**ABSTRACT**

This project “**E-PORTAL FOR CAR SPARES”** is a web application designed using HTML, CSS as front end and My-SQL as back end. The coding language used here is PHP

This system that enables the customers who are workshop owners to order their spare item’s for cars through online at any time and any place. The purpose of developing this website is to deliver all type of spares including new model. Also the delivery charges are free. The customer may also know that the path of delivery address, and also the delivery time from the time of ordering.

The system incorporate comprehensive tools and functionalities to actively support intelligence shop automation. The system is fabricated around the overall objectives of providing a flexible, use-specific work environment within management defined limit.

Online car spares shop goes beyond conventional shop automation to provide strategic inventory tracking and control, billing, shop profitability and sales in the shop work culture. The system recommends a facility to accept the order and home delivery system which can improve customer’s satisfaction.

**SYSTEM ANALYSIS**

**EXISTING SYSTEM**

The existing system was carried out manually. There is lot of difficulties available in the existing system.The existing online car spares shop is not completely computerized. The customer has to visit any ordering websites if they want to order furniture. Major problem was lack of security. The near matching samples of the new car spares yet to be processed cannot be view prior to ordering.

**DRAWBACKS OF EXISTING SYSTEM**

The existing system has following disadvantages,

* Highly suspected to data loss since all the process is handled as hard copies.
* Difficult to maintain customer order details in large scale.
* Less accuracy in maintaining the stock of the products
* Difficult in promoting the business online.
* Customer finds difficult to find the proper store.
* The traditional approach requires manual report generation.

**PROPOSED SYSTEM**

The proposed system is a web based application which is connected to a complete database. The database includes information about car spares materials, availability and payment etc…The proposed system is maintaining data through software. The proposed system helps in managing fast data processing and report generation. Customer can also cancel their order easily. The application gives utmost important to security and usability. The near matching samples of the new spares yet to be processed can be view prior to ordering. Comparison is possible if the new spares is under production. Item wise profit analysis is also carried out.

**ADVANTAGE OF PROPOSED SYSTEM**

The proposed system has following advantages,

* Searching for specified product is easier since it is online.
* Reduces traversal time between workshop/customer and company.
* User Interface is efficiently designed for minimum input.
* Orders would be verified online.
* Proper security is provided with authentication.

**SYSTEM SPECIFICATION**

**HARDWARE REQUIREMENTS**

This section gives the details and specification of the hardware on which the system is expected to work.

Processor : Intel I3

RAM : 8 GB DDR RAM

Monitor : 17” Color

Hard disk : 1TB

Keyboard : Standard102 keys

Mouse : Optical

**SOFTWARE REQUIREMENTS**

This section gives the details of the software that are used for the development.

Front-End : HTML, CSS

Coding Language : PHP 5.4

Back-End : My-SQL 5.6

Operating System : Windows 10

**MODULE DESCRIPTION**

The project contains following modules,

**ADMIN MODULE**

* + ADMIN LOGIN
  + SUPPLIER ENTRY
  + PRODUCT ENTRY
  + SUPPLIER LIST
  + WORKSHOP OWNERS LIST
  + PRODUCT LIST
  + PURCHASE INVOICE
  + STOCK REPORT
  + ORDER/SALES PROCESSING
  + SALES INVOICE LIST

**WORKSHOP OWNER MODULE**

* + REGISTRATION
  + LOGIN
  + SEARCH PRODUCT
  + ADD TO CART
  + MY ORDERS
  + VIEW RECEIPTS

**ADMIN LOGIN**

In this module, the admin user enters username and password. These details are checked against the ‘admin’ table and if matched login is made. Otherwise invalid username message will be displayed.

**SUPPLIER ENTRY**

This module is used for storing the supplier information. The details such as supplier is, company name, address, contact no, email id are stored in the table. These details are stored in the appropriate with the help of sql.

**ITEM CREATION**

In this module admin creates new product for sales. This product includes the information such as product id, name, purchase price, tax, selling price, rating and available stock. These details are store in database.

**SUPPLIER REPORT**

In this module admin views all the information about the saved supplier details. This includes the information such as supplier id, name, address, and contact no and email id. This information is fetched from the database.

**ITEM LIST**

In this module admin views product details. This product includes the information such as product id, name, purchase price, tax, selling price, rating and available stock. These details are taken from database.

**PURCHASE INVOICE**

In this module admin increases the product stock by purchasing the stock from the supplier. This includes the information such as seller id, product id, name, and category, purchase price, selling price, quantity and total.

**STOCK REPORT**

In this module admin views the current stock availability and product information. This includes the information such as product id, name, purchase price, tax, selling price and rating. This information is fetched from the database.

**ORDER PROCESSING**

In this module customer orders are processed by admin. Admin delivers or rejects the customer’s order after verifying the stock details. After the order has been processed the status will be mailed to the customer.

**SALES INVOICE LIST**

In this module admin views the purchase invoice list. This includes the information such as seller id, purchase value, tax value, quantity and total value.

**CUSTOMER REGISTRATION**

In this module customer register their details into the website so as make the order. This details includes the information such as customer id, name, address, contact no, email id and payment information.

**VIEW/SEARCH ITEM**

In this module customer searches for the product using the key words. This result includes the information such as product id, name, price, available stock and description. This information is fetched from the database.

**ADD TO CART**

In this module customer adds the wished product to the cart by entering the required quantity and proceeding. The cart can be modified at any time till the order placement.

**MY ORDERS**

In this module customer tracks the order status by entering the order no and check status. This will be used as a future reference for the invoice list of purchased product details.

**VIEW RECEIPTS**

In this module customer their past receipts entries made by admin. The details are taken from ‘receipt’ table.

**SYSTEM FLOW DIAGRAM**

**E-PORTAL FOR CAR SPARES**

ADMIN

USER

REGISTRATION

SUPPLIER ENTRY

ITEM CREATION

PURCHASE

CUSTOMER REPORT

ORDER APPROVAL

SEARCH PRODUCT

ADD TO CART

MY ORDERS

RECEIPT

DELIVERS ORDERS

LOW STOCK INTIMATION

VIEW RECEIPTS

FORGET PASSWORD

SUPPLIER DETAILS

PAYMENT

**DATA FLOW DIAGRAM**

**LEVEL 0**

Customer

Search Products

Make Order

Manage Products, Customer Orders

Add Item, View Customer order, invoice maintenance

Admin

**DATA FLOW DIAGRAM (ADMIN)**

Admin Table

Items

Admin

Supplier

Purchase Master/Trans

Receipt

Payment

SalesMaster/Trans

OrderDetails

**DATA FLOW DIAGRAM (CUSTOMER)**

Receipts

SalesTrans

SalesMaster

OrderDetails

Items

Customer

Customers

**DATABASE DESIGN**

The most important consideration in designing the database is how information will be used. The main objectives of designing a database are:

Data Integration

In a database, information from several files are coordinated, accessed and operated upon as through it is in a single file. Logically, the information are centralized, physically, the data may be located on different devices, connected through data communication facilities.

Data Integrity

Data integrity means storing all data in one place only and how each application to access it. This approach results in more consistent information, one update being sufficient to achieve a new record status for all applications, which use it. This leads to less data redundancy; data items need not be duplicated; a reduction in the direct access storage requirement.

Data Independence

Data independence is the insulation of application programs from changing aspects of physical data organization. This objective seeks to allow changes in the content and organization of physical data without reprogramming of applications and to allow modifications to application programs without reorganizing the physical data.

The tables needed for each module were designed and the specification of each and every column was given based on the records and details collected during record specification of the system study.

**TABLE STRUCTURE**

**Table Name** : Admin Table

**Description** : This table is used to maintain admin login details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| Username | Varchar | 15 | Primary key |
| Password | Varchar | 15 | Not Null |

**Table Name** : Supplier Table

**Description** : This table is used to maintain supplier details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| CustomerId | Varchar | 10 | Primary key |
| CustomerName | Varchar | 50 | Not Null |
| Street | Varchar | 50 |  |
| Address | Varchar | 500 | Not Null |
| City | Varchar | 50 | Not Null |
| PinCode | Varchar | 6 | Not Null |
| PhoneNo | Varchar | 12 | Not Null |
| Mobile | Varchar | 12 | Not Null |
| EmailId | Varchar | 50 | Not Null |

**Table Name** : Customer Table

**Description** : This table is used to maintain customer details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| CustomerId | Varchar | 10 | Primary key |
| CustomerName | Varchar | 50 | Not Null |
| Street | Varchar | 50 |  |
| Address | Varchar | 500 | Not Null |
| City | Varchar | 50 | Not Null |
| PinCode | Varchar | 6 | Not Null |
| PhoneNo | Varchar | 12 | Not Null |
| Mobile | Varchar | 12 | Not Null |
| EmailId | Varchar | 50 | Not Null |
| Password | Varchar | 20 | Not Null |

**Table Name** : Order Details

**Description** : This table is used to maintain order details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| OrderNo | Int | 4 | Primary key |
| OrderDate | DateTime | 10 | Not Null |
| CustomerId | Varchar | 20 | Not Null |
| ProductId | Varchar | 20 | Not Null |
| Price | Float | 4 | Not Null |
| Quantity | Float | 4 | Not Null |
| Total | Float | 4 | Not Null |

**Table Name** : PurchaseMaster Table

**Description** : This table is used to maintain purchase details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| BillNo | Int | 10 | Primary key |
| BillDate | DateTime | 8 |  |
| SupplierId | Varchar | 10 | Not Null |
| NetAmount | Double | 8 | Not Null |

**Table Name** : PurchaseTrans Table

**Description** : This table is used to maintain purchase details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| BillNo | Int | 10 | Primary key |
| SNo | Int | 10 | Not Null |
| ItemCode | Varchar | 10 | Not Null |
| Quantity | Int | 4 | Not Null |
| Rate | Double | 8 | Not Null |
| Amount | Double | 8 |  |
| Tax | Double | 8 | Not Null |
| TaxValue | Double | 8 | Not Null |
| Total | Double | 8 | Not Null |

**Table Name** : SalesMaster Table

**Description** : This table is used to maintain sales details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| BillNo | Int | 10 | Primary key |
| BillDate | DateTime | 8 |  |
| CustomerId | Varchar | 10 | Not Null |
| NetAmount | Double | 8 | Not Null |

**Table Name** : Sale Trans

**Description** : This table is used to maintain sale transaction details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **SIZE** | **CONSTRAINTS** |
| Sno | Int | 11 | Primary key |
| InvoiceNo | Int | 11 | Not Null |
| ProductId | Varchar | 20 | Not Null |
| ProductName | Varchar | 50 | Not Null |
| Price | Decimal | 8,2 | Not Null |
| Quantity | Float | 4 | Not Null |
| Total | Decimal | 8,2 | Not Null |

**TABLE NAME: PAYMENT**

**Primary Key:** PaymentNo

Purpose: It is used to store payment details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELDNAME** | **TYPE** | **SIZE** | **DESCRIPTION** |
| PaymentNo | Int | 4 | Unique |
| PaymentDate | DateTime | 8 |  |
| SupplierId | Varchar | 10 | Foreign Key |
| Amount | Decimal | 9 |  |
| PaymentMode | Varchar | 10 |  |
| Details | Varchar | 200 |  |

**TABLE NAME: RECEIPT**

**Primary Key:** ReceiptNo

Purpose: It is used to store receipt details.

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELDNAME** | **TYPE** | **SIZE** | **DESCRIPTION** |
| ReceiptNo | Int | 4 | Unique |
| ReceiptDate | DateTime | 8 |  |
| CustomerId | Varchar | 10 | Foreign key |
| Amount | Decimal | 9 |  |
| ReceiptMode | Varchar | 10 |  |
| Details | Varchar | 200 |  |