NAME: PATHAK SEJAL SUDHIR
ROLL NO: BE 46
SUBJECT: LP III - GROUP C
Blockchain Technology
ASSIGNMENT NO: 01

TITLE: Installation of MetaMask & study spending Ether per transaction.

OBJECTIVE: Should able to learn new technology such as metamask. Its application & implementation

Prerequisite: 1. Basic knowledge of cryptography
2. Basic knowledge of distributed computing concept
3. Working of blockchain

THEORY:-

INTRODUCTION to BLOCKCHAIN:- ① Blockchain is data structure that holds transactional records and while ensuring security, transparency, decentralization. You can also think of it as a chain or records stored in the forms of blocks which are controlled by no single authority.

② Each transaction on a blockchain is secured with a digital signature that proves its authenticity. Due to the use of encryption & digital signatures, the data stored on the blockchain is tamper-proof and cannot be changed.

Blockchain Features:

1] Decentralized :- no single person or group holds the authority of the overall network. This unique feature of it is allows transparency and security while giving power to the users.

2] Peer-to-Peer Network :- It allows all the network participants to hold an identical copy of transactions, enabling approval through a machine consensus.

3] Immutable :- Any data once written on the blockchain cannot be changed. There are two key ways of detecting tampering namely, hashes and blocks.

Popular Applications of Blockchain Technology.

1. Smart Property.
2. Distributed Cloud Storage
3. Healthcare
4. Machine Learning
5. Digital Identity
6. Energy.

Benefits of Blockchain Technology :-

1) Time saving - No central Authority verification needed for settlements making the process faster and cheaper.

2) Cost-saving :- A blockchain network reduces expenses in several ways. No need for the third-party verification. Participants can share assets directly. Intermediaries are reduced. Transaction efforts are minimized as every participant has a copy of shared ledger.

3) Tighter security : No one can temper with Blockchain data as it is shared among millions of participants. The system is safe against cybercrimes and fraud.

4) In finance market-trading, Fibonacci retracement levels are widely used in technical analysis.

How to use MetaMask

Metamask is one of the most popular extensions that serves as a way of storing your Ethereum and other ERC-20 tokens. The extension is free & secure, allowing web applications to read and interact with Ethereum's blockchain.

Step 1 - Install MetaMask on your browser.
Step 2 Create an account
Step 3 Depositing Funds

ADVANTAGES OF MetaMask :-

1] Popular :- It is commonly used, so users only need one plugin to access a wide range of apps.

2] Simple :- Instead of managing private keys, users just need to remember a list of words, and transactions are signed on their behalf.

3] Saves space ; Users don't have to download the Ethereum blockchain, as MetaMask sends requests to nodes outside of the user's computer.

4] Integrated :- Dapps are designed to work with MetaMask, so it becomes much easier to send Ether in and out.

CONCLUSION :-

In this way we explored concept Blockchain and meramat Wallet For transaction of digital currency.

Page No.

NAME: PATHAK SEJAL SUDHIR
ROLL NO: BE 46
SUBJECT: LP III GROUP C
Blockchain Technology
ASSIGNMENT NO:-02

TITLE:- Create your own wallet using Metamask for crypto transactions.

OBJECTIVE: to learn about cryptocurrencies and learn how transaction done by using different digital currency.

Prerequisite: 1. Basic knowledge of cryptocurrency
2. Basic knowledge of distributed computing concept
3. Working of blockchain.

THEORY:-

Introduction to cryptocurrency:-

- It is digital payment system that doesn't rely on banks to verify transactions. It's peer-to-peer system that can enable anyone anywhere to send and receive payments. Instead of physical money carried around & exchanged in the real world, cryptocurrency payments.
- Cryptocurrency is stored in digital wallets.

- The first cryptocurrency was Bitcoin, which was founded in 2009 & remain the best known today.

How does cryptocurrency work ?

- It run on distributed public ledger called blockchain, a record of all transactions updated and held by currency holders.
- Units of cryptocurrency are created through a process called mining, which involves using computer power to solve complicated mathematical problems that generate coins. Users can buy the currencies from brokers, than store and spend them using cryptographic wallets.
- If you own It, you don't own anything tangible key allows to move a record or unit of measure from 1 person to another without a 3rd party.
- Although Bitcoin has been around since 2009, cryptocurrencies are still emerging in financial terms, & more uses are expected in the future.

cryptocurrency example :-

1] Bitcoin :- It was 1st cryptocurrency & is still the most commonly traded., which was developed by Satoshi Nakamoto

2] Ethereum :- developed in 2015, It is blockchain platform with its own cryptocurrency called Ether or Ethereum.

3] Litecoin :- It is most similar to bitcoin but has moved more quickly to develop new innovations, including faster payment and processes to allow more transactions.

4] Ripple :- It is distributed ledger system that was founded in 2012. It can be used to track different kind of transactions, not just cryptocurrency.

Non-Bitcoin cryptocurrencies are collectively known as 'altcoins' to distinguish them from the original.

How to store cryptocurrency.

There are different wallet providers to choose from. The terms "hot wallet" and "cold wallet" are used:

1] Hot wallet :- It refers to crypto storage that uses online software to protect the private keys to your assets.

2] Cold wallet storage :- It rely on offline electronic devices to securely store your private keys.

CONCLUSION:

In this way we have explored concept cryptocurrency and learn how the transactions are done using digital currency.

NAME: PATHAK SEJAL SUDHIR
ROLL NO: BE 46
SUBJECT: LP III GROUP C
Blockchain Technology
ASSIGNMENT NO:- 03

TITLE :- Write a smart contract on a test network, For Bank account of a customer For following operations:

- Deposit money
- Withdraw money
- Show balance

OBJECTIVE:- To learn new technology such as metamask. Its application and implementations.

Prerequisite :- 1. Basic knowledge of cryptocurrency
2. Basic knowledge of distributed computing concept
3. Working of blockchain.

THEORY:-

The contract will allow deposits From any account, and can be trusted to allow withdrawl only by accounts that have sufficient funds to cover the requested withdrawl.

That post demonstrated how to restrict ether withdrawals to an "owner's" account. It did this by persistenly storing the owner account's address, and then comparing it to the msg.sender value for withdrawl attempt. Here is simplified version of smart contract, which allows anybody to deposit money but only owner allow to withdrawals:

```
Pragma solidity ^0.4.19;
contract TipJar {
        address owner;
    function TipJar() public {
        owner = msg.sender;
}
    function withdraw() public {
        require(owner == msg.sender);
        msg.sender.tranfer(address(this).
                            balance);
}
    function deposit(uint256 amount)
                    public payable {
        require(msg.value == amount);
}
    function getBalance() public view returns
                    (uint256) {
        return address(this).balance;
    }
}
```

To generalize this contract keep track of ether deposits based on the amount address of the depositor, & then only allow that same account to make withdrawals of that ether.
To do this we need to keep track on the balance after each transactions.

Code to accept deposits and track balances:

```
pragma solidity ^0.4.19;
contract Bank {
    mapping (address => uint256) public balanceOf;
    function deposit (uint256 amount) public payable {
        require (msg.value == amount );
        balanceOf [msg.sender] += amount
    }
}
```

- mapping (address => uint256) public balanceOf; declares a persistent public variable, balanceOf that is a mapping from account addresses to 256 bit unsigned integers. Those int will represent the current balance of ether stored by the contract on behalf of corresponding address.
- mapping can be indexed just like array/ list/ dictionaries / tables in most modern programming languages.
- The value of missing value is 0. Therefore we can trust that the beginning balance for all account

addresses will effectively be zero prior to the first deposit.

Now add the withdrawl Function

```
function withdraw (uint 256 amount) public {
    require (amount <= balanceOf [msg.sender]);
    balanceOf [msg.sender] -= amount;
    msg.sender.transfer (amount);
    }
```

- The require (amount <= balances [msg.sender]) checks to make sure the sender has sufficient funds to cover the requested withdrawl. If not, then the transaction aborts without making any state changes or ether transfers
- The balanceOf mapping must be updated to reflect the lowered residual amount after the withdrawl.
- The funds must be sent to the sender requesting the withdrawl.

Withdrawal() function is very important to adjust BalanceOf [msg.sender] before transferring ether to avoid an exploitable vulnerability. The reason is to specific to smart contracts and the fact that a transfer to a smart contract executes code in that smart contract.

Suppose that msg.sender was malicious smart contract. Upon receiving the transfer - handled by msg. sender's fallback function - that malicious contract could initiate another withdrawl. When banking contract handles second withdrawl request, and allow the second withdrawl.

This vulnerability is called "reentrancy" bug because it happens when a smart contract invokes code in a different smart contract that then calls back into the original, thereby reentering the exploitable contract. For this reason, it's essential to always make sure a contract's internal state is fully updated before it potentially invokes code in another smart contract.

CONCLUSION:-

In this way we have explored concept of Smart Contract on the test network, for bank account.

TSSM
Page No. :

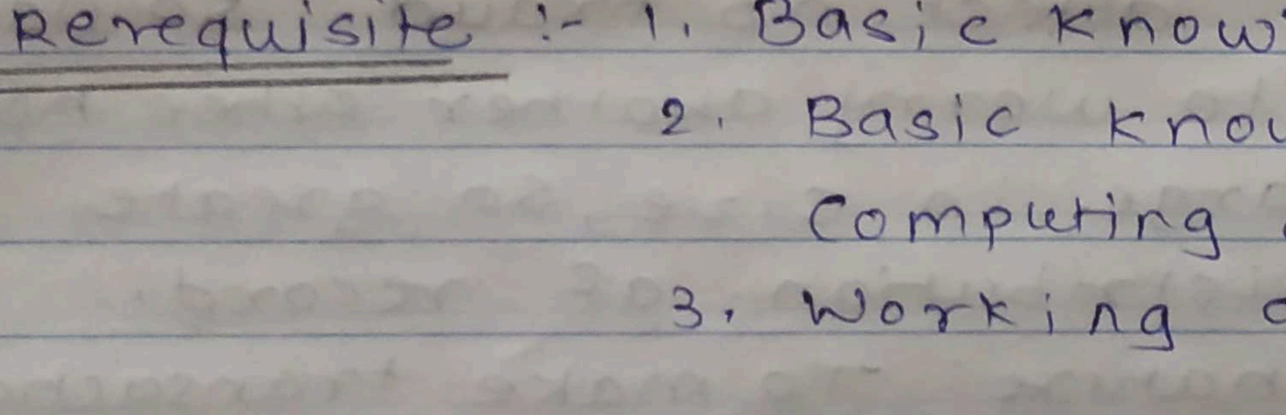
NAME : PATHAK SEJAL SUDHIR
ROLL NO : BE 46
SUBJECT : GROUP C LAB 3
Blockchain Technology
ASSIGNMENT NO :- 04

TITLE :- Write a survey report on types of Block chains and its real time usecase

OBJECTIVE :- To learn new technology such as MetaMask. Its application and Implementation.

PREREQUISITE :- 1. Basic knowledge of cryptocurrency
2. Basic knowledge of distributed computing concept
3. Working of blockchain.

THEORY :- Consortium Blockchain.



A] PUBLIC BLOCKCHAIN :-

These blockchain are completely open to following idea of decentralization. They don't have any restriction, open to the public which means not owned by anyone
Anyone having internet & computer can participate in public blockchain
All computer in the network hold the copy of other nodes or block present in the nerwork
Advantages :-

1) Trustable : There are algorithms to detect no fraud. so user need not to worry abother other node.
2) Secure : large in size, so greate distribution of record.
3) Anonymous nature : To make transaction no need to reveal your name and identity in order to participate.
4) Decentralized :- No single platform that maintain the network.

Disadvantage :-
1) Processing :- process is very slow, due to large size.
2) Energy consumption :- high energy consumption
3) Acceptance :- No authority is there so govt. facing issue to implement it fast

Example :- Bitcoin, Ethereum.

B] PRIVATE BLOCKCHAIN:-

These are not as decentralized as public blockch
only selected nodes can participate. These
blockchains are operated in closed network
few people are allowed to participate in
network within company.

Advantages :-

1) Speed - rate of transaction is high, due to sma
size
2) Scalability - we can modify scalability
3) Privacy - increased level of privacy for
confidentiality.
4) Balanced - more balance as few participant

Disadvantages:-

1) security :- these are more vulnerable
2) Centralized - Trust building is disadvantage
due to central nature
3) Count - few nodes go offline whole system
can be endangered.
4) Use cases - with proper maintainance,
great asset to secure informatio
without exposing to public eye.
Example - Hyperledger, Corda.

HYBRID BLOCKCHAIN:-

It is mixed content of private & public blockchai
where some part is controlled by some

Organization and other makes are made visible as a public blockchain.
Premission-based & permissionless systems are used. User access information via smart contracts.

Advantages :-

1) ecosystem :- It can't hacked as 51% of users dont have access
2) Cost :- trasactions are cheap as only few nodes verify the transaction.
3) Architecture :- highly customizable.
4) operations :- can choose participants in the the blockchain & decide which transaction can be made public

Disadvantages :-

1) Efficiency : Not everyone implement it.
2) Transparency :- possibility that user can hide information.
3) Ecosystem :- due to closed ecosystem, this blockchain lack incentives for participation.

Usecase :- provide greater solution to the health case industry, government.

Example :- Ripple network, XRP token.

D) CONSORTIUM BLOCKCHAIN :-

It is a creative approach that solves the needs of organization. This validates the transaction and also initiates or receives transactions. Some part is public & some part is private. more than one organization manage the blockchain

Advantages :-

1) Speed - A limited no of users makes it fast
2) Authority - Decentralized authority, make it more secure
3) Privacy - information of checked block is unknown to public view
4) Flexible - There is much divergence in the flexibility of blockchain.

Disadvantages :-

1) Approval :- All member approves protocol make it less flexible
2) Transparency :- It can hacked if organization become corrupt.
3) vulnerability :- high chances if few node are getting compromised

⇒

TSSM
Page No. :

Usecase :- high potential in business, ~~bau~~ bank, payment processors

Example :- Tendermint, multichain

CONCLUSION :-

In this way we have explored types of blockchain and its applications in real time.

NAME: PATHAK SEJAL SUDHIR
ROLL NO: BE46
SUBJECT: LP III GROUP C
Blockchain Technology
ASSIGNMEN NO:- 05

TITLE:- Write a program to create a Busine Network using Hyperledger.

OBJECTIVE:- To learn hyperledger, its application & implimentation

prerequisite:- 1. Basic knowledge of cryptocurrenc
2. Basic knowledge of disinbuted computing concept
3. working of blockchain

THEORY.

Hyperledger composer is an extensive, open development toolset and framework to make developing blockchain applications easier.

- You can use composer to rapidly develop use case and deploy a blockchain solution in days
- composer allows you to model your business network and integrate existing systems & data with your blockchain applications.
- Hyperledger composer supports existing Hyperledger Fabric blockchain infrastructure & runtime

- It generate business network archive (bna) file which you can deploy on existing Hyperledger Fabric network.

Key concepts of Hyperledger composer.

1. Blockchain state storage: It store all transaction that happens in your hyperledger composer application in hyperledger fabric network.

2. Connection Profiles: Connection profiles to configuration JSON file which help composer to connect to Hyperledger fabric.

3. Assets: Assets are tangible or intangible goods, services, or property and are stored in registries. It can represent almost anything in a business network. Assets must have unique identifier, but other than that, they can contain whatever properties you define

4. Participants :- They are member of business network. They may own assets & submit transactions.

5] Identities and ID cards :- Participants are members of a business network. assosiated with identity. ID cards are combination of an identity a connection profile and metaddata.

6] Transactions :- They are mechanism by which participants interact with assers. It's processing logic you can define in JS and emit event for transaction.

7] Queries :- It is used to return data about blockchain world-state. They are sent by using the Hyperledger composer API

8] Events :- Events are defined in model file. Once events have been defined, they can be emitted by transaction processor functions to indicate to external systems that something of importance has happened to the ledger.

9] Access control :- Hyperledge is enterprise blockchain & access control is core feature of any business blockchain. Using Access control rules you can define who can do what in Business networks The access control language is rich enough to capture sophisticated conditions.

10] Historian registry :- The historian is a specialised registry which records successful transactions, including the participants and identities that submitted them. The historian stores transactions as HistorianRecord assets, which are defined in Hyperledger Composer system namespace.

CONCLUSION :-

In this way we have learnt about hyperledger and its use case in business world.