Project Proposal: 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 goal of this project is to develop a simulated blockchain system in C++ that tracks cryptocurrency transactions between wallets, ensuring integrity and security. The system will feature a blockchain implementation, transaction validation, and wallet management. It will mimic the core functionalities of cryptocurrency systems like Bitcoin or Ethereum.

---

3. Key Features

1. **Blockchain Implementation**:
    - Each block stores transaction details, a timestamp, wallet address of sender and receiver, and links to the previous block using hashes.
2. **Transaction Management**:
    - Record cryptocurrency transfers between wallets.
    - Validate transactions based on wallet balances.
3. **Wallet Management**:
    - Create wallets with unique IDs and initial balances.
    - Display wallet balances and transaction histories.
4. **Blockchain Security**:
    - Use hashing to secure blocks and prevent tampering.
    - Validate blockchain integrity by checking the hash links.

5. **Dynamic Updates**:
    - Add or remove wallets dynamically.
    - Add new transactions or blocks to the chain.

---

4. Proposed Technologies

- **Programming Language**: C++
- **Data Structures**:
    - **Linked Lists**: To implement the blockchain.
    - **Hash Maps**: To store wallet details (balances and IDs).

---

5. Implementation Plan

Phase 1: System Design

- Define the structure for wallets, transactions, and blocks.
- Design the blockchain data structure.

Phase 2: Core Features

- Implement wallet creation and management.
- Develop the blockchain and integrate hashing for blocks.
- Enable transaction recording and validation.

Phase 3: Blockchain Security

- Add functions to validate blockchain integrity.
- Handle edge cases like tampered or invalid blocks.
- Data encryption.

Phase 4: Testing and Debugging

- Test with multiple wallets and transactions.
- Validate edge cases like insufficient funds or tampered data.

---

6. Timeline

Phase	Estimated Time
System Design	1 Week
Core Features	2 Weeks
Blockchain Security	1 Week
Testing and Debugging	1 Week

---

7. Expected Outcomes

- A fully functional blockchain simulation.
- A secure system for managing wallets and transactions.
- Insights into blockchain mechanics, including hashing and data integrity.

---

8. Challenges

- Ensuring proper handling of blockchain security (e.g., hash verification).
- Efficient management of large-scale transactions and wallets.
- Data encryption for secure the assets.

---

9. Future Scope

- Adding **mining** with proof-of-work for block creation.
- Simulating a distributed network with multiple users.
- Enhancing visualization with real-time updates.
- Expanding the project to include consensus algorithms like Proof-of-Stake.
