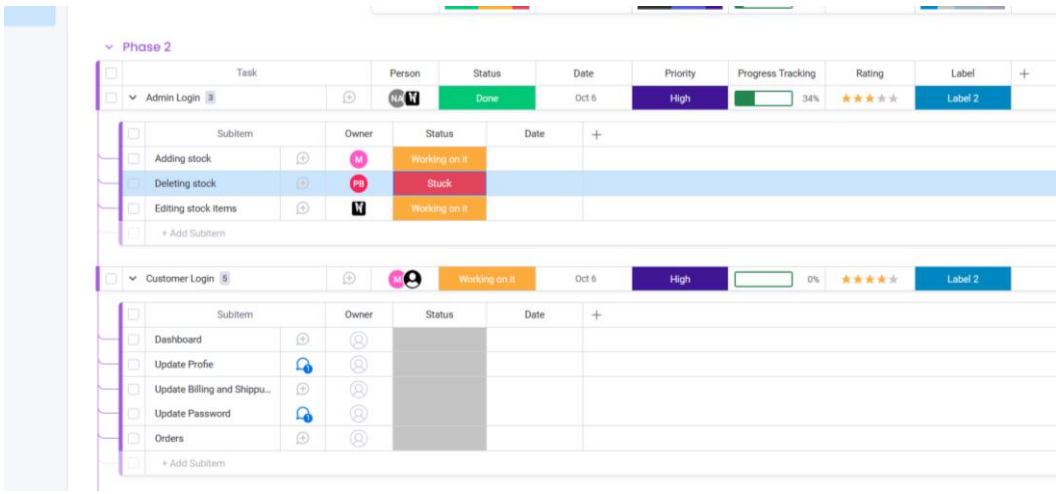
It shows organization and their dependence among the set of components. These diagrams are particularly useful in connection with workflow and in describing behaviour that has a lot of parallel processing. An activity is a state of doing something: either a real-world process, or the execution of a software routine.



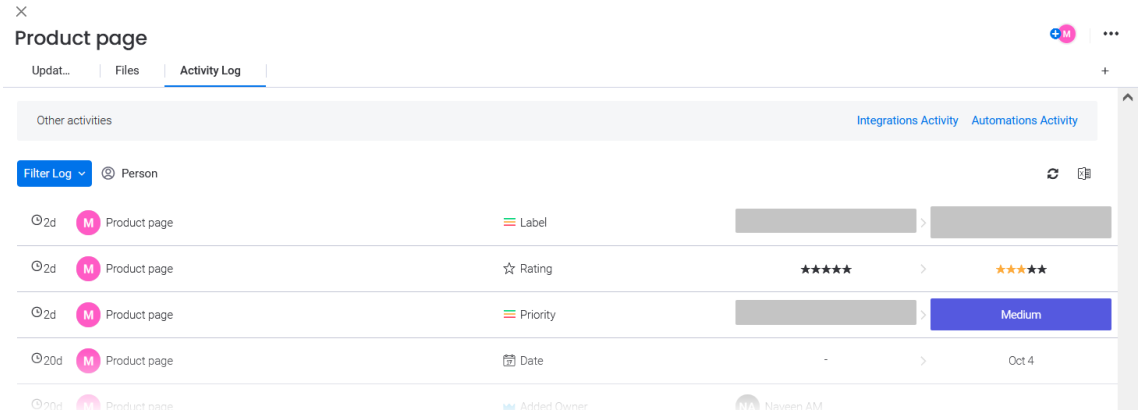
4.4 CLASS DIAGRAM

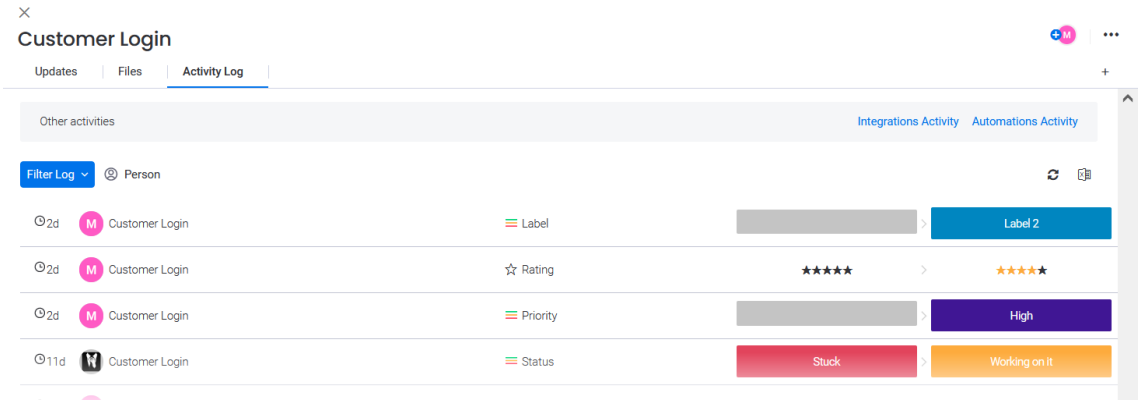


4.5 USER STORY CREATION

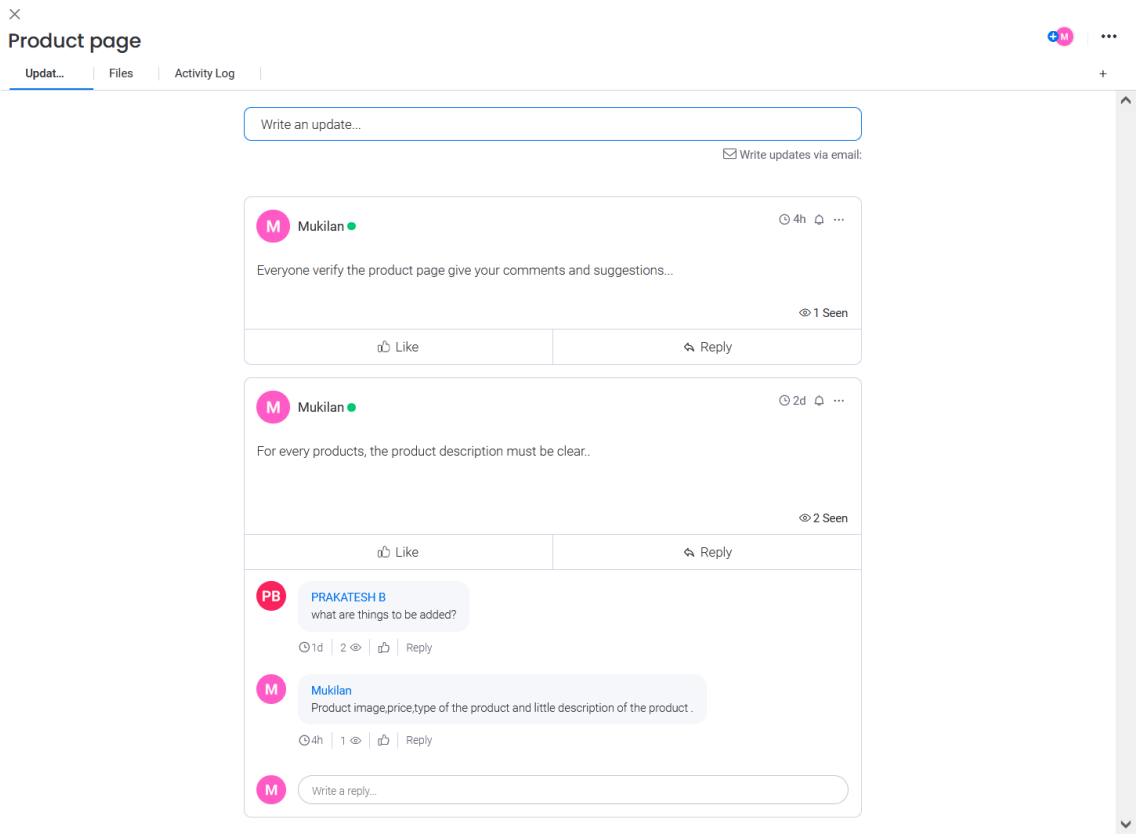


4.6 CREATING A PRODUCT BACKLOG USING SCRUM

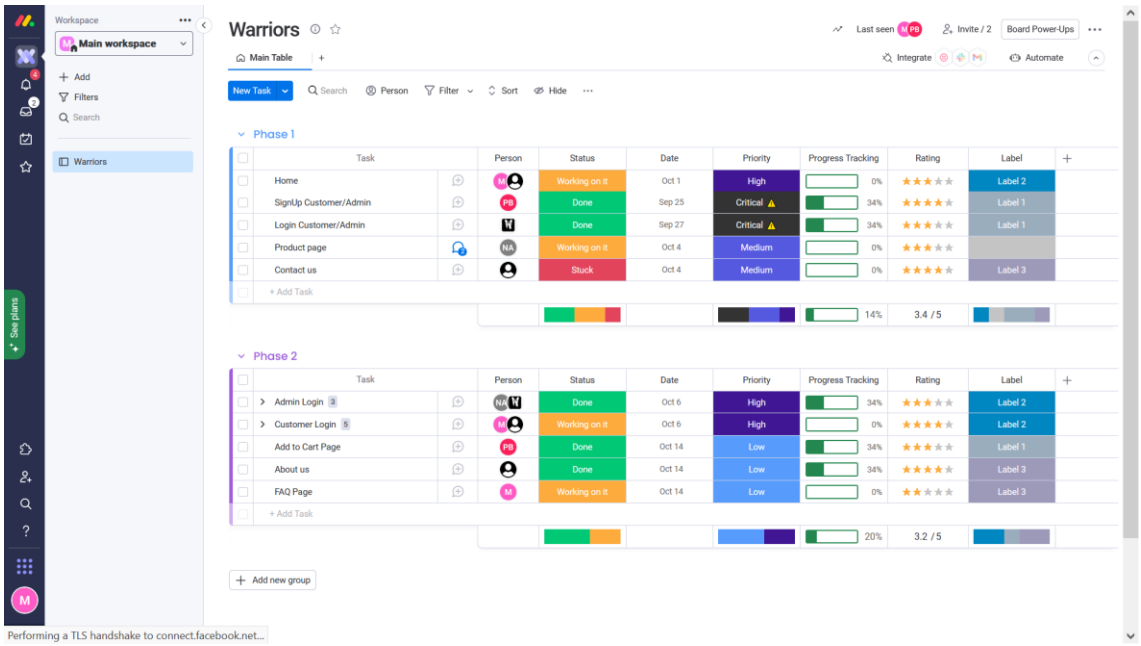




4.7 CREATING AND ASSIGNING SPRINT BACKLOG USER STORIES



4.8 USER STORY NAVIGATION IN TASKBOARD



CHAPTER 5

TESTING

SYSTEM TESTING

Once the entire system has been built then it has to be tested against the Software Requirement Specification and System Specification to check if it delivers the features required. System testing can involve a number of specialist type of tests to see if all the functional and non-functional requirements have been met.

MANUAL TESTING

Manual Testing will be done to ensure the correctness of various parts of the code using test cases generated by the tester. Some of the tests performed in this project are given below:

5.1 UNIT TESTING

Table 5.1-unit testing

Test no	Module	Test Data	Expect ed Result	Actual result	comments	Final result
1	Admi n login	Email abcd@gmail.com	Should accept the data	Accepts the value	It checks whether the email id contain @gmail.co m	Valid Data
2	Admi n login	Email: abcd.com	Should accept the data	Throw s error	An email id should contain @gmail.co m	Invalid data

5.2 INTEGRATION TESTING

Integration tests are done to test integrated application components were individually satisfactory, as shown by successful unit testing; the combination of components is correct and consistent. Authorization and authentication module

Table 5.2 integration testing

no	Module	Test Data	Expected Result	Actual result	comment	Final result
1	Login page	Username: admin@gmail. com Password: admin	Should allow to login	Login successfully	It checks whether the user is authorized for the access	Vali d Data
2	Login page	Username: admin Password: xxx	Should allow to login	Not allowing to login	When the input data is wrong, it fails to login into the application	Invalid data

CHAPTER 6

RESULTS

As a result, the application is highly secured and increased in performance, the system is user friendly, efficient, reliable and easily maintainable. Thus, this application helps in achieving complaint management easily and efficiently.

CHAPTER 7

7.1 CONCLUSION

Here we have presented the design of a spare parts management system that can provide the users to order the products or spare parts in the shop. This is a combination of smartphone and Internet services. It can be used any were to order directly to the shop. The customer can use the web application .We developed the system so the user buy the products from anywhere. Our site provides readily available information. A customer can see the relevant detailed information within seconds, rather than waiting for days or weeks.

7.2 FUTURE WORKS

There is always a room for the improvement in any software package, however good and efficient. It may be the important thing is that the software should be flexible enough for further modifications. In future we will enhance this project by adding some more essential fields if needed.

APPENDIX 1

SAMPLE CODING

Index Page

```
<?php require_once('header.php'); ?>

<?php
$statement = $pdo->prepare("SELECT * FROM tbl_settings WHERE id=1");
$statement->execute();
$result = $statement->fetchAll(PDO::FETCH_ASSOC);
foreach ($result as $row)
{
   $cta_title = $row['cta_title'];
   $cta_content = $row['cta_content'];
   $cta_read_more_text = $row['cta_read_more_text'];
   $cta_read_more_url = $row['cta_read_more_url'];
   $cta_photo = $row['cta_photo'];
   $featured_product_title = $row['featured_product_title'];
   $featured_product_subtitle = $row['featured_product_subtitle'];
   $latest_product_title = $row['latest_product_title'];
   $latest_product_subtitle = $row['latest_product_subtitle'];
   $popular_product_title = $row['popular_product_title'];
   $popular_product_subtitle = $row['popular_product_subtitle'];
   $total_featured_product_home = $row['total_featured_product_home'];
   $total_latest_product_home = $row['total_latest_product_home'];
   $total_popular_product_home = $row['total_popular_product_home'];
   $home_service_on_off = $row['home_service_on_off'];
   $home_welcome_on_off = $row['home_welcome_on_off'];
   $home_featured_product_on_off = $row['home_featured_product_on_off'];
   $home_latest_product_on_off = $row['home_latest_product_on_off'];
   $home_popular_product_on_off = $row['home_popular_product_on_off'];

}
?>
<div id="bootstrap-touch-slider" class="carousel bs-slider fade control-round indicators-
line" data-ride="carousel" data-pause="hover" data-interval="false" >
   <!-- Indicators -->
   <ol class="carousel-indicators"><?p
```

```
$i=0;
      $statement = $pdo->prepare("SELECT * FROM tbl_slider");
      $statement->execute();
      $result = $statement->fetchAll(PDO::FETCH_ASSOC);
      foreach ($result as $row) {              ?>
         <li data-target="#bootstrap-touch-slider" data-slide-to="<?php echo $i; ?>" <?php
if($i==0) {echo 'class="active"';} ?>></li>
         <?php
         $i++;  }?>
   </ol>
   <!-- Wrapper For Slides -->
   <div class="carousel-inner" role="listbox">
      <?php
      $i=0;
      $statement = $pdo->prepare("SELECT * FROM tbl_slider");
      $statement->execute();
      $result = $statement->fetchAll(PDO::FETCH_ASSOC);
      foreach ($result as $row) {              ?>
         <div class="item <?php if($i==0) {echo 'active';} ?>" style="background-
image:url(assets/uploads/<?php echo $row['photo']; ?>);">
            <div class="bs-slider-overlay"></div>
            <div class="container">
               <div class="row">
                  <div class="slide-text <?php if($row['position'] == 'Left') {echo
'slide_style_left';} elseif($row['position'] == 'Center') {echo 'slide_style_center';}
elseif($row['position'] == 'Right') {echo 'slide_style_right';} ?>">
                     <h1 data-animation="animated <?php if($row['position'] == 'Left') {echo
'zoomInLeft';} elseif($row['position'] == 'Center') {echo 'flipInX';} elseif($row['position']
== 'Right') {echo 'zoomInRight';} ?>"><?php echo $row['heading']; ?></h1>
                     <p data-animation="animated <?php if($row['position'] == 'Left') {echo
'fadeInLeft';} elseif($row['position'] == 'Center') {echo 'fadeInDown';}
elseif($row['position'] == 'Right') {echo 'fadeInRight';} ?>"><?php echo
nl2br($row['content']); ?></p>
                     <!-- <a href="<?php echo $row['button_url']; ?>" target="_blank"
class="btn btn-primary" data-animation="animated <?php if($row['position'] == 'Left')
{echo 'fadeInLeft';} elseif($row['position'] == 'Center') {echo 'fadeInDown';}
elseif($row['position'] == 'Right') {echo 'fadeInRight';} ?>"><?php echo
$row['button_text']; ?></a> -->
                  </div>
               </div>   </div>  </div>
    <?php    $i++;  }?>
   </div>
   <!-- Slider Left Control -->
```

```
    <a class="left carousel-control" href="#bootstrap-touch-slider" role="button" data-
slide="prev">
        <span class="fa fa-angle-left" aria-hidden="true"></span>
        <span class="sr-only">Previous</span>
    </a>

    <!-- Slider Right Control -->
    <a class="right carousel-control" href="#bootstrap-touch-slider" role="button" data-
slide="next">
        <span class="fa fa-angle-right" aria-hidden="true"></span>
        <span class="sr-only">Next</span>
    </a>
</div>

<?php if($home_service_on_off == 1): ?>
<div class="service bg-gray">
    <div class="container">
        <div class="row">
            <?php
                $statement = $pdo->prepare("SELECT * FROM tbl_service");
                $statement->execute();
                $result = $statement->fetchAll(PDO::FETCH_ASSOC);
                foreach ($result as $row) {
                    ?>
                    <div class="col-md-4">
                        <div class="item">
                            <div class="photo"><img src="assets/uploads/<?php echo $row['photo'];
?>" width="150px" alt="<?php echo $row['title']; ?>"></div>
                            <h3><?php echo $row['title']; ?></h3>
                            <p>
                                <?php echo nl2br($row['content']); ?>
                            </p>    </div></div>
                <?php
            }  ?>
        </div>   </div>
</div>
<?php endif; ?>

<?php if($home_featured_product_on_off == 1): ?>
<div class="product pt_70 pb_70">
    <div class="container">
        <div class="row">
            <div class="col-md-12">
                <div class="headline">
```

```
                <h2><?php echo $featured_product_title; ?></h2>
                <h3><?php echo $featured_product_subtitle; ?></h3>
            </div>
        </div>
    </div>
    <div class="row">
        <div class="col-md-12">

            <div class="product-carousel">
                <?php
                $statement = $pdo->prepare("SELECT * FROM tbl_product WHERE
p_is_featured=? AND p_is_active=? LIMIT ".$total_featured_product_home);
                $statement->execute(array(1,1));
                $result = $statement->fetchAll(PDO::FETCH_ASSOC);
                foreach ($result as $row) {
                    ?>
                    <div class="item">
                        <div class="thumb">
                            <div class="photo" style="background-image:url(assets/uploads/<?php
echo $row['p_featured_photo']; ?>);"></div>
                            <div class="overlay"></div>
                        </div>
                        <div class="text">
                            <h3><a href="product.php?id=<?php echo $row['p_id']; ?>"><?php
echo $row['p_name']; ?></a></h3>
                            <h4>
                                $<?php echo $row['p_current_price']; ?>
                                <?php if($row['p_old_price'] != ''): ?>
                                <del>
                                    $<?php echo $row['p_old_price']; ?>
                                </del>
                                <?php endif; ?>
                            </h4>
                            <div class="rating">

                            <?php if($row['p_qty'] == 0): ?>
                                <div class="out-of-stock">
                                    <div class="inner">
                                        Out Of Stock
                                    </div>
                                </div>
                            <?php else: ?>
                                <p><a href="product.php?id=<?php echo $row['p_id']; ?>"><i
class="fa fa-shopping-cart"></i> Add to Cart</a></p>
```

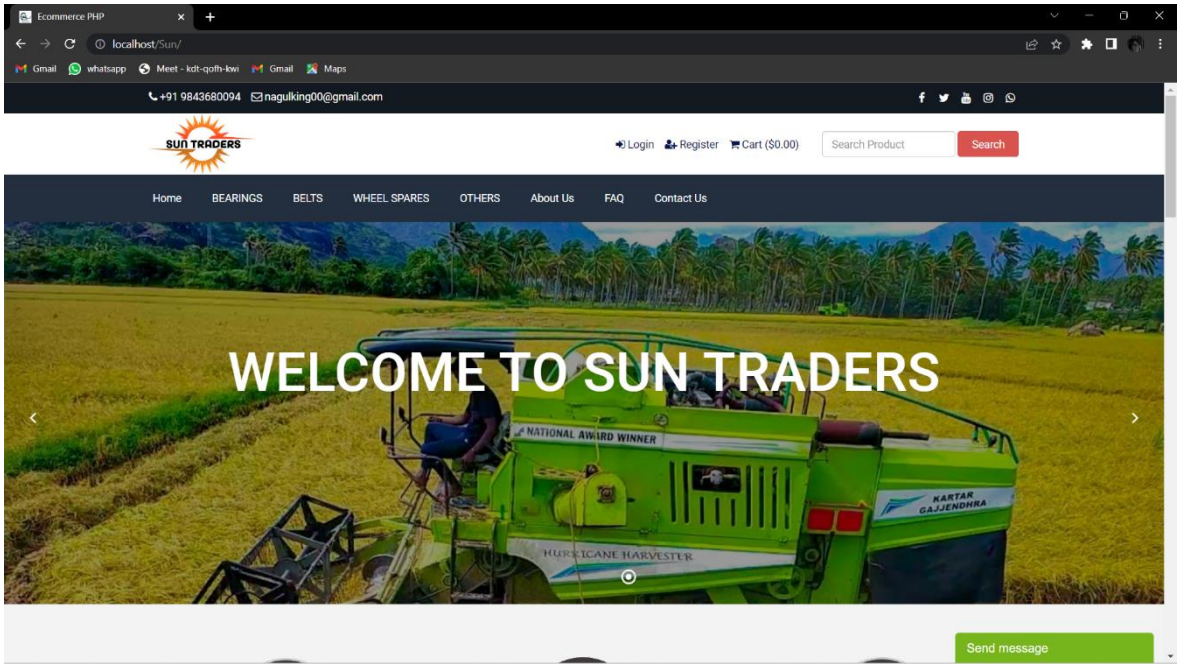
```
                        <?php endif; ?>
                    </div></div>
                <?php
            }?></div>  </div></div></div></div>
<?php endif; ?>


<?php if($home_latest_product_on_off == 1): ?>
<div class="product bg-gray pt_70 pb_30">
   <div class="container">
      <div class="row">
         <div class="col-md-12">
            <div class="headline">
               <h2><?php echo $latest_product_title; ?></h2>
               <h3><?php echo $latest_product_subtitle; ?></h3>
            </div>
         </div>
      </div>
      <div class="row">
         <div class="col-md-12">

            <div class="product-carousel">

               <?php
               $statement = $pdo->prepare("SELECT * FROM tbl_product WHERE
p_is_active=? ORDER BY p_id DESC LIMIT ".$total_latest_product_home);
               $statement->execute(array(1));
               $result = $statement->fetchAll(PDO::FETCH_ASSOC);
               foreach ($result as $row) {
                  ?>
                  <div class="item">
                     <div class="thumb">
                        <div class="photo" style="background-image:url(assets/uploads/<?php
echo $row['p_featured_photo']; ?>);"></div>
                        <div class="overlay"></div>
                     </div>
                     <div class="text">
                        <h3><a href="product.php?id=<?php echo $row['p_id']; ?>"><?php
echo $row['p_name']; ?></a></h3>
                        <h4>
                           $<?php echo $row['p_current_price']; ?>
                           <?php if($row['p_old_price'] != ''): ?>
                           <del>
                              $<?php echo $row['p_old_price']; ?>
```

```
                        </del>
                        <?php endif; ?>
                    </h4>
                    <div class="rating">
                        <?php
                        $t_rating = 0;
                        $statement1 = $pdo->prepare("SELECT * FROM tbl_rating
WHERE p_id=?");
                        $statement1->execute(array($row['p_id']));
                        $tot_rating = $statement1->rowCount();
                        if($tot_rating == 0) {
                            $avg_rating = 0;
                        } else {
                            $result1 = $statement1->fetchAll(PDO::FETCH_ASSOC);
                            foreach ($result1 as $row1) {
                                $t_rating = $t_rating + $row1['rating'];
                            }
                            $avg_rating = $t_rating / $tot_rating;
                        }
                        ?>
                    </div>
                    <?php if($row['p_qty'] == 0): ?>
                        <div class="out-of-stock">
                            <div class="inner">
                                Out Of Stock
                            </div>     </div>
                    <?php else: ?>
                        <p><a href="product.php?id=<?php echo $row['p_id']; ?>"><i
class="fa fa-shopping-cart"></i> Add to Cart</a></p>
                    <?php endif; ?>
                </div>
            </div>
            <?php
        }     </div></div></div></div>
</div>
<?php endif; ?>
<?php if($home_popular_product_on_off == 1): ?>
<div class="product pt_70 pb_70">
    <div class="container">
        <div class="row">
            <div class="col-md-12">
                <div class="headline">
                    <h2><?php echo $popular_product_title; ?></h2>
                    <h3><?php echo $popular_product_subtitle; ?></h3>
```

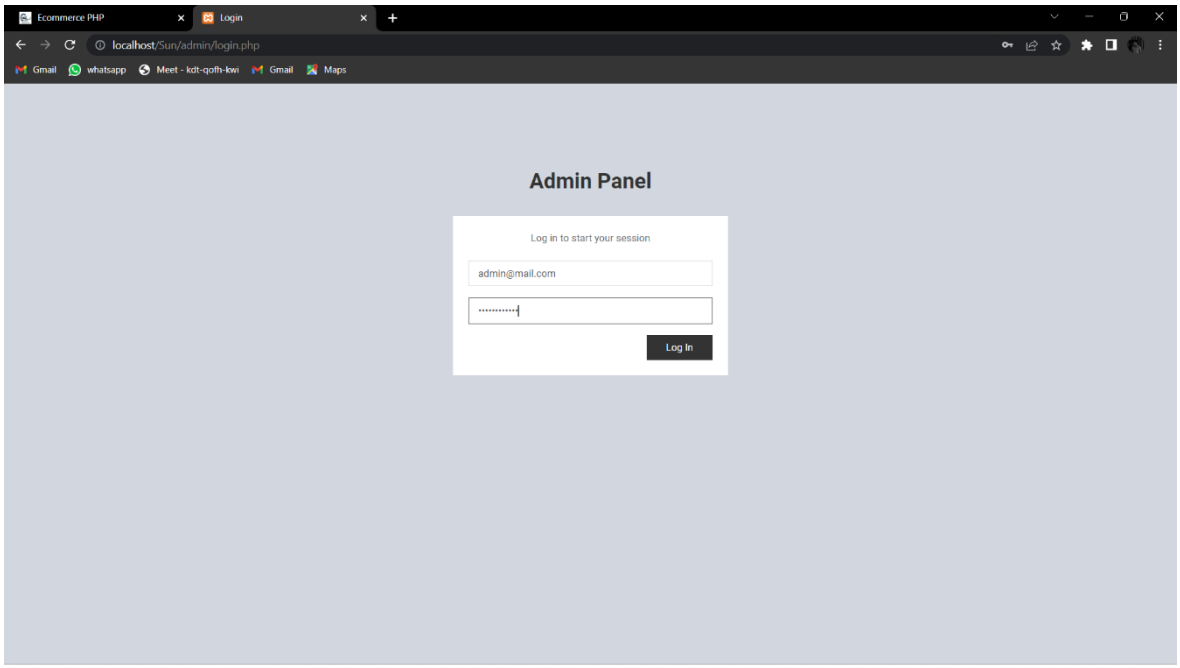
```
            </div>
          </div>
        </div>
        <div class="row">
          <div class="col-md-12">
            <div class="product-carousel">

              <?php
              $statement = $pdo->prepare("SELECT * FROM tbl_product WHERE
p_is_active=? ORDER BY p_total_view DESC LIMIT ".$total_popular_product_home);
              $statement->execute(array(1));
              $result = $statement->fetchAll(PDO::FETCH_ASSOC);
              foreach ($result as $row) {  ?>
                <div class="item">
                  <div class="thumb">
                    <div class="photo" style="background-image:url(assets/uploads/<?php
echo $row['p_featured_photo']; ?>);"></div>
                    <div class="overlay"></div>
                  </div>
                  <div class="text">
                    <h3><a href="product.php?id=<?php echo $row['p_id']; ?>"><?php
echo $row['p_name']; ?></a></h3>
                    <h4>
                      $<?php echo $row['p_current_price']; ?>
                      <?php if($row['p_old_price'] != ''): ?>
                      <del>
                        $<?php echo $row['p_old_price']; ?>
                      </del>
                      <?php endif; ?>
                    </h4>
                    <div class="rating">                              <?php else: ?>
                      <p><a href="product.php?id=<?php echo $row['p_id']; ?>"><i
class="fa fa-shopping-cart"></i> Add to Cart</a></p>
                    <?php endif; ?>
                  </div>
                </div>
                <?php
              }
              ?>
            </div>
</div></div></div>
</div>
<?php endif; ?>
<?php require_once('footer.php'); ?>
```

APPENDIX 2

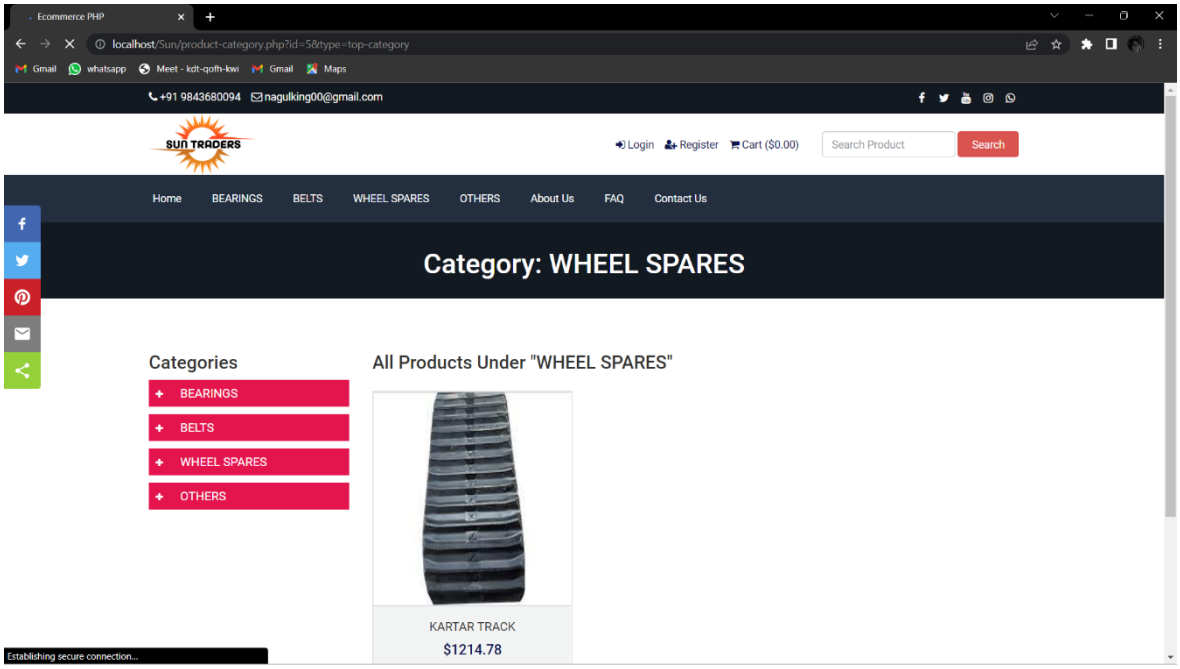
SCREENSHOTS



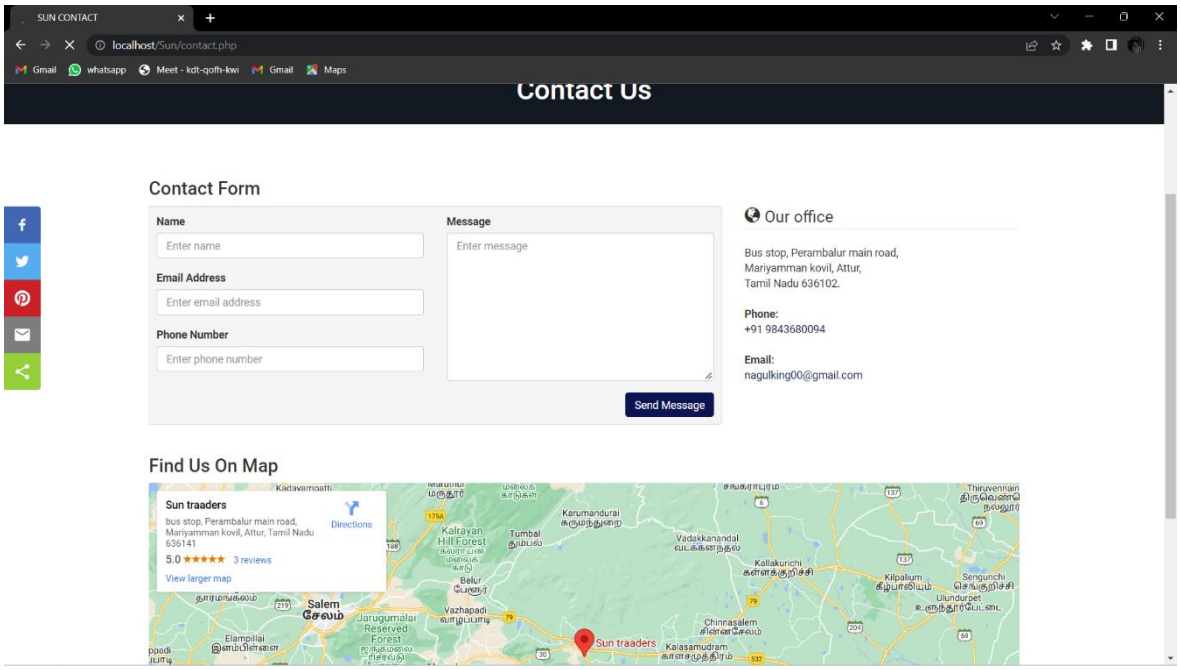
A2.1 Home Page



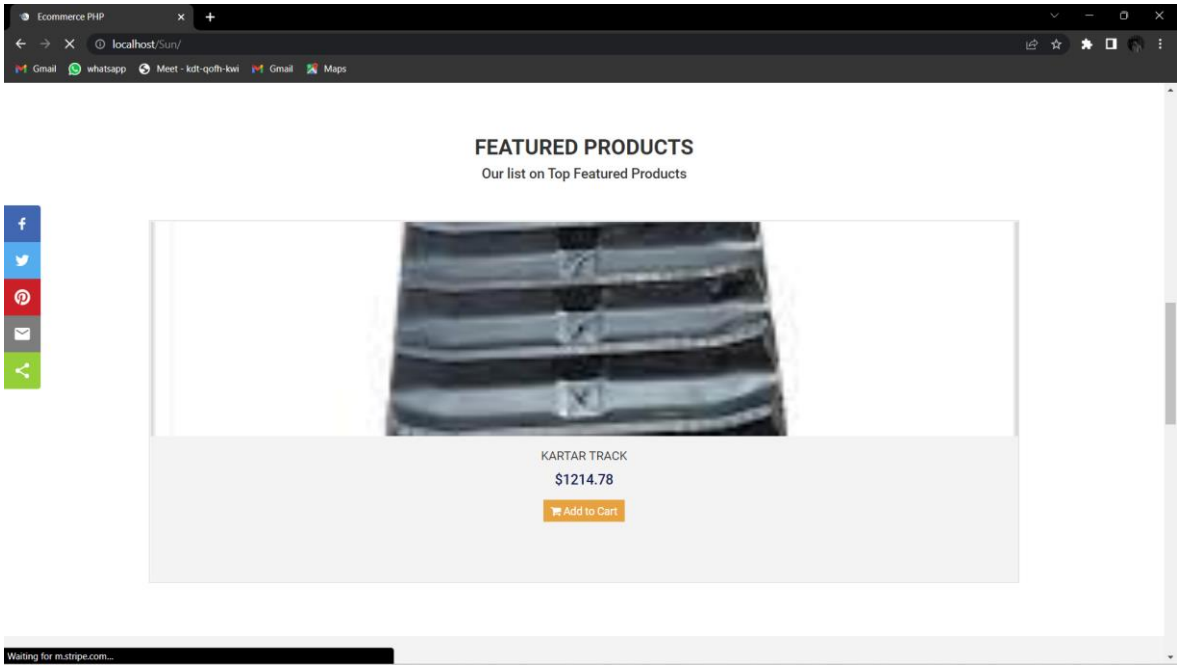
A2.2 Admin Login



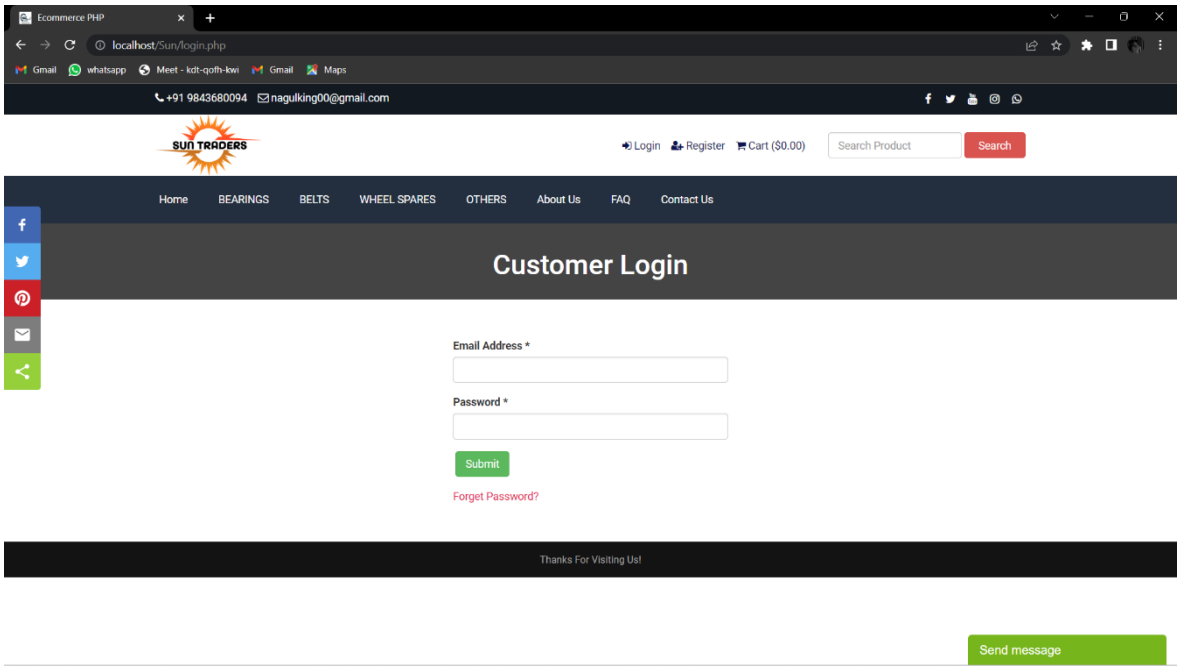
A2.3 Product Page



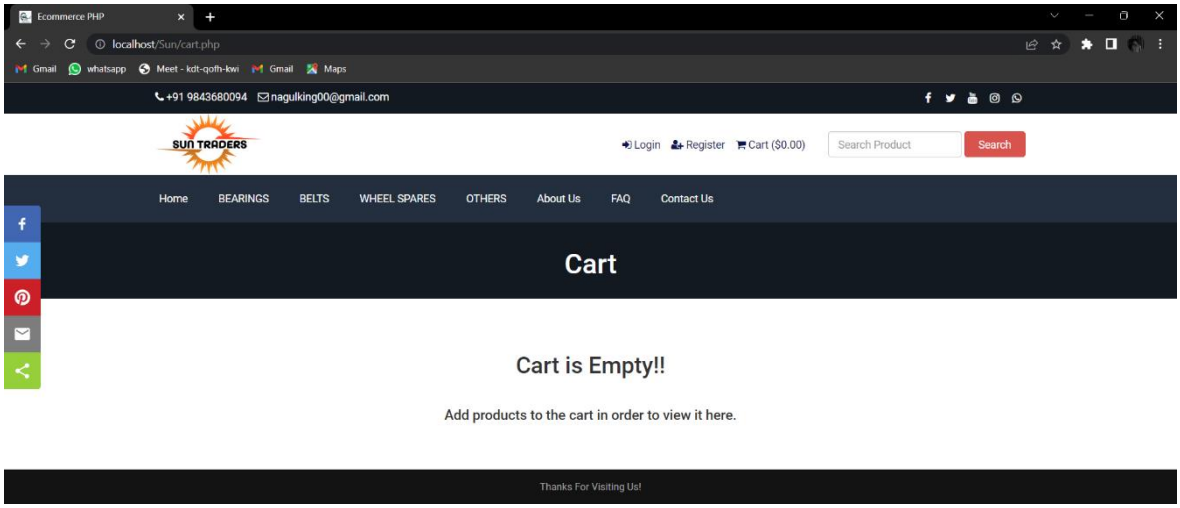
A2.4 Contact us Page



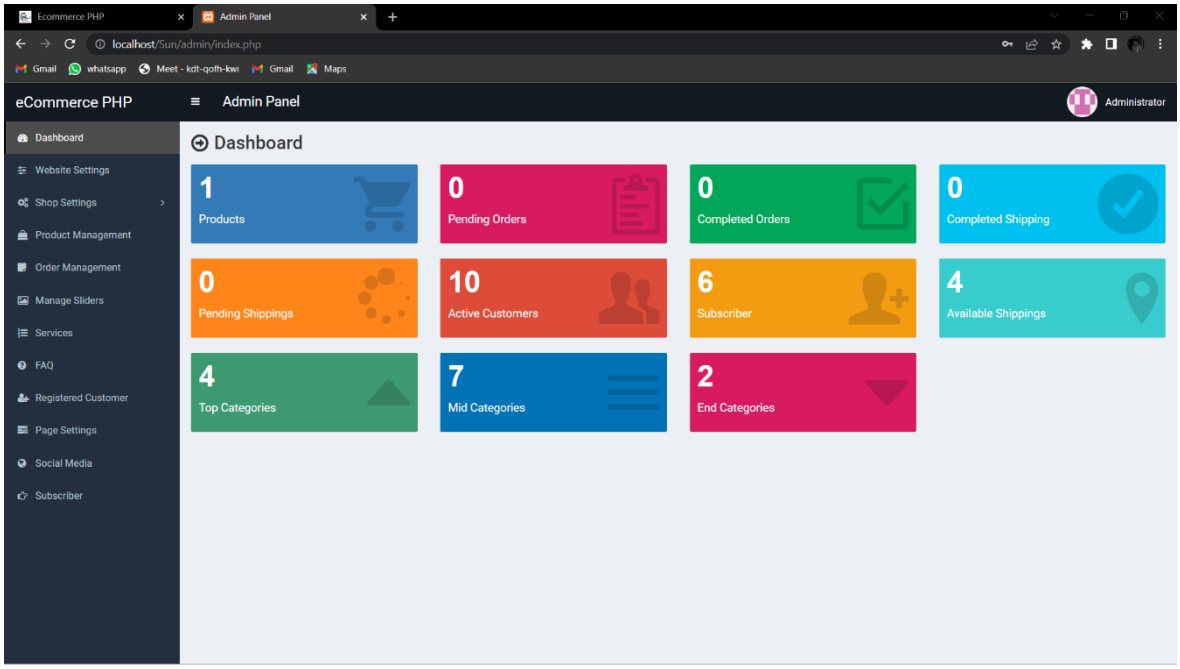
A2.5 Product Description Page



A2.6 Customer Login Page



A2.7 Cart Page



A2.8 Admin Dashboard

REFERENCES

1. Chetan Soni, Aarti Harne, Pooja Gowari, Amruta Sankhe. 'Integrated Web Based Complaint Management System' in International Journal for Scientific Research & Technology MAY/2017.
2. Dilip B. Gupta, Dr. Sujit G. Metre, "Decentralization and Delegation of Authority at Nagpur Municipal Corporation (NMC) Nagpur", International Journal of Management (IJM) Volume 6, Issue 3, March (2015),
3. Swapna kodali -the design and implementation of an e-commerce site for online book sales by– no.8 September 2015.
4. B. Sudhir, HOD Dept. Computer Applications MITS College, Madanapalle AP., S. India' Electronic Complaint Management System for Municipal Corporation' Communications on Applied Electronics (CAE) – ISSN: 23944714 Foundation of Computer Science FCS, New York, USA Volume 2 – No.8 – December/2013
5. System' International Journal of Trend in Research and Development DECEMBER/2016
6. Online Shopping Spare Parts Submitted by Mohammad Rudwan Yasin MARCH/2001.