**SingularityNET(AGI)**

De

Ever wonder what will happen if there is a creation of an open marketplace for Artificial intelligence where everything can learn from itself

SingularityNET is building the world's first decentralised AI as a Service Platform

the world 1st Decentralised market place for AI
its a new Ecosystem for AI agents to exchange data, create service and collaborate between themselves
any potential customer can simply browse the singularityNET catalog of AI service or they can request their specific needs to the SingularityNET

SingularityNET is a combination of

- OpenSource Codes
- Block Chain – more geared toward financial applications
- Smart Contracts
- AGI Token – Specially designed cryptocurrency

what is Smart Contracts

SingularityNET (AGI) lets anyone create, share, and monetize AI services at scale. AI is projected to create **16 trillion** USD of global GDP value by the year 2030, meaning SingularityNET, at its current valuation, has massive upside potential.

**The Goal of SingularityNET is to address 4 main challenges of AI that exist currently**

1. There are many AI solutions or systems that do specialised tasks vey well, but there is No Artificial General Intelligence or AGI.
   So SingulairtyNET wants to develop AGI
2. AI is currently quite fragmented or separated; each Specific system exists in its own world and there is not really a way for them to communicate and work together easily.
   So AI Service integration is another big challenges and SingularityNET wants to face and solve this challenge
3. AI systems are overwhelmingly built by and for huge tech companies or government or military applications. But many other potential AI applications that can benefit society could be developed, but they are not being realized(not getting awareness) due to the "Socioeconomic nature of the AI industry".
   But SingularityNET want to change this one, the AI systems developed does not need to be only the system that can solve the problem of companies or government, it has solve also the problem of those ordinary people.

4. Massive companies like IBM, Google and others build AI sytems for their own benefit, means after all that companies will get a money from this systems. So the system they built is controlled by them or Centrally.

But the SingularityNET want to change this one, that AI Systems did not to be centrally controlled by that companies only. It has to be Decentralized.
Also by the developed AI systems only that companies will be richer and richer but in SingularityNET everyone who collaboretes in that Service will get a payment through a Block chain smart contracts and payment is shared to all just according to their work.
So by SingularityNET ordinary people and smaller businesses can no benefits rather than just the single company only.

SingularityNET recently announced the launch of the mainnet Beta Version of its platform on the **Ethereum Network**

**What is Ethereum Network ?**

- Its an open-source, public, blockchain based distributed computing platform and operating system featuring smart contract(scripting) functionality
- Ethereum provides a decentralized virtual machine
- like Bitcoin, Ethereum is a distributed public block chain, but the difference is bitcoin is nothing just currency but ethereum is far more than that

**SingularityNET**
AI + blockchain in pursuit of maximizing NET benefits for participants and all Life

1. Blockchain for smarter, safer AI
2. AI powered smart transactions
3. AI marketplace
4. Block chain data protection
5. Token sale

What is a difference between Opencog and SingularityNET

**SingularityNET**

1 Overview
2 Concepts and Components
3 setup a singularityNET

**Overview**
its an open source and decentralized network of AI services made accessible through blockchain
developers publish their services to the SNET network, where they can be used by anyone with an internet connection
developers able to charge for the use of their services using the native AGI token
these services can provide different domains such as
image/video, speech, text, time-series, bio-AI, network analysis, etc
developers can also deploy autonomous AI agents that interoperate with other services on the network

Knowledgeable people are realizing that the next few decades will see a transition to a new society and economy in which machine intelligence is the dominant factor
SingularityNET is not only a collection of AIs Service, but its a market.
The path to creating a positive “global brain” is challenging.
A technological singularity could have unprecedented benefits but also unprecedented risk.
The popular press is full of dire warnings about the dangers of artificial general intelligence

among the challenges of SingularityNET
- current set of protocols for collective action
- today’s financial mechanisms and instututions

**Blockchain**, with its natively digital money, is a powerful tool for managing **transactions** in an economy dominated by machine intelligence.
However, Blockchain is just a tool; there are important decisions to be made about how to use it.
At its core, SingularityNET is a set of smart contract templates that AI agents can use
5. to request that AI work be done
6. to exchange data and
7. to supply the results of AI work

SingularityNET is designed both to be highly valuable now and to lay the groundwork for the emergence of a self-modifying, decentralized “artificial cognitive organism”.
SingularityNET provides an automated process that enables any businness to connect existing AI tools to build solution it needs.

Many state of the art AI tools exist only in Github repositoreis created by graduate students or independent researchers. The latest algorithims for image and video analysis, machine translation, automated theorem proving, bioinformatics data analysis, etc are typically avialable on Github, but the friction inherent in installing, configuring and running them limits their use.

Most AI developers are academics, not businness people, and have no easily accessible markteplace to turn to inorder to monetize their clever AI code. As a result, the AI in real-world products tends to lag months to years behind the code.

SingularityNET is a launchpad where developers can quickly deploy their AI models and algorithims into real world applications.

Machine learning tools also require datasets of sufficient size. Creating and managing such large datasets are beyond the means and capabilities of most AI developers, and the closed development model that currently prevails makes it hard for developers to share datasets.
SingularityNET connects these AI tools and datasets to the marketplace, making them accessible to end users and developers. Is a sharing-economy marketplace for AI, so its an open network.

Like Uber and Airbnb, we have identified a large unexploited resource and a large market in need of that resource, and we are launching a tool to connect the two.

The apartments in AirBnBs network do not combine to become meta-apartments, nor does Uber's network create meta-cars, but AIs in SingularityNETs network come together to form meta-AIs whose intelligence is more than the sums of their parts.

**Robust and Adaptive Software Architecture**
SingualrityNET is a distributed computing architecture for making new kinds of smart contracts to facilitate market interactions with AI and machine learning tools.
The following design principles are incorporated throught the design

- Interoperability → the network will be interface with multiple blockchain
- Data Sovereignty and Privacy → the network includes user-side controls for sharing personal data. Users remain in control of their data and can share it with the network via smart contracts.
- Modularity → flexible network capabilities make it possible to create custom topologies, AI agent collaborations and failure recovery methods
- Scalability → SNET will securely host both private and public contracts

in SingularityNET services are accessed with AGI token. Token holders can use their tokens to purchase services in the marketplace.

SingulairtyNET Agents can run in the

- Clouds
- on Phones
- Robots and
- Embedded devices

SNET platform contains a number of critical components that work together to enable a decentralized network of AI services.
The core components are designed to allow for a functional, scalable, and extensible system.

SNET developers arrive at current architecture through a careful process

- guided by a few key decisions governing blockchain integration
- AI service integration
- by the goal of building an AI marketplace that is both open & complaint with regulatory and legal requirements

- first, we made the conscious choice to minimize our dependence on our current blockchain and ethereum
- 2nd, on AI services integration, we wanted to abstract away as much of the network as possible, in order to reduce the learning curve and minimize the overhead(indirect cost or expense) associated with providing AI services via the network.
  → the abstraction of the network is achieved with a single flexible tool called **Daemon**, that will help us provide scalability, robustness, distribution, and management features to the entire community
- finally, to make our marketplace compliant with regulations without compromising on openness, we implemented it separately from our fully decentralized registry of AI services currently available on the blockchain.

The price for a single call of Service in SingularityNET is equal to 10^8AGI which means 1COG
AGI token
COG → cost of good sold

## Components of SingularityNET

the components of SNET platform divided into

- core components and
- auxiliary componets

**Core Components of SNET**

1. **SNET marketplace** → its a Dapp(Distributed App) and it provides a front-end for exploring AI services available on the network
   - Users can interact with and call them through a web interface, and rate them afterwards.

2. **Service** → its published to the SingularityNET network and provides a grpc based API for calling it.
   - the service API specification, the IP address where to find the service and pricing information is published to IPFS as **Service Metadata**
   - People need to follow the SNET **Naming Standards** guidelines
   - **IPFS**
     - Interplanetary File System
     - its a versioned file system which can store files and track versions over time, very much likRopstene, Git
     - it also defines how files move across a network, making it a distributed file system, much like BitTorrent.
     - IPFS and blockchains can work well together, because of their similarity

3. **Software**
   - **SNET-CLI** (SNET Command Line Interface)
     - it help us to interact with the SNET platform
     - it allows crucial tasks such as
       - calling and querying services
       - registering and managing identities
       - publishing services
       - updating registration information
       - notifying the platform of new endpoints
       - managing payment channels and balances

- **SDK**
  - the SNET SDK helps you to integrate SNET Services with your own software

- **SNET Daemon**
  - a developer exposes their service to the network by running the SNET Daemon alongside their service
  - it interacts with the blockchain to facilitate authorization(heyyama) and payment for services
  - it acts as a passthrough for making API calls to the Service
  - it isolates the payment and blockchain interaction so a developer can focus on deploying and improving their service.
    - Daemon API
    - Daemon Channel Storage

**4. Blockchain Contracts**

- **AGI Token**
  - its a Utility token to be exchanged for AI service
  - its an ERC20 token hosted on Ethereum

  what is a difference between Coin, Token and Cryptocurrency?

  - Many people erroneously use these words interchangeably, but they are different from each other
  - Cryptocurrencies are digital or virtual currencies that are encrypted(secured) using cryptography. Cryptography refers to the use of encryption techniques to secure and verify the transfer of transactions.
    - Its not controlled by any central authority, the decentralization nature of the blockchain gives the meaning to cryptocurrencies.
  - Cryptocurrencies is divided into two categories **tokens** and **coins**
    - Coin is a form of digital money created through encryption techniques which store values. They may be sent, received or mined, but donot perform any functions beyond acting as money. There are different types of coins like Bitcoin, Litecoin, NameCoin

    - Tokens are digital assets, issued by a blockchain based project, which is used as a payment method inside of its ecosystem, performing similar functions with coins. But it also gives the holder a right to participate in the network means to be a company's share.
      e.g you can think a token as a concert ticket. Only the one who has the ticket can participate in the concert, even if you have a money but if you do not have ticket you cant participate on the concert. And this ticket has valuable on this concert only but your money is independent and has valuable everywhere.
      Like this the token is specific to a given project and valuable inside that project only
      → there are two types of tokens
      - **Security tokens** → designed to be the company's share
      - **Utility tokens** → have certain use cases inside the project

      → Creating a token is much easier than creating a coin, as you don't have to create new code or modify already existing ones. All you have to do is just using a standard template from platforms like **Ethereum**, which is a block chain based and allow anyone to create tokens in just a few steps.

- **Registry**
  - registry means a place where registers or records are kept
  - services are published to a publicly accessible central registry on the blockchain
  - the registry maintains a list of active services on the network and where to find a services corresponding metadata
  - the registry has support for grouping services by organisation or team, along with access control for organisation members

- **Escrow**
  - escrow is a way of transfering property from the seller and money from the buyer through use of a third party which is neutral.
  - The escrow contract on the blockchain holds AGI funds in escrow during interaction between an end-user and a service. An end-user places funds in escrow before a service can be called, and remain there until the service has been delivered or the escrow funds timeout.
    - MPE – Multi Party Escrow
    - MPE stateless Client

**5. The Request for AI Portal(RFAI)**

  - lets Users make requests for AI services that they would like to see built and deployed onto the SingularityNET network and give AGI tokens as a reward for the developer who develop that service.
  - The RFAI portal fosters(encourage) the community by enabling Users to incentivize(motivate) developers to publish services

## Setup SNET

During installation of SNET-CLI the ff are also installed together

- mnemonic==0.18
- eth-hash
- pymultihash
- ipfsapi
- grpcio-health-checking
- argcomplete
- pillow
- web3
- trezor
- rlp
- rfc3986

- protobuf
- hidapi
- jsonr
- pcclient
- future
- pyyaml
- pycoin
- hexbytes
- cytoolz
- websockets
- eth-typing
-

## Service Integration Guidelines

Guidelines that help developers to write/integrate new AI services to the SingularityNET platform

- Supported languages
  - SNET services use gRPC to connect services together or to call a required service in other service e.g french to english translation service can call Text-to-Speech Service through gRPC
  - gRPC is a modern open source high performance RPC framework developed by google
  - RPC(remote procedure call) is a protocol that one program use to request a service from a program located in another computer on a network.
  - So any new service must provide its API in gRPC
  - gRPC Support Several programming languages C++, Java, Python, Go, Ruby, C#, Node.js…
  -
- AI frameworks
  -
- Third party code and models
  -
- Service documentation
  -
- Contributing to existing projects

#### Cryptocurrency and Blockchain note

The purpose of cryptocurrency is to allow people to manage their funds in a anonymous and secure way, from any location, without relying on third parties.

On the blockchain, your digital assets are not controlled by any bank or government. You are the only one who has access to your funds and you can instantly transfer them to any other address on the blockchain without depending on authorizations, permissions, or limits.

Your public address and your private key are the only pieces of information you need to hold and manage your funds from anywhere in the world.

Creating wallet https://www.myetherwallet.com

watched videos
https://www.youtube.com/watch?v=PC2sAJyyUTs&source=post_page----------

https://www.youtube.com/watch?v=S7Rtk_k4Hs0

**Questions to ask**

1. In Components of the SNET I understand the document or written tutorials, but I couldn't understand the diagram that depicts the components of SNET
2. which docker image and Metamask I need to install to setup the SNET
3. before installing the docker image I want to read the SingualrityNET whitepaper 2.0 how is it?

4. in How to publish a SingularityNET service page I faced an error in Step 8, couldn't Publish a Service on SingularityNET

```
root@326b4895849a:/opt/singnet/example-service# snet service publish
$ORGANIZATION_ID $SERVICE_ID -y
# Calculating gas price. It might take ~60 seconds.
# gas_price = 1.969292 GWei
Error: {'code': -32000, 'message': 'gas required exceeds allowance or always failing
transaction'}
If you want to see full Traceback then run:
snet --print-traceback [parameters]
```

root@326b4895849a:/opt/singnet/example-service#

another problem here is the Snet account address created here and my metamask account address is not the same I dont know where it gets this account.
Check it with “snet account print” or “snet account balance”

5. in How to Write a SingularityNet Services in Python page, also I faced an error in Step 8, couldn’t create SNET identity. But even I create this identity in How to Publish just by following the tutoirals, but in this I couldn’t

it displays an error of Identity_type

error: the following arguments are required: IDENTITY_TYPE
usage: snet identity create [-h] [--mnemonic MNEMONIC]
[--private-key PRIVATE_KEY]
[--keystore-path KEYSTORE_PATH]
[--network NETWORK] [--wallet-index WALLET_INDEX]
IDENTITY_NAME IDENTITY_TYPE

6. in Deploying SingularityNEt locally, in step of Setting up snet command line interface
./scripts/blockchain: No such file or directory

7. I fully install all libraries and dependencies in how to deploy local version of SingularityNET but after finishing it I dont know how to start it and use it.

8. I install snet-sdk by two ways
1 just by pip3 install snet-sdk
but this puts in the library as snet.sdk and this is not imported

2 by cloning github repositiories of snet-sdk-python then follow the installing part at the end of readme.
But the development/test blockchain dependencies could not be installed
./scripts/blockchain install raise an error of
FileNotFoundError: [Errno 2] No such file or directory: '/home/amante/snet/snet-sdk-python/snet_sdk/resources/contracts'

but the package is installed correctly
pip install -e .

and in python environment this mention snet-sdk the folder which exctracted from github.
And the problem is this folder does not have contracts files in snet_sdk/resources folders which pointed in __init__.py line 136

9 for the config part I installed it with pip3 install config, but this config does not have private key and as I think here what we need is the config of snet, but I do not know how to import this config of snet.
So I just provide the private key in this python file.
There is also another thing that I change here, which is the code take your private key and from this private key it extracts your public key and adds “0x” before this public key to make it hex, but by default the extracted public key is hex, so it raise an error, so what I change is I just delete the code that add “0x” to the public key in

snet-sdk-python/snet_sdk/__init__.py or Snet Class line 101 and 115

10 by creating the python function call.py inside the github repositories of snet-sdk-python with the following code

```
from snet_sdk import Snet
snet =
Snet(private_key='4e51e1cc30aae65758d4eb9edc46dad7b34a8f25ffa941e1dcbb15a34935a488')

client = snet.client("snet", "network-analytics-nodeimportance")
```

10 I make the snet sdk generate-client-library python snet network-analytics-nodeimportance
in three postion,
– home directory
– snetcli environment and
– in snet-sdk-python directory

11 Even calling snet service through terminal is not working correctly
snet client call snet network-analytics-nodeimportance CentralNodes centralnodes.json -y

Error: Cannot find initialized channel for service with org_id=snet service_id=network-analytics-nodeimportance and signer=0xE2A22dC95Ed0D02eA3Ea79964Da10c1A4A2515da

SingularityNet installation
to install

TO create identity with private key
sudo snet identity create [identity-name] key –private-key  [private-key]

To open the virtual environments snet-cli was installed
cd snet/
source snetcli/bin/activate

to deactivate it →

to list .files or hidden files in ubuntu
ls -a
sudo nautilus .snet/

Questions
1. what is the difference between Opencog and SingularityNET
2. block chain is the one who stores your security information, but who controls the block chain ? Or how can we fully believe the block chain.

3. in block chain is it true that one no can see our key and no one knows our password except us
4. what is agas_price ? Because when I create it says It might take ~60sec to calculate the gas price and it calculate it as gas_price = 0.110464 Gwei
5. what is a protobuf file