

VISVESVARAYA TECHNOLOGICAL UNIVERSITY
JNANA SANGAMA, BELAGAVI – 590 018



A Mini Project Report on
MILITARY CANTEEN MANAGEMENT SYSTEM

Submitted in partial fulfillment of the requirements as a part of the DBMS Lab for the V Semester of degree of Bachelor of Engineering in Information Science and Engineering of Visvesvaraya Technological University, Belagavi

Submitted by

**ABHIUDAY OJHA
1RN19IS007**

**AMARDEEP KUMAR
1RN19IS020**

Under the Guidance of

**Faculty Incharge
Dr. R Rajkumar
Asso. Professor
Dept. of ISE, RNSIT**

**Guide
Dr. Bhagyashree A
Asso. Professor
Dept. of ISE, RNSIT**



**Department of Information Science and Engineering
RNS Institute of Technology
Channasandra, Dr. Vishnuvardhan Road, RR Nagar Post,
Bengaluru – 560 098
2021 – 2022**

RNS Institute of Technology
Channasandra, Dr. Vishnuvardhan Road, RR Nagar Post,
Bengaluru – 560 098

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING



CERTIFICATE

This is to certify that the Mini project report entitled **MILITARY CANTEEN MANAGEMENT SYSTEM** has been successfully completed by **ABHIUDAY OJHA** bearing **USN 1RN19IS007** & **AMARDEEP KUMAR** bearing **USN 1RN19IS020**, presently V semester students of **RNS Institute of Technology** in partial fulfillment of the requirements as a part of the DBMS Laboratory for the award of the degree **Bachelor of Engineering in Information Science and Engineering** under **Visvesvaraya Technological University, Belagavi** during academic year 2021 – 2022. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements as a part of DBMS Laboratory for the said degree.

Dr. Bhagyashree A
Guide

Dr . R Rajkumar
Faculty Incharge

Dr. Suresh L
Professor and HOD

External Viva

Name of the Examiners

Signature with date

1. _____

2. _____

DECLARATION

WE, ABHIUDAY OJHA [USN: 1RN19IS007] & AMARDEEP KUMAR [USN: 1RN19IS020] student of V Semester BE, in Information Science and Engineering, RNS Institute of Technology hereby declare that the Mini project entitled **MILITARY CANTEEN MANAGEMENT SYSTEM** has been carried out by me and submitted in partial fulfillment of the requirements for the *V Semester degree of Bachelor of Engineering in Information Science and Engineering of Visvesvaraya Technological University, Belagavi* during academic year 2021-22.

Place: Bengaluru

Date:

ABHIUDAY OJHA

(1RN19IS007)

AMARDEEP KUMAR

(1RN19IS020)

ABSTRACT

The mini project is aimed at developing a desktop-based application named Military Canteen Management System for managing the inventory system of any military canteen. The Military Canteen Management System refers to the system and processes to manage the stock of military canteen with the involvement of Technology system. This system can be used to store the details of the inventory, stock maintenance, update the inventory based on the sales details, generate sales and inventory reports daily or weekly based. This mini project categorizes individual aspects for the sales and inventory management system. In this system we are solving different problems affecting direct sales management and purchase management.

Military Canteen Management System is important to ensure quality control in businesses that handle transactions resolving around consumer goods. Without proper inventory control, a large retail store may run out of stock on an important item. A good inventory management system will alert the wholesaler when it is time to record. Inventory Management System is also an important means of automatically tracking large shipment. An automated Military Canteen Management System helps to minimize the errors while recording the stock.

ACKNOWLEDGMENT

The fulfillment and rapture that go with the fruitful finishing of any assignment would be inadequate without specifying the people who made it conceivable, whose steady direction and support delegated the endeavors with success.

We would like to profoundly thank **Management of RNS Institute of Technology** for providing such a healthy environment to carry out this mini Project work.

We would like to express our thanks to our Principal **Dr. M K Venkatesha** for his support and inspired us towards the attainment of knowledge.

We wish to place on record my words of gratitude to **Dr. Suresh L** Professor and Head of the Department, Information Science and Engineering, for being the enzyme and master mind behind our mini Project work.

We would like to express our profound and cordial gratitude to our Faculty Incharge **Dr. R Rajkumar**, Associate Professor, Department of Information Science and Engineering for his valuable guidance, constructive comments and continuous encouragement throughout the mini Project work.

We would like to express our profound and cordial gratitude to our Guide **Dr. Bhagyashree A**, Associate Professor, Department of Information Science and Engineering for her valuable guidance in preparing the mini project report.

We would like to thank all other teaching and non-teaching staff of Information Science & Engineering who have directly or indirectly helped us to carry out the mini project work.

And lastly, We would hereby acknowledge and thank our parents who have been a source of inspiration and also instrumental in carrying out this min Project work.

Abhiuday Ojha
USN: 1RN19IS007

Amardeep Kumar
USN: 1RN19IS020

TABLE OF CONTENTS

CERTIFICATE DECLARATION	ii
ABSTRACT	iii
ACKNOWLEDGMENT	iv
TABLE OF CONTENTS	v
LIST OF FIGURES	vi
ABBREVIATIONS	vii
1. INTRODUCTION	viii
1.1 Background	1
1.2 Introduction to Canteen Management System	1
2. E R DIAGRAM AND RELATIONAL SCHEMA DIAGRAM	2
2.1 Description of ER Diagram	2
2.2 Description of Relational Schema Diagram	3
3. SYSTEM DESIGN	5
3.1 Table Description	5
3.2 Stored Procedure	7
3.3 Triggers	8
4. IMPLEMENTATION	9
4.1 Front-end Development	9
4.2 Back-end Development	10
4.3 User Flow Diagram	11
4.4 Discussion of code Segment	12
4.5 Discussion of Results	14
4.6 Application of project	21
5. CONCLUSION AND FUTURE ENHANCEMENT	22
5.1 Conclusion	22
5.2 Future Enhancement	22
REFERENCES	23

LIST OF FIGURES

Figure. No.	Descriptions	Page
2.1	E-R Diagram for Canteen Management System	02
2.2	Relational Schema - Canteen Management System	03
4.1	Admin Flow Diagram	11
4.2	Login Page	14
4.3	Landing Page	15
4.4	Add Product Page	15
4.5	Product Page	16
4.6	Add Employee Page	16
4.7	Employee Page	17
4.8	Add Personnel Page	17
4.9	Personnel Page	18
4.10	Point of sale Page	18
4.11	Sales invoice Page	19
4.12	Sales invoice detail	19
4.13	Sold products pages	20
4.14	Low Stock Page	20
4.15	Transaction Report Page	21

ABBREVIATIONS

API	-	Application Programming Interface
CSD	-	Canteen Store Department
CSS	-	Cascading style sheets
DBMS	-	Database Management System
ER	-	Entity Relationship
HTML	-	Hypertext Markup Language
HTTP	-	Hypertext Transfer Protocol
ID	-	Identification
JS	-	JavaScript
MVD	-	Multi Valued Dependency
PHP	-	PHP Hypertext Preprocessor
SQL	-	Structured Query Language

CHAPTER 1

INTRODUCTION

1.1 Background

A database is an organized collection of data, generally stored and accessed electronically from a computer system. Where databases are more complex they are often developed using formal design and modeling techniques.

The database management system (DBMS) is the software that interacts with end users, applications, the database itself to capture and analyze the data and provides facilities to administer the database. Often the term "database" is also used to loosely refer to any of the DBMS, the database system or an application associated with the database. The DBMS manages three important things: the data, the database engine that allows data to be accessed, locked and modified and the database schema, which defines the database's logical structure. Typical database administration tasks supported by the DBMS include change management, performance monitoring/tuning and backup and recovery. Many database management systems are also responsible for automated rollbacks, restarts and recovery as well as the logging and auditing of activity.

1.2 Introduction to Military Canteen Management System

Canteen Stores Department (CSD) is responsible for providing quality consumer goods at rates cheaper than the market rates to the Service Personnel and defence civilians. From a modest beginning six decades ago, CSD has grown rapidly with annual turnover of over ` 15000 crore during 2015-16. The CSD through its chain of one Base Depot and 34 Area Depots serves as the Wholesaler and the retail operations are carried out through about 4000 Unit Run Canteens (URCs). These URCs, some of which are in remote areas, in turn sell these goods to the ultimate beneficiaries.

Military Canteen Management System is a management system to manage CSD(canteen store department).Government of India provides grocery and liquor cards to military personnel using which they can buy grocery,liquor products at low rates.This management system allows the user(admin) to perform CRUD(create,read,update,delete) operations on entities such as working staffs, inventory products, Military personnel and tracks the sales made.This management system tracks products with low stocks and also generate sales reports.

CHAPTER 2

ER DIAGRAM AND RELATIONAL SCHEMA DIAGRAM

2.1 Description of ER Diagram

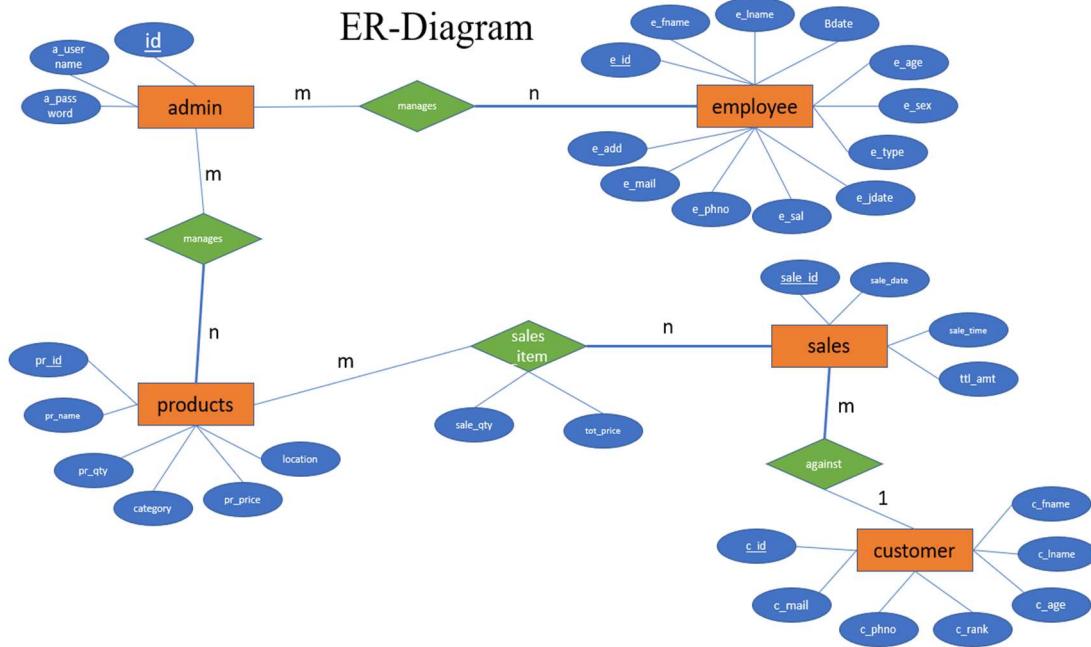


Figure 2.1 ER Diagram

Entity relationship diagram displays the relationships of entity set stored in a database. In other words, we can say that ER diagrams help you to explain the logical structure of databases. At first look, an ER diagram looks very similar to the flowchart. However, ER Diagram includes many specialized symbols, and its meanings make this model unique.

2.1.1 Components of Military Canteen Management System, E-R Diagram

1. **Entity** types like `admin` and `employee` are in rectangular boxes.
2. **Relationships** like `manages` and `sales_item` are in diamond boxes, attached to entity types with straight lines.

3. Attributes are shown in ovals like Name and id each attached by a straight line to entity or relationship type.

4. Key attributes like pr_id and c_id are underlined.

2.1.2 E-R Diagram Relationships Description

It has the following relationships:

1. admin manages employee: admin can add new employee or update or remove existing one. M admin can manage n employee.

2. admin manages products: admin can add new product or update or remove existing one. M admin can manage n products.

3. sales_item linking product and sales: it shows that some product from the product list take part in the sales. M number of products can take part in n number of sales.

4. sales against customer: : it shows that some customer from the customer list take part in the sales. Any particular sale can have only one customer against it hence m:1 relation.

2.2 Description of Relational Schema Diagram

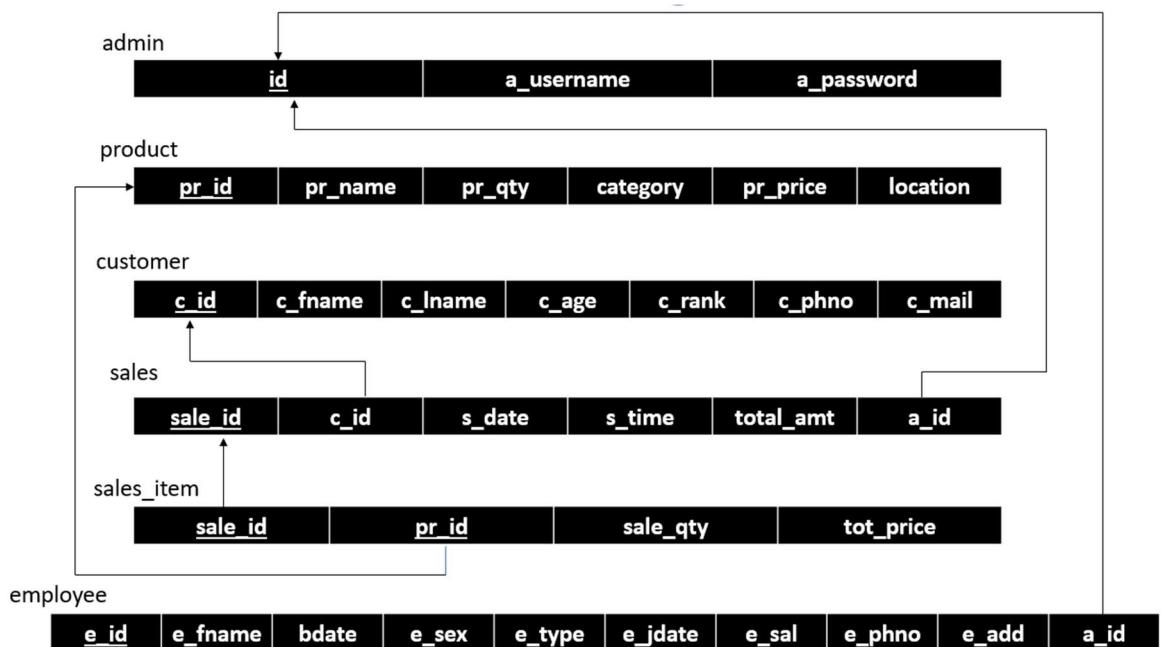


Figure 2.2 Schema Diagram

2.2.1 General Constraints

1. NULL Constraint: Attributes that are under NOT NULL constraints have to be filled compulsorily.

2. Entity Integrity Constraint: This constraint makes sure that no primary key can have a NULL value assigned to it. The primary keys involved in the project include: • id • pr_id • c_id • sale_id • e_id

3. Referential Integrity Constraints: A table in the back end of the project may have references pointing to an attribute in another table. The foreign keys involved in the project include: • sale_id in the sales_items table refers to sale_id in sales table • a_id in the sales table refers to a_id in admin table • a_id in the employee table refers to a_id in admin table • pr_id in the sales_item table refers to pr_id in product table • c_id in the sales table refers to c_id in customer table.

2.2.2 Schema Description

It has the following entities:

1. Admin: This table has attribute like admin id, username, password. Admin id is the primary key. This table provides credential for login.

2. Product: This table has attribute like product id, quantity, name, category(general, liquor and special), price and their location. Here, product id is the primary key.

3. Customer: This table has the details of the military personnel such as customer id, name, age, rank and their contact. Here, customer id is the primary key.

4. Sales: It has the details of sales with respect to customer. It contains attributes like sales id, customer id, date & time, total amount and admin id of admin who was handling it. Here, sales_id is the primary key.

5. Sales_item: It has the details of sales with respect to product and sales id(from sales). It includes other attributes such as quantity sold and total price.

6. Employee: This table has details of all the employee working in the store. It includes employee details such as employee id, name, bdate, sex, type(technician, custodian, cashier, etc) and their contact. Here, employee id is the primary key.

CHAPTER 3

SYSTEM DESIGN

3.1 Table Description

Admin

Field	Type	Null	Key	Default
ID	decimal(7,0)	NO	MUL	NULL
A_USERNAME	varchar(50)	NO	PRI	NULL
A_PASSWORD	varchar(50)	NO		NULL

Table 3.1 admin table

Customer

Field	Type	Null	Key	Default
C_ID	decimal(6,0)	NO	PRI	NULL
C_FNAME	varchar(30)	NO		NULL
C_LNAME	varchar(30)	YES		NULL
C_AGE	int(11)	NO		NULL
C_RANK	varchar(6)	NO		NULL
C_PHNO	decimal(10,0)	NO	UNI	NULL

Table 3.2 customer table

Employee

Field	Type	Null	Key	Default
E_ID	decimal(7,0)	NO	PRI	NULL
E_FNAME	varchar(30)	NO		NULL
E_LNAME	varchar(30)	YES		NULL
BDATE	date	NO		NULL
E_AGE	int(11)	NO		NULL
E_SEX	varchar(6)	NO		NULL
E_TYPE	varchar(20)	NO		NULL
E_JDATE	date	NO		NULL
E_SAL	decimal(8,2)	NO		NULL
E_PHNO	decimal(10,0)	NO		NULL
E_MAIL	varchar(40)	YES		NULL
E_ADD	varchar(40)	YES		NULL

Table 3.3 employee table

Products

Field	Type	Null	Key	Default
PR_ID	decimal(6,0)	NO	PRI	NULL
PR_NAME	varchar(50)	NO		NULL
PR_QTY	int(11)	NO		NULL
CATEGORY	varchar(20)	YES		NULL
PR_PRICE	decimal(6,2)	NO		NULL
LOCATION_RACK	varchar(30)	YES		NULL

Table 3.4 products table

Sales

Field	Type	Null	Key	Default
SALE_ID	int(11)	NO	PRI	NULL
C_ID	decimal(6,0)	NO	MUL	NULL
S_DATE	date	YES		NULL
S_TIME	time	YES		NULL
TOTAL_AMT	decimal(8,2)	YES		NULL
E_ID	decimal(7,0)	NO	MUL	NULL

Table 3.5 sales table

Sales_items

Field	Type	Null	Key	Default
SALE_ID	int(11)	NO	PRI	NULL
PR_ID	decimal(6,0)	NO	PRI	NULL
SALE_QTY	int(11)	NO		NULL
TOT_PRICE	decimal(8,2)	NO		NULL

Table 3.6 sales_item table

3.2 Stored Procedures

1. **Search_inventory:**

During the order process, we can search for the product and it will show all the details such as price, quantity left.

```

BEGIN
DECLARE prid DECIMAL(6);
DECLARE prname VARCHAR(50);
DECLARE prqty INT;
DECLARE prcategory VARCHAR(20);
DECLARE prprice DECIMAL(6,2);
DECLARE location VARCHAR(30);
DECLARE exit_loop BOOLEAN DEFAULT FALSE;
DECLARE PR_CURSOR CURSOR FOR SELECT
PR_ID,PR_NAME,PR_QTY,CATEGORY,PR_PRICE,LOCATION_RACK FROM
PRODUCTS;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET exit_loop=TRUE;
CREATE TEMPORARY TABLE IF NOT EXISTS T1 (prid decimal(6),prname
varchar(50),prqty int,prcategory varchar(20),prprice decimal(6,2),prlocation varchar(30));
OPEN PR_CURSOR;
pr_loop: LOOP
FETCH FROM PR_CURSOR INTO prid,prname,prqty,prcategory,prprice,location;
IF exit_loop THEN
LEAVE pr_loop;
END IF;

IF(CONCAT(prid,prname,prcategory,location) LIKE CONCAT('%',search,'%')) THEN
INSERT INTO T1(prid,prname,prqty,prcategory,prprice,prlocation)
VALUES(prid,prname,prqty,prcategory,prprice,location);
END IF;
END LOOP pr_loop;
CLOSE PR_CURSOR;
SELECT prid,prname,prqty,prcategory,prprice,prlocation FROM T1;
END

```

2. **Stock:**

Helps to show all the product with stock less than 50.

```

BEGIN
SELECT pr_id, pr_name,pr_qty,category,pr_price,location_rack FROM products where
pr_qty<=50;
END

```

3. **Total_amt:**

As the product is added, the total amount is automatically calculated and updated.

```
BEGIN
UPDATE SALES SET
S_DATE=SYSDATE(),S_TIME=CURRENT_TIMESTAMP(),TOTAL_AMT=(SELECT
SUM(TOT_PRICE) FROM SALES_ITEMS WHERE SALES_ITEMS.SALE_ID=ID)
WHERE SALES.SALE_ID=ID;
SELECT TOTAL_AMT INTO AMT FROM SALES WHERE SALE_ID=ID;
END
```

3.3 Trigger

1. Sale_delete:

It helps us to update products quantity when the sale is discarded.

```
BEGIN
UPDATE products SET PR_QTY=PR_QTY+old.SALE_QTY WHERE
products.PR_ID=old.PR_ID;
END
```

2. Sale_insert:

It helps us to update products quantity when the sale is done.

```
BEGIN
UPDATE products SET PR_QTY=PR_QTY-new.SALE_QTY WHERE
products.PR_ID=new.PR_ID;
END
```

4. Sale_id_delete:

It helps us to delete the sales id automatically from sales item if any sale is discarded.

```
BEGIN
DELETE from sales_items WHERE sales_items.SALE_ID=old.SALE_ID;
END
```

Chapter 4

IMPLEMENTATION

4.1 Front-end Development

The front-end is built using a combination of technologies such as Hypertext Markup Language (HTML), JavaScript and Cascading Style Sheets (CSS). Front-end developers design and construct the user experience elements on the web page or app including buttons, menus, pages, links, graphics and more.

4.1.1 Hypertext Markup Language

HTML is a computer language devised to allow website creation. These websites can then be viewed by anyone else connected to the Internet. It is relatively easy to learn, with the basics being accessible to most people in one sitting; and quite powerful in what it allows you to create. HTML is the standard markup language for creating Web pages. It stands for Hyper Text Markup Language. It describes the structure of a Web page. It consists of a series of elements. Its elements tell the browser how to display the content. Its elements are represented by tags. HTML tags label pieces of content such as "heading", "paragraph", "table", and so on. Browsers do not display the HTML tags, but use them to render the content of the page.

4.1.2 Cascading style sheets

CSS stands for 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. CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications. Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. CSS solved that issue. CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file. CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

4.1.3 JavaScript

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language

with object-oriented capabilities. Client-side JavaScript is the most common form of the language. The script should be included in or referenced by an HTML document for the code to be interpreted by the browser. It means that a web page need not be a static HTML, but can include programs that interact with the user, control the browser, and dynamically create HTML content. The JavaScript client-side mechanism provides many advantages over traditional CGI server-side scripts. The JavaScript code is executed when the user submits the form, and only if all the entries are valid, they would be submitted to the Web Server. JavaScript can be used to trap user-initiated events such as button clicks, link navigation, and other actions that the user initiates explicitly or implicitly.

4.2 Back-end Development

Backend is server side of the website. It stores and arranges data, and also makes sure everything on the client-side of the website works fine. It is the part of the website that you cannot see and interact with. It is the portion of software that does not come in direct contact with the users. The parts and characteristics developed by backend designers are indirectly accessed by users through a front-end application. Activities, like writing APIs, creating libraries, and working with system components without user interfaces or even systems of scientific programming, are also included in the backend.

4.2.1 Backend scripting language: PHP Hypertext Preprocessor

PHP is used as the server side scripting language. PHP is an acronym for "PHP: Hypertext Preprocessor". PHP is a widely-used, open source scripting language. PHP scripts are executed on the server. It is compatible with all servers used today. It is easy to use and runs efficiently on the server side. It can run on various platforms like windows, Linux, UNIX, Mac OS-X etc. and since it is a scripting language, it comes with predefined functions which makes it easy to implement any logic necessary.

4.2.2 Web Server – Apache

Apache is an open-source and free web server software that powers around 46% of websites around the world. The official name is Apache HTTP Server, and it's maintained and developed by the Apache Software Foundation. It allows website owners to serve content on the web — hence the name “web server”. Although we call Apache a web server, it is not a physical server, but rather a software that runs on a server. Its job is to establish a connection between a server and the browsers of website visitors (Firefox, Google Chrome,

Safari, etc.) while delivering files back and forth between them (client-server structure). Apache is a cross-platform software, therefore it works on both UNIX and Windows servers.

4.2.3 Database – MySQL

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. It is developed, marketed and supported by MySQL AB, which is a Swedish company. It is released under an open-source license. So you have nothing to pay to use it. It is a very powerful program in its own right. It handles a large subset of the functionality of the most expensive and powerful database packages. It uses a standard form of the well-known SQL data language. It works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, etc. It works very quickly and works well even with large data sets. It is very friendly to PHP, the most appreciated language for web development. MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB). It is customizable. The open-source GPL license allows programmers to modify the MySQL software to fit their own specific environments.

4.3 Admin Flow Diagram

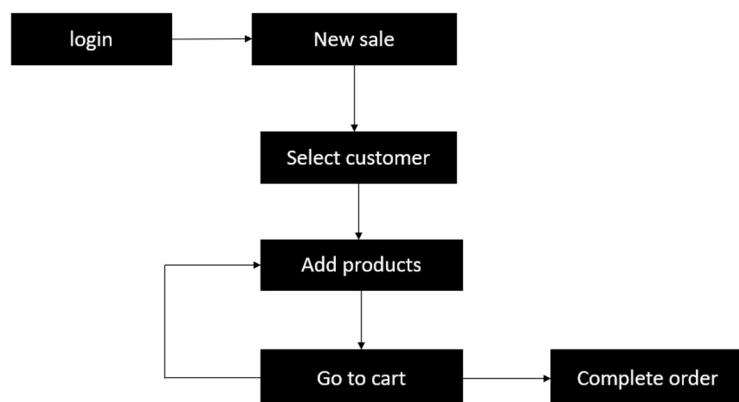


Figure 4.1 Admin Flow Diagram

4.4 Discussion of Code Segment

4.4.1 Login Module

```

if(isset($_POST['submit'])){

    $uname = mysqli_real_escape_string($conn,$_POST['uname']);
    $password = mysqli_real_escape_string($conn,$_POST['pwd']);
    if($uname != "" && $password != ""){

        $sql="SELECT * FROM admin WHERE a_username='".$uname' AND
a_password='".$password."'";
        $result = $conn->query($sql);
        $row = $result->fetch_row();
        if(!$row) {
            echo "<p style='color:red;'>Invalid username or password!</p>";
        }
        else {
            session_start();
            $_SESSION['user']=$uname;
            header('location:adminmainpage.php');
        }
    }
}

```

4.4.2 select customer

```

$qry1="SELECT id from admin where a_username='".$_SESSION[user]"';

$result1=$conn->query($qry1);
$row1=$result1->fetch_row();
$eid=$row1[0];

if(isset($_GET['sid']))
{
    $sid=$_GET['sid'];
}

```

```

if(isset($_POST['cid']))
    $cid=$_POST['cid'];

if(isset($_POST['custadd'])) {

    $qry2="INSERT INTO sales(c_id,e_id) VALUES ('$cid','$eid')";
    if(!$result2=$conn->query($qry2)) {
        echo "<p style='font-size:8; color:red;'>Invalid! Enter valid Customer ID to
record Sales.</p>";
    }
}

```

4.4.3 select product

```

$qry3="SELECT pr_name FROM products";
$result3 = $conn->query($qry3);
echo mysqli_error($conn);
if ($result3->num_rows > 0) {
    while($row4 = $result3->fetch_assoc()) {
        echo "<option>".$row4["pr_name"]."</option>";
    }
}

```

4.4.4 cart

```

if(isset($_POST['add'])) {

    $qry5="select sale_id from sales ORDER BY sale_id DESC
LIMIT 1";
    $result5=$conn->query($qry5);
    $row5=$result5->fetch_row();
    $sid=$row5[0];
    echo mysqli_error($conn);

    $mid=$_POST['medid'];
    $aqty=$_POST['mqty'];
    $qty=$_POST['mcqty'];

```

```

if($qty>$aqty||$qty==0)
{echo "QUANTITY INVALID!";}
else {
$price=$_POST['mprice']*$qty;
$qry6="INSERT INTO
sales_items(`sale_id`,`pr_id`,`sale_qty`,`tot_price`)
VALUES($sid,$mid,$qty,$price)";

$result6 = mysqli_query($conn,$qry6);
echo mysqli_error($conn);
echo "<br><br> <center>";
echo "<a class='button1 view-btn'
href=pos2.php?sid=".$sid.">View Order</a>";
echo "</center>";
}
}

```

4.5 Discussion of Results

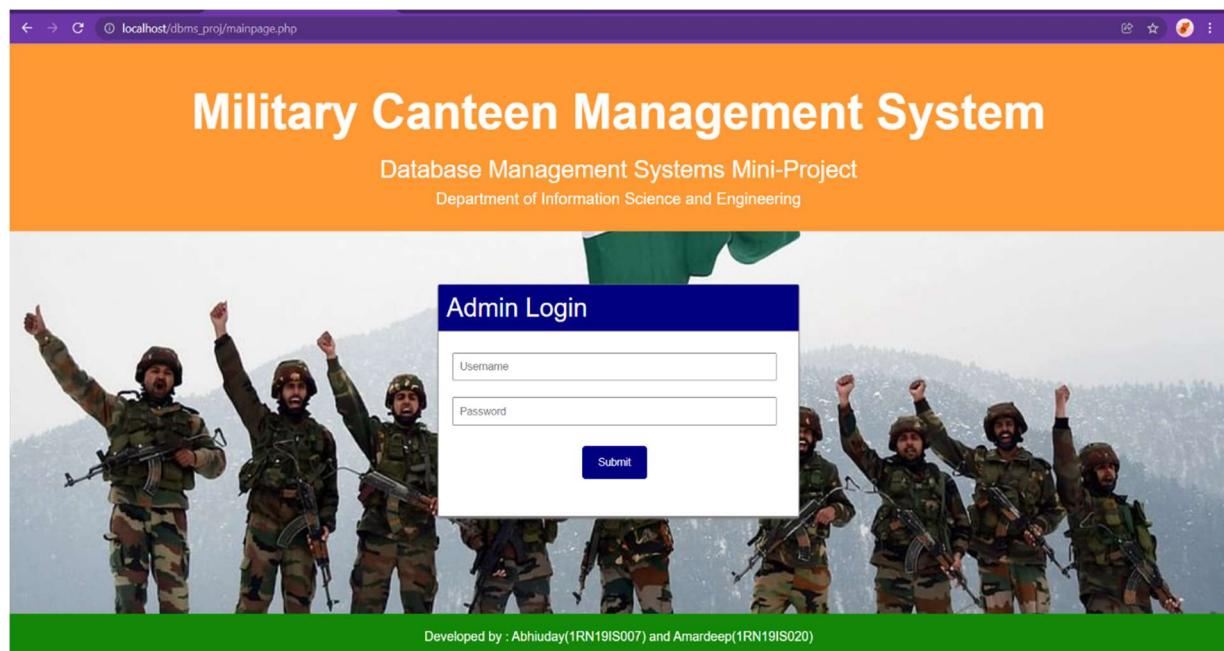


Figure 4.2 login page

Admin can login using username and password.

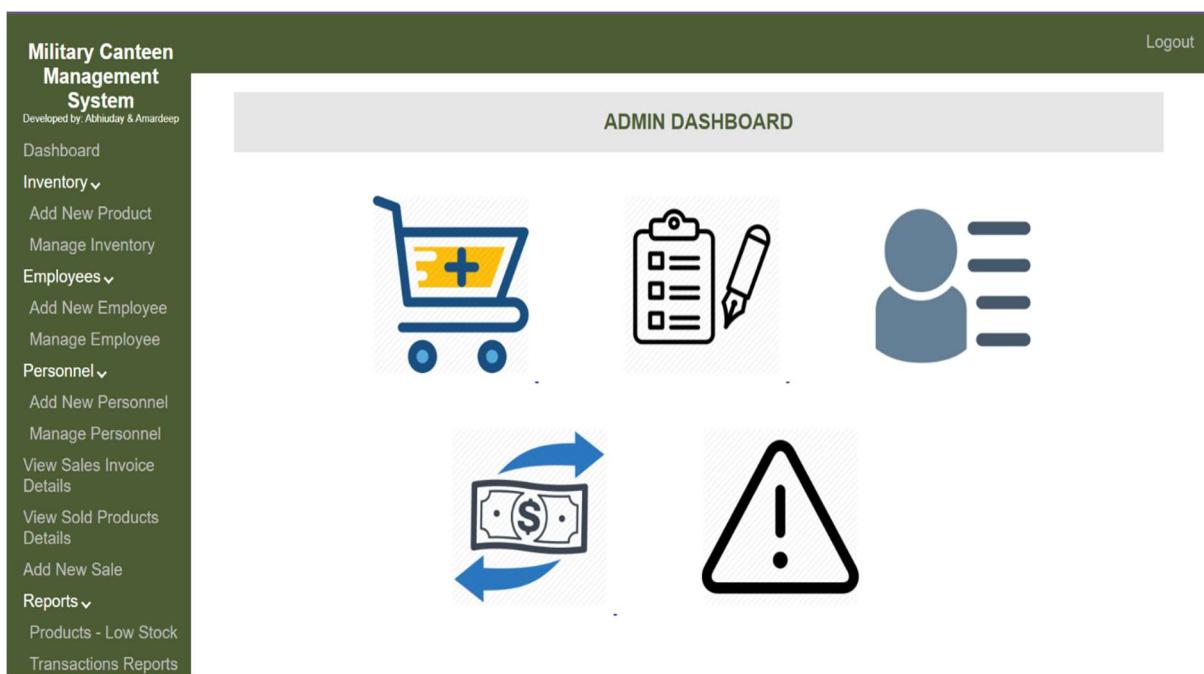


Figure 4.3 landing page

This is the landing page.

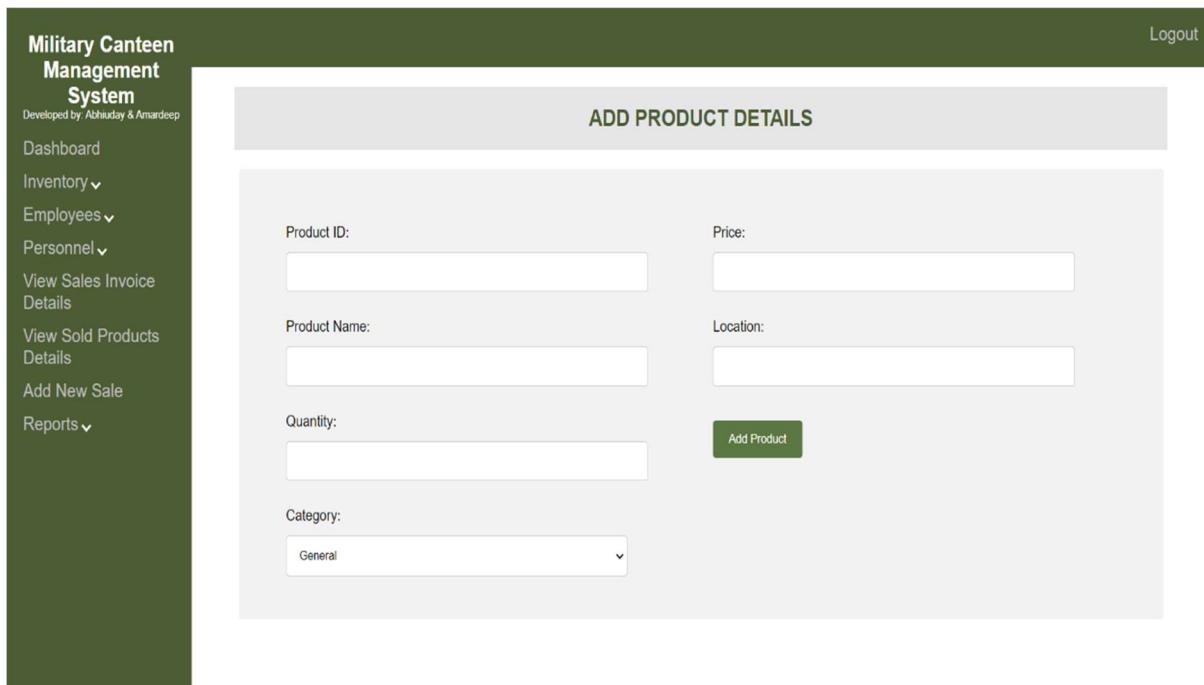


Figure 4.4 add product page

We can add new products from this page.

Product ID	Product Name	Quantity Available	Category	Price	Location in Store	Action
1111	Dettol	87	General	119.00	rack 1	<button>Edit</button> <button>Delete</button>
1112	Wall Clock	19	General	110.00	rack 8	<button>Edit</button> <button>Delete</button>
1113	Bottle	129	General	69.00	rack 3	<button>Edit</button> <button>Delete</button>
1114	Black Dog	27	Liquor	3399.00	rack 5	<button>Edit</button> <button>Delete</button>
1115	Whey Protien	89	General	600.00	rack 8	<button>Edit</button> <button>Delete</button>
1116	washing machine	1	Special	4999.00	row 5	<button>Edit</button> <button>Delete</button>
1117	Extension board	80	General	199.00	rack 6	<button>Edit</button> <button>Delete</button>
1118	Old Monk 300 ml	55	Liquor	169.00	rack 6	<button>Edit</button> <button>Delete</button>

Figure 4.5 product page

This page shows the list of all the products available.

From here, we can either delete or update products details.

Employee ID:	Employee Type:
<input type="text"/>	<input type="select"/>
First Name:	Date of Joining:
<input type="text"/>	<input type="date"/>
Last Name:	Salary:
<input type="text"/>	<input type="text"/>
Date of Birth:	Phone Number:
<input type="date"/>	<input type="text"/>
Age:	Email ID:
<input type="text"/>	<input type="text"/>
Sex:	Address:

Figure 4.6 add employee page

We can add new employee from this page.

Employee ID	First Name	Last Name	Date of Birth	Age	Sex	Employee Type	Date of Joining	Salary	Phone Number	Email Address	Home Address	Action
2	Amardeep	Kumar	1999-07-03	22	Male	Admin	2014-01-01	95000.00	9660656269	amar@gmail.com	rr nagar, bangalore	<button>Edit</button> <button>Delete</button>
6789001	Aditya	kumar	1994-01-06	28	Male	Custodian	2015-04-24	52000.00	9187678234	aditya@gmail.com	malviya nagar, new delhi	<button>Edit</button> <button>Delete</button>
6789002	Harsh	rajput	1999-02-24	22	Male	Technician	2019-10-25	22000.00	9467834109	harsh@gmail.com	salt lake, kolkata	<button>Edit</button> <button>Delete</button>
6789003	Ragini	Vishwa	1996-11-07	25	Female	Cashier	2021-03-22	27000.00	9729371269	ragini@gmail.com	boring road, patna	<button>Edit</button> <button>Delete</button>

Figure 4.7 employee page

This page shows the list of all the employee.

From here, we can either delete or update employee details.

The form consists of several input fields:

- Personnel ID:
- Phone Number:
- First Name:
- Email ID:
- Last Name:
- Add Personnel:
- Age:
- Rank:

Figure 4.8 add personnel page

We can add customer(personnel) from this page.

The screenshot shows the 'PERSONNEL LIST' page. On the left is a sidebar with navigation links. The main area displays a table with columns: Personnel ID, First Name, Last Name, Age, Rank, Phone Number, Email Address, and Action (with 'Edit' and 'Delete' buttons). The data in the table is as follows:

Personnel ID	First Name	Last Name	Age	Rank	Phone Number	Email Address	Action
987101	Safia	Malik	22	Major	9632587415	safia@gmail.com	<button>Edit</button> <button>Delete</button>
987102	Varun	Narula	24	Lieute	9987565423	varun@gmail.com	<button>Edit</button> <button>Delete</button>
987103	Suresh	Rathor	45	Subeda	7896541236	suresh@gmail.com	<button>Edit</button> <button>Delete</button>
987104	Ankit	Raj	30	Brigad	7845129635	ankit@gmail.com	<button>Edit</button> <button>Delete</button>
987105	Sayed	Shah	40	Marsha	6789541235	Sayed@gmail.com	<button>Edit</button> <button>Delete</button>
987106	Vijay	Kumar	60	Colone	8996574123	vijayk@gmail.com	<button>Edit</button> <button>Delete</button>
987107	Meera	Das	35	Captai	7845963259	meera@gmail.com	<button>Edit</button> <button>Delete</button>

Figure 4.9 personnel page

This page shows the list of all the customer.

From here, we can either delete or update personnel details.

The screenshot shows the 'POINT OF SALE' page. On the left is a sidebar with navigation links. The main area has two search boxes: one for Product ID ('987107') with an 'Add to Proceed' button, and another for Product Name ('Wall Clock') with a 'Search' button. Below these are input fields for Product ID (1112), Category (1), Quantity Available (19), Product Name (1), Location (1), and Price of One Unit (110.00). There is also a 'Quantity Required' input field and an 'Add Product' button.

Figure 4.10 point of sale page

Here, we can add products to the cart.

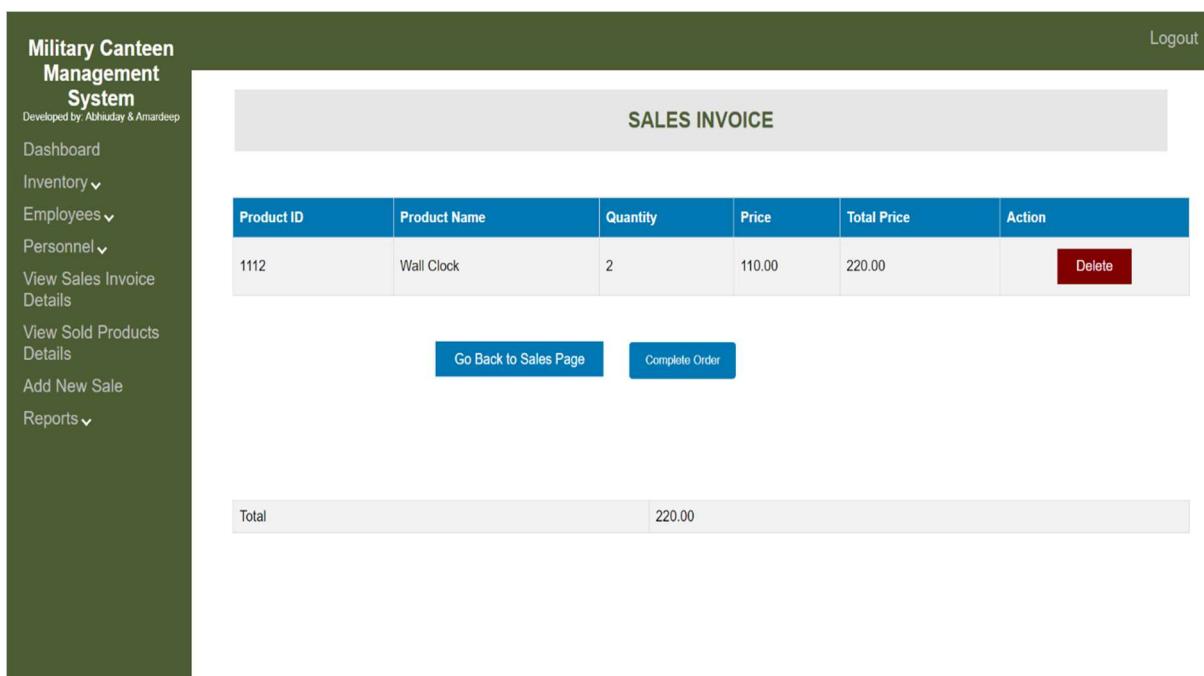


Figure 4.11 sales invoice page

This is the cart showin all the products.

From here, we can go back to add more products or complete the order.

Sale ID	Personnel ID	Date and Time	Sale Amount	Admin ID
24	987105	2022-01-31 20:03:38	957.00	2
25	987104	2022-01-31 20:04:03	4999.00	2
26	987107	2022-01-31 20:04:44	7643.00	2
27	987103	2022-01-31 20:05:21	725.00	2
28	987106	2022-01-31 20:07:13	1702.00	2
29	987101	2022-01-31 20:07:56	10197.00	2
30	987107			2
31	987102	2022-01-31 20:28:35	19999.98	2

Figure 4.12 sales invoice page

This is the page showing sales with respect to customer.

Sale ID	Product ID	Product Name	Quantity Sold	Total Price
24	1111	Dettol	3	357.00
24	1115	Whey Protein	1	600.00
25	1116	washing machine	1	4999.00
26	1114	Black Dog	2	6798.00
26	1118	Old Monk 300 ml	5	845.00
27	1121	N-95 mask	23	345.00
27	1122	Hand Sanitizer	20	380.00
28	1112	Wall Clock	1	110.00
28	1117	Extension board	8	1592.00
29	1114	Black Dog	3	10197.00
31	1123	LG double door ref.	2	19999.98

Figure 4.13 sold products page

This is the page showing sales with respect to products.

Product ID	Product Name	Quantity Available	Category	Price
1112	Wall Clock	19	General	110.00
1114	Black Dog	27	Liquor	3399.00
1116	washing machine	1	Special	4999.00
1122	Hand Sanitizer	13	General	19.00
1123	LG double door ref.	2	Special	9999.99

Figure 4.14 low stock page

This is the report page showing all the products with stock less than 50.

The screenshot shows a web-based application for managing a military canteen. The left sidebar contains a navigation menu with options like Dashboard, Inventory, Employees, Personnel, View Sales Invoice Details, View Sold Products Details, Add New Sale, and Reports. The main content area is titled "TRANSACTION REPORTS". It features two date input fields for "Start Date" and "End Date", both set to "dd-mm-yyyy" and showing "2022-01-31". Below these is a blue "View Records" button. To the right is a table with the following data:

Sale ID	Personnel ID	Admin ID	Date	Sale Amount(in Rs)
24	987105	2	2022-01-31	957.00
25	987104	2	2022-01-31	4999.00
26	987107	2	2022-01-31	7643.00
27	987103	2	2022-01-31	725.00
28	987106	2	2022-01-31	1702.00
29	987101	2	2022-01-31	10197.00
31	987102	2	2022-01-31	19999.98
Total				Rs.46222.98

Figure 4.15 transaction report page

This is the report page showing the transaction between the specified interval of time.

4.6 Application of project

These types of Canteen Management System are examples of inventory management system. These helps admin to manage all the different areas of a canteen system such as customer, employee and product and their sales.

Major applications of Canteen Management System are:

- All data are available at one place.
- Easy access to inventory and managing low stock product.
- Easy management of employee working in the csd.
- Supports report generation.
- Subsidized goods available for all mmilitary personnels at one place.
- Ease in placing special orders for bigger goods.
- Tracking of all the employees and their salary.

CHAPTER 5

Conclusion and Future Enhancement

5.1 Conclusion

- This project was an attempt to make the structure and working of an Canteen Management System simpler and user-friendly.
- This was an attempt to make it similar to the real world implementation.
- In this scenario, all the undertakings of the Canteen Management System was achieved in a constructive manner.
- Given the right guidance and support its applications and availability can be enhanced.

5.2 Future Enhancements

- Making the UI more interactive.
- Adding new features such as management of buying products (which are low stock) from the sellers.
- Making the project more real time based.
- Hosting it on an online web server.

References

- [1] Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, McGRAW HILL, 3rd Edition.
- [2] Ramez Elmasri and Shamkant B. Navathe, Fundamentals of Database Systems, Pearson, 7th Edition.
- [3] www.stackoverflow.com
- [4] www.mysql.com/doc
- [5] <https://docs.microsoft.com>
- [6] <https://www.geeksforgeeks.org>
- [7] <https://www.rapidapi.org>