Q – Hash, A step towards Post Quantum Banking

1) Introduction:

1.1. Purpose:

This document captures the system's intended functionality, interfaces, behaviour, and constraints to offer a clear guideline for developers, stakeholders, and other participants involved in its design, development, and deployment.

1.2. Scope

As quantum computing becomes more powerful, traditional cryptographic algorithms are at risk of becoming obsolete, threatening the security of current blockchain technologies.

1) Quantum-Proof Blockchain: Creating a blockchain resistant to quantum decryption
2) Falcon Algorithm: Implementing Falcon for secure signature processes.
3) Quantum Algorithm: Implementing Modern Quantum
4) The dual key feature enables us to have transactions where people have responsibility but not the responsibility

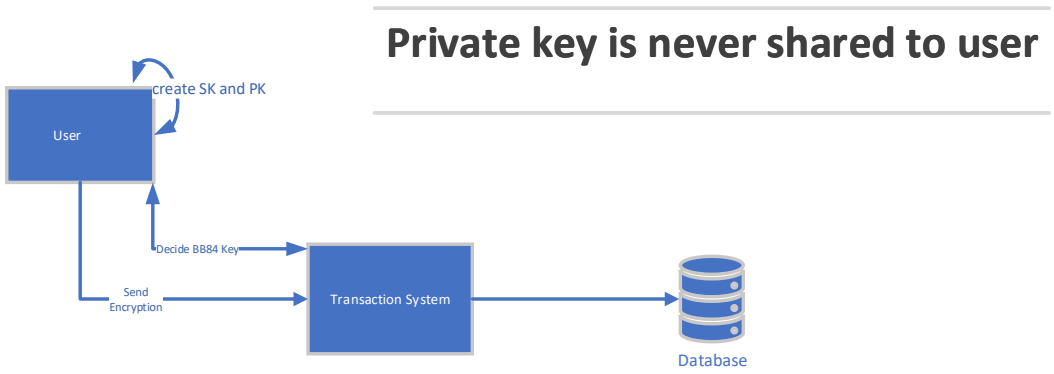
1.3. References:

- Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU https://falcon-sign.info/falcon.pdf
- Computer Security Resource Centre (NIST) : https://csrc.nist.gov/projects/post-quantum-cryptography

2) Overall Description:

The Q-Hash is a post-quantum designed as an independent, decentralized platform that provides a transparent, secure, and verifiable mechanism for transactions, especially those prone to disputes or requiring third-party supervision. By integrating with modern technologies like Quantum Algorithm for BB8 4 key distribution and blockchain.

2.1. System Context Diagram



2.2. Product Functions:

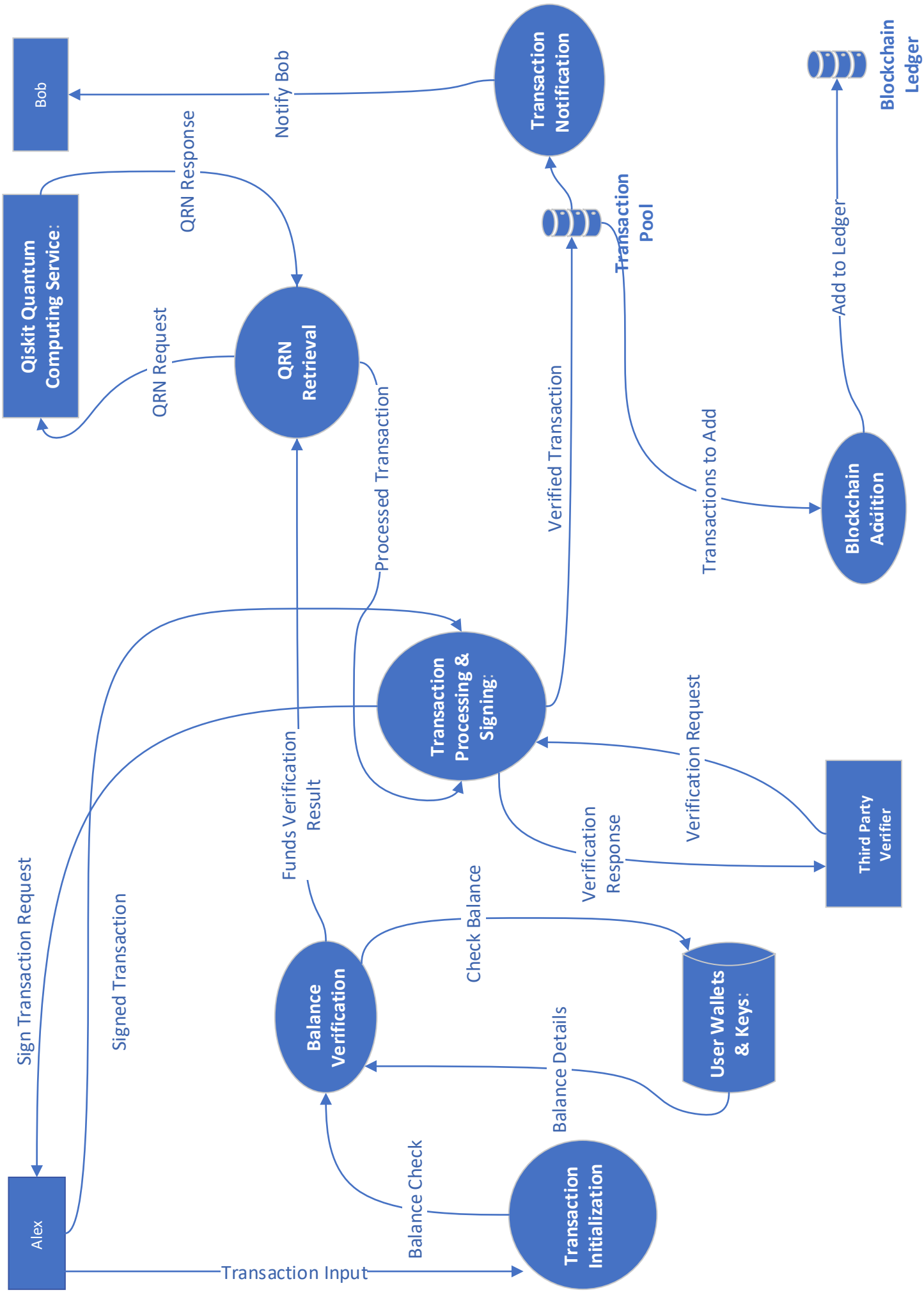
1) Account Creation: Creation of Public Key and Private Key

2) Secure Transmission of the Public Key via BB84 Protocol

3) Transaction Verification

4) Addition to blockchain

2.3. Data Flow Diagram



2.3. User Classes and Characteristics:

1. Regular Users: Individuals or entities initiating or receiving transactions. They require a wallet and will interact primarily with transaction initiation and verification functions.
2. Validators/Miners: Participants responsible for verifying and adding transactions to the blockchain. They ensure the system's integrity and are incentivized by rewards.
3. Third-party Supervisors: External entities or individuals, like legal representatives or mediators, need to verify a transaction's authenticity and details. They don't participate in the transaction but oversee its legitimacy.

2.4. Operating Environment:

1)Web Interface: Nodes can run lightweight web interfaces locally, developed using HTML, CSS, and JavaScript. This allows users to interact with the system directly from their devices without relying on external centralized servers.

2) Flask-powered backend,

3) Platform Compatibility: Designed to be platform-agnostic, nodes can operate on major operating systems like Windows, macOS, Linux, iOS, and Android

2.5. Design and Implementation Constraints:

1. Quantum Computing Availability: Access to quantum computers for true random number generation may be limited or costly.
2. Performance: Lattice-based cryptography can be computationally intensive, potentially slowing down transaction verification times.
3. Integration: Integrating the system with existing legal and financial systems might require additional interfaces or compliance checks.

2.6. Assumptions and Dependencies:

1. Network Participation: The system's security and functionality depend on active participation by validators/miners
2. Legal Acceptance

3) Specific Requirements 3.1. External Interfaces:

1. Web Interface: A user-friendly interface developed using HTML, CSS, and JavaScript, allowing users to initiate transactions, manage their wallets, and interact with the network.
2. Quantum RNG API with Qiskit: A RESTful interface that communicates with Qiskit services. This API facilitates the generation of truly random numbers by initiating quantum computations within Qiskit and retrieving the results.

3.1. Software System Attributes:

1. Reliability: The system should have high uptime, with redundancy measures in place to counteract potential node failures.
2. Availability: Given its decentralized nature, the system should always be available for transactions and verifications, barring individual node downtimes.

3.2. Other Requirements:

1. Regulatory Compliance: If the cryptocurrency system is to be used for legal settlements or in specific jurisdictions, it may need to comply with financial and data protection regulations.
2. User Education: Due to the novel integration of quantum RNG and specific cryptographic methods, resources or tutorials might be required to educate users about the system's operations and benefits.
3. Backup and Recovery: Mechanisms should be in place for users to back up their wallets or recover lost keys, ensuring they don't lose access to their assets.