MODULE 5

Introduction to Ethereum

Ethereum is a blockchain-based computing platform that enables developers to build and deploy decentralized applications—meaning not run by a centralized authority. You can create a decentralized application for which the participants of that particular application are the decision-making authority. Ethereum is considered by many to be the second most popular cryptocurrency, surpassed now only by Bitcoin.

Ethereum Features

Ether: This is Ethereum's cryptocurrency.

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.

These are Ethereum's essential features. Let's discuss each of these features in more detail.

i. 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.

ii. Smart Contracts

Smart Contracts are revolutionizing how traditional contracts work. A smart contract is a simple computer program that facilitates the exchange of any asset between two parties. It could be money, shares, property, or any other digital asset that you want to exchange. Anyone on the Ethereum network can create these contracts. The contract consists primarily of the terms and conditions mutually agreed on between the parties (peers).

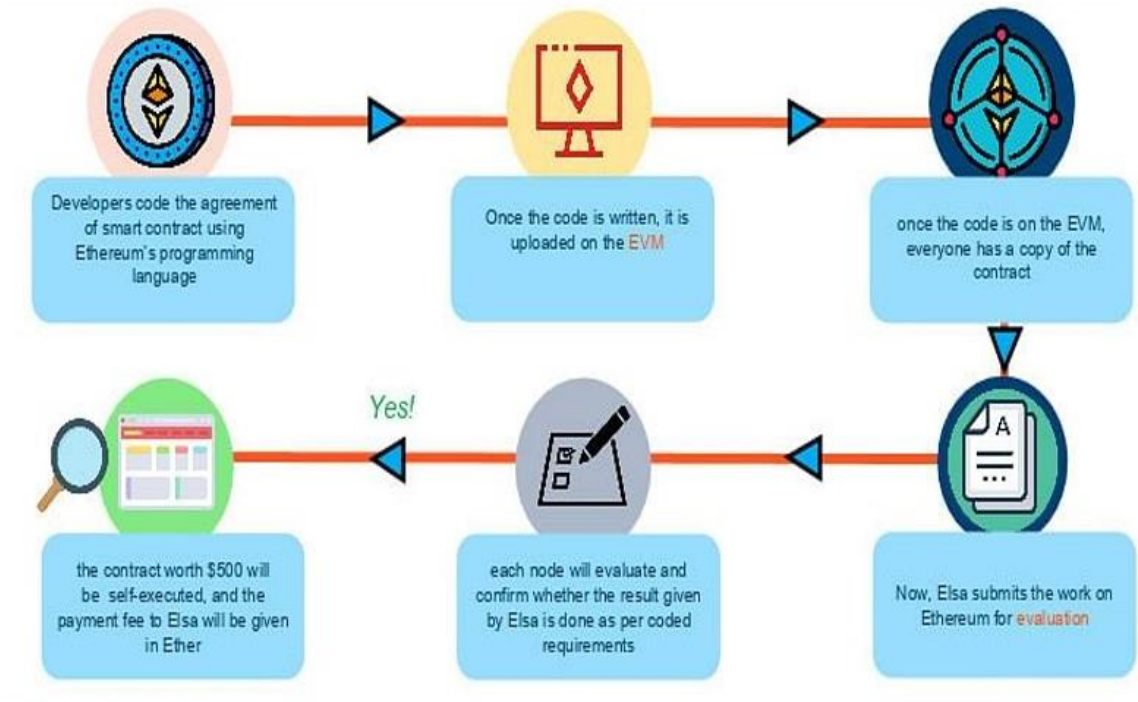
The smart contract's primary feature is that once it is executed, it cannot be altered, and any transaction done on top of a smart contract is registered permanently—it is immutable. So even if you modify the smart contract in the future, the transactions correlated with the original contract will not get altered; you cannot edit them. The verification process for the smart contracts is carried out by anonymous parties in the network without the need for a centralized

authority, and that's what makes any smart contract execution on Ethereum a decentralized execution.

The transfer of any asset or currency is done in a transparent and trustworthy manner, and the identities of the two entities are secure on the Ethereum network. Once the transaction is successfully done, the accounts of the sender and receiver are updated accordingly, and in this way, it generates trust between the parties.

In conventional contract systems, you sign an agreement, then you trust and hire a third party for its execution. The problem is that in this type of process, data tampering is possible. With smart contracts, the agreement is coded in a program. A centralized authority does not verify the result; it is confirmed by the participants on the Ethereum blockchain-based network. Once a contract is executed, the transaction is registered and cannot be altered or tampered, so it removes the risk of any data manipulation or alteration.

Let's take an example in which someone named Zack has given a contract of $500 to someone named Elsa for developing his company's website. The developers code the agreement of the smart contract using Ethereum's programming language. The smart contract has all the conditions (requirements) for building the website. Once the code is written, it is uploaded and deployed on the Ethereum Virtual Machine (EVM). EVM is a runtime compiler to execute a smart contract. Once the code is deployed on the EVM, every participant on the network has a copy of the contract. When Elsa submits the work on Ethereum for evaluation, each node on the Ethereum network will evaluate and confirm whether the result given by Elsa has been done as per the coding requirements. Once the result is approved and verified, the contract worth $500 will be self-executed, and the payment will be paid to Elsa in ether. Zack's account will be automatically debited, and Elsa will be credited with $500 in ether.



iii. Ehereum Virtual Machine

EVM is designed to operate as a runtime environment for compiling and deploying Ethereum-based smart contracts. EVM is the engine that understands the language of smart contracts, which

are written in the Solidity language for Ethereum. EVM is operated in a sandbox environment—basically, you can deploy your stand-alone environment, which can act as a testing and development environment. You can then test your smart contract (use it) “n” number of times, verify it, and once you are satisfied with the performance and the functionality of the smart contract, you can deploy it on the Ethereum main network.

Any programming language in the smart contract is compiled into the bytecode, which the EVM understands. This bytecode can be read and executed using the EVM. Solidity is one of the most popular languages for writing a smart contract. Once you write your smart contract in Solidity, that contract gets converted into the bytecode and gets deployed on the EVM, thereby guaranteeing security from cyberattacks.

Suppose person A wants to pay person B 10 ethers. The transaction will be sent to the EVM using a smart contract for a fund transfer from A to B. To validate the transaction; the Ethereum network will perform the proof-of-work consensus algorithm. The miner nodes on Ethereum will validate this transaction—whether the identity of A exists or not, and if A has the requested amount to transfer. Once the transaction is confirmed, the ether will be debited from A’s wallet and will be credited to B’s wallet, and during this process, the miners will charge a fee to validate this transaction and will earn a reward. All the nodes on the Ethereum network execute smart contracts using their respective EVMs.

Proof of Work

Every node in the Ethereum network has:

- The entire history of all the transactions—the entire chain
- The history of the smart contract, which is the address at which the smart contract is deployed, along with the transactions associated with the smart contract
- The handle to the current state of the smart contract

The goal of the miners on the Ethereum network is to validate the blocks. For each block of a transaction, miners use their computational power and resources to get the appropriate hash value by varying the nonce. The miners will vary the nonce and pass it through a hashing algorithm—in Ethereum, it is the Ethash algorithm. This produces a hash value that should be less than the predefined target as per the proof-of-work consensus. If the hash value generated is less than the target value, then the block is considered to be verified, and the miner gets rewarded. When the proof of work is solved, the result is broadcast and shared with all the other nodes to update their ledger. If other nodes accept the hashed block as valid, then the block gets added to the Ethereum main blockchain, and as a result, the miner receives a reward, which as of today stands at three ethers. Plus, the miner gets the transaction fees that have been generated for verifying the block. All the transactions that are aggregated in the block—the cumulative transaction fees associated with all the transactions are also rewarded to the miner.

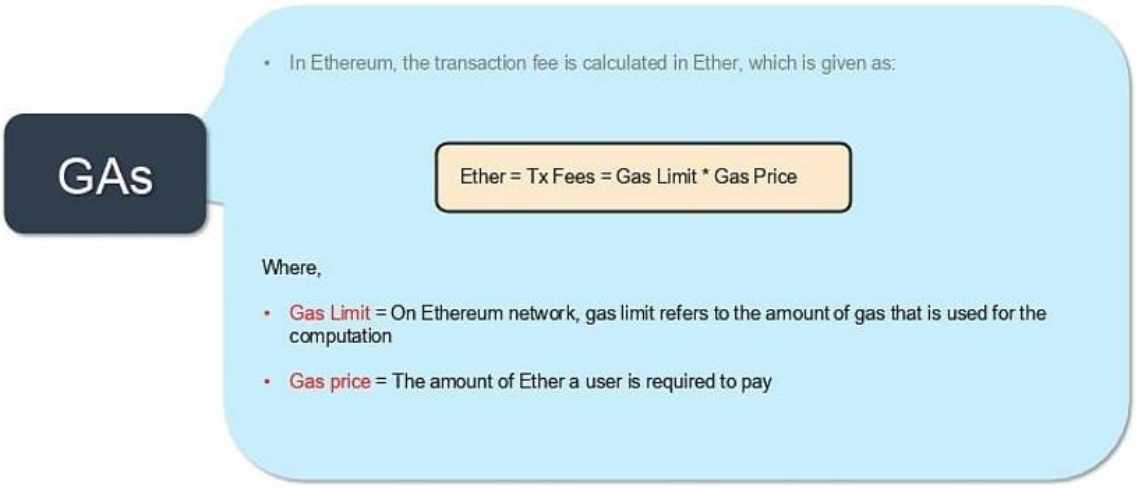
Proof of Stake

In Ethereum, a process called proof of stake is also under development. It is an alternative to proof of work and is meant to be a solution to minimize the use of expensive resources spent on mining using proof of work. In proof of stake, the miner—who is the validator—can validate the transactions based on the number of crypto coins he or she holds before actually starting the mining. So, based on the accumulation of crypto coins the miner has beforehand, he or she has a higher probability of mining the block. However, proof of stake is not widely used as of now compared to proof of work.

Gas

Just like we need fuel to run a car, we need gas to run applications on the Ethereum network. To perform any transaction within the Ethereum network, a user must make a payment, in this case paying out ethers, to get a transaction done, and the intermediary monetary value is called gas. On the Ethereum network, gas is a unit that measures the computational power required to run a smart contract or a transaction. So, if you must do a transaction that updates the blockchain, you would have to shell out gas, and that gas costs ethers.

In Ethereum, the transaction fees are calculated using a formula. For every transaction, there is gas and its correlated gas price. The transaction fees equal the amount of gas required to execute a transaction multiplied by the gas price. "Gas limit" refers to the amount of gas used for the computation and the amount of ether a user is required to pay for the gas.



For a particular transaction, if the gas limit was 21,000, the gas used by the transaction was 21,000, and the gas price was 21 Gwei, which is the lowest denomination of ether. So, 21 Gwei * 21,000 gave the actual transaction fees: 0.000441 ethers, or about 21 cents as of today. As mentioned, the transaction fee goes to the miner, who has validated the transaction.

To understand the gas limit and price, let's consider an example using a car. Suppose your vehicle has a mileage of 10 kilometers per litre and petrol costs $1 per litre. Under these parameters, driving a car for 50 kilometers would cost you five litres of petrol, which is worth $5. Similarly, to perform an operation or to run code on Ethereum, you need to obtain a certain amount of gas, like petrol, and the gas has a per-unit price, called gas price. If the user provides less than the amount of gas to run an operation, then the process will fail, and the user will be given the message "out of gas." And Gwei is the lowest denomination of ether used for measuring a unit of a gas price.

iv. Decentralized Applications (Dapps)

Let's compare decentralized applications with traditional applications. When you log in to Twitter, for example, a web application gets displayed that is rendered using HTML. The page will call an API to access your data (your information), which is centrally hosted. It's a simple process: your front end executes the backend API, and the API goes and fetches your data from a centralized database.

If we transform this application into a decentralized application when you log in, the same web application gets rendered, but it calls a smart contract-based API to fetch the information from the blockchain network. So, the API is replaced by a smart contract interface, and the smart contract will bring the data from the blockchain network, which is its back end. That blockchain network is not a centralized database; it's a decentralized network in which the participants of the network (the miners) validate (verify) all the transactions that are happening using the smart contract on the blockchain network. So, any transaction or action happening on a Twitter-type application that has now been transformed will be a decentralized transaction.

A Dapp consists of a backing code that runs on a distributed peer-to-peer network. It is a software designed to work in the Ethereum network without being controlled by a centralized system, as mentioned, and that is the primary difference: it provides direct interaction between the end-users and the decentralized application providers.

An application qualifies as a Dapp when it is open-source (its code is on Github), and it uses a public blockchain-based token to run its applications. A token acts as fuel for the decentralized application to run. Dapp allows the back end code and data to be decentralized, and that is the primary architecture of any Dapp.

v. Decentralized Autonomous Organizations (DAOs)

DAO is a digital organization that operates without hierarchical management; it works in a decentralized and democratic fashion. So basically, a DAO is an organization in which the decision-making is not in the hands of a centralized authority but preferably in the hands of certain designated authorities or a group or designated people as a part of an authority. It exists on a blockchain network, where it is governed by the protocols embedded in a smart contract, and thereby, DAOs rely on smart contracts for decision-making—or, we can say, decentralized voting systems—within the organization. So, before any organizational decision can be made, it must go through the voting system, which runs on a decentralized application.

People add funds through the DAO because the DAO requires funding in order to execute and make decisions. Based on that, each member is given a token that represents that person's percentage of shares in the DAO. Those tokens are used to vote in the DAO, and the proposal status is decided based on the maximum votes. Every decision within the organization must go through this voting process.

Real-World Applications of Ethereum

- **Voting Systems**

As we've seen with DAO, voting systems are adopting Ethereum. The results of polls are publicly available, ensuring a transparent and fair democratic process by eliminating voting malpractices.

- **Banking Systems**

Ethereum is getting adopted widely in banking systems because with Ethereum's decentralized system; it is challenging for hackers to gain unauthorized access. It also allows payments on an Ethereum-based network, so banks are also using Ethereum as a channel to make remittances and payments.

- **Shipping**

Deploying Ethereum in shipping helps with the tracking of cargo and prevents goods from being misplaced or counterfeited. Ethereum provides the provenance and tracking framework for any asset required in a typical supply chain.

- **Agreements**

With Ethereum smart contracts, agreements can be maintained and executed without any alteration. So in an industry that has fragmented participants, is subject to disputes, and requires digital contracts to be present, Ethereum can be used as a technology for developing smart contracts and for digitally recording the agreements and the transactions based on them.

Advantages and Disadvantages of Ehereum

Here are the advantages of Ethereum enjoyed by the enterprises:

- Decentralization:

The decentralized design of Ethereum effectively distributes knowledge and trust among network members, removing the need for a central body to run the system and mediate transactions.

- Rapid deployment:

Instead of building a blockchain implementation from scratch, organizations can quickly create and administer private blockchain networks using an all-in-one SaaS platform like Hyperledger Besu.

- Permissioned network:

There are many open-source protocol layers that allow enterprises to build on public or private Ethereum networks, guaranteeing that their solution meets all regulatory and security standards.

- Network size:

The Ethereum mainnet demonstrates that a network with hundreds of nodes and millions of users can function. Most business blockchain competitors run networks with less than ten nodes and have no precedent for a large and successful network. For corporate collaborations that are bound to outgrow a few nodes, network scale is important.

- Private transactions:

In Ethereum, businesses may obtain privacy granularity by joining private partnerships with private transaction layers. Private information is encrypted and only shared with those who need to know.

- Scalability and performance:

Consortium networks created on Ethereum may outperform the public mainnet and grow up to hundreds of transactions per second or more depending on network setup, thanks to Proof of Authority consensus and bespoke block time and gas limits. Ethereum will be able to boost its throughput in the near future because of protocol-level solutions like sharding and off-chain, as well as layer 2 scaling solutions like Plasma and state channels.

- Finality:

The consensus method of a blockchain ensures that the transaction record is tamper-proof and canonical. For different enterprise network instances, Ethereum offers customizable consensus

mechanisms such as RAFT and IBFT, ensuring immediate transaction finality and reducing the required infrastructure that the Proof of Work algorithm requires.

- Tokenization:

Any item that has been registered in a digital format can be tokenized on Ethereum. Organizations may fractionalize formerly monolithic assets (real estate), broaden their product line (provably rare art), and open new incentive models by tokenizing assets (crowdsourced data management).

- Interoperability and open source:

On Ethereum, consortiums are not bound by a single vendor's IT environment. Customers of Amazon Web Services, for example, can use Kaleido's Blockchain Business Cloud to run private networks. The Ethereum ecosystem, like the Java community, encourages contributions to the codebase through Ethereum Improvement Proposals (EIPs).

- Standards:

The ecosystem is kept from being fragmented through protocols for token design (ERC20), human-readable names (ENS), decentralized storage (Swarm), and decentralized communications (Whisper). The Client Specification 1.0 of the Corporate Ethereum Alliance outlines the architectural components for compatible enterprise blockchain implementations.

Ethereum has some disadvantages as well:

- Uses a Complicated Programming Language:

While Ethereum is Turing complete and uses a programming language similar to C++, Python, and Java, learning Solidity, the native language of Ethereum, may be challenging. One of the most significant concerns is the scarcity of beginner-friendly classes.

- Issues with Scaling:

Unlike Bitcoin, which has a singular purpose, Ethereum has a ledger, a platform for smart contracts, and so on, all of which may lead to errors, malfunctions, and hacks.

- Ethereum Investing Can Be Risky:

Ethereum investing, like any other cryptocurrency, can be risky. Cryptocurrencies are very volatile, resulting in significant gains as well as significant losses. The price of Ether has changed significantly in the past, which might be a significant disadvantage for certain investors, particularly beginners. In addition, Ethereum's fees change, which is inconvenient.

Ethereum Vs. Bitcoin

The Bitcoin vs. Ethereum argument has been garnering more attention these days. Bitcoin has become a very popular and well-known cryptocurrency around the world. It also has the highest market cap among all the cryptocurrencies available right now. In a way, it’s the current world champion when it comes to cryptocurrencies. On the other side is Ethereum. Ethereum did not have the revolutionary effect that Bitcoin did, but its creator learned from Bitcoin and produced more functionalities based on the concepts of Bitcoin. It is the second-most-valuable cryptocurrency on the market.

- **History**

Bitcoin was the first cryptocurrency to be created; as mentioned, it was released in 2009 by Satoshi Nakamoto. It is not known if this is a person or group of people, or if the person or people are alive or dead. Ethereum, as noted above, was released in 2015 by a researcher and programmer named Vitalik Buterin. He used the concepts of blockchain and Bitcoin and improved upon the platform, providing a lot more functionality. Buterin created the Ethereum platform for distributed applications and smart contracts.

- **Concepts**

Bitcoin enables peer-to-peer transactions. It acts as a replacement for fiat currencies but doesn't have all the problems associated with fiat currencies. You don't have to pay high transaction fees, and you also don't have a centralized authority that regulates how bitcoins work. Ethereum enables peer-to-peer transactions as well, but it also provides a platform for creating and building smart contracts and distributed applications. A smart contract allows users to exchange just about anything of value: shares, money, real estate, and so on.

- **Mining**

In Bitcoin, miners can validate transactions with the method known as proof of work. This is the same case for Ethereum. With proof of work, miners around the world try to solve a complicated mathematical puzzle to be the first one to add a block to the blockchain. Ethereum, however, is working on moving to a different form of transaction validation known as proof of stake. With proof of stake, a person can mine or validate transactions in a block based on how many coins he owns. The more coins a person holds, the more mining power he will have. In Bitcoin, every time a miner adds a block to the blockchain, he is rewarded with 6.25 bitcoins, a rate set in November 2021. In Etherium a miner, or validator, receives a value of 3 ether every time a block is added to the blockchain, and the reward will never be halved.

- **Fees**

The transaction fees in Bitcoin are entirely optional. You can pay the miner more money to have him pay special attention to your transaction; however, the transaction will go through even if you don't pay a fee. On the other hand, you must provide some amount of ether for your transaction to be successful on Ethereum. The ether you offer will get converted into a unit called gas. This gas drives the computation that allows your transaction to be added to the blockchain.

- **Time**

As for the average amount of time it takes to add a block to the blockchain, in Bitcoin it takes 10 minutes. In Ethereum, it takes only about 12 to 15 seconds.

- **Hashing Algorithms**

Hashing algorithms are how these systems can maintain their privacy and ensure security. Bitcoin uses a hashing algorithm known as SHA-256. Ethereum uses a cryptographic algorithm called Ethash.

- **By the numbers**

Bitcoin has over 18 million bitcoins currently in existence, and Ethereum has 118 million ether. Now even though Ethereum has easily crossed the 100 million mark, the market capitalization for Bitcoin is $781 billion, whereas for Ethereum it's only $368 billion. So even though Ethereum has more coins on the market, it isn't at the level of Bitcoin.

The number of Bitcoin transactions that take place in a day currently hovers around 260,000; for Ethereum, it's about 1.2 million. As for the number of blocks that have been mined, for Bitcoin, it's over 718,000, and for Ethereum it's about 13 million. This has a lot to do with the fact that it takes a lot less time for a block to be added to Ethereum than to Bitcoin.

The current block size is 1,268 kilobytes for Bitcoin and 94 kilobytes for Ethereum. And while the market value of Bitcoin is significantly higher than that of any form of digital currency on the market right now, it is closely followed by Ethereum, which hopes to take over one day.

The answer to the question of which cryptocurrency is better in the choice between Bitcoin vs. Ethereum, it depends entirely on requirements. While Bitcoin works better as a peer-to-peer transaction system, Ethereum works well when you need to create and build distributed applications and smart contracts.

Introduction to Smart Contracts

Smart contracts are a type of digital agreement based on blockchain technology and are executed to form a legal contract between two parties involved. Smart contracts have been in the market even before the advent of blockchain technology. However, recent developments in the blockchain have paved the way to an increasingly secure form of smart contracts. These are now gradually being adopted in the market. Smart Contracts are one of the most promising applications based on blockchain technology.

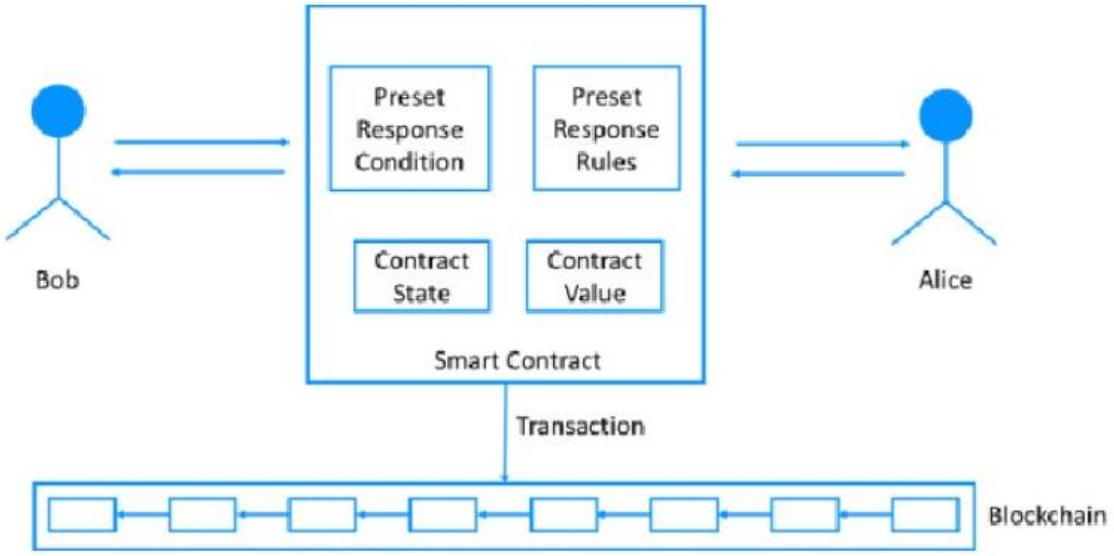
i. What are Smart Contracts?

Smart contracts are a set of conditions written in the form of code that meet the agreed criteria between two involved parties. The code that resides on the blockchain is distributed and highly secure. This piece of code, when executed, gets registered in the form of a blockchain transaction while ensuring the terms of the agreement are met. This execution of smart contracts on the blockchain is immutable and irreversible. Hence, any smart contract execution is tracked from beginning to end of the transaction cycle between the two parties involved. It can be tracked chronologically in a transparent manner.

ii. Why did the need to introduce smart contracts arise?

With the ever-advancing digital era, smart contracts have gained popularity for preparing and executing agreements. These can be implemented among any two parties residing anywhere across the globe. It is easy to implement, transparent, accessible, and free of distance and geographical constraints.

For instance, an organization based out of India can get into an agreement with a client from Europe without having to physically travel and sign a legal agreement. Once both parties reach a consensus on the terms and conditions of the agreement, a smart contract is designed using the ' if, else … then' – statements with the help of a code. It is implemented over blockchain and registered as a blockchain transaction while minimizing frictions involved in a legal process.



The flow of a Smart Contract execution

Smart contracts can be used for many different domains, as explained below:

Real estate transactions

Any real estate transaction in buying, selling, renting, or listing a property for rent/sale can be executed using multiple contracts hosted on a blockchain.

Cryptocurrency investments

To invest in Cryptocurrency, there are multiple platforms in the market like CoinDCX, CoinBase, UnoCoin, etc.

Other popular use – cases are in the domains of Credit Lending, Medical Records preservation, and access and Identity verification and management.

iii. Features of Smart Contracts

Secure – Smart contracts are a piece of code designed to minimize human error and issues. They are highly secure as they reside on the blockchain distributed across multiple systems and are immutable.

Scalable – Smart contracts are easy, efficient, offer greater execution speed, and promise higher accuracy.

Easy to implement and track – Once a smart contract is written in code on blockchain technology, it can be easily executed with the click and consent of both parties. Since it is registered as a transactional record in the blockchain, it is also easy to track.

Reduce Friction – Smart contracts reduce friction incidents involving intermediaries, geographical constraints, fees of execution, and commission.

Transparent – data, agreement terms, and execution history can be shared transparently with all involved parties.

Encourages Savings in terms of time, energy and finances – Smart contracts are executed when certain conditions decided by the participating parties are met and do not need an intermediary to solicit the agreement. This significantly reduces the time spent in the implementation of the contract. Moreover, they are a cost-effective proposition as they save on the cost of legal intervention.

iv. Use of Smart Contracts in Day-to-day Business Deals and Projects

Smart contracts can be widely used by businesses across domains like healthcare, e-commerce, real estate, decentralized finance, and more. They are developed on the Ethereum blockchain using the Solidity programming language, an open-source blockchain where community help is available. Smart contracts are used in many day-to-day business processes, including legal agreements, timeline bindings, and business terms between two parties.

Let's discuss this with the help of two examples:

- Suppose you are a cosmetics vendor who wants to list and sell your products on an online e-commerce platform. As a cosmetics vendor wanting to sell your products on an e-commerce platform, you must first register on their portal and list yourself as a retailer. Post registration, you will be required to sign an agreement with details of the listed products, a revenue sharing agreement based upon the sales, and other contract terms. This form of agreement can be developed on the Ethereum blockchain with the help of Solidity, a programming language. The execution will be much faster once both parties agree on the terms of the agreement.
- Imagine you are a mediator who deals in rental properties and sales in a specific area in Bengaluru. If you deal with property and real estate rentals and purchases in Bengaluru, Smart Contracts are immensely beneficial.
- Smart Contract for the Sale of the Real Estate Property: The terms of the deal would involve –

    - The cost of the property
    - Ownership transfer
    - Purchase agreement
    - Number of parking
    - Maintenance charges
    - Society formation details along with other information

    A smart contract listing the above agreement terms is developed and executed, making the deals faster and more robust. Moreover, it is highly secure as the transaction records are immutable and stored as a Blockchain transaction.

- Smart contract for renting a Real estate property: For rental agreements, smart contracts are designed in a standard format listing the following:

    - Details of the Broker, owners, and tenant
    - Lease period
    - Lock-in period
    - Date of rent transfer every month as per the rent cycle
    - Terms of hand-over at the end of the rent tenure

Suppose a standard smart contract format is designed using solidity on the Ethereum blockchain and exposed as an API. In that case, brokers can feed in the required details and execute the rent agreement without any third party.

Law and Regulations

Given the unique nature of smart contracts and the ways they differ from traditional paper agreements, there are concerns surrounding their enforceability. In general, smart contracts are enforceable as long as they follow the basic rules of contractual agreements. These include the following.

a. Offer, Acceptance, Consideration

As with any agreement, there must be an offer, an acceptance of that offer and consideration.

• Offer: One or both parties offer the terms of the agreement.
• Acceptance: Both parties accept the terms as offered (often after some negotiation).
• Consideration: There is something of value being offered to each party.
If any of these components are lacking, it is not a legally enforceable contract.

b. Legally Permissible Terms

In general, you cannot use a contract to bind parties to terms that are illegal to enforce. For instance, asking parties to waive certain rights that cannot be legally waived will likely nullify that section of the agreement. This may present a special challenge for smart contracts since making sure such terms are severable from the rest of the agreement—which cannot be edited once executed—may be more difficult than it would be with a paper contract.

c. Legal To Sign Electronically

Finally, smart contracts need to be legally eligible for electronic signatures. Some types of agreements cannot be signed electronically, including wills and other estate documents; court orders; product recall notices involving health and safety; documents required to accompany hazardous substances being transported; notices of cancellation of utility services; and eviction notices. Most transactions involving smart contracts won't involve any of these categories, but it's still worth remembering what can and can't be signed electronically.

Legal Challenges of Smart Contracts

Smart contracts execute automatically, and once they're set in place, they cannot be modified. These facts create some interesting challenges, particularly in the event of disputes or unenforceability.

- **Automatic Enforcement:** If it turns out that the terms of a smart contract are not legal to enforce, it creates a more difficult situation than one would have with a traditional paper contract. Once the contract is programmed and agreed upon, it will execute automatically, which may lead to some difficulty in remedying any unlawful enforcement.
- **Modifying the Contract:** Making modifications to the contract can also be a challenge, at least once it's set into motion. Once a smart contract is in force, it cannot be modified. This means if any changes are desired, the entire contract needs to be canceled and redrawn. For this reason, maintaining a backup copy of the code is recommended.

- **Handling Disputes:** Due to the difficulty of adjusting a smart contract once it's in place, it's important for each party to be absolutely clear on the terms from the outset. The agreement needs to be treated as if it's going to be permanent from the very beginning, so great care should be taken to make sure it doesn't lead to disputes. If a dispute does occur, both the contract's permanence and automatic execution could pose a barrier to enacting changes.

Case Studies

Now you understand how smart contracts work, let's look at some smart contract examples from the real world.

Clinical trials

Data sharing between institutions is vital to effective clinical trials. With the support of smart contracts, professionals can seamlessly share data across the industry. Blockchain technology can also help with the authentication of the data to ensure it is accurate. This is a gamechanger for those trying to launch wide-reaching clinical trials. Smart contracts have many uses in the healthcare industry.

Music industry

Emerging music artists depend on streaming income as they get started in the industry. Smart contract applications can make royalty payments easier. For instance, these contracts can include which percentage of the royalty income goes to the record label and the artist. These payments can happen instantly, which is a major win for all parties involved. Tune.fm, for example, is a tokenized music economy that helps artists get paid directly for every second streamed using JAM tokens. Artists can mint NFTs for exclusive content and sell them directly to fans for JAM tokens.

Supply chain management

As self-enforcing contracts, smart contracts can operate autonomously without the need for any intermediaries or third parties. If you designed a smart contract for an end-to-end supply chain, this would require no daily management or auditing. Any deliveries received outside the schedule could trigger pre-agreed escalation measures to ensure a smooth operation.

Datahash, formerly Entrust, is Australia's first full-service agricultural supply chain platform. It is working to thwart the $3 billion-a-year market in fraudulent wine. The platform relies on Hedera Consensus Service to trace its data in a trusted way.

Property ownership

You can use smart contract technology to offer fractional ownership of real estate. Rather than one person owning a property, you can segment ownership so people can buy tokens of the property. When someone owns a token, they co-own a percentage of the property. This makes it easy for anyone to jump on the property market and make micro-investments.

Mortgages

The mortgage industry needs a massive overhaul. It's currently bloated with costly third parties and time-consuming processes. Smart contracts can ensure that lenders and loan seekers agree to clear terms and conditions, such as proof-of-funds and payment planning. This emerging technology can validate mortgage transactions without the need for any lawyers or other third parties.

Retail

Smart contracts can help to streamline administrative processes that are often a burden to brick-and-mortar retailers. Retailers can create smart contracts to enable fast payments to contractors. Another possibility: Digitize payroll administration and track it in real-time. Retailers also can place unique blockchain identifiers on inventory units to create visibility across supply chains.

In this sector, Dropp enables micropayments for small value transactions in both cryptocurrency and dollars. Merchants save money and grow their business, and consumers get convenient access to products and services.

Digital identity

From reputational data to digital assets, you can store components on a smart contract to form a digital identity. When smart contracts are connected to various online services, the counterparties can learn about the individuals without revealing their identities. Smart contracts could contain credit scores that lenders can use to measure potential risk.

For example, MyEarth ID is a decentralized Identity Management System that allows users to control their digital identity data and securely verify it with third parties.

Recording financial data

Smart contracts can help to facilitate accurate and transparent data collection. When it comes to recording financial data, smart contracts can radically reduce costs for auditing and ensure compliance. These smart contracts can execute set financial rules without the need for any intervention. This can streamline administrative workflows and save accountants time.

At another intersection of smart contracts and finance, AllianceBlock is building a protocol to bridge decentralized finance (DeFi) and traditional financial services (TradFi). AllianceBlock's AllianceBridge is a validator network leveraging the Hedera Consensus Service.

Voting in elections

Smart contracts could create a secure environment for voting, reducing the risk of potential voter manipulation. Each vote using a smart contract is ledger-protected. Due to the encryption, these are incredibly hard to decode. Smart contracts could also increase voter turnout. With an online system powered by smart contracts, there is no need to travel to a polling station.

Insurance sector

The insurance world is full of disputes. With this in mind, smart contracts have an important role to play in automating policies and services in the insurance industry. This can help to reduce insurer costs and result in lower premiums. With automated claims payment processes powered by smart contract technology, policy-holders can get paid faster than through current manual processes.