**Lecture 4-4 Notes**

**DECENTRALIZED APPLICATIONS**

* They are more flexible, transparent, distributed, resilient, and have a better incentivized structure than current software models.
* It will work with cryptographically stored ledger.
* It is a scarce-asset model, and peer-to-peer technology.
* Decentralized applications don‘t necessarily need to run on top of a blockchain network.
* Tor, BitTorrent, Popcorn Time, BitMessage, are examples for decentralized applications that run on a P2P network, but not on a blockchain which is a special kind of P2P network.
* Decentralized applications are a piece of so ware that communicates with the blockchain, which manages the state of all network actors.
* The interface of the decentralized applications does not look any different than any website or mobile app today.
* The smart contract represents the core logic of a decentralized application. Smart contracts are integral building blocks of blockchains, that process information from external sensors or events and help the blockchain manage the state of all network actors.

**4.2 Key Features of Dapps**

* A distributed app, abbreviated as “DApp", is a piece of software that is run on a distributed or cloud network rather than on a single dedicated server.
* By distributing the processing power and storage space across many devices, DApps are decentralized, making them more resistant to attack as there is no single point of failure that can be undermined.
* Blockchain-based systems have been the platforms upon which many DApps have been built and deployed; however, DApps can also runs on cloud platforms or other network architectures.
* **Open Source:** Ideally, it should be governed by autonomy and all changes must be decided by the consensus, or a majority, of its users. Its code base should be available for scrutiny.
* **Decentralized:** All records of the application’s operation must be stored on a public and decentralized blockchain to avoid pitfalls of centralization.
* **Incentivized:** Validators of the blockchain should be incentivized by rewarding them accordingly with cryptographic tokens.
* **Protocol:** The application community must agree on a cryptographic algorithm to show proof of value. For example, Bitcoin uses Proof of Work (PoW) and Ethereum is currently using PoW with plans for a hybrid PoW/Proof of Stake (PoS) in the future.

**4.3 Illustration of Dapp**

* The frontend of a decentralized application represents what you see, and the backend represents the entire business logic.
* This business logic is represented by one or several smart contracts interacting with the underlying blockchain.
* The frontend, as well as files like a photo, a video, or audio, could be hosted on decentralized storage networks such as Swarm or IPFS.
* Traditional Web applications use HTML, CSS, and javascript or the like to render a webpage.
* This page interacts with a centralized database, where all the data is stored. When you use a service like Twitter, Facebook, Amazon, or Airbnb, for example, the webpage will call an API to process your personal data and other necessary information stored on their servers, to display them on the page.
* User ID and passwords are used for identification and authentication, with low levels of security, since personalized data is stored on the server of the service provider.

Traditional websites: Front End → API → Database.

* Decentralized applications are to a traditional Web application.
* The frontend uses the exact same technology to render the page. It contains a “wallet” that communicates with the blockchain.
* The wallet manages cryptographic keys and the blockchain address. Public-key infrastructure is used for user identification and authentication.
* Instead of an API connecting to a database, a wallet software triggers activities of a smart contract, which interacts with a blockchain: Web3 compatible website:

Front End (including wallet) → Smart Contract → Blockchain.

* In contrast to Web2 applications, Web3 applications need a connection to the blockchain, which is managed by a special application called “wallet.”
* It keeps a record of the private keys and blockchain address, which represents the unique 30 identities and point of reference.
* Without a so ware that manages our digital identity, we will not be able to interact with the blockchain.
* The Web3, therefore, builds on the current Web2 stack and introduces additional elements on an application level.
* In the backend, the Web3 adds a whole new infrastructure layer for decentralized applications to interact with the decentralized protocol stack.
* Decentralized apps need to have a component that manages a user’s private keys, with which one can sign transactions on the state layer, the blockchain.

**4.4 Blockchain and Dapps**

* In the crypto economy, the blockchain used by most cryptocurrencies uses Distributed Apps to maintain an efficient digital marketplace.
* Rather than the conventional client-server network adopted by most centralized organizations, blockchains run on a peer-to-peer network where transactional information carried out between two parties is recorded and shared across multiple computers on the network.

**4.5 Traditional V/S Distributed Apps**

* Decentralized applications are a significant shift from traditional apps, mainly because of the underlying principle of decentralization that drives the entire lifecycle.
* Trust and data immutability are another factor that differentiates the two. Unlike traditional apps that have a centralized controlling authority, dApps run on a decentralized blockchain and the data once written on the chain cannot be changed or erased.
* Therefore, at every stage of the dApp development lifecycle, you need to ensure that you apply the principles of the decentralized network.
* Furthermore, in the case of traditional apps, one assumes that once the interface design is completed, the app will run faster since there is less dependency on third parties.
* On the other hand, when developing a decentralized app, one needs to design for trust and scalability.
* Another major difference between the two is the rigor with which a dApp code needs to be tested before it is pushed to mainnet.
* Since a smart contract cannot be changed once it is launched, it is important to ensure that it is bug-free.
* This approach of ensuring certainty by maintaining caution is very different from the iterative “fail fast, learn faster” approach followed in the case of traditional app development.

**4.6 How dApps fit in the real world?**

the first known dApp in the world was the Bitcoin. Popular as an apex cryptocurrency, bitcoin solves the centralization issue and gives users the power to perform transactions without any middleman or central authority via a self-sustaining public ledger.

Coming to the use case of decentralized applications, we can classify these apps based on the scenario they can be infused into. This classification categorizes dApps into three segments that are also its potential use cases:

* **Money Management & Transfer**

Decentralized apps can be used to smoothen money transfer in the world. We have already seen the benefits in the form of success of bitcoin and other cryptocurrencies. Using blockchain network and its own crypto token, dApps can speed up money management, transfer, and lending by eliminating middlemen and enhance security due to the consensus mechanism that is impossible to change without a majority.

* **Business Process Management**

Companies can integrate decentralized applications to streamline processes without human intervention. With the help of smart contracts- an essential cog in the blockchain network, critical issues can be solved and process efficiency can be enhanced. For example, logistics companies can integrate RFID chips into their shipment that can be scanned at the destination ports on which the payment can be settled automatically through a smart contract between buyer and seller.

* **DAO (Decentralized Autonomous Organization)**

DAO is a completely new phenomenon of starting faceless organizations without leaders. These organizations can work as a corporate and run via rules that are defined by programming languages on the blockchain. How members will vote, what business segment will the organization operate in, who can be members, how will the value token be exchanged, everything can be programmed on the blockchain that will run the organization. These organization cannot stop once deployed and can work worldwide, without any trouble riding on the blockchain network bandwagon.

4.7 **WHY CHOOSE DAPP DEVELOPMENT OVER CONVENTIONAL APP DEVELOPMENT?**

* Faster & payment processing without needing to integrate payment gateway to accept funds.
* High levels of data security due to smart contracts governed by private keys.
* Greater anonymity without needing the users to follow the lengthy signup process.
* Reliable data records as users can access the public blockchain to verify transaction information.