UNIT 5

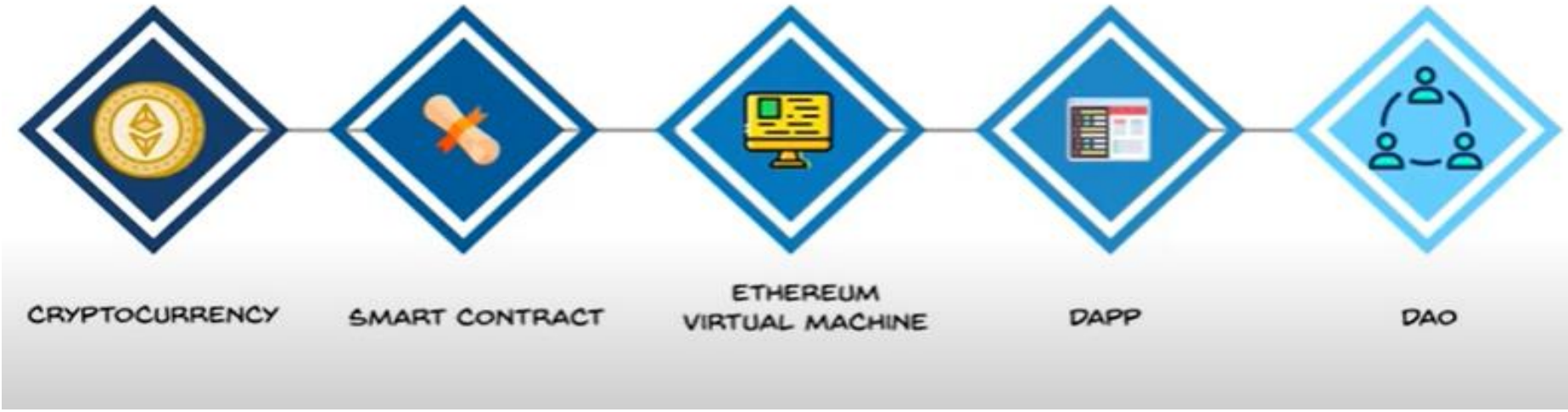
- **Ethereum Introduction**

- Ethereum is an open software platform based on blockchain technology that enables developers to build and deploy decentralized applications.
- Ethereum is distributed public blockchain network.
- The Ethereum blockchain focuses on running the programming code of any decentralized application.
- Ether is cryptocurrency whose blockchain is generated by the Ethereum platform.

- The record of all transactions and the EVM's present state gets stored on the blockchain, which in turn is stored and agreed upon by all nodes.
- Cryptographic mechanisms ensure that once transactions are verified as valid and added to the blockchain, they can't be tampered with later.
- The same mechanisms also ensure that all transactions are signed and executed with appropriate "permissions" (no one should be able to send digital assets from Alice's account, except for Alice herself).

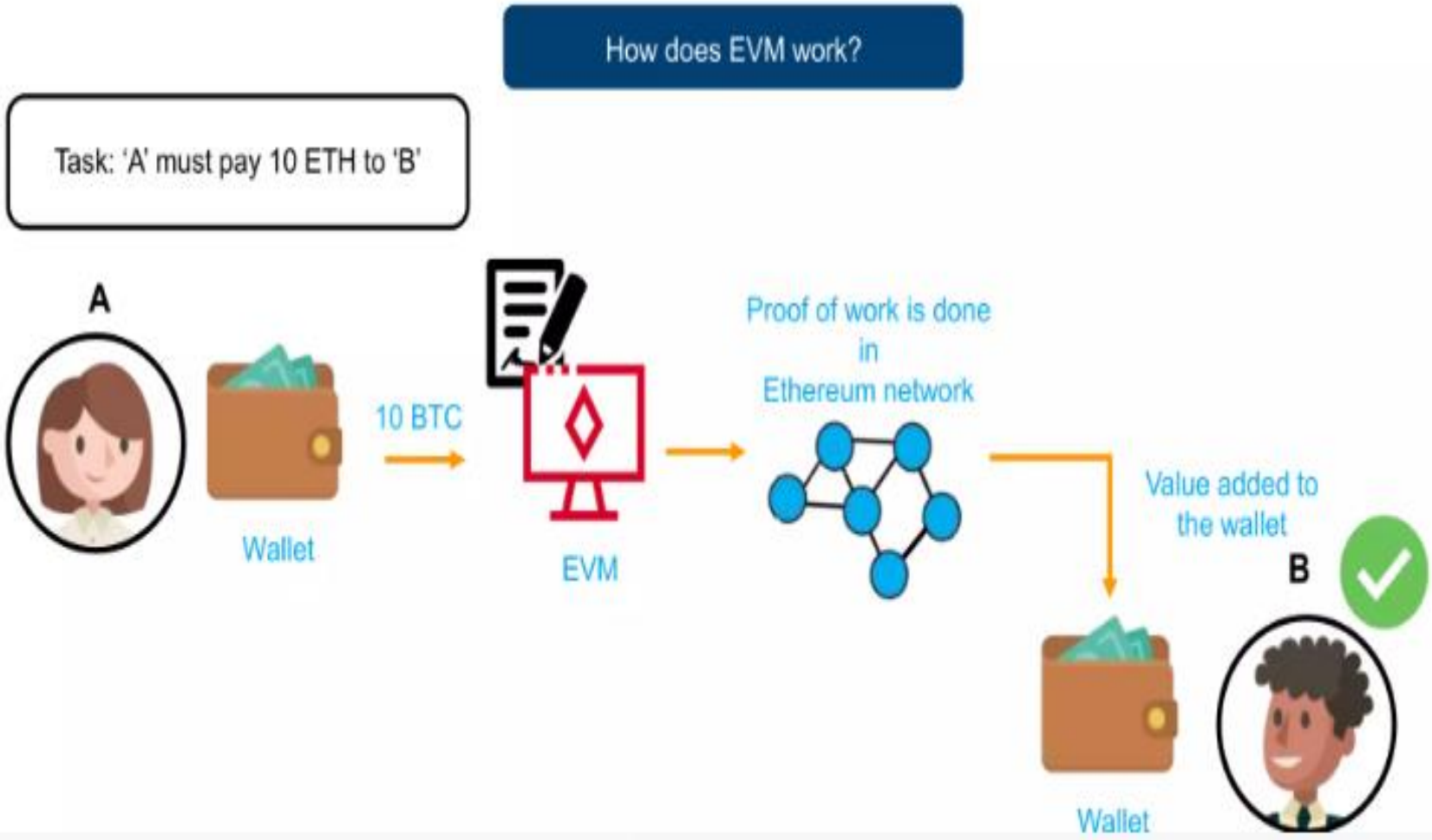
- Ethereum was proposed in 2013 by vitalik buterin a crypto currency researcher and programmer.
- In the Ethereum universe, there is a single, canonical computer (called the Ethereum Virtual Machine, or EVM) whose state everyone on the Ethereum network agrees on.

Features of Ethereum



- **Ethereum Features**
- **Ether**: This is Ethereum's cryptocurrecy.
- **Smart contracts**: Ethereum allows the development and deployment of these types of contracts.
- **Ethereum Virtual Machine:** Ethereum provides the underlying technology—the architecture and the software—that understands smart contracts and allows you to interact with it.
- **Decentralized applications (Dapps):** A decentralized application is called a Dapp (also spelled DAPP, App, or DApp) for short. Ethereum allows you to create consolidated applications, called decentralized applications.
- **Decentralized autonomous organizations (DAOs):** Ethereum allows you to create these for democratic decision-making.

Ethereum Virtual Machine



Note: All the nodes on Ethereum network execute smart

Ethereum Virtual Machine - Gas

Why do we need gas?

To run a car, we need fuel

In the same way, to run an application on Ethereum network, we need gas

- Ether

- Ether (ETH) is Ethereum's cryptocurrency. It is the fuel that runs the network. It is used to pay for the computational resources and the transaction fees for any transaction executed on the Ethereum network.
- Like Bitcoins, ether is a peer-to-peer currency. Apart from being used to pay for transactions, ether is also used to buy gas, which is used to pay for the computation of any transaction made on the Ethereum network.

- Also, if you want to deploy a contract on Ethereum, you will need gas, and you would have to pay for that gas in ether. So gas is the execution fee paid by a user for running a transaction in Ethereum.
- Ether can be utilized for building decentralized applications, building smart contracts, and making regular peer-to-peer payments.
- In Ethereum, the transaction fee is calculated in Ether, which is given as:
- **Ether=Tx Fees =Gas Limit*Gas price**

- **Network Types**
- As of March 2018, there are three types of Ethereum networks
- Public mainnet
- Private mainnet
- Testnet.
- According to Ethereum Foundation's research center, a new hybrid network called Whisper Network is under development.

- **1) Public mainnet**

- The public Ethereum network has two variations: Homestead and Metropolis.
- The first one was launched in March 2015 via a hard fork which resulted in a different set of network parameters that enabled smart contract transactions.

- In 2016, the Foundation planned another hard fork called Metropolis to improve efficiency and flexibility when dealing with transactions and smart contracts on the blockchain.

- **2) Private mainnet**

- In a private Ethereum network, membership is controlled and limited to certain users who are carefully selected by administrators.
- It is used to protect against malicious attacks and limit access only to trusted parties — which makes it an ideal solution for companies looking to develop applications based on cryptocurrency.
- Private networks can be built in-house or hosted by a third-party service provider such as Microsoft Azure or Amazon Web Services.

- **3) Testnet**

- The testnet is used by those who want to experiment with their code before launching it on the mainnet. Users who want to try out the testnet can create a new node and join the network — or they can connect their private networks to the public testnet.

- A functional public testnet is a necessary component of an Ethereum-based app as most developers use it to test their apps, smart contracts, and other solutions based on blockchain technology.

➢ Tools of Ethereum application development

1. Truffle

Truffle is a command-line development tool that offers a complete ecosystem for developing and testing Ethereum-based applications. What's more, is that Truffle comes with a configurable build pipeline support to make the development process more convenient.

2. Remix IDE

Remix IDE is an open-source, JavaScript-based debugging and compiling tool that is primarily used for writing Solidity contracts. The best aspect of Remix IDE is that you can use it both in the browser as well as locally. It uses Metamask to connect to the Ethereum framework.

3. MetaMask

MetaMask is a browser-based tool designed for Ethereum. In essence, it is a wallet that functions as a browser extension. As a browser extension for all major browsers (Chrome, Firefox, and Opera), MetaMask allows you to interact with the Ethereum framework in a and hassle-free manner.

4. Solidity

Solidity is the primary programming language used to write smart contracts on the Ethereum framework. It is a statically typed, high-level, contract-oriented programming language that draws inspiration from JavaScript, Python, and C++. By the phrase “contract-oriented,” we mean that smart contracts are designed to innately store all the programming logic that occurs within the Ethereum Blockchain.

- # Introduction to solidity programming
- Solidity is an object-oriented programming language created specifically by the Ethereum Network team for constructing and designing smart contracts on Blockchain platforms.
- It is High level language.
- Case sensitive.
- Solidity programs are Saved with Extension .sol
- It's used to create smart contracts that implement business logic and generate a chain of transaction records in the blockchain system.
- It acts as a tool for creating machine-level code and compiling it on the Ethereum Virtual Machine (EVM).
- It has a lot of similarities with C and C++ and is pretty simple to learn and understand. For example, a “main” in C is equivalent to a “contract” in Solidity.

- **Solidity program example**

```
pragma solidity >= 0.4.16 < 0.9.0;          //version
contract Test
{
   uint public var1;                    // Declaring state variables
   uint public var2;
   uint public sum;

     function set(uint x, uint y) public       // Defining public function
   {
      var1 = x;                          // that sets the value of
      var2 = y;
      sum=var1+var2;                      // the state variable
   }
      function get(  )                   // Defining function to
        public view returns (uint)
         {
      return sum;
         }
}
```

➢ What is Docker?

- Docker is an **open-source centralized platform designed** to create, deploy, and run applications.
- Docker uses **container** on the host's operating system to run applications.
- It allows applications to use the same **Linux kernel** as a system on the host computer, rather than creating a whole virtual operating system.
- Containers ensure that our application works in any environment like development, test, or production.

- **Docker Containers**
- Docker containers are the **lightweight** alternatives of the virtual machine.
- It allows developers to package up the application with all its libraries and dependencies, and ship it as a single package.
- The advantage of using a docker container is that you don't need to allocate any RAM and disk space for the applications. It automatically generates storage and space according to the application requirement.

- **Advantages of Docker**
- There are the following advantages of Docker -
  It runs the container in seconds instead of minutes.
- It uses less memory.
- It provides lightweight virtualization.
- It does not a require full operating system to run applications.
- It uses application dependencies to reduce the risk.
- Docker allows you to use a remote repository to share your container with others.
- It provides continuous deployment and testing environment.

- Challenges of blockchain technologies
- Blockchain technologies offer various benefits, such as decentralization, immutability, transparency, and security. However, they also come with their own set of challenges:

1.Scalability: One of the most significant challenges facing blockchain technology is scalability. Traditional blockchains like Bitcoin and Ethereum have limitations in terms of transaction throughput. As more transactions are added to the network, it becomes increasingly difficult to process them quickly and efficiently.

2.Energy Consumption: Proof-of-Work (PoW) consensus mechanisms, which are used by some blockchains like Bitcoin, require significant computational power and, consequently, consume alot of energy. This has raised concerns about the environmental impact of blockchain technology.

3. **Interoperability:** Different blockchain platforms often operate in isolation, making it challenging for them to communicate and share data. Interoperability solutions are needed to enable seamless interaction between different blockchains and traditional systems.

4. **Regulatory Uncertainty:** Blockchain technology operates across borders, which can create regulatory challenges. Different countries have varying regulations regarding cryptocurrencies, initial coin offerings (ICOs), and blockchain technology in general. Regulatory uncertainty can hinder adoption and innovation in the blockchain space.

5. **Privacy:** While blockchain transactions are often considered transparent and immutable, privacy can be a concern. Public blockchains typically store transaction data openly, which can compromise user privacy. Various techniques, such as zero-knowledge proofs and privacy coins, aim to address this issue.
6. **Security:** Although blockchain technology is touted for its security features, it's not immune to attacks. Potential vulnerabilities in smart contracts, consensus mechanisms, and decentralized applications (DApps) can be exploited by malicious actors. Additionally, the concentration of mining power in PoW blockchains poses a risk of 51% attacks.

➢ Applications of blockchain technology

- Blockchain technology has a wide range of applications across various industries due to its unique features such as decentralization, transparency, immutability, and security. Here are some notable applications:

1.Cryptocurrencies: The most well-known application of blockchain technology is cryptocurrencies like Bitcoin, Ethereum, and many others. These digital currencies enable peer-to-peer transactions without the need for intermediaries like banks.

2.Supply Chain Management: Blockchain can improve supply chain transparency and efficiency by enabling the tracking of goods from the point of origin to the final destination. This helps in verifying the authenticity of products, reducing counterfeiting, and streamlining processes like inventory management and logistics.

3. **Smart Contracts:** Smart contracts are self-executing contracts with the terms of the agreement directly written into code. Blockchain-based smart contracts automate and enforce the execution of contractual agreements, reducing the need for intermediaries and minimizing the risk of fraud or manipulation.
4. **Identity Management:** Blockchain technology can provide secure and decentralized identity management solutions. Individuals can have control over their digital identities, reducing the risk of identity theft and providing easier access to services that require identity verification.

5. **Healthcare:** Blockchain technology can improve the integrity and security of healthcare data by enabling the secure storage and sharing of medical records, facilitating interoperability between different healthcare providers, and ensuring patient consent and privacy.
6. **Voting Systems:** Blockchain-based voting systems can enhance the security, transparency, and integrity of elections by providing tamper-proof records of votes. This can help in reducing electoral fraud and increasing voter trust in the democratic process.