Summary

Various Ethereum applications, with a focus on tokens:

- Fungible Tokens & ICO's
- ERC20
- DAO's (Decentralised Autonomous Organisations)
- Nonfungible Tokens (NFT)
- Games
- Artworks

Reading: embedded links

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

Fungible Tokens

Coloured coins were an attempt to represent assets like shares, tickets, etc. on Bitcoin

Ethereum provides a platform for representing holdings of such assets that is

- more easily programmable
- more expressive with respect to enforcement of rules associated to the asset

Assignment Project Exam Help

Representation of **tokens** has proved to be the main use case of Ethereum so far

https://tutorcs.com

A type of assets is called **fungible** if one instance of the type is indistinguishable from any other instance of the same type: they have all the same properties as far as their use or value is concerned.

WeChat: cstutorcs

Examples: dollars, unreserved tickets, common shares in a company, commodities (steel, sugar, ….)

The typical representation pattern for tokens is a contract account that stores data about ownership, with functions for transferring ownership

ERC20

ERC20 is a contract account interface that captures typical token contract features
https://github.com/ethereum/EIPs/blob/master/EIPS/eip-20.md

Use of this standard
— aids user familiarity
— simplifies work of exchanges wanting to list tokens for trading

Assignment Project Exam Help

function totalSupply() public view returns (uint256)
— what is the total number of tokens issuable?

https://tutorcs.com

function balanceOf(address _owner) public view returns (uint256)
— how many tokens does _owner own?

WeChat: cstutorcs

function transfer(address _to, uint256 _value) public returns (bool)
— transfer _value tokens from msg.sender to _to, and
log event Transfer(msg.sender,_to,_value)
throw an error if the balance of msg.sender is too low

event Transfer(address _from, address _to, uint256 _value)

ERC20 continued: delegated transfer

In some applications, it is desirable to delegate rights to transfer tokens to another user or contract

Example:

An online game uses its own tokens as a virtual currency when playing the game
To transfer tokens, players would need to also own Ethereum
They avoid this by delegating rights to spend some of their tokens to the game platform
The game platform pays gas costs and charges users in tokens

Assignment Project Exam Help

The following functions in ERC20 support this:

https://tutorcs.com

event Approval(address _owner, address _spender, uint256 _value)

WeChat: cstutorcs

function approve(address _spender, uint256 _value) public returns (bool)

— allow _spender to transfer up to _value of my tokens
logs Approval(msg.sender, _spender, _value)

function transferFrom(address _from, address _to, _value) public returns (bool)

— transfer _value number of tokens from account _from to account _to

function allowance(address _owner, address _spender) public view returns (uint256)

— how much may _spender still transfer of _owner's tokens?

ERC20 Optional Methods

function name() public view returns (string) OPTIONAL

— name of the token

function symbol() public view returns (string) OPTIONAL

— short symbol, as would be used on an exchange, e.g. “BTC” for Bitcoin

function decimals() public view returns (uint8) OPTIONAL

— number of decimal places in user representation of a number of tokens, e.g. 2 for AUD

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Consensys ERC20 Implementation

From: https://github.com/ConsenSys/Tokens/blob/fdf687c69d998266a95f15216b1955a4965a0a6d/contracts/eip20/EIP20.sol

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

```
pragma solidity ^0.4.21;

import "./EIP20Interface.sol";

contract EIP20 is EIP20Interface {

    uint256 constant private MAX_UINT256 = 2**256 - 1;

    mapping (address => uint256) public balances;

    mapping (address => mapping (address => uint256)) public allowed;

    /*
    NOTE: The following variables are OPTIONAL vanities. One does not have to include them.
    */
    string public name;                   //fancy name: eg Simon Bucks

    uint8 public decimals;                //How many decimals to show.

    string public symbol;                 //An identifier: eg SBX
```

Constructor

```
function EIP20(
        uint256 _initialAmount,
        string _tokenName,
        uint8 _decimalUnits,
        string _tokenSymbol
    ) public {
        balances[msg.sender] = _initialAmount;              // Give the creator all initial tokens
        totalSupply = _initialAmount;                       // Update totalSupply (inherited from EIP20Interface)
        name = _tokenName;                                  // Set the name for display purposes
        decimals = _decimalUnits;                           // Amount of decimals for display purposes
        symbol = _tokenSymbol;                              // Set the symbol for display purposes
    }
```

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Transfer

```
function transfer(address _to, uint256 _value) public returns (bool success) {
        require(balances[msg.sender] >= _value);
        balances[msg.sender] -= _value;
        balances[_to] += _value;
        emit Transfer(msg.sender, _to, _value);
        return true;
    }
```

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

Exercise: This code is for a version of Solidity that “wrapped around” overflowing and underflowing arithmetic. Don’t we need to worry about overflow and underflow in this code? No! Why not?

TransferFrom

```
function transferFrom(address _from, address _to, uint256 _value) public returns (bool success) {
        uint256 allowance = allowed[_from][msg.sender];
        require(balances[_from] >= _value && allowance >= _value);
        balances[_to] += _value;
        balances[_from] -= _value;
        if (allowance < MAX_UINT256) {
            allowed[_from][msg.sender] -= _value;
        }
        emit Transfer(_from, _to, _value); //solhint-disable-line indent, no-unused-vars
        return true;
    }
```

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

```
    function approve(address _spender, uint256 _value) public returns (bool success) {
        allowed[msg.sender][_spender] = _value;
        emit Approval(msg.sender, _spender, _value);
        return true;
    }

    function balanceOf(address _owner) public view returns (uint256 balance) {
        return balances[_owner];
    }

    function allowance(address _owner, address _spender) public view returns (uint256 remaining) {
        return allowed[_owner][_spender];
    }
}
```

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

See “ERC20: an attack vector on approve/transferFrom methods”
https://docs.google.com/document/d/1YLPtQxZu1UAvO9cZ1O2RPXBbT0mooh4DYKjA_jp-RLM
for a subtlety relating to approve.

Initial Coin Offerings

Ethereum Tokens have been a popular approach to fundraising for application projects working on Ethereum:

- project creates an Ethereum token contract
- investors pay Ether into the contract, receiving tokens in exchange
- project spends the Ether on project development costs
- tokens are later useable in the application developed, or represent other forms of rights (e.g. rights to project profits, voting rights)

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

This fundraising approach is called “Initial Coin Offerings” and was particularly popular 2016-2018 (more later)

The majority of ICO’s have used a form of the ERC20 token standard

Some Prominent ICO's

NAME	PURPOSE	AMOUNT RAISED $US	CLOSING DATE
EOS	Blockchain Platform	4.1 B	Jun 2018
Telegram	Encrypted Messaging App & Blockchain	1.7 B	Feb 2018
Dragon	Currency for Casinos	320 M	Mar 2018
Filecoin	Decentralised Storage	257 M	Oct 2017
Tezos	Currency	232 M	Jul 2017
Bancor	Prediction Market	153 M	Dec 2017
TheDAO	Decentralised Venture Fund	152 M	May 2017

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Some Australian ICO's

Australian ICO's have included:

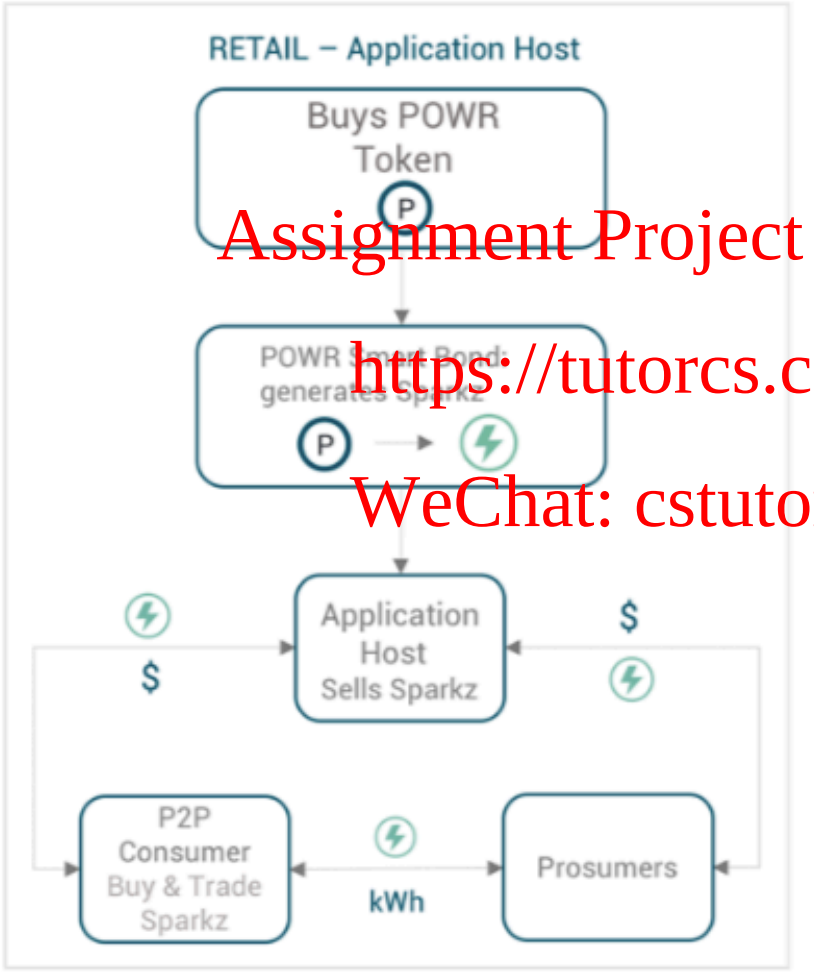
NAME	PURPOSE	AMOUNT RAISED $A	CLOSING DATE
Havven	StableCoin	38.6 M	Jun 2018
PowerLedger	Energy Trading Platform	34 M	Feb 2018
CanYa	Service Marketplace	12 M	Dec 2017
Shping	Consumer Provenance Information	8.5 M	Mar 2018
Chronobank	Labor hire platform	6.8 M	Jul 2017
Intimate	Adult Industry Platform	5.5 M	Jun 2018
Blockgrain	Agricultural Supply Chain Tracking	3.5 M	Apr 2018
Blockbid	Crypto Exchange	3.2 M	Dec 2017
Horizon State	Voting Platform	1.4 M	Oct 2017

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Utility Token Example: PowerLedger

From PowerLedger White Paper: https://www.powerledger.io

Figure 4.1.1: The retail model for working with existing market structures



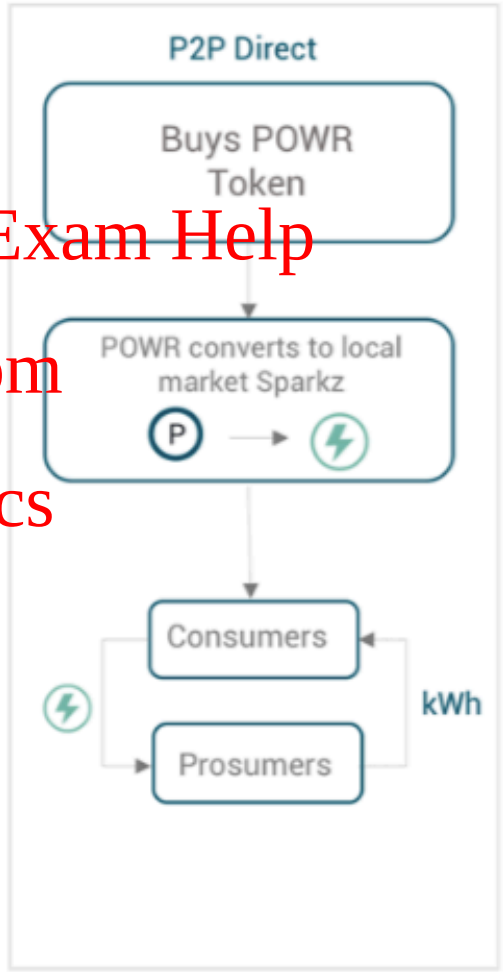


Figure 4.1.2: The direct peer-to-peer model for working within deregulated market structures

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Decentralised Autonomous Organisations

Proposed in Dan Larimer, The Hidden Costs of Bitcoin (2013)
https://letstalkbitcoin.com/is-bitcoin-overpaying-for-false-security

> "Think of a crypto-currency as shares in a Decentralized Autonomous Corporation (DAC) where the source code defines the bylaws. The goal of the DAC is to earn a profit for the shareholders by performing valuable services for the free market. With this goal in mind set out to maximize shareholder value at every stage as you design the bylaws that govern operation of the DAC.

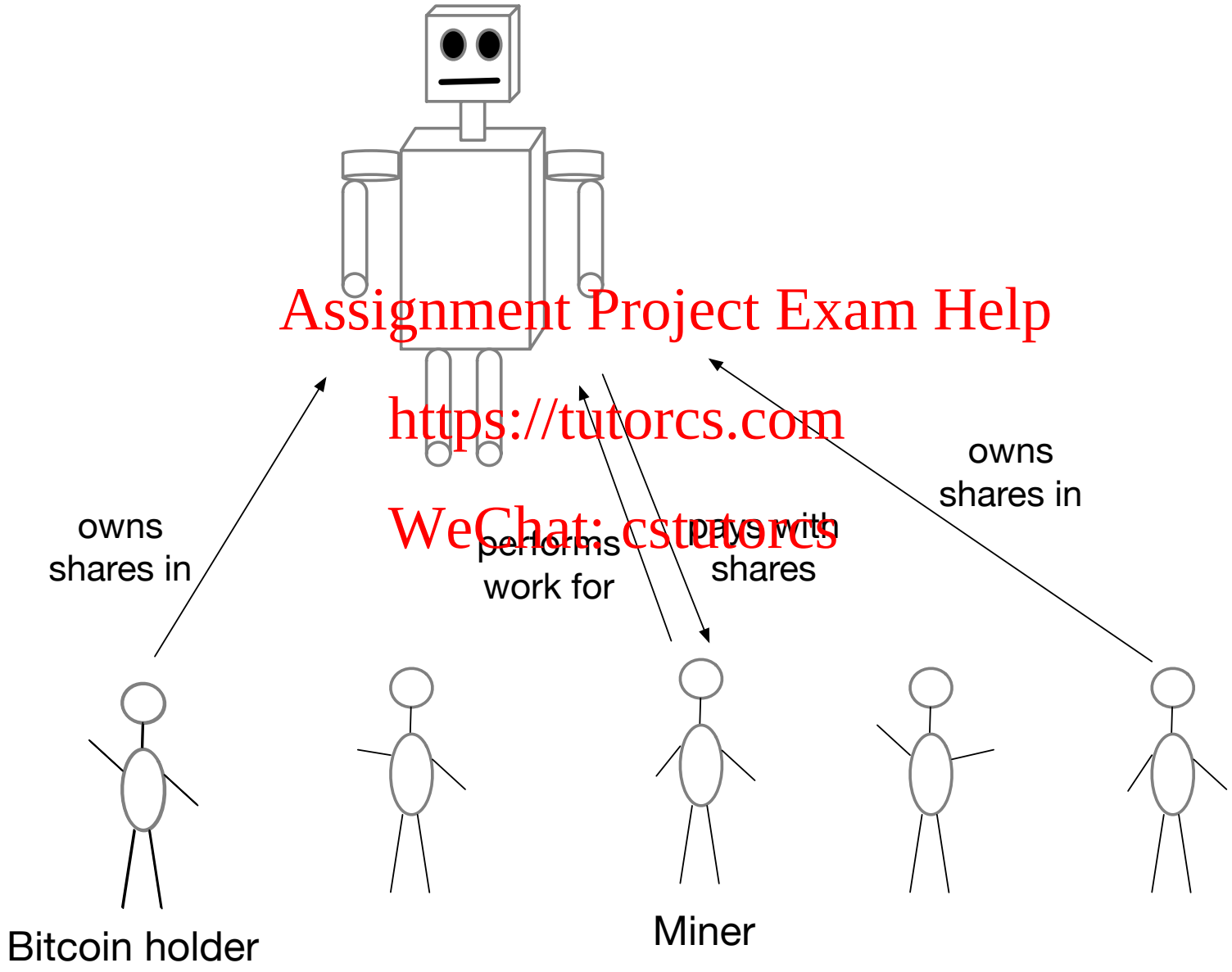
Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

> "The DAC only has one way that it can acquire the services it requires to operate and that is to pay for them with shares in the decentralized company. One service that is required is transaction validation, another is security against double-spend attacks by private (for-profit) criminals. Another service that is required is a viral marketing campaign. Other services include but are not limited to privacy for the customers and defense against traffic filtering."

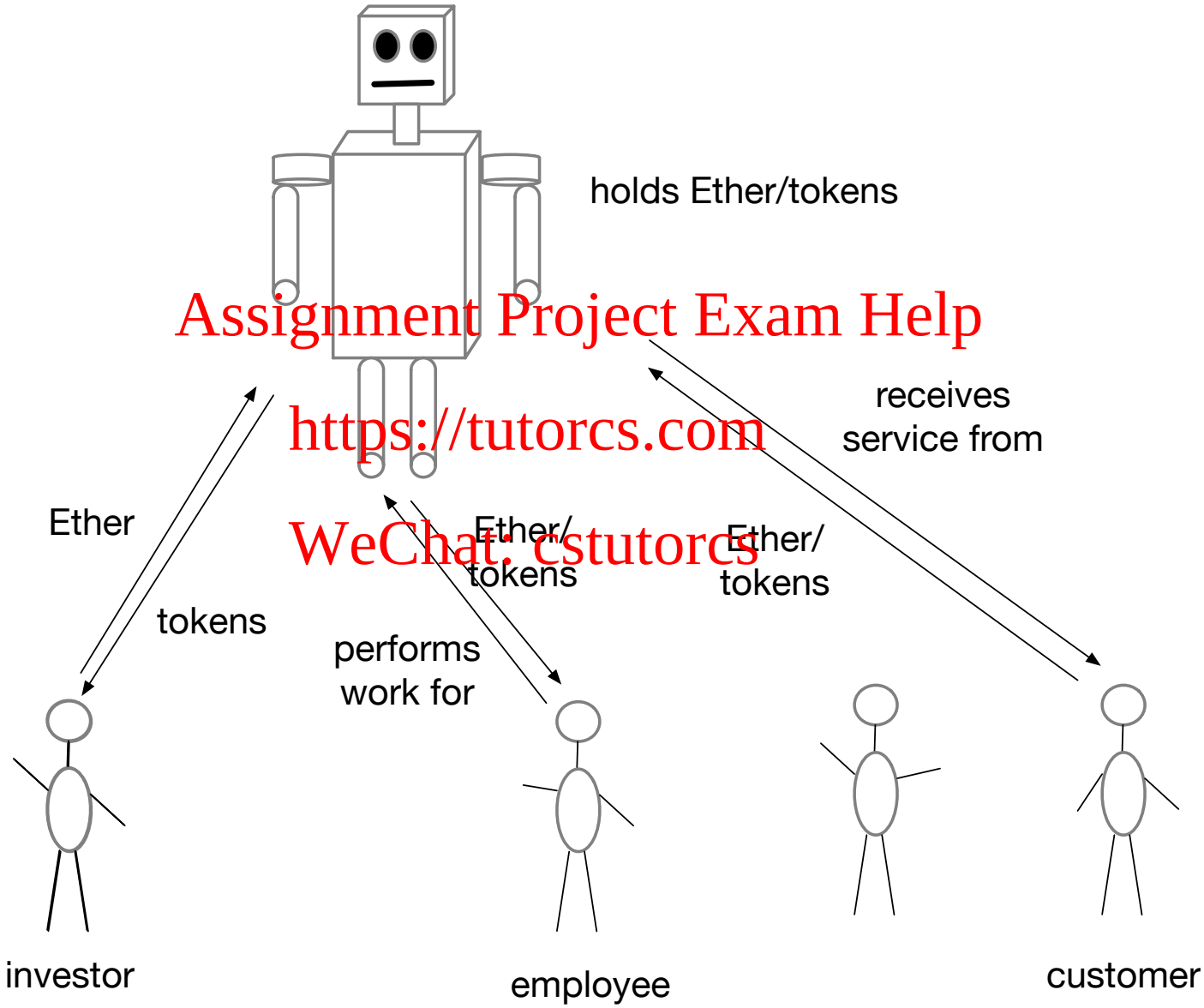
Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs



DAO's as Ethereum Smart Contracts



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

TheDAO - A Decentralised Venture Fund

TheDAO was a decentralised autonomous organisation
https://download.slock.it/public/DAO/WhitePaper.pdf

- An Ethereum smart contract, launched May 2016
- Funded by an ICO, that raised 11.5 M Ether = c. $A 200M
- Investors paid Ether into the contract, received TheDAO tokens in exchange
- Ether funds in the contract intended for financing Ethereum development projects
- Projects could submit a proposal (plus deposit, to prevent spam) to the contract
- Token holders vote via the TheDAO contract on allocation of TheDAO funds to projects
  (quorum of votes required, proposer deposit lost if not met)
- Governing committee including prominent names such as Vitalik Buterin, with white-list powers only
- Projects to return profit shares to TheDAO as per proposal
- To prevent “tyranny of the majority” objectors to a proposal may “split” the DAO into two (original plus a new child DAO) and move their share of funds to the child DAO
- Earlier similar idea: BitShares (https://bitshares.org)

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Questions concerning the legal status of TheDAO

The overall scheme raises multiple questions concerning its legal status:

- TheDAO has no legal jurisdiction where disputes can be resolved
- Investors were asked to invest on the basis of a white paper describing the proposal + code, which has precedence in case they disagree?
- Proposers of TheDAO state "The code is the law". Is it?
- Is a share in TheDAO a "security" such as a company share?
- Did the sale of TheDAO tokens break any country's securities regulation?
- If someone wants to sue a DAO, who do they sue?
- Companies are usually "limited liability", protecting shareholders from being sued for company losses — are TheDAO shareholders protected?

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

ERC 721 Non-Fungible Tokens

In ERC 721, a specific asset of some type is represented by a number (uint256)

```
    function balanceOf(address _owner) external view returns (uint256);
       // number of tokens owned by an address

    function ownerOf(uint256 _tokenId) external view returns (address);
       // who owns a particular token?

    function safeTransferFrom(address _from, address _to, uint256 _tokenId, bytes data) external payable;
       // transfer a token, call the recipient address _to with onERC721Received(data) if it is a contract

    function safeTransferFrom(address _from, address _to, uint256 _tokenId) external payable;
       // transfer a token

    function transferFrom(address _from, address _to, uint256 _tokenId) external payable;
       // transfer a token, caller needs to check that _to can receive the token

    function approve(address _approved, uint256 _tokenId) external payable;
       // give a (unique) address the rights to transfer a token, _approved = 0 means no address is approved

    function setApprovalForAll(address _operator, bool _approved) external;
       // give/revoke operator the right to transfer ALL tokens belonging to the sender

    function getApproved(uint256 _tokenId) external view returns (address);
       // get the address approved to transfer a token

    function isApprovedForAll(address _owner, address _operator) external view returns (bool);
       // is the operator approved to transfer all of an owners tokens?
```

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Images: cryptokitties.co

Cryptokitties

Because everyone knows that the real purpose of the internet is … cute cats ….

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

Cryptokitties (https://www.cryptokitties.co) is a game that

- enables users to own digital cat images (ownership via an Ethereum smart contract)
- breed their cats to create new kitties (based on a secret genetic algorithm)
- sell their kitties to other users

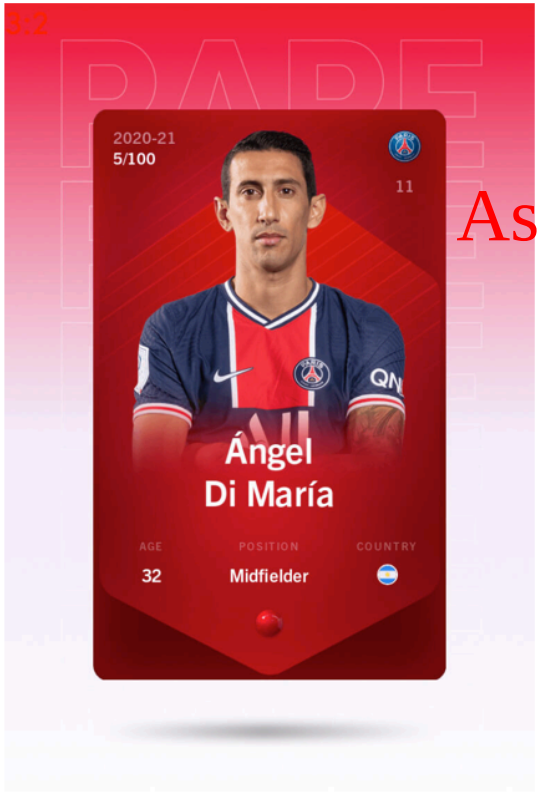
Popularity of this app caused major congestion on the Ethereum network in Dec 2017 (up to 25% of network traffic), causing ICO's to extend their fundraising periods!

https://qz.com/1145833/cryptokitties-is-causing-ethereum-network-congestion/

Sorare

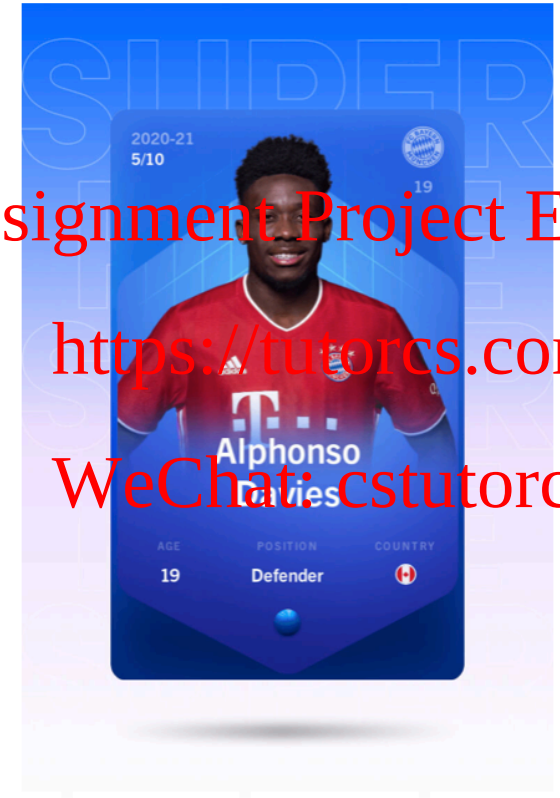
Collect & Trade

limited edition digital cards



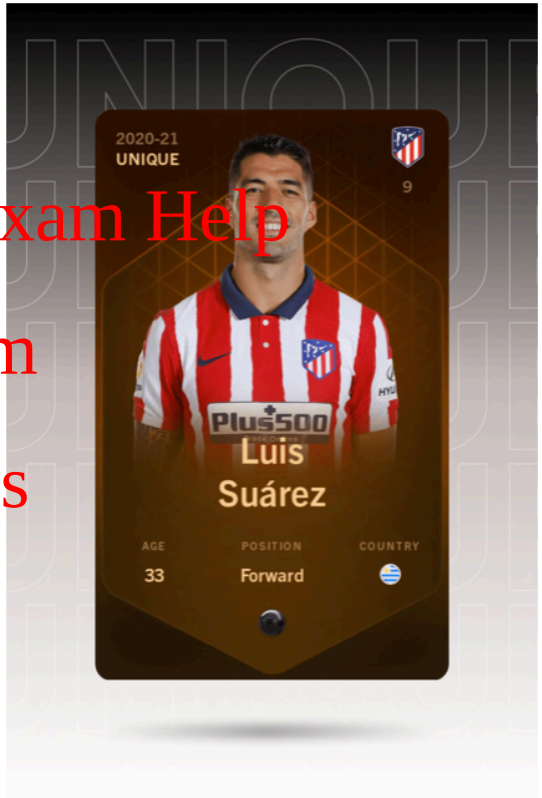
Rare

100 per season



Super Rare

10 per season



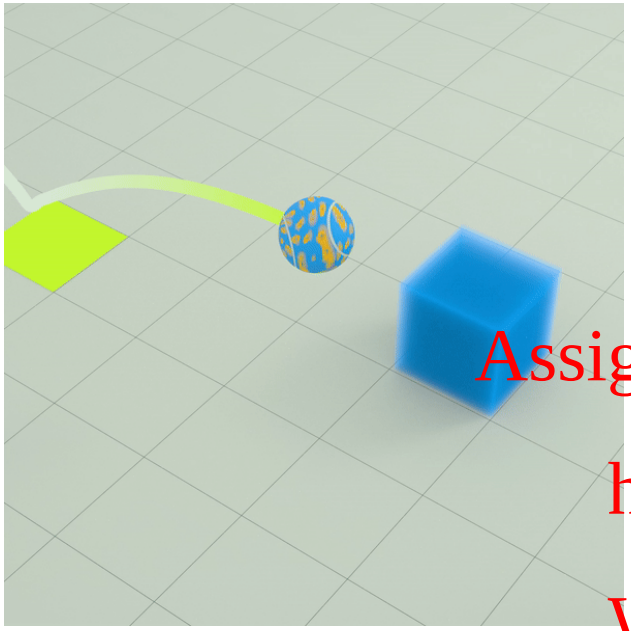
Unique

1 per season

Image: sorare.com

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

NFT’s as a marketing tool



Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

- Australian Open Tennis Art Ball Collection “fan club” membership
- Norwegian Cruise Line,
- Anheuser-Busch (beverages)
- Nike, Adidas (sportswear)
- Warner Bros (films)

Images: https://ao-2023.artball.io

Play-to-Earn / Pay-to-Play-to-Earn

Assignment Project Exam Help
https://tutorcs.com
WeChat: cstutorcs

NFT represented, salable in-game assets that can be earned from game play

In game economy, game-play as a job popular in the Philippines

Prominent hack of side-chain with theft of 175k Ether and 25.5 million USDC in March 2022

NFTs for Digital Art

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

https://onlineonly.christies.com/s/first-open-beeple/beeple-b-1981-1/112924
EVERYDAYS: THE FIRST 5000 DAYS, Beeple
Sold $US 69m, Mar 12, 2021

Nyan cat GIF, Chris Torres,
Sold 300 ETH = $US 560,000, Feb 20, 2021

Artist benefits:

- royalty collection
- share of resale value

Summary

Various Ethereum applications, with a focus on tokens:

- Fungible Tokens
- ERC20
- ICO's (Initial Coin Offerings)
- DAO's (Decentralised Autonomous Organisations)
- Nonfungible Tokens (NFT)
- ERC 721
- Games
- Artworks

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs