**Project Report: Chain Flow – A Cryptocurrency Transaction Tracker**

**Team:**

* Muhammad Jibran: 01-131232-062
* Abdul Rafay: 01-131232-004

**1. Project Title:** Chain Flow: A Blockchain-based Cryptocurrency Transaction Tracker

**2. Project Objective:** The primary objective of this project was to design and implement a simulated blockchain system in C++ that tracks cryptocurrency transactions between wallets, ensuring data integrity and security. The system features a blockchain implementation, transaction validation, and wallet management, simulating the core functionalities of real-world cryptocurrency systems like Bitcoin and Ethereum.

**3. Key Features:**

**1. Blockchain Implementation:**

* Each block stores transaction details, timestamps, sender and receiver wallet addresses, and links to the previous block using SHA-256 hashes.

**2. Transaction Management:**

* Facilitates cryptocurrency transfers between wallets.
* Validates transactions based on wallet balances.

**3. Wallet Management:**

* Creates wallets with unique IDs and initial balances.
* Displays wallet balances and transaction histories.

**4. Blockchain Security:**

* Utilizes hashing to secure blocks and prevent tampering.
* Validates blockchain integrity by verifying hash links between blocks.

**5. Dynamic Updates:**

* Allows the dynamic addition or removal of wallets.
* Supports the addition of new transactions or blocks to the blockchain.

**6. Coin Simulation:**

* Demonstrates fluctuations in cryptocurrency values to simulate real-world market conditions.

**4. Technologies Used:**

**Programming Language:**

* C++ with Visual Studio IDE

**Frontend:**

* WinForms (C++/CLI) for user interface

**Database:**

* Microsoft SQL Server for data storage and retrieval

**Data Structures:**

* Linked Lists: To implement the blockchain.
* Hash Maps: To store wallet details (balances and IDs).

**Security Library:**

* SHA-256 for secure block hashing.

**5. Key Functions and Features:**

**1. Wallet Management:**

* Users can register with a username, password, and receive a unique wallet address.
* Users can view their wallet balances and transaction histories through the GUI.

**2. Transaction Management:**

* Transactions are validated before execution to ensure sufficient balances.
* Successful transactions update both the sender’s and receiver’s wallet balances.

**3. Blockchain Integrity:**

* Each block is secured using a SHA-256 hash.
* Blockchain validation ensures no tampering has occurred by recalculating hashes and verifying links.

**4. Coin Simulation:**

* Demonstrates market value fluctuations for different coins.
* Users can buy or sell coins at current market rates, with transactions recorded on the blockchain.

**6. Challenges and Solutions:**

**Challenge 1:** Ensuring secure and efficient blockchain validation.

* **Solution:** Used SHA-256 for hashing and implemented a method to recursively validate hash links.

**Challenge 2:** Managing large-scale transactions and wallet data efficiently.

* **Solution:** Integrated a SQL database for persistent storage and retrieval of user, coin, and blockchain data.

**Challenge 3:** Creating a user-friendly interface.

* **Solution:** Developed a WinForms GUI in C++/CLI to simplify interaction.

**7. Results and Outcomes:**

**Achieved:**

* A fully functional blockchain simulation in C++.
* Secure management of wallets and transactions.
* Successful integration of SQL Server for data persistence.
* Dynamic coin market simulation demonstrating real-world cryptocurrency principles.

**8. Future Scope:**

* Implement mining with proof-of-work for block creation.
* Simulate a distributed network with multiple nodes.
* Enhance the visualization of blockchain data with real-time updates.
* Expand to include consensus algorithms like Proof-of-Stake.

**9. Conclusion:**

The Chain Flow project successfully demonstrates the core functionalities of a blockchain system, including transaction tracking, wallet management, and data security. By integrating modern technologies like MSSQL Server and SHA-256, the project provides insights into blockchain mechanics, laying the foundation for future advancements.

**10. References:**

* **Bitcoin** Whitepaper by **Satoshi Nakamoto**.
* Documentation for **SHA-256** hashing.
* **SQL** Server documentation for database management.