1.INTRODUCTION

Alcohol Mart is a comprehensive e-commerce platform designed to revolutionise the online sale of alcoholic beverages. With a focus on user convenience and a broad selection of products, our website aims to cater to both casual consumers and connoisseurs seeking diverse and high quality options. The core of this project revolves around a robust database management system (DBMS) that ensures efficient handling of various aspects of the business, including order fulfilment. The Alcohol Mart platform integrates several advanced features such as personalised recommendations and contact support. One of the key objectives of Alcohol Mart is to significantly improve the customer experience. This is achieved through a user-friendly interface, quick and reliable order processing, and tailored product recommendations based on customer preferences.

Our DBMS project plays a crucial role in facilitating these enhancements by maintaining data integrity, optimising performance, and enabling sophisticated data analytics to understand and anticipate customer needs. This project outlines the development and implementation of the Alcohol Mart e-commerce website, focusing on the database design, entity-relationship modelling, and the SQL queries that power the application. Through this endeavour, we aim to demonstrate the critical role of DBMS in modern e-commerce solutions, highlighting its importance in maintaining data integrity, optimising performance, and enhancing user satisfaction.

2.OBJECTIVES

The aim of the Alcohol Mart Database Management System (DBMS) is to create a comprehensive and efficient solution tailored to the needs of a retail store specializing in alcoholic beverages. This system aims to streamline various aspects of the store's operations, customer relationship management, and reporting. The objective is to automate product information, and pricing details. Furthermore, It aims to enhance customer engagement and satisfaction through personalized marketing initiatives and efficient customer service. Overall, it aims to improve operational efficiency, enhance data-driven decision-making, and drive business growth in the competitive retail landscape focused on alcoholic beverages.

3. HARDWARE AND SOFTWARE REQUIREMENT

3.1 Hardware Requirement

- Processor: Multi-core processor (Intel Core i5 or equivalent)
- RAM: Minimum 8 GB RAM (16 GB recommended for optimal performance)
- Storage: SSD storage recommended for faster data

3.2 Software Requirement

- Operating System: Windows server, Linux, MacOS
- Database management system: MariaDb, Mysql

4. DATABASE DESIGN

4.1 ER-Diagram

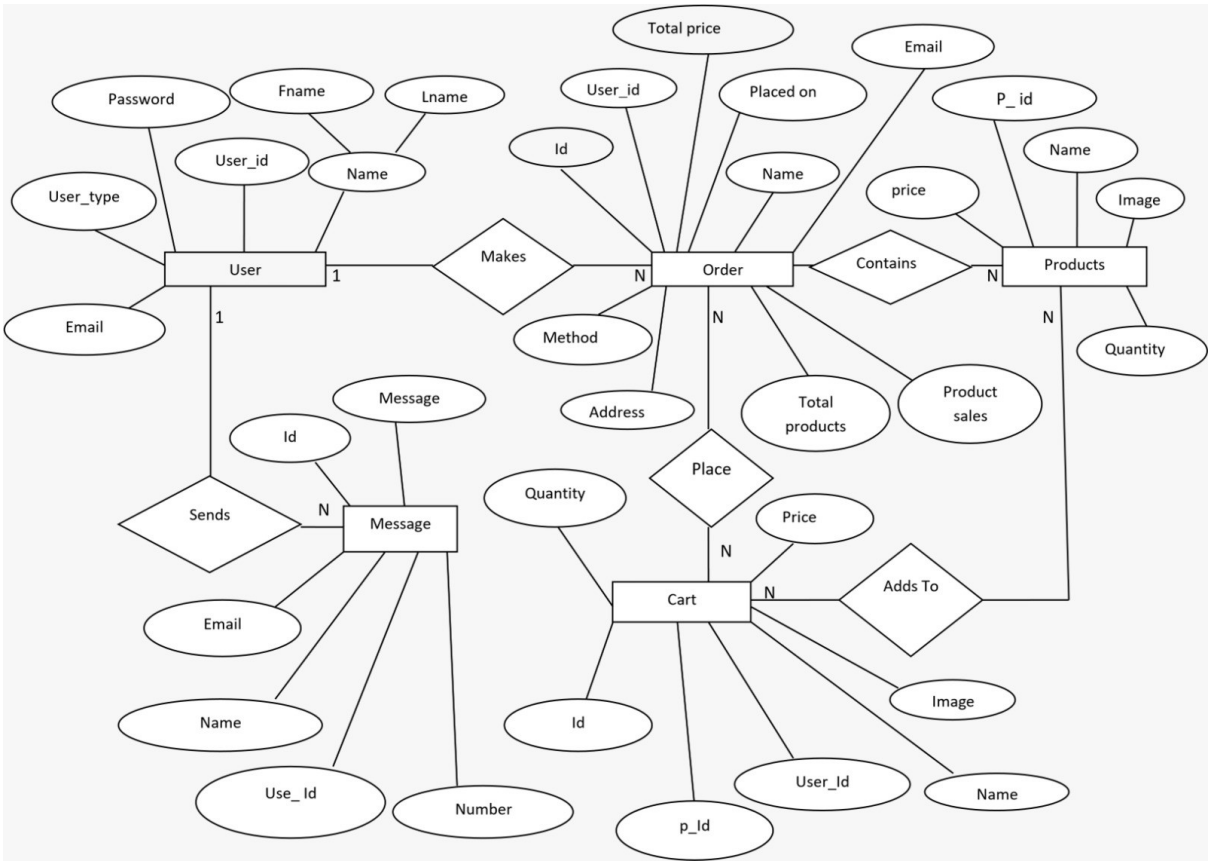

Figure 4.1: ER-Diagram

4.2 Schema Diagram

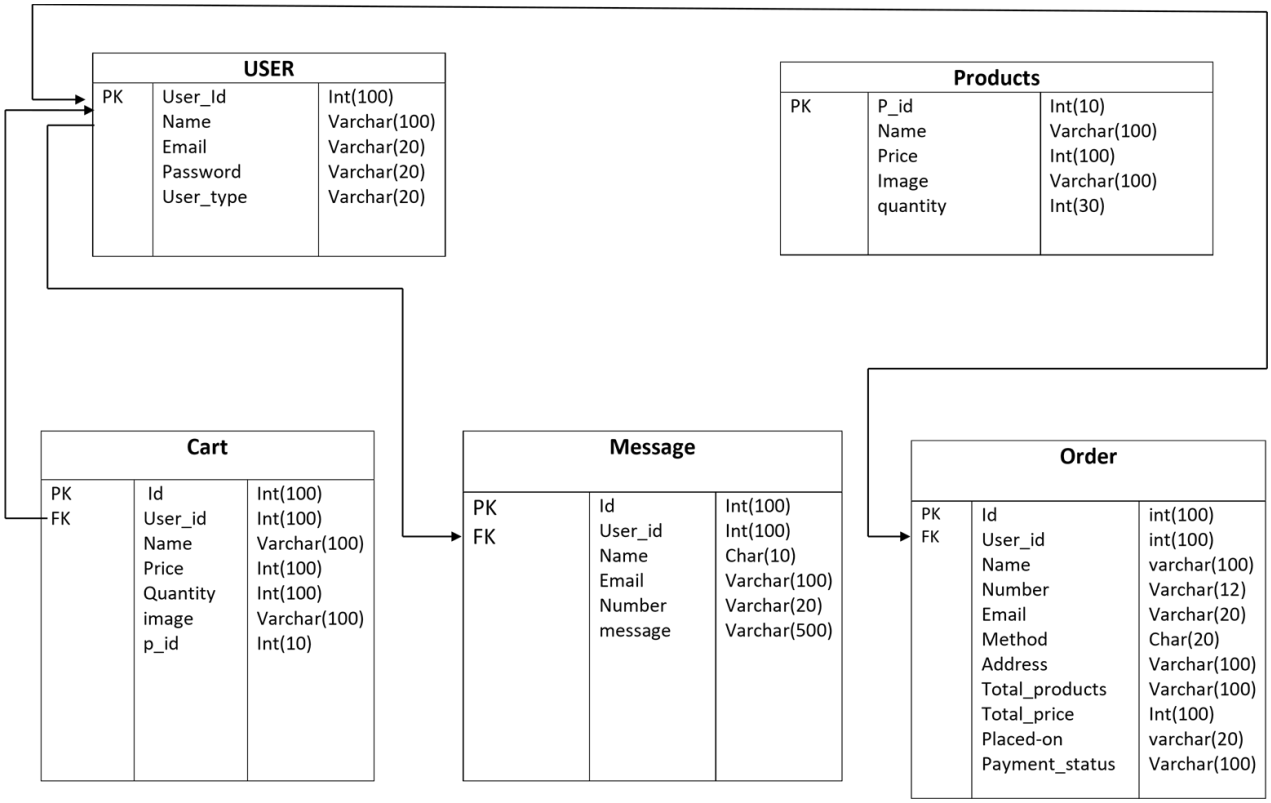

Figure 4.2: Schema Diagram

5. QUERIES AND REPORTS

Create database user_db;

use user_db;

CART TABLE:-

CREATE TABLE `cart` (`id` int(100) NOT NULL, `user_id` int(100) NOT NULL, `name` varchar(100) NOT NULL, `price` int(100) NOT NULL, `quantity` int(100) NOT NULL, `image` varchar(100) NOT NULL);

INSERTION INTO CART TABLE:-

INSERT INTO `cart` (`id`, `user_id`, `name`, `price`, `quantity`, `image`) VALUES (59, 3, 'KINGFISHER STRONG BEER', 110, 1, 'p1.jpeg'), (60, 3, 'Kingfisher Premium', 90, 1, 'p2.jpeg');

select * from cart;

```
MariaDB [user_db]> select * from cart;
+----+---------+-----------------------+-------+----------+---------+
| id | user_id | name                  | price | quantity | image   |
+----+---------+-----------------------+-------+----------+---------+
| 59 |       3 | KINGFISHER STRONG BEER |   110 |        1 | p1.jpeg |
| 60 |       3 | Kingfisher Premium     |    90 |        1 | p2.jpeg |
+----+---------+-----------------------+-------+----------+---------+
```

MESSAGE TABLE:-

CREATE TABLE `message` (`id` int(100) NOT NULL, `user_id` int(100) NOT NULL, `name` varchar(100) NOT NULL, `email` varchar(100) NOT NULL, `number` varchar(12) NOT NULL, `message` varchar(500) NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

INSERTION INTO MESSAGE TABLE:-

INSERT INTO `message` (`id`, `user_id`, `name`, `email`, `number`, `message`) VALUES (10, 1, 'CANVIL Lobo', 'canvil1212@gmail.com', '09019826618', ‘hi');

select * from message;

```
MariaDB [user_db]> select * from message;
+----+---------+-------------+----------------------+-------------+---------+
| id | user_id | name        | email                | number      | message |
+----+---------+-------------+----------------------+-------------+---------+
| 10 |       1 | CANVIL Lobo | canvil1212@gmail.com | 09019826618 | hi      |
+----+---------+-------------+----------------------+-------------+---------+
```

ORDERS TABLE:-

CREATE TABLE `orders` (`id` int(100) NOT NULL, `user_id` int(100) NOT NULL, `name` varchar(100) NOT NULL, `number` varchar(12) NOT NULL, `email` varchar(100) NOT NULL, `method` varchar(50) NOT NULL, `address` varchar(500) NOT NULL, `total_products` varchar(1000) NOT NULL, `total_price` int(100) NOT NULL, `placed_on` varchar(50) NOT NULL, `payment_status` varchar(20) NOT NULL DEFAULT 'pending');

INSERTION INTO ORDERS TABLE:-

INSERT INTO `orders` (`id`, `user_id`, `name`, `number`, `email`, `method`, `address`, `total_products`, `total_price`, `placed_on`, `payment_status`) VALUES (10, 1, 'canvil', '9019826618', 'canvil1212@gmail.com', 'cash on delivery', 'flat no. 54, kunjthabail, mangalore, India - 66666', ', Kingfisher Premium (1) ', 90, '02-May-2024', 'pending'), (11, 3, 'joyal', '0080080808', 'joyal1212@gmail.com', 'cash on delivery', 'flat no. 54, kunjthabail, mangalore, India - 66666', ', KINGFISHER STRONG BEER (1) ', 110, '02-May-2024', 'pending');

select * from orders;

```
MariaDB [user_db]> select * from orders;
+----+---------+--------+------------+---------------------+-----------------+-----------------------------------------------------+-----------------------------+-------------+-------------+-----------
-----+
| id | user_id | name   | number     | email               | method          | address                                             | total_products              | total_price | placed_on   | payment_st
atus |
+----+---------+--------+------------+---------------------+-----------------+-----------------------------------------------------+-----------------------------+-------------+-------------+-----------
-----+
| 10 |       1 | canvil | 9019826618 | canvil1212@gmail.com | cash on delivery | flat no. 54, kunjthabail, mangalore, India - 66666 | , Kingfisher Premium (1)    |          90 | 02-May-2024 | pending
    |
| 11 |       3 | joyal  | 0080080808 | joyal1212@gmail.com  | cash on delivery | flat no. 54, kunjthabail, mangalore, India - 66666 | , KINGFISHER STRONG BEER (1) |         110 | 02-May-2024 | pending
    |
| 12 |       1 | canvil | 9019826618 | canvil1212@gmail.com | cash on delivery | flat no. 54, kunjthabail, mangalore, India - 66666 | , Kingfisher Premium (1)    |          90 | 02-May-2024 | pending
    |
+----+---------+--------+------------+---------------------+-----------------+-----------------------------------------------------+-----------------------------+-------------+-------------+-----------
-----+
```

PRODUCTS TABLE:-

CREATE TABLE `products` (`id` int(100) NOT NULL, `name` varchar(100) NOT NULL, `price` int(100) NOT NULL, `image` varchar(100) NOT NULL) ;

INSERTION INTO PRODUCTS TABLE:-

INSERT INTO `products` (`id`, `name`, `price`, `image`) VALUES (4, 'Kingfisher Premium', 90, 'p2.jpeg'), (5, 'KINGFISHER STRONG BEER', 110, 'p1.jpeg'), (6, 'Heineken Beer, 330 ml ', 160, 'p4.jpeg'), (7, 'Corona Extra Premium Beer, 330 ml', 200, 'p5.jpeg'), (8, 'Black & White Blended Whisky 1L', 1790, 'p6.jpeg'), (9, 'Johnnie Walker Black Label Aged 12 YO Blended Scotch', 4500, 'p7.jpeg'), (10, 'Signature Rare Aged Whisky, 750 Ml', 900, 'p8.jpeg'), (11, 'Johnnie Walker Red Label Blended Scotch Whisky 1L', 2490, 'p9.jpeg'), (12, 'Magic Moments Vodka 750 Ml', 800, 'p10.jpeg');

select * from products;

```
MariaDB [user_db]> select * from products;
+----+-------------------------------------------------------+-------+----------+
| id | name                                                  | price | image    |
+----+-------------------------------------------------------+-------+----------+
|  4 | Kingfisher Premium                                    |    90 | p2.jpeg  |
|  5 | KINGFISHER STRONG BEER                                |   110 | p1.jpeg  |
|  6 | Heineken Beer, 330 ml                                 |   160 | p4.jpeg  |
|  7 | Corona Extra Premium Beer, 330 ml                     |   200 | p5.jpeg  |
|  8 | Black & White Blended Whisky 1L                       |  1790 | p6.jpeg  |
|  9 | Johnnie Walker Black Label Aged 12 YO Blended Scotch  |  4500 | p7.jpeg  |
| 10 | Signature Rare Aged Whisky, 750 Ml                    |   900 | p8.jpeg  |
| 11 | Johnnie Walker Red Label Blended Scotch Whisky 1L     |  2490 | p9.jpeg  |
| 12 | Magic Moments Vodka 750 Ml                            |   800 | p10.jpeg |
| 13 | Kingfisher                                            |    80 | p2.jpeg  |
+----+-------------------------------------------------------+-------+----------+
```

USERS TABLE:-

CREATE TABLE `users` (`id` int(100) NOT NULL, `name` varchar(100) NOT NULL, `email` varchar(100) NOT NULL, `password` varchar(100) NOT NULL, `user_type` varchar(20) NOT NULL DEFAULT 'user') ;

INSERTION INTO USERS TABLE:-

INSERT INTO `users` (`id`, `name`, `email`, `password`, `user_type`) VALUES (1, 'canvil', 'canvil1212@gmail.com', '202cb962ac59075b964b07152d234b70', 'user'), (2, 'admin', 'admin1212@gmail.com', '202cb962ac59075b964b07152d234b70', 'admin'), (3, 'joyal', 'joyal1212@gmail.com', '202cb962ac59075b964b07152d234b70', 'user');

select * from users;

```
MariaDB [user_db]> select * from users;
+----+----------+----------------------+----------------------------------+-----------+
| id | name     | email                | password                         | user_type |
+----+----------+----------------------+----------------------------------+-----------+
|  1 | canvil   | canvil1212@gmail.com | 202cb962ac59075b964b07152d234b70 | user      |
|  2 | admin    | admin1212@gmail.com  | 202cb962ac59075b964b07152d234b70 | admin     |
|  3 | joyal    | joyal1212@gmail.com  | 202cb962ac59075b964b07152d234b70 | user      |
|  4 | lobo     | lobo1212@gmail.com   | 123                              | user      |
|  6 | John Doe | johndoe@example.com  | password123                      | user      |
|  7 | John Doe | johndoe@example.com  | password123                      | user      |
+----+----------+----------------------+----------------------------------+-----------+
```

1)Display the IDs, names, and images of products in the shopping cart? Perform an inner join between the `cart` and `products` tables based on matching product prices. Display the IDs, names, and images of products that have the same price in both tables. Provide the SQL query and results.?

Queries:-

SELECT cart.ID, products.name, products.image FROM cart INNER JOIN products ON cart.price = products.price;

```
+----+-----------------------+---------+
| ID | name                  | image   |
+----+-----------------------+---------+
| 60 | Kingfisher Premium    | p2.jpeg |
| 59 | KINGFISHER STRONG BEER | p1.jpeg |
+----+-----------------------+---------+
```

2)Which users have placed orders and what are their names? Perform a left join between the users and orders tables based on matching email addresses. Display user IDs and corresponding order names, showing users who have placed orders and those who haven't?

Queries:-

SELECT users.id,orders.name FROM users LEFT JOIN orders ON users.email = orders.email;

```
+----+--------+
| id | name   |
+----+--------+
|  1 | canvil |
|  3 | joyal  |
|  1 | canvil |
|  2 | NULL   |
|  4 | NULL   |
|  6 | NULL   |
|  7 | NULL   |
+----+--------+
```

3)Display the names and prices of products that have been added to the shopping cart? Perform a right join between the products and cart tables based on matching product images. Display all products from the cart along with their corresponding prices?

Queries:-

SELECT products.name,cart.price FROM products RIGHT JOIN cart ON products.image = cart.image;

```
+------------------------+-------+
| name                   | price |
+------------------------+-------+
| Kingfisher Premium     |    90 |
| KINGFISHER STRONG BEER |   110 |
| Kingfisher             |    90 |
+------------------------+-------+
```

4) How many unique customers have placed orders less than three times? Provide the count of orders and the customer names?

Queries:-

SELECT COUNT(id), name FROM orders GROUP BY name HAVING COUNT(user_id) < 3;

```
+-----------+--------+
| COUNT(id) | name   |
+-----------+--------+
|         2 | canvil |
|         1 | joyal  |
+-----------+--------+
```

5)Retrieve orders sorted by total price in descending order. Display all order details including customer name, email, products, and payment status. Provide the SQL query and the resulting table showing orders with the highest total prices at the top?

Queries:-

SELECT * FROM orders ORDER BY total_price DESC;

```
MariaDB [user_db]> SELECT *  FROM orders ORDER BY total_price DESC;
+----+---------+--------+------------+---------------------+-----------------+-----------------------------------------------------------------+-----------------------------+-------------+-------------+-----------
-----+
| id | user_id | name   | number     | email               | method          | address                                                         | total_products              | total_price | placed_on   | payment_st
atus |
+----+---------+--------+------------+---------------------+-----------------+-----------------------------------------------------------------+-----------------------------+-------------+-------------+-----------
-----+
| 11 |       3 | joyal  | 0080080808 | joyal1212@gmail.com  | cash on delivery | flat no. 54, kunjthabail, mangalore, India - 66666 | , KINGFISHER STRONG BEER (1)  |         110 | 02-May-2024 | pending
     |
| 10 |       1 | canvil | 9019826618 | canvil1212@gmail.com | cash on delivery | flat no. 54, kunjthabail, mangalore, India - 66666 | , Kingfisher Premium (1)      |          90 | 02-May-2024 | pending
     |
| 12 |       1 | canvil | 9019826618 | canvil1212@gmail.com | cash on delivery | flat no. 54, kunjthabail, mangalore, India - 66666 | , Kingfisher Premium (1)      |          90 | 02-May-2024 | pending
     |
+----+---------+--------+------------+---------------------+-----------------+-----------------------------------------------------------------+-----------------------------+-------------+-------------+-----------
-----+
```

6)Using the `orders` table, write a SQL query to calculate the total number of orders (`num_orders`) and the combined total spent (`total_spent`) for each unique customer (`customer_name`). Provide the resulting customer names along with their respective counts of orders and total spending?

Queries:-

SELECT name AS customer_name, COUNT(id) AS num_orders, SUM(total_price) AS total_spent FROM orders GROUP BY name;

```
+---------------+------------+-------------+
| customer_name | num_orders | total_spent |
+---------------+------------+-------------+
| canvil        |          2 |         180 |
| joyal         |          1 |         110 |
+---------------+------------+-------------+
```

7)Create a stored procedure named `UserLogin` that checks if a user exists with a given email and validates the password. If the user exists and the password matches, return 'Login successful'; otherwise, return 'Login failed: Incorrect password' or 'Login failed: User not found'?

Queries:-

```
DELIMITER //
CREATE PROCEDURE UserLogin(
    IN p_email VARCHAR(100),
    IN p_password VARCHAR(100),
    OUT p_result VARCHAR(255)
)
BEGIN
    DECLARE stored_password VARCHAR(100);
    DECLARE user_exists INT;
    -- Check if the user exists and retrieve the password
    SELECT COUNT(*), password INTO user_exists, stored_password
    FROM users
    WHERE email = p_email;
    -- Validate user existence and password
    IF user_exists = 1 THEN
        IF stored_password = p_password THEN
            SET p_result = 'Login successful';
        ELSE
            SET p_result = 'Login failed: Incorrect password';
        END IF;
    ELSE
        SET p_result = 'Login failed: User not found';
    END IF;
END //
DELIMITER ;
```

-- Declare a variable to hold the output SET @result = '';

```
MariaDB [user_db]> CALL UserLogin('nonexistentuser@gmail.com', 'anyPassword', @result);
Query OK, 1 row affected (0.004 sec)

MariaDB [user_db]> SELECT @result;
+------------------------------+
| @result                      |
+------------------------------+
| Login failed: User not found |
+------------------------------+
1 row in set (0.000 sec)

MariaDB [user_db]> CALL UserLogin('canvil1212@gmail.com', 'wrongpassword', @result);
Query OK, 1 row affected (0.000 sec)

MariaDB [user_db]> SELECT @result;                                                  +----------------------
| @result                        |
+--------------------------------+
| Login failed: Incorrect password |
+--------------------------------+
1 row in set (0.000 sec)

MariaDB [user_db]> CALL UserLogin('canvil1212@gmail.com','202cb962ac59075b964b07152d234b70', @result);
Query OK, 1 row affected (0.001 sec)

MariaDB [user_db]> SELECT @result;                                                  +------------------+
| @result          |
+------------------+
| Login successful |
+------------------+
1 row in set (0.000 sec)

MariaDB [user_db]>
```

8)Describe a trigger query that logs new user additions into a `user_log` table. When a user is inserted into the `users` table, this trigger captures the user's name, email, and a message indicating the addition, storing these details along with a timestamp in the `user_log` table?

Queries:-

```
CREATE TRIGGER `after_insert_user`
AFTER INSERT ON `users`
FOR EACH ROW
BEGIN
   DECLARE user_message VARCHAR(500);
   SET user_message = CONCAT('New user added: ', NEW.name, ' (Email: ', NEW.email,
')');
   INSERT INTO `user_log` (`user_id`, `name`, `email`, `message_content`, `created_at`)
   VALUES (NEW.id, NEW.name, NEW.email, user_message, NOW());
END;
//
DELIMITER ;
```

INSERT INTO `users` (`name`, `email`, `password`, `user_type`) VALUES ('John Doe', 'johndoe@example.com', 'password123', 'user');

```
MariaDB [user_db]> SELECT * FROM `user_log`;
+----+---------+----------+--------------------+-----------------------------------------------------+---------------------+
| id | user_id | name     | email              | message_content                                     | created_at          |
+----+---------+----------+--------------------+-----------------------------------------------------+---------------------+
|  1 |       6 | John Doe | johndoe@example.com | New user added: John Doe (Email: johndoe@example.com) | 2024-05-16 00:01:11 |
+----+---------+----------+--------------------+-----------------------------------------------------+---------------------+
1 row in set (0.002 sec)
```

```
MariaDB [user_db]> select * from users;
+----+----------+----------------------+----------------------------------+-----------+
| id | name     | email                | password                         | user_type |
+----+----------+----------------------+----------------------------------+-----------+
|  1 | canvil   | canvil1212@gmail.com | 202cb962ac59075b964b07152d234b70 | user      |
|  2 | admin    | admin1212@gmail.com  | 202cb962ac59075b964b07152d234b70 | admin     |
|  3 | joyal    | joyal1212@gmail.com  | 202cb962ac59075b964b07152d234b70 | user      |
|  4 | lobo     | lobo1212@gmail.com   | 123                              | user      |
|  6 | John Doe | johndoe@example.com  | password123                      | user      |
|  7 | John Doe | johndoe@example.com  | password123                      | user      |
+----+----------+----------------------+----------------------------------+-----------+
```

6. CONCLUSION

In conclusion, the Alcohol Mart Database Management System (DBMS) represents a significant advancement in the management and optimization of retail operations within the alcoholic beverages .Through the implementation customer relationship management, and contact functionalities, the system offers a holistic solution to address the unique challenges faced by retail stores specializing in alcoholic beverages. Additionally, the DBMS enhances customer engagement and satisfaction through personalized marketing initiatives, and streamlined customer service processes. Overall, the this project not only improves operational efficiency but also fosters business growth and competitiveness within the dynamic and evolving alcoholic beverages retail landscape.

7. FUTURE WORK

1.Membership Card:- The future scope for implementing a membership card on the Alcohol Mart e-commerce website includes offering exclusive benefits such as discounts, early access to new products, and personalised recommendations.

2.Secure Payment Gateway:-The Alcohol Mart e-commerce website can enhance its secure payment gateway by integrating additional payment methods like cryptocurrency and buy-now-pay-later options. Implementing advanced encryption technologies and multi-factor authentication will further bolster transaction security.

3.Search Page:- For the Search Page of the Alcohol Mart e-commerce website, future enhancements could include implementing advanced search filters, improving search result relevance, and integrating voice search capabilities for enhanced user experience.

4.Rating System:- Future scope for the Alcohol Mart e-commerce website includes enhancing the Search Page with advanced filters and improving the Rating System to allow customers to provide detailed feedback on products, enhancing user experience and engagement.

8. REFERENCES

[**1**] Database Management Systems" by Raghu Ramakrishnan and Johannes Gehrke Published in 2000, Latest published in the year 2019.

[**2**] Database System Concepts" by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, Published in the year 1986,7th edition, published in 2020 is the latest edition.

[**3**] Introduction to Database Systems" by C.J. Date,Published in the Year 2000,8th edition, which was published in 2021.

[**4**] Fundamentals of Database Systems" by Ramez Elmasri and Shamkant B. Navathe, Published in the year 1989 ,7th edition, published in 2016 is the latest edition.