University of Information Technology and Sciences

Department of CSE

Lab Project

Course Code	CSE0612216S
Course Name	Database Management System Lab
Project Title	Crypto Wallet Management System

Submitted by:

Name: Sabbir Ahmed

ID No: 0432410005101106

Batch: 55th

Section: 3C1

Project Name: Crypto Wallet Management System

1. Objectives

The main objective of this project is to design and implement a secure, database-driven system that allows users to manage cryptocurrency wallets. The system aims to:

- Allow users to create accounts and securely log in.
- Enable wallet creation for multiple cryptocurrencies.
- Perform transactions between user wallets.
- Track balances, transaction history, and audit logs.
- Provide administrative control for user, wallet, and transaction management.
- Ensure transparency and traceability with audit logs.

2. Introduction

As cryptocurrencies become more widely adopted, users need efficient systems to manage digital wallets and transactions securely. This project introduces a simplified **Crypto Wallet Management System** using **MySQL** as the database engine, with core features like:

- User authentication
- Wallet handling for multiple cryptocurrencies
- Transaction management
- Role-based access (user/admin)
- Action tracking through audit logs

This backend system simulates the structure of a real-world wallet app, laying the groundwork for future integration with frontend technologies and APIs.

Main Modules & Tables:

1. **Users**
   - `user_id`, `name`, `email`, `password`, `role` (user/admin)
2. **Wallets**
   - `wallet_id`, `user_id`, `currency_id`, `address`, `balance`
3. **Currencies**
   - `currency_id`, `name`, `symbol`
4. **Transactions**
   - `tx_id`, `sender_wallet_id`, `receiver_wallet_id`, `currency_id`, `amount`, `timestamp`, `status`
5. **AuditLogs**
   - `log_id`, `user_id`, `action`, `timestamp`, `details`
6. **Notifications**
   - `notification_id`, `user_id`, `message`, `is_read`, `timestamp`
7. **TransactionFees**
   - `fee_id`, `currency_id`, `fee_percent`
8. **AddressGenerationLogs**
   - `log_id`, `user_id`, `currency_id`, `generated_address`, `timestamp`

Each table has proper relationships using **foreign keys**, and supports essential operations like `INSERT`, `SELECT`, `UPDATE`, and `JOIN`.

3. Experimental Setup

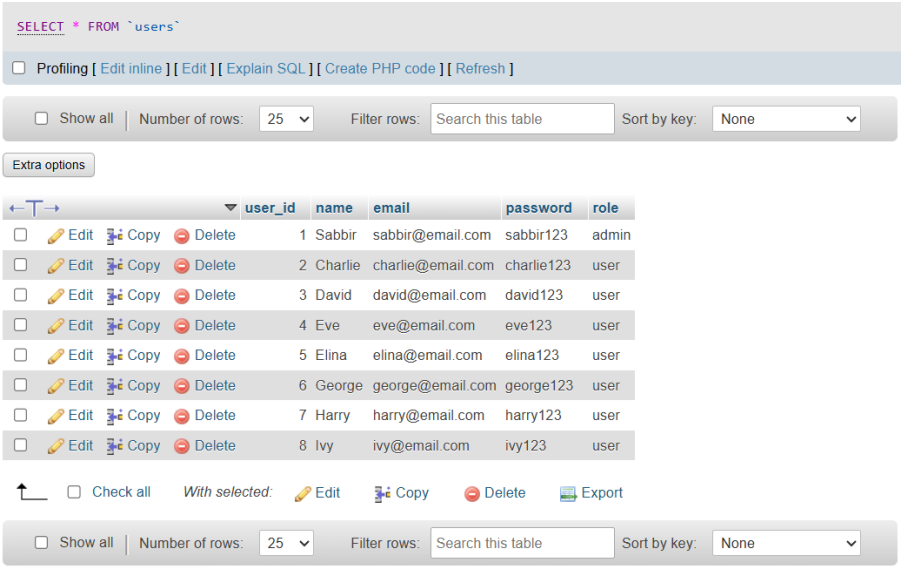
The following tools and technologies were used in the development and testing of this project:

- **XAMPP** – Local server environment with MySQL and phpMyAdmin.
- **MySQL** – For creating and managing the relational database.
- **phpMyAdmin** – For database GUI interaction.
- **SQL** – For table creation, data insertion, and query execution.

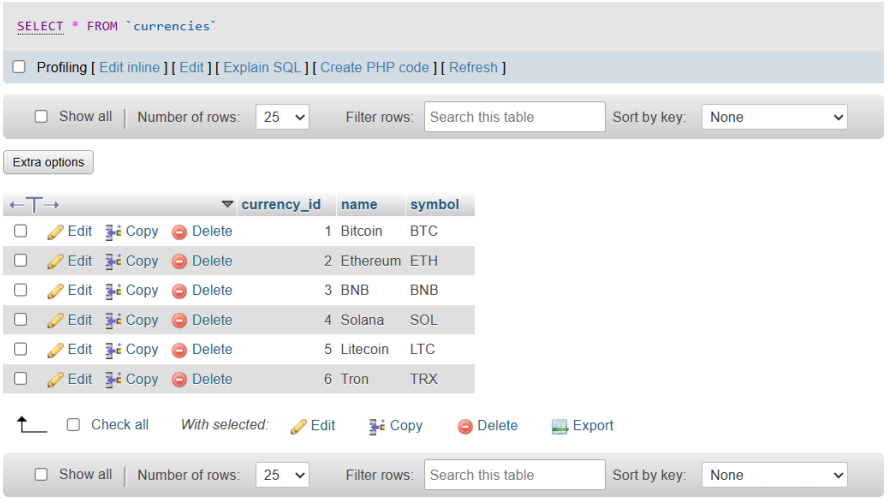
The project was implemented and tested in **phpMyAdmin** provided by **XAMPP** running locally on a Windows environment.

4. Create the following tables:

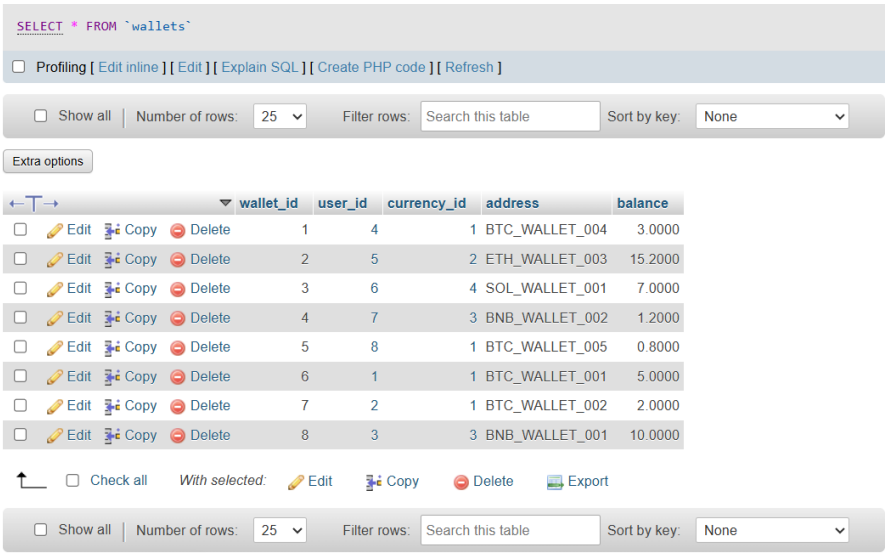
Users Table: **Stores user details such as name, email, password, and role (user/admin).**



Currencies Table: Contains the list of supported cryptocurrencies



Wallets Table: **Tracks each user's crypto wallets, including address and balance for each currency.**



```
SELECT * FROM `wallets`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

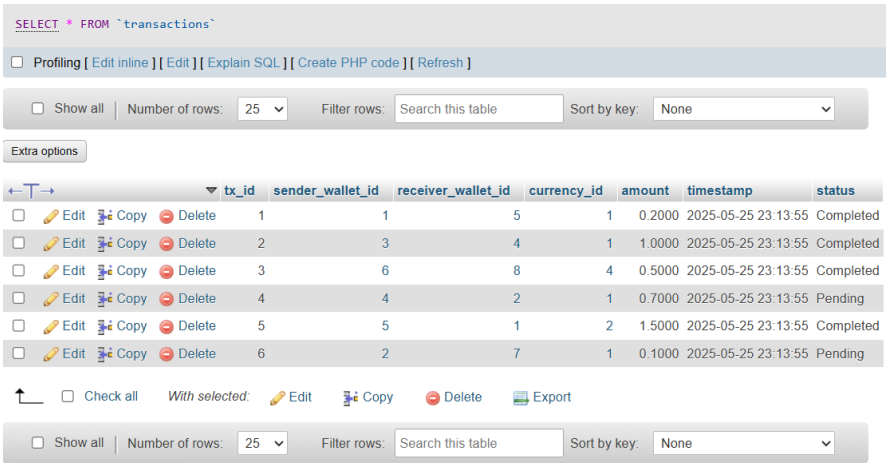
Extra options

	wallet_id	user_id	currency_id	address	balance
Edit Copy Delete	1	4	1	BTC_WALLET_004	3.0000
Edit Copy Delete	2	5	2	ETH_WALLET_003	15.2000
Edit Copy Delete	3	6	4	SOL_WALLET_001	7.0000
Edit Copy Delete	4	7	3	BNB_WALLET_002	1.2000
Edit Copy Delete	5	8	1	BTC_WALLET_005	0.8000
Edit Copy Delete	6	1	1	BTC_WALLET_001	5.0000
Edit Copy Delete	7	2	1	BTC_WALLET_002	2.0000
Edit Copy Delete	8	3	3	BNB_WALLET_001	10.0000

Check all With selected: Edit Copy Delete Export

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Transactions Table: Logs all transactions between wallets, including status, amount, and timestamp.



```
SELECT * FROM `transactions`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

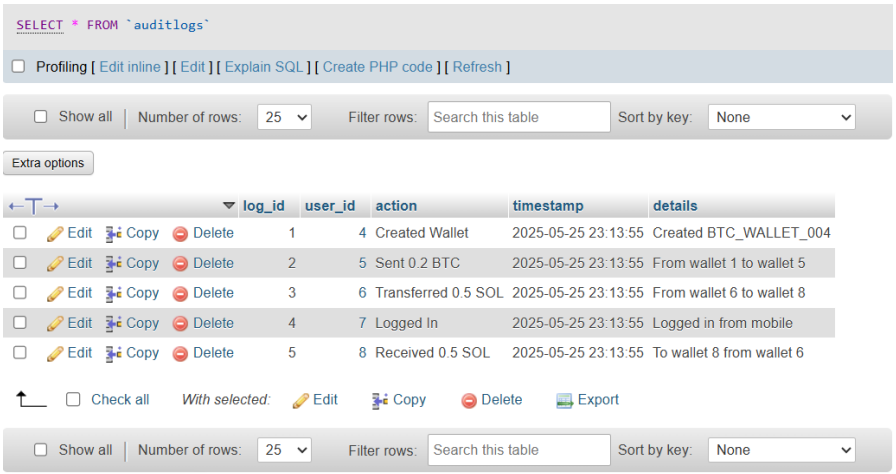
Extra options

	tx_id	sender_wallet_id	receiver_wallet_id	currency_id	amount	timestamp	status
Edit Copy Delete	1	1	5	1	0.2000	2025-05-25 23:13:55	Completed
Edit Copy Delete	2	3	4	1	1.0000	2025-05-25 23:13:55	Completed
Edit Copy Delete	3	6	8	4	0.5000	2025-05-25 23:13:55	Completed
Edit Copy Delete	4	4	2	1	0.7000	2025-05-25 23:13:55	Pending
Edit Copy Delete	5	5	1	2	1.5000	2025-05-25 23:13:55	Completed
Edit Copy Delete	6	2	7	1	0.1000	2025-05-25 23:13:55	Pending

Check all With selected: Edit Copy Delete Export

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

AuditLogs Table: Records all significant user actions like logins, transfers, or wallet creation for audit purposes.



```
SELECT * FROM `auditlogs`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

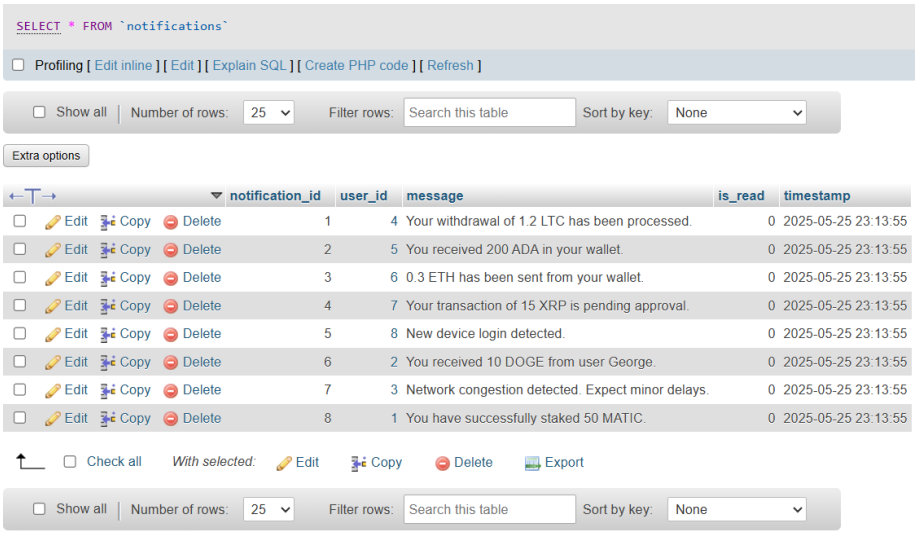
Extra options

	log_id	user_id	action	timestamp	details
Edit Copy Delete	1	4	Created Wallet	2025-05-25 23:13:55	Created BTC_WALLET_004
Edit Copy Delete	2	5	Sent 0.2 BTC	2025-05-25 23:13:55	From wallet 1 to wallet 5
Edit Copy Delete	3	6	Transferred 0.5 SOL	2025-05-25 23:13:55	From wallet 6 to wallet 8
Edit Copy Delete	4	7	Logged In	2025-05-25 23:13:55	Logged in from mobile
Edit Copy Delete	5	8	Received 0.5 SOL	2025-05-25 23:13:55	To wallet 8 from wallet 6

Check all With selected: Edit Copy Delete Export

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Notifications Table: Stores alerts or system messages sent to users, including read/unread status.



```
SELECT * FROM `notifications`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	notification_id	user_id	message	is_read	timestamp
Edit Copy Delete	1	4	Your withdrawal of 1.2 LTC has been processed.	0	2025-05-25 23:13:55
Edit Copy Delete	2	5	You received 200 ADA in your wallet.	0	2025-05-25 23:13:55
Edit Copy Delete	3	6	0.3 ETH has been sent from your wallet.	0	2025-05-25 23:13:55
Edit Copy Delete	4	7	Your transaction of 15 XRP is pending approval.	0	2025-05-25 23:13:55
Edit Copy Delete	5	8	New device login detected.	0	2025-05-25 23:13:55
Edit Copy Delete	6	2	You received 10 DOGE from user George.	0	2025-05-25 23:13:55
Edit Copy Delete	7	3	Network congestion detected. Expect minor delays.	0	2025-05-25 23:13:55
Edit Copy Delete	8	1	You have successfully staked 50 MATIC.	0	2025-05-25 23:13:55

Check all *With selected:* Edit Copy Delete Export

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

TransactionFees Table: Holds fee percentages applicable per currency for transaction processing.

```
SELECT * FROM `transactionfees`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

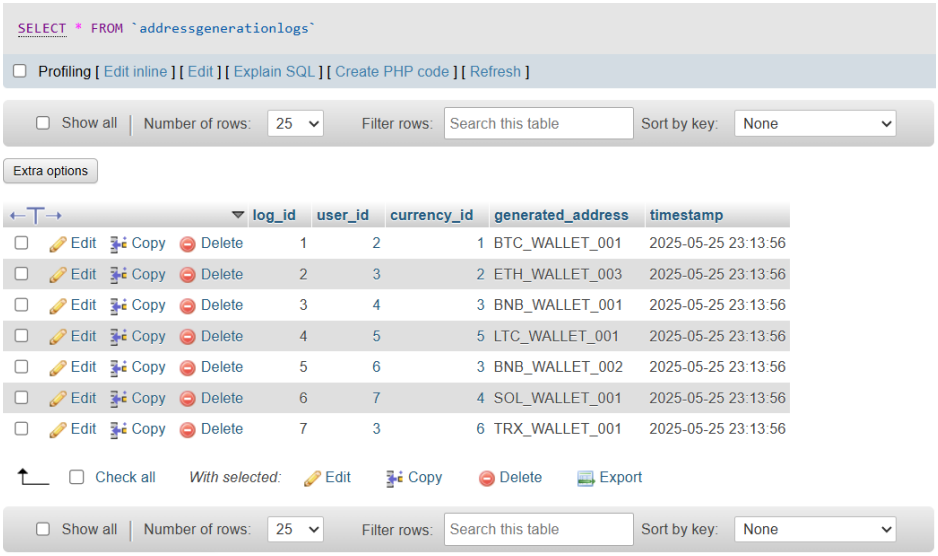
Extra options

	fee_id	currency_id	fee_percent
Edit Copy Delete	1	1	1.50
Edit Copy Delete	2	2	2.00
Edit Copy Delete	3	3	0.75
Edit Copy Delete	4	4	1.20
Edit Copy Delete	5	5	0.60
Edit Copy Delete	6	6	0.70

Check all *With selected:* Edit Copy Delete Export

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

AddressGenerationLogs Table: Tracks address generation events for users and specific currencies.



```
SELECT * FROM `addressgenerationlogs`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	log_id	user_id	currency_id	generated_address	timestamp
Edit Copy Delete	1	2	1	BTC_WALLET_001	2025-05-25 23:13:56
Edit Copy Delete	2	3	2	ETH_WALLET_003	2025-05-25 23:13:56
Edit Copy Delete	3	4	3	BNB_WALLET_001	2025-05-25 23:13:56
Edit Copy Delete	4	5	5	LTC_WALLET_001	2025-05-25 23:13:56
Edit Copy Delete	5	6	3	BNB_WALLET_002	2025-05-25 23:13:56
Edit Copy Delete	6	7	4	SOL_WALLET_001	2025-05-25 23:13:56
Edit Copy Delete	7	3	6	TRX_WALLET_001	2025-05-25 23:13:56

Check all *With selected:* Edit Copy Delete Export

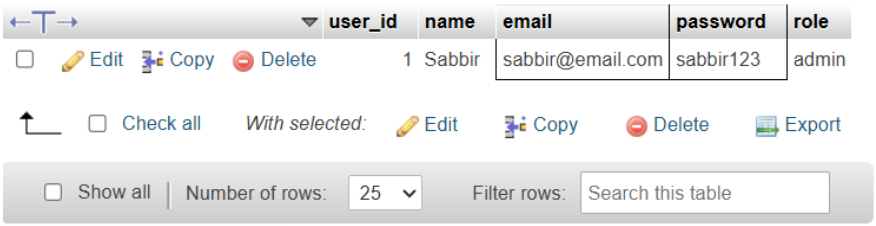
Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

Essential queries & Outputs:

1. Login (check credentials)

SELECT * FROM Users

WHERE email = 'sabbir@email.com' AND password = 'sabbir123';



2. Create a transaction (Pending)

INSERT INTO Transactions (sender_wallet_id, receiver_wallet_id, currency_id, amount, status)

VALUES (1, 2, 1, 0.25, 'Pending');

tx_id	sender_wallet_id	receiver_wallet_id	currency_id	amount	timestamp	status
1	1	5	1	0.2000	2025-05-25 23:13:55	Completed
2	3	4	1	1.0000	2025-05-25 23:13:55	Completed
3	6	8	4	0.5000	2025-05-25 23:13:55	Completed
4	4	2	1	0.7000	2025-05-25 23:13:55	Pending
5	5	1	2	1.5000	2025-05-25 23:13:55	Completed
6	2	7	1	0.1000	2025-05-25 23:13:55	Pending
7	1	2	1	0.2500	2025-05-26 01:11:36	Pending

3. Mark transaction as completed

UPDATE Transactions SET status = 'Completed' WHERE tx_id = 1;

tx_id	sender_wallet_id	receiver_wallet_id	currency_id	amount	timestamp	status
1	1	5	1	0.2000	2025-05-25 23:13:55	Completed
2	3	4	1	1.0000	2025-05-25 23:13:55	Completed
3	6	8	4	0.5000	2025-05-25 23:13:55	Completed
4	4	2	1	0.7000	2025-05-25 23:13:55	Pending
5	5	1	2	1.5000	2025-05-25 23:13:55	Completed
6	2	7	1	0.1000	2025-05-25 23:13:55	Completed
7	1	2	1	0.2500	2025-05-26 01:11:36	Pending

4. View all transactions for a wallet

SELECT * FROM Transactions WHERE sender_wallet_id = 1 OR receiver_wallet_id = 1 ORDER BY timestamp DESC;

tx_id	sender_wallet_id	receiver_wallet_id	currency_id	amount	timestamp ▼ 1	status
7	1	2	1	0.2500	2025-05-26 01:11:36	Pending
1	1	5	1	0.2000	2025-05-25 23:13:55	Completed
5	5	1	2	1.5000	2025-05-25 23:13:55	Completed

4. View unread notifications for a user

SELECT * FROM Notifications WHERE user_id = 4 AND is_read = FALSE;

notification_id	user_id	message	is_read	timestamp
1	4	Your withdrawal of 1.2 LTC has been processed.	0	2025-05-25 23:13:55

5. Mark notification as read

UPDATE Notifications SET is_read = TRUE WHERE notification_id = 1;

notification_id	user_id	message	is_read	timestamp
1	4	Your withdrawal of 1.2 LTC has been processed.	1	2025-05-25 23:13:55

6. Calculate fee for 0.5 BTC transfer (example)

SELECT 0.5 * (fee_percent / 100) AS fee

FROM TransactionFees WHERE currency_id = 1;

fee
0.0075000

7. View all address generation logs of a user

SELECT * FROM AddressGenerationLogs WHERE user_id = 3;

log_id	user_id	currency_id	generated_address	timestamp
2	3	2	ETH_WALLET_003	2025-05-25 23:13:56
7	3	6	TRX_WALLET_001	2025-05-25 23:13:56

8. Find all wallets belonging to the user named 'Charlie', showing the currency name and balance.

SELECT u.name AS user_name, c.name AS currency_name, w.address, w.balance

FROM Wallets w JOIN Users u ON w.user_id = u.user_id

JOIN Currencies c ON w.currency_id = c.currency_id WHERE u.name = 'Charlie';

user_name	currency_name	address	balance
Charlie	Bitcoin	BTC_WALLET_002	2.0000

9. What is the total balance of Bitcoin (BTC) across all wallets?

SELECT c.name AS currency_name, SUM(w.balance) AS total_btc_balance

FROM Wallets w JOIN Currencies c ON w.currency_id = c.currency_id

WHERE c.symbol = 'BTC';

currency_name	total_btc_balance
Bitcoin	10.8000

10. For each currency, what is its transaction fee percentage?

SELECT c.name AS currency_name, c.symbol, tf.fee_percent

FROM Currencies c LEFT JOIN TransactionFees tf ON c.currency_id = tf.currency_id;

currency_name	symbol	fee_percent
Bitcoin	BTC	1.50
Ethereum	ETH	2.00
BNB	BNB	0.75
Solana	SOL	1.20
Litecoin	LTC	0.60
Tron	TRX	0.70

11. **Show users and their unread notifications, ordered by the most recent notification first.**

SELECT u.name, n.message, n.timestamp FROM Notifications n JOIN Users u ON n.user_id = u.user_id

WHERE n.is_read = FALSE ORDER BY n.timestamp DESC;

name	message	timestamp ▾ 1
Elina	You received 200 ADA in your wallet.	2025-05-25 23:13:55
George	0.3 ETH has been sent from your wallet.	2025-05-25 23:13:55
Harry	Your transaction of 15 XRP is pending approval.	2025-05-25 23:13:55
Ivy	New device login detected.	2025-05-25 23:13:55
Charlie	You received 10 DOGE from user George.	2025-05-25 23:13:55
David	Network congestion detected. Expect minor delays.	2025-05-25 23:13:55
Sabbir	You have successfully staked 50 MATIC.	2025-05-25 23:13:55

12. **What is the average balance of wallets for each currency?**

SELECT c.name AS currency_name, c.symbol, AVG(w.balance) AS average_balance

FROM Wallets w JOIN Currencies c ON w.currency_id = c.currency_id

GROUP BY c.currency_id, c.name, c.symbol ORDER BY average_balance DESC;

currency_name	symbol	average_balance ▾ 1
Ethereum	ETH	15.20000000
Solana	SOL	7.00000000
BNB	BNB	5.60000000
Bitcoin	BTC	2.70000000

13. **List all currencies and the number of wallets created for each currency.**

SELECT c.name AS currency_name, c.symbol, COUNT(w.wallet_id) AS number_of_wallets

FROM Currencies c LEFT JOIN Wallets w ON c.currency_id = w.currency_id

GROUP BY c.currency_id, c.name, c.symbol ORDER BY number_of_wallets DESC;

currency_name	symbol	number_of_wallets ▾ 1
Bitcoin	BTC	4
BNB	BNB	2
Ethereum	ETH	1
Solana	SOL	1
Litecoin	LTC	0
Tron	TRX	0

5. Discussion

During the project, I faced several challenges, such as:

- **Database design**: Ensuring proper normalization and relationships between tables like `Users`, `Wallets`, `Transactions`, and `AuditLogs`.
- **Address uniqueness**: Guaranteeing unique wallet addresses in `Wallets` table.
- **Transaction logic**: Designing logic to reflect balance deduction/addition correctly with transaction status handling.
- **Audit tracking**: Creating meaningful and useful logs without redundancy.
- **Data simulation**: Inserting realistic and meaningful test data for better output validation.

I overcame these challenges through multiple testing iterations, reading MySQL documentation, and referring to real-world wallet system structures. Gradually, the system became more functional and realistic.

6. Conclusion

This project successfully simulates the backend of a **Crypto Wallet Management System**, providing all essential database operations including:

- Multi-currency wallet handling
- User and role management
- Transaction processing with fee logic
- Real-time balance tracking
- Full auditing for actions