**Theoretical Part:**

**Blockchain Basics**

**Define blockchain in your own words (100–150 words).**

Blockchain is a decentralized digital ledger that records data in a secure and tamper-proof way across a network of computers. Instead of being stored in a single location, the data is distributed across many nodes, making it highly secure and transparent. Each record, known as a "block," contains a list of transactions and is linked to the previous block through cryptographic hashes, forming a chain—hence the name "blockchain." Once data is recorded on a blockchain, it is nearly impossible to alter without changing all subsequent blocks, which ensures integrity and trust. It operates without a central authority and uses consensus mechanisms like Proof of Work or Proof of Stake to validate new entries. This makes blockchain ideal for applications that require transparency, security, and decentralization.

### **List 2 real-life use cases:**

1. **Supply Chain Management:** Blockchain can track the origin and journey of products (like food or medicine) from manufacturer to consumer, improving transparency and reducing fraud. Like Walmart uses to track its packages
2. **Digital Identity:** Blockchain can provide secure, tamper-proof digital IDs that individuals control, helping to prevent identity theft and giving people better access to services.

**Block Anatomy**

Draw a block showing: data, previous hash, timestamp, nonce, and Merkle root.  
  
A screenshot of a computer

Description automatically generated

**Briefly explain with an example how the Merkle root helps verify data integrity.**

A **Merkle root** is a cryptographic hash that represents all transactions in a block. It's created by repeatedly hashing pairs of transaction hashes (called leaves) until one final hash remains at the top of the tree—the Merkle root.

#### ****Example:****

Let’s say a block contains 4 transactions:

* Tx1: A → B
* Tx2: C → D
* Tx3: E → F
* Tx4: G → H

1. Each transaction is hashed:
   * H1 = Hash(Tx1)
   * H2 = Hash(Tx2)
   * H3 = Hash(Tx3)
   * H4 = Hash(Tx4)
2. Hashes are paired and hashed again:
   * H12 = Hash(H1 + H2)
   * H34 = Hash(H3 + H4)
3. Finally:
   * Merkle Root = Hash(H12 + H34)

If any single transaction (like Tx3) is tampered with, its hash (H3) changes, which causes H34 and then the Merkle root to change. This allows quick detection of tampering without checking every transaction.

**Diagram of it can be:-**

A diagram of hashtags and hashtags

Description automatically generated

**Consensus Conceptualization**

**What is Proof of Work and why does it require energy?**

Proof of Work (PoW) is a consensus mechanism where miners solve complex mathematical puzzles to validate transactions and add new blocks to the blockchain. The first miner to solve the puzzle gets to add the block and is rewarded with cryptocurrency. This process requires high computational power, which in turn consumes a large amount of electricity. The energy use comes from running powerful hardware continuously to perform the necessary calculations, ensuring network security and preventing double spending.

**What is Proof of Stake and how does it differ?**

Proof of Stake (PoS) is a consensus method where validators are chosen to create new blocks based on the number of coins they "stake" or lock up as collateral. Unlike PoW, PoS does not rely on solving computational puzzles, making it far more energy-efficient. Validators are rewarded based on their stake and the time they've held it. This method reduces energy use and promotes long-term participation over brute computational power.

**What is Delegated Proof of Stake and how are validators selected?**

Delegated Proof of Stake (DPoS) is an advanced version of PoS where token holders vote to elect a small number of trusted delegates (also called witnesses or validators) to produce blocks on their behalf. The voting power is proportional to the amount of tokens held. Validators are selected through a continuous voting process, and those with the most votes become responsible for validating transactions and securing the network. This approach increases efficiency and scalability but slightly sacrifices decentralization.