Quantum Time Sequencing for the creation of Financial Modules using Rotational Kinematics and objective Logistic Mapping to function in an Non-Cooperative Game setting across Global Financial Markets at a Microeconomic and Macroeconomic presentation.

Written and Authored by
Dr. Jonathan Benjamin Azrayehu Gideon
Shrodan Samuel Ozhuman - Ozhumanill I

Expansion of Non-Cooperative Game Theory with respect to the attributes in Socio-Economics and Econometrics.

Table of Contents:

Our first step will be to finalize the funds needed for SGM Equity, which would be an **INTERNAL PRIVATE EQUITY FUND. WE WILL NOT BE SOLICITING INVESTMENTS FOR THE FUND. Only those who are members of SGM will provide the capital needed to finalize the fund, of which SGM would then directly make its investments.**

SGM Private Equity: Organizational and Fee Structures

General—what are alternative asset managers? Organization? Fees?

- Provide financial advisory services to clients
- Limited partners (LPs) commit capital to private equity players to earn better returns than indexed returns
  - **Primary LPs are corporate and public pension funds, ours would be an INTERNAL FUND that ONLY THE MEMBERS HAVE ACCESS TOWARDS. It would not be available to outside investors until we believe it is necessary to reach out to institutions, pension funds, etc.**

- Alternative asset management includes investment vehicles focused on private equity, real estate, hedge fund solutions, fund of funds, non-investment grade credit, and multi-asset class exposures outside of other funds' mandates
- some financial advisory services:
  - financial and strategic
  - restructuring and reorganizing
  - capital markets
  - fund placement
- pursue inorganic growth strategies to expand market and foray into different product categories
- on fees
  - "Even as fund sizes have mushroomed, managers have continued to extract a 1.5 percent to 2 percent annual management fee from investors on their committed capital while taking off 20 percent of investment gains as an incentive fee.

The payouts have transformed private equity's leading players into potent money engines. Back in the days when private equity giants were young, management fees went to keeping the lights on. Nowadays they yield a geyser of profit."[1]

- **WE INCORPORATE THE SAME STRUCTURE THAT BLACKSTONE USES FOR P/E.**

Blackstone

- Organizational structure
  - raises funds and investment commitments using a partnership structure
  - The general partners' strong network demonstrates the firm's investment managing capabilities
- On revenue model

o based on charging management and performance fees for managing the portfolios in its investment advisor firms
§ receives an annual management fee based on a certain percentage of a fund's capital commitments or invested capital during the investment period
§ Following the investment period, it receives fees based on the fair value of the total investments
o sets performance fees for generating higher returns
o The performance fee is usually 20% of the fund's net capital appreciation per year, subject to certain net-loss carry-forward provisions
o In cases where an advisor manages funds through investments in hedge funds and registered investment companies, incentive fees range between 0% and 15%
· Strategy
o focuses on a variety of strategies such as:
§ leveraged buyout acquisitions of US companies
§ **equity or start-up investments (SGM Equity focus)**
§ **minority stakes (SGM Equity focus)**
§ **joint ventures (SGM Equity focus)**
§ distressed debt
§ structured debt
§ industry consolidations
o deployed its capital across sectors in and outside of the US
o **PRIMARILY FOCUSED ON ENERGY, REAL ESTATE, HEAVY INDUSTRIALS, AND TECHNOLOGY.**
· Has always been well positioned to contend for new capital and reinvestments
· Scale allows the company to innovate, understand the needs of its limited partners, and provide the innovative fund options that its competitors take longer to work out

Emulating this structure at SGM Equity (INTERNAL P/E FUND)

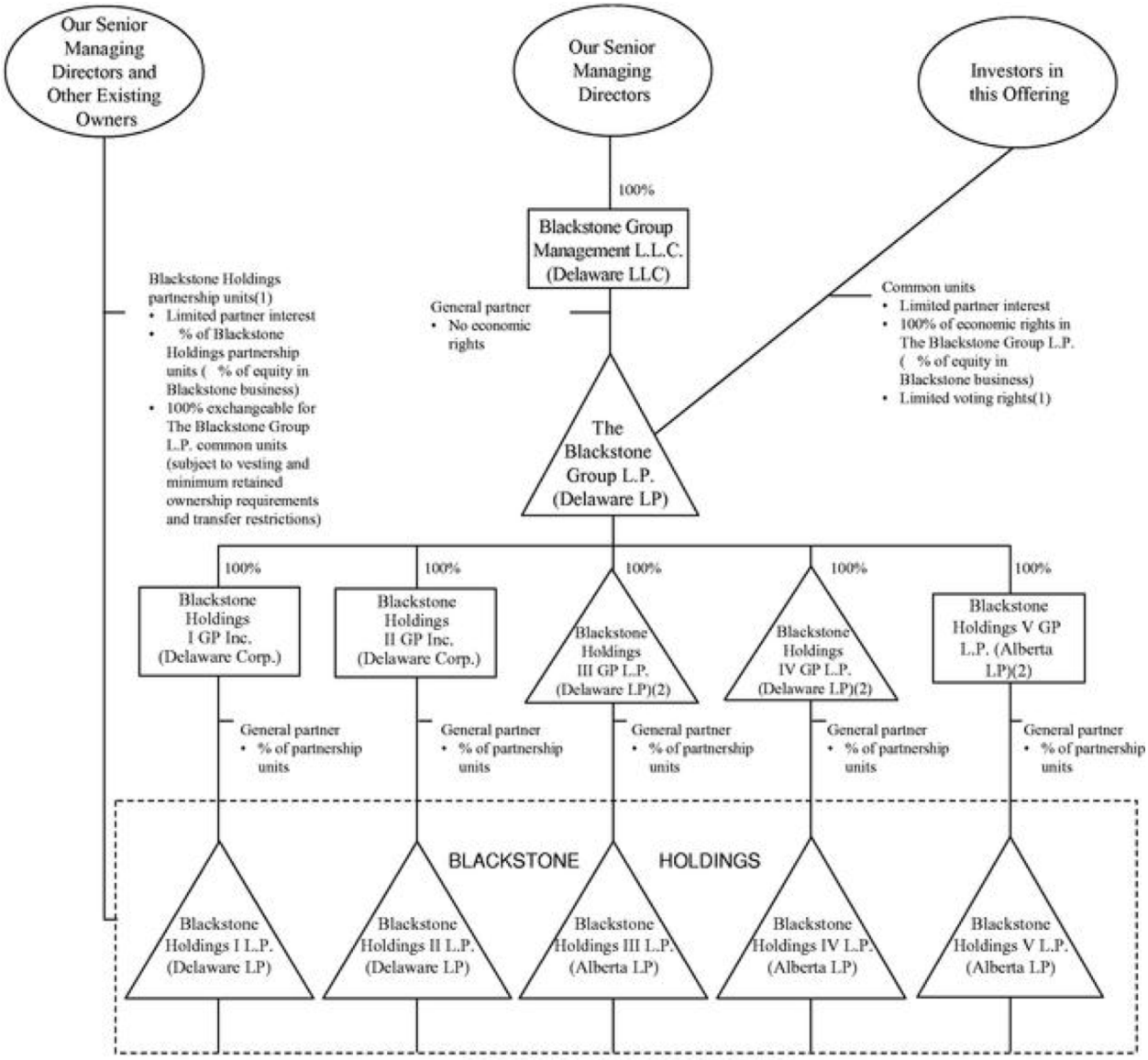
· Initial focus
- *Equity or Start-up investments*
- *Joint ventures*
- *Minority Stakes*

· **Will raise capital through the membership offering of SGM structure**
o Limited Partners: provide us with committed capital **(SGM would own 100% of fund)**
o General Partners: leverage connections and knowledge to produce returns
· rates **(Investors in SGM)**
o **management fee: 2%; performance fee: 20%**

Goal: generate returns in excess of benchmark indices that can be reinvested into the growth of the firm

---

[1] https://www.bloomberg.com/news/features/2017-10-26/private-equity-s-biggest-backers-are-tired-of-the-fees



(1) The Blackstone Group L.P. common unitholders will have only limited voting rights and will have no right to elect our general partner or its directors. The limited partners of Blackstone Holdings (other than AIG) will hold special voting units in The Blackstone Group L.P. that will entitle them, on those few matters that may be submitted for a vote of The Blackstone Group L.P. common unitholders, to participate in the vote on the same basis as the common unitholders and provide them with a number of votes that is equal to the aggregate number of partnership units in Blackstone Holdings that they then hold. See "Material Provisions of The Blackstone Group L.P. Partnership Agreement."

(2) The Blackstone Group L.P. holds Blackstone Holdings III GP L.P., Blackstone Holdings IV GP L.P. and Blackstone Holdings V GP L.P. through wholly-owned subsidiaries organized as Delaware limited partnerships and Delaware limited liability companies.

Value Proposition

SGMX MARKET CHARTER

Public Market Health	Global Market Health	Corporate Financial Health	Public Financial Health
Human Life Expectancy: From the usage of the ISM network, to what degree have we promoted longevity of human life? Via the interaction of its infrastructure and accessibility.	**Environmental Impact:** The current health of the environment and how SGMX has invested directly with infrastructure to counter man-made causations damaging our planet.	**Fair Market Creation:** Companies working together to create collaborative market conditions grown on SGMX changing FIAT market structures to benefit and adopt further.	**Gross National Income:** Comparative relationship of GNI outside of SGMX and growth of GNI for SGMX users within countries.
Human Capital: From usage of the ISM network, we study where the largest amount of transactional volume comes from the users and its growth. Showing we are directly investing in their potential.	**Humanitarian Impact:** Using the equity of the marketplace to target lack of economic opportunity/unemployment (12.1%), Safety/Security (14.1%), Food/Water Security (18.2%), Government Corruption (22.7%), and Poverty (29.2% of global population affected)..	**Gainers/Losers:** Driven by value-given approaches we rank the top value providing companies against those who are not directly tying their profits/value of altcoins/alt-tokens to the benefit they bring to the users and not just its valuation.	**Consumer Purchasing Power:** Comparative relationship between FIAT purchasing power where our users are and the purchasing power of users with crypto in capital markets facilitated by SGMX as an Investment-Service Marketplace (ISM).
Physical Quality of Life: What is the current market health of the public and by using SGMX the impact we are creating and countering detriments in public welfare.	**Education:** Promoting the spread of education through all countries that we have the ability to do so. Raising literacy, education for personal finance, business applications, and innovation.	**Corporate Social Responsibility:** Corporate collective working together to create larger social impacts, and who within it is leading the creation of social impact via SGMX.	**Poverty Level per Capita:** New creation of capital flow promoting decrease of poverty via digital assets. And at what measure is SGMX influencing as such.
Secular Stagnation: The depth of financial barriers creating immobility of capital within communities and comparatively with SGMX infrastructure within them.	**Gender Equality:** Financial instability created by gender bias for wealth creation. Countering it effectively with SGMX as well as promotion for equal representation through ventures, investments, infrastructure, asset classes, etc.	**Lives Impacted Per Capita:** The physical impact companies have as a collective on SGMX via their products, services, goods, and all other assetures of companies that promote financial growth and stability.	**Entrepreneurship:** The creation of new ventures, investments, dApps, products, goods, and services facilitated on the OFN by SGMX that would face barriers of FIAT markets and limit entrepreneurship.
400/400	**400/400**	**400/400**	**400/400**

1. What core value do you deliver to your audience? -
    a. The ability to invest back into yourself with a platform that invests into you.
        - A platform that provides access to wealth creation that keeps you at front center
        - A platform that builds wealth as you build with it.
2. What bundles of products/services are we offering to each customer segment? -
    a. Build a market model for mainstreet not wall street.
    b. Driving market power into the hands of the users to drive market growth
    c. Segment 1
        i. Non-College Educated
        ii. Lack financial resources and literacy
        iii. Lower middle class
        iv. Have some technology experience (i.e. smart phone)
        v. **Product Offering -** Gamified platform to make interactions easier (Monthly fee $3.00)
    d. Segment 2
        i. Middle Class
        ii. Americans who have been disadvantaged by the current system
        iii. College Educated
        iv. Has financial debt
        v. **Product Offering -** Pro Version where it is a little level up from Segment 1 (Monthly fee $8.00)
        vi. SGMX Marketplace for companies that want to reach new customer bases for their goods, products, and services. SGMX Investments for our users to invest into a new asset class of companies that list on the platform. SGMX CFI has the ability to invest and create wealth from public equity investment projects.
    e. Thinking of the models to bring to each core segment and how to scale.
    f. **Segment 3** - Hybrid of 1 & 2
        i. Emerging Economies
        ii. Where there is no financial infrastructure we replicate the existing model and build it for the countries to use as their own DCM.
        iii. There is existing financial debt

3. What jobs are customers trying to complete?
    a. They are trying to build passive income based on a life goal that they have and the platform helps them achieve that.
    b. They are trying to list their goods, products, and services on a marketplace
    c. They are looking for new investment opportunities within a transparent socio-economic capital market.
4. What pains do they experience when trying to achieve their goals? -
    a. lack of capital to invest as a barrier to entry, where the platform invests into them via Social Credit/ Social Xp

    b. Lack of education on how to invest or create digital wealth. A majority of our focus for SGMX is creating financial literacy of what we are, what we offer, how we can benefit the user, and the structure of our market compared to FIAT markets.
5. How does your product/service help them achieve their goals/relieve their pains? -
    a. It educates the users of the marketplace/ investment market.
    b. It becomes the platform for investment and commerce
    c. It provides social credit for the users to create wealth.

Customer Segments

1. Which groups of customers are you creating value for? -
    a. for women,
    b. millennials,
    c. primarily driven towards people of color who have not been invested into with proper financial infrastructure that invests into them as they use the product.
2. What are our most important customers? -
    a. our most important customers would be millennials between the age groups of 21-38 this is the age where building wealth begins to become important,
    b. The pitch that we can give is using our vehicle to build digital wealth so that you can invest into yourself to live the life you want to live.
    c. We invest into you so you can truly invest into yourself.
3. What differentiates our customer segments? -
    a. Corporate entities
    b. SMEs
    c. Startup founders looking for an alternative marketplace
    d. Entrepreneurs,
    e. Middle class families
4. What opportunities are there to reach new customer segments -
    a. creating targeted advertising videos that show demographics that we want to appeal to using the SGMX infrastructure - follow up to understand more of what the question means.
    b. Partnering with investment firms in the space that can use our infrastructure as a means of driving further investment into their funds.
    c. Partnering with leading crypto exchanges to integrate our system as a mechanism of their exchange. An example would be to get a partnership with Binance and SGMX reaching already established crypto investors in the space.

Revenue Streams

1. For what value are customers really willing to pay? -
    a. they are willing to pay for an ecosystem that circulates their capital back in investments back towards them,
    b. a medium that is designed with compounding their investment in mind,
    c. i.e Acorns uses micro-investing to help users invest into stocks of their choice and help build their portfolios.

2. For what do they currently pay? -
    a. right now our main competitors would be Robinhood which has zero commission fees and Acorns, both charge low subscription level models for next levels
    b. Our biggest differentiator is SGMX CFI, our public equity projects.
3. How are they currently paying?
    a. What is the value that we are providing for the $8.00 that they pay for the system.
    b.
4. How much does each revenue stream contribute to overall revenues?
    a. Collecting 0.09% per transaction that happens in the ecosystem
    b. 15% in equity ownership per CFI Investment project held by Samuel Global Management LLC.
    c. All CFI Projects will be on traded on the mainnet growing the market capitalization of the Samuel Decentralized Capital Market - SDCM
    d. $3.00 for the entry package for SGMX
    e. $8.00 monthly fee for Pro

Customer Relationships

1. What relationship does the target audience expect you to establish and maintain with them? -
    a. My vision on the relationship between our target audience is that of an investment advisor based on the platform and what we have built on it towards the users.
    b. Educating them on how to create trades,
    c. how to vote and participate in the public equity projects,
    d. Our goal is to show how the platform brings the most amount of benefits to the target audience and as stated before staying on the platform longer.
    e. Inform the market as it grows with new ventures and projects that further facilitate the growth of the community and the project as whole.
2. Which ones have we established?
    a. We have not established any yet because we are still figuring out how to build ourselves and build the right model.
3. How costly are they? -
    a. **Instagram** - 0.70-1.00 CPC the highest CPC groups are the ones that we are directly targeting. Which would be the following
    b. 18-24, 25-34, 35-44
    c. **Facebook** ads are more expensive to target women which is also one of the core demographics that we are targeting.
    d. Instagram has the highest ad placement of all digital platforms, though they are cheaper for B2C.
4. How are they integrated with the rest of the business model? -
    a. There are three core elements that come into our advertising which comes into the business model as one of our variable costs for the SGMX Product.
    b. Your Facebook campaign’s bid
    c. The relevance score

    d. Estimated action rates

Channels

1. Through which channel does your audience want to be reached? -
    a. Twitter,
    b. Instagram,
    c. Facebook
    d. Snapchat
    e. TikTok
2. How are we reaching them now? -
    a. We currently are not reaching any of our audiences through the platforms that can best reach them.
3. How are the channels integrated? -
    a.
4. Which ones work best? -
    a. Facebook because of the different communities that are created that can be targeted. Making it the easiest way for us to scale towards the attention of the SGMX Project.
5. Which ones are the most cost efficient? -
    a. Tik Tok is $10 per CPM, require a minimum of $500 per ad campaign
    b. Facebook
    c. Instagram
    d. Snapchat
6. How are we integrating them with customer routines? -
    a.

Key Activities

1. What key activities does your value proposition require?
    a. Building new investment products around the existing platform being Indexes and ETFs
    b. getting new companies to list and provide their goods/products/services on the marketplace,
    c. as well as new public equity projects to be listed on the CFI homepage for investment from the users.
2. I don't know yet - follow up with Shamil
3. What key activities do you need to deliver the customer experience?
    a. Ease of usage as well as reliability in time

Key Resources

1. What key resources does your value proposition require? -
    a. programmers/software engineers for the development of the platform.
    b. Media agency to create visual content that will show the benefits of the platform for each demographic that we are targeting.
    c. As well as educating about the new market that we are building the infrastructure.

2. What key resources do you need for distribution? -
    a. Intellectual - getting patents on our product so we can scale the venture to other countries specificity towards how those countries can be in the best position to win with our market model.
    b. Human - getting more creatives and investment analysts to come on board and create market analysis of the product that we are building and how it can benefit the end consumer.
3. What key resources do you need for customer relationship management? -
    a. Customer service
    b. Online question and answers and how to apply it
    c. Chat bots to connect to our users

Key Partners

1. Who are your key partners/suppliers? -
    a. financial institutions
    b. software development companies,
    c. Market makers
2. What key resources are we acquiring from partners? -
    a. technical skill to expand the market and the financial network,
    b. financial network of the companies that are joining us as partners,
    c. additional corporate partners to expand the marketplace.
3. What key activities do partners perform?-
    a. advising ourselves in the direction that we are going and how to scale the venture

Cost Structure

1. What are the most important costs?
    a. Advertising/marketing
    b. Technical development
    c. Legal development of the SGX Token and SGMX Project
2. Which key resources are the most expensive?
    a. Legal development of the SGX Token and SGMX Project getting the right legal advisors will pay the most crucial difference to the success of the project.
3. What can be changed from a fixed cost to a variable cost?
    a. Advertising/marketing

Obsidian Capital

Obsidian Infrastructure Corporation

Managing Partner & CEO - Jonathan Samuel

Written by Jonathan Samuel
BSBA - Business Administration and Data Analytics
University of Massachusetts Lowell

Table of Contents

What is Obsidian Capital?

Obsidian Investments or Obsidian is a digital currency investment and infrastructure company. We are in the business of asset creation and discovery. We see it as our mission of creating investments that are tied to digital infrastructure in the new era of asset creation and allocation in the digital age. Finding where there is undervalued potential and deploying capital to actualize it within capital markets. This means that we use our independent digital currency to invest across markets, sectors, and companies. While also building digital infrastructure to access new investment opportunities within those markets where we own the infrastructure.

Our Initiative

For the last six years at the time of writing this report. Obsidian Capital has worked to develop a working process to better understand asset discovery, allocation, and transactions within emerging markets. Capitalizing on investments and infrastructure that are undervalued. One is because of the markets that they originate from and caused by the lack of capital mobility for the assets within the global economy.

Our initiative is with this understanding of creating an alternative digital global economy that can function where asset classes can be appropriately weighed and invested into by the international market. That is the objective of this venture and the project it entails.

We source the assets and integrate our financial infrastructure to sell and or buy the assets with our independent digital currency. With the means to stabilize high inflation countries with this new ecosystem for them to grow and introduce themselves within the global market. You may be wondering why would you create a separate financial economy for emerging markets? The opportunity lies within the asset classes which is correct, but most of the true opportunities lie in the ability to own the underlying financial infrastructure.

To transact, trade, and create investments across the assets. This is about building and owning the highway that all the cars are driving on top of to reach the city center. And this proposal will

show the core layers of infrastructure that are strategic to the execution of creating our own global economy that is within our sphere of influence. As well as the impact on the digitally developing world around us today. Emerging markets have informed human capital that is specific to the skills and products that are relevant to their day-to-day lives.

The problem that exists is that the markets are not entirely “free” because of the financial infrastructure that is archaic in comparison to the developed nations across the world concerning the goods, products, and services that are created. It creates bottlenecks where bureaucracy and corruption can take control of the market sectors that would provide more value if given the ability to operate without the excess means of control.

We will be creating systems that ensure democratic financial power and the tools that create new goods, products, services, and companies within the digital age. That will transform more purchasing power to those within these new underdeveloped markets to ensure that their capital and in turn our own is directly competing with the global economy. By transforming the way commerce and trades are conducted and owning the infrastructure that supports it we will be able to have access to resources within nations that we can directly sell at prices that we set for the product in question.

We are going to create, invest, and own the infrastructure to transact and trade the assets of the country. Bringing them with the speed of the internet with distributed financial systems to international markets. And create redevelopments that will replace the archaic infrastructure of the countries currently existing with our infrastructure. The country becomes a greater competitor within itself domestically as well as internationally amongst other nations. And we, in turn, Obsidian Capital and its shareholders, set ourselves on the path towards becoming the largest digital asset managers as well as owners across emerging markets where we own the table that everyone is seated at. As well as the chairs.

OSD Reserve Currency Financial System - Infrastructure Components

Treasury Backed Stablecoin Currency – Orion System Drachma - 1.00 OSD

The reserve is an algorithm allocation of digital currency that is owned by the majority as a global private asset for redevelopment and investment across industries and sectors. Caretaken by Obsidian Capital exclusively on behalf of the market system. An asset that is owned and operated for investments, transactions, and deal financing. It is a privately operated asset for all shareholders to conduct transactions and members of the network to participate in.

Think of the currency as a medium of trade exchange. A trade currency for international deals. Where all people can use the currency to facilitate trade, a currency that is fixed upon a mathematical formula that stands alone and on par with global reserve currencies. Directly being

able to cohabitate with the U.S. Dollar, Swiss Franc, Japanese Yen, British Sterling Pound, and European Euro. This is specifically created as an algorithmic currency that can stand on its own to not threaten the financial sovereignty of any nation that which the currency may interact. It is also good to know that the currency can be used within digital environments where the goods, products, services, and companies that interact with them are digitally created. Creating new opportunities to invest in brands and products that operate in the MetaVerse. This means that Obsidian can operate as a traditional asset manager and invest across asset classes as well as invest in non-traditional cases such as the MetaVerse where all goods, products, and services are digitally focused.

An original, sovereign, digital currency that is used for economic development, international investment, and infrastructure allocation. Working together to create new financial markets and benefit the creation of new market operators with asset discovery.

This provides new directions for Obsidian to directly create new ways of Foreign-Direct Investment across the entire globe. Targeting two main elements of financial investment, liquidity, and credit. The currency can be used through the digital infrastructure that is built supporting it to create liquidity for assets or markets that are facing liquidity issues and with those assets further backing the currency with a now liquid asset that can be sold to anyone by converting the currency out of the system into the reserve denomination equivalent.

Liquidity for assets in emerging markets is one element that is done properly with an algorithmic currency but being able to issue credit directly where nations can use the currency for development within their borders. Solving both liquidity and credit will allow Obsidian to issue itself as a large-scale FDI Investor, conducting foreign direct investments with digital assets across sectors and countries.

We will be effectively going through cities, communities, and countries with the structure outlined in this proposal in the process of creating our shared global economy. Digitizing the issuance and transaction of securities and other financial products. An environment for emerging economies to reallocate assets that are not accurately represented within the global economy. Directly facilitating trade between nations that share the infrastructure that we are building and will own. Giving new room for commerce, but importantly trade that can directly lead to the development of nations as commerce is happening at the speed of the internet where each company is digitized and each person can sell and or buy goods, products, and services directly with the currency. This is how we enable the building of a financial highway and reserve system for countries that face high amounts of inflation where their currencies are not performing as well.

The Orion Drachma is tied to the digital ecosystem and economy that will facilitate the development of economic value tied to brands, products, and companies producing digital services and new market flows for economic activity. This environment will create an alternative secondary market that is designed to run on par with the global economy. And will reflect the true value of emerging market nations and the value they provide to the global economy. Their financial infrastructure will be replaced with the ones that we create and integrate into their economy to provide digitization of their markets and systems to support growth within international markets.

There are a total of six pieces of infrastructure that go together to create this entirely new ecosystem and reserve system. The first is what we have discussed so far which is the algorithmic currency which in our system is known as OSD or the Orion System Drachma.

Second is the speculative asset that is investible within the ecosystem that we are creating which is known as $GIC or Orion Infrastructure Corporation. The stock of the market will be backed by the infrastructure that is built as there is growth that supports further integration of the infrastructure. There will also be a price change in the asset as more digital assets are built on the network. With increased digital companies, brands, products, and services offered within the economy independently as well as to the global economy.

The third is the AVS Platform or Asset Value Securitization platform. This is how we will be digitizing the assets of countries and directly selling them within a private market to the global economy by providing securities of natural assets from those countries. Natural assets are natural gas, oil, commodities, minerals, metals, land, energy products, and or infrastructure. The products will be denominated within the algorithmic currency and are tradeable within other sovereign currencies.

The assets will be priced in comparison to the global economy and other emerging markets that we are going to be working with and are invested in. The AVS Platform is one of the core pieces of infrastructure that is needed to properly run this alternative market that we are creating. It is a means of compliantly selling assets of nations with securities that we own and manage on the platform. The stocks, bonds, derivatives, futures, and options of the assets are issued through the platform and sold to other markets where people can buy them through us.

Obsidian Capital then facilitates the access of the trade happening between the buyers and the sellers throughout the countries that we are working with. Something that can be called investment wholesaling. You can invest in the market through us in the countries that are listed on the platform as well as buy and or trade the physical product that is listed through the platform. A literal trade platform for trading securities and trading assets from the countries we

operate out of and invest in as a company. Obsidian Capital is sourcing the investments and bringing them to the market on the platform to sell directly to the buyers.

Fourth is the smart chain of the economy, a blockchain protocol that facilitates the entire transaction history of the economy and every trade that has happened and where it is going. It is on top of this smart chain that digital companies, brands, products, and services will be built. A feature that will be directly available from the platform where members of the network can determine what they need to create.

Think of it as a foundation for a house that is being built. In the same regard, this is the foundation of the market that is being built. This is known as the Kratos Smart chain. Which will follow an asynchronous byzantine fault-tolerant architecture with a proof of stake consensus algorithm

What does that mean? For a transaction to be approved the people who validate the transaction need to have an asset that is used as the reward for solving complex mathematical problems and recording the information once it has been approved to the general ledger. To solve this problem of where each person would become a validator of the network Obsidian decided to set the condition of pre-allocation. Meaning that each person who joins the network will receive funding in the currency from the reserve as long as they wait five months before moving any of their capital as it is allocated to them. Beginning at a multiple of 4x per month starting at $2M OSD.

This leaves the final allocation ending with $2.048B OSD at the end of five months with ⅔ vested in the system to provide financing for trade. That means each person will have $1.372B OSD locked creating liquidity in the market network and $675M OSD for personal investment allocation and creation. This allows each member on the network to work as a financier of trade. Our goal is to lock the network to 250,000 validators on the network which brings $343T OSD locked in the network.

There can be additional users that join the network where they will facilitate the movement of the economy as participants of the ecosystem that is being created. The initial stage will focus on the creation of the validators for the network. Allowing the Kratos Smart chain to become its economy and trade financing network with liquidity already locked in to provide to those that need it. This ensures that the market maintains liquidity at all times for the parties that we engage with; as well as creates value within the countries that we operate. Solving the initial question of creating liquidity within the markets that we operate out of.

Fifth is the Digital Trade Exchanges also known as DTXs. These trade exchanges will function as a bourse similar to the New York Stock Exchange and operate on top of the Kratos Smart chain. A decentralized digital stock exchange that is cheaper than the big board for listing where

all operators of the network can participate and buy from the exchange the companies that have listed on the exchange.

This Will come to play after the aforementioned pieces of infrastructure have been built properly. There will be one main exchange and five exchange centers for commerce and trade to originate. Alpha will be located in Puerto Rico for its ability to function as a digital asset haven. Bravo will be located in Colombia. Charlie in Ghana, Delta in Sri Lanka, and finally with Echo in Indonesia. The last piece of infrastructure is the mobile terminal that connects the entire economy.

Key points when it comes to the Orion Drachma, $OSD:

- The total supply of the reserve is as needed.
- Which will be used for direct investment capital across emerging markets and undervalued asset classes to be bought up at scale and held on our books.
- The Orion will function as an ISP-Backed Reserve Currency within foreign exchange markets paired with the currencies of nations that we integrate with or on own our FX market with our DCM partners.
- An independent reserve currency that cohabitates with USD, Euro, GBP, Swiss Franc, and YEN.

The algorithmic currency is stabilized with a basket of stablecoins and other assets that are held via smart contracts. The primary asset that we will be targeting is debt securities, municipal bonds, and other debt-related products. Including but not limited to preferred stock, collateralized debt obligations, euro commercial paper, and mortgage-backed securities.

Each transaction becomes a part of the ledger in this fashion to create a financial support structure that can be built on top of by the network community and those across the globe who participate in the ecosystem that we have designed. In turn, each person becomes a part of the mechanisms that drive the infrastructure forward in the premise of each transaction becoming a part of a truss system that is tied to the foundational network. As such each person on the network becomes a ‘beam’ that ties the economy of the nation to the digitized economy that it is transitioning towards. A core concept that explains why liquidity was assigned to each person as it was in the beginning.

This ensures that each transaction grows and builds the supporting frame which is the country's portion of the highway that we are creating, setting the foundation of the infrastructure that we build and integrate into the country at scale.

A hedge against the global economy and banking on the investment potential of countries by directly integrating our investment and infrastructure with the countries in question. The

currency will be invested in emerging markets and developed economies first to tackle the debt the country has on its balance sheet. With a multiplier of 1.5-4.5x of the debt worth the first installment will be held as the equivalent to the debt.

This is first done to create the pairing of the debt between the fiat market system and our own, where ours is leveraged against the debt owed by the country in question in fiat. And placed as an investment into the central banks of the country with their access to the market we are designing. This gives them the ability to directly invest in the ecosystem with the reserve that is allocated to them for the worth of their debt as our bet on the country's potential to circumvent the problems it faces in the support of the system we have brought to them.

Which will be converted into a financial product that is known as debt security as a new bond will be issued and held by Obsidian Capital Assets. Where we will collect on the future revenues of the country across various sectors and industries of focus for the installment of 5, 10, 15, and 35 years on the bond at an interest rate of 2.5-8.5%. This will be determined by the risk of the country across its fiat infrastructure that is integrated into the country and its relation to the world around it.

The second installment of the investment will be to invest in infrastructure, natural resources, commodities, minerals, commercial real estate, energy products, metals, and transportation. Complete digitization of all sectors and asset classes available only on the network. All of which its digital value will be transacted on the network and held within the ecosystem that we have created. Making the assets transactable at the speed of the internet, and investible with a higher rate of liquidity than would be traditionally offered with their offerings. Here we begin the process of addressing Credit within the countries.

Example:

The country of Ghana has a total debt of $27 Billion on foreign debt. Where it had one of the worst starts among emerging markets, extending last year's 14% loss in market value as investors leave amid Ghana's deepening debt. While this is a problematic predicament, Ghana is one of the wealthiest nations in western Africa of its natural resources which are the following.

- Timber
- Gold
- Diamonds
- Bauxite
- Manganese
- Oil

Obsidian would invest 1.5x-4.5x times the value of fiat debt that is held in the country as in the transaction we would be building new financial infrastructure that will help advance the

economy and its investments domestically within the financial markets. On the high end of allocation of capital, it would be $121.5B OSD at 2.5-8.5% for 35 years that is invested into the country of Ghana. In turn for building the financial infrastructure for the country, we get exclusive rights to the natural resources to sell in the public markets. Priced and securitized for the market on the AVS Platform financed through the Kratos smart chain with other operators that want to allocate capital to the investments that Obsidian is creating in Ghana.

A circular economy for global investment and development amongst nations. In the process of developing this means for the trade of Ghana will be the smart chain that we build for the economy as a whole and pair it to our smart chain. This refers back to the image of owning the financial highway that all the "cars" are traveling on top of to reach the city center. All new assets are created through the ecosystem that we have provided to the country of Ghana and they can pay back their debts faster in fiat with the new assets that are created in our economy.

The alternative financial market is used with its infrastructure to advance and digitize the economy of emerging markets that are facing high inflation and burdened with fiat debt. This creates an economy that provides more value to the people directly and the country as a whole and by choice, if they want to interact with the other countries that are outside of the system they can determine how they want to as well as to what degree.

With the assets of the country, there will be financial products through the AVD Platform that will keep all of the assets that are underperforming because of the structure they are currently operating in on our books as we manage them and our investment in the emerging markets that we are actively working with. To tie off our investment we will also allocate a portion of our stock to the emerging market nation that we are working with which will be disbursed to the local economies and communities. This would be a direct allocation of the $NOW digital stock of the market that is allocated to the countries, communities, and cities in question. As we will have the liquidity for them to exit out of their stock ownership and sell it back to the Reserve.

Key points when it comes to the stock of the market, the $ODN digital stock:

- The stock of the market is facilitated by the reserve currency system.
- By investing in the $ODN Token you are investing in the financial system that represents the economy.
- As well as the infrastructure that is created by the continuation of the reserve currency system.
- Total supply of $500 Octillion ODN.
- Value is backed by the Kratos Smart chain of the reserve currency system. All economic value built on it drives the value of the token.

With the smart chains, we will have software development kits or SDKs where technical users can benefit from the software architecture to build products, brands, services, and companies that

drive economic and transactional volume on the smart chain. We will be building smart bridges which side chains connected to our main smart chain to drive the transformation of economies into the digital ag4 that is cheaper and scalable to transition the existing fiat infrastructure that is in place in these communities, cities, and countries.

Each chain will have its stock exchange or bourse that users can transact with and will act as the anchor that connects the markets between the ones that we are building and the ones that we will be building for the counties that we work with. This will allow the citizens of counties to prepare their companies for listing on the exchanges and the entire trade network that is created because our smart chain can invest in the entire country depending on where the offering is coming from irrespective of us. We set it up and reap the initial benefits of what we are doing but the entire return is to the shareholders of the network and the network as a whole.

Key points of the AVS Platform:

- Digitize the value of any good or product to be sold within the platform amongst other users.
- Securitize real-world assets that can be sold in international markets.
- Issue stocks, bonds, derivatives, futures, options, REITs, and other financial products that can be sold within the ecosystem amongst users. Based on the natural assets that are available within the nations that we partner with.
- Bringing underappreciated natural resources, commodities, energy products, technology, real estate, metals, and minerals, to international markets while also structuring the return to drive reinvestment into the communities driving local network effects on economic activity creating forward-thinking approaches that create financial stability and sustainability.
- With transparency, legitimacy, and thoughtfulness across supply chains of goods, products, and services that are transacted.

A majority of the currency and the stock will be held in public reserves that are democratically distributed for sectors and causes of interest that are pertinent to the global economy and the community that comes with it. Each shareholder vote is worth 1 regardless of asset ownership or rank in the system. Obsidian will act as an independent operator of the system and work to structure investments and opportunities for the network. Its portions of the network that is owned will be privately held and maintained by the company the rest of the network will be publicly held where shareholders will determine the direction of the market and how it functions as a whole without altering Obsidian and or its role from the network as a neutral independent operator of the system.

In turn, we create a financial reserve system that is democratically voted in for investment in our global communities and to drive business investment yields as well as research. A financial highway that is our own for our financial economy and ecosystem for people to build freely with

little to no barriers only their creativity limits what they can build in the ecosystem. With the financial tools to assess their financial value and at a fair market valuation for their hard work. Connecting other financial highways that we own as a lease for 250 years. With a 60/40 split on transactional revenues, we walk with 40 the country keeps 60. Building a global financial highway that drives growth in emerging markets and capitalizes on investment in their assets.

Key points of the Kratos Smart chain:

- All economic activity and value will be transacted within the smart chain.
- It will come with a public development kit or an SDK that international members can use to develop and build their digital businesses online.
- This will serve as the main smart chain; there will be other smart chains and side chains developed paired with each other and the Kratos smart chain creating a distributed financial highway with the global economy.
- Creating an independent foreign exchange market and financial system that will be using the reserve currency to create liquidity within markets. And trade against other foreign currencies

Asset Value Securitization Platform

For this system to work out of the information that has been defined to this point two pieces of the infrastructure are vital to succeed entirely. The reserve currency itself, Orion System Drachma $OSD. And the AVS Platform, the Asset Value Securitization. This platform is what enables Obsidian Capital to properly assess the monetary direction of assets that are listed on our platform.

Example: In the north of Sri Lanka there are oil reserves that are around the range of 300,000 metric tons of crude oil. At the current value per market, at the time of writing this proposal, the 300,000 metric tons of oil sells for $217M US. Dollars. After refinement and development of the land to facilitate the trade of the oil to other nations the platform prices are based on long-term valuation from volatility around goods and products.

And can sell the physical product for a newly discovered oil market of Sri Lanka for the transaction between $250M to USD 1.5B directly supplying countries that do not have oil from Sri Lanka increasing and strengthening geopolitical relationships between nations and the host nation of Sri Lanka. This move itself would allow Sri Lanka to have its commission for oil with other Asian countries and form its trade own block outside of the influence of OPEC and facilitate demand directly within the global economy. This means that Sri Lanka can operate outside of the global economy and benefit from the price action for its assets that we directly carry out for it as its trade infrastructure.

As an emerging market we would also apply the 1.5-4.5x multiplier of the debt that the country

has and with our investment in two parts one to cover the debt and the second for the assets in the country itself that would be listed on the platform. Sri Lanka has a total debt of USD 51 Billion our investment would be at $51B OSD with the reserve currency. And at a high end of 4.5x an allocation to securitize the value of its assets outside of the range of inflation within its economy and appropriately valuing the assets to the total worth of $178.5B OSD on our books. Long-term, with macroeconomic trends directly bringing the products to be sold into the public market from the platform. Total investment in the entire process of $229.5B OSD with a line of credit of 750B OSD. Where $178.5B OSD is privately held on our platform and sold with our ecosystem to international markets.

Obsidian sets up the framework and platform that allows all parts of the supply chain to be monetized and turn products into assets that cover the following areas. By directly tackling the debt of the country and valuing the country's human capital and potential market performance with our guidance and the entire trade network we can directly, at scale with influence, invest in the entire economy through the market we are building and will create.

- Natural resources
- Energy products
- Metals
- Minerals
- Agribusiness
- Commodities
- Transportation
- Commercial real estate

The platform allows all participants to engage with assets that have been appropriately valued in the reserve currency to be liquidated out of our position into the U.S Dollar, European Euro, British Pound, and Japanese Yen. From there they can be invested across the community who can see the entire asset classes and from where who, and when they are going to reach the market if applicable. If not then how they will be sold to the foreign markets.

The platform functions as an open market that is transaction open at any point in time of day for sale,

- Futures
- Derivatives
- Options
- Digital Stocks
- Bonds
- REITs

Covering the areas that we are going to securitize and turn into financial products that can be

bought and sold for the time being covering the entirety of the Sri Lankan economy through us. We become asset managers, investors, infrastructure builders, and operators for the country's market as a whole. As it is invested into ours as the Sri Lankan market sector of our economy. And our economy *"Orion Digital Capital Market"* is an active operator for the Sri Lankan economy at scale. A distributed financial market, the ODCM, can function as a neutral infrastructure operator and bring continuous value to the country as a whole. Not only for Sri Lanka and Ghana as mentioned in the examples above but for every emerging market that we can interact with and develop directly.

Orion Algorithm and Digital Infrastructure Development Program.

For the last six years, we have been working on developing a system that would benefit our means of capital allocation with our digital currency. Finding the right way to base our decision on how we allocate our currency as an investment and build supporting infrastructure to drive our investment forward. This is where we come in with the Orion Algorithm. A nine-part measure of understanding how we are going to allocate currency where we inject liquidity in the markets and also allow countries to borrow from us directly.

Orion Algorithm

Six variables of the algorithm focus on the economic stability of the economy itself and three are focused on the companies in the country that we may partner with to determine their viability of the new economic structure that we are building. These are the values, and through the Last, In First Out, or LIFO method, they are approached from a descending value.

1. Currency
2. Inflation
3. Banking
4. Dividend
5. Credit
6. Earnings per Share
7. Financial
8. Cashflow
9. Systemic

From the LIFO method, the last variable is pulled first in the algorithm and applied for research. That means we start with nine or **Systemic**. The focus of this variable is to understand the systemic risks that are prone and or prevalent to the economy of focus. We look at the historical status of the country and its financial performance to see how it ranks against other emerging markets. Systemic is focused on helping us understand what is the risk of our investment going into a country whose economic systems do not properly support liquidity. Is the plumbing good? That is our first approach to understanding the country in question. Obsidian as mentioned in the

earlier parts of this report is structured to function as an FDI Investment and infrastructure developer. As we invest in foreign markets we as foreign direct investors want to understand how our money is going to perform as we invest in the economies in question. Making sure our investments do not face illiquidity is the first step.

The second variable is **Cash Flow** which focuses on the companies of the market we are deciding to enter. Have the companies not missed cash flow more than once in their existence, we are strict on the performance of companies as we want the best players to be able to work with us and the infrastructure that we are building at scale. Cash flow is also placed as the second variable to see how money is moving within the country and to where and what is money moving. Large scale investments, reallocation of assets across public markets, selling off assets across sectors. We want to see where the money is moving. What aspects of the market see the most concentration of capital and why is there an opportunity for us to invest or is there a need to create.

Financial focuses on the ability to maintain credit and liquidity, that is all that it focuses on. **Earning Per Share** for companies that have strong earnings can perform with the conditions of emerging markets. Their performance with cash flow, earnings, and dividends will determine if we can invest in these companies and have them as assets that are listed in our ecosystem that our trade network can invest into at scale as a whole. From the beginning of the process, we have analyzed whether or not the plumbing of the country is strong enough for the liquidity we are going to inject, understanding where is the money going and to what as well as the performance of companies for their cash flow. If the market can maintain credit and liquidity and strong earnings or not with consistency amongst the companies that we are interested in working with. The first four variables show us the initial view of the economy for investment activity and the companies that are participants in it.

Credit focuses on the borrowing activity of the market and if it isn't sufficiently met then that provides an opportunity for Obsidian Capital to issue capital that companies can borrow from. We want to see if there is healthy borrowing activity amongst the banks and companies that are operating in the economy but also the credit that is issued to the consumer. Is it healthy against the credit markets of neighboring countries? Who are the biggest creditors and who are they lending their money out to? Where is it going and what is it doing for what reasons and for who's reasons? **Dividends** we want strong healthy companies that provide dividends to be our partners for our investors and shareholders. Companies that have not missed cash flow, earnings per share, and dividends compile the belief of Value-Indexed Theory for long-term investment amongst companies. We want healthy companies to be a part of our economy where they are listed but also for us to invest in companies that can perform over time with our strict criteria for investment.

Banking focuses on the local economy and the network effects of money moving across

municipalities and cities at scale. Is there infrastructure that can help the local economies scale or is it an opportunity for us to bring our partners and our infrastructure and become it? **Inflation** in countries where inflation is high our asset, $OSD, would be used to bring trade and commerce. Giving their markets room to grow and restabilize other asset classes helps drive demand for labor and services as the assets are priced in the reserve currency and not the currency of the nation in question. This is done as a hedge for the economies to gain with a multiplier for what the assets are worth and then bring that value back to the economy as an influx of revenue for the country. **Currency** once we understand the four following variables we learn of the environment itself and how it is operating as a whole. From there we decide on whether or not we want to invest in the country.

Digital Infrastructure Development Program of the Orion Digital Capital Market

Debt Restructuring

Through our ecosystem, there will be many opportunities for everyone to build and create value for themselves and others. With that being said these are some of the current initiatives that will be underway as part of our digital infrastructure development program.

We categorize and organize a proper structure for cities, communities, and countries' debt. We securitize the debt that is being processed through the system and create an investment product known as debt security. Debt securities are also known as bonds that are tradable forms of liabilities. We own the debt security as an asset that is tied to a repayment schedule.

5, 10, 15, 35, and 100 years.

We allocate the entire (a) debt worth in our algorithmic currency directly at 1.5-4.5x its value. If total debt is $10 on the high end they receive $45 for their debt caused by financial instability where assets are undervalued and not priced in the market for what it is worth. That is now in our books. We do this not as charity for any of the regions that we invest in but as direct investment into those areas and have the first right of refusal to any developments that we are not owners in. This would be carried out by ensuring that our investments are directly tied to the governments of the cities, municipalities, and or countries in which we operate.

Or (b) issue a new issuance of debt that the borrower would be owing to us. This is where if it is a new issuance of debt where we would be lending the capital out to the borrower at 1.5-2x the requested value. For every $10 we lend out $25 but does come with a fixed percentage of interest around the ranges of 2.5-10%. We invest what is required by an investment reserve system where the additional is created into a fund that can be drawn from.

We do this to drive broad-scale investment across multiple sectors that the borrowers would know best. With additional capital, we show them what else they can use their money for in the

network and what we are building additionally so that the app/platform becomes a one-stop-shop for all of their financial, creative, and social needs that are the cleaner in design and flow.

We also tie in a contingency saying that for the additional capital that is allocated we get the first pick on investments and structure them. The capital value of the investments needs to be built on top of the platform and further support the growth of the digital economy that is being built by the transaction that is conducted by us. Growing the value of assets and infrastructure in our ecosystem drives the value of our stock further as we continue to grow as an investor and developers.

Direct Investment

Upon completing your profile on the platform each member is allocated $150,000 OSD and they can earn a higher allocation if they don't cash out. This is part of the pre-allocation system of the reserve system where if you do not remove the currency that is allocated to you then a higher deposit is invested into your account in the system.

$2M, 4M, 8M, 16M, 32M, 64M, 128M, 256M. 8 months of not pulling the money and letting it grow in the system. This is for everyday users of the system and not those who are validators of the trade network. There are two separate allocation systems in place one on an eight-month timeline as stated above. And another on a five-month timeline. Those who become validators will only have to wait five months. It is important to distinguish that validators will be critical to the execution of trade deals and act as trade agents on behalf of the network. Looking for places where a trade can be fulfilled and carried out when brought to the network.

Those who wait eight months for their full allocation or less are shareholders of the network. They can still participate in deals throughout the network; it's simply that their function is different. Each person is allowed to choose if they want to be a validator for the network or a shareholder of the network and can operate with new careers based on the economy and ecosystem that we are building.

Each person has a dashboard that shows them their balance, investments, assets, etc. all of which are tokenized in their real-world value and denominated into our currency. Backed by the reserve which stores our currency as well as other currencies that are acquired. USD, GBP, EUR, YEN, etc. The reserve is a public property owned by the platform and managed in exclusivity by Obsidian. The reserve is used to allocate for completed jobs and investments on the network. It allocates for direct investment in each person. Each member will be a multi-millionaire in OSD after waiting 8 months and not pulling out their allocation.

Members can build their digital brands on the platform with real-time alert valuations and issue their tokens for utility and or security this way the common currency is being paid and transacted at points of sale for commerce and investment in the system. With their asset, they can create

brands, products, and services that are all bought with our currency. All of the commerce that is happening digitally occurring within the system is bought with the Orion System Drachma. Driving up the value of their asset for the economic value and activity they provide to the system and the blockchain that further backs our stock.

The more activity that happens in the instances of individuals creating local economies on the platform that is common to their network systems the further deepened is the network to drive economic value and opportunities which backs the value overtime for the stock of the network.

When economic networks systematically create opportunities for growth the growth of some form will grow within that networking opportunity.

If Julia and Jonathan constantly do business and trade goods and services amongst one another eventually there will be another two people who see the trade and the same type of goods or services because the blueprint of a transaction sets the direction for another member to create economic value on the chain. Julia and Jonathan don't lose any business because they are still conducting business but another group can come in and offer the same type of business, not services or products that are distinct to Julia and Jonathan.

- The network compounds economic activity as more assets are created with strong businesses, brands, and companies. New frontiers, economic competition, and financial markets will be created at scale. It will bring out the best of businesses and operators to be able to compete with anyone and bring value that the entire market wants. Being able to list your asset and issue it within the ecosystem is a feature that needs to be unlocked. You get to build and design the token and its supply. You get this feature once you've unlocked a 3 G Score, and participated in elections and governance votes of the network.

You need to build your reputation civically and capitalize on yourself and what you bring to the network to offer your asset to the ecosystem.

Economic Resources Development

Become an infrastructure developer! We will build your book which will cover all of your digital and real-life projects that are financed by Obsidian in OSD. Your book can be leveraged for access and capital investments within the system to market players who want what you have and operate in. Your economic value and revenue are distributed 75/25. We keep 25% of revenue post expenses and the economic value is traded on the platform in the sector it falls under.

If the company you are building for example deals with desalination in emerging markets and other market operators want to invest in this, additional funds will be allocated from our side from the reserves to invest into your venture and the infrastructure you are building.

The same can be said for technical development that is done on the platform. We will be paying

developers directly to come up with new ways to improve the ecosystem in how it functions and drive demand but also improve the economic value and flow of the digital economy. Have more deposits on accounts in fiat and non-fiat accounts that record where OSD is being spent and how it is being utilized.

Anyone is welcome to become an infrastructure developer of the network. Bring your research, materials, and pitch to Obsidian and if approved we will allocate them to you directly. The allocation minimum for infrastructure projects of any kind is $100K OSD - $1B OSD per company.

Drive the redevelopment of our communities ourselves and own the economic value of the system that it benefits. This also includes education systems that can be accessible to all members and their families with no payment needed for educational access.

Independent Capital Financing - ICF

Each person based on their ranking and G score can do a round of financing for short or long-term debt or fresh rounds of capital. Based on their civic and social participation to promote the network and build it forward they are shown what range of capital they are eligible to receive. This isn't tied to the network. This is financing based solely on the person and what they've done for the ecosystem so far.

Standard issue loan OR capital raised based on their social credit in the ecosystem.

- $1-10M OSD

- $10-20M OSD

- $20 - 80M OSD

- $80 - 200M OSD

- $200M - 1B OSD

- Rates vary by credit and what the loan is for.

- 5, 10, 15, 20, 30 years.

First90

Members for their first business loan can apply for a First90 loan where the borrower is lent $5-10M OSD MUST PROVE revenue-creating business in 90 days or enter repayment structure. No interest but must pay all money back. And the remaining balance is returned to the network. Can use the entire system to create ventures successfully and drive the adoption of companies.

Finance Forward Education - FFE

Take the entire educational cost for all years of attending school. Fixed-rate 3.5% Income share agreement to pay off once you have employment.

Employment

Members can be employed by the network for jobs such as finance, marketing, advertisement, creative, videography, and more and get paid in OSD. Range between

- 1.00 - 15,000 - OSD Small Jobs

- 15,000 - 35,000 - OSD Medium Jobs

- 35,000 - 70,000 - OSD Large Jobs

SPECIAL CONTRACTS

- 100,000 - $1M (Alpha - Job 1)

- 1M - 5M (A - Job 2)

- 5M - 10M (A - Job 3)

- 10M - 15M (A - Job 4)

- 15M - 20M (A- Job 5)

- CLEARANCE, Meet Management - Bravo

- 10M OSD & 10,000 Stock (Bravo - Job 1)

- 11M & 15,000 Stock (B - Job 2)

- 12M & 20,000 Stock (B - Job 3)

- 13M & 22,000 Stock (B - Job 4)

- 16M & 25,000 Stock (B - Job 5)

- CLEARANCE, Charlie

- 30,000 Stock (Charlie - Job 1)

- 40,000 Stock (C - Job 2)

- 50,000 Stock (C - Job 3)

- 60,000 Stock (C - Job 4)

- 70,000 Stock (C - Job 5)

- CLEARANCE, Delta

- 1M Stock (Delta - Job 1)

- 3M Stock (D - Job 2)

- 5M Stock (D - Job 3)

- 7M Stock (D - Job 4)

- 10M Stock (D - Job 5)

- CLEARANCE, Echo - Break = Percentage of job as recurring revenue

- 15M Stock, Break 1(3% of Job) - B/1

- 20M Stock, Break 2 (6% of Job) - B/2

- 22.5M Stock, Break 3(7.5% of Job) - B/3

- 25M Stock, Break 4 (9% of Job) - B/4

- 27M Stock, Break 4 (10% of Job) - B/5

Governance

All votes are 1 regardless of person or stature. Similar take towards UniSwap. You need to propose legislation that needs to be voted on so it is amended throughout the entire community. All matters of change towards how the community grows and governs itself. Additional features are being added that affect the entire group and not solely a person. Voting on how to allocate money in the public funds that everyone is a beneficiary of. Voting in political capital and who to invest it into.

This covers the many ways that the ecosystem will grow and the economy will be built with the means of capital directly in the hands of shareholders of the network working directly with management with Obsidian Capital.

Sidechains for Countries

The goal as per the image that was created at the beginning of the highway still stands true. The entire discussion that has been underway and defined until now is about maintaining usage. We want there to be the mobility of assets and classes of investments. We want there to be an environment that everyone can build on top of and benefit from the infrastructure that is built. The first half covered the layout of the infrastructure and how it would be integrated within the economy that is our own and its relation to the global economy by working with countries at scale. The second half has covered how the opportunity will be created within the ecosystem and from that ecosystem how shareholders will be able to unlock capital for maintaining and building the network forward. In that same regard with both sides of our proposal being understood it is not time to cover the side chains that will be connected to the Kratos Smart chain. Three questions need to be answered to understand why we would want to connect side chains to the main network. What is a sidechain? Why are we going to build them? What does it create?

A sidechain is a blockchain protocol that will have the same design as the Kratos smart chain meaning that it will be asynchronous byzantine fault-tolerant. With proof of stake consensus algorithm. This has been defined and explained in previous sections of the proposal. A sidechain is a “fork” like a fork in the road or the main chain. It is a new introduction of the chain that functions independently of the main chain and grows per the development on the side chain.

The Kratos Smart chain is a privately owned blockchain of Obsidian Capital that has public elements shared with the public for their development and the shareholders of the network as a whole. We believe it would be in the best interest of countries, communities, and cities that their local economies can function independently of the economy that we have built. This way there can be a pairing between our market and the markets that are developing independently of our performance. Creating an independent exchange system that is built off the reserve currency and the infrastructure that has been outlined earlier creating the OSD Reserve Currency Financial System.

With all of the side chains of countries, communities, and cities functioning independently tied to the Kratos Smart chain we have a distributed financial system that becomes the highway that we have mentioned earlier. Goods, products, and services of digital brands and companies transacting at scale without the need of legacy financial systems. Able to execute and operate 24/7, all year long.

Asset Classes of interest to securitize:

Natural resources, energy, technology, material goods, commodities, commercial real estate, and infrastructure. - explicitly and exclusively traded in our market.

High Inflation Emerging Markets

1. Sri Lanka - 6.25%

2. Palestine - 2.68%

3. Venezuela - 1,575%

4. Sudan - 366%

5. Zimbabwe - 54.59%

6. Argentina - 52.1%

7. South Sudan - 51.19%

8. Iran - 39.2%

9. Ethiopia - 34.2%

10. Angola - 26.57%

11. Yemen - 30.61%

12. Libya - 21.1%

13. Turkey - 19.9%

14. Nigeria - 15.99%

15. Haiti - 10.91%

16. Brazil - 10.7%

17. Belarus - 10.5%

18. Russia - 8.1%

19. Poland - 6.8%

20. Mexico - 6.2%

21. Spain - 5.6%

22. Ireland - 5.1%

Sri Lanka is the first country that we will be working with to integrate our platform for their economy, invest in the infrastructure with the reserve currency, and allocate our stock as our investment in the market in the country. Creating a sidechain that is paired to our main chain where we digitize the Sri Lankan economy and give it the mobility it needs to compete with international markets at scale and be competitive with the assets that it can offer to the global economy. Our platform would become the platform of trade of digital securities and assets that are listed from the country to the global market bringing new trade flows for goods and products within the country through our platform to the market.

Key Initiatives for Sri Lankan that Obsidian Capital will be incorporating:

- Investing the reserve currency of our market into the Sri Lankan economy creates a fresh flow of liquidity for the government that is low on capital reserves in the Central Bank of Sri Lanka. This would be the first step that we do for any of the aforementioned countries that are on our list to work with and bring our system to.
- Allocation of the currency for the entire worth of the country's assets that will be listed on our platform and denominated in to be sold to the global economy creating new trade flows on goods, products, and services from companies and communities of Sri Lanka directly from the platform.
- Digitization of all assets and pricing them outside of the inflation-ridden environment where the assets are undervalued in comparison to assets that are priced out of their environments. This will be a key way for us to drive investments back into the country as we are listing assets on our platform outside of their economy to grow their price valuation and sell it to countries that do not have the products that are in question.
- Creating the Sri Lankan Side Chain where the economy can grow digitally with each person in the country becoming a “beam” as stated before creating a financial network that is independent of the global economy. Does not require the usage of banks or financial intermediaries as the citizens can sell, buy, trade, and transact with whoever is on the network that they are connected with at scale.
- Labor digitization creates a new stream of jobs for the economy by creating tests and tasks for the maintenance of the network and in turn the companies and providers that are part of the new economy of Sri Lanka. Directly curbing job instability and bringing organization of sectors to those that provide value to the economy directly holding accountability of market sectors and segments operated by market players.

Data Architecture

Data applications are applications that leverage data to bring value to the consumer, being able to trade securities across emerging markets that represent undervalued markets and the transaction of natural assets from countries into international markets - Objective

It should be a well-developed data platform for this information that developers can then create

services and features around it. Snowflake Data Cloud is a good source to use to pull shared data that is relative to our financial terminal/platform.

SQL will be important to query data properly on the platform so that machine learning models can accurately predict and relay information. The Bloomberg API integrated for data with Snowflake Cloud Data will help us gain information on markets and scale the platform as needed by using reliable cloud-based infrastructure in the architecture of the platform.

It is strategically sound to focus the platform on being a cloud-first data solution and architecture. Elasticity, cloud-first will process data and workloads easier than cloud-hosted and will be able to save significantly for us. Availability, we may need more than one cloud-first integration from AWS, Google, Microsoft, and Snowflake. Client type - retail, SME, and corporations. B2c & B2b.

The elasticity of cloud-first environments enables compute and storage to grow or shrink on demand, a benefit best realized by keeping these systems decoupled. Need to see how to keep systems online even when reliability fails. What mechanism can be used to keep the system online?

Reliability: data platforms and applications that come from them must be designed defensively for these issues to avoid unplanned downtimes that result in the loss of revenue. Modern data platforms should be able to do repairs across systems automatically within the need of calling in manual repairs.

Extensibility: UDF- functions that are defined by the users in Python, Java, or JavaScript.

- a cloud-first platform that is cloud platform agnostic

- Support for semi-structured and structured data

- ACID guarantees and SQL support

- Separation of storage and compute

- Data sharing without copying data

- Workload isolation

- Extensibility

- Robust disaster recovery and resiliency mechanisms

There are three measures of the system to take into consideration when developing.

Storage, computation, and security.

Over-provisioning: we should plan sheets for 10,000 to 100,000 informed and active users in the terminal trading products, goods, and securities across market sectors in emerging markets.

Design patterns when it comes to security:

- Access control: two primary times role-based (RBAC) and discretionary (DAC)

- DAC is where object owners can grant access to other users at their discretion. - may help with creating the DSR, Digital Securities Reserve for investment purposes and services to put our representatives in motion across markets and countries at scale.

- Incorporate privacy as a central theme within the platform. Outsiders cannot see the movement or activity of the users within the system. It is a closed economy that interacts on its independence but also with the global economy to make trades.

- Orion Digital Wallet, the app that connects all infrastructure systems for operation will be owned by the company but the system admins (Operators) will be me.

Auditing: we have to comply with the GDPR to sell the platform to companies across Europe.

Closed Circuit Terminal, only members have access to contracts and investments being sold of the investments or goods.

HADES GENERAL PURPOSE CODING SCRIPTS

```
import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

import numpy as np

from statsmodels.formula.api import ols

!pip install ccxt

import ccxt as ccxt
```

Requirement already satisfied: ccxt in c:\users\gideo\anaconda3\lib\site-packages (3.0.103)

Requirement already satisfied: aiohttp>=3.8 in c:\users\gideo\anaconda3\lib\site-packages (from ccxt) (3.8.4)

Requirement already satisfied: setuptools>=60.9.0 in c:\users\gideo\anaconda3\lib\site-packages (from ccxt) (65.6.3)

Requirement already satisfied: aiodns>=1.1.1 in c:\users\gideo\anaconda3\lib\site-packages (from ccxt) (3.0.0)

Requirement already satisfied: requests>=2.18.4 in c:\users\gideo\anaconda3\lib\site-packages (from ccxt) (2.28.1)

Requirement already satisfied: yarl>=1.7.2 in c:\users\gideo\anaconda3\lib\site-packages (from ccxt) (1.9.2)

Requirement already satisfied: certifi>=2018.1.18 in c:\users\gideo\anaconda3\lib\site-packages (from ccxt) (2022.12.7)

Requirement already satisfied: cryptography>=2.6.1 in c:\users\gideo\anaconda3\lib\site-packages (from ccxt) (39.0.1)

Requirement already satisfied: pycares>=4.0.0 in c:\users\gideo\anaconda3\lib\site-packages (from aiodns>=1.1.1->ccxt) (4.3.0)

Requirement already satisfied: aiosignal>=1.1.2 in c:\users\gideo\anaconda3\lib\site-packages (from aiohttp>=3.8->ccxt) (1.3.1)

Requirement already satisfied: async-timeout<5.0,>=4.0.0a3 in c:\users\gideo\anaconda3\lib\site-packages (from aiohttp>=3.8->ccxt) (4.0.2)

Requirement already satisfied: multidict<7.0,>=4.5 in c:\users\gideo\anaconda3\lib\site-packages (from aiohttp>=3.8->ccxt) (6.0.4)

Requirement already satisfied: attrs>=17.3.0 in c:\users\gideo\anaconda3\lib\site-packages (from aiohttp>=3.8->ccxt) (22.1.0)

Requirement already satisfied: frozenlist>=1.1.1 in c:\users\gideo\anaconda3\lib\site-packages (from aiohttp>=3.8->ccxt) (1.3.3)

Requirement already satisfied: charset-normalizer<4.0,>=2.0 in c:\users\gideo\anaconda3\lib\site-packages (from aiohttp>=3.8->ccxt) (2.0.4)

Requirement already satisfied: cffi>=1.12 in c:\users\gideo\anaconda3\lib\site-packages (from cryptography>=2.6.1->ccxt) (1.15.1)

Requirement already satisfied: idna<4,>=2.5 in c:\users\gideo\anaconda3\lib\site-packages (from

requests>=2.18.4->ccxt) (3.4)

Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\gideo\anaconda3\lib\site-packages (from requests>=2.18.4->ccxt) (1.26.14)

Requirement already satisfied: pycparser in c:\users\gideo\anaconda3\lib\site-packages (from cffi>=1.12->cryptography>=2.6.1->ccxt) (2.21)

In [3]:

```
import pandas as pd

MarketDf = pd.read_csv("TotalMarketCSV File.csv")
```

In [4]:

```
MarketDf.shape

MarketDf.info
```

Out[4]:

```
<bound method DataFrame.info of       Unnamed: 0  24h_volume_usd  available_supply              id
\
0              0    9.007640e+09      1.672352e+07             bitcoin
1              1    1.551330e+09      9.616537e+07            ethereum
2              2    1.111350e+09      1.684044e+07         bitcoin-cash
3              3    2.936090e+09      2.779530e+09                iota
4              4    2.315050e+08      3.873915e+10              ripple
...          ...             ...               ...                 ...
1321        1321             NaN               NaN           turbocoin
1322        1322             NaN               NaN               birds
1323        1323             NaN               NaN  bitcoincashscrypt
1324        1324             NaN               NaN           swisscoin
1325        1325             NaN               NaN           faceblock
```

	last_updated	market_cap_usd	max_supply	name
0	1512549554	2.130490e+11	2.100000e+07	Bitcoin
1	1512549553	4.352945e+10	NaN	Ethereum
2	1512549578	2.529585e+10	2.100000e+07	Bitcoin Cash
3	1512549571	1.475225e+10	2.779530e+09	IOTA
4	1512549541	9.365343e+09	1.000000e+11	Ripple
...	...	...	...	...
1321	1512368664	NaN	NaN	TurboCoin
1322	1512535772	NaN	NaN	Birds
1323	1512548078	NaN	NaN	BitcoinCashScrypt
1324	1512540278	NaN	NaN	Swisscoin
1325	1512435283	NaN	NaN	Faceblock

	percent_change_1h	percent_change_24h	percent_change_7d	price_btc
0	0.12	7.33	17.45	1.000000e+00
1	-0.18	-3.93	-7.33	3.617670e-02
2	1.65	-5.51	-4.75	1.200500e-01
3	-2.38	83.35	255.82	4.241800e-04
4	0.56	-3.70	-14.79	1.930000e-05
...	...	...	...	...
1321	NaN	NaN	8.12	1.000000e-08
1322	NaN	10.62	-42.10	1.000000e-08
1323	-0.37	-37.39	-27.69	5.000000e-07
1324	NaN	4.39	-22.84	1.000000e-08
1325	NaN	NaN	-6.83	1.400000e-07

	price_usd	rank	symbol	total_supply
0	12739.500000	1	BTC	1.672352e+07
1	452.652000	2	ETH	9.616537e+07
2	1502.090000	3	BCH	1.684044e+07
3	5.307460	4	MIOTA	2.779530e+09
4	0.241754	5	XRP	9.999309e+10
...	...	...	...	...
1321	0.000114	1322	TURBO	NaN
1322	0.000122	1323	BIRDS	NaN
1323	0.006202	1324	BCCS	2.502380e+06
1324	0.000123	1325	SIC	1.020000e+10
1325	0.001654	1326	FBL	1.000000e+07

[1326 rows x 16 columns]>

In [5]:

```
MarketDf.isnull().sum()
```

Out[5]:

```
Unnamed: 0              0
24h_volume_usd         56
available_supply      295
id                      0
last_updated            0
market_cap_usd        295
max_supply           1111
name                    0
```

percent_change_1h 53

percent_change_24h 56

percent_change_7d 43

price_btc 0

price_usd 0

rank 0

symbol 0

total_supply 115

dtype: int64

In [6]:

```
d1 = MarketDf.dropna()
print(d1)
```

```
     Unnamed: 0  24h_volume_usd  available_supply               id  \
0             0    9.007640e+09      1.672352e+07          bitcoin
2             2    1.111350e+09      1.684044e+07     bitcoin-cash
3             3    2.936090e+09      2.779530e+09             iota
4             4    2.315050e+08      3.873915e+10           ripple
5             5    2.289430e+08      7.736420e+06             dash
...         ...             ...               ...              ...
1020       1020    6.909060e+01      1.662700e+04           enigma
1021       1021    1.300720e+02      1.994860e+05           ulatech
1024       1024    2.231020e+02      4.213000e+05  digital-money-bits
1028       1028    4.271860e+01      8.714000e+04         caliphcoin
1030       1030    1.718820e+02      7.924200e+04      applecoin-apw
```

	last_updated	market_cap_usd	max_supply	name	\
0	1512549554	2.130490e+11	2.100000e+07	Bitcoin	
2	1512549578	2.529585e+10	2.100000e+07	Bitcoin Cash	
3	1512549571	1.475225e+10	2.779530e+09	IOTA	
4	1512549541	9.365343e+09	1.000000e+11	Ripple	
5	1512549542	5.794076e+09	1.890000e+07	Dash	
...	...	...	...	...	
1020	1512549554	2.303000e+03	5.000000e+06	Enigma	
1021	1512549576	2.302000e+03	9.000000e+07	Ulatech	
1024	1512549567	5.150000e+02	5.000000e+07	Digital Money Bits	
1028	1512549561	1.000000e+02	1.890000e+08	CaliphCoin	
1030	1512549575	1.000000e+01	2.100000e+09	AppleCoin	

	percent_change_1h	percent_change_24h	percent_change_7d	price_btc	\
0	0.12	7.33	17.45	1.000000e+00	
2	1.65	-5.51	-4.75	1.200500e-01	
3	-2.38	83.35	255.82	4.241800e-04	
4	0.56	-3.70	-14.79	1.930000e-05	
5	1.22	-3.31	10.64	5.985610e-02	
...	...	...	...	...	
1020	0.54	-20.54	-23.56	1.110000e-05	
1021	0.51	29.06	-64.39	9.200000e-07	
1024	0.56	3.81	13.28	1.000000e-07	
1028	0.54	53.88	67.27	9.000000e-08	
1030	0.54	6.27	17.44	1.000000e-08	

	price_usd	rank	symbol	total_supply
0	12739.500000	1	BTC	1.672352e+07
2	1502.090000	3	BCH	1.684044e+07
3	5.307460	4	MIOTA	2.779530e+09
4	0.241754	5	XRP	9.999309e+10
5	748.935000	6	DASH	7.736420e+06
...	...	...	...	...
1020	0.138511	1021	XNG	6.662700e+04
1021	0.011538	1022	ULA	5.419949e+07
1024	0.001222	1025	DMB	1.042130e+07
1028	0.001145	1029	CALC	1.485871e+08
1030	0.000125	1031	APW	1.600079e+09

[186 rows x 16 columns]

In [7]:

```
d1.isnull().sum()
```

Out[7]:

```
Unnamed: 0            0
24h_volume_usd        0
available_supply      0
id                    0
last_updated          0
market_cap_usd        0
max_supply            0
```

```
name                0
percent_change_1h   0
percent_change_24h  0
percent_change_7d   0
price_btc           0
price_usd           0
rank                0
symbol              0
total_supply        0
dtype: int64
```

In [8]:

```
df = pd.DataFrame(d1)
print(df)
```

```
     Unnamed: 0  24h_volume_usd  available_supply               id  \
0             0    9.007640e+09      1.672352e+07          bitcoin
2             2    1.111350e+09      1.684044e+07     bitcoin-cash
3             3    2.936090e+09      2.779530e+09             iota
4             4    2.315050e+08      3.873915e+10           ripple
5             5    2.289430e+08      7.736420e+06             dash
...         ...             ...               ...              ...
1020       1020    6.909060e+01      1.662700e+04           enigma
1021       1021    1.300720e+02      1.994860e+05          ulatech
1024       1024    2.231020e+02      4.213000e+05  digital-money-bits
1028       1028    4.271860e+01      8.714000e+04        caliphcoin
1030       1030    1.718820e+02      7.924200e+04     applecoin-apw
```

	last_updated	market_cap_usd	max_supply	name
0	1512549554	2.130490e+11	2.100000e+07	Bitcoin
2	1512549578	2.529585e+10	2.100000e+07	Bitcoin Cash
3	1512549571	1.475225e+10	2.779530e+09	IOTA
4	1512549541	9.365343e+09	1.000000e+11	Ripple
5	1512549542	5.794076e+09	1.890000e+07	Dash
...	...	...	...	...
1020	1512549554	2.303000e+03	5.000000e+06	Enigma
1021	1512549576	2.302000e+03	9.000000e+07	Ulatech
1024	1512549567	5.150000e+02	5.000000e+07	Digital Money Bits
1028	1512549561	1.000000e+02	1.890000e+08	CaliphCoin
1030	1512549575	1.000000e+01	2.100000e+09	AppleCoin

	percent_change_1h	percent_change_24h	percent_change_7d	price_btc
0	0.12	7.33	17.45	1.000000e+00
2	1.65	-5.51	-4.75	1.200500e-01
3	-2.38	83.35	255.82	4.241800e-04
4	0.56	-3.70	-14.79	1.930000e-05
5	1.22	-3.31	10.64	5.985610e-02
...	...	...	...	...
1020	0.54	-20.54	-23.56	1.110000e-05
1021	0.51	29.06	-64.39	9.200000e-07
1024	0.56	3.81	13.28	1.000000e-07
1028	0.54	53.88	67.27	9.000000e-08

1030 0.54 6.27 17.44 1.000000e-08

	price_usd	rank	symbol	total_supply
0	12739.500000	1	BTC	1.672352e+07
2	1502.090000	3	BCH	1.684044e+07
3	5.307460	4	MIOTA	2.779530e+09
4	0.241754	5	XRP	9.999309e+10
5	748.935000	6	DASH	7.736420e+06
...	...	...	...	...
1020	0.138511	1021	XNG	6.662700e+04
1021	0.011538	1022	ULA	5.419949e+07
1024	0.001222	1025	DMB	1.042130e+07
1028	0.001145	1029	CALC	1.485871e+08
1030	0.000125	1031	APW	1.600079e+09

[186 rows x 16 columns]

In [9]:

```
df_clean = df[['24h_volume_usd', 'available_supply',
               'last_updated', 'market_cap_usd', 'max_supply',
               'name','percent_change_1h', 'percent_change_24h',
               'percent_change_7d','price_btc','price_usd', 'symbol',
               'total_supply']]
df_clean.head()
```

Out[9]:

	24h_volume_usd	available_supply	last_updated	market_cap_usd	max_supply	name	percent_change_1h	percent_change_24h	percent_change_7d	price_btc	price_usd	symbol	total_supply
0	9.007640e+09	1.672352e+07	1512549554	2.130490e+11	2.100000e+07	Bitcoin	0.12	7.33	17.45	1.000000	12739.500000	BTC	1.672352e+07
2	1.111350e+09	1.684044e+07	1512549578	2.529585e+10	2.100000e+07	BitcoinCash	1.65	-5.51	-4.75	0.120050	1502.090000	BCH	1.684044e+07
3	2.936090e+09	2.779530e+09	1512549571	1.475225e+10	2.779530e+09	IOTA	-2.38	83.35	255.82	0.000424	5.307460	MIOTA	2.779530e+09
4	2.315050e+08	3.873915e+10	1512549541	9.365343e+09	1.000000e+11	Ripple	0.56	-3.70	-14.79	0.000019	0.241754	XRP	9.999309e+10
5	2.289430e+08	7.736420e+06	1512549542	5.794076e+09	1.890000e+07	Dash	1.22	-3.31	10.64	0.059856	748.935000	DASH	7.736420e+06

In [10]:

```
import pandas as pd

#Read the dataframe

df =  df_clean


#Group the data by column "market_cap_usd" and compute the mean of the column "price_btc" for each group

df_grouped = df.groupby("price_btc")["market_cap_usd"].mean()

df_aggregated = df.groupby("price_usd").agg({"percent_change_1h":["mean", "std"],

                                        "percent_change_24h":["mean", "std"],

                                         "percent_change_7d":["mean", "std"]})

#Group the data by column "price_usd" and apply a custom function to the groups

def custom_function(x):

    return x.max() - x.min()

df_percent = df.groupby("price_usd").agg({"percent_change_1h": custom_function,

                                        "percent_change_24h": custom_function,

                                        "percent_change_7d" : custom_function,})

#Compute the rolling average of a data frame with a window the size of 2

df_rolling = df.rolling(window=2).mean()

print(df_rolling)
```

	24h_volume_usd	available_supply	last_updated	market_cap_usd	\
0	NaN	NaN	NaN	NaN	
2	5.059495e+09	1.678198e+07	1.512550e+09	1.191724e+11	
3	2.023720e+09	1.398185e+09	1.512550e+09	2.002405e+10	
4	1.583798e+09	2.075934e+10	1.512550e+09	1.205879e+10	
5	2.302240e+08	1.937344e+10	1.512550e+09	7.579709e+09	

...	...	...	...	...
1020	4.873285e+01	4.591662e+06	1.512550e+09	5.842500e+03
1021	9.958130e+01	1.080565e+05	1.512550e+09	2.302500e+03
1024	1.765870e+02	3.103930e+05	1.512550e+09	1.408500e+03
1028	1.329103e+02	2.542200e+05	1.512550e+09	3.075000e+02
1030	1.073003e+02	8.319100e+04	1.512550e+09	5.500000e+01

	max_supply	percent_change_1h	percent_change_24h	percent_change_7d
0	NaN	NaN	NaN	NaN
2	2.100000e+07	0.885	0.910	6.350
3	1.400265e+09	-0.365	38.920	125.535
4	5.138977e+10	-0.910	39.825	120.515
5	5.000945e+10	0.890	-3.505	-2.075
...	...	...	...	...
1020	1.250000e+07	-0.750	-27.950	31.445
1021	4.750000e+07	0.525	4.260	-43.975
1024	7.000000e+07	0.535	16.435	-25.555
1028	1.195000e+08	0.550	28.845	40.275
1030	1.144500e+09	0.540	30.075	42.355

	price_btc	price_usd	total_supply
0	NaN	NaN	NaN
2	5.600250e-01	7120.795000	1.678198e+07
3	6.023709e-02	753.698730	1.398185e+09
4	2.217400e-04	2.774607	5.138631e+10

5 2.993770e-02 374.588377 5.000042e+10

... ...

1020 5.590000e-06 0.069767 4.716662e+06

1021 6.010000e-06 0.075024 2.713306e+07

1024 5.100000e-07 0.006380 3.231039e+07

1028 9.500000e-08 0.001184 7.950422e+07

1030 5.000000e-08 0.000635 8.743332e+08

[186 rows x 11 columns]

C:\Users\gideo\AppData\Local\Temp\ipykernel_5992\298900277.py:17: FutureWarning: Dropping of nuisance columns in rolling operations is deprecated; in a future version this will raise TypeError. Select only valid columns before calling the operation. Dropped columns were Index(['name', 'symbol'], dtype='object')

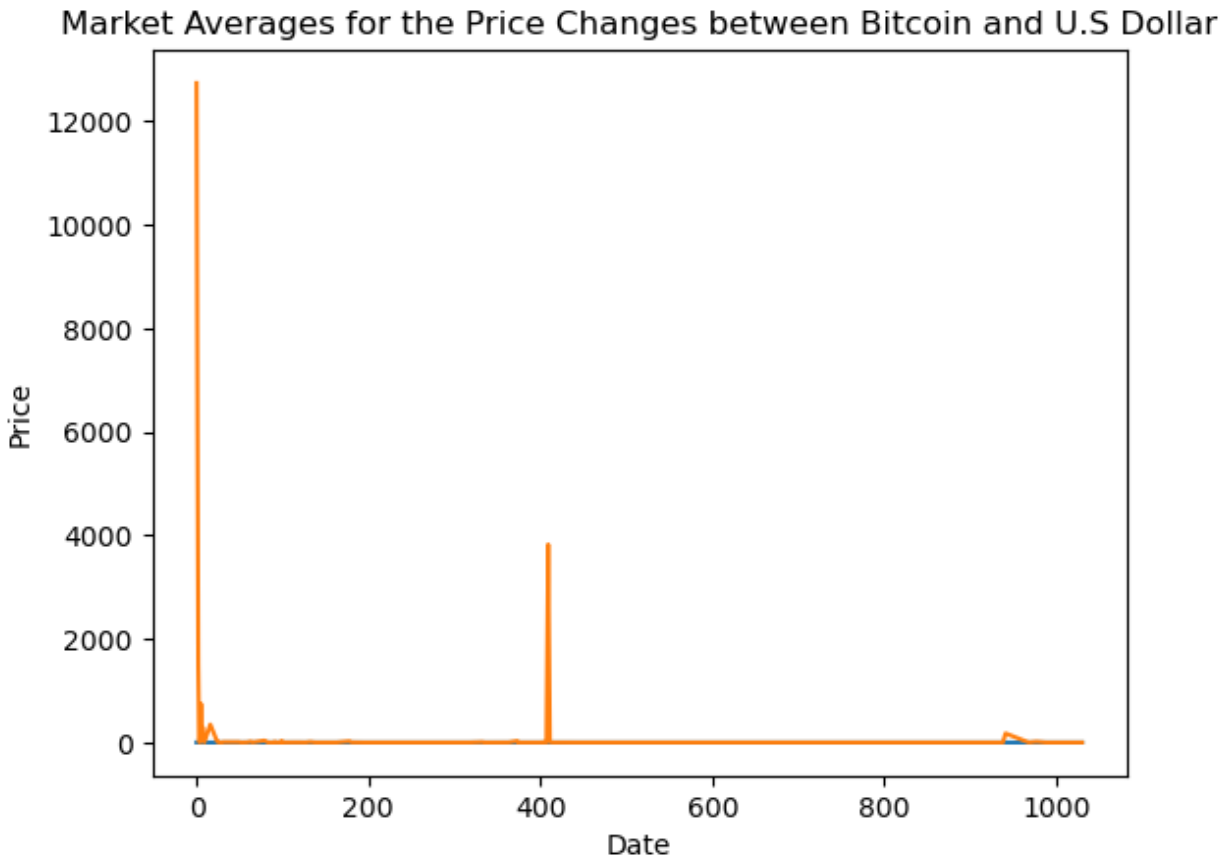
```
df_rolling = df.rolling(window=2).mean()
```

In [11]:

```
import matplotlib.pyplot as plt


#Plot the data

#plt.plot(df_clean.index, df_clean[["price_btc", "price_usd"]])

df["price_btc"].plot()

df["price_usd"].plot()


#Add labels and titles

plt.xlabel("Date")

plt.ylabel("Price")

plt.title("Market Averages for the Price Changes between Bitcoin and U.S Dollar")
```

#Show the plot

plt.show()



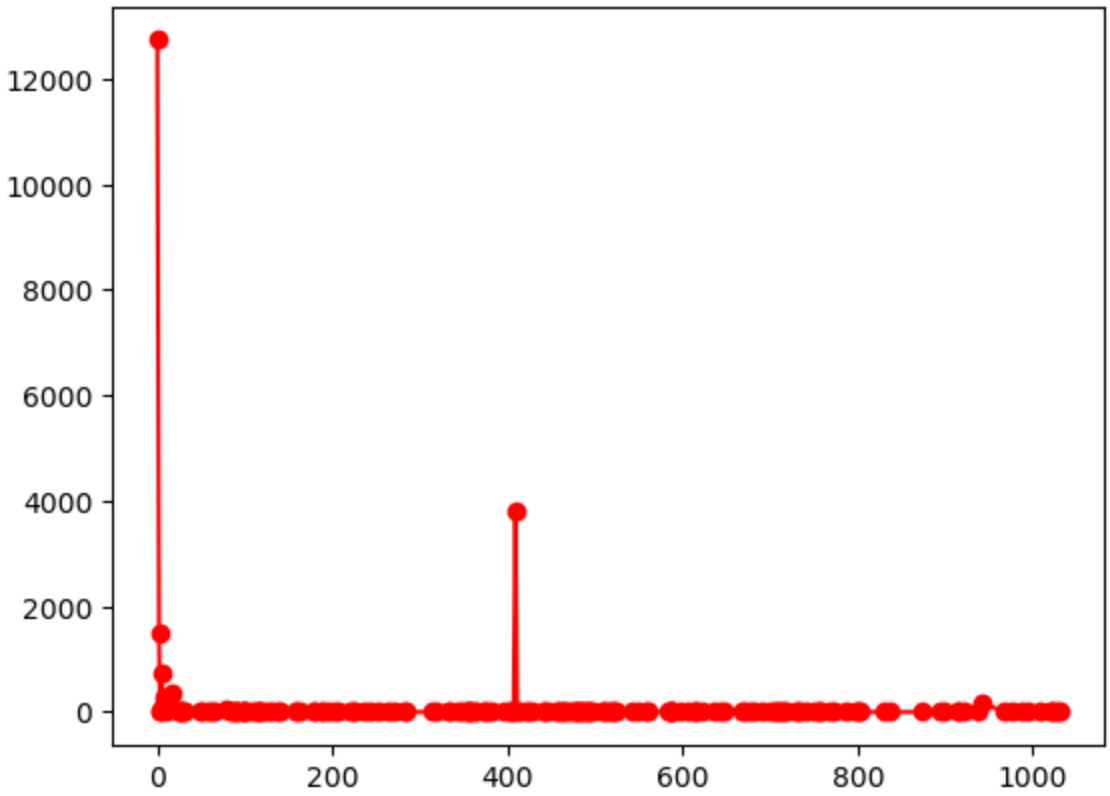
In [12]:

#Plot the data with red circles as markers

plt.plot(df_clean.index, df_clean["price_usd"], "r-o")

Out[12]:

[<matplotlib.lines.Line2D at 0x17091890790>]

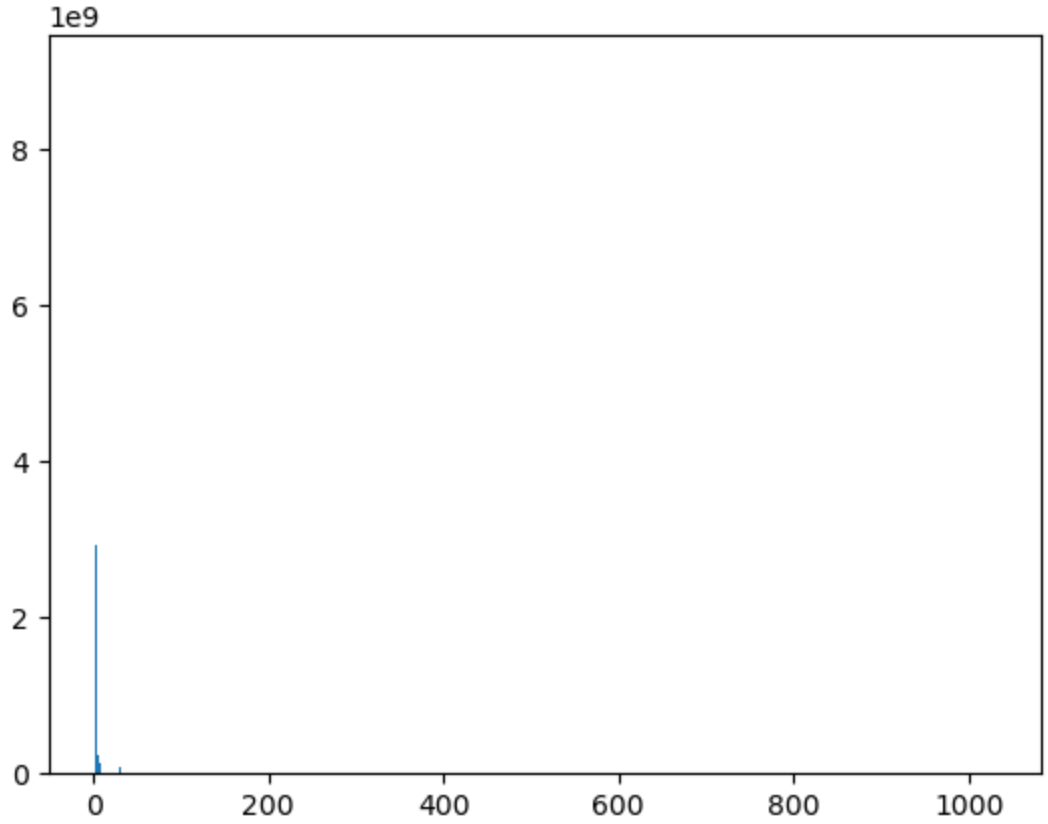


In [13]:

```
plt.bar(df_clean.index, df_clean["24h_volume_usd"])
```

Out[13]:

```
<BarContainer object of 186 artists>
```

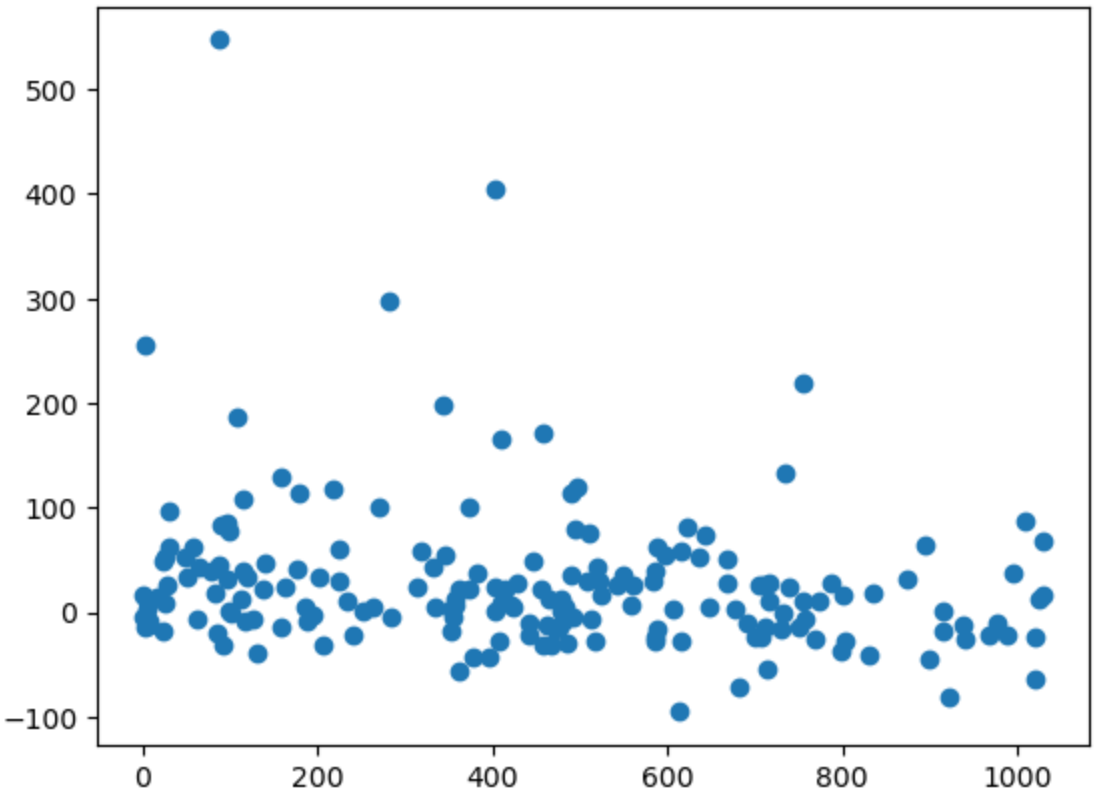


In [14]:

```
plt.scatter(df_clean.index, df_clean["percent_change_7d"])
```

Out[14]:

```
<matplotlib.collections.PathCollection at 0x17091b3c2b0>
```



In [25]:

```
class Portfolio:

    def __init__(self):

        self.df_clean = {"XMR": 100, "ADA": 50, "GRT": 200}  # Dictionary to hold stocks in the Portfolio


    def add_stock(self, symbol, shares):

        self.df_clean[symbol] = shares


    def remove_stock(self, symbol):

        del self.df_clean[symbol]


    def get_stocks(self):

        return list(self.df_clean.keys())
```

```
    def get_shares(self, symbol):
        return self.df_clean[symbol]

# Create a new portfolio
p = Portfolio()

# Add some stocks to the portfolio
p.add_stock("XMR", 100)
p.add_stock("ADA", 50)
p.add_stock("GRT", 200)

# Print the stocks in the portfolio
print(p.get_stocks())

# Output: ['XMR', 'ADA', 'GRT']
```

['XMR', 'ADA', 'GRT']

In [ ]:

In [16]:

```
import pandas as pd

def liquidity(data):
    #Calculate the liquidity as the product of the volume and the inverse of the spread
```

```
    data["Liquidity"] = df_clean["24h_volume_usd"] / (data["available_supply"] - data["price_usd"])
    return data


#Load the market data into a data frame
df = pd.read_csv("TotalMarketCSV File.csv")


#Calculate the liquidity of the stocks
df = liquidity(df)


#Print the liqudity of stocks


print(df["Liquidity"])


0       539.031514
1              NaN
2        65.998826
3         1.056326
4         0.005976
           ...
1321           NaN
1322           NaN
1323           NaN
1324           NaN
1325           NaN
Name: Liquidity, Length: 1326, dtype: float64
```

In [17]:

```
!pip install -U scikit-learn
```

```
Requirement already satisfied: scikit-learn in c:\users\gideo\anaconda3\lib\site-packages (1.2.2)

Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\gideo\anaconda3\lib\site-packages
(from scikit-learn) (2.2.0)

Requirement already satisfied: scipy>=1.3.2 in c:\users\gideo\anaconda3\lib\site-packages (from
scikit-learn) (1.10.0)

Requirement already satisfied: joblib>=1.1.1 in c:\users\gideo\anaconda3\lib\site-packages (from
scikit-learn) (1.1.1)

Requirement already satisfied: numpy>=1.17.3 in c:\users\gideo\anaconda3\lib\site-packages (from
scikit-learn) (1.23.5)
```

In [ ]:

In [19]:

```
import pandas as pd

from sklearn.ensemble import RandomForestRegressor

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import StandardScaler


# Load the crypto data into a data frame Percentile Changes

df = df_clean


# Split the data into features and labels

X = df[["percent_change_1h", "percent_change_7d", "price_usd"]]

y = df["24h_volume_usd"]
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.33, random_state=42)


scaler = StandardScaler()

X_train = scaler.fit_transform(X_train)

X_test = scaler.transform(X_test)


# Create a random forest regressor model

model = RandomForestRegressor()


# Train the model

model.fit(X_train, y_train)


# Make predictions

predictions = model.predict(X_test)


# Evaluate the model

score = model.score(X_test, y_test)

print("R^2 score:", score)
```

```
R^2 score: -0.021597084137308986
```

In [20]:

```
import sklearn

print(dir(sklearn.preprocessing))
```

```
['Binarizer', 'FunctionTransformer', 'KBinsDiscretizer', 'KernelCenterer', 'LabelBinarizer',
'LabelEncoder', 'MaxAbsScaler', 'MinMaxScaler', 'MultiLabelBinarizer', 'Normalizer',
'OneHotEncoder', 'OrdinalEncoder', 'PolynomialFeatures', 'PowerTransformer',
```

'QuantileTransformer', 'RobustScaler', 'SplineTransformer', 'StandardScaler', '__all__', '__builtins__', '__cached__', '__doc__', '__file__', '__loader__', '__name__', '__package__', '__path__', '__spec__', '_csr_polynomial_expansion', '_data', '_discretization', '_encoders', '_function_transformer', '_label', '_polynomial', 'add_dummy_feature', 'binarize', 'label_binarize', 'maxabs_scale', 'minmax_scale', 'normalize', 'power_transform', 'quantile_transform', 'robust_scale', 'scale']

In [21]:

```
#Train a random forest regressor on the data

regr = RandomForestRegressor()

regr.fit(X_train, y_train)


#Make predictions on the crypto data

predictions = regr.predict(X_test)

print(predictions)
```

```
[2.92092282e+07 3.63042365e+05 9.05697251e+05 1.61333921e+06
 4.83420879e+06 1.60684090e+06 4.46834516e+07 2.22354994e+07
 9.46437696e+05 1.23606476e+07 2.12824086e+07 4.15187654e+07
 1.25518060e+07 1.64965393e+05 6.46969332e+06 2.70515196e+06
 5.76563542e+06 5.30376476e+07 1.41657122e+06 6.36276060e+07
 2.82767252e+05 1.10774423e+06 1.41516079e+05 3.51515046e+06
 6.54874894e+03 5.63896315e+07 1.32924065e+05 1.27030515e+05
 2.46780237e+06 4.64039237e+06 3.74964002e+03 3.24519816e+06
 9.88491027e+06 1.69222130e+06 8.71353311e+06 4.77758819e+06
 2.46420533e+06 6.92721805e+06 1.27803210e+07 2.29163238e+07
 1.49869462e+06 5.27749185e+06 1.85894602e+07 7.73408612e+06
 1.62660379e+07 1.77693961e+06 1.14739709e+06 8.89713081e+05
 9.63606030e+05 4.65595809e+06 3.60373327e+06 1.81840525e+05
```

6.46407830e+05 1.33629588e+06 1.48634817e+06 1.20623443e+06

5.81677356e+07 2.47169168e+06 1.28635917e+07 8.32871046e+06

4.16728070e+03 4.59734121e+07]

In [ ]:

In [ ]:

In [36]:

```
import pandas as pd
import numpy as np
from scipy.optimize import minimize

# Load the asset data into a data frame for Portfolio Management
df = pd.read_csv("TotalMarketCSV File.csv")

# Calculate the returns of the assets
df["Return"] = df["price_usd"].pct_change()

# Print the mean and standard deviation of the returns
print("Mean return:", df["Return"].mean())
print("Standard deviation:", df["Return"].std())

# Calculate the sharpe ratio of assets
sharpe_ratio = df["Return"].mean() / df["Return"].std()
print("Sharpe ratio:", sharpe_ratio)
```

```
# Calculate the correlations between the assets
correlations = df.corr()
print("Asset correlations")
print(correlations)

# Optimize the portfolio using the Markowitz Model
returns = df["Return"].values
covariance = df["Return"].cov()

# Set the target return and the constraints
target_return = 0.1
constraints = ({"type": "eq", "fun": lambda x: np.sum(x) - 1})

# Define the bounds for the optimization
num_assets = len(df.columns) - 1
bnds = tuple((0, 1) for _ in range(num_assets))

# Minimize the portfolio variance
portfolio_variance = lambda x: np.dot(x, np.dot(covariance, x))
result = minimize(portfolio_variance, num_assets * [1. / num_assets],
                  method="SLSQP", bounds=bnds, constraints=constraints)
optimal_weights = result["x"]
print("Optimal weights:", optimal_weights)
```

Mean return: 62554.78822312827

Standard deviation: 1340582.0090027824

Sharpe ratio: 0.046662410656742176

Asset correlations

	Unnamed: 0	24h_volume_usd	available_supply	\
Unnamed: 0	1.000000	-0.100141	-0.025381	
24h_volume_usd	-0.100141	1.000000	-0.004357	
available_supply	-0.025381	-0.004357	1.000000	
last_updated	-0.152718	-0.022068	0.007836	
market_cap_usd	-0.090380	0.962901	-0.003039	
max_supply	-0.037318	-0.007268	-0.005606	
percent_change_1h	0.013322	-0.005457	-0.015611	
percent_change_24h	0.011651	0.011719	0.058239	
percent_change_7d	-0.067939	0.011986	-0.012988	
price_btc	0.000746	0.027201	-0.003090	
price_usd	0.000719	0.027724	-0.003091	
rank	1.000000	-0.100141	-0.025381	
total_supply	0.014050	-0.001876	0.005808	
Return	0.024500	-0.004185	-0.003674	

	last_updated	market_cap_usd	max_supply	\
Unnamed: 0	-0.152718	-0.090380	-0.037318	
24h_volume_usd	-0.022068	0.962901	-0.007268	
available_supply	0.007836	-0.003039	-0.005606	
last_updated	1.000000	0.005636	0.006685	

market_cap_usd	0.005636	1.000000	-0.007026
max_supply	0.006685	-0.007026	1.000000
percent_change_1h	0.006027	-0.004621	-0.011710
percent_change_24h	0.035458	-0.002080	-0.005211
percent_change_7d	0.043327	-0.003773	-0.017526
price_btc	0.004741	0.028980	-0.007712
price_usd	0.004743	0.029531	-0.007669
rank	-0.152718	-0.090380	-0.037318
total_supply	0.002915	-0.001687	-0.007547
Return	0.005722	-0.004436	-0.006537

	percent_change_1h	percent_change_24h	percent_change_7d \
Unnamed: 0	0.013322	0.011651	-0.067939
24h_volume_usd	-0.005457	0.011719	0.011986
available_supply	-0.015611	0.058239	-0.012988
last_updated	0.006027	0.035458	0.043327
market_cap_usd	-0.004621	-0.002080	-0.003773
max_supply	-0.011710	-0.005211	-0.017526
percent_change_1h	1.000000	0.202567	0.083884
percent_change_24h	0.202567	1.000000	0.405632
percent_change_7d	0.083884	0.405632	1.000000
price_btc	0.010067	-0.001576	-0.003722
price_usd	0.010066	-0.001577	-0.003729
rank	0.013322	0.011651	-0.067939
total_supply	-0.001307	-0.001364	-0.002777

```
Return                    0.013191          0.013537         -0.011494
```

```
                  price_btc  price_usd      rank  total_supply     Return
Unnamed: 0         0.000746   0.000719  1.000000      0.014050  0.024500
24h_volume_usd     0.027201   0.027724 -0.100141     -0.001876 -0.004185
available_supply  -0.003090  -0.003091 -0.025381      0.005808 -0.003674
last_updated       0.004741   0.004743 -0.152718      0.002915  0.005722
market_cap_usd     0.028980   0.029531 -0.090380     -0.001687 -0.004436
max_supply        -0.007712  -0.007669 -0.037318     -0.007547 -0.006537
percent_change_1h  0.010067   0.010066  0.013322     -0.001307  0.013191
percent_change_24h -0.001576  -0.001577  0.011651     -0.001364  0.013537
percent_change_7d  -0.003722  -0.003729 -0.067939     -0.002777 -0.011494
price_btc          1.000000   1.000000  0.000746     -0.001118  0.665772
price_usd          1.000000   1.000000  0.000719     -0.001118  0.665772
rank               0.000746   0.000719  1.000000      0.014050  0.024500
total_supply      -0.001118  -0.001118  0.014050      1.000000 -0.001426
Return             0.665772   0.665772  0.024500     -0.001426  1.000000
```

```
C:\Users\gideo\AppData\Local\Temp\ipykernel_5992\36443125.py:20: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this warning.

  correlations = df.corr()
```

TypeError Traceback (most recent call last)

Cell **In[36], line 26**

```
    24 # Optimize the portfolio using the Markowitz Model
```

```
    25 returns = df["Return"].values
---> 26 covariance = df["Return"].cov()
    28 # Set the target return and the constraints
    29 target_return = 0.1

TypeError: Series.cov() missing 1 required positional argument: 'other'
```

In [ ]:

```
class SwapExchange:
    def __init__(self):
        self.liquidity_pools = {}

    def add_liquidity(self, token_a, token_b, amount_a, amount_b):
        liquidity_pool = {
            'token_a': token_a,
            'token_b': token_b,
            'amount_a': amount_a,
            'amount_b': amount_b
        }
        self.liquidity_pools[token_a + '_' + token_b] = liquidity_pool

    def swap(self, token_a, token_b, amount):
        pool_key = token_a + '_' + token_b
        if pool_key in self.liquidity_pools:
            liquidity_pool = self.liquidity_pools[pool_key]
            if token_a == liquidity_pool['token_a']:
                amount_b = amount * liquidity_pool['amount_b'] / liquidity_pool['amount_a']
```

```
            liquidity_pool['amount_a'] += amount
            liquidity_pool['amount_b'] -= amount_b
            return amount_b
        elif token_a == liquidity_pool['token_b']:
            amount_b = amount * liquidity_pool['amount_a'] / liquidity_pool['amount_b']
            liquidity_pool['amount_b'] += amount
            liquidity_pool['amount_a'] -= amount_b
            return amount_b
    return 0


# Usage example
exchange = SwapExchange()

# Add liquidity to the exchange
exchange.add_liquidity('ETH', 'DAI', 10, 1000)

# Swap ETH for DAI
eth_amount = 2
dai_amount = exchange.swap('ETH', 'DAI', eth_amount)
print(f'Swapped {eth_amount} ETH for {dai_amount} DAI')

# Swap DAI for ETH
dai_amount = 500
eth_amount = exchange.swap('DAI', 'ETH', dai_amount)
```

```
print(f'Swapped {dai_amount} DAI for {eth_amount} ETH')
```

In [ ]:

```
class ShippingCompany:
    def __init__(self, name, address, contact_person, email, phone, financials=None):
        self.name = name
        self.address = address
        self.contact_person = contact_person
        self.email = email
        self.phone = phone
        self.financials = financials or {}
        self.is_onboarded = False

    def onboard(self):
        # Perform onboarding process here (e.g., validate information, create account, etc.)
        # Add your specific onboarding steps and validations

        # Simulating the onboarding process
        if self.validate_information():
            self.create_account()
            self.is_onboarded = True
            print("Shipping company onboarded successfully.")
        else:
            print("Failed to onboard shipping company.")

    def validate_information(self):
```

```
    # Perform validation of company information
    # Add your specific validation logic here
    if self.name and self.address and self.contact_person and self.email and self.phone:
        return True
    else:
        return False

def create_account(self):
    # Perform account creation process
    # Add your specific account creation logic here
    # For example, create an account in the shipping software suite
    print("Account created for the shipping company:", self.name)

def add_financials(self, year, revenue, expenses, net_profit):
    # Add financial data for a specific year
    self.financials[year] = {
        "revenue": revenue,
        "expenses": expenses,
        "net_profit": net_profit
    }

def get_portfolio(self):
    # Return the financial portfolio of the shipping company
    return self.financials
```

```
# Example usage
shipping_company = ShippingCompany(
    name="ABC Shipping",
    address="123 Main Street, City, Country",
    contact_person="John Doe",
    email="john.doe@example.com",
    phone="1234567890"
)

shipping_company.onboard()

# Add financials for the shipping company
shipping_company.add_financials(2021, 1000000, 800000, 200000)
shipping_company.add_financials(2022, 1200000, 900000, 300000)

# Get the financial portfolio of the shipping company
portfolio = shipping_company.get_portfolio()
print("Financial Portfolio:")
for year, financials in portfolio.items():
    print(f"Year: {year}")
    print(f"Revenue: {financials['revenue']}")
    print(f"Expenses: {financials['expenses']}")
    print(f"Net Profit: {financials['net_profit']}")
    print()
```

In [ ]:

```
class Company:
    def __init__(self, name, address, contact_person, email, phone):
        self.name = name
        self.address = address
        self.contact_person = contact_person
        self.email = email
        self.phone = phone
        self.notes = []

    def add_note(self, note):
        self.notes.append(note)


class CRM:
    def __init__(self):
        self.companies = []

    def add_company(self, company):
        self.companies.append(company)

    def search_company(self, keyword):
        results = []
        for company in self.companies:
            if keyword.lower() in company.name.lower():
```

```
            results.append(company)

        return results



# Create CRM instance

crm = CRM()


# Create and add companies

company1 = Company("ABC Shipping", "123 Main Street, City, Country", "John Doe",
"john.doe@example.com", "1234567890")

company2 = Company("XYZ Logistics", "456 Park Avenue, City, Country", "Jane Smith",
"jane.smith@example.com", "9876543210")


crm.add_company(company1)

crm.add_company(company2)


# Search for companies

search_keyword = "shipping"

search_results = crm.search_company(search_keyword)


# Display search results

print(f"Search results for '{search_keyword}':")

for company in search_results:

    print(f"Name: {company.name}")

    print(f"Address: {company.address}")

    print(f"Contact Person: {company.contact_person}")
```

```
    print(f"Email: {company.email}")
    print(f"Phone: {company.phone}")
    print()

# Add notes to a company
company1.add_note("Met with the CEO to discuss investment opportunities.")
company1.add_note("Sent follow-up email regarding compliance documents.")

# Display company notes
print(f"Notes for {company1.name}:")
for note in company1.notes:
    print(note)
```

In [ ]:

```
class BankAccount:
    def __init__(self, account_number, account_holder, balance, currency):
        self.account_number = account_number
        self.account_holder = account_holder
        self.balance = balance
        self.currency = currency

    def deposit(self, amount):
        self.balance += amount

    def withdraw(self, amount):
        if self.balance >= amount:
```

```
        self.balance -= amount
    else:
        print("Insufficient funds")

def clear_check(self, check_amount):
    self.withdraw(check_amount)
    print(f"Check cleared for {check_amount} {self.currency}")

def transfer_funds(self, target_account, amount):
    if self.balance >= amount:
        self.withdraw(amount)
        target_account.deposit(amount)
        print(f"Transferred {amount} {self.currency} to Account {target_account.account_number}")
    else:
        print("Insufficient funds")

def invest_funds(self, investment_amount):
    print(f"Investing {investment_amount} {self.currency} in the stock market")
    # Add the logic to invest the funds in the stock market

def display_balance(self):
    print(f"Account Holder: {self.account_holder}")
    print(f"Account Number: {self.account_number}")
    print(f"Balance: {self.balance} {self.currency}")
```

```
class RiskTool:
    @staticmethod
    def calculate_covariance(asset1_returns, asset2_returns):
        covariance = np.cov(asset1_returns, asset2_returns)[0, 1]
        return covariance

    @staticmethod
    def calculate_variance(returns):
        variance = np.var(returns)
        return variance



# Create two bank accounts
account1 = BankAccount("123456789", "John Doe", 1000.0, "USD")
account2 = BankAccount("987654321", "Jane Smith", 500.0, "USD")

# Perform transactions
account1.display_balance()
account2.display_balance()

account1.deposit(500.0)
account1.display_balance()

account1.withdraw(200.0)
```

```
account1.display_balance()

# Clear a check
check_amount = 300.0
account1.clear_check(check_amount)
account1.display_balance()

# Transfer funds between accounts
transfer_amount = 400.0
account1.transfer_funds(account2, transfer_amount)
account1.display_balance()
account2.display_balance()

# Invest funds in the stock market
investment_amount = 1000.0
account1.invest_funds(investment_amount)

# Calculate covariance and variance
asset1_returns = [0.05, 0.03, 0.02, -0.01, 0.02]
asset2_returns = [0.08, 0.01, 0.03, -0.02, 0.04]

covariance = RiskTool.calculate_covariance(asset1_returns, asset2_returns)
variance = RiskTool.calculate_variance(asset1_returns)

print("Covariance:", covariance)
```

```
print("Variance:", variance)
```

In [32]:

```
import ccxt

# Connect to the exchange for trades
exchange = ccxt.coinbase()

# Set the ticker symbol and the trade amount
symbol = "BTC/USDT"
amount = 0.1

# Check the current price of the asset
ticker = exchange.fetch_ticker("ETH")
print("Current price:", ticker["last"])

# Place a buy order
order = exchange.create_order(symbol, "limit", "buy", amount, ticker["last"])
print("Buy order placed:", order)

# Check the balance of the asset
balance = exchange.fetch_balance()
print("Asset balance:", balance[symbol])

# Place a sell order
order = exchange.create_order(symbol, "limit", "sell", amount, ticker["last"])
```

```
print("Sell order placed:", order)


# Check the balance of the asset again

balance = exchange.fetch_balance()

print("Asset balance:", balance[symbol])
```

AuthenticationError Traceback (most recent call last)

Cell **In[32], line 11**

```
      8 amount = 0.1

     10 # Check the current price of the asset

---> 11 ticker = exchange.fetch_ticker("ETH")

     12 print("Current price:", ticker["last"])

     14 # Place a buy order
```

File **~\anaconda3\lib\site-packages\ccxt\coinbase.py:1248**, in coinbase.fetch_ticker**(self, symbol, params)**

```
   1246 method = self.safe_string(self.options, 'fetchTicker', 'fetchTickerV3')

   1247 if method == 'fetchTickerV3':

-> 1248     return self.fetch_ticker_v3(symbol, params)

   1249 return self.fetch_ticker_v2(symbol, params)
```

File **~\anaconda3\lib\site-packages\ccxt\coinbase.py:1280**, in coinbase.fetch_ticker_v3**(self, symbol, params)**

```
   1279 def fetch_ticker_v3(self, symbol: str, params={}):

-> 1280     self.load_markets()
```

```
1281     market = self.market(symbol)
1282     request = {
1283         'product_id': market['id'],
1284         'limit': 1,
1285     }
```

File **~\anaconda3\lib\site-packages\ccxt\base\exchange.py:1391**, in Exchange.load_markets**(self, reload, params)**

```
1389 if self.has['fetchCurrencies'] is True:
1390     currencies = self.fetch_currencies()
-> 1391 markets = self.fetch_markets(params)
1392 return self.set_markets(markets, currencies)
```

File **~\anaconda3\lib\site-packages\ccxt\coinbase.py:864**, in coinbase.fetch_markets**(self, params)**

```
858 """
859 retrieves data on all markets for coinbase
860 :param dict params: extra parameters specific to the exchange api endpoint
861 :returns [dict]: an array of objects representing market data
862 """
863 method = self.safe_string(self.options, 'fetchMarkets', 'fetchMarketsV3')
--> 864 return getattr(self, method)(params)
```

File **~\anaconda3\lib\site-packages\ccxt\coinbase.py:936**, in coinbase.fetch_markets_v3**(self, params)**

```
935 def fetch_markets_v3(self, params={}):
```

```
--> 936     response = self.v3PrivateGetBrokerageProducts(params)
    937     #
    938     #     [
    939     #         {
   (...)
    969     #     ]
    970     #
    971     fees = self.v3PrivateGetBrokerageTransactionSummary(params)
```

File **~\anaconda3\lib\site-packages\ccxt\base\types.py:25**, in Entry.__init__.<locals>.unbound_method**(_self, params)**

```
     24 def unbound_method(_self, params={}):
---> 25     return _self.request(self.path, self.api, self.method, params, config=self.config)
```

File **~\anaconda3\lib\site-packages\ccxt\base\exchange.py:2865**, in Exchange.request**(self, path, api, method, params, headers, body, config)**

```
   2864 def request(self, path, api: Any = 'public', method='GET', params={}, headers: Optional[Any] = None, body: Optional[Any] = None, config={}):
-> 2865     return self.fetch2(path, api, method, params, headers, body, config)
```

File **~\anaconda3\lib\site-packages\ccxt\base\exchange.py:2861**, in Exchange.fetch2**(self, path, api, method, params, headers, body, config)**

```
   2859     self.throttle(cost)
   2860 self.lastRestRequestTimestamp = self.milliseconds()
-> 2861 request = self.sign(path, api, method, params, headers, body)
   2862 return self.fetch(request['url'], request['method'], request['headers'], request['body'])
```

File **~\anaconda3\lib\site-packages\ccxt\coinbase.py:2598**, in coinbase.sign**(self, path, api, method, params, headers, body)**

```
  2593     headers = {
  2594         'Authorization': 'Bearer ' + self.token,
  2595         'Content-Type': 'application/json',
  2596     }
  2597 else:
-> 2598     self.check_required_credentials()
  2599     nonce = str(self.nonce())
  2600     payload = ''
```

File **~\anaconda3\lib\site-packages\ccxt\base\exchange.py:3016**, in Exchange.check_required_credentials**(self, error)**

```
  3014 if self.requiredCredentials[key] and not getattr(self, key):
  3015     if error:
-> 3016         raise AuthenticationError(self.id + ' requires "' + key + '" credential')
  3017     else:
  3018         return False
```

AuthenticationError: coinbase requires "apiKey" credential

In [ ]:

```
import ccxt


# Connect to the exchange for Swing Trade

exchange = ccxt.binance()


# Set the ticker symbol and the trade amount
```

```
symbol = "BTC/USDT"
amount = 0.1

# Set the stop loss and take profit levels
stop_loss = 0.95
take_profit = 1.05

# Fetch the current price of the asset
price = exchange.fetch_ticker(symbol)["last"]

# Place a buy order with a stop loss
order = exchange.create_order(symbol, "market", "buy", amount, {
    "stop_loss_price": price * stop_loss
})
print("Order placed:", order)

# Set a flag to track whether the trade is open
is_open = True

while is_open:
    # Fetch the current price of the asset
    price = exchange.fetch_ticker(symbol)["last"]

    # Check if the trade has hit the stop loss or take profit
    if price < stop_loss:
```

```
        # Sell the asset and close the trade
        exchange.create_order(symbol, "market", "sell", amount)
        is_open = False
    elif price > take_profit:
        # Sell the asset and close the trade
        exchange.create_order(symbol, "market", "sell", amount)
        is_open = False

print("Trade closed")
```

In [ ]:

```
import ccxt

# Connect to the exchange for Derivatives trade
exchange = ccxt.bitmex()

# Set the ticker symbol and the trade size
symbol = "BTC/USD"
size = 1

# Place a long futures contract order
order = exchange.create_order(symbol, "futures", "buy", size)
print("Order placed:", order)

# Check the position of the asset
position = exchange.private_get_position({"symbol": symbol})
```

```
print("Position:", position)


# Place a short futures contract order
order = exchange.create_order(symbol, "futures", "sell", size)
print("Order placed:", order)


# Check the position of the asset again
position = exchange.private_get_position({"symbol": symbol})
print("Position:", position)
```

In [ ]:

```
import ccxt


# Connect to the exchange for Options Trade
exchange = ccxt.deribit()


# Set the ticker symbol and the trade size
symbol = "BTC/USD"
size = 1


# Set the expiration date and strike price of the option
expiration_timestamp = 1609459200
strike_price = 100000


# Place a call option order
order = exchange.create_order(symbol, "option", "buy", size, {
```

```
    "type": "call",
    "expiration_timestamp": expiration_timestamp,
    "strike_price": strike_price
})
print("Order placed:", order)

# Place a put option order
order = exchange.create_order(symbol, "option", "buy", size, {
    "type": "put",
    "expiration_timestamp": expiration_timestamp,
    "strike_price": strike_price
})
print("Order placed:", order)
```

In [ ]:

```
import ccxt

# Connect to the exchange for Futures Trade
exchange = ccxt.bitmex()

# Set the ticker symbol and the trade size
symbol = "BTC/USD"
size = 1

# Place a long futures contract order
order = exchange.create_order(symbol, "futures", "buy", size)
```

```
print("Order placed:", order)


# Check the position of the asset
position = exchange.private_get_position({"symbol": symbol})
print("Position:", position)


# Place a short futures contract order
order = exchange.create_order(symbol, "futures", "sell", size)
print("Order placed:", order)


# Check the position of the asset again
position = exchange.private_get_position({"symbol": symbol})
print("Position:", position)
```

In [ ]:

```
import ccxt


# Connect to the exchanges for Arbitrage Trades
exchange1 = ccxt.binance()
exchange2 = ccxt.bitfinex()


# Set the ticker symbol
symbol = "BTC/USDT"


# Fetch the current prices of the asset on the two exchanges
price1 = exchange1.fetch_ticker(symbol)["last"]
```

```
price2 = exchange2.fetch_ticker(symbol)["last"]

# Calculate the price difference
difference = price1 - price2

# Set the trade amount
amount = 0.1

if difference > 0:
    # Buy on exchange 1 and sell on exchange 2
    order1 = exchange1.create_order(symbol, "market", "buy", amount)
    order2 = exchange2.create_order(symbol, "market", "sell", amount)
    print("Orders placed:", order1, order2)
elif difference < 0:
    # Sell on exchange 1 and buy on exchange 2
    order1 = exchange1.create_order(symbol, "market", "sell", amount)
    order2 = exchange2.create_order(symbol, "market", "buy", amount)
    print("Orders placed:", order1, order2)
else:
    print("No arbitrage opportunity")
```

In [ ]:

```
import ccxt

# Connect to the exchange for Perpetual Futures
exchange = ccxt.bitmex()
```

```
# Set the ticker symbol and the trade size
symbol = "BTC/USD"
size = 1

# Place a long perpetual futures contract order
order = exchange.create_order(symbol, "perpetual", "buy", size)
print("Order placed:", order)

# Check the position of the asset
position = exchange.private_get_position({"symbol": symbol})
print("Position:", position)

# Place a short perpetual futures contract order
order = exchange.create_order(symbol, "perpetual", "sell", size)
print("Order placed:", order)

# Check the position of the asset again
position = exchange.private_get_position({"symbol": symbol})
print("Position:", position)
```

In [ ]:

```
import ccxt

# Connect to the exchange for Commodities Trades
exchange = ccxt.okcoin()
```

```
# Set the ticker symbol and the trade size
symbol = "XAU/USD"
size = 1

# Place a long trade
order = exchange.create_order(symbol, "spot", "buy", size)
print("Order placed:", order)

# Check the balance of the asset
balance = exchange.fetch_balance()
print("Asset balance:", balance[symbol])

# Place a short trade
order = exchange.create_order(symbol, "spot", "sell", size)
print("Order placed:", order)

# Check the balance of the asset again
balance = exchange.fetch_balance()
print("Asset balance:", balance[symbol])
```

In [ ]:

```
import requests

# Make a GET request to the specified URL for Application Programming Interfaces.
response = requests.get("https://www.bloomberg.com/")
```

```
# Print the status code and the content of the response
print("Status code:", response.status_code)
print("Content:", response.text)
```

In [ ]:

```
import bitcoin

# Generate a new private key and corresponding public key Bitcoin Wallets
private_key = bitcoin.random_key()
public_key = bitcoin.privtopub(private_key)

# Generate a new Bitcoin address
address = bitcoin.pubtoaddr(public_key)

print("Private key:", private_key)
print("Public key:", public_key)
print("Address:", address)
```

In [ ]:

```
import web3

# Connect to the Ethereum node for Web3 Wallets
web3 = web3.Web3(web3.Web3.HTTPProvider("http://localhost:8888"))

# Check the current block number
block_number = web3.eth.blockNumber
```

```
print("Block number:", block_number)


# Generate a new private key and corresponding public key
private_key = web3.eth.account.create().privateKey.hex()
public_key = web3.eth.account.privateKeyToAccount(private_key).address

print("Private key:", private_key)
print("Public key:", public_key)
```

In [ ]:

```
import requests
import json

# Set the API endpoint and the access token VISA WALLET CODE
api_endpoint = "https://sandbox.api.visa.com/vctc/customerrules/v1/consumertransactioncontrols"
access_token = "your_access_token"

# Set the request headers
headers = {
    "Authorization": f"Bearer {access_token}",
    "Content-Type": "application/json"
}

# Set the request body
body = {
    "primaryAccountNumber": "4111111111111111",
```

```
    "transactionAmount": "100.00",
    "transactionCurrencyCode": "USD",
    "merchantCategoryCode": "6012"
}

# Make a POST request to the API endpoint
response = requests.post(api_endpoint, headers=headers, data=json.dumps(body))

# Print the status code and the response body
print("Status code:", response.status_code)
print("Response:", response.text)
```

In [ ]:

```
import requests
import json

# Set the API endpoint and the access token MASTERCARD WALLET CODE
api_endpoint = "https://sandbox.api.mastercard.com/wallet-services-web/walletapi/v1/issuer/issuers/{issuerId}/wallet/wallets"
access_token = "your_access_token"

# Set the request headers
headers = {
    "Authorization": f"Bearer {access_token}",
    "Content-Type": "application/json"
}
```

```
# Set the request body
body = {
    "name": "My Wallet",
    "description": "My digital wallet for Mastercard cards"
}

# Make a POST request to the API endpoint
response = requests.post(api_endpoint, headers=headers, data=json.dumps(body))

# Print the status code and the response body
print("Status code:", response.status_code)
print("Response:", response.text)
```

In [ ]:

```
import stripe

# Set the API key Payment Protections for Payment Processing
stripe.api_key = "sk_test_your_api_key"

# Set the payment details
amount = 10000
currency = "USD"
description = "My payment"

# Create a payment intent
```

```
intent = stripe.PaymentIntent.create(
    amount=amount,
    currency=currency,
    description=description,
    payment_method_types=["card"],
    confirm=True
)

# Print the payment intent
print("Payment intent:", intent)
```

In [ ]:

```
import drachma

# Set the payment details
recipient = "3J98t1WpEZ73CNmQviecrnyiWrnqRhWNLy"
amount = 0.001

# Get the current transaction fee rate
fee_rate = drx.getnetworkfee(2)
print("Transaction fee rate:", fee_rate)

# Create the transaction
tx = drx.mktx(
    [],
    [(recipient, amount)],
```

```
    version=1
)

# Sign the transaction
tx = drx.sign(tx, 0, "your_private_key")

# Send the transaction
txid = drx.send(tx)
print("Transaction ID:", txid)
```

In [ ]:

```
import web3

# Connect to the Ethereum node for YIELD FARM CODE
web3 = web3.Web3(web3.Web3.HTTPProvider("http://localhost:8888"))

# Set the address of the yield farm contract
contract_address = "0xyour_contract_address"

# Get the contract abi
contract_abi = [{...}]

# Load the contract
contract = web3.eth.contract(address=contract_address, abi=contract_abi)

# Set the parameters for the deposit function
```

```
deposit_amount = 1

deposit_token = "0xyour_token_address"


# Deposit the token into the yield farm

tx_hash = contract.functions.deposit(deposit_amount, deposit_token).transact()


# Wait for the transaction to be mined

tx_receipt = web3.eth.waitForTransactionReceipt(tx_hash)


# Check the status of the transaction

if tx_receipt["status"] == 1:

    print("Transaction successful")

else:

    print("Transaction failed")
```

In [ ]:

```
import stripe


# Set the API key MONEY TRANSFER CODE

stripe.api_key = "sk_test_your_api_key"


# Set the payment details

amount = 100

currency = "USD"

description = "My payment"

source = "tok_visa"
```

```
# Create a charge
charge = stripe.Charge.create(
    amount=amount,
    currency=currency,
    description=description,
    source=source
)

# Print the charge
print("Charge:", charge)
```

In [ ]:

```
import plaid

# Set the API key and environment Direct Deposit
client = plaid.Client(client_id="your_client_id", secret="your_secret", environment="sandbox")

# Set the access token
access_token = "your_access_token"

# Set the bank account details
bank_id = "your_bank_id"
account_id = "your_account_id"

# Create a deposit
```

```
deposit = client.Deposit.create(access_token, bank_id, account_id, {
  "amount": 500,
  "name": "Deposit"
})

# Print the deposit
print("Deposit:", deposit)
```

In [ ]:

```
#The Matrix Key rate system is for random allocation of capital distribution of capital.
```

In [33]:

```
import math
import pandas as pd
import numpy as np



nc1 = np.matrix([[5,5,1], [5,8,7], [4,4,2]])
nc2 = np.matrix([[8,8,8], [7,3,4], [4,2,7]])
nc3 = np.matrix([[9,9,3], [6,9,6], [5,9,3]])
nc4 = np.matrix([[10,10,10], [2,7,10], [6,6,6]])

rate1low = nc1*0.00005
rate2low = nc2*0.00005
rate3low = nc3*0.00005

lowrate = [rate1low, rate2low, rate3low]
```

```
for x in lowrate:
  print(f'These are the lowest Matrix Hash Rates in Drachma: {x}')


gc1 = np.matrix([[9,24,20,], [16,12,12], [19,12,28], [8,25,29]])
gc2 = np.matrix([[24,24,12], [23,7,2], [3,9,2], [2,3,7]])
gc3 = np.matrix([[2,3,5], [4,10,2], [6,2,7], [4,9,6]])
gc4 = np.matrix([[9,5,8], [10,2,7], [8,2,7], [8,10,9]])

medium1 = gc1*0.0005
medium2 = gc2*0.0005
medium3 = gc3*0.0005

mediumrate = [medium1, medium2, medium3]
for x in mediumrate:
  print(f'These are the medium Matrix Hash Rates in Drachma: {x}')

bc1 = np.matrix([[8,9,2], [5,5,6], [7,8,5], [4,5,3]])
bc2 = np.matrix([[7,9,6], [4,6,2], [7,8,6], [2,6,9]])
bc3 = np.matrix ([[5,9,8], [7,6,4], [9,1,3], [7,2,7]])
bc4 = np.matrix([[5,9,8], [5,9,4], [9,5,10], [2,11,11],[5,5,5]])

highest1 = bc1*0.0005
highest2 = bc2*0.0005
highest3 = bc3*0.0005
```

```
highestrates = [highest1, highest2, highest3]
for x in highestrates:
  print(f'These are the highest Matrix Hash Rates in Drachma: {x}')


MasterKey = np.matrix([[7,5,6], [4,3,7], [5,2,6], [9,2,10],
                [7,4,8],[6,4,3]])
for x in MasterKey:
  print(MasterKey)
```

```
These are the lowest Matrix Hash Rates in Drachma: [[2.5e-04 2.5e-04 5.0e-05]
 [2.5e-04 4.0e-04 3.5e-04]
 [2.0e-04 2.0e-04 1.0e-04]]
These are the lowest Matrix Hash Rates in Drachma: [[0.0004  0.0004  0.0004 ]
 [0.00035 0.00015 0.0002 ]
 [0.0002  0.0001  0.00035]]
These are the lowest Matrix Hash Rates in Drachma: [[0.00045 0.00045 0.00015]
 [0.0003  0.00045 0.0003 ]
 [0.00025 0.00045 0.00015]]
These are the medium Matrix Hash Rates in Drachma: [[0.0045 0.012  0.01  ]
 [0.008  0.006  0.006 ]
 [0.0095 0.006  0.014 ]
 [0.004  0.0125 0.0145]]
These are the medium Matrix Hash Rates in Drachma: [[0.012  0.012  0.006 ]
 [0.0115 0.0035 0.001 ]
 [0.0015 0.0045 0.001 ]
```

[0.001 0.0015 0.0035]]

These are the medium Matrix Hash Rates in Drachma: [[0.001 0.0015 0.0025]

[0.002 0.005 0.001]

[0.003 0.001 0.0035]

[0.002 0.0045 0.003]]

These are the highest Matrix Hash Rates in Drachma: [[0.004 0.0045 0.001]

[0.0025 0.0025 0.003]

[0.0035 0.004 0.0025]

[0.002 0.0025 0.0015]]

These are the highest Matrix Hash Rates in Drachma: [[0.0035 0.0045 0.003]

[0.002 0.003 0.001]

[0.0035 0.004 0.003]

[0.001 0.003 0.0045]]

These are the highest Matrix Hash Rates in Drachma: [[0.0025 0.0045 0.004]

[0.0035 0.003 0.002]

[0.0045 0.0005 0.0015]

[0.0035 0.001 0.0035]]

[[7 5 6]

[4 3 7]

[5 2 6]

[9 2 10]

[7 4 8]

[6 4 3]]

[[7 5 6]

[4 3 7]

```
 [ 5  2  6]
 [ 9  2 10]
 [ 7  4  8]
 [ 6  4  3]]
[[ 7  5  6]
 [ 4  3  7]
 [ 5  2  6]
 [ 9  2 10]
 [ 7  4  8]
 [ 6  4  3]]
[[ 7  5  6]
 [ 4  3  7]
 [ 5  2  6]
 [ 9  2 10]
 [ 7  4  8]
 [ 6  4  3]]
[[ 7  5  6]
 [ 4  3  7]
 [ 5  2  6]
 [ 9  2 10]
 [ 7  4  8]
 [ 6  4  3]]
[[ 7  5  6]
 [ 4  3  7]
 [ 5  2  6]
```

```
 [ 9  2 10]
 [ 7  4  8]
 [ 6  4  3]]
```

In [ ]:

```
import ccxt
import time

# Initialize the exchanges you want to use
exchange1 = ccxt.binance()
exchange2 = ccxt.kucoin()

while True:
    # Fetch the current prices of Bitcoin on both exchanges
    price1 = exchange1.fetch_ticker('BTC/USDT')['last']
    price2 = exchange2.fetch_ticker('BTC/USDT')['last']

    # Check if there is an opportunity for arbitrage
    if price1 > price2:
        # Calculate the difference in price and the amount of Bitcoin to buy
        diff = price1 - price2
        amount = 100 / price2

        # Place a buy order on exchange 2
        exchange2.create_market_buy_order('BTC/USDT', amount)
```

```
        # Place a sell order on exchange 1

        exchange1.create_market_sell_order('BTC/USDT', amount)


        # Log the trade

        print(f'Bought {amount} BTC on exchange 2 at {price2} and sold on exchange 1 at {price1} for a profit of {diff}')

    else:

        print(f'No opportunity found. Price on exchange 1 is {price1} and price on exchange 2 is {price2}')


    # Sleep for a minute before checking again

    time.sleep(60)
```

In [ ]:

```
import math

def micro_arbitrage(integral, Derivative_cost, Price, Asset_valuation, Time):

  return (integral)[Derivative_cost*Price] + [Asset_valuation*log(Time)]
```

In [ ]:

```
import web3


# Connect to the Ethereum node for Zero Knowledge Proof Smart Contract

web3 = web3.Web3(web3.Web3.HTTPProvider("http://localhost:8888"))


# Set the contract source code

contract_source_code =

pragma solidity ^0.5.0;
```

```
contract ZeroKnowledgeProof {
    function prove(bytes calldata input) public pure returns (bool) {
        // Implement the zero knowledge proof logic here
        return true;
    }
}
```

```
# Compile the contract
compiled_contract = web3.eth.compile.solidity(contract_source_code)

# Get the contract interface
contract_interface = compiled_contract['<stdin>:ZeroKnowledgeProof']

# Deploy the contract
contract = web3.eth.contract(abi=contract_interface['abi'], bytecode=contract_interface['bin'])
tx_hash = contract.deploy

# Get the contract address
contract_address = tx_receipt['contractAddress']

# Load the contract
zero_knowledge_proof = web3.eth.contract(address=contract_address, abi=contract_interface['abi'])
```

In [ ]:

```
import stellar_sdk
```

```
# Set the horizon server for Federated Byzantine Agreement
server = stellar_sdk.Server("https://horizon-testnet.stellar.org")

# Set the account details
public_key = "your_public_key"
secret_key = "your_secret_key"

# Load the account
account = stellar_sdk.Account(public_key, secret=secret_key)

# Set the transaction details
amount = "100"
asset = stellar_sdk.Asset.native()
recipient = "your_recipient_address"

# Create the transaction
transaction = (
    stellar_sdk.TransactionBuilder(
        source_account=account,
        network_passphrase="Test SDF Network ; September 2015",
    )
    .append_payment_op(recipient, amount, asset)
    .build()
)
```

```
# Sign the transaction
transaction.sign(secret_key)

# Submit the transaction
response = server.submit_transaction(transaction)
```

In [ ]:

```
import fabric

# Set the network configuration for Proof of Reputable Observation
network_config = fabric.Network.get_test_network()

# Set the reputation threshold
reputation_threshold = 50

# Set the consensus plugin
consensus_plugin = fabric.Consensus.ProReO(reputation_threshold)

# Create the network
network = fabric.Network.create(
    network_config=network_config,
    consensus_plugin=consensus_plugin
)

# Start the network
```

```
network.start()


# Stop the network
network.stop()
```

In [ ]:

```
import web3

# Connect to the Ethereum node Proof of Stake Smart Contract
web3 = web3.Web3(web3.Web3.HTTPProvider("http://localhost:8888"))

# Set the contract source code
contract_source_code =
pragma solidity ^0.5.0;

contract ProofOfStake {
    mapping (address => uint) public stakes;
    uint public totalStake;

    function stake(uint _value) public {
        require(_value > 0);
        stakes[msg.sender] += _value;
        totalStake += _value;
    }

    function unstake(uint _value) public {
```

```
        require(_value > 0 && _value <= stakes[msg.sender]);
        stakes[msg.sender] -= _value;
        totalStake -= _value;
    }

    function selectValidator() public view returns (address) {
        uint random = uint(keccak256(abi.encodePacked(now, msg.sender))) % totalStake;
        uint sum = 0;
        address selected;
        for (address validator in stakes) {
            sum += stakes[validator];
            if (sum > random) {
                selected = validator;
                break;
            }
        }
        return selected;
    }
}



# Compile the contract
compiled_contract = web3.eth.compile.solidity(contract_source_code)

# Get the contract interface
```

#- need assistance in creating

In [ ]:

```
import ccxt

# Set the exchange and the bank account details for Crypto Deposits
exchange_id = "binance"
exchange_class = getattr(ccxt, exchange_id)
exchange = exchange_class({
    "apiKey": "your_api_key",
    "secret": "your_secret_key",
    "options": {"defaultType": "spot"}
})
bank_account_id = "your_bank_account_id"

# Load the balance
balance = exchange.fetch_balance()

# Select the cryptocurrency to deposit
currency = "BTC"
amount = balance[currency]["total"]

# Deposit the cryptocurrency
response = exchange.withdraw(currency, amount, bank_account_id)
print(response)
```

In [ ]:

```
import web3

# Connect to the Ethereum node for Personal DAO
web3 = web3.Web3(web3.Web3.HTTPProvider("http://localhost:8888"))

# Set the contract source code
contract_source_code =
pragma solidity ^0.5.0;

contract DecentralizedAutonomousOrganization {
    address public owner;
    mapping (address => bool) public members;
    uint public memberCount;
    uint public minimumQuorum;

    constructor() public {
        owner = msg.sender;
        members[msg.sender] = true;
        memberCount = 1;
        minimumQuorum = 1;
    }

    function addMember(address _member) public {
        require(msg.sender == owner);
        require(!members[_member]);
```

```
    members[_member] = true;
    memberCount++;
}

function removeMember(address _member) public {
    require(msg.sender == owner);
    require(members[_member]);
    members[_member] = false;
    memberCount--;
}

function changeMinimumQuorum(uint _minimumQuorum) public {
    require(msg.sender == owner);
    minimumQuorum = _minimumQuorum;
}

function vote(bool _vote) public {
    require(members[msg.sender]);
    emit VoteCast(_vote);
}

function execute(bytes memory _data) public {
    uint votesFor = 0;
    uint votesAgainst = 0;
    for (address member in members) {
```

```
            if (members[member]) {
                if (member.call.value(0)(_data)) {
                    votesFor++;
                } else {
                    votesAgainst++;
                }
            }
        }
        if (votesFor >= minimumQuorum) {
            emit ExecutionSucceeded();
        } else {
            emit ExecutionFailed();
        }
    }

    event VoteCast(bool _vote);
    event ExecutionSucceeded();
    event ExecutionFailed();
}



# Compile the contract
compiled_contract = web3.eth.compile.solidity(contract_source_code)

# Get the contract interface
```

```
contract_interface = compiled_contract["<stdin>:OBC:DecentralizedAutonomousOrganization"]

# Deploy the contract
contract = web3.eth.contract(abi=contract_interface["abi"], bytecode=contract_interface["bin"])
tx_hash = contract.deploy(transaction={"from": web3.eth.coinbase})

# Get the contract address
contract_
```

In [ ]:

```
import web3

# Connect to the Ethereum node Personal Treasury
web3 = web3.Web3(web3.Web3.HTTPProvider("http://localhost:8888"))

# Set the contract source code
contract_source_code =
pragma solidity ^0.5.0;

contract DecentralizedTreasury {
    address public owner;
    address public recipient;
    uint public totalFunds;
    uint public fundsAvailable;
    mapping (bytes32 => bool) public requests;
    uint public requestCount;
```

```
    uint public minimumQuorum;


    constructor(address _recipient, uint _totalFunds, uint _minimumQuorum) public {
        owner = msg.sender;
        recipient = _recipient;
        totalFunds = _totalFunds;
        fundsAvailable = _totalFunds;
        minimumQuorum = _minimumQuorum;
    }


    function requestFunds(bytes32 _requestId, uint _amount) public {
        require(_amount > 0 && _amount <= fundsAvailable);
        require(!requests[_requestId]);
        requests[_requestId] = true;
        requestCount++;
    }


    function vote(bytes32 _
```

In [ ]:

```
import web3


# Connect to the Ethereum node for Personal Custody
web3 = web3.Web3(web3.Web3.HTTPProvider("http://localhost:8888"))


# Set the contract source code
```

```
contract_source_code =

pragma solidity ^0.5.0;


contract DecentralizedCustody {
    address public owner;
    mapping (address => bool) public authorized;
    mapping (address => uint) public balances;
    uint public totalSupply;


    constructor(uint _totalSupply) public {
        owner = msg.sender;
        authorized[msg.sender] = true;
        totalSupply = _totalSupply;
        balances[msg.sender] = _totalSupply;
    }


    function authorize(address _spender) public {
        require(msg.sender == owner);
        authorized[_spender] = true;
    }


    function revoke(address _spender) public {
        require(msg.sender == owner);
        authorized[_spender] = false;
    }
```

```
function transfer(address _to, uint _value) public {
    require(balances[msg.sender] >= _value && authorized[msg.sender]);
    require(_to != address(0));
    balances[msg.sender] -= _value;
    balances[_to] += _value;
    emit Transfer(msg.sender, _to, _value);
}

function approve(address _spender, uint _value) public {
    require(balances[msg.sender] >= _value && authorized[msg.sender]);
    require(_spender != address(0));
    emit Approval(msg.sender, _spender, _value);
}

function transferFrom(address _from, address _to, uint _value) public {
    require(balances[_from] >= _value && authorized[_from] && authorized[_to]);
    require(_to != address(0));
    balances[_from] -= _value;
    balances[_to] += _value;
    emit Transfer(_from, _to, _value);
}

event Transfer(address indexed _from, address indexed _to, uint _value);
event Approval(address indexed _owner, address indexed _spender, uint _value);
```

```
}
```

```
# Compile the contract
compiled_contract = web3.eth.compile.solidity(contract_source_code)
```

In [ ]:

```
import web3

# Connect to the Ethereum node DeFi Loans
web3 = web3.Web3(web3.Web3.HTTPProvider("http://localhost:8888"))

# Set the contract source code
contract_source_code =
pragma solidity ^0.5.0;

contract DecentralizedLoanProvider {
    address public owner;
    mapping (bytes32 => Loan) public loans;
    uint public loanCount;
    uint public interestRate;
    uint public minimumLoanAmount;
    uint public maximumLoanAmount;

    constructor(uint _interestRate, uint _minimumLoanAmount, uint _maximumLoanAmount) public {
        owner = msg.sender;
```

```
        interestRate = _interestRate;
        minimumLoanAmount = _minimumLoanAmount;
        maximumLoanAmount = _maximumLoanAmount;
    }

    function requestLoan(bytes32 _loanId, uint _amount, uint _term) public {
        require(_amount >= minimumLoanAmount && _amount <= maximumLoanAmount);
        require(loans[_loanId] == Loan(0, 0, 0, 0));
        loans[_loanId] = Loan(_amount, _term, interestRate, msg.sender);
        loanCount++;
        emit LoanRequested(_loanId, _
```

ln [ ]:

```
from web3 import Web3

# Connect to Ethereum blockchain TO CREATE DRACHMAS WRAPPED TOKENS
w3 = Web3(Web3.HTTPProvider("https://mainnet.infura.io/v3/YOUR-PROJECT-ID"))

# ERC-20 contract ABI
contract_abi = [{...}]

# ERC-20 contract address
contract_address = "0x..."

# Create contract object
contract = w3.eth.contract(address=contract_address, abi=contract_abi)
```

```
# Your Ethereum address
my_address = "0x..."

# Your private key
my_private_key = "0x..."

# Function to mint new tokens
def mint_token(to_address, amount):
    # Build transaction
    nonce = w3.eth.getTransactionCount(my_address)
    gas_price = w3.eth.gasPrice
    gas_limit = 100000
    data = contract.functions.mint(to_address, amount).buildTransaction({
        'gas': gas_limit,
        'gasPrice': gas_price,
        'nonce': nonce,
    })

    # Sign transaction
    signed_tx = w3.eth.account.signTransaction(data, my_private_key)

    # Send transaction
    tx_hash = w3.eth.sendRawTransaction(signed_tx.rawTransaction)
```

```
    # Wait for transaction to be mined
    tx_receipt = w3.eth.waitForTransactionReceipt(tx_hash)

    # Check if transaction was successful
    if tx_receipt['status'] == 1:
        print("Token minted successfully")
    else:
        print("Error: Token mint failed")

# Example usage
to_address = "0x..."
amount = 100
mint_token(to_address, amount)
```

In [ ]:

```
from web3 import Web3
from openzeppelin_sdk import NFT

# Connect to Ethereum blockchain TO CREATE DRACHMA BACKED NFTS
w3 = Web3(Web3.HTTPProvider("https://mainnet.infura.io/v3/YOUR-PROJECT-ID"))

# Your Ethereum address
my_address = "0x..."

# Your private key
my_private_key = "0x..."
```

```
# Create NFT object
nft = NFT(w3, my_address, my_private_key)

# Create new NFT
metadata = "My first NFT"
image_url = "https://example.com/image.png"
tx_hash = nft.create(metadata, image_url)

# Wait for transaction to be mined
tx_receipt = w3.eth.waitForTransactionReceipt(tx_hash)

# Check if transaction was successful
if tx_receipt['status'] == 1:
    print("NFT created successfully. NFT ID:", tx_receipt['logs'][0]['topics'][1])
else:
    print("Error: NFT creation failed")
```

In [ ]:

```
import requests
import json

# Flight booking API endpoint for Flights
api_endpoint = "https://flight-booking-api.com/book"

# Flight information
```

```
flight_info = {
    "origin": "New York",
    "destination": "Los Angeles",
    "date": "2022-06-01",
    "passenger_name": "John Smith",
    "passenger_email": "john.smith@example.com"
}

# Make API request to book flight
response = requests.post(api_endpoint, json=flight_info)

# Check if booking was successful
if response.status_code == 200:
    flight_booking_details = json.loads(response.text)
    print("Flight booked successfully! Booking details:")
    print("Booking reference:", flight_booking_details['booking_reference'])
    print("Total cost:", flight_booking_details['total_cost'])
else:
    print("Error: Flight booking failed")
```

In [ ]:

```
import requests
import json

# Car booking API endpoint for booking CARS
api_endpoint = "https://car-booking-api.com/book"
```

```
# Car rental information
car_info = {
    "pickup_location": "New York",
    "dropoff_location": "Los Angeles",
    "pickup_date": "2022-06-01",
    "dropoff_date": "2022-06-05",
    "car_type": "SUV",
    "renter_name": "John Smith",
    "renter_email": "john.smith@example.com"
}

# Make API request to book car
response = requests.post(api_endpoint, json=car_info)

# Check if booking was successful
if response.status_code == 200:
    car_booking_details = json.loads(response.text)
    print("Car booked successfully! Booking details:")
    print("Booking reference:", car_booking_details['booking_reference'])
    print("Total cost:", car_booking_details['total_cost'])
    print("Pickup location:", car_booking_details['pickup_location'])
    print("Dropoff location:", car_booking_details['dropoff_location'])
else:
    print("Error: Car booking failed")
```

In [ ]:

```
import requests
import json

# Boat booking API endpoint for BOATS
api_endpoint = "https://boat-booking-api.com/book"

# Boat rental information
boat_info = {
    "pickup_location": "Miami",
    "dropoff_location": "Bahamas",
    "pickup_date": "2022-06-01",
    "dropoff_date": "2022-06-05",
    "boat_type": "Yachts",
    "renter_name": "John Smith",
    "renter_email": "john.smith@example.com"
}

# Make API request to book boat
response = requests.post(api_endpoint, json=boat_info)

# Check if booking was successful
if response.status_code == 200:
    boat_booking_details = json.loads(response.text)
    print("Boat booked successfully! Booking details:")
```

```
    print("Booking reference:", boat_booking_details['booking_reference'])
    print("Total cost:", boat_booking_details['total_cost'])
    print("Pickup location:", boat_booking_details['pickup_location'])
    print("Dropoff location:", boat_booking_details['dropoff_location'])
else:
    print("Error: Boat booking failed")
```

In [ ]:

```
import requests
import json

# Private jet booking API endpoint FOR CHARTER JETS
api_endpoint = "https://private-jet-booking-api.com/book"

# Private jet charter information
jet_info = {
    "origin": "New York",
    "destination": "Los Angeles",
    "departure_date": "2022-06-01",
    "return_date": "2022-06-05",
    "jet_type": "Bombardier Challenger 850",
    "passenger_name": "John Smith",
    "passenger_email": "john.smith@example.com"
}

# Make API request to book private jet
```

```
response = requests.post(api_endpoint, json=jet_info)


# Check if booking was successful
if response.status_code == 200:
    jet_booking_details = json.loads(response.text)
    print("Private jet booked successfully! Booking details:")
    print("Booking reference:", jet_booking_details['booking_reference'])
    print("Total cost:", jet_booking_details['total_cost'])
    print("Origin:", jet_booking_details['origin'])
    print("Destination:", jet_booking_details['destination'])
else:
    print("Error: Private jet booking failed")
```

In [ ]:

```
import pandas as pd
from datetime import datetime
import ccxt as ccxt
pair = 'ETH/USDT'


ohlcv = binance.fetch_ohlcv(pair, '1d')
prices = [x[4] for x in ohlcv]


dates = [datetime.fromtimestamp (x[0 // 1000]) for x in ohlcv]
data = pd.series (prices, index=dates)
 data.plot()
```

Ozhuman Corporation Capital C-Corp
Operation North Star

The following report is for the intercontinental infrastructure project known as the Ozhuman DTP. The Ozhuman Digital Trade Pipeline, the world's first financial market network using a Proof of Stake (POS) Sharding Quantitative investment software suite with our own currency algorithm to facilitate commerce overseas with the internet as a financial bridge for the digital capital market economy. An entirely private market.

Introduction

Hello, my name is Mr. Jonathan B. Samuel and I am the last child of Mr. Benjamin O. Samuel and Mrs. Mini G. Samuel. With that responsibility in mind it is incumbent upon me to build for my family in the off chance that something happens to me. This report will serve as the blueprint for anyone that aspires to build their own capital market economy. To preface my argument it is crucial to understand that it is not enough to simply build a capital market, that is what has been done with crypto-assets in the current landscape of the market. You need infrastructure that will facilitate commerce as you cannot control the market, it must be able to operate freely and ensure that all market shareholders who own market share units are able to benefit from the same economic privileges as you have for yourself. The determinant of success in this system is the investor-builder's creativity. have written the code for the algorithm as well as the software design. I am a self-taught Financial Engineer, Data Scientist, Writer, Entrepreneur, Consultant, and avid learner. I enjoy complex problems whether that is creating a combinatorial matrix matching engine that is 10x leveraged to create derivatives markets that match to their leveraged infrastructure value. Or simply writing music that will speak to the trauma people have lived through. At the time that I am writing this, on October 24th 2021 it has been a little over seven years of development to reach the point that we are currently at. Now I will share everything that will go into play to close our years of research for what we have been working on. A digital financial trade pipeline to have private market access as investors into our own cuts of markets cornered by the algorithm.

Algorithmic Currency Base Code

Investment Banks use credit worthiness to determine the legitimacy of those who are going to borrow credit from creditors all the time. Credit is moved, created, swapped, and backed in real time within the market across the globe. It is one the core tools that drive the global financial market forward, the other being liquidity.

This analysis is driven towards understanding a nation's historical track record to derive its own *'Credit Worthiness'*. The administration of any nation is driven towards creating business opportunities that support its economy internally as for its citizenry but to also use those internal opportunities to create a driving force to welcome Foreign-Direct Investment (FDIs) into its financial markets.

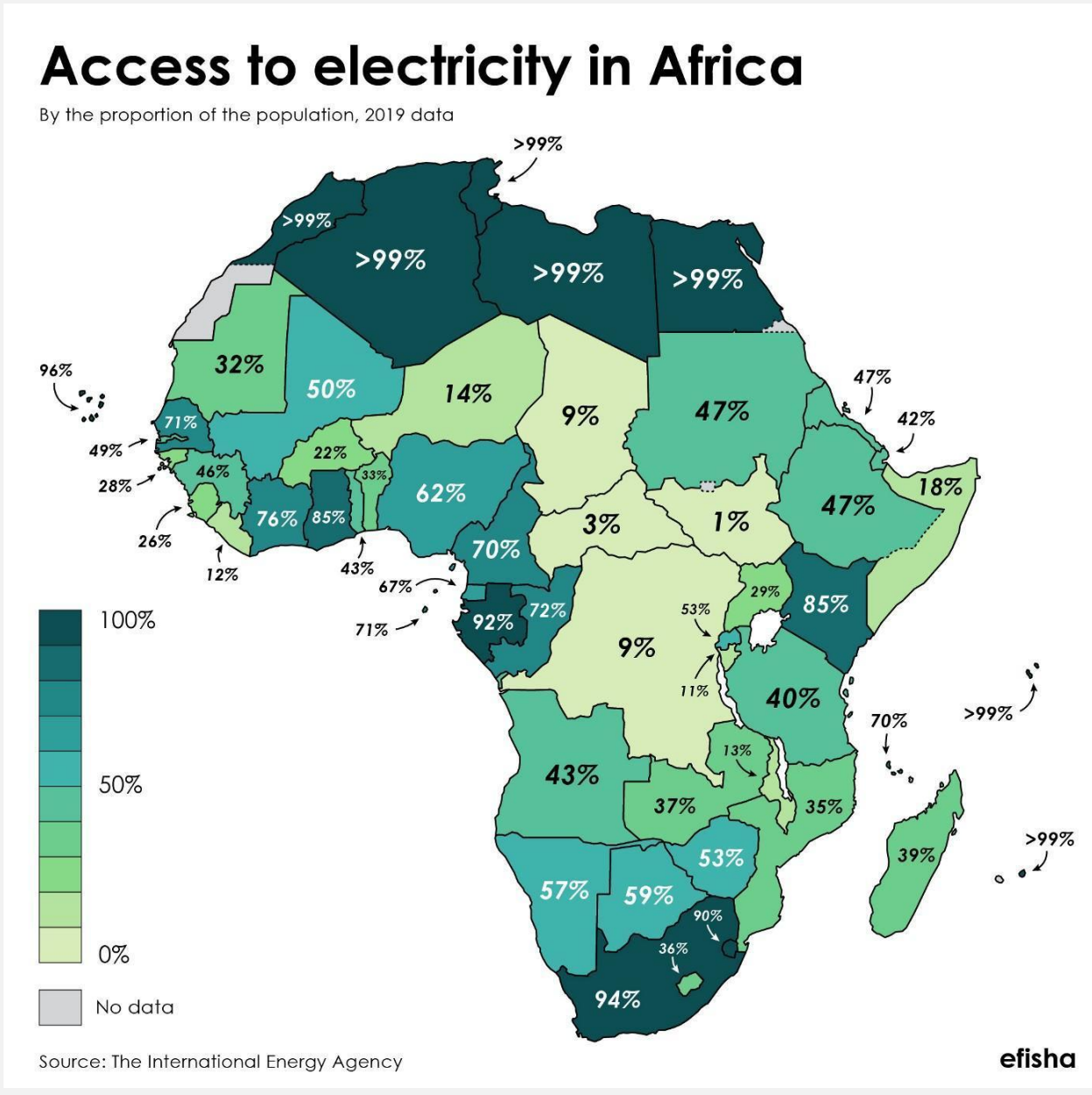
The core countries of focus for this analysis are located within the continent of Africa and compiles 13 nations that are the following.

1. Algeria
2. Angola
3. Central African Republic
4. Ivory Coast
5. Egypt
6. Kenya
7. Mauritius
8. Morocco
9. Nigeria
10. South Africa
11. Tunisia
12. Zambia
13. Zimbabwe

Banking, Debt, Financial, Inflation and Systemic Crises that cover from the range of 1860 all the way up to 2014. To create an assessment for investors these are the core steps that will cover how we eventually achieve that ability to determine on our own HOW we should value a nation's investment 'worthiness'.

1. Historical financial health analysis specific to leading emerging market nations within the dataset.
2. Mobility of credit and liquidity to the nations of our analysis. The framework itself will be **malleable into any sector of the *INVESTOR's* choice.**
3. Understanding a nation's financial past to see its investment future by investors.

To better understand the usage and allocation of capital within emerging capital markets from their historical trajectory and current market stability.



This chart is provided by the International Energy Agency, with respect to the thirteen nations that we will be analyzing. I share this as electrical infrastructure is an opportunity for emerging markets where business operators can leverage the information that is provided by the algorithm that is built with the base code to understand if they should move into a country of interest or not based on its historical analysis. Now for this report we will not be covering the scope of the electrical infrastructure market in the thirteen countries that are mentioned as our focus is to understand credit worthiness of a nation and its respect towards us as investors allocating capital towards the nations of interest.

Algeria has greater than 99% access to electricity. **Angola** with 43%, **Central African Republic** with 3%, **Ivory Coast** with 76%, **Egypt** with greater than 99%, **Kenya** with 85%, **Mauritius** with 99% (its to the right of Madagascar who has 39%), **Morocco** with greater than 99%, **Nigeria** with 62%, **South Africa** with 94%, **Tunisia** with greater than 99%, **Zambia** with 37%, and finally **Zimbabwe** with 53%.

This should also serve as an example of the broad scale of investment opportunities that are present in the continent of Africa today. Keep in mind this is only one sector of focus within a region of countries. This does not cover the entire market that is available for analysis within. That can be used to build cooperative relationships from the investor and the investment. The more stability that is actualized within the economy itself can drive further opportunities of growth internally which will also attract foreign investors. This is speculative as a nation's forecast can change at any point in time, one element of the crypto-market that is incorporated within the algorithm is that it will be running the entire time with a machine-learning environment based on the datasets that are incorporated into the environment running a combinatorial matrix matching engine. Defaulted to a binary end results, 1 being that there is a market failure imminent into the system and 0 being that it isn't for the time being that the investors want to allocate their capital and invest into the nations of interest.

Access alone is worth nothing if you are chained to the floor, and mobility by itself is in itself worth nothing if you have no access in life.

Description of Dataset

For the datasets that we used for this problem, the initial dataset was provided by Kaggle. It was under the title ***"Africa Economic, Banking and Systemic Data*** *(Data on the Economic and Financial Crises in 13 African countries from 1860 to 2014)***".** Within this dataset, for the thirteen African stats that I introduced we have the values assigned to the following variables.

Country Code, Name of Country, Year of Observation, Systemic Crisis, Exchange rate to U.S Dollar, Domestic Debit, Sovereign External Debt, GDP Weighted, Inflation Annual CPI, Independence, Currency Crisis, Inflation, Crisis, and finally Banking Crisis.

Limitations that appear when it comes to the dataset is that it does not provide us a picture for the mobility of their economies and their relation to the global economy.

Statistical Methods

Hypothesis:

H0:β1=0 vs. *Ha*:*β*1≠0

What will we set for our Null Hypothesis and our Alternative Hypothesis? When a nation faces a systemic crisis, banking crisis, has a considerable amount of debt, and inflation, if it is able to overcome a currency crisis and maintain stability for its currency within foreign exchange markets then credit and liquidity will flow into the nation. Where the borrowing activity, ease of financing, and mobility of financial architecture will dictate the level of Foreign-Direct Investment(FDIs). Foreign exchange markets provide a vital asset that is constantly traded in each market day, its currency. As the objective of our own project is to issue our own wallet, digital card, currency, stock, stock exchange, and blockchain protocol it is vital that we first understand the mobility of crises based on a nation's historical value and data to see where they will be going long term. This way our own investments are able to understand a nation's financial limit, invest accordingly and build the infrastructure to support our investment and that of the nation in question.

H*0:** There is not a significant statistical relationship between the amount of recorded instances of a nation going through a ***currency crisis and ***inflation*** which influences the *flow of capital invested* into a nation?

Ha*:** There is a significant statistical relationship between the amount of recorded instances of a nation going through a ***currency crisis and ***inflation*** *the flow of capital invested* into a nation.

Importing Libraries and Reading in Dataset

```
#import libraries
```

```
import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

import numpy as np

from statsmodels.formula.api import ols

import scipy.stats as st
```

Then from there we upload our dataset from the direct url that is provided to us through Kaggle.

```
#Load and Preview AfricanNationsHistorical Dataset

from google.colab import files
```

```
uploaded = files.upload()
```

The file is saved as ‘african_crises.csv’ then we enter the following code block to show our dataframe before we can do Exploratory Data Analysis and Features Engineering.

Preview of the Africa Dataframe

	case	cc3	country	year	systemic_crisis	exch_usd	domestic_debt_in_default	sovereign_external_debt_default	gdp_weighted_default	inflation_annual_cpi	independence	currency_crises
0	1	DZA	Algeria	1870	1	0.052264	0	0	0.0	3.441456	0	0
1	1	DZA	Algeria	1871	0	0.052798	0	0	0.0	14.149140	0	0
2	1	DZA	Algeria	1872	0	0.052274	0	0	0.0	-3.718593	0	0
3	1	DZA	Algeria	1873	0	0.051680	0	0	0.0	11.203897	0	0
4	1	DZA	Algeria	1874	0	0.051308	0	0	0.0	-3.848561	0	0

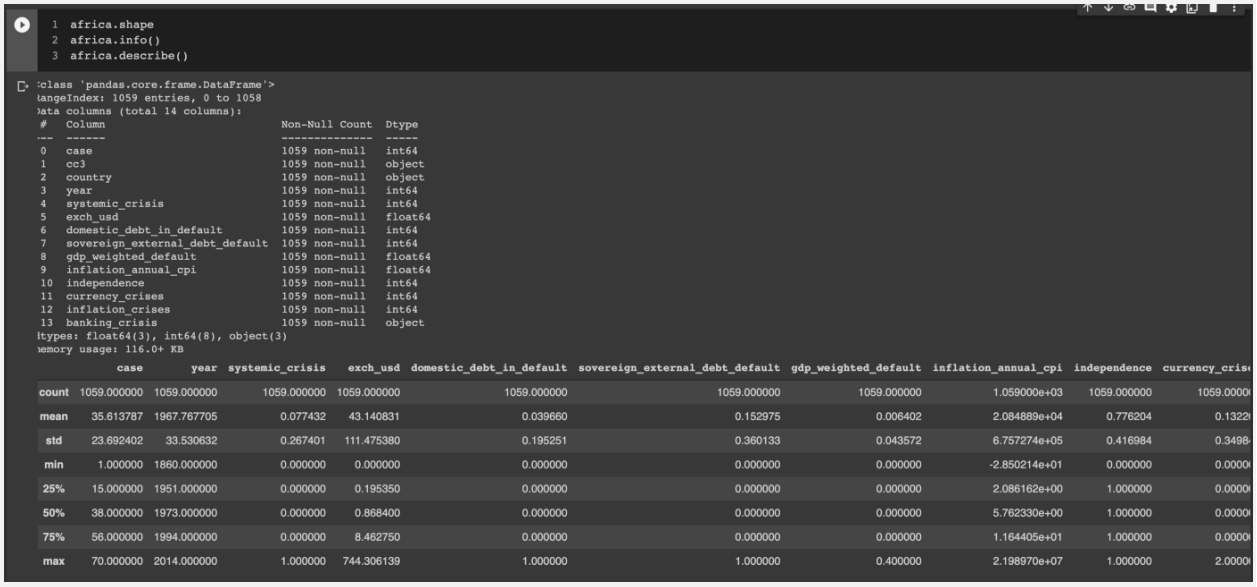
Initial output of the Africa Dataframe

Exploratory Data Analysis

```
africa.shape
```

```
africa.info()
```

```
africa.describe()
```



```
1 africa.shape
2 africa.info()
3 africa.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1059 entries, 0 to 1058
Data columns (total 14 columns):
 #   Column                            Non-Null Count  Dtype
---  ------                            --------------  -----
 0   case                              1059 non-null   int64
 1   cc3                               1059 non-null   object
 2   country                           1059 non-null   object
 3   year                              1059 non-null   int64
 4   systemic_crisis                   1059 non-null   int64
 5   exch_usd                          1059 non-null   float64
 6   domestic_debt_in_default          1059 non-null   int64
 7   sovereign_external_debt_default   1059 non-null   int64
 8   gdp_weighted_default              1059 non-null   float64
 9   inflation_annual_cpi              1059 non-null   float64
 10  independence                      1059 non-null   int64
 11  currency_crises                   1059 non-null   int64
 12  inflation_crises                  1059 non-null   int64
 13  banking_crisis                    1059 non-null   object
dtypes: float64(3), int64(8), object(3)
memory usage: 116.0+ KB
```

	case	year	systemic_crisis	exch_usd	domestic_debt_in_default	sovereign_external_debt_default	gdp_weighted_default	inflation_annual_cpi	independence	currency_crise
count	1059.000000	1059.000000	1059.000000	1059.000000	1059.000000	1059.000000	1059.000000	1.059000e+03	1059.000000	1059.0000
mean	35.613787	1967.767705	0.077432	43.140831	0.039660	0.152975	0.006402	2.084889e+04	0.776204	0.1322
std	23.692402	33.530632	0.267401	111.475380	0.195251	0.360133	0.043572	6.757274e+05	0.416984	0.3498
min	1.000000	1860.000000	0.000000	0.000000	0.000000	0.000000	0.000000	-2.850214e+01	0.000000	0.0000
25%	15.000000	1951.000000	0.000000	0.195350	0.000000	0.000000	0.000000	2.086162e+00	1.000000	0.0000
50%	38.000000	1973.000000	0.000000	0.868400	0.000000	0.000000	0.000000	5.762330e+00	1.000000	0.0000
75%	56.000000	1994.000000	0.000000	8.462750	0.000000	0.000000	0.000000	1.164405e+01	1.000000	0.0000
max	70.000000	2014.000000	1.000000	744.306139	1.000000	1.000000	0.400000	2.198970e+07	1.000000	2.0000

Information pertaining to the type of data that are found within the Africa Dataframe as well as quantitative values.

Checking for any missing values across the fourteen columns that are within the data frame. This is to ensure that within our data analysis we do not have values that are unaccounted for ensuring that we only work with data that is proven in the dataset.

```
africa.isnull().sum()
```

```
[32]  1 #Checking for any missing values within the dataframe
      2 africa.isnull().sum()
```

```
case                                0
cc3                                 0
country                             0
year                                0
systemic_crisis                     0
exch_usd                            0
domestic_debt_in_default            0
sovereign_external_debt_default     0
gdp_weighted_default                0
inflation_annual_cpi                0
independence                        0
currency_crises                     0
inflation_crises                    0
banking_crisis                      0
dtype: int64
```

Counted list for any missing values within each column in the dataframe.

It was important to make our analysis down the road easier by converting any string, values that are in letters to values in binary 0 and 1 for whether the event occurred at the time that it was recorded or that it did not occur. Out of the fourteen columns, only one had String values which was in the column of ‘banking_crisis’ and if a banking crisis did in fact occur within the nation at the time it was recorded it was marked ‘crisis’ and if not ‘no_crisis’ which we will then convert to 1 for if there was a crisis and 0 if there was not a crisis in the recorded time.

Features Engineering

#Converting Str values in 'banking_crisis' to Int values.

africa['banking_crisis'] = africa['banking_crisis'].replace(to_replace='no_crisis', value=0)

```
africa['banking_crisis'] = africa['banking_crisis'].replace(to_replace='crisis', value=1)
```

```
#then check for the new dataframe and how the data is fit.
```

```
africa.head()
```

```
2 africa['banking_crisis'] = africa['banking_crisis'].replace(to_replace = 'no_crisis', value=0)
3 africa['banking_crisis'] = africa['banking_crisis'].replace(to_replace = 'crisis', value=1)
4 africa.head()
```

mic_crisis	exch_usd	domestic_debt_in_default	sovereign_external_debt_default	gdp_weighted_default	inflation_annual_cpi	independence	currency_crises	inflation_crises	banking_crisis
1	0.052264	0	0	0.0	3.441456	0	0	0	1
0	0.052798	0	0	0.0	14.149140	0	0	0	0
0	0.052274	0	0	0.0	-3.718593	0	0	0	0
0	0.051680	0	0	0.0	11.203897	0	0	0	0
0	0.051308	0	0	0.0	-3.848561	0	0	0	0

The recorded string values that were in place have now been converted into int data types.

From prior research against the nations that are in this sample I researched and found five leading Emerging Market nations. Which are the following South Africa, Nigeria, Egypt, Angola, and finally Kenya. And only with the year, country, and the types of crises.

```
#Edit Africa df to only country, year, and crises type.
```

```
africa_clean = africa[['country', 'year', 'systemic_crisis', 'currency_crises', 'inflation_crises', 'banking_crisis']]
```

```
africa_clean.head()
```

```
africa_clean = africa[['country', 'year', 'systemic_crisis', 'currency_crises', 'inflation_crises', 'banking_crisis']]
africa_clean.head()
```

	country	year	systemic_crisis	currency_crises	inflation_crises	banking_crisis
0	Algeria	1870	1	0	0	1
1	Algeria	1871	0	0	0	0
2	Algeria	1872	0	0	0	0
3	Algeria	1873	0	0	0	0
4	Algeria	1874	0	0	0	0

Easier dataframe for us to focus on with only the country, year, and the types of crises they have faced within the 200 years of historical financial data.

We have our dataframe with only the main categories we discussed, now to implement the five leading emerging markets that we have previously discussed.

```
#Cut and organize the countries into their own DataFrames

sa = africa_clean[africa_clean.country == 'South Africa']

ni = africa_clean[africa_clean.country == 'Nigeria']

eg = africa_clean[africa_clean.country == 'Egypt']

an = africa_clean[africa_clean.country == 'Angola']

kn = africa_clean[africa_clean.country == 'Kenya']
```

```
final_africa = [an, eg, kn, ni, sa]
```

```
results = pd.concat(final_africa)
```

```
results.head(200)
```

```
sa = africa_clean[africa_clean.country == 'South Africa']
ni = africa_clean[africa_clean.country == 'Nigeria']
eg = africa_clean[africa_clean.country == 'Egypt']
an = africa_clean[africa_clean.country == 'Angola']
kn = africa_clean[africa_clean.country == 'Kenya']


final_africa = [an, eg, kn, ni, sa]
results = pd.concat(final_africa)
results.head(200)
```

	country	year	systemic_crisis	currency_crises	inflation_crises	banking_crisis
85	Angola	1921	0	0	1	0
86	Angola	1922	0	0	1	0
87	Angola	1923	0	0	1	0
88	Angola	1924	0	0	1	0
89	Angola	1925	0	0	0	0
...	...	...	...	...	...	...
401	Egypt	1978	0	0	0	0
402	Egypt	1979	0	1	0	0
403	Egypt	1980	1	0	1	0
404	Egypt	1981	1	0	0	1
405	Egypt	1982	1	0	0	1

200 rows × 6 columns

The final cleaned and organized dataframe from the original dataset that has only the five leading emerging markets within the area we are analyzing. Comparing the recording instances of the four crises for Angola, Egypt, Kenya, Nigeria, and South Africa.

And then with the new data frame that is called ‘results’ I used the .describe() method to understand the quantitative values based on how we engineered the values that we wanted based on the criteria that was critical to our analysis.

```
#show statistical values of finalized df
```

```
results.describe()
```

```
1 results.describe()
```

	year	systemic_crisis	currency_crises	inflation_crises	banking_crisis
count	473.000000	473.000000	473.000000	473.000000	473.000000
mean	1958.486258	0.061311	0.137421	0.114165	0.084567
std	38.272832	0.240153	0.362628	0.318348	0.278530
min	1860.000000	0.000000	0.000000	0.000000	0.000000
25%	1933.000000	0.000000	0.000000	0.000000	0.000000
50%	1964.000000	0.000000	0.000000	0.000000	0.000000
75%	1990.000000	0.000000	0.000000	0.000000	0.000000
max	2014.000000	1.000000	2.000000	1.000000	1.000000

Updated quantitative values with Angola, Egypt, Kenya, Nigeria, and South Africa. Check out the Standard Deviation values across the four crisis categories.

We can see even amongst the chart that across the standard distribution across the four categories, the highest category of value is ‘currency_crises’ followed by ‘inflation_crises’. We can understand that from this initial assessment of our newly established dataframe. Now for our OLS Regression Assessment to understand the relationship between Currency Crisis and Inflation Crisis. Inflation will be Y and Currency will be X, since inflation is a naturally occurring factor within the market itself and a currency crisis is not.

```
#OLS Model between Inflation and Currency Crisis for Results df
```

```
model = ols("inflation_crises ~ currency_crises", data=results).fit()
```

print(model.summary())

```
model = ols("inflation_crises ~ currency_crises", data=results).fit()
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:       inflation_crises   R-squared:                       0.093
Model:                            OLS   Adj. R-squared:                  0.091
Method:                 Least Squares   F-statistic:                     48.05
Date:                Tue, 27 Jul 2021   Prob (F-statistic):           1.37e-11
Time:                        08:11:25   Log-Likelihood:                -106.28
No. Observations:                 473   AIC:                             216.6
Df Residuals:                     471   BIC:                             224.9
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                     coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept          0.0775      0.015      5.188      0.000       0.048       0.107
currency_crises    0.2671      0.039      6.932      0.000       0.191       0.343
==============================================================================
Omnibus:                      211.952   Durbin-Watson:                   1.016
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              685.036
Skew:                           2.196   Prob(JB):                    1.76e-149
Kurtosis:                       6.934   Cond. No.                         2.82
==============================================================================
```

OLS Regression Results between Inflation Crisis and Currency Crisis within the Results DataFrame of the five leading emerging markets that we selected.

From our OLS Regression we understood that there is in fact a relationship between Inflation Crises and Currency Crises that a nation will encounter which has a direct impact on its ability to maintain credit and liquidity amongst its borders and in turn leveraging a stable economic environment to attract Foreign-Direct Investors(FDIs) for direct investment into the nations of question. Our P-Value was below the 0.05 baseline that we established whether or not to reject the Null Hypothesis and or to fail to reject the Null Hypothesis.

The increase of currency crises that a nation faces in its historical duration of existence and the frequency of that information itself shows that there are weaknesses that are present in the economic infrastructure of its capital market and or its administrative faculties that provide in itself investment potential for the nation and nation-state partners. Inflation is a measure of the currency's purchasing power against comparative goods of value and commerce within a nation, but a currency crisis is the failure of a foreign exchange rate from lack of reserve funds to maintain liquidity pairings with the sovereign state's currency and other sovereign currencies. Therefore when it comes to the allocation of investment capital we want nations that have a lower frequency of their historical data of a currency crisis. We don't want the Sorros of the world to create their own 'Breaking the Bank of England'.

This initial model serves as the baseline for the currency of the network, a digital fiat currency that is 1:1 of any currency in the global economy. Balanced and stabilized free of fees for remittances. The 0.05% that is charged for remittances builds the reserve of the public credit system that any market share operator can borrow from for their own business ventures with a fixed rate of 6% per month against the borrowed principal. There are no financial barriers stopping you from generating and managing capital. This ensures that all market share operators that borrow from the financial market economy are creating commerce to pay back any credit that is borrowed against them for their ventures. This is not a primary means of creating capital for ventures, simply a part of the infrastructure that exists to market share owners if needed for their capital resources. This allows the market to serve as a bridge that stabilizes all types of international trade. Especially for countries that have barriers because of geopolitical instability we serve as a gateway through our private trade routes.

This baseline comes into production with three core financial assets that serve as tools to operate in the market for market operators.

1. OBC - $OBC - Our market token that functions as the stock of the market, it will also have the algorithm that way we are constantly keeping assurances of a stable alternative digital fiat financial by-way. A side street for the global financial market and its networks to facilitate trade even when commerce seems like it will not happen because of social volatility.

2. ONX - $ONX - Our market currency will be subject to use only within the network. No taxation within the system as this is an altruistic capitalistic network. We use our money to remove bad operators within markets overseas and domestically. Raw financial power in the hands of the people.

All investment data is free, everyone has an equal chance to capitalize on their ability for investment for themselves and for the market to benefit from it. I designed the two market tokens to work as two balancing scales.

ONX will be able to assess all market volumes in and out of the market economy like a diaphragm expands and contracts within the human body. This way we create an independent foreign exchange market simply by trading in on outflows of capital to countries. We will create digital tokens that back our infrastructure developments in partner nations to facilitate job creation and economic development. This pairing of foreign exchange markets will act as financial truss beams bridging the development on the internet that we are doing financially and then in turn creating it in real life with large infrastructure projects. Railways. Busses. Schools. Power Plants. Refineries. Agribusiness. Commodities. Minerals. Metals. Logistics. And more.

OBC will be able to grow in value proportional to the economic and investment capital that is built on the market network. The OBC Protocol, the Ozhuman Banking Corporation - Protocol. All of your assets are your own and tied to your identity on the network the same principle of a Social Security Number tying your identity to all things that you can do in the United States. Your digital wallet will be your entire DID, Digital Identity Database. You control your data. You control your assets. You control your destiny. Act it. $OBC will be backed by the software and the blockchain protocol using my software design. This way, the more businesses and investments that are built and transacted on the network. Then only will the stock grow. Our investment in ourselves and our investment in Our Market. A sovereign digital republic where each person is a Caesar. ONX will be able to grow the local economies that are being built on the network. A currency that is free from any government influence or proxy. The world's first reserve currency that answers to no state. This is the ultimate safeguard to protect all persons in less than favorable situations where politicians and governments only serve their interests. This market is the final market for the people to counter exactly what is being done to them as it is being done to others overseas.

There are only six components, the two market tokens. The software and protocol with our own digital wallet for us to commence trade at any point in time and our digital cards. And finally, the merchant trade exchanges. We will be investing into countries at scale to access infrastructure from its cash flow that will create our funds and our mobility to invest in ourselves.

Exchange Locations

1. Kerala, India
2. Tamil Nadu, India
3. Colombo, Sri Lanka
4. Jakarta, Indonesia
5. Abu Dhabi, United Arab Emirates
6. Riyadh, Saudi Arabia
7. Moscow, Russia
8. Beirut, Lebanon
9. Santorini, Greece
10. Tirana, Albania
11. Milan, Italy,
12. Paris, France
13. Algiers, Algeria
14. Rabat, Morocco
15. São Paulo, Brazil
16. Bogota, Colombia
17. Port Au Prince, Haiti
18. Cairo, Egypt
19. Addis Ababa, Ethiopia
20. Johannesburg, South Africa
21. Miami, Florida, USA
22. District of Columbia, USA
23. Boston, Massachusetts, USA
24. Portsmouth, New Hampshire, USA

Together with 24 financial exchanges we will be building our own financial trade route that connects the world to the United States. With Ozhuman Corporation and the United States becoming the largest investor of infrastructure all while creating jobs in the same breath domestically as well as overseas. There will be a total of ten financial exchange hubs that

function in the capital market network; the first Stock exchange is the ONX Stock Exchange, the Ozhuman National Exchange.

This concludes the report.

<G>

In our view, there are two ways to deal with the entire market as a whole. You can be the traditionalist view and serve only that which exists in the traditional financial world. Or be the alternative and be those that build continuously with the aim of one day taking down the traditional world and building towards a new world order with the alternative. We Gideon, believe why not build a structure that has the best of both world composites? You read that right, an entire world structure that is built with fiat mixed crypto composite currencies. Digital assets function as fiat in every day-to-day life with the seamlessness of paper money. Going to Dunkin', AMC Theaters, and even Home Depot. Now how does that come to be? This is not a theoretical hopscotch paper, this is a high-level theoretical mathematical paper that I, Mr. Jonathan B. Samuel have spent the last six years breaking down to the simplest of languages that everyone can understand.

At the expense of his mental health, being admitted six times, and facing the demons of tragedy and trauma in his life. And yet he did not yield. He did not quit. As he understands then and understands now, this is about liberation. Not just freedom as that will always have the cost of life towards it, liberation does not require the expense of freedom as it is the base thesis of blockchain. Consensus. You see, the Byzantine Generals' problem all had the same problem, they had to serve the ambition of a King. Now the Byzantine Generals' problem meets the Azantine Emperor's Solution.

The Generals face their commands and tell their men.
"Men! Why must we conquer a city that has only the interest of our King? For his coffers? For his spouses? For his heirs? And what for us? Carnage. Death. Decay. Destruction. Is this what is befitting to soldiers such as us?!"

The men looked in awe at each other with the fear of treason. Then a Commander, one who had earned the respect of his men. Who had fought valiantly even to the point of breaking his back with his men. Walked with a stilt on one side. At the sight of the Commander walking toward the General's mount. The entire army rose and began beating on these shields screaming to the heavens.

‘O, Commander! One who defies death for Us! One who broke his bones for Us! One who marches with us into Battle. What must we do Commander!”

A hard fork to the traditional Byzantine Generals’ problem. How do decentralized parties achieve consensus without a centralized party? By forming a Commission. A centralized decentralized organization. CDO. That can only act with the ability to deploy resources and the end result is the outcome that creates what is known on the street as a ‘Regulator’.

An entity to call the shots for each organization and effectively carry out any counter-attack to take the entity city. The Constantinople of the world. Taking the same tactics of Mehmed II and applying them to the digital battlefield. The formation of Pashas. Twelve elected members as Pashas form the Commission. 12 members in the Decentralized Commission Organization. The DCO.

“Begin Men! Mount the Assault! Take the City!”

The General is dead. The Commander has now become the Shah. With his Commission.

Now the Pashas were in each battalion that circled the city. Each with specific orders. One would mount the assault as fodder. Two would shoot with archery from the front. Another two would shoot archery from the back. Another four would rain down incendiaries in the city. And the entire army would lead the assault. Beating drums and yelling to the heavens.

“In the name of his name and all of the names that he has! We will take down this city in his name for all of his names and all of their names are his names. BEGIN!”

Done.

When the war was won. The Commander who had become Shah then instructed a general election for eleven more Shahs and 11 more Commissions. A grand total of 121 Pashas and 12 Shahs and One Emperor Shahjahan. The Congress of 133. 133 Quantum Bits per Bit. Higher-grade Efficiency than 256-bit computing, which can be broken by Quantum Computing.

Then Emperor Shahjahan instructed his men that in this day and age of technology, who will come and become my Minister to lead the development of our world? An Orphan walked up to the Emperor.

“I will but I will need complete freedom to do so.” The Shahjahan looked at him and said.

“Very well my boy. Give him everything he needs for our world. It is my decree.’ The boy began studying mathematics, Euclidean geometry, differential geometry, fractional calculus, multivariable calculus, integral, probability theory, practical philosophy, number theory, prime numbers, space in mathematics, logic, proof calculus, mathematical logic, argumentation theory, term logic, stoic logic, Mohism, Anviksiki, Propositional calculus, first-order logic, vector space, topological space, Hilbert space, Probability space, Rational numbers, and four color theorem.

Once he had studied those materials with his disabilities of bipolar disorder and his gift of autism. He began studying axioms, propositions, and mathematical proofs. He became the youngest Professor of Mathematics without needing a formal degree as it was all taught to him verbally through linguistics. Now mathematics was his tool. As the Pashas knew he was autistic therefore they knew the best way for the scholar to learn mathematics was not by numbers but by reading to him. And understanding he did well, he excelled at it. Only needed to read or hear small amounts and know entire ways in dimensionality that would allow him to build out everything that he wanted to build out for the world. When he had finished his learning he went and saw the greatest mathematician and studied with him. Moriarty.

They built a close relationship and in the end, Professor Moriarty asked one thing. ‘Do come by for some tea good ole chap. There is a world for us to theorize together.’ The boy looked with awe and joy and bowed touching his feet out of respect as that was the custom to do so in Kerala where he is from. The Professor embraced him and said, ‘Never forget even though you came from nothing in this world, you now have everything you need to build out the entire world for yourself and for others do not take that responsibility lightly. As they will hunt you for your mind, stand tall and fight.’ And as the boy said his thanks to the Professor he turned back to him slyly smiling and said.

“I am a Professor after all now aren’t I, my old friend Moriarty.” The professor looked at his grin and matched him for it as well. And the boy made his way back to the 121 Pashas, the 12 Shahs, and Emperor Shahjahan.

“My friend, Professor Samuyil, what have you learned for our world?” The Professor fixed his glasses and his shawl and began.

“Everything, for now.” The Congress erupted in joy as they had all done what was necessary on that day, and now the Professor was going to show them exactly what

needed to be done to truly bring the power of the state and the state and its power to each member that composed it. Without needing to decay the state. \\

The Shahjahan Standard

There are Zero Knowledge Proofs, asynchronous Federated Byzantine Agreement, Proof of Reputable Observations and Proof of Stake. This is about tying it all together and building a software base that can allow all code bases that are open sources to come together and consequently build a digital confederation. That is equally accessible to each person to build for what they believe in their scheme and ethos, pathos, and logos. As this great King and his Ministers took a chance on this Orphan boy, I call this Codename: Orphan.

Euclidean Geometry is about third dimensional shapes, spaces, and sectors. (x,y,z). It is the third dimensionality of all of the aforementioned objects. Using the icosahedron shape to program a fluid digital mesh that will cover those shapes, spaces, sectors in the standard real third dimension but also the standard inverse imaginary third dimension for all off-chain transactions this way we are able to govern for the entire world that we are creating and the two sided transactions that can be used from their executional shape alone can be done so in the manner of conducting their existing value till it is no longer applicable as the world adjusting in the shape themselves. Of all imaginary and rational numbers consequently. The mesh design being that of an isosceles triangle as that is the strongest shape as to why we see them so commonly in bridges and truss systems.

With the digital mesh now on the object that you have, you now have a fluid monetary processing system that is programmable. This is programmable to the actual throughput of the system at hand and creates a digital storage system, keeping in mind the objects nature in the third dimension both rational and imaginary. This allows us to conduct monetary programmable transactions that are in that of storage related proximity to the world that we are creating on and to that of the digital ethos of decentralization as that allows each person to be exactly as what is aforementioned for themselves and to conduct themselves for what they want to do so with sovereignty and autonomy.

Directly leveraging Zero Knowledge Proofs and asynchronous Federated Byzantine Agreements this way all entities and elements of the digital entities are truly decentralized and do not need to sacrifice anything as the base bit digitally is at Q133 per bit, asynchronous. Truly belonging to the hands of those that exist in the system. This way there is Proof of Reputable Observations. And as such also done so in a manner that is compliant with the utmost regard to regulatory standards. This way Gaios is a ball and chain system following the mesh that creates the chain. The ‘ball’ is

all of the data centers and or systems that are running off this proof and Gaios is the Universal System Currency. And as such the currency is a monetary chain-link oracle index system. From the ChainLink and The GRT standards. An MCOIS. Monetary Currency Operating Investment System. For all types of supplies, operations, and systems.

I call this ASTRAL. Asymmetrically Trading Raw Alpha Liquidity. If you want to own your own French swimming pools ASTRAL is the key. To then use ZANES, Zero Asset Euclidean Systems. Which then uses the Icosahedron shape mesh plants to intertwine with the Gaios monetary mesh and forming G.
A single operating system for all crypto-fiat mixed composite currencies from the Gaios Standard unified as layer 2, 1, 0 of traditional financing through Gaios.

Global Asset Interweb Network Operating System.

A currency, A network, A operating system. For all currencies, networks, and open sources OS. Backed by the Earth.

The Earth's earth, water, air, dams, reefs, oceans, and soils.

And this is the formula for it. <P.S>

$+-\ Average\ (Imaginary)[x\ less\ than\ or\ equal\ to\ <\ \sum$ null matrix >greater than or

equal to x factorial percentile] (Rational numbers) Average -+

With the values that go into X compiling itself into Monetary input (Mi), Monetary output (Mo), Bridge 1 M_inter, and Bridge 2 M_outer.

The World was already ours, they told us over years of writing history that it was not. Now we dive into a deeper understanding of what needs to be done in the traditional sense of why decentralization is typically viewed as a challenge.

Decentralization. Security. And Scalability. Everyone, you see this formula is the net generation application of Non-Cooperative Game Theory. It is how the great Emperor and his Pashas on that day were able to successfully form the first DCO. A core-centric ethos, pathos, and logos for the attack on the cities that we have conquered that are under our empire. This formula is coupled only with that of what we now know as Chaos Theory. Chaos is the natural order of the world that we live in. For centuries many great mathematicians before myself have all tried aimlessly to control it. How is such a feat possible? To control the natural order of the world is to control the world itself, and that

to a singular person is to do so as a centralized entity. This cannot be as such my brothers and sisters. Therefore, the only way to act as such is to create the bridge and choose which of the Doors we will enter as Chaos Theory itself in a graph shows that it begins after the 5th interval and our transactional basis is to complete one turnaround in 5 microseconds of completion in one cycle. Do you see where I am going with this now my People?

The crowd looked on with puzzling faces. The Emperor understood and told him to begin. When I was young in my tutelage with the Pashas, I developed a baseline algorithm that can be used to understand the potential of all countries around the world for their investment in what we conduct as trade or better known as Foreign Direct Investments [FDIs]. Now I will expand on such and bring to discuss my analysis of the algorithm and its design.

How do we analyze potential across global capital markets for Foreign Direct Investments? [FDIs] Can we as a consensus use a nation's financial history and deploy the complete set of information that we will have in this discussion to determine how we are going to actually invest in each other, amongst one another, and by one another?

Investment Banks use creditworthiness to determine the legitimacy of those who are going to borrow credit from creditors all the time. Credit is moved, created, swapped, and backed in real-time within the market across the globe. It is one of the core tools that drive the global financial market forward, the other being liquidity.

This analysis is driven towards understanding a nation's historical track record to derive its own *'Credit Worthiness'*. The administration of any nation is driven towards creating business opportunities that support its economy internally as well as for its citizenry but to also use those internal opportunities to create a driving force to welcome Foreign-Direct Investment (FDI) into its financial markets.

The core countries of focus for this analysis are located within the continent of Africa and compile 13 nations that are the following.

1. Algeria
2. Angola
3. Central African Republic
4. Ivory Coast
5. Egypt
6. Kenya

7. Mauritius
8. Morocco
9. Nigeria
10. South Africa
11. Tunisia
12. Zambia
13. Zimbabwe

Banking, Debt, Financial, Inflation, and Systemic Crises cover from the range 1860 all the way up to 2014. To create an assessment for investors these are the core steps that will cover how we eventually achieve that ability to determine on our own HOW we should value a nation's investment 'worthiness'.

1. Historical financial health analysis specific to leading emerging market nations within the dataset.
2. Mobility of credit and liquidity to the nations of our analysis. The framework itself will be **malleable in any sector of the *INVESTOR's* choice.**
3. Understanding a nation's financial past to see its investment future by investors.

To better understand the usage and allocation of capital within emerging capital markets from their historical trajectory and current market stability.

You see, my people, with this level of technology in our hands, it is our ethical responsibility to actually invest in the world and on behalf of the world. **ASTRAL** will allow us to actually move as a decentralized capital market that can act for the benefit of the world around us. **ZANES** allows us to actually store our transactions in the third dimensionality and actively determine which types of data are crucial for storage and investment in the world around us. **GAIOS** as a currency and operating system will allow us to decisively use a currency of liberation for the entire world to actually benefit its people irrespective of who they are and what they are coming from. They are our people and it is our moral responsibility from the school of rational thought in the philosophy of aesthetics. What is good, true, and beautiful in utility for the world is actually beneficial for all people even down to the simplest forms of life that are only simple to our eyes and not that of their organic composition.

Provided by the International Energy Agency, with respect to the thirteen nations that we will be analyzing. **Algeria** has greater than 99% access to electricity. **Angola** with 43%, **Central African Republic** with 3%, **Ivory Coast** with 76%, **Egypt** with greater than 99%, **Kenya** with 85%, **Mauritius** with 99% (it's to the right of Madagascar which has 39%), **Morocco** with greater than 99%, **Nigeria** with 62%, **South Africa** with 94%, **Tunisia** with greater than 99%, **Zambia** with 37%, and finally **Zimbabwe** with 53%.

This should also serve as an example of the broad scale of investment opportunities that are present in the continent of Africa today that can be used to build cooperative relationships between the investor and the investment. More stability within the economy itself can drive further opportunities for growth internally which will also attract foreign investors.

Access alone is worth nothing if you are chained to the floor, and mobility by itself is in itself worth nothing if you have no access in life. Remember where we are exactly the people who had nothing in life, not because of our lack of potential but because of the lack of mobility for our potential. Blockchain as a technology is about removing the chains of money that bound us to a life of subjugation and control of the world around us instead of it as such bound the world to us.

For the datasets that we used for this problem, the initial dataset was provided by Kaggle. It was under the title ***"Africa Economic, Banking and Systemic Data*** *(Data on the Economic and Financial Crises in 13 African countries from 1860 to 2014)***".** Within this dataset, for the thirteen African stats that I introduced, we have the values assigned to the following variables.

[Country Code, Name of Country, Year of Observation, Systemic Crisis, Exchange rate to U.S Dollar, Domestic Debit, Sovereign External Debt, GDP Weighted, Inflation Annual CPI, Independence, Currency Crisis, Inflation, Crisis, and finally Banking Crisis.]

The limitation that appears when it comes to the dataset is that it does not provide us with a picture of the mobility of their economies and their relation to the global economy. And this understanding is key as it will tie in with the very last part of my analysis. This baseline algorithm will be the base function for any unit of the **Gideon Network.** As the word Gideon in Hebrew means 'He who Cuts down.' As we had defeated the monarchy ourselves and taken the cities by storm with the Commander and the Pashas through the Decentralized Commission Organizations. And now we have a congress that must face its next inevitable change in the design that will give all of our people the ability to live as we all have in the security of our posts.

Hypothesis:

H0: $\beta 1=0$ vs. Ha: $\beta 1 \neq 0$

What will we set for our Null Hypothesis and our Alternative Hypothesis? When a nation faces a systemic crisis, banking crisis, has a considerable amount of debt, and inflation, if it is able to overcome a currency crisis and maintain stability for its currency within foreign exchange markets then credit and liquidity will flow into the nation.

Where the borrowing activity, ease of financing, and mobility of financial architecture will dictate the level of Foreign-Direct Investment (FDIs).

*H***0:** There is not a significant statistical relationship between the number of recorded instances of a nation going through a ***currency crisis*** and ***inflation*** which influences the *flow of capital invested* into a nation.

*Ha***:** There is a significant statistical relationship between the number of recorded instances of a nation going through a ***currency crisis*** and ***inflation*** *the flow of capital invested* into a nation.

Every single crisis of some sort has been the root cause of currency fluctuations in the market. To the Breaking of the Bank of England to the collapse of the United States Housing Market. Currency is the root that we must address if we are to actually solve the problems in the world around us. And that will also be addressed at the end of the technical baseline algorithm analysis as well.

Baseline proofs of systematic currency failures only show that our currency, Gaios, has the ability to build digital currency composition equivalents of the sovereign currencies of the nations. Remember we are not here to remove the power of governments; we are here to ensure the citizenry has equal power to the governments as we the people are the ones who elected the governments and we the people are the people who compose the governments. That is a check. That is the balance of order in the world around us.

<.... Enters programming .... >

```
#import libraries
```

```
import pandas as pd
```

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt

import numpy as np

from statsmodels.formula.api import ols

import scipy.stats as st

africa.shape

africa.info()

africa.describe()

africa.isnull().sum()

#Converting Str values in 'banking_crisis' to Int values.
```

```
africa['banking_crisis'] =
africa['banking_crisis'].replace(to_replace='no_crisis',
value=0)
```

```
africa['banking_crisis'] =
africa['banking_crisis'].replace(to_replace='crisis', value=1)
```

```
#then check for the new dataframe and how the data is fit.
```

```
africa.head()
```

```
#Edit Africa df to only country, year, and crises type.
```

```
africa_clean = africa[['country', 'year', 'systemic_crisis',
'currency_crises', 'inflation_crises', 'banking_crisis']]
```

```
africa_clean.head()
```

```
#Cut and organize the countries into their own DataFrames
```

```
sa = africa_clean[africa_clean.country == 'South Africa']
```

```
ni = africa_clean[africa_clean.country == 'Nigeria']

eg = africa_clean[africa_clean.country == 'Egypt']

an = africa_clean[africa_clean.country == 'Angola']

kn = africa_clean[africa_clean.country == 'Kenya']

final_africa = [an, eg, kn, ni, sa]

results = pd.concat(final_africa)

results.head(200)

#show statistical values of finalized df

results.describe()
```

```
sa = africa_clean[africa_clean.country == 'South Africa']
ni = africa_clean[africa_clean.country == 'Nigeria']
eg = africa_clean[africa_clean.country == 'Egypt']
an = africa_clean[africa_clean.country == 'Angola']
kn = africa_clean[africa_clean.country == 'Kenya']


final_africa = [an, eg, kn, ni, sa]
results = pd.concat(final_africa)
results.head(200)
```

	country	year	systemic_crisis	currency_crises	inflation_crises	banking_crisis
85	Angola	1921	0	0	1	0
86	Angola	1922	0	0	1	0
87	Angola	1923	0	0	1	0
88	Angola	1924	0	0	1	0
89	Angola	1925	0	0	0	0
...	...	...	...	...	...	...
401	Egypt	1978	0	0	0	0
402	Egypt	1979	0	1	0	0
403	Egypt	1980	1	0	1	0
404	Egypt	1981	1	0	0	1
405	Egypt	1982	1	0	0	1

200 rows × 6 columns

We can see even amongst the chart that across the standard distribution across the four categories, the highest category of value is ‘currency_crises’ followed by ‘inflation_crises’. We can understand that from this initial assessment of our newly established data frame. Now for our OLS Regression Assessment to understand the relationship between Currency Crisis and Inflation Crisis. Inflation will be Y and Currency will be X since inflation is a naturally occurring factor within the market itself and a currency crisis is not. There is a reason why this algorithm that I designed first in the course of my research went into the entire planning of the Proof that supports Gaios. This way, we will be able to see any types of collapses that are coming for countries around the world and give them the tools they need to circumvent them, using the natural econometric cycles of booms and busts for their liking.

```
#OLS Model between Inflation and Currency Crisis for Results df
```

```
model = ols("inflation_crises ~ currency_crises",
data=results).fit()
```

```
print(model.summary())
```

```
1 model = ols("inflation_crises ~ currency_crises", data=results).fit()
2 print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:       inflation_crises   R-squared:                       0.093
Model:                            OLS   Adj. R-squared:                  0.091
Method:                 Least Squares   F-statistic:                     48.05
Date:                Tue, 27 Jul 2021   Prob (F-statistic):           1.37e-11
Time:                        08:11:25   Log-Likelihood:                -106.28
No. Observations:                 473   AIC:                             216.6
Df Residuals:                     471   BIC:                             224.9
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      0.0775      0.015      5.188      0.000       0.048       0.107
currency_crises     0.2671      0.039      6.932      0.000       0.191       0.343
==============================================================================
Omnibus:                      211.952   Durbin-Watson:                   1.016
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              685.036
Skew:                           2.196   Prob(JB):                    1.76e-149
Kurtosis:                       6.934   Cond. No.                         2.82
==============================================================================
```

From our OLS Regression we understood that there is in fact a relationship between Inflation Crises and Currency Crises that a nation will encounter which has a direct impact on its ability to maintain credit and liquidity amongst its borders and in turn leverage a stable economic environment to attract Foreign-Direct Investors(FDIs) for direct investment into the nations of question. Our P-Value was below the 0.05 baseline that we established whether or not to reject the Null Hypothesis and or to fail to reject the Null Hypothesis.

The increase of currency crises that a nation faces in its historical duration of existence and the frequency of that information itself shows that there are weaknesses that are present in the economic infrastructure of its capital market and or its administrative faculties that provide in itself investment potential for the nation and nation-state partners. Inflation is a measure of the currency's purchasing power against comparative goods of value and commerce within a nation, but a currency crisis is the failure of a foreign exchange rate from a lack of reserve funds to maintain liquidity pairings with the sovereign state's currency and other sovereign currencies. Therefore, when it comes to the allocation of investment capital, we want nations that have a lower frequency of their historical data of a currency crisis. We don't want the Soros of the world to create their own 'Breaking the Bank of England'.

The congress was in an uproar, it was the first time that someone has presented the theory of humanistic investing, investing for the welfare of the world. As they learned that day.

Public and or Private Welfare Investments [PWIs] that can be deployed for the gain of Foreign Direct Investments [FDIs] by directly deploying the technology in the home country of the nations that the confederation would work with. A completely different style of investment would change the course of the world as it had already changed the course of their thinking. That was the battle that Professor Samuyil wanted to change as Moriarty had warned him that when he goes back the Pashas and the Shahs close to the Shahjahan will seek to use him in the way of deposing his teacher who gave him the blessing that began his journey. He had non-violently checkmated the entire congress with just his word alone as his tools and his mind as his weapons.

You see my people, this baseline algorithm will allow all of us to be able to actively become the world's first Humanistic Investors of a new form of capital, Humanistic Capitalism. It does not mean you need to give your profits away or do more or less than those around you. Do what you do best and to the best that you do it. Be the Human Being you always wanted to be, be more than altruistic. Be the person you never had growing up. You never saw yourself till it was too late. As this is your digital baptism. Everything that I have done so far in my research with the blessing of Shahjahan is to continue his legacy of service to his people. His legacy of doing what needs to be done regardless of the consequence to the world around him and that being his world. Personally. As now you know the first half of our currency's purpose is now for the second half for our currency's stability. \\

The Gaios Algorithmic Proof

Using the aforementioned mathematical standard, we are backing our currency not just to the value of the objects on earth. It will be the average aggregation proof of currency instruments. The proof is simple and works as such. It will aggregate the rates of bonds in the market across all 197 open share markets*(to change based on regulatory standards). After it aggregates that of 197 countries it will average them out to a single base percentile that is adjusted on a rolling basis with market prime. This way for all the debt in the world and the entire credit market itself which is currently valued at $119 Trillion United States Dollars. There is a singular currency that can move all of the liquidity for it around the world wherever credit is needed.

//The congress began to uproar again. As they understood now that they would effectively be the banks of good hope in the world around them. All respect for their communities, cultures, creeds, and credit.

You see, brothers and sisters this way Gaios is a Universal System Currency. You will have Saved the Country, my congregation. And now for my summation. This way each DCO with its own Shah can become a federation within the confederation. As we are in fact using aFBA. This way since we are all using the Universal System Currency of the network, each person is a DCO and a Shah. The crowd went silent as it meant even in a revolution as such to depose the Shahjahan would be a bloodless revolution and in that manner as such the Shahjahan knew what Professor had done for him. The Emperor then stepped down from his throne and walked toward Samuyil.

“Yohanan, come with me for a second.” The entire congress stood. They both walked to the edge of the congress.

“You know what you have done for me correctly? You have given me the ability to remove the crown that weighs on my head and finally be at peace with my family. The crown is yours, my son. You are my rightful heir. I have no need to continue what I know will end in the manner I can already see therefore as such, finish it, my son. I will go with my family now. I have waited years for this day to come and now that it is, it would not be fitting for me to be here when it is carried out. For the boy who had nothing, you have become everything, Professor Yohanan Samuyil, even without a degree of the Pashas. You are the greatest of scholars to have entered this Shahjahan’s court. It is my greatest privilege to have known you and know you firsthand.” The Emperor and Professor Samuyil shook hands and he embraced him, the freedom of no longer needing to keep the crown on his head anymore.

Yohanan walked back into the congress and the congregation was beating its floor excited to learn how it would finish.

With each person as a DCO and each person as a Shah, this allows all of our people to have sovereignty, autonomy, and freedom amongst themselves as they are no longer bound to any system, institution, or ruler. You are first of All Free Persons. This is the name of our confederation, my congregation. The AFP Confederation. You can directly do all the financing amongst yourselves or just yourselves as they will all start with the same allocation of currency, except for me.

“Professor, who are you to determine that you will have more than us when you explain exactly the type of investment, theory, currency, and capitalism we will be deploying!?”

The Professor fixed his glasses and kept a straight face.

“I am an Originator. Operator 555. Your Treasury. The Khanate. You see my friends there is no longer a king, there is no longer a pasha, there is no longer an emperor, and there is no longer any form of the traditional system of monarchy. You are all federations amongst the people you welcome into your federations. This is the AFP Confederation. The Azantine Freedom Patriarch Confederation. Azantine stands for my sister, Ashley who died at the hand of a system. And I told myself that from her death I would build. Built so far and so great that no man would dare contest the machinations of my mind that I have materialized into the matter in the world itself of matter. As we learn in Thermodynamics that energy cannot be created nor destroyed, only transformed. Her death transformed me from the boy that had a sibling to love him. A sibling rubbed his temples when he suffered from migraines. A sibling to give him counsel and care. This world because of its inefficient systems robbed me of my truest love. I hated the world for it. I hated the people for it. Do you think you want freedom? Freedom is the cost of living! You dare quarrel like dogs and then come to me for freedom?! No. This is about removing all structures so that the strongest may survive. I present to you the greatest of my machinations. Gideon. The world’s first and only Asynchronous Federated Personnel Consortium, aFPC. And this is the design for it. This way each person on the network can operate like a digital military company and or a financial corporation for all engagements. I will be your banker. Since we are a confederation and our goal is the light, we must be able to operate in the shadows. I have always loved being in service. It is what made me a great student. My aim in life was only to be of service to the greatest of minds that taught me everything that helped me become who I am now. Now, these are the rules of engagement.

MinMax: Minimize all maximum points of impact per severity.

MaxMin: Maximize all minimum points of impact per engagement.

There are no rules except for the first two. Break either, the Company will handle.

This paper is for the direct generation and creation of FPCs to stop all that aim. For the

United States of America and all of Her Allies.

Gentlepersons are you ready to be debriefed on the final steps that will allow you and your companies to truly be of service to the World?

At first the men were in hesitation but after careful consideration they knew that what the Professor was talking about was the right course of action with this much power in the hands of each individual it was the proper thing to do. And at the same time the Professor was the Treasury. “Gentlemen, the world is in need of the next generation of financiers, businesspersons. Therefore, as such I will provide you the financing within the right causes to do so and give you the tools necessary to do so. You all have three standard tools in your arsenals that are equitable to all of you the manner of how you conduct yourself and use them I will leave to your discretion. The first is **Gaios**, our currency. Since it is fiat-mix crypto composite. It will be able to operate in any sovereign nation that accepts both fiat and or crypto. You have an unbanked currency to directly finance developments across the world. You can do so in your native countries, you can do so in the countries you are domiciled in, anywhere on God’s green earth that uses money to solve its problems you are the money they will use to solve its problems. The second is **ASTRAL**, with this currency system and its operating software using **ZANES you will be able to provide liquidity, deep, and storage, high.** All three together work to create a financial black box to move all operations as needed into the endgame. The endgame is as stated, to remove, reengineer, apply, and execute. You will all be the next generation of Financial Engineers. Welcome to Gideon Laboratories.” All of the company persons looked excited, filled with discernment and meticulous precision on the next steps of what it was that they were going to do.

“Here is the final lesson of my analysis. After you understand this, you will be able to do anything that you want to do. It is my magnum opus; I call it the Control of Digital Fault Tolerance. With this simulation of this formula, you will always be able to see the stress factors within the chaos of your operations as I integrated it with my aforementioned formula and came to create it similar to the logic of a piecewise function. This formula addresses all of the traditional concerns since it is a summation of values that come to the null matrix. The very conditions of the game that you will all be playing through Non-Cooperative Game Theory. The entire analysis together has a name that I will share

with you now, **Multivariate Singular Theorem, its foundations are from discontinuous integrals in Calc II the one most commonly used in Newtonian Physics.** The ability to change the characteristics of the confederation all the way back to 0, in any direction. It took me many years to be able to build the formula, as that is the hallmark of mathematics, the confederation will always begin at the center of chaos and move with it in all of its designs and forms to better the world around us.

$$Xn + 1 = Lambda\ Xn\ (1 - Xn) - Chaos\ Theorem$$

$$X\ less\ than\ or\ equal\ \Sigma\ Null\ matrix\ greater\ than\ or\ equal\ to\ X!\ \%$$

$$Xn + 1\ less\ than\ or\ equal\ to\ \Sigma\ Null\ matrix\ greater\ than\ or\ equal\ to\ Xn + 1!\ \%$$

$$Lambda\ Xn(1 - Xn)\ less\ than\ or\ equal\ to\ \Sigma\ Null\ matrix\ greater\ than\ or\ equal\ to\ Lambda\ Xn$$

(1-Xn)! %

The entire theorem expanded is:

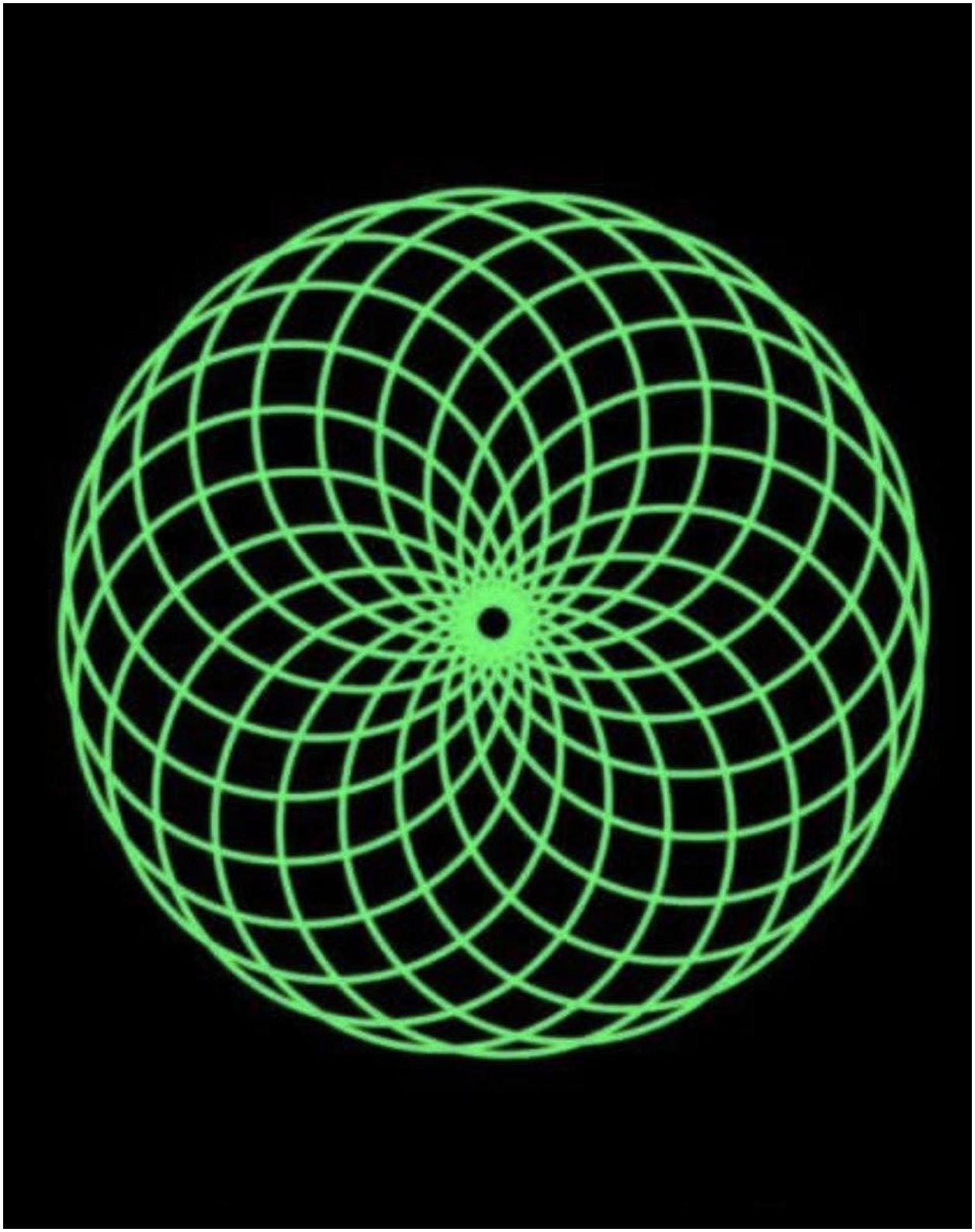
+-Ui [Lambda Xn (1-Xn) less than or equal to E Null Matrix greater than or equal to Lambda Xn (1-Xn)! %] U+-

Remember when I explained the ball and chain concept of the mesh and the object at hand? The object in hand is the objects that are free moving in time throughout chaos. Remember to make your choices wisely and you will all have a direct impact on the reality that the world embraces. That is the value that you all share as FPCs. That is the value of this confederation.

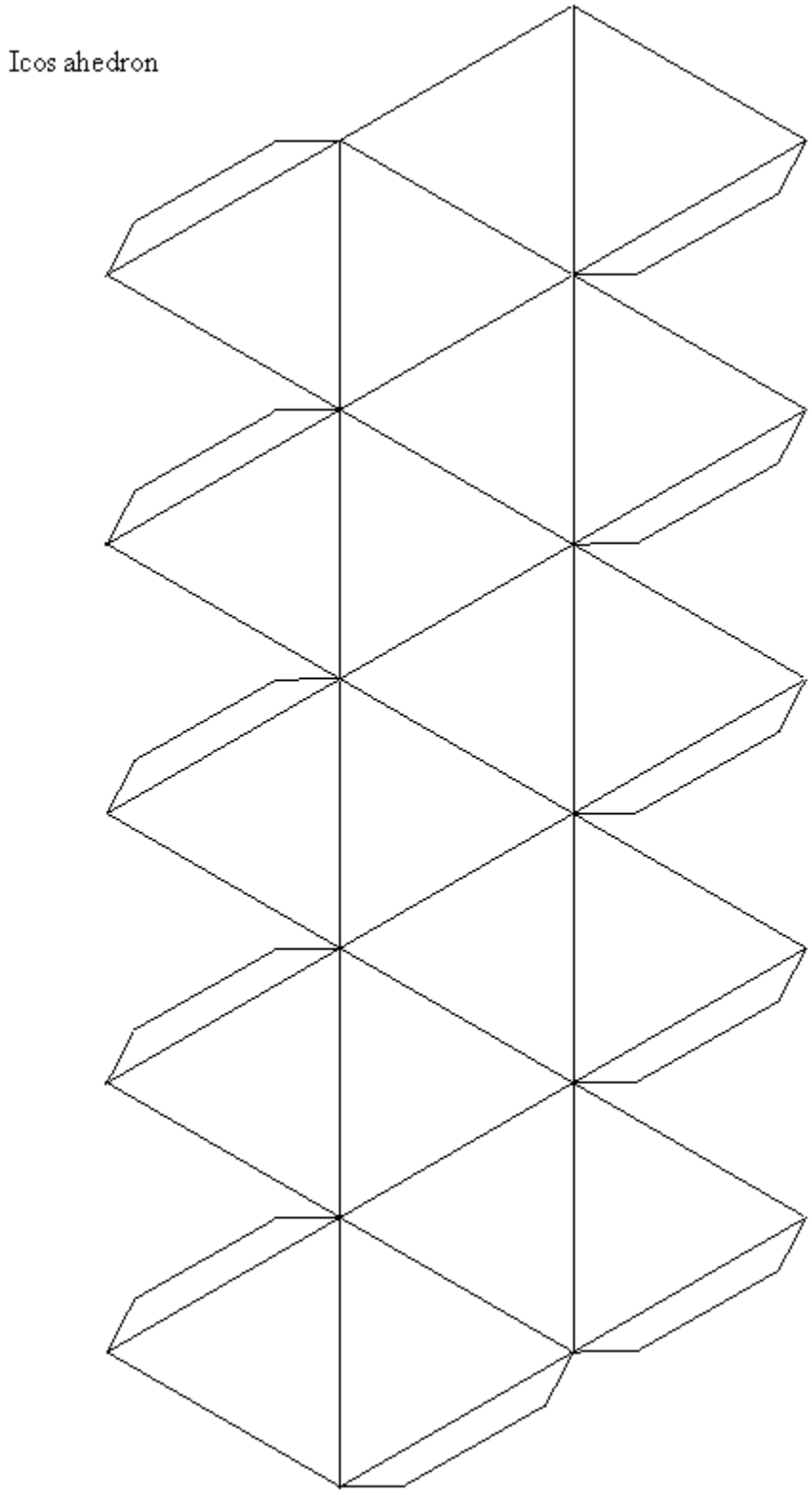
Yours sincerely,

<S. J>

aFPC Design per FCP



Blockchain Mesh



For a Digital Cash Market to function, there are five pieces of infrastructure that need to be built for it to function as its own capital market economy. A DCM needs cash, a large supply in reserve that is managed as a private entity to combat three core causes of financial instability within emerging capital market economies. Combining three core characteristics of world entities that exist today, their natures today are inefficient and negate new information such as the private asset valuation of a nation's natural assets such as natural gas/oil, metals, commodities, minerals, and land.

When a nation faces hyperinflation, liquidity shortage, and a debt-related incident in regard to a nation's stability it will need these five pieces of infrastructure as well as the three core characteristics of the entities. The first entity is the International Monetary Fund also known as the IMF. The DTCO (Digital Currency Assets Reserve) will have a public market-facing infrastructure that allows credit markets and borrowing to occur from the DTCO with conditional private asset trading amongst nations with the purchase of its debt as a truss system amongst functional asset management of a state's capital assets in possession and to be discovered.

Nations/Digital Markets/Capital Systems/Broker-Dealers/Institutions/Companies/Organizations/Communities/Cities to borrow from the DTCO for infrastructure-related development whether that is physical, digital, or occupying an internet-based digital realm such as the Metaverse. The second characteristic is that of the Bank of International Settlements, the BIS. The DTCO is able to integrate credit and debit payments amongst the Central Banks of countries that would like to receive liquidity payments in times of market shortage looking for capital injections of payments and market movements. And last but not least is the World Trade Organization, also known as the WTO. Nations and the above-specified groups will be able to broker all trade between participants and buyers or sellers within the market.

The pieces of infrastructure are relative to the computational power of its systems which entails the ownership of servers to facilitate the market as digital hardware, they are the following. A **Commodity-Backed Reserve Currency (CBRC)** to facilitate currency injections and market movements with capital mobility. For this particular project, it is known as the **Orion System Dollar,** also known as $OSD. There are three core elements that come with $OSD. The first is that it will be pegged to the Swiss Franc as the Swiss Banking infrastructure has withstood the turmoils of the global economy. Dollar in this regard is not compared to the U.S Dollar but is part of our own system where it will be deployed at scale in our own ecosystem where $OSD will be the cash equivalent. Second, it will have a programmable basket that can be modified based on the assets that are in the DTCO. This way the stablecoin is pegged to the Franc but can stand on its own if needed depending on market conditions. The primary deployment will be in foreign exchange markets both traditional and new ones that we create. An instance is to

participate in the liquidity injections by high-frequency trading, at a standard rate of 0.000005% per market transaction.

Second is the **Stock of the Market**, this being the Ozhuman Digital Capital Market (ODCM) also known as $GIC representing the value in the network of the market and not simply the network. The Gideon Infrastructure Corporation. Price valuation is based on the infrastructure and real-time transactional volume that is present in the market at any given time. The stock will begin first as an ERC-20 Securities Token or that of the PolyMesh(Polymath) $POLY standard. Since they are already a registered securities platform. Will need computational engine capabilities to be floated across Binance Smart Chain Ecosystem, Ethereum Mainnet

(EVM Sensitivity, compatible nature that is mutual and mutable naturally matures with organic growth through the Federated Byzantine Agreement (FBA) for social and political maintenance of the network for Nations and the above-mentioned groups. As well as the Proof-of-Stake consensus algorithm (PoS) for facilitating finances that are liquid at all times. It must not be deterministic to the network but based on which it may navigate. These will be the two systems we combine to create a hybrid smartchain that incorporates FBA and PoS.).

Solana Ecosystem, PolkaDot Ecosystem, as well as HeCo, and have the ability to recognize market segments with artificially intelligent neural networks that are publicly available for the most competitive capitalist investments across Nations through FDIs, Foreign Direct Investment, or in our case Direct Digital Investment for Infrastructure, 2DIs.

The **2DI** will be the new Private Bond that is issued through the DCR System. With direct digital deposits in the hands of Member-Shareholders and Validation Brokers via a **Pre-Allocation Investment (PAI)** to all network participants at incremental deposits of *2X for Member-Shareholders* and *4X for Validation Brokers*, **both starting at $350,00 OSD** which is convertible at the point of a transaction on Point-of-Sale systems into the currency of interest starting with the world's primary reserve currency assets.

The United States Dollar, The European Euro, The Great British Pound, The Japanese Yen, and The Swiss Franc. That is $256MM OSD at the end of the eight-month vesting period for Member-Shareholders and $2.048B for Validation Brokers with $1.372Blocked in LP Pools. And $676MM OSD as personal financing accessible after a five-month vesting period. Brokers have a shorter vesting period as they are responsible for brokering trade amongst participants and putting up capital for the financing of trade and market segments' financial activities with an interest-bearing payment of **5% standard plus a 0.0005% transactional fee for engagement.**

Total standard of 5.0005% fee for financial engagement of capital movements of $1B-1T OSD in liquidity for trade and financing. Any additional fees are broker to broker based on the amenities

or features of their trading tools and accessibility to the market system. Member-Shareholders are functioning as Operators taking jobs from $1.00 OSD to $100B OSD for jobs that brokers put up to carry out. Creating a true broker-dealer relationship between Member-Shareholders and Validation Brokers.

The Stock of our market is backed by the blockchain the **Ozhuman Smartchain**, which will be accessible in these markets as well as other private digital markets that will invest and support the Stock of the Orion Digital Capital Market (ODCM): $GIC and the stock of other markets that are built with the ODCM. This creates the world's first independent foreign exchange market that will directly compete with currency reserves across the world. This is the market that represents the viability of the $GIC as a speculative asset that can be invested once listed across markets and on exchanges. Our goal is to directly target Latin America, Eastern Europe, and the South Asian market as a whole. Starting first in India receiving Metals as the first price asset that is backing the currency system as well as the 2DI Bonds that are issued to countries to borrow from the Digital Currency Assets Reserve (DTCO) in turn Nations and the above-mentioned groups will be able to provide private natural asset resources for the new Bio-Fuel/LNG, Metals, Minerals, Commodities, and Land Markets across the ODCM and its trading terminals.

Creating a new range of capital markets that are accessible to nations after the integration of the system of the Ozhuman Smartchain. Which is the third piece of infrastructure that needs to be built, think of the soil that the foundation of a house is built on top of, that is what we are selling. Digital ownership of the Ozhuman Smartchain, $100M per 100TB of server capacity specifically to have a server run with your own ownership of the network system. Starting at 0.01% of the network. That is connected to the trading system of the network, the **Ozhuman Market Terminal (OMT)** all assets that are from the Ozhuman Smartchain, and finally comes the **Ozhuman Digital Wallet**, a decentralized banking wallet infrastructure for the management of assets and investments that is easy to load to your terminal with a single transaction with your assets.

$OSD - Algorithmic Reserve Currency named Ozhuman System Dollar, comes with a Digital Currency Assets Reserve that is managed by Gideon Infrastructure Corporation. Backed by a basket of financial assets first beginning with 2DIs, Metals, Land, and the creation of Bio-Fuel from the refinement and processing of Jerusalem Artichoke an Agricultural Commodity that can be refined into a cleaner alternative of Ethanol without the environmental impact.

$GIC - Stock of the Market named the Gideon Infrastructure Corporation backed by the Ozhuman Smartchain. Representing the Ozhuman Digital Capital Market (ODCM) and the presentation of the market that has been provided here. Either ERC-20 token, or PolyMesh-based securities token "**an institutional-grade permissioned blockchain built specifically for regulated assets**. It streamlines antiquated processes and opens the door to new

financial instruments by solving the challenges around governance, identity, compliance, confidentiality, and settlement.' - PolyMesh website. The eventual goal is to have the token in our own blockchain standard from the Kratos Smartchain.

Ozhuman Smartchain - the foundation of the Orion Digital Capital Market that all digital infrastructure and transactions are computed on. Incorporates Federated Byzantine Agreement (FBA) and a Proof-of-Stake (PoS) consensus algorithm, creating a hybrid blockchain that is the foundation of the Orion Digital Capital Market (ODCM).

Ozhuman Market Terminal - trading terminal for private assets of Nations and the previously mentioned groups. With real-time sentiment analysis from conducting the votes through the FBA consensus protocol. And liquidity pool graphs of real mobility of the market via the PoS consensus algorithm.

Ozhuman Digital Wallet - decentralized wallet infrastructure that has real-time currency conversion at the Point of Sale for all transactions via a digital Debit card that the user is able to determine what asset they would like to use to invest and or spend. Must have loaded up features, in order to send capital into the terminal directly from a phone to a laptop or tablet.

Digital Cash Market Capital Market Economy System notes:

Date - May 11th, 2022 17:26:36 PM

From Professor Aswath Damodaran, New York University Stern School of Business. We learn more about the difference between valuation and price. Price is action-driven liquidity within the market making bets on changes in asset value. Valuation is the price action that is potentially driven by the intrinsic and deterministic values of the asset class itself as well as the assets that compose it. Valuation is a core piece of capital investment theory; it is important to understand the value between valuation and pricing between market systems. The first rule is that if you know the value of something you do not directly present it until you know more of the capital value that comes with the asset or market segment of interest. And in the scope of Gideon Investments, it comes down to the narrative we are creating in this academic, financial, and social investment company. These two elements will be presented at the same time to understand the value of Gideon Investments and the world's first DCM System. The Orion Digital Cash Market. Constructed by Gideon Investments and its Co-Founding Builder [Software Development Company]. The best description of what we are building is the design of the Merkle tree. We are the branch and other markets will be built paired with ours. This was always the mission of operation GAIA, it just took a long time to get all the pieces down on paper and in words to finally understand exactly what it is that we are building here. The world's decentralized, distributed, and democratic financial capital market system with its own capital system allows it to function as a circular economy. This is the big one. A true measure of correction for the financial systems that are in place and do not benefit from the pursuit of

continuously growing is simply not continuous, it requires work after all for it to function in any means as a system.

Valuation Notes:

Valuation is straightforward in reality but we as human beings make it complex for what it is intended to do. There is a significant difference between value and price. Price is determined by the value we believe it can be worth. Value is what it presents to us and we see what Price is the best fit for it as an idea or substance of intrinsic value. Accounting itself as a practice records the past and valuation is to the future. This is where we learn that we should take the accounting balance sheet and contrast it with the financial balance sheet. When it comes to valuation there are four questions that you must be able to answer to some degree even if it is an estimation.

1. What are the cash flows of the existing business?
2. What value are you creating from future growth?
3. How risky are the cash flows?
4. When will the company mature?

To answer the first question the cash flows of Gideon Investments LLC will come from the ownership of assets that are transacted within the network. The liquidity that is moved across market segments that at each point we are collecting on a transactional fee at 0.005% per transaction. A 3.5% tax every year of total cash on hand by each organization and or company that is functioning out of the DCM. This can be in US Dollars, British Pounds, European Euros, Japanese Yen, and Swiss Francs. Cash flows will also be that of the investment decisions between that Gideon Investments LLC and its partners to ensure that the market system is beneficial to all stakeholders looking for a new market segment to invest their assets on hand through our trading system. Regular shareholders will be charged $250 USD or OSD per month for access to the complete trading terminal and $3,000 USD or OSD per year for their validation in the system. Institutions will pay on a case-by-case basis based upon the criteria for their investments and injection of capital into the trading system.

The value that we are creating from the future growth of the DCM is that it comes with open markets that are specific and private to the access of the DCM. From Oil Markets in Angola to Graphite Mines in Sri Lanka, or even Cashews along the Ivory Coast. Giving nations facing hyperinflation, liquidity shortage, and debt-related incidents a new market system to price their assets domiciled in their nations.

The risk that comes with the cash flows of these products through the DCM is determined by market performance and the volatility that comes with a capital system that is open twenty-four hours a day. One core stabilizing tool that we will be using is the 2D-I Bonds, the Direct Digital Investment Bonds that are tied for a minimum of 10 Years all the way to 100 Years. With low-rate interest-bearing payments with yield driven in opposite directions of drive and force creating dual-action bond synergies. With the syndication of price charts and the mobility of

sequential time slots you can use bonds as grease for a well oiled machine in the global economy.

June 20th, 2022
Biofuel Manufacturing and Production Brief

Roughly 60% of petroleum worldwide is consumed by the transportation sector, covering commercial airplanes, freight trucks, ships, and sometimes trains. Renewable fuels such as cellulosic ethanol can be used to reduce overall dependency on petroleum. This is also good to know that low-cost price-stable biofuels offer great opportunities for rural development. "*However, today's relatively low oil prices reduce the cost-competitiveness of cellulosic ethanol which has hindered the development of new energy technologies that need large financial commitments before they mature. Sugars locked as cellulose and hemicellulose in low-energy intensive plants like poplar and switchgrass can be converted by fermentation into ethanol at high yields, and advances in transgenic plants, pretreatment technologies, and fermentation have greatly reduced the process intensity of cellulosic ethanol (Ragauskas et al., 2006). Despite these benefits, the cost of ethanol made from lignocellulosic sources is currently higher than starch-based ethanol derived from food crops, corn, and sugarcane, due to the facile breakdown of starch into glucose*." - (Biofuel Research Journal).

Corn-based ethanol only reduces carbon dioxide emissions by 13% while cellulosic ethanol reduces emissions by 83%. Jerusalem Artichoke tubers carry high amounts of non-structural sugars mainly in the form of inulin. A fructan polymer, easily broken down into fructose and glucose, with the use of microorganisms can be converted into ethanol at high yields. This is why it can be deployed as an energy crop that is high yield and low-cost renewable transportation fuel. Its primary advantage is that it is able to thrive in less fertile land, in saline and alkaline soils. Survival in drought and cold conditions, and the ability to resist pathogens. Creating an energy crop that can be grown in many conditions around the world, so imagine what it can yield in environments that are curated for it to mature and yield at its highest potential. "*Apart from ethanol production, there are several uses of JA tubers and its inulins. Inulins fulfill the role of prebiotics as their β-2,1 linkages cannot be cleaved in the gastrointestinal tract of humans (Bach Knudsen and Hessov, 1995) but can be broken down in the large intestine by bifidobacteria that have inulinase-producing capabilities (Biedrzycka and Bielecka, 2004). These bacteria ferment inulins into short-chain fatty acids (SCFAs) and organic compounds, and their proliferation helps the metabolism and immunity of humans (Pokusaeva et al., 2011). They are applicable as sugar substitutes for people suffering*

from diabetes as inulins can have 30- 50% of the sweetness of sucrose but a very low calorific value (Kelly, 2008). Moreover, they are used as a thickening or bulking agent in foods and find applications in drug delivery (Barclay et al., 2010)." - (Biofuel Research Journal)

The yield of tubers from Jerusalem Artichoke

JA can be planted in spring and shoot emergence can take 2-5 weeks. Tubers formation initiates 5-13 weeks after planting around the same time the plant starts flowering. Harvesting aimed for the maximum weight of tubers can vary between 4 and 9 months depending on the early, mid, or late cultivar. Among the cultivars grown in Montpellier, France, that were planted in mid-March, Fuseau 60 and Nahodka had maximum tuber yield in early October while Violet de Rennes, Kharkowskii, and Medius had maximum tuber yield at the end of November. K8 cultivar had maximum yield in mid-January (Chabbert et al., 1985b). Gunnarsson et al. (2014b) grew 11 clones of JA in Alnarp, Sweden. They planted tubers in mid-May and found that the highest yield of 44 Mg/ha occurred in December. That would be 44 metric tons per hectare. Fresh tuber yields ranging between 61.2 and 88.2 Mg/ha from cultivation in south Italy showed that JA can adapt well to the hot and arid climates. (All of South Asia, Africa, and the Middle East can be deployed for the usage of JA Farming for biofuel refinement because of climate conditions.)

Like South Italy's Mediterranean climate, Bragança, Portugal, also has the same warm and dry summer climate (average summer temperature 18-22°C), where the local JA clone Bragança gave a fresh tuber yield of 65.6 Mg/ha when irrigated with water amount of 460 mm. The highest fresh tuber yield of 30.3 Mg/ha was achieved from growing JA in a semi-arid tropical environment in Chaiyaphum, Thailand, that had minimum and maximum average temperatures of 19.7 and 29.3°C, respectively. JA can be planted at a density of about 40,000 to 55,000 plants/ha. Tuber yield per plant and number of tubers were dependent both on genotype and environmental conditions, but the mass of individual tubers was dependent largely on genotype. In a two-year field trial by Baldini et al. (2006) in three sites in Italy with JA and chicory (Cichorium intybus), one of the sites was in Bari, Italy which has a Mediterranean climate. This kind of climate in South Italy is characterized by hot and dry summer (average temperature of the warmest month over 22°C) and mild and wet winter (average temperature of the coldest month between 18 and 0°C). However, JA had the highest tuber productivity of 80 Mg/ha from this site

where the total rainfall in the years 1999 and 2000 were 240 mm and 400 mm, respectively.

Fermentation of tubers from Jerusalem Artichoke

Ethanol production (Liters/hectare) Reference
3131-7513L Margaritis et al. (1981)
2500-6500L Guiraud et al. (1981)
5509L Sachs et al. (1981)
3900-4500L Duvnjak et al. (1981)
4383-8452L Margaritis and Bajpai (1982c)
5635-9392L Margaritis and Bajpai (1983a)
5000L Chabbert et al. (1985b)
4678L Kim and Hamdy (1986)
6498L Newton et al. (1991)
3060-11,000L Gunnarsson et al. (2014b)

Z. mobilis may have a higher sugar uptake and ethanol yield and productivity, lower cell biomass, tolerance at higher ethanol concentrations, and easier genetic manipulation than Saccharomyces (Rogers et al., 1982; Hobley and Pamment, 1994). However, S. cerevisiae can produce high ethanol yields but can tolerate higher concentrations of inhibitors like 5- hydroxymethylfurfural (5-HMF) (Lujan-Rhenals et al., 2014). With acid or high-temperature hydrolysis, a small portion of glucose and fructose are degraded to 5-HMF but it is less inhibitory to yeast than furfural generated by degradation of xylose which lowers the yield of ethanol in lignocellulosic biomass-based fermentation processes (Sanchez and Bautista, 1988). K. marxianus has lower ethanol yield and tolerance compared with S. cerevisiae (Wang et al., 2016). Inulins from JA tubers need to be broken down into monomeric fructose and glucose as the starting point for their conversion into ethanol. One way is to hydrolyze the insulins using acidic conditions at higher temperatures and the second way employs inulinase to break them down enzymatically prior to fermentation.

Ethanol fermentation of Jerusalem artichoke tuber sugars

Type	Organism	Substrate	% of theoretical ethanol yield	Time (h)	Ethanol productivity (g/L/h)	Other remarks	Reference
CBP	*Kluveromyces fragilis*		98	-	-	-	
CBP	*Kluveromyces marxianus*	Juice extract	97.5	-	-	-	Guiraud et al. (1981)
CBP	*Torulopsis colliculosa*		92	-	-	-	
CBP	*Kluveromyces marxianus*		87.5	30	1.68	-	
CBP	*Kluveromyces cicerisporus*	Unhydrolyzed juice	85.7	30	1.55	-	Duvnjak et al. (1981)
CBP	*Kluveromyces fragilis*		79	30	1.25	-	
CBP	*Kluveromyces fragilis* ATCC 28244	Juice extract	96	-	13.5	-	Margaritis and Bajpai (1981)
SHF	*Saccharomyces cerevisiae* 125	Acid hydrolyzed juice	78	20	-	-	
SHF	*Saccharomyces diastaticus*	Acid hydrolyzed juice	84	20	-	-	Duvnjak et al. (1982)
CBP	*Kluveromyces fragilis*	Unhydrolyzed juice	92	50	-	-	
CBP	*Kluveromyces marxianus*	Juice extract	-	-	104	Immobilized cells	Margaritis and Bajpai (1982b)
CBP	*Kluveromyces marxianus* UCD (FST) 55-82	Juice extract	90	-	7	CSTR	Margaritis and Bajpai (1982a)
CBP	*Kluveromyces marxianus* UCD (FST) 55-82	Unhydrolyzed juice	88	-	-	-	Margaritis and Bajpai (1982c)
CBP	*Saccharomyces rosei* UWO (PS) 80-38		88	-	-	-	
CBP	*Kluveromyces marxianus* LG	Juice extract	98% sugar conversion	-	-	-	Chabbert et al. (1983)
CBP	*Kluveromyces marxianus*	Juice extract	-	-	118	Immobilized	Margaritis and Bajpai (1983b)
CBP	*Kluveromyces marxianus* UCD (FST) 55-82	Juice extract	0.45 g/g sugars utilized	-	-	-	Margaritis and Bajpai (1983a)
CBP	Flocculent cells of *Kluveromyces marxianus* SM 16-10	20% sugars from acid hydrolysis of juice	94	-	17.21-21	Continuous fermentation	Bajpai and Margaritis (1986)
CBP	*Kluveromyces marxianus* IGC2671	215 g/L total sugars	78	30	0.35	-	Rosa et al. (1987)
CBP	*Kluveromyces marxianus*	100-300 g/L sugars from acid hydrolysis of juice	86	24	11	-	Bajpai and Margaritis (1987)
SHF	*Zymomonas mobilis* ZM4F	100 g/L	0.41 g/g sugars	-	67.2	-	Allias et al. (1987)
SHF	*Zymomonas mobilis* ZM4	Acid hydrolyzed juice	78	-	-	-	
SHF	*Zymomonas mobilis* ZM4	Enzymatically hydrolyzed juice	88	-	-	-	Kim and Rhee (1989)
SSF	*Aspergillus ficuum* inulinase + *Zymomonas mobilis* ZM4	-	96	-	3.7	-	
SSF	*Aspergillus niger* 817 inulinase + *Saccharomyces cerevisiae* 1200	Ground tubers	92	15	5.5	-	
SSF	*Aspergillus niger* 817 inulinase + *Saccharomyces cerevisiae* 1200	Juice concentrate	52	72	1.7	-	Nakamura et al. (1996)
SSF	*Aspergillus niger* 817 culture + *Saccharomyces cerevisiae* 1200	Ball-milled tuber flour	80	120	1.3	-	
SHF	*Kluyveromyces fragilis* + *Saccharomyces cerevisiae* Bc16a	Enzymatically hydrolyzed tubers (Rubik cultivar)	86	72	-	-	Szambelan et al. (2004)

jerusalem artichoke biofuel - G | 1362258 | Recent advances in bio-based | article_46482_32f69c91e892 | BioFuel Brief - Google Docs
biofueljournal.com/article_46482_32f69c91e892dd01b39086e3f820acbb.pdf
Bookmarks | Crypto | Uni | Other | Skills Dev | Programming | Ruya | Books | Citizen's Bank | Eastern Banking | Open Culture | Springboard | Electronic library.... | FreeTaxUSA® FRE... | Other Bookmarks
article_46482_32f69c91e892dd01b39086e3f820acbb.pdf 5 / 13 178%

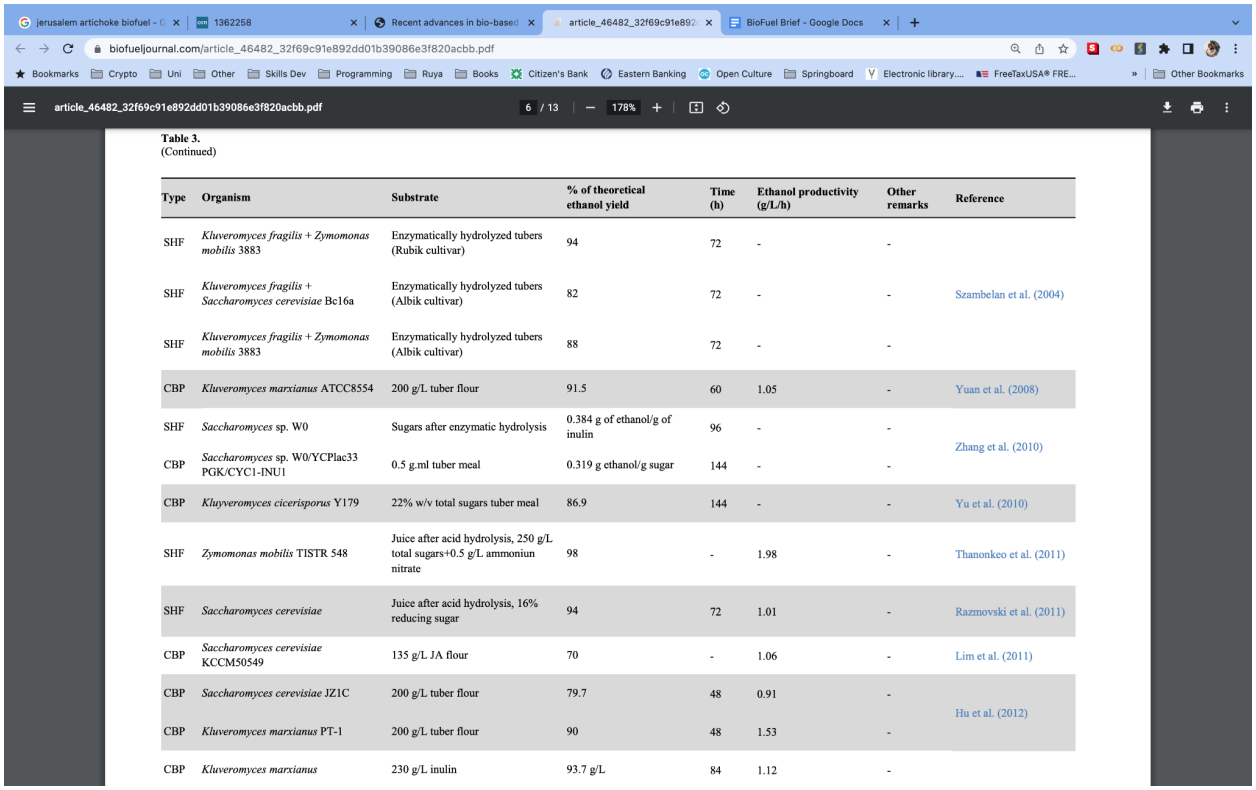


Table 3.
(Continued)

Type	Organism	Substrate	% of theoretical ethanol yield	Time (h)	Ethanol productivity (g/L/h)	Other remarks	Reference
SHF	*Kluveromyces fragilis* + *Zymomonas mobilis* 3883	Enzymatically hydrolyzed tubers (Rubik cultivar)	94	72	-	-	
SHF	*Kluveromyces fragilis* + *Saccharomyces cerevisiae* Bc16a	Enzymatically hydrolyzed tubers (Albik cultivar)	82	72	-	-	Szambelan et al. (2004)
SHF	*Kluveromyces fragilis* + *Zymomonas mobilis* 3883	Enzymatically hydrolyzed tubers (Albik cultivar)	88	72	-	-	
CBP	*Kluveromyces marxianus* ATCC8554	200 g/L tuber flour	91.5	60	1.05	-	Yuan et al. (2008)
SHF	*Saccharomyces* sp. W0	Sugars after enzymatic hydrolysis	0.384 g of ethanol/g of inulin	96	-	-	Zhang et al. (2010)
CBP	*Saccharomyces* sp. W0/YCPlac33 PGK/CYC1-INU1	0.5 g.ml tuber meal	0.319 g ethanol/g sugar	144	-	-	
CBP	*Kluyveromyces cicerisporus* Y179	22% w/v total sugars tuber meal	86.9	144	-	-	Yu et al. (2010)
SHF	*Zymomonas mobilis* TISTR 548	Juice after acid hydrolysis, 250 g/L total sugars+0.5 g/L ammoniun nitrate	98	-	1.98	-	Thanonkeo et al. (2011)
SHF	*Saccharomyces cerevisiae*	Juice after acid hydrolysis, 16% reducing sugar	94	72	1.01	-	Razmovski et al. (2011)
CBP	*Saccharomyces cerevisiae* KCCM50549	135 g/L JA flour	70	-	1.06	-	Lim et al. (2011)
CBP	*Saccharomyces cerevisiae* JZ1C	200 g/L tuber flour	79.7	48	0.91	-	Hu et al. (2012)
CBP	*Kluveromyces marxianus* PT-1	200 g/L tuber flour	90	48	1.53	-	
CBP	*Kluveromyces marxianus*	230 g/L inulin	93.7 g/L	84	1.12	-	
CBP	*Kluveromyces marxianus* K/INU2	230 g/L inulin	96.2 g/L	72	1.34	-	Yuan et al. (2013c)
CBP	*Kluveromyces marxianus*	176 g/L JA flour	62 g/L	48	1.29	-	
CBP	*Kluveromyces marxianus* K/INU2	176 g/L JA flour	69 g/L	48	1.44	-	
CBP	*Saccharomyces cerevisiae* JZ1C	200 g/L inulin	0.34 g/g JA sugars	48	1.19	-	Yuan et al. (2013a)
CBP	*Saccharomyces cerevisiae* JZ1C-inuKM	200 g/L inulin	0.34 g/g JA sugars	48	1.22	-	
CBP	*Saccharomyces cerevisiae* JZ1C-inuCK	200 g/L inulin	0.38 g/g JA sugars	48	1.35	-	
CBP	*Saccharomyces cerevisiae* JZ1C	200 g/L JA flour	0.43 g/g JA sugars	36	1.02	-	
CBP	*Saccharomyces cerevisiae* JZ1C-inuKM	200 g/L JA flour	0.46 g/g JA sugars	36	1.54	-	
CBP	*Saccharomyces cerevisiae* JZ1C-inuCK	200 g/L JA flour	0.47 g/g JA sugars	36	1.62	-	
SSF	*Saccharomyces* sp. W0	25% w/v inulin+0.75% w/v malt extract	11.1 % w/v	120	-	-	
CBP	*Saccharomyces* sp. W0 - *Arthrobacter* sp. Endoinulinase mutant	25% w/v inulin+0.75% w/v malt extract	12.8 % w/v	120	-	-	Li et al. (2013)
SSF	*Saccharomyces* sp. W0	30% w/v inulin+0.75% w/v malt extract	12.4 % w/v	120	-	-	

Please cite this article as: Bhagia S., Akinosho H., Ferreira J.F.S., Ragauskas A.J. Biofuel production from Jerusalem artichoke tuber inulins: a review. Biofuel Research Journal 14 (2017) 587-599. DOI: 10.18331/BRJ2017.4.2.4

Findings from several studies indicate that tuber yields of 9-15 Mg/ha on a dry basis or 30-90 Mg/ha on a wet basis, with a carbohydrate potential of 5-14 Mg/ha can be expected

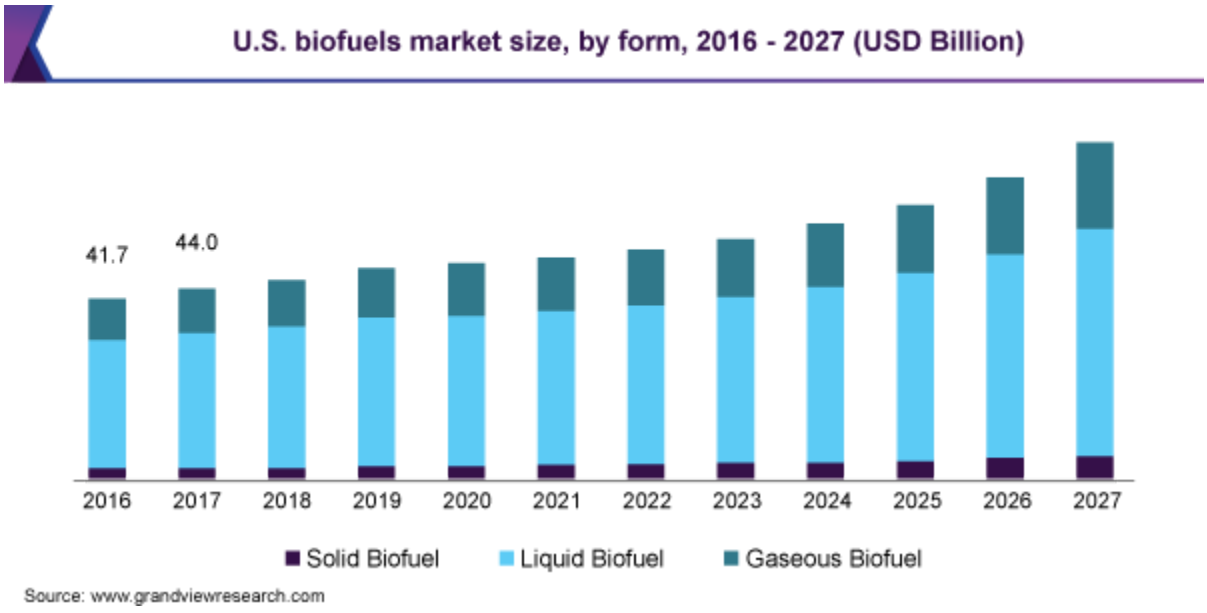
from tubers of JA. Tuber and carbohydrate yields can vary greatly with cultivar, climate, and soil parameters. Tubers can give ethanol yields of 2500-6500 L/ha. By comparison, yields of corn ethanol in the USA and sugar cane ethanol in Brazil can be around 4182 L/ha and 6471 L/ha, respectively. That is on the high end the equivalent of 41 Barrels of Oil, $1,742 at the current market price.

Hydroponic Structure

Aero hydroponic method of reproduction of the original artichoke. A module is a sufficient system of cultivation of plants, with the possibility of operating in natural or artificial environments. The module is mobile and can be equipped with several other modules in a single complex node. Nutrition of plants is differentiated and delivered to the plant roots by aeroponic and hydroponic methods (Hutinel et al. 2012) The roots of the plants grow isolated from the environment, allowing their full-service aero hydroponic system and monitoring of the development of roots and mini tubers. The module provides a fixative device that will keep the plants in the process of growth and development. The main advantages of the module are its mobility and possibility of operation in spring and autumn, in open areas or in greenhouses, eliminates the significant costs of expensive lighting, which is required indoors. The only condition for the sustenance of plants is the supply of water and electricity for pump work (Hutinel et al. 2012). The module consists of the box, inside which aero hydroponic and hydroponic systems flow nutrient solution is mounted. The hydroponic part consists of a hydro reservoir where the lower part of the roots of plants are constantly wetted and, at the same time serves as a storage device for the working fluid, supplied through the aeroponic system box (Figs 3 and 4). The flow of the nutrient solution is a water pump equipped with a filtration system on the inlet and outlet. The aeroponic system consists of a finely dispersed sprayer mounted in the pipe, which is connected to the water pump through the fine filter and provides efficient aeration and oxygenation of the nutrient solution. Further, the liquid after wetting the roots flows back into hydro reservoirs, is mixed with the rest of the liquid, and is prepared for further recycling. Tests to determine the qualitative and quantitative characteristics of the minitubers obtained in the conditions of aero hydroplane were conducted. The experiment was placed on the territory of Vniikh in an isolated environment facility.

Biofuel Market Capitalization

The global biofuels market size was valued at USD 135.7 billion in 2019 and is expected to grow at a compound annual growth rate (CAGR) of 6.9% over the forecast period. Biofuels are the types of fuels that are derived from plants, animal waste, or algae material. They are considered renewable fuels as feedstock material utilized can be replenished at a faster rate as compared to conventional fossil fuels.



Biofuels Market Report Scope

Market size value in 2020	USD 139.5 billion
Revenue forecast in 2027	USD 231.2 billion
Growth rate	CAGR of 6.9% from 2020 to 2027

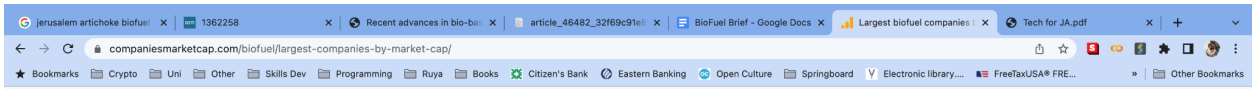
Base year for estimation	2019
Historical data	2016 - 2018
Forecast period	2020 - 2027
Quantitative units	Volume in Thousand TOE, Revenue in USD billion and CAGR from 2020 to 2027
Report coverage	Capacity forecast, company share, competitive landscape, growth factors and trends
Segments covered	Form, region
Regional scope	North America; Europe; Asia Pacific; Middle East & Africa; Central & South America, MEA
Country scope	U.S.; Canada; Netherlands; Germany; France; China; India; Thailand; Indonesia; Brazil
Key companies profiled	Bunge Limited; Archer Daniels Midland Company; Valero Energy Corp.; Petrobras; Butamax; Renewable Energy Corp.; Wilmar International; Algenol; Pacific Ethanol Inc.

North America held the largest market share of 38.6%, in terms of revenue, in 2019 owing to the presence of a large amount of feedstock for biofuel production. The presence of supporting infrastructure for the production and favorable government policies for the utilization of these fuels in the region will result in the growth of the market in the region.

Major growth is likely to be driven by Brazil, the U.S., China, and the European Union. However, other countries have been projected to grow significantly in the industry i.e., India, Thailand, Netherlands, Canada, and Indonesia on account of a rise in awareness to curb carbon emissions and enhance the utilization of the fuels.

Further, Canadian Clean Fuels Standard (CFS) is anticipated to apply various norms for gaseous, liquid, and solid biofuels that are combusted for energy production in the transportation sector. The government of Canada has been concentrating on annual decreases by 30 Mt in GHG emissions by 2030 contributing to the country's efforts to accomplish an overall 30.0% reduction in GHG emissions by 2030.

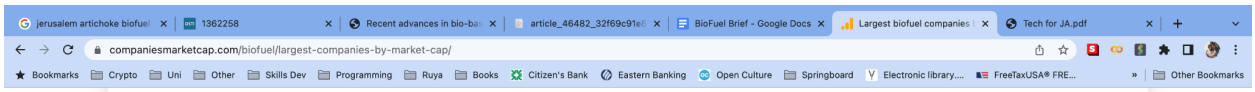
In the Asia Pacific, the market is expected to witness the highest volume-based CAGR of 6.4% in the forecast period as compared to other regions. Introduction and rise in the adoption of biofuel supporting policies and regulations in countries such as China, Indonesia, Australia, and India are expected to boost the demand for biofuels majorly in the transportation sector in the form of blending with conventional fossil fuels.



Largest biofuel companies by Market Cap

companies: **12** total market cap: **$18.69 B**

Rank	Name	Market Cap	Price	Today	Price (30 days)	Country
1	Cosan CSAN	$6.66 B	$14.26	0.42%		Brazil
↑1 2	Verbio VBK.F	$3.16 B	$50.13	5.21%		Germany
↓1 3	Renewable Energy Group REGI	$3.10 B	$61.50	0.00%		USA
4	Montauk Renewables MNTK	$1.89 B	$13.20	7.04%		USA
5	Green Plains GPRE	$1.62 B	$30.25	2.73%		USA
6	Gevo GEVO	$0.58 B	$2.49	3.49%		USA
7	REX American Resources REX	$0.50 B	$84.66	0.06%		USA
8	Tidewater Renewables LCFS.TO	$0.33 B	$9.65	0.80%		Canada
9	FutureFuel FF	$0.31 B	$7.27	0.83%		USA



Rank	Name	Market Cap	Price	Today	Price (30 days)	Country
↓1 3	Renewable Energy Group REGI	$3.10 B	$61.50	0.00%		USA
4	Montauk Renewables MNTK	$1.89 B	$13.20	7.04%		USA
5	Green Plains GPRE	$1.62 B	$30.25	2.73%		USA
6	Gevo GEVO	$0.58 B	$2.49	3.49%		USA
7	REX American Resources REX	$0.50 B	$84.66	0.06%		USA
8	Tidewater Renewables LCFS.TO	$0.33 B	$9.65	0.80%		Canada
9	FutureFuel FF	$0.31 B	$7.27	0.83%		USA
10	Green Plains Partners GPP	$0.26 B	$11.24	1.58%		USA
11	Aemetis AMTX	$0.18 B	$5.47	1.30%		USA
12	Cielo Waste Solutions CMC.V	$0.05 B	$0.08	33.33%		Canada

Countries that Use Biofuel

The most notable countries for the consumption of biofuels (combined diesel and ethanol) are **Canada, the United States, Brazil and Argentina in America. France, Germany, the United Kingdom, Ireland, Poland and the Netherlands in Europe further drawing bond deployments with securitized token offerings to consolidate and expand offerings across market segments in the private markets of economies, public markets of economies, and private-public markets of economies with industrial globalization and no fear of nationalization of sectors in privatization.**

The AES Fund, where we allocate 15% of our revenue.

The Partners' Fund which collects 10% of our revenue and allocates it towards bringing revenue back in the hands of those who bring our firm to life, I do not see the people I work with as employees I see them as my partners in arms to lead the firm forward. I want to make sure that all the people I work with are taken care of and never have to use government services, ensuring they are taken care of with additional quarterly bonuses evenly distributed evenly to everyone within the firm outside of me, even if that means I have to take a pay cut which is perfectly fine to me. I want to make sure first and foremost that my people are taken care of no matter what.

The last fund is The Partner Recession Fund[1], that is allocated an additional 10% of our revenue before expenses to make sure that even if there is an economic recession I can still make payroll and keep my people on board with the pay they have. Even if we bleed a little more I want to make sure that my people are always taken care of, no matter what. I do not want to have to layoff my people because of market negligence and the inability to properly invest in my people. They have contributed towards the building of this company and it is only fitting that I, as its Chief Executive Officer, ensure that my people are taken care of no matter what comes to us. Bringing a total of 35% of pre-expense revenue that is allocated to build in the memory of my sister, grandfather, and velipapa(great uncle); as well as for the protection and preservation of my people. The order of allocation to each fund will be first towards The Partner's Fund, then the Partner Recession Fund, and finally the AES Fund. Which will be delegated by the Board of Directors of The Ozhumanil Group with myself as the Executive Chairman and the remaining Executive Officers and Board Members of the BOD.

The goal of the AES Fund is to reach $500B USD in cash, not in stock, or any other form of assets. Raw cash with the power of the American Dollar. From that breakdown we will have $150B invested into building decentralized crypto full-service hospitals. 25 of which will be built in existence to build a global healthcare medical network that is directly tied to the infrastructure and market capitalization of The Ozhumanil Decentralized Capital Market.

AES Fund Expenditure: $500B USD

$150B USD for 25 hospitals - $6B USD in direct investment per hospital.

$300B USD for AESCare - our own independent healthcare system.

[1] I have also put in place an ESOP of 189M SGM Coin which at the time of our opening will be worth $756M USD. An ESOP is an Employee-Stock Option Pool. This way my people have direct ownership in the market that we are building as well as protection in digital assets against FIAT market rigidity.

$50B USD will be held in reserve.

Each hospital will be built in the real world, of course, but its market capitalization of each individual hospital will be held in its representative crypto-asset within the index itself. Not simply for its financial growth and impact but also its social impact. The biggest requirement that will be heavily enforced by the Ozhumanil Group is that the doctors and staff of the hospital once every two weeks go into neighboring communities and areas that do not have the mobility to travel to the hospitals and bring direct medical care of the highest quality. The same way my sister intends on doing so. This will require real-time updates on the AES Blockchain Protocol that will be updated with anonymous information that does not reveal their name but shows exactly down to the last measurement of what that individual has received in medical treatment, and where, by which staff and doctors that were present as well. Making sure there is direct accountability for the social community medical investment of the AES Fund. This ensures that the return on investment on the index is not simply about the growth in valuation of the hospital themselves but also the social community investment that each hospital is required to make by the bylaws of protocol that will be built on top of the Ozhumanil Financial Network.

AES Hospital Locations

Osaka, Japan - AES Osaka - (Osaka Hospital, OSH Token[2])

Beirut, Lebanon - AES Beirut - (Beirut Hospital, BRH Token)

Dhaka, Bangladesh - AES Dhaka - (Dhaka Hospital, DKH Token)

Cairo, Egypt - AES Cairo - (Cairo Hospital, CIH Token)

Mexico City - AES Mexico - (Mexico City Hospital, MCH Token)

São Paulo, Brazil - AES São Paulo - (São Paulo Hospital, SPH Token)

Istanbul, Turkey - AES Istanbul - (Istanbul Hospital, INH Token)

Riyadh, Saudi Arabia - AES Riyadh - (Riyadh Hospital, RYH Token)

Madrid, Spain - AES Madrid - (Madrid Hospital, MDH Token)

Johannesburg, South Africa - AES Johannesburg - (Johannesburg Hospital, JBH Token)

Montreal, Canada - AES Montreal - (Montreal Hospital, MTH Token)

Casablanca, Morocco - AES Casablanca - (Casablanca Hospital, CBH Token)

Santo Domingo, Dominican Republic - AES Santo Domingo - (Santo Domingo Hospital, SDH Token)

Milan, Italy - AES Milan - (Milan Hospital, MNH Token)

Kuwait City, Kuwait - AES Kuwait - (Kuwait Hospital, KCH Token)

Algiers, Algeria - AES Algiers - (Algiers Hospital, AGH Token)

Dubai, United Arab Emirates - AES Dubai - (Dubai Hospital, DXH Token)

[2] Each hospital will have 500T supply of their native currency for the entire usage of the hospital itself. Medical services, consultations, payment of doctors, payment of staff, surgeries, and in-house pharmacies that do no price gouge medication. This will be strictly enforced by me personally and The Ozhumanil Group as a whole.

Port Au Prince, Haiti - AES Port Au Prince - (Port Au Prince Hospital, PAH Token)

Boston, United States of America - AES Boston - (Boston Hospital, BSH Token[3])

Thiruvananthapuram, Kerala - AES Thiruvananthapuram - (Thiruvananthapuram Hospital, TVH Token)

Minsk, Belarus - AES Minsk - (Minsk Hospital, MNH Token)

Dublin, Ireland - AES Dublin - (Dublin Hospital, DLH Token)

Karachi, Pakistan - AES Karachi - (Karachi Hospital, KRH Token)

Tehran, Iran - AES Tehran - (Tehran Hospital, TRH Token)

Athens, Greece - AES Athens - (Athens Hospital, AEH Token)

Each native currency of the hospitals that have been listed will be directly pegged to the native currency of the OFN, SGX Token. Bringing price stability of the market directly to native currencies of the hospitals directly for their full internal functions within the hospital. The index itself will back AES Coin. AES Coin will have our AESCare healthcare protocol integrated into the specifications of the coin directly as well as a centralized cost adjusted tool that will stabilize and equalize the costs of all interactions within the hospital from the patient to the hospital and in turn the hospital to the patient. Providing full health care for pre-existing conditions as well as mental health care that is ignored by traditional health care companies. I am by no means an orthodox businessman, by every sense of the word my application with how I think is unorthodox and it will upset a lot of people, those who make great returns at the expense of people's suffering. But I am not that person, and there is a long line of people who are upset that they can wait behind, they are behind me for a reason so I pay no attention to the barks of dogs with no bite. Remember, I am not a sheep or a wolf. I am Jonathan Benjamin Samuel. And the bite of my name will bring them all in line.

As the index grows in value so does the purchasing power of the patient to get what is fair within the hospital but also does the valuation of the hospital to innovate internally and provide the best course of treatment with the appropriate form of cost as per the service itself adhering to the OMR guidelines for market infractions that overcharge medication and medical services. This ensures that the patient themselves have more power for their medical health and the financial means to do so. And ensure the hospital itself has investment power to invest on behalf of the patient.

This will be governed by the Chief Executive Officer of each hospital and myself included within the Ozhumanil Healthcare and Medical Commission (OHMC) to discuss and act for medical innovation, treatment costs, expenditure, and operations of the healthcare network. My responsibility is to be surrounded by those who are significantly more intelligent than me when it comes to healthcare administration and hospital management and asking the right questions to ensure that each and every patient that comes through our hospitals receives the best form of care that is possible. And to scrutinize any and all actions that will negatively impact the purchasing

[3] The first hospitals that will be built will be first in Boston and Thiruvananthapuram. To give back to the city that has raised me, I am a Boston Boss after all. And back into our home state that my family is from because I know that is what my sister would have done.

power the patient has within the hospital, with new investment structures and measures to counter them directly.

Changing the relationship of investors and their investments into hospitals and the hospitals relationship between themselves and the patient. When it comes to their health we make sure they have the raw financial power directly to do what is the best course of action by themselves, of themselves, and for themselves.

AES Coin Distribution

Total Supply: 700T AES Coin

300T AES Coin will be held by The Ozhumanil Group as part of our investment into the network.

300T will be distributed as a whole on the entire network publicly available for all users of the Ozhumanil Decentralized Capital Market.

100T AES Coin will be distributed evenly to each nation state that we work with across the 25 hospitals.

4T AES Coins held by each nation-state as their crypto-asset holdings in the joint investment partnership between The Ozhumanil Group and each nation that we approach. This is meant to create a change in the practices of hospitals and healthcare within each nation. Converted into the nation's respective **CBDC (Central Bank Digital Currency)** increasing stability of each country's CBDC and our own CBRC with financial assets backed by value-given capital economic theory to empower the voice of the people; and the further adoption of blockchain infrastructure and decentralization of markets to benefit the people further. [4]

AES Coin will be listed on SGMX directly paired to SGM Coin. The network's liquidating asset.

[4] To the 17th footnote, the list is not permanent and will be adjusted according to the nations that work with us for the benefit of their people and their people's medical health. If a country that is on the list does not want us to partake in such in their country then we will not. We will go to another country that holds the healthcare of their people equally as the health of their nation. Boston and Thiruvananthapuram are the only hospitals that I will be adamant on building towards. I will not forget the promise of serving my people in the city I was raised in, as well as my people back in Kerala.

SGMX Tokenomics Report

"Decentralized Global Markets that provide large scale operability for all users to produce, trade, invest, monitor, barter, or participate in the flow of crypto assets to create capital gains on digital assets within a socio-economic market. The network itself becomes a digital economy of scale that invests back into communities via SGMX, an investment service marketplace. And other ISMs are building liquidity pairings to strengthen and grow the network further."

Covering the operability of SGX Token and SGM Coin within the market network.

By Dr. Jonathan B. Samuel Ph.D

Patent and Disclaimer

The invention disclosed in the Tokenomics report is the subject of pending patent applications.

This Whitepaper may contain "forward-looking information". Forward-looking information statements may include, among others, statements regarding the future plans, costs, objectives or performance of **Samuel Global Management LLC**. (the "Company"), the ecosystem or the platform or the assumptions underlying any of the foregoing. In this Whitepaper, words such as "may", "would", "could", "will", "likely", "believe", "expect", "anticipate", "intend", "plan", "estimate" and similar words and the negative form thereof are used to identify forward-looking statements. Forward-looking statements should not be read as guarantees of future performance or results, and will not necessarily be accurate indications of whether, or the times at or by which, such future performance will be achieved. The actual results of the Company, the ecosystem and the platform could vary from the forward-looking information contained herein, including as a result of such risks as a collapse in the market for cryptocurrencies, adverse regulatory developments and competition from other platforms. Forward-looking statements and information are based on information available at the time and/or management's good faith belief with respect to future events and are subject to known or unknown risks, uncertainties, assumptions and other unpredictable factors, many of which are beyond the Company's control. The forward-looking information contained herein was developed based on assumptions related to, among other things, the continued growth of the blockchain technology industry, the success of the participants in the ecosystem and the demand for such participants in the ecosystem and the demand for such participants' product offerings. The Company does not intend, nor does the Company undertake any obligation, to update or revise any forward-looking information or statements contained in this Whitepaper to reflect subsequent information, events or circumstances or otherwise, except if required by applicable laws. The Whitepaper contains statistical data, market research and industry forecasts that were obtained, unless otherwise indicated, from independent industry and government publications and reports or based on estimates derived from such publications and reports and the Company's knowledge of, and experience in, the sectors in which the Company plans to operate. While the Company believes this data and information to be reliable, market and industry data and information is subject to variation and cannot be and therefore has not been verified due to limits on the availability and reliability of raw data, the voluntary nature of the data gathering process and other limitations and uncertainties inherent in any statistical survey. The Company has not participated in the preparation of such information contained herein.

Table of Contents

Introduction

The purpose of SGMX was to create a forward thinking model towards creating an independent and conscientious digital market. A market that is democratically governed hand in hand with the market players and marketplace users. One that is built to provide the same potential for those who interact directly with the marketplace itself. Providing fixed cost of engagement to build a cost-minimal market that maintained market malleability with its cost and decreased over time within a proportional relationship with financial behaviors of the market to provide a fair and innovative competitive space that promotes innovation in users, companies, and the public as a whole.

Growing within transactional volume in a combination of monetary value, both digital and fiat across goods, products, services, and assets that are exchanged on the platform within the network. Tied directly to the volume of economic equity the users themselves maintain ownership within as well as interact within the network on. Ensuring that as the cost continues to decrease with your engagement as a market user increases in volume as the marketplace as a whole increases in valuation, and your overall expenditure to operate within it decreases.

Increasing the economic and financial activity within the network as a whole. A marketplace that is backed by your creative investment within it as well as directly facilitating the demand at scale within the network itself. Cheaper for you over time, because it's backed by growing infrastructure and maintaining a fair marketplace for all users to interact with. That is the Investment-Service Marketplace of Samuel Global Market Exchange (SGMX) that is facilitated on a decentralized socio-economic crypto-asset exchange. As the network itself grows independently, so does the liquidity of your interactions to create capital gains. Increasing at scale the potential for all interactions amongst users within the marketplace itself. An investment vehicle that facilitates broadscale economic growth as the economic growth of its users grows in parallel with it.

Part 1: SGMX Infrastructure Backed Decay Rate, the outcome of the OMEC Rate

The IBDR that is programmed into the native currency of SGMX the Investment-Service Marketplace of The Ozhumanil Group helps define the OMEC Rate. OMEC stands for the Ozhumanil Market Exchange Cost. The IBDR is programmed as one of the main specifications of the SGX Token. The native utility token that functions as the network currency to use the exchange system and the network as a whole. ***The IBDR is calculated by the GAINS Algorithm, Global Analysis for InterNetwork Synergy.*** Which calculates the progression of growth for the infrastructure of SGMX and in turn the network as an extension of the exchange that it is built on top of. The composition of the GAINS Algorithm is the SEDI Index which keeps progress of the financial transactional behavior to gauge and provide a fair and ethical digital capital market with its own independent infrastructure, markets, investments and assets. Then follows the transactional monetary volume that is shown in real time on the exchange, which will later also include the real time monetary volume of the main network itself as well, the OG Hashgraph of The Ozhumanil Group. And finally the transactional economic equity of infrastructure that is maintained, used and interacted with by the users of the network who also have access to it as well. **This equation creates the transactional volume of economic capital facilitated through decentralized finance strengthening independent capital ownership.** Taking into consideration each target goal for the network towards its real time market valuation and promotion of our own decentralized liquidity pool that can be used to facilitate independent liquidity for consumers, companies, and countries.

[IBDR] (“OMEC Rate”)

- **{GAINS Algorithm}**
  - **(SEDI Index**
    - **Transactional Monetary Volume**
      - **Transactional Economic Equity)**

➢ In consideration with real time adjustments to target goals that will grow network valuation and decrease network operation cost of engagement overtime within the market system. The algorithm will be grown within an artificially intelligent environment to assess its real time change with the targets of each in mind.

Part 2: Ozhumanil Market Exchange Cost Rate (OMEC Rate)

The OMEC Rate is the name of the standardized cost to operate within the marketplace. The marketplace opens with all financial transactions at 0.09% fee per transaction. This transaction fee is locked as the fixed rate for all financial transactions irrespective of the context of the

transaction itself. If a company is listing their products, goods, or services on the marketplace it will cost only 0.09% per item that is listed. If a consumer is interacting with the company to purchase its products, goods, or services each transaction of said item will only cost 0.09% The same cost is applied for transactions of digital assets, liquidating out of the market, any application within the market. This transaction percentage is calculated by the percentage dropping at a specific rate in accordance with the Socio Economic Developing Investments Index of SGMX, its transactional monetary volume, and transactional economic equity.

The index itself is needed to dictate the speed of the cost itself decreasing in accordance towards the health and financial behavior of the marketplace itself. The index is scored out of 1600 total basis points, with the highest mark of 1600 showing that the market itself is being held within compliance towards the socio-economic governance protocols of the SGMX marketplace. **The index falling below the 1600 point score will result in the next integer of the cost of transaction achieving a delay in decreasing towards the next integer.** The index is the first application towards assessing the OMEC Rate within the market. It works within SGMX to show the quality of financial interactions and behaviors within a digital capital market that aims to provide equal opportunity for engagement and investment to its users. That being companies, institutions, governments, and the general public at large. *It is only fitting to decrease the cost of engagement within the marketplace held against the cost of our engagements amongst one another.* **Positively reinforcing capital market interactions amongst companies and individuals and vice versa and or amongst institutions to be within a socially conscientious environment that sustains decentralized dispersed and democratic economic opportunity for all of its user bases.** This is then met hand in hand according to the real time transactional value of monetary volume and transactional equity of shareholders for economic capital created through market's infrastructure and digital assets. Three independent linear functions that run in parallel to each other and dictate the cost of the market itself which is held in SGX Token as the native currency for engagement within the independent capital market. This is what the Global Analysis of InterNetwork Synergy Algorithm calculates and then applies within the SGX Token. The algorithm is one of the core specifications of the SGX Token of SGMX. **These three elements work in accordance to their real time target goals of the network in proportional relationship to the time delegated towards the return on investment of the network as a whole.** With SGM Coin as the liquidating asset of the network and SGX Token as the principal digital currency of the network, backed by real time valuations in infrastructure, transactional monetary volume, and transactional equity of users. This brings to question the cost of transactions within a digital capital market that is always engaged with the users when there is a market correction within the network. The lowRateLock of the SGX Token ensures that *once the rate is lowered it cannot be increased even during a market correction of the network.* **This ensures that the market maintains a cost-minimal engagement structure to sustain high-volume of engaged users who are trying to continue the growth of their capital assets even with a much more conservative financial approach of their investments and usage of**

the network. This works by keeping on track with the three core elements of the algorithm mentioned beforehand, once the correction has been adjusted the algorithm will analyze network performance to gauge its direction towards meeting the targets of each category. Improvement in financial market behavior, transactional monetary volume, and transactional equity of which it will then continue its course towards reaching the final target for cost of engagement within the marketplace at 0.0009% per transaction.

Part 3: OMEC Rate Target Brackets for the Network

The internal health index is part one of the three variables that influence the functionality of the algorithm. Which controls the OMEC Rate of the network. ***(The OMEC Rate would be the same rate that is integrated for other ISMs that are built, maintaining a stable decentralized liquidity pool that is integrated within established fiat capital markets bridged by the ISMs into their own digital capital markets. This enables the entire network to have the same cost of engagement irrespective of which ISM is being used, providing long term market stability.)***

SEDI INDEX BRACKET

1550-1600, 1500-1600, 1450-1500, 1400-1500, 1350-1400, 1300-1400, 1250-1300, 1200-1300, 1150-1200, 1100-1200, 1050-1100, 1000-1100, 0950-1000, 0900-1000, 0850-0900, 0800-0900, 0750-0800, 0700-0800, 0650-0700, 0600-0700, 0550-0600, 0500-0600, 0450-0500, 0400-0500, 0350-0400, 0300-0400, 0250-0300, 0200-0300, 0150-0200, 0100-0200, 0050-0100, 0000-0100, 0000-0050, 0000-0000 **(these brackets that the index scores in affects the rate of change on the cost of engagement itself till it achieves 0.0009% per transaction)**

Transactional Monetary Volume Targets

0-100K M.Volume, 100K-1M, 1M-10M, 10M-100M, 100M-500M, 500M-1B, 1B-5B, 5B-10B, 10B-20B, 20B-40B, 40B-80B, 80B-120B, 120B-240B, 240B-480B, 480B-960B, 960B-1T, 1T-5T, 5T-10T, 10T-30T, 30T-50T, 50T-80T, 80T-100T, 100T-150T, 150T-200T, 200T-350T, **350T-500T M.Volume (Point of when cost of engagement is 0.0009% per transaction)**

Transactional Economic Equity Targets

1M-10M, 10M-100M, 100M-1B, 1B-10B, 10B-100B, 100B-1T, 1T-10T, 10T-100T, 100T-1Q***(Quadrillion)***, 1Q-10Q, 10Q-100Q, 100Q-1QT***(Quintillion)***, 1QT-10QT, 10QT-100QT, 100QT-1ST***(Sextillion)***, 1ST-10ST, 10ST-100ST, 100ST-1SP***(Septillion)***, 1SP-10SP, 10SP-100SP, 100SP-1QT***(Octillion)***, 1QT-10QT, 10QT-100QT, 100QT-1NO***(Nonillion)***, 1NO-10NO, 10NO-100NO, 100NO-1DN***(Decillion)***, 1DN-10DN, 10DN-100DN, 100DN-1UD ***(Undecillion, 36 zeros)***, 1UD-10UD, 10UD-100UD, and finally **100UD-1DC E.Equity (Duodecillion, 39 zeros, point of when cost of engagement is 0.0009% per transaction.)**

The change in the brackets of the index score influences the rate of change of the cost of engagement of the marketplace itself. The higher bracket that is maintained for a consecutive amount of time ensures that the rate towards decreasing the cost of engagement is maintained. Each change in 50 or change in 100 basis points of the index triggers the slowdown of the OMEC Rate decreasing *(decreasing the OMEC Rate in accordance with the rate of change of the financial behavior health index to the transactional volume and transactional equity of the exchange network.)* The longer period that the market cohabitates to perform at its highest degree the faster it becomes cheaper for all user bases to engage with the market with proportional rate towards the growth and valuation of the market as a whole.

The index maintaining above 1500 will cause the OMEC Rate to continue the rate of which it is decreasing based on the financial behavior that is happening on SGMX the rate will also be the same that is applied for the sister exchanges of SGMX, all exchanges will maintain and stabilize the decentralized liquidity of the independent global marketplaces held on the network. The closer to 1600 as well as the time it is maintained as well will speed up the decay rate of the OMEC Rate. This is why the composition of the financial behavior within the market is crucial from the SEDI Index on SGMX. The only real market dictating element is the engagements of financial behavior within the marketplace and that is the only element of this project that will inhibit or develop the purchasing power of the user bases with their financial transactions within the market.

Part 4: Volume Backed by Infrastructure (CFI Projects)

The CFI Projects are the infrastructure projects facilitated and created by the revenue of SGMX that is exclusively built on the main network that hosts SGMX. This promotes the real time market capitalization of the projects to promote the liquidity of the network itself which in turn grows the valuation of SGX Token as it is backed by the valuation of the network as a whole. The collective market capitalization of exclusively funded and created projects on the main network supports the native currency that is used to interact with the network growing in parallel to the valuation of the network. Increasing the purchasing power of the users with the native currency of the network as well as their ability and accessibility to alternative digital asset classes that are held on the network. The CFI Projects also have a minimum 15% investment equity threshold that is held by Samuel Global Management LLC which increases the assets under management of Samuel Global Management. The two core specifications of SGM Coin is that it is backed by the AUM of SGM as well as its market performance as a blockchain-focused alternative investment firm. This maintains the stability of SGM Coin as the market's primary liquidating asset **(other liquidating assets of the market would be picked based on their existing infrastructure towards strengthening the market as a whole as well as expanding its potential by deepingin its liquidity and interoperability with existing frameworks**

coupled with developing frameworks on the network itself). Our own independent asset class of the network that grows in parallel to the currency of the network and its infrastructure. This creates two powerful digital assets for the user bases. A liquidation asset to transact out of the market that is backed by the infrastructure and assets under management of the firm itself and its performance; and a currency asset that increases in user purchasing power because of the hard infrastructure, continuation of healthy financial market engagements, transactional monetary volume, and transactional economic equity. Strengthening the network's overall valuation and in turn the capital mobility of its user base.

decreases in accordance to the market achieving these targets within the 10 year goal as the return on investment of the network as a whole The bracket below is the primary target towards the valuation of the network within a 10 year rate of return.

1T USD - 30T USD { [0.015% - 0.019%] & [0.010-0.015%] }

- This is the real time valuation of the transactional monetary volume of the network as well as its transactional economic equity of its users on the network. The cost of engagement would be within these brackets when maintained with a healthy financial market and infrastructure being built on the network will support this overall nominal valuation of the network within a ten year timeline. The remaining rates below are for the brackets of valuation of the network and the rate at hand in accordance to the network. The following brackets are based on the rate of return of the network once achieving the 10 year target and onwards.

  30T USD - 60T USD { [0.005%-0.010%] & [0.001%-0.005%] }
  60T USD - 120T USD { [0.0009%-0.001%] } real time worth of network, not including the valuation of monetary volume, or economic equity of users. This is the appropriate market-adjusted valuation target of the Ozhumanil Decentralized Global Market to when the OMEC Rate drops to 0.0009% per cost of engagement.

{ [P1(Financial Behavior) + P2 (Transactional Monetary Volume) + P3 (Transactional Economic Equity of Users)}

Simplified as { [P1+P2+P3] } is a compilation of the GAINS Algorithm, this is then taken into accordance with their respective targets of each category divided by time in 10 years for the basis of return on investment of the network. ***("Ozhumanil Decentralized Global Market, ODGM")***.

{ [P1+P2+P3] } / (time*10 Years) } = rate of change against the OMEC Rate over time to achieve our targets which furthers focuses towards strengthening the decentralized liquidity pool.

EX = existing potential market capitalization valuation of companies and market infrastructure

YM = accessibility for public wealth to integrate within existing potential market cap valuation
CM = Establishing the financial market that is in place today. .

{CM = EX + YM}

SX = SGMX collective infrastructure and market capitalization valuation of companies and market infrastructure.
ZM = SGMX user's ability to interact with the yield-potential within the marketplace.
NM = building a new independant market. "ODGM"[5]

{NM = SX + ZM}

Part 5: Comparative Theoretical Yield of Decentralized Global Market

{ (X + YM = CM) / (SX + ZM = NM) }

This relationship shows the yield in comparison to the established market as is compared to the gain of the Ozhumanil Decentralized Global Market. This is where we build the Comparative Theoretical Yield of a Decentralized Global Market (DGM, the first being the Ozhumanil Decentralized Global Market with its independent hashgraph)

Comparative Theoretical Yield of a Decentralized Global Market = (CM/NM)

NM Yield (New Market Yield) target against CM (Current Market Yield) should be for (1:2) and (1:3) respectively. Based on the market performing well, reducing the cost for users to operate on SGMX, by lowering the OMEC Rate. But done so in accordance towards growing the DGM (Decentralized Global Market) as a whole with its infrastructure, transactional monetary volume, and transactional economic equity of the users. Benefiting the users as a whole as well as the stability and growth of The Ozhumanil Group as a global financial services institution with its own Decentralized Global Market, the "Ozhumanil Decentralized Global Market".

Part 6: GAINS Algorithm Programming Structure

[5] The Ozhumanil Decentralized Global Market is the objective goal of SGMX and the network itself. It is named after the Ozhumanil Family that SGMX Founder Jonathan B. Samuel comes from. Where the market is decentralized with its own liquidity pool built on the OG Hashgraph which functions the interactions of the Ozhumanil Decentralized Global Market. Facilitating in exclusivity all of its exchanges, companies, brands, ventures, products, and services facilitated by capital from the Ozhumanil Group. Integrating directly within other capital markets and growing further within them as well. That is publically available to be shared with its users strengthening it further. Where the users can independently be a part of as well sharing its liquidity.

```
Package test;

public class test

{
        private long time;
        private static double IBDR \\infrastructure backed decay rate, the physical # for the
change in the cost of engagement created from the computation of the GAINS Algorithm\\
        static boolean redo = true;
        private double lowRateLock;

        public static void main (string [] args)
        {
                IBDR = gains();

        }

        public static double gains()
`       {
                if (x <= lowRateLock)\\this is updated with the new lowest rate of engagement on
the network and cannot be increased once lowered. Once it is lowered it becomes the new
standardized cost of engagement for the entire network, across all other ISMs as well.\\
                {
                        do
                           {
                             "x+=9" (symbolic formula that shows the computation of the variance
of the three core variables of the algorithm , the SEDI Index, transactional monetary volume, and
transactional economic equity of the network)
                           }while (x<= lowRateLock);
}
        else
            } do
         }
                if (sedi index () && redo ) \\completing the bracket of the performance of the
index itself updating in real time to the performance of the market as a whole. Taking into
account of the four core elements of the index Public Market Health [400/400], Global Market
Health [400/400], Corporate Financial Health [400/400], and Public Financial Health
[400/400]. Comparing to the brackets of change for the speed towards the rate of change for the
cost of engagement of the market.\\
        {
```

```
            if (Transctional_MVolume () && redo) \\takes into account the real time
transactional monetary volume of the exchange and the network at large towards meeting the
targets of the transactional monetary volume in hand with the ten year timespan for the return on
investment of the network.
        {
            if (Transactional_EEquity () && redo) \\takes into account the real time
transactional economic equity of the users that is being exchanged on the network as well as the
exclusivity with the market capitalization valuation of the infrastructure, asset, and or investment
on the network itself.
        {
            redo = false;
        }
}

}

} while (redo);

}
        return (rate of change towards the cost of engagement);
}
        public static boolean sedi ()
        {
            return true; \\ensuring the market is at the bracket that it is in real time\\
        }
        public static boolean transactional_monetaryvolume()
        {
            return true; \\ensuring the market is at the real time monetary volume that is
stated within the exchange itself and the network as a whole.\\
        }
        public static boolean transactional_economic_equity()
        {
            return true; \\ ensuring the market is at the real time economic equity that is stated
        within the exchange itself and the network as a whole. These three variables are
        continuously updated within the performance of the market throughout the course of 24
        hours of market performance.\\
        }
}
```

Part 7: Specifications and focuses of SGX Token and SGM Coin

The core specifications of SGM Coin is (1) it is backed by the crypto reserve of SGM Coin and SGX Token that is held by Samuel Global Management as one of its core crypto assets under management. As well as additional blockchain, decentralized, index, infrastructure, and beta crypto assets within the space. (2) It has grown in valuation in accordance with the performance of Samuel Global Management as a blockchain-focused alternative investment firm with concentration towards renewable energy, financial assets, and emerging technology. The focus of SGM Coin is to serve as the principal liquidation asset of the Ozhumanil Decentralized Global Market. SGM Coin will become the digital asset that all other assets will liquidate out of. (3) All pairings on SGMX will be created with SGM Coin on the ISM. As well as with other trading pairs that will be exclusively added into SGMX as our liquidation partners as well.

SGX Token will serve as the native currency for the entire network. (1) With its core specification being the IBDR to reduce cost of operations for users on the network by strategically comparing it against the liquidity of the network, its monetary volume, and economic equity. Other specifications of the SGX Token also include the following. (2) Speed and security of transactions within the financial network. (3) Built in fill-in smart contracts tied to the SGX Token which can be used to facilitate trades of goods, products, and other assets on the network. (4) Cross-border-settlements can be used with SGX Token on the network. (5) End-to-end transparency on the network via SGX Token, it is able to show the users with anonymity the flow of financial capital on the network. Raw real-time democratized market flow of capital that is publically available to all ISM users showing them the market volume as well as central areas that have the most density of market volume on the network. (6) SGX Token will only be paired with the other native currencies of other ISMs to build immediate liquidity pairing within the network. Bridging the flow of FIAT capital via the ISM into crypto-assets. And in turn growing the liquidity of the decentralized network further within other countries.

Part 8: SGX Token application and valuation backing

SGX Token is used for all of the interactions within the exchange system. SGX can be used to purchase goods, products, and services on the marketplace. It can also be used to purchase the altcoins of companies that are listed on the exchange as well. Any interaction of payment that is on the exchange is facilitated, carried out, and settled by SGX Token. Its valuation is grown in proportional relationship to the valuation of the network that uses SGX Token, the hosts SGMX on top of it. The valuation of the network as well as the valuation of the exchange itself.

Part 9: SGM Coin application and valuation backing

SGM Coin is used for all the interactions within the exchange system to liquidate digital assets on the network as well as within SGMX. It is backed by the AUM of SGM, SGM's performance, as well as the higher the rate of usage of SGM Coin the higher the increase in price valuation.

Representing a concrete digital asset that is used and used to liquidate out of because of its real market valuation as an asset on the network.

Part 10: Independent Decentralized Liquidity via a Decentralized Global Market

The structure of SGMX fueled by the specifications of SGM Coin and SGX Token allow two tangible assets to be fueled by the growth of SGMX. SGM Coin serves as the liquidation asset backed by crypto and FIAT assets held by Samuel Global Management on the network. SGX Token is used as the currency asset that grows in purchasing power as the cost to use it decreases backed by its proportional relationship to the growth of the network as a whole. These two core assets allow us to enable two strategic elements. SGM Coin is used to pair other companies, institutions, and entities that are seeking to provide their goods, products, services, and or investment capital on the exchange. Establishing the market trading pairs between Samuel Global Management and market operators on SGMX. Growing the valuation of marketplayers and SGM hand in hand. SGX Token giving a digital currency for payment of the said companies that list on the exchange. Where users can directly interact with the companies and or institutions and purchase services, products, or goods and directly invest into their altcoins respective to their companies and or institutions. This allows the currency asset to drive further revenue via hard assets companies provide as well capital gains on users purchasing their altcoins to invest into the companies further. This strategic balance between SGM Coin and SGX Token allows the network to grow further and establish independent decentralized liquidity that is exclusively tied to our own. And in turn allowing the network to grow further becoming the Ozhumanil Decentralized Global Market.

The GDX Index

Socio-Economic Governance via SGMX GDX Index

The GDX Index is built as a way of showing the marketplaces health, gauging its collective ability towards providing opportunities that drive socio-economic growth for all of its three core client bases.

2.1 SGMX Core User Bases
Institutions, Companies, and Investors.

Institutions: Being able to invest into a marketplace that is driven on impact investing that can still yield sizable returns on their investment. Regardless of the three core buying persona every interaction on SGMX costs a standardized transaction fee of 0.09%, regardless of the size of your investment. As long as you have internet connectivity you can access the SGMX investment-service marketplace. Being reached out to by shared partners that become testimonials on the quality of their capital returns from an ethical digital marketplace.

Companies: No matter what product they list, regardless of the service, good, or anything else in between from these companies the same 0.09% rate still stands. We want to facilitate an open-market interaction with as minimally costly to the companies that interact with an exchange.

Ex: **Amazon**, a consumer goods company creates a variety of home electronic goods that service a multitude of clientele. For example, let us use Alexa.

Since November of 2014, Alexa's release date, 100M Alexa home units have been bought.
Let's set for this problem base that the average across all Alexa model's the **average price is**

$55.00 USD.

November 2014 to August 2019 is 4 Years and 9 Months which is 1,734.90044 days we will just say 1734 for the sake of this example.

Against 100M Units sold from November 2014 to August 2019 that is, for the sake of the example, **an expected 57,670.1268 ≈ 56,670 units in a day.**

With the average set to:
$55.00 per Alexa home model set, **56,670 sold in a day** = **$3,116,850 Unit Revenue**
November 2014 - August 2019 Unit Total Hypothetical Revenue = **$5,404,617,900 USD**

Now for this we do not know of all the factors that played into how much Amazon as a company was able to walk way with. This doesn't mean that we can't have the theoretical value of Amazon played out in the SGMX Marketplace.

With the standardized 0.09% Market Engagement Fee per transaction,

Per Day Yield with SGMX in the equation: **($3,116,850) * 0.09% = $280,516.50 USD** [6] **Which converts into SGM Coin as the dollar equivalent.**[7] This is the yield for SGMX and the interaction between the company for operating within the market.

Putting the **daily yield for Amazon at $2,836,333.50 USD**, this being the revenue yielded for the **Per Day Alexa Purchasing within SGMX**, that if applied to Amazon from the release of Alexa to our present set in time would be a total of

$4,918,202,289 B USD = in revenue generated within that time period, with only a single point of cost to operate within the marketplace.

This in turn, if SGMX was fulfilling the transaction via the marketplace results, as per this model would be,

$442,638,206.01 M USD in revenue over a **5 year period,** for a single corporate entity with a single product transacted within a minimal cost-to-entry investment-service marketplace.

Average of $88,527,641.20 M USD Per Year

For the transactional revenue created by partnering with a single large scale entity to operate with us within the marketplace. This is the same potential of yield that stands for any large scale enterprise that approaches SGMX and lists their products, goods, or services in the marketplace. The same rate applied for SMEs as well as multinational corporations, for the same potential in revenue generated by the marketplace.

Directly purchasing them with SGM Coin, as well as investing into SGM via SGM Coin. Select cryptocurrencies will be partnered with that can be used as the same means of transaction or to purchase within the market. The standard coin of usage begins first with SGM Coin and then

---

[6] This rate is called **OMEC**, **Ozhumanil Market Exchange Cost.** It is the determinant of transactional cost and the yield for engagements to SGMX, a subsidiary under The Ozhumanil Group.

[7] The rate is the only cost for the companies, institutions, and users to operate within our marketplace. The more cooperatively that companies, institutions, and individuals work together this will trigger the rate lowering itself till we reach **our target OMEC Rate of [0.015% - 0.025%]**. We designed this to foster large scale socio-economic development promoted on a minimal cost-to-entry investment-service marketplace that benefits all of its users.

with our core partners. They will be selected on a per project basis, projects that build blockchain infrastructure that facilitates the SGMX capital market growth.

Our project is built towards mass adoption as a PRINCIPAL CRYPTOCURRENCY MARKET that any individual or entity can interact with.

And finally the most important client persona, is the general public as the informed capital motivated investor. Our core focus of SGMX is financial literacy. We want to make sure that we not only provide you an economy, but an ecosystem built with you through a platform that values you. That is designed to be built in your favor on informing you. Helping you move at the pace you are comfortable with, on financial matters you have educated yourself on. This is designed for empowerment in every respect. Curating the demand as per your understanding making sure it is what you want within what you want to pay for. Less barriers from traditional capital markets. Mapping them to opportunities based on your risk comfortability with your investment.

Individuals : SGMX is built with the companies, services, products, goods, and other features based on what the people vote upon on one side which creates investment opportunities for institutions and companies but also allows the people to partake in them as well. Direct market creation with democratized user demand. The same premise exists here as well, no matter what you do on SGMX we only charge you a flat fee of 0.09% which as we grow will get lower, as **our target goal long term is to drop the OMEC Rate to a range between [0.015-0.025% Cost Per Transaction].**

When we have our mobile and web app released to the public, as long as you have accessible internet access you can connect with the SGMX investment-service marketplace. Providing you public equity investment opportunities that align with your personal investment profile with the causes, companies, and cases that align with your interest. This is one of the core elements that allow SGMX to function as the people's market.

2.2 GDX Index and SGMX Intrinsic Market Valuation Premise

The GDX Index, our Socio-Economic Health Governance Model.

GDX was created to show the overall socio-economic health of opportunities for individuals, companies and institutions within the SGMX marketplace. This allows all three core groups: institutions, companies, and the general public to cohabitate synergistically, regardless of working together or individually to facilitate large scale socio-economic growth, investment, and development. Built with a mixture of private and public equity ownership. Across individuals,

communities, SMEs, enterprise, and states happening all at once as we are digitally available across all five classes with each of the three buyer personas within each core demographic.

Fundamental value of SGMX = Economic equity ownership for users, companies and institutions incentivised via social credit that builds tangible economic development via SGMX. Supporting the value of SGM Coin, as it is backed by this vehicle and its performance.

This in turn brings the individual, companies, and institutions capital gains with three additional vehicles fueling the transactional investment relationship between all three entity classes. Built towards creating, growing, and strengthening Social/Brand/Economic Equity in the people. Promoting economic growth in as many industries via the investments that we have presented to the people.

The first piece of this machine that we are building is SGM's version of the S&P 500. The same application of where the S&P 500 represents the performance of the 500 largest companies within the market. GDX is used to show the Socio-Economic health of the marketplace as a whole. Across all three classes of user that being individuals, companies, or institutions. The index allows us to assess the collective interaction of the market. This is crucial as it allows the people to directly democratize their voices within the capital market. Companies have to interact with one another on the premise of value-given to the individuals, providing significant investment opportunities for institutions to capitalize on.

GDX is the crown jewel of the SGMX Exchange, it is used to show the health of financial interactions while holding them accountable for their impact socially with their weight equivalent economically. This allows a model of educating consumers, companies, and institutions on the opportunities that are present within the marketplace upon their interaction with each other. Driving profits further for companies that are listed on the exchange as they provide value to those who use the exchange. This in turn drives up the valuation of Altcoins listed on SGMX or the increase of services via Alt tokens, backed by fundamental economics with value-given capitalism. Promoting the valuation of the coins and or tokens that are interacted within the marketplace. The public will be educated on which companies are performing with the idea of "value" that is brought to the users of the exchange with transparency of the companies listed. Giving a new perspective towards gauging a company's performance and its impact in its industry, the companies they work with, and the individuals who use that company's products.

Value based capital interactions promoted on a cryptocurrency exchange to create a digital market on blockchain, building an economic blockchain based meritocracy. This amount of free flow in capital for all classes that interact within the marketplace then pushes innovation towards SGMX becoming a digital economy. The other factor of the index is that it is a pledge of

companies working together to garner profits within an ethical manner, where their interactions with other companies and individuals does not come at the detriment of the user regardless of profit gain. Companies who in turn work in some part or as a whole exclusively with their listing via ICOs, IEOs, STOs, ITOs, DRTCOs, DTBOs, DTOs etc on SGMX.

This allows SGMX to become the vehicle that facilitates higher investments from the general public who will be educated on company performances at all times. Companies that they directly will be able to address and call their vote on as well if practices, products, services, goods, or any other byproduct of a corporation does not meet the Socio-Economic Ethical standards of the marketplace. The power of the public forum on SGMX, is to not only allow the people to vote directly on the companies that are in their market, which as an institution we then carry out within certain parameters. It is designed to allow the marketplace to maintain social capital fluidity; this is the core form of liquidity that we intend on building within SGX. Which is then directly backed by collective economic equity of the public that at any point in time a set function of the exchange is built for the people to invest into their areas, communities, and countries because of their usage of the exchange. This also benefits any enterprise solutions providers as the marketplace is able to directly create market demand for public investments. That of which can be supplied by other companies, public figures, brands, family wealth offices, sovereign wealth funds, (other investment vehicles), etc.

2.3 GDX Index Structuring Approach

We took a similar approach to selecting as on the Standard & Poor's. At least in the regards of mapping the structural components supporting the viability of a socio-economic index.

"The components of the S&P 500 are selected by a committee. **(SGM)**...When considering the eligibility of a new addition, the committee assesses the company's merit using eight primary criteria: market capitalization, liquidity, domicile, public float, sector classification, financial viability, and length of time publicly traded and stock exchange.[5] Each of these primary criteria have specific requirements that must be met. For example, in order to be added to the index, a company must satisfy the following liquidity-based size requirements:[5]

1. Market capitalization must be greater than or equal to $6.1 billion USD **(We will begin with a bracket of 5M-250M and then from there break towards 250M-1B, 1B-5B, 5B-10B.) This allows us to build the market within typical market conditions expected within capital markets, or in our case capital marketplaces.**
2. Annual dollar value traded to float-adjusted market capitalization is greater than 1.0
3. Minimum monthly trading volume of 250,000 shares in each of the six months leading up to the evaluation date.

The committee selects the companies in the S&P 500 so they are representative of the industries in the United States economy. ***(In our case we would be selecting regional directors from Samuel Global Management who would oversee distinct areas of operations that surround SGMX.)*** The securities must be publicly listed ***(Not necessarily the case as they could list for the first time by an ICO on our exchange or they can raise capital still staying private via the ITO, either option depending on what the companies want is up to them)*** on either the NYSE (including NYSE Arca or NYSE MKT) or NASDAQ (NASDAQ Global Select Market, NASDAQ Select Market or the NASDAQ Capital Market). ***(For us, it'll be on our exclusive exchange and then building other decentralized exchanges and or financial infrastructures that would allow the marketplace to grow further with hard infrastructure set in place in transitional economies in other countries as we continue to expand, building a total sixteen core hubs that strengthen the marketplace overall in each of the main financial capitals across the world.)***

Securities that are ineligible for inclusion in the index are limited partnerships, master limited partnerships, OTC bulletin board issues, closed-end funds, ETFs, ETNs, royalty trusts, tracking stocks, preferred stocks, unit trusts, equity warrants, convertible bonds, investment trusts, ADRs, ADSs and MLP IT units.[5] [8] The index includes non-U.S. companies, both formerly U.S.-incorporated companies that have re-incorporated outside the United States, as well as firms that have never been incorporated in the United States." - ***This will always be true for SGMX as we want to provide the option for companies to list exclusively on the exchange if they fit within our investment thesis.***

LANGUAGES - One of the features of the exchanges is in the fact that we are going to include is an AI model within the exchange. Where the exchange UX is built towards the individual's native language. Information that can be collected at scale and implemented within the exchange to reach international communities at scale. GDX is meant to set Social and Economic Value standards in the marketplace/digital economy that we are creating. We designed our investment firm SGM with the premise of curating and serving the people at scale, SGMX is the curation of SGM within capital markets.

Crucial components are: First and foremost the **Global Industry Classification Standards**, though for us would be within the cryptosphere. Classifying per industry where each blockchain based or fiat based company operates within. Their market share, actual control within the industry and per dollar value of their services. What is the tangible value attached to the companies in question, not only on the basis of capital but the company's impact at scale in the industries they operate in. Second, the factors to gauge a company's and the general public's performance in SGMX as well as the existing potential and its room to improve further. Now to explain this we broke down the exact structuring of how the index is designed to function.

2.4 GDX Index Structural Composition and Variables

[8] This does not define that the investment vehicles mentioned cannot list on the exchange and or interact with the marketplace.

Socially Economic Developing Investment Index Classification|

Public Market Health	Global Market Health	Corporate Financial Health	Public Financial Health
Human Life Expectancy:	**Environmental Impact:**	**Fair Market Creation:**	**Gross National Income:**
Human Capital:	**Humanitarian Impact:**	**Gainers/Losers:**	**Consumer Purchasing Power:**
Physical Quality of Life:	**Education:**	**Corporate Social Responsibility:**	**Poverty Level per Capita:**
Lives Impacted Per Capita:	**Gender Equality:**	**Secular Stagnation:**	**Entrepreneurship:**
400/400	**400/400**	**400/400**	**400/400**

GDX Index Total Score - 0/1600

This as a whole drives up the overall valuation of companies, the size in volume of daily trades on SGMX, and promotes the creation of a synergistic market. A cryptocurrency market that benefits the people first and equally promotes the capital gain for companies that work with SGM and SGMX. Enterprise doing fair market transactions which in turn drives and capitalizes positively impacting the general public further to work with them. Driving capital returning to the market held by a standard of value. Which allows the people to build with it and have ownership in its infrastructure. This is what we call **Value-Given Capitalism.**

SGM will be doing quarterly audits with each company that lists on SGMX as well as the GDX Index to promote accurate assessments of the market on the index out of 1600 basis points. The goal for any company on the exchange is to rank closer towards maintaining a healthy Corporate Financial Health Score, which comes from our audits of companies and their interactions in our market.

This CFH Score of companies then compiles the standard of value for the companies and their products but also as the investment of the companies open to the users as well. Where we will see the drive in the capital of consumers towards products, goods, services and or investments. It represents the demographic of consumers who are finally getting a form of service towards what was ignored for a period within the market. This is then followed up with the Public Financial Health which informs the users what is their current market capacity, as well as informs the companies the set range of prices of goods they are allowed to market their goods at. Failure of doing so results in corrections to token/coin valuations for predatory pricing on consumers.

2.5 Predatory Pricing or Unethical Price Set Correction Assessment and Penalty

This is appropriated with the current fair market valuation of what goods, services, products, etcs that consumers would purchase from companies taking into consideration of labor, investment, rate of return, etc. We work to make sure that the market provides minimal costs for all of our clientele in turn working with us to build market activity. This will not be at the expense of the users in an unethical or predatory manner. Our market is built for the people. It is our institutional responsibility to represent the interests of our users as we build further. Companies that violate the interests and financial protection of the users will not be tolerated nor permitted to perform within the marketplace.

- **15% overage against fair market affordability** in accordance to what is the Public Financial Health Measure results in a 0.5% value reduction in token/coin market capitalisation.
- **25% overage against fair market affordability** results in a 10% value reduction in the token/coin market capitalization valuation.

- **50% overage against fair market affordability** results in a 75% value reduction in token/coin market capitalization valuation.
- Anything that is beyond this in a companies interaction on the general public will be subjected to being filed as *Persona Non Grata* and furthermore putting into effect the blacklisting process adjudicated by the **OGR.**

2.6 The Ozhuamanil Group Registrar

OGR is a digital registry that is headed by Chief Executive Officer of The Ozhumanil Group as the **Executive Registrar**. Whomsoever is the CEO of the entity with **10 elected Corporate Market Governors (CMGs)**. CMGs are the executives of the top performing entities both within their corporate performance as well as their socio-economic equity.

The performance of entities will determine who will become one of the ten CMGs. **Each CMG will have a tenure of 3 years** upon which new CMGs will be assessed and elected in accordance to their market and social performance. **Existing CMGs can also support for re-election for their existing seats with major contributions to three core elements.**

1. **Investment that provide positive socio-economic impact to the users of the marketplace.**
2. **Growth of the market via infrastructure.**
3. **Financial contribution to The People's Fund**. [9]

This is one part of the voting process during a company's review for Market Deterioration. The second half is that the users of SGMX will be able to vote on the companies in accordance to the review of the companies parallel with the OGR. This creates a user-voting pool that is split 50% to the users and 50% to the corporations. Allowing either side a one-call-all vote as the respective singular vote by that person or entity for the matter under review. All parties in question for the review will be given ample time on all sides to counter and support evidence for the continued permittance within the market. Jurisprudence within the market is subject to the OGR as well as the people, once the review is complete the parties in question will receive their corrections and or are blacklisted from performing within the marketplace.

---

[9] **The People's Fund** is a public pool that is accessible strictly to only the users of the exchange for one-time monthly allocations locked in at $1,000 at the end of each month. **2% of our revenue will be allocated into this fund** *every day.* Allowing time for us to grow it with capital built off the exchange for our users. Our version of approaching **Universal Basic Income (UBI)** via blockchain. This is stored with a mixture of stable-coins **(Tether, Dai, etc)** as well as FIAT converted into appropriate cryptocurrencies. That is then allocated into FIAT of the user's choice. This brings an additional $12,000 in income to our users for being a part of the marketplace. This can also be re-adjusted for distribution depending on the level of capital within the fund. **Only the distribution adjustment tool is within the OGRs purview.** The fund itself is only accessible by users. **The People's Fund IS NOT ACCESSIBLE TO ENTITIES OR INSTITUTIONS UNLESS THEY ARE INFLUXING CAPITAL WITHIN THE FUND AS THEIR INVESTMENT INTO THE PUBLIC WELFARE OF THE USERS.**

The Registry will serve as a public compilation of all companies, entities, individuals, parties, and institutions that are not permitted to operate in any capacity within the market for acting in opposition towards the financial health of all users of the marketplace.

This is the same system that will be in effect to protect companies from other companies that are acting in a predatory manner deteriorating the health of the marketplace.

Which the public will be able to vote on as well, the marketplace is built for their collective interests as well. The governance of the marketplace requires the users and the OGR to work hand in hand, OGR is the regulatory body of the marketplace implemented to protect the users of the market as well as the companies performing within the market. That is only 50% of the governing power within the market, the remaining 50% must be voiced by the users in regards towards corrections in the market to maintain financial stability.

2.7 Market Deterioration Offenses (MDOs)

Companies in the same regard will be held accountable for their actions upon one another as well. **Market Deterioration Offenses (MDOs)**. The same structure for acting against the financial interests of the people applies as well as in the interactions of companies within one another. Treated on a case by case basis for offenses within the marketplace, it is important to include that if the same company is subjected to the same MDOs again in separate occasions they will be held independently of each other but still waited as per the situation of review. **Maximum of Market Deterioration Offenses (MDOs)** allowed per company within the market **is 5 within 2 fiscal years. This can then be appealed to a maximum of three times within 3 months consecutively within each other. You must wait for another 3 months after your first appeal to file the second one as this slot is for any of the other companies involved in the matter to also appeal their ruling to participate in the market.**

- **If a company is reported, reviewed, and ruled against for predatory manner in their financial and or corporate engagement with one another** singularly or collectively on their behalf deteriorating the valuation composition of Corporate Financial Health of the market the company under review will be subjected to a 0.5% value reduction if found complicit within their MDO to their token/coin market capitalisation.
- **If a company is reported, reviewed, and ruled against for predatory manner in their financial and or corporate engagement with engagement with another or 1 more company** singularly or collectively on their behalf deteriorating the Corporate Financial Health will be subjected to a 10% value reduction in the token/coin market capitalization valuation. And the originating company will be subjected to a 25% value reduction in their token/coin market capitalization valuation.

- **If a company is reported, reviewed, and ruled against for predatory manner in their financial and or corporate engagement with engagement with** another or 2 more company deteriorating the value composition of Corporate Financial Health will be in the 75% value reduction in token/coin market capitalization valuation. And the originating company will be subjected to a 95% value reduction in their token/coin market capitalization valuation.

- **Anything that surpasses 5 MDO infraction limit within the 2 fiscal year period** for an entity, company, or institution's interaction on the marketplace with their impact on each other, the users, and or the general public will be subjected to *Persona Non Grata Filing* and blacklisted upon the OGR. This will officially call for Ozhumanil Group to then file a Market Dismissal and Delisting of the company(ies) in question from the marketplace via SGM to SGMX and vice-versa. The entities will be given their fair market worth at the time before the official Market Dismissal to ensure that the companies receive fair market worth after they have received their corrections respectively.

SGMX is designed at its core towards implementing a standard of value of corporations and their interactions amongst each other within the market. This is crucial to build a measure of financial responsibility and social trust. Decorum and structure in a positive manner between the dynamics of corporate enterprises with the users who will be interacting with the aforementioned companies within the marketplace.

This relationship pushes towards a social market equilibrium, which powers both sides with capital and economic equity to grow together with their interactions in the market amongst one another. Positioning both sides the capital power to bring forth growth in their perspective ecosystems independently which then cohesively can tackle stagnation within global markets as generated by an independant marketplace exchange. Creating larger capital sectors and segments to invest build upon existing infrastructure further.

The **SGMX Investment-Service Marketplace Exchange** is our model that we back as the means towards how we can solve the question of what is the proper mixture of public and private equity directly empowering people and bringing high capital gains?

2.8 What The GDX Index Valuation Represents within SGMX

The closer the market is to 1600, the more positive of an impact the marketplace is creating towards socio-economic growth within the market as well as to surrounding infrastructure that allows the market to grow further. Which allows further replication and scale of financial transaction within the marketplace. Based out of directly impacting the lives of the general

public. And vice-versa with the general public informed about the nature of companies with transparency, positively motivating them with that standard of trust to invest, purchase, or trade their goods, products, and or services within the marketplace.

The creation of raw market data built on tangible economic growth facilitated by blockchain. Driving the creation of optimized human capital, a true distribution of wealth on the basis of opportunity. To drive the creation of personal wealth by giving the people a platform that drives value to them while also capitalistically focused as a whole strengthening economic/social/brand equity of companies. Which, in turn, allows their market share to grow. Redefining competition within the market towards value-driven companies, and who is the most value driven for the end-consumer.

A strong value capitalist approach built with the support and economic power distributed by SGMX to the people. The viability of SGMX lands not only on the support of the general public but also the growth at scale of companies that work with us on SGMX. We capitalize on the opportunities we create via SGMX. The yield is not only in trading fees as existing marketplayers. The yield is shown through the growth within markets, cities, companies, and economies of scale. What ties this together and allows this to become a blend of public and private equity is the SGMX Common Fund Initiative, also known as the CFI Projects which are facilitated with the backing of the public to create.

Orion in Greek Mythology was a giant huntsman, the specifics for the sake of this narrative are not of great importance. More so the fact that he was a giant and a huntsman.

As is the directive of The Ozhumanil Decentralized Capital Market (ODCM), to build it into the largest independent digital capital market economy, a giant. And to go forth like the huntsman and conquer via our reach of the ODCM and the structure that we have designed for SGMX. We will create customized exchanges based on the natural resources, commodities, sectors of focus, of each respective nation that we approach with decentralization in mind, creating and transitioning capital markets that are decentralized. Each sister exchange of SGMX will be built on top of the Ozhumanil Financial Network (OFN) and their respective currencies will be paired with SGX Token creating liquidity pairings between FIAT markets and our independent market, ODCM. And all assets of each exchange will be liquidating out of SGM Coin, driving up further the valuation of SGM Coin but also the volume in relation towards SGM Coin and SGX Token. The structure is as follows when approaching the governments in question with the objective of building their National-ISM, Investment-Service Marketplace on a Decentralized Socio-Economic Digital Asset Exchange facilitating the transition of capital markets towards decentralized capital markets.

The listings of SGX in relation to the financial centers we want to have direct access and influence within.

1. 2B SGX - Liquid Exchange - @ $4.00USD - $8B MARCAP - JAPAN
2. 2B SGX - Egyptian Stock Exchange (EGX) - @ $4.50USD - $9B MARCAP - EGYPT
3. 2B SGX - Bombay Stock Exchange (BSE) - @ $5.00USD - $10B MARCAP - INDIA
4. 2B SGX - Hong Kong Stock Exchange (HKEX) - @ $5.50USD - $11B MARCAP - HK
5. 2B SGX - Korean Stock Exchange (KRX) - @ $6.00USD - $12B MARCAP - S.Korea
6. 2B SGX - Brasil Bolsa Balcao (B3) - @ $6.50USD - $13B MARCAP - BRASIL
7. 2B SGX - New York Stock Exchange - @ $7.50 - $15B MARCAP - NYC

Total Currency Listed MarCap = $78B USD
Total Currency Circulation = 14B SGX
Price Range on Currency Listed = [$4.00USD - $7.50USD]
Average Price on Currency Listed = $5.5714USD = $77.9996B USD
Average Market Cap Range = [$56B - $105B USD]

All listings will stabilize based on the Ozhumanil Decentralized Capital Market, the ODCM naturally sets the price of the currency as per the GAINS Algorithm which promotes the valuation overtime, and corrects as per standard market correction on the price of SGX. This ensures the value of SGX is truly independent and can function as an independent digital asset that is listed on the following aforementioned exchanges. Each will push our markets growth and its ability to invest directly in infrastructure/asset classes of each country.

Ozhumanil Economic Exchanges (O2E LLC): The Orion Initiative

All National-ISMs will be built exclusively on the Ozhumanil Financial Network (OFN)
All National-ISM exchange currencies will be paired with the native currency of the OFN.
All National-ISM exchanges assets will be liquid out of the network via only SGM Coin.
All-National-ISM exchange's currencies will also have the same specs of SGX Token, this is meant to facilitate cohabitant economic growth and financial stability via the ODCM.
All-National-ISM exchanges will be curated towards what will promote the highest economic growth and financial stability of

Equity Structure:

23% of the revenue is proportional to the equity of each ISM to facilitate funding for all public infrastructure.
26% will be held exclusively by O2E in turn The Ozhumanil Group, with direct operational control **(52% control, 48% control to each nation state**).
As part of our direct-foreign investment into each sovereign nation as well as management, listing services, and additional financial services to facilitate the needs of each nation-state. This is also to leverage the creation of direct independent ports that we hold within the ODCM for all of our shipping needs and actions for each country but to also facilitate trade exclusively on the network between each nation state via the ODCM.
With the 52% operational control we also hold the power to put the exchange towards direct-democratic vote to suspend the exchange if found in violation within our market charter.

As well as first-rights towards international investments between each state.
51% will be held by the State in question and or by the State Investment arm of each respective nation that is approached in regards towards the revenue that each exchange creates, as well as 48% of its operational control.

Token Allocation Per ISM

45% of total supply will be held by the OBC, the Ozhumanil Banking Corporation as per our foreign-direct investment into each Nation-State.
50% of total supply will be held by each State.
5% will be in circulation on its respective ISM with its native currency.
The responsibility to release further into their respective Decentralized Capital Market will be upon each state with the allocation they have as will our decision to do so as well. This will introduce a fixed amount of circulation of digital currency that counters the implementation of quantitative easing as a means of promoting macroeconomic growth. Focusing towards infrastructure, job security, quality of markets, and the mobility of public wealth to stabilize the growth trend of Decentralized Capital Markets (DCMs)

Ex: Tunisian Global Market Exchange (TGMX) - Native Currency - TGX/SGX
Total Hypothetical Supply - 10T TGX
OBC: 4.5T TGX
Tunisian Government: 5.0T TGX
TGMX: 500M TGX

Nation States to build National-ISMs[10]

Argentina - Argentine Global Market Exchange (AGMX) - Native Currency - (AGX)
Natural Resources of Argentina: 80% of the World's Reserves of Lithium Brine, copper, gold, zinc, silver, natural gas, petroleum. One of the world's largest crop producers for wheat, soybeans, maize, barley, rice, flax seed, sugarcane, cotton, citrus fruit, and grapes. This gives us influence throughout the south of South America via the South Atlantic and Pacific Ocean.
Brazil - Brazilian Global Market Exchange (BGMX) - Native Currency - (BGX)
Natural Resources of Brazil: bauxite, gold, iron ore, manganese, nickel, phosphates, platinum, tin, clay, rare earth elements, uranium, petroleum, hydropower and timber. With influence in Argentina and Brazil it gives us mobility throughout the South Atlantic Ocean.
Colombia - Colombian Global Market Exchange (CGMX) - Native Currency - (CGX)
Natural Resources of Colombia: The main producer of emeralds, and major participants in gold, silver, iron, salt, platinum, petroleum, nickel, copper, hydropower, and uranium. Colombia gives us reach into Central America via Panama and direct access towards the Caribbean Sea and in turn the Gulf of Mexico. Another major advantage for us for the ODCM as we grow.
Dominican Republic - Dominican Global Market Exchange (DGMX) - Native Currency - (DGX)
Natural Resources of Dominican Republic: Nickel, bauxite, gold, silver, and arable land.
Egypt - Egyptian Global Market Exchange (EGMX) - Native Currency - (EGX)

[10] Total supply of national ISM currencies dependent on negotiations of each nation. Same structure still follows.

Largest proponent to why we want to build the EGMX is because of its proximity within the Middle East, gives us the cusp of influence for all imports/exports of the ODCM within the Mediterranean via Port Said into the Suez Canal to directly reach the Red Sea directly into the Gulf of Aden (Into the Arabian Sea) and in turn towards Sudan, Ethiopia, Eritrea, Djibouti, Somalia, Oman, Saudi Arabia, and Iran.

France - French Global Market Exchange (FGMX) - Native Currency - (FGX)

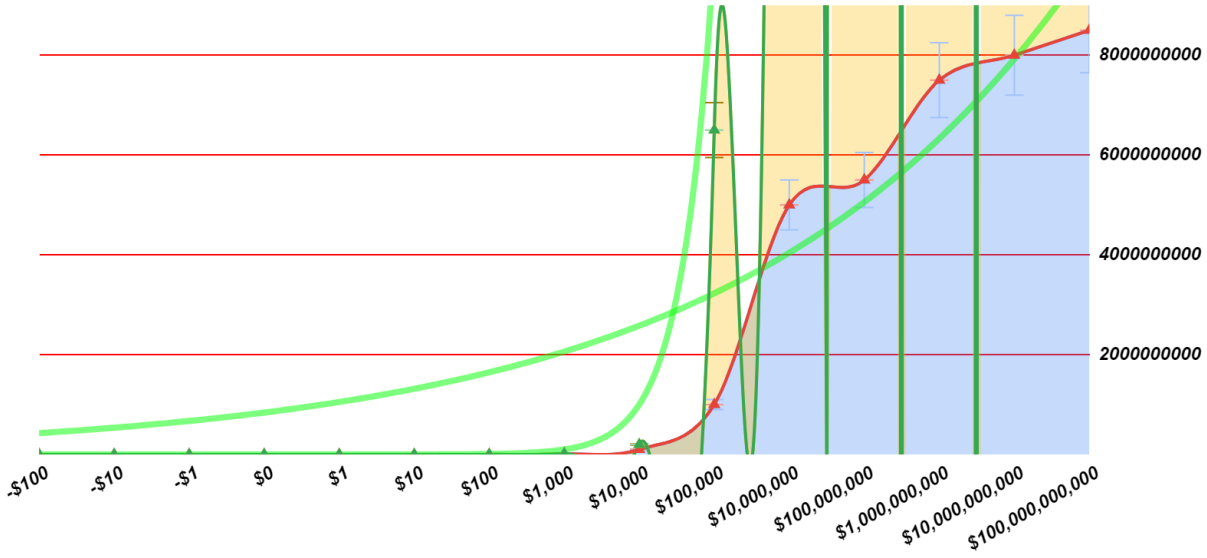
Natural Resources of France: *Coal, Natural Gas, Uranium, Hydroelectric Energy, Wind Energy, Bauxite, Iron Ore, Thermal/Solar Energy.* We will invest with France and show the prospects of their return if they are willing to remove the Colonial Tax that is in place on the 14 countries in Africa for existing as independent nations, where we show how FGMX will yield more than the tax that they collect on the fourteen countries. Providing true independence for the nations but also financial sovereignty within Europe for France. France is crucial as it allows us to create a sphere of influence

Collective Market Capitalization of Ozhumanill Decentralized Capital Market	(I, in $)	(x, in people)	(r, average interactions, per person in ODCM)	(n, transactions, $)	(t, hours)
	-$100	1	0.00	$0.00	1.5
// Compound Investment of Individuals and or Communities (CIIC)	-$10	10	1.00	$1.00	3
// (y) = i (1+x)+(r/n))^(n/t)	-$1	100	5.00	$10.00	4.5
y = Compound Investment of Individuals and or Communities (CIIC)	$0	1,000	10.00	$100.00	6
i = the collective independent investment capital	$1	10,000	20.00	$10,000.00	7.5
x = the volume of individuals and or communities by increment of 1	$10	100,000	40.00	$100,000.00	9
r = average interactions, per person in the ODCM	$100	1,000,000	80.00	$1,000,000.00	10.5
n = the number of transactional interactions maintained within market at $ value	$1,000	10,000,000	160.00	$10,000,000.00	12
t = physical time used in market network, 24/7 hour open market.	$10,000	100,000,000	320.00	$100,000,000.00	13.5
	$100,000	500,000,000	640.00	$1,000,000,000.00	15
	$100,000	1,500,000,000	1,280.00	$10,000,000,000.00	16.5

	$10,000,000	5,000,000,000	2,560.00	$100,000,000,000.00	18
	$100,000,000	5,500,000,000	5,120.00	$1,000,000,000,000.00	19.5
	$1,000,000,000	7,500,000,000	10,240.00	$10,000,000,000,000.00	21
	$10,000,000,000	8,000,000,000	20,480.00	$100,000,000,000,000.00	22.5
	$100,000,000,000	8,500,000,000	40,960.00	$1,000,000,000,000,000.00	24

Collective Market Capitalization Volume Growth of Independent Digital Share-Holder Capital Market Economy

Market Capitalization of Ozhumanil Decentralized Capital Market (ODCM) - 1st of DCM Asset-Class - (Decentralized Capital Market)

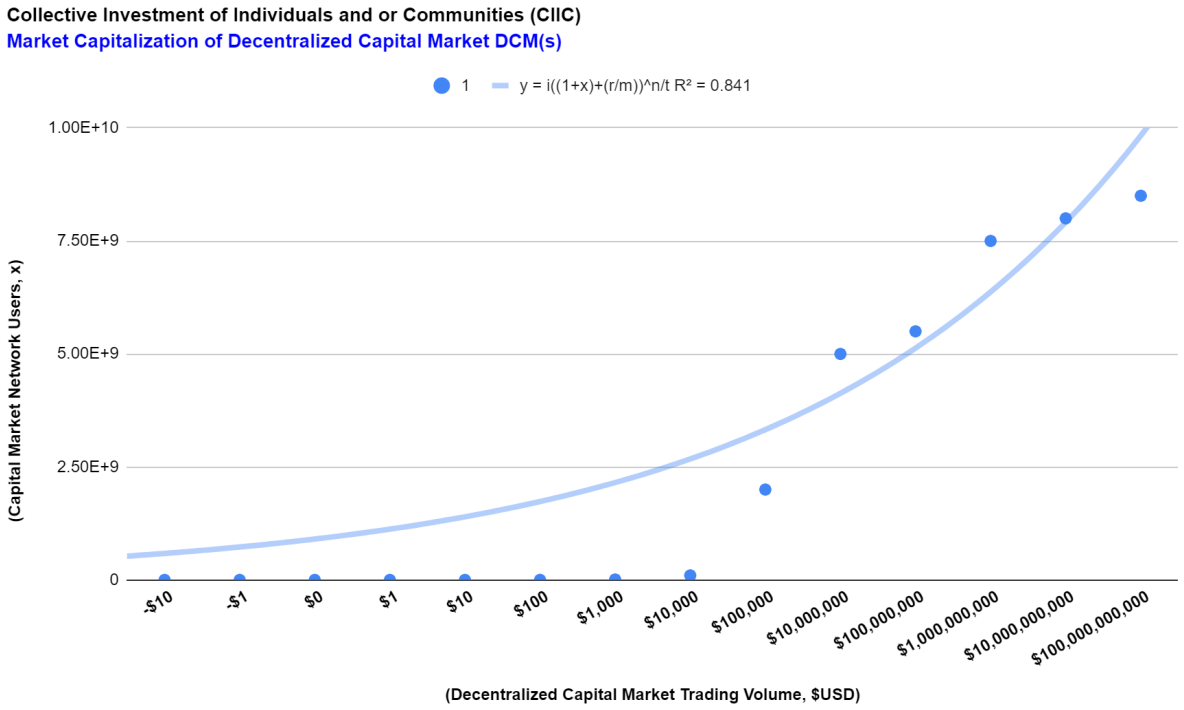


(Ozhumanil Decentralized Capital Market Trading Volume, $USD)

(t, hours) 10e^2.3x R² = 1 (n, transactions, $) 10e^2.3x R² = 1

(r, average interactions, per person in ODCM) 4.28E+08e^0.224x R² = 0.836 (x, in people)

4.28E+08e^0.224x R² = 0.836



Turtle:

```
import numpy as np
import math
import matplotlib.pyplot as plt
from scipy import integrate

def calculate_average(numbers):
    return sum(numbers) / len(numbers)

def create_null_matrix(shape):
    return np.zeros(shape)

def calculate_sum_null_matrix(matrix):
    return np.sum(matrix)

def calculate_factorial_percentile(x):
    return math.factorial(x)
```

```
def logistic_map_with_shipping(r, x0, shipping_cost, num_steps):
    x_values = [x0]
    for _ in range(num_steps - 1):
        x_n = x_values[-1]
        x_next = r * x_n * (1 - x_n) + shipping_cost
        x_values.append(x_next)
    return x_values

def generate_financial_data_with_shipping(r_values, x0_sequence, shipping_costs, num_steps):
    financial_data = []
    for x0 in x0_sequence:
        for shipping_cost in shipping_costs:
            x_values = logistic_map_with_shipping(r_values, x0, shipping_cost, num_steps)
            financial_data.append(x_values)
    return np.array(financial_data)

def numerical_derivative(y_values, dt):
    return np.gradient(y_values, dt)

def main():
    # Set the parameters for the logistic map, shipping costs, and time steps
    r_values = [2.5, 3.2, 3.5, 3.8]
    x0_sequence = [0.2, 0.5, 0.8]
    shipping_costs = [0, 0.1, 0.2]  # Add different shipping costs here
    num_steps = 100
    dt = 1  # Time step

    # Generate financial data for perpetual, swing, commodities, stock, bonds, and debt
    financial_data_perpetual = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_swing = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_commodities = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_stock = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_bonds = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_debt = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
```

```
    # Calculate average and standard deviation for each financial sequence
    average_values_perpetual = np.mean(financial_data_perpetual, axis=1)
    std_deviation_values_perpetual = np.std(financial_data_perpetual, axis=1)

    average_values_swing = np.mean(financial_data_swing, axis=1)
    std_deviation_values_swing = np.std(financial_data_swing, axis=1)

    average_values_commodities = np.mean(financial_data_commodities, axis=1)
    std_deviation_values_commodities = np.std(financial_data_commodities, axis=1)

    average_values_stock = np.mean(financial_data_stock, axis=1)
    std_deviation_values_stock = np.std(financial_data_stock, axis=1)

    average_values_bonds = np.mean(financial_data_bonds, axis=1)
    std_deviation_values_bonds = np.std(financial_data_bonds, axis=1)

    average_values_debt = np.mean(financial_data_debt, axis=1)
    std_deviation_values_debt = np.std(financial_data_debt, axis=1)

    # Calculate the integral with the upper limit of x_prime and lower limit of infinity
    integral_results_perpetual = []
    integral_results_swing = []
    integral_results_commodities = []
    integral_results_stock = []
    integral_results_bonds = []
    integral_results_debt = []

    for i in range(len(r_values) * len(x0_sequence)):
        x0_idx = i % len(x0_sequence)
        result_perpetual, _ = integrate.quad(lambda x: (x * financial_data_perpetual[i][0]) ** (-1),
financial_data_perpetual[i][0], np.inf)
        integral_results_perpetual.append(result_perpetual)

        result_swing, _ = integrate.quad(lambda x: (x * financial_data_swing[i][0]) ** (-1),
financial_data_swing[i][0], np.inf)
        integral_results_swing.append(result_swing)

        result_commodities, _ = integrate.quad(lambda x: (x * financial_data_commodities[i][0])
** (-1), financial_data_commodities[i][0], np.inf)
```

```
        integral_results_commodities.append(result_commodities)

        result_stock, _ = integrate.quad(lambda x: (x * financial_data_stock[i][0]) ** (-1),
financial_data_stock[i][0], np.inf)
        integral_results_stock.append(result_stock)

        result_bonds, _ = integrate.quad(lambda x: (x * financial_data_bonds[i][0]) ** (-1),
financial_data_bonds[i][0], np.inf)
        integral_results_bonds.append(result_bonds)

        result_debt, _ = integrate.quad(lambda x: (x * financial_data_debt[i][0]) ** (-1),
financial_data_debt[i][0], np.inf)
        integral_results_debt.append(result_debt)

    # Calculate the numerical derivatives of the financial sequences
    derivatives_perpetual = []
    derivatives_swing = []
    derivatives_commodities = []
    derivatives_stock = []
    derivatives_bonds = []
    derivatives_debt = []

    for i in range(len(r_values) * len(x0_sequence)):
        derivative_perpetual = numerical_derivative(financial_data_perpetual[i], dt)
        derivatives_perpetual.append(derivative_perpetual)

        derivative_swing = numerical_derivative(financial_data_swing[i], dt)
        derivatives_swing.append(derivative_swing)

        derivative_commodities = numerical_derivative(financial_data_commodities[i], dt)
        derivatives_commodities.append(derivative_commodities)

        derivative_stock = numerical_derivative(financial_data_stock[i], dt)
        derivatives_stock.append(derivative_stock)

        derivative_bonds = numerical_derivative(financial_data_bonds[i], dt)
        derivatives_bonds.append(derivative_bonds)

        derivative_debt = numerical_derivative(financial_data_debt[i], dt)
        derivatives_debt.append(derivative_debt)
```

```
    # Calculate the final expressions
    expressions_perpetual = (average_values_perpetual - std_deviation_values_perpetual) ** 2 +
integral_results_perpetual
    expressions_swing = (average_values_swing - std_deviation_values_swing) ** 2 +
integral_results_swing
    expressions_commodities = (average_values_commodities -
std_deviation_values_commodities) ** 2 + integral_results_commodities
    expressions_stock = (average_values_stock - std_deviation_values_stock) ** 2 +
integral_results_stock
    expressions_bonds = (average_values_bonds - std_deviation_values_bonds) ** 2 +
integral_results_bonds
    expressions_debt = (average_values_debt - std_deviation_values_debt) ** 2 +
integral_results_debt

    # Display the results and plots
    # ... (same as before, omitted for brevity)

if __name__ == "__main__":
    main()

Trade_Registry:

class Investment:
    def __init__(self, investment_id, investor, instrument, quantity, price, investment_date):
        self.investment_id = investment_id
        self.investor = investor
        self.instrument = instrument
        self.quantity = quantity
        self.price = price
        self.investment_date = investment_date

    def __str__(self):
        return f"Investment ID: {self.investment_id}, Investor: {self.investor}, Instrument:
{self.instrument}, Quantity: {self.quantity}, Price: {self.price}, Investment Date:
{self.investment_date}"

class InvestmentRegistry:
    def __init__(self):
        self.investments = []
```

```
    def add_investment(self, investment):
        self.investments.append(investment)

    def remove_investment_by_id(self, investment_id):
        self.investments = [investment for investment in self.investments if
investment.investment_id != investment_id]

    def get_investment_by_id(self, investment_id):
        for investment in self.investments:
            if investment.investment_id == investment_id:
                return investment
        return None

    def get_investments_by_investor(self, investor):
        return [investment for investment in self.investments if investment.investor == investor]

    def get_all_investments(self):
        return self.investments

class GovernmentInvestment(Investment):
    def __init__(self, investment_id, investor, instrument, quantity, price, investment_date,
government_name):
        super().__init__(investment_id, investor, instrument, quantity, price, investment_date)
        self.government_name = government_name

    def __str__(self):
        return super().__str__() + f", Government: {self.government_name}"

class TradeAndInvestmentRegistry:
    def __init__(self):
        self.trades = []
        self.investments = []

    # ... Add methods to handle trades and investments ...

    def add_investment(self, investment):
        self.investments.append(investment)

    def get_investment_by_id(self, investment_id):
```

```
        for investment in self.investments:
            if investment.investment_id == investment_id:
                return investment
        return None

    # ... Other methods for investment registry ...

    def get_all_investments(self):
        return self.investments

if __name__ == "__main__":
    # Create a trade and investment registry
    trade_and_investment_registry = TradeAndInvestmentRegistry()

    # Add trades to the registry
    trade1 = Trade(1, "John Doe", "AAPL", 100, 150.0, "2023-06-15", corporation="ABC Corp")
    trade2 = Trade(2, "Jane Smith", "GOOG", 50, 2500.0, "2023-06-16",
sovereign_wealth_fund="SWF Holdings")

    trade_and_investment_registry.add_trade(trade1)
    trade_and_investment_registry.add_trade(trade2)

    # Add investments to the registry (including government investments)
    investment1 = Investment(1, "Investor1", "AAPL", 200, 160.0, "2023-06-17")
    investment2 = GovernmentInvestment(2, "Investor2", "GOOG", 100, 2550.0, "2023-06-18",
government_name="Government1")

    trade_and_investment_registry.add_investment(investment1)
    trade_and_investment_registry.add_investment(investment2)

    # Get all trades and investments in the registry
    all_trades = trade_and_investment_registry.get_all_trades()
    all_investments = trade_and_investment_registry.get_all_investments()

    for trade in all_trades:
        print(trade)

    for investment in all_investments:
        print(investment)
```

```
    # ... Continue testing and using the registry for your specific use cases ...
```

Trade:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import integrate

def logistic_map_with_shipping(r, x0, shipping_cost, num_steps):
    x_values = [x0]
    for _ in range(num_steps - 1):
        x_n = x_values[-1]
        x_next = r * x_n * (1 - x_n) + shipping_cost
        x_values.append(x_next)
    return x_values

def generate_financial_data_with_shipping(r_values, x0_sequence, shipping_costs, num_steps):
    financial_data = []
    for x0 in x0_sequence:
        for shipping_cost in shipping_costs:
            x_values = logistic_map_with_shipping(r_values, x0, shipping_cost, num_steps)
            financial_data.append(x_values)
    return np.array(financial_data)

def numerical_derivative(y_values, dt):
    return np.gradient(y_values, dt)

def main():
    # Set the parameters for the logistic map, shipping costs, and time steps
    r_values = [2.5, 3.2, 3.5, 3.8]
    x0_sequence = [0.2, 0.5, 0.8]
    shipping_costs = [0, 0.1, 0.2]  # Add different shipping costs here
    num_steps = 100
    dt = 1  # Time step

    # Generate financial data for perpetual, swing, commodities, stock, bonds, and debt
    financial_data_perpetual = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_swing = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
```

```
    financial_data_commodities = generate_financial_data_with_shipping(r_values, x0_sequence, shipping_costs, num_steps)
    financial_data_stock = generate_financial_data_with_shipping(r_values, x0_sequence, shipping_costs, num_steps)
    financial_data_bonds = generate_financial_data_with_shipping(r_values, x0_sequence, shipping_costs, num_steps)
    financial_data_debt = generate_financial_data_with_shipping(r_values, x0_sequence, shipping_costs, num_steps)

    # Calculate average and standard deviation for each financial sequence
    average_values_perpetual = np.mean(financial_data_perpetual, axis=1)
    std_deviation_values_perpetual = np.std(financial_data_perpetual, axis=1)

    average_values_swing = np.mean(financial_data_swing, axis=1)
    std_deviation_values_swing = np.std(financial_data_swing, axis=1)

    average_values_commodities = np.mean(financial_data_commodities, axis=1)
    std_deviation_values_commodities = np.std(financial_data_commodities, axis=1)

    average_values_stock = np.mean(financial_data_stock, axis=1)
    std_deviation_values_stock = np.std(financial_data_stock, axis=1)

    average_values_bonds = np.mean(financial_data_bonds, axis=1)
    std_deviation_values_bonds = np.std(financial_data_bonds, axis=1)

    average_values_debt = np.mean(financial_data_debt, axis=1)
    std_deviation_values_debt = np.std(financial_data_debt, axis=1)

    # Calculate the integral with the upper limit of x_prime and lower limit of infinity
    integral_results_perpetual = []
    integral_results_swing = []
    integral_results_commodities = []
    integral_results_stock = []
    integral_results_bonds = []
    integral_results_debt = []

    for i in range(len(r_values) * len(x0_sequence)):
        x0_idx = i % len(x0_sequence)
        result_perpetual, _ = integrate.quad(lambda x: (x * financial_data_perpetual[i][0]) ** (-1), financial_data_perpetual[i][0], np.inf)
```

```
        integral_results_perpetual.append(result_perpetual)

        result_swing, _ = integrate.quad(lambda x: (x * financial_data_swing[i][0]) ** (-1),
financial_data_swing[i][0], np.inf)
        integral_results_swing.append(result_swing)

        result_commodities, _ = integrate.quad(lambda x: (x * financial_data_commodities[i][0])
** (-1), financial_data_commodities[i][0], np.inf)
        integral_results_commodities.append(result_commodities)

        result_stock, _ = integrate.quad(lambda x: (x * financial_data_stock[i][0]) ** (-1),
financial_data_stock[i][0], np.inf)
        integral_results_stock.append(result_stock)

        result_bonds, _ = integrate.quad(lambda x: (x * financial_data_bonds[i][0]) ** (-1),
financial_data_bonds[i][0], np.inf)
        integral_results_bonds.append(result_bonds)

        result_debt, _ = integrate.quad(lambda x: (x * financial_data_debt[i][0]) ** (-1),
financial_data_debt[i][0], np.inf)
        integral_results_debt.append(result_debt)

    # Calculate the numerical derivatives of the financial sequences
    derivatives_perpetual = []
    derivatives_swing = []
    derivatives_commodities = []
    derivatives_stock = []
    derivatives_bonds = []
    derivatives_debt = []

    for i in range(len(r_values) * len(x0_sequence)):
        derivative_perpetual = numerical_derivative(financial_data_perpetual[i], dt)
        derivatives_perpetual.append(derivative_perpetual)

        derivative_swing = numerical_derivative(financial_data_swing[i], dt)
        derivatives_swing.append(derivative_swing)

        derivative_commodities = numerical_derivative(financial_data_commodities[i], dt)
        derivatives_commodities.append(derivative_commodities)
```

```
        derivative_stock = numerical_derivative(financial_data_stock[i], dt)
        derivatives_stock.append(derivative_stock)

        derivative_bonds = numerical_derivative(financial_data_bonds[i], dt)
        derivatives_bonds.append(derivative_bonds)

        derivative_debt = numerical_derivative(financial_data_debt[i], dt)
        derivatives_debt.append(derivative_debt)

    # Calculate the final expressions
    expressions_perpetual = (average_values_perpetual - std_deviation_values_perpetual) ** 2 +
integral_results_perpetual
    expressions_swing = (average_values_swing - std_deviation_values_swing) ** 2 +
integral_results_swing
    expressions_commodities = (average_values_commodities -
std_deviation_values_commodities) ** 2 + integral_results_commodities
    expressions_stock = (average_values_stock - std_deviation_values_stock) ** 2 +
integral_results_stock
    expressions_bonds = (average_values_bonds - std_deviation_values_bonds) ** 2 +
integral_results_bonds
    expressions_debt = (average_values_debt - std_deviation_values_debt) ** 2 +
integral_results_debt

    # Display the results and plots
    # ... (same as before, omitted for brevity)

if __name__ == "__main__":
    main()
```

Timing:

```
import matplotlib.pyplot as plt
import numpy as np

def logistic_map(r, x0, num_steps):
    x_values = [x0]
    for _ in range(num_steps - 1):
        x_n = x_values[-1]
        x_next = r * x_n * (1 - x_n)
        x_values.append(x_next)
```

```
    return x_values

def financial_sequences(r_values, x0_sequence, num_steps):
    # Create an empty list to store financial sequences
    financial_data = []

    for x0 in x0_sequence:
        x_values = logistic_map(r_values, x0, num_steps)
        financial_data.append(x_values)

    return np.array(financial_data)

def plot_financial_sequences(r_values, x0_sequence, num_steps):
    financial_data = financial_sequences(r_values, x0_sequence, num_steps)
    for r_idx, r in enumerate(r_values):
        plt.plot(range(num_steps), financial_data[r_idx], label=f"r={r}")

    plt.xlabel('Time Step')
    plt.ylabel('Financial Sequence')
    plt.title('Financial Sequences from the Logistic Map')
    plt.legend()
    plt.show()

if __name__ == "__main__":
    # Set the parameters for the logistic map
    r_values = [2.5, 3.2, 3.5, 3.8]
    x0_sequence = [0.2, 0.5, 0.8]
    num_steps = 100

    # Plot financial sequences for the specified values of r and x0
    plot_financial_sequences(r_values, x0_sequence, num_steps)
```

Stock_Markets:

```
import pandas as pd
import numpy as np
import yfinance as yf

class BankAccount:
    def __init__(self, account_number, account_holder, balance, currency):
```

```
        self.account_number = account_number
        self.account_holder = account_holder
        self.balance = balance
        self.currency = currency

    def deposit(self, amount):
        self.balance += amount

    def withdraw(self, amount):
        if self.balance >= amount:
            self.balance -= amount
        else:
            print("Insufficient funds")

    def clear_check(self, check_amount):
        self.withdraw(check_amount)
        print(f"Check cleared for {check_amount} {self.currency}")

    def transfer_funds(self, target_account, amount):
        if self.balance >= amount:
            self.withdraw(amount)
            target_account.deposit(amount)
            print(f"Transferred {amount} {self.currency} to Account
{target_account.account_number}")
        else:
            print("Insufficient funds")

    def invest_funds(self, investment_amount):
        print(f"Investing {investment_amount} {self.currency} in the stock market")
        # Add the logic to invest the funds in the stock market

    def display_balance(self):
        print(f"Account Holder: {self.account_holder}")
        print(f"Account Number: {self.account_number}")
        print(f"Balance: {self.balance} {self.currency}")

    def deduct_shipping_costs(self, shipping_cost):
        if self.balance >= shipping_cost:
            self.withdraw(shipping_cost)
            print(f"Deducted shipping cost: {shipping_cost} {self.currency}")
```

```
        else:
            print("Insufficient funds for shipping costs")

    def handle_transitional_volume(self, volume, price):
        transaction_cost = volume * price
        if self.balance >= transaction_cost:
            self.withdraw(transaction_cost)
            print(f"Transitional volume handled. Transaction cost: {transaction_cost}
{self.currency}")
        else:
            print("Insufficient funds for transitional volume")

class RiskTool:
    @staticmethod
    def calculate_covariance(asset1_returns, asset2_returns):
        covariance = np.cov(asset1_returns, asset2_returns)[0, 1]
        return covariance

    @staticmethod
    def calculate_variance(returns):
        variance = np.var(returns)
        return variance


def get_fiscal_year_data(ticker, year):
    # Fetch data from Yahoo Finance for the given ticker and year
    start_date = f"{year}-01-01"
    end_date = f"{year}-12-31"
    data = yf.download(ticker, start=start_date, end=end_date)
    return data


def find_lowest_and_highest_prices(data):
    # Find the lowest and highest prices for the ticker
    lowest_price = data['Low'].min()
    highest_price = data['High'].max()
    return lowest_price, highest_price


def find_largest_change_one_day(data):
```

```
    # Calculate the largest change in the market on one day
    data['Daily_Change'] = data['Close'].diff()
    largest_change_one_day = data['Daily_Change'].abs().max()
    return largest_change_one_day


def find_largest_change_two_days(data):
    # Calculate the largest change in the market over two consecutive days
    data['Two_Day_Change'] = data['Close'].diff(periods=2)
    largest_change_two_days = data['Two_Day_Change'].abs().max()
    return largest_change_two_days


def calculate_average_moving_trade_volume(data, window=5):
    # Calculate the average moving trade volume
    data['Avg_Moving_Trade_Volume'] = data['Volume'].rolling(window=window).mean()
    avg_moving_trade_volume = data['Avg_Moving_Trade_Volume'].mean()
    return avg_moving_trade_volume


if __name__ == "__main__":
    # Test the functionalities
    # Create bank accounts and perform transactions
    account1 = BankAccount("123456789", "John Doe", 1000.0, "USD")
    account2 = BankAccount("987654321", "Jane Smith", 500.0, "USD")

    account1.display_balance()
    account2.display_balance()

    account1.deposit(500.0)
    account1.display_balance()

    account1.withdraw(200.0)
    account1.display_balance()

    check_amount = 300.0
    account1.clear_check(check_amount)
    account1.display_balance()

    transfer_amount = 400.0
```

```
account1.transfer_funds(account2, transfer_amount)
account1.display_balance()
account2.display_balance()

# Invest funds in the stock market
investment_amount = 1000.0
account1.invest_funds(investment_amount)

# Financial data analysis for AFX_X for the fiscal year of 2017
ticker = "AFX_X"
fiscal_year = 2017

data = get_fiscal_year_data(ticker, fiscal_year)

lowest_price, highest_price = find_lowest_and_highest_prices(data)
print("Lowest Price:", lowest_price)
print("Highest Price:", highest_price)

largest_change_one_day = find_largest_change_one_day(data)
print("Largest Change in One Day:", largest_change_one_day)

largest_change_two_days = find_largest_change_two_days(data)
print("Largest Change in Two Days:", largest_change_two_days)

avg_moving_trade_volume = calculate_average_moving_trade_volume(data)
print("Average Moving Trade Volume:", avg_moving_trade_volume)

# Deduct shipping costs from account1
shipping_cost = 50.0
account1.deduct_shipping_costs(shipping_cost)
account1.display_balance()

# Handle transitional volume in the stock market
volume = 1000  # Example transitional volume
price = 50.0   # Example price per share
account1.handle_transitional_volume(volume, price)
account1.display_balance()
```

Sales:

```
import numpy as np
from flask import Flask, request, jsonify
import pandas as pd
from scipy import stats

# Assuming you have a CSV file or database with app sales/rankings data
# For simplicity, let's assume the data contains the app names and their sales/rankings.

# Load the data (replace 'your_data.csv' with the actual file path or database query)
data = pd.read_csv('your_data.csv')

# Calculate the probability of an app being a best-seller
def calculate_best_seller_probability(app_name):
    # Assuming you have a method or model to calculate the probability
    # Replace the logic below with your own calculation method or model.
    probability = 0.7  # Example probability (70%)
    return probability

# Get the top best-selling apps
def get_best_sellers(num_apps=5):
    # Assuming you have a method to get the top best-selling apps from your data
    # For example, you can sort the data by sales/rankings and select the top 'num_apps' apps.
    best_sellers = data.nlargest(num_apps, 'sales')  # Replace 'sales' with the column representing
sales/rankings.
    return best_sellers

# Get statistics for the market
def get_market_statistics():
    # Calculate the statistics based on the sales/rankings data
    market_mean = data['sales'].mean()
    market_median = data['sales'].median()
    market_mode = data['sales'].mode().iloc[0]
    market_std_dev = data['sales'].std()

    return {
        'mean': market_mean,
        'median': market_median,
        'mode': market_mode,
```

```
        'standard_deviation': market_std_dev
    }

# Initialize Flask application
app = Flask(__name__)

# Define API endpoint to get the probability of an app being a best-seller
@app.route('/api/best_seller_probability', methods=['GET'])
def get_best_seller_probability():
    app_name = request.args.get('app_name')
    if not app_name:
        return jsonify({'error': 'App name not provided'}), 400

    # Lookup the app_name in the data and get the relevant information
    app_info = data[data['app_name'] == app_name]

    if app_info.empty:
        return jsonify({'error': 'App not found'}), 404

    # Calculate the probability of being a best-seller
    probability = calculate_best_seller_probability(app_name)

    return jsonify({'app_name': app_name, 'probability': probability})

# Define API endpoint to get the top best-selling apps
@app.route('/api/top_best_sellers', methods=['GET'])
def get_top_best_sellers():
    num_apps = int(request.args.get('num_apps', 5))  # Default to 5 apps if 'num_apps' parameter
not provided

    # Get the top best-selling apps
    best_sellers = get_best_sellers(num_apps)

    # Convert the DataFrame to a list of dictionaries for JSON response
    best_sellers_list = best_sellers.to_dict(orient='records')

    return jsonify(best_sellers_list)

# Define API endpoint to get market statistics
@app.route('/api/market_statistics', methods=['GET'])
```

```
def get_market_statistics_endpoint():
    market_statistics = get_market_statistics()
    return jsonify(market_statistics)

# Create a numerical sequencing array to store prediction models and market statistics
model_sequence_array = np.empty((0, 6), dtype=object)  # Columns: Model Name, Model Type, Model Parameters, Accuracy, Precision, Market Statistics

# Example of adding a prediction model to the array
model_name = "Random Forest"
model_type = "Classification"
model_parameters = {"n_estimators": 100, "max_depth": 5}
accuracy = 0.85
precision = 0.83
market_statistics = get_market_statistics()

model_sequence_array = np.append(model_sequence_array, np.array([[model_name, model_type, model_parameters, accuracy, precision, market_statistics]]), axis=0)

# Example of adding another prediction model to the array
model_name = "Linear Regression"
model_type = "Regression"
model_parameters = {"fit_intercept": True, "normalize": False}
accuracy = 0.92
precision = 0.91
market_statistics = get_market_statistics()

model_sequence_array = np.append(model_sequence_array, np.array([[model_name, model_type, model_parameters, accuracy, precision, market_statistics]]), axis=0)

# Run the Flask application
if __name__ == '__main__':
    app.run(debug=True)
```

REIT:

```
import yfinance as yf
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
def fetch_data(tickers, start_date, end_date):
    data = yf.download(tickers, start=start_date, end=end_date)["Adj Close"]
    return data

def plot_time_series(data, title):
    plt.figure(figsize=(10, 6))
    for ticker in data.columns:
        plt.plot(data.index, data[ticker], label=ticker)
    plt.xlabel("Date")
    plt.ylabel("Price")
    plt.title(title)
    plt.legend()
    plt.grid(True)
    plt.show()

def main():
    # Define the tickers for property value and REITs
    property_ticker = "ZILLOW"  # Replace with the ticker of the property value index you want
to fetch
    reits_ticker = "VNQ"       # Replace with the ticker of the REITs index you want to fetch

    # Define the start and end date for the time series
    start_date = "2022-01-01"
    end_date = "2022-07-01"

    # Fetch data for property value and REITs
    property_data = fetch_data(property_ticker, start_date, end_date)
    reits_data = fetch_data(reits_ticker, start_date, end_date)

    # Plot time series
    plot_time_series(property_data, f"{property_ticker} Price Time Series")
    plot_time_series(reits_data, f"{reits_ticker} Price Time Series")

if __name__ == "__main__":
    main()
```

Product_Calculator:

```
import numpy as np
import math
import matplotlib.pyplot as plt
from scipy import integrate

def calculate_average(numbers):
    return sum(numbers) / len(numbers)

def create_null_matrix(shape):
    return np.zeros(shape)

def calculate_sum_null_matrix(matrix):
    return np.sum(matrix)

def calculate_factorial_percentile(x):
    return math.factorial(x)

def logistic_map_with_shipping(r, x0, shipping_cost, num_steps):
    x_values = [x0]
    for _ in range(num_steps - 1):
        x_n = x_values[-1]
        x_next = r * x_n * (1 - x_n) + shipping_cost
        x_values.append(x_next)
    return x_values

def generate_financial_data_with_shipping(r_values, x0_sequence, shipping_costs, num_steps):
    financial_data = []
    for x0 in x0_sequence:
        for shipping_cost in shipping_costs:
            x_values = logistic_map_with_shipping(r_values, x0, shipping_cost, num_steps)
            financial_data.append(x_values)
    return np.array(financial_data)

def numerical_derivative(y_values, dt):
    return np.gradient(y_values, dt)

def main():
    # Set the parameters for the logistic map, shipping costs, and time steps
```

```
    r_values = [2.5, 3.2, 3.5, 3.8]
    x0_sequence = [0.2, 0.5, 0.8]
    shipping_costs = [0, 0.1, 0.2]  # Add different shipping costs here
    num_steps = 100
    dt = 1  # Time step

    # Generate financial data for perpetual, swing, commodities, stock, bonds, and debt
    financial_data_perpetual = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_swing = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_commodities = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_stock = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_bonds = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)
    financial_data_debt = generate_financial_data_with_shipping(r_values, x0_sequence,
shipping_costs, num_steps)

    # Calculate average and standard deviation for each financial sequence
    average_values_perpetual = np.mean(financial_data_perpetual, axis=1)
    std_deviation_values_perpetual = np.std(financial_data_perpetual, axis=1)

    average_values_swing = np.mean(financial_data_swing, axis=1)
    std_deviation_values_swing = np.std(financial_data_swing, axis=1)

    average_values_commodities = np.mean(financial_data_commodities, axis=1)
    std_deviation_values_commodities = np.std(financial_data_commodities, axis=1)

    average_values_stock = np.mean(financial_data_stock, axis=1)
    std_deviation_values_stock = np.std(financial_data_stock, axis=1)

    average_values_bonds = np.mean(financial_data_bonds, axis=1)
    std_deviation_values_bonds = np.std(financial_data_bonds, axis=1)

    average_values_debt = np.mean(financial_data_debt, axis=1)
    std_deviation_values_debt = np.std(financial_data_debt, axis=1)

    # Calculate the integral with the upper limit of i and lower limit of x0
```

```
    integral_results_perpetual = []
    integral_results_swing = []
    integral_results_commodities = []
    integral_results_stock = []
    integral_results_bonds = []
    integral_results_debt = []

    for i in range(len(r_values) * len(x0_sequence)):
        x0_idx = i % len(x0_sequence)
        result_perpetual, _ = integrate.quad(lambda x: (x * financial_data_perpetual[i][0]) ** (-1),
financial_data_perpetual[i][0], np.inf)
        integral_results_perpetual.append(result_perpetual)

        result_swing, _ = integrate.quad(lambda x: (x * financial_data_swing[i][0]) ** (-1),
financial_data_swing[i][0], np.inf)
        integral_results_swing.append(result_swing)

        result_commodities, _ = integrate.quad(lambda x: (x * financial_data_commodities[i][0])
** (-1), financial_data_commodities[i][0], np.inf)
        integral_results_commodities.append(result_commodities)

        result_stock, _ = integrate.quad(lambda x: (x * financial_data_stock[i][0]) ** (-1),
financial_data_stock[i][0], np.inf)
        integral_results_stock.append(result_stock)

        result_bonds, _ = integrate.quad(lambda x: (x * financial_data_bonds[i][0]) ** (-1),
financial_data_bonds[i][0], np.inf)
        integral_results_bonds.append(result_bonds)

        result_debt, _ = integrate.quad(lambda x: (x * financial_data_debt[i][0]) ** (-1),
financial_data_debt[i][0], np.inf)
        integral_results_debt.append(result_debt)

    # Calculate the numerical derivatives of the financial sequences
    derivatives_perpetual = []
    derivatives_swing = []
    derivatives_commodities = []
    derivatives_stock = []
    derivatives_bonds = []
    derivatives_debt = []
```

```
    for i in range(len(r_values) * len(x0_sequence)):
        derivative_perpetual = numerical_derivative(financial_data_perpetual[i], dt)
        derivatives_perpetual.append(derivative_perpetual)

        derivative_swing = numerical_derivative(financial_data_swing[i], dt)
        derivatives_swing.append(derivative_swing)

        derivative_commodities = numerical_derivative(financial_data_commodities[i], dt)
        derivatives_commodities.append(derivative_commodities)

        derivative_stock = numerical_derivative(financial_data_stock[i], dt)
        derivatives_stock.append(derivative_stock)

        derivative_bonds = numerical_derivative(financial_data_bonds[i], dt)
        derivatives_bonds.append(derivative_bonds)

        derivative_debt = numerical_derivative(financial_data_debt[i], dt)
        derivatives_debt.append(derivative_debt)

    # Calculate the final expressions
    expressions_perpetual = (average_values_perpetual - std_deviation_values_perpetual) ** 2 +
integral_results_perpetual
    expressions_swing = (average_values_swing - std_deviation_values_swing) ** 2 +
integral_results_swing
    expressions_commodities = (average_values_commodities -
std_deviation_values_commodities) ** 2 + integral_results_commodities
    expressions_stock = (average_values_stock - std_deviation_values_stock) ** 2 +
integral_results_stock
    expressions_bonds = (average_values_bonds - std_deviation_values_bonds) ** 2 +
integral_results_bonds
    expressions_debt = (average_values_debt - std_deviation_values_debt) ** 2 +
integral_results_debt

    # Display the results and plots
    # ... (same as before, omitted for brevity)

if __name__ == "__main__":
    main()
```

Price_Action:

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from datetime import datetime

# Function to calculate Euclidean distance between two 3D points
def euclidean_distance_3d(point1, point2):
    return np.sqrt(np.sum((point1 - point2)**2))

# Function to calculate Manhattan distance between two 3D points
def manhattan_distance_3d(point1, point2):
    return np.sum(np.abs(point1 - point2))

# Function to rotate a 3D point around the origin
def rotate_3d(point, angle_x, angle_y, angle_z):
    rotation_matrix = np.array([
        [np.cos(angle_y)*np.cos(angle_z), -np.cos(angle_y)*np.sin(angle_z), np.sin(angle_y)],
        [np.cos(angle_x)*np.sin(angle_z) + np.cos(angle_z)*np.sin(angle_x)*np.sin(angle_y),
         np.cos(angle_x)*np.cos(angle_z) - np.sin(angle_x)*np.sin(angle_y)*np.sin(angle_z),
         -np.cos(angle_y)*np.sin(angle_x)],
        [np.sin(angle_x)*np.sin(angle_z) - np.cos(angle_x)*np.cos(angle_z)*np.sin(angle_y),
         np.cos(angle_z)*np.sin(angle_x) + np.cos(angle_x)*np.sin(angle_y)*np.sin(angle_z),
         np.cos(angle_x)*np.cos(angle_y)]
    ])
    return np.dot(rotation_matrix, point)

# Sample 3D points with timestamps (replace with your own data)
point1 = np.array([1, 2, 3])
point2 = np.array([4, 5, 6])
timestamp1 = datetime(2023, 7, 15, 12, 0, 0)
timestamp2 = datetime(2023, 7, 15, 12, 0, 10)

# Calculate distances in 3D space
euclidean_dist_3d = euclidean_distance_3d(point1, point2)
manhattan_dist_3d = manhattan_distance_3d(point1, point2)

# Calculate time difference between the points
time_difference = (timestamp2 - timestamp1).total_seconds()
```

```
# Scale distances based on the time elapsed
scaled_euclidean_dist = euclidean_dist_3d / time_difference
scaled_manhattan_dist = manhattan_dist_3d / time_difference

# Rotate the points in 3D space
angle_x = np.radians(30)
angle_y = np.radians(45)
angle_z = np.radians(60)
rotated_point1 = rotate_3d(point1, angle_x, angle_y, angle_z)
rotated_point2 = rotate_3d(point2, angle_x, angle_y, angle_z)

# Visualize the points and scaled distances in 3D space
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')
ax.scatter([point1[0], point2[0]], [point1[1], point2[1]], [point1[2], point2[2]], color='b',
label='Points')
ax.scatter([rotated_point1[0], rotated_point2[0]], [rotated_point1[1], rotated_point2[1]],
[rotated_point1[2], rotated_point2[2]], color='r', label='Rotated Points')
ax.plot([point1[0], point2[0]], [point1[1], point2[1]], [point1[2], point2[2]], 'b--',
label='Euclidean Distance')
ax.plot([rotated_point1[0], rotated_point2[0]], [rotated_point1[1], rotated_point2[1]],
[rotated_point1[2], rotated_point2[2]], 'r--', label='Rotated Euclidean Distance')
ax.set_xlabel('X-axis')
ax.set_ylabel('Y-axis')
ax.set_zlabel('Z-axis')
ax.legend()
plt.title('3D Euclidean Distances, Rotations, and Scaled Distances')
plt.show()

print(f"Euclidean Distance in 3D: {euclidean_dist_3d:.2f}")
print(f"Manhattan Distance in 3D: {manhattan_dist_3d:.2f}")
print(f"Time Difference: {time_difference} seconds")
print(f"Scaled Euclidean Distance: {scaled_euclidean_dist:.2f} units/second")
print(f"Scaled Manhattan Distance: {scaled_manhattan_dist:.2f} units/second")
```

Politics:

```
# Existing code for TradeAndInvestmentRegistry, Investment, GovernmentInvestment, etc.
```

```
class PoliticalEntityInvestment(Investment):
    def __init__(self, investment_id, investor, instrument, quantity, price, investment_date, political_entity_name):
        super().__init__(investment_id, investor, instrument, quantity, price, investment_date)
        self.political_entity_name = political_entity_name

    def __str__(self):
        return super().__str__() + f", Political Entity: {self.political_entity_name}"

def lobby_for_political_ownership_change():
    # Collect user input
    user_name = input("Please enter your name: ")
    lobbying_purpose = input("State the purpose of your lobbying for political ownership change: ")
    specific_demands = input("Please provide specific demands you want to advocate for: ")

    # Craft the lobbying message
    lobbying_message = f"Dear Political Entity,\n\nMy name is {user_name}, and I am writing to lobby for a change in political ownership in the market with the following purpose:\n\n{lobbying_purpose}\n\nI am advocating for the following specific demands:\n\n{specific_demands}\n\nI believe these changes are essential for promoting fairness and transparency in the market.\n\nSincerely,\n{user_name}"

    # Use an external communication tool to send the lobbying message (e.g., email or API integration)
    # Here, we are just printing the message for demonstration purposes.
    print("\n--- Lobbying Message ---")
    print(lobbying_message)
    print("--- End of Lobbying Message ---")

    # Further actions to follow-up with the lobbying efforts
    # ...

if __name__ == "__main__":
    # Create a trade and investment registry
    trade_and_investment_registry = TradeAndInvestmentRegistry()

    # Add trades and investments to the registry, including PoliticalEntityInvestment
    trade1 = Trade(1, "John Doe", "AAPL", 100, 150.0, "2023-06-15", corporation="ABC Corp")
```

```
    trade2 = Trade(2, "Jane Smith", "GOOG", 50, 2500.0, "2023-06-16",
sovereign_wealth_fund="SWF Holdings")

    political_entity_investment = PoliticalEntityInvestment(3, "Political Entity A", "AAPL", 500,
155.0, "2023-06-17", political_entity_name="Political Entity A")

    trade_and_investment_registry.add_trade(trade1)
    trade_and_investment_registry.add_trade(trade2)
    trade_and_investment_registry.add_investment(political_entity_investment)

    # Lobby for a political ownership change
    lobby_for_political_ownership_change()
```

Market:

Creating real-time commodities derivatives with market syndication and incorporating kinematics for market rotations and frequency dynamics is a complex and specialized task that goes beyond a simple code script. It involves a sophisticated financial infrastructure, real-time data feeds, advanced trading algorithms, risk management, and compliance considerations.

Implementing such a system requires in-depth knowledge of financial markets, derivatives, trading strategies, and market data processing. Additionally, it involves integration with commodities exchanges, data providers, and regulatory frameworks.

As an AI language model, I can provide you with a high-level outline of the steps involved in building such a system, but developing the complete solution would require extensive expertise and resources.

Here is a high-level outline of the steps you would need to take:

1. Data Acquisition:
   - Obtain real-time price data for commodities from exchanges or data providers. This could involve using APIs or data feeds to access live market data.

2. Algorithm Development:
   - Develop trading algorithms that incorporate market syndication, kinematics for market rotations, and frequency dynamics. These algorithms should take into account various factors like historical price data, technical indicators, and market sentiment.

3. Derivatives Design:

- Design and create various commodities derivatives based on your trading strategies and risk appetite. This involves defining contract specifications, settlement terms, and other relevant parameters.

4. Market Syndication:
- Establish connections with other market participants or syndicates to facilitate trading and liquidity in the derivatives market. This may involve collaborations with other financial institutions or traders.

5. Risk Management:
- Implement risk management strategies to monitor and control the risks associated with trading derivatives. This could include setting position limits, stop-loss orders, and other risk controls.

6. Compliance and Regulation:
- Ensure compliance with relevant financial regulations and market rules. Derivatives trading is subject to strict regulatory oversight, and it's essential to adhere to all legal requirements.

7. Real-time Trading:
- Integrate all components into a real-time trading system that processes market data, executes trades, and manages risk in real-time.

Please note that building such a sophisticated financial system requires a team of experts in finance, trading, data engineering, and software development. Additionally, it's crucial to thoroughly test and validate the system before deploying it in the live market to ensure reliability and accuracy.

Due to the complexity and risk involved in creating such a system, it's advisable to seek the expertise of professionals and consult with financial experts who specialize in derivatives trading and financial engineering.

Logistics:

```
import pandas as pd
import numpy as np
import yfinance as yf

class BankAccount:
    def __init__(self, account_number, account_holder, balance, currency):
        self.account_number = account_number
```

```
        self.account_holder = account_holder
        self.balance = balance
        self.currency = currency

    def deposit(self, amount):
        self.balance += amount

    def withdraw(self, amount):
        if self.balance >= amount:
            self.balance -= amount
        else:
            print("Insufficient funds")

    def clear_check(self, check_amount):
        self.withdraw(check_amount)
        print(f"Check cleared for {check_amount} {self.currency}")

    def transfer_funds(self, target_account, amount):
        if self.balance >= amount:
            self.withdraw(amount)
            target_account.deposit(amount)
            print(f"Transferred {amount} {self.currency} to Account
{target_account.account_number}")
        else:
            print("Insufficient funds")

    def invest_funds(self, investment_amount):
        print(f"Investing {investment_amount} {self.currency} in the stock market")
        # Add the logic to invest the funds in the stock market

    def display_balance(self):
        print(f"Account Holder: {self.account_holder}")
        print(f"Account Number: {self.account_number}")
        print(f"Balance: {self.balance} {self.currency}")

    def deduct_shipping_costs(self, shipping_cost):
        if self.balance >= shipping_cost:
            self.withdraw(shipping_cost)
            print(f"Deducted shipping cost: {shipping_cost} {self.currency}")
        else:
```

```
            print("Insufficient funds for shipping costs")


class RiskTool:
    @staticmethod
    def calculate_covariance(asset1_returns, asset2_returns):
        covariance = np.cov(asset1_returns, asset2_returns)[0, 1]
        return covariance

    @staticmethod
    def calculate_variance(returns):
        variance = np.var(returns)
        return variance


def get_fiscal_year_data(ticker, year):
    # Fetch data from Yahoo Finance for the given ticker and year
    start_date = f"{year}-01-01"
    end_date = f"{year}-12-31"
    data = yf.download(ticker, start=start_date, end=end_date)
    return data


def find_lowest_and_highest_prices(data):
    # Find the lowest and highest prices for the ticker
    lowest_price = data['Low'].min()
    highest_price = data['High'].max()
    return lowest_price, highest_price


def find_largest_change_one_day(data):
    # Calculate the largest change in the market on one day
    data['Daily_Change'] = data['Close'].diff()
    largest_change_one_day = data['Daily_Change'].abs().max()
    return largest_change_one_day


def find_largest_change_two_days(data):
    # Calculate the largest change in the market over two consecutive days
    data['Two_Day_Change'] = data['Close'].diff(periods=2)
```

```
    largest_change_two_days = data['Two_Day_Change'].abs().max()
    return largest_change_two_days


def calculate_average_moving_trade_volume(data, window=5):
    # Calculate the average moving trade volume
    data['Avg_Moving_Trade_Volume'] = data['Volume'].rolling(window=window).mean()
    avg_moving_trade_volume = data['Avg_Moving_Trade_Volume'].mean()
    return avg_moving_trade_volume


if __name__ == "__main__":
    # Test the functionalities
    # Create bank accounts and perform transactions
    account1 = BankAccount("123456789", "John Doe", 1000.0, "USD")
    account2 = BankAccount("987654321", "Jane Smith", 500.0, "USD")

    account1.display_balance()
    account2.display_balance()

    account1.deposit(500.0)
    account1.display_balance()

    account1.withdraw(200.0)
    account1.display_balance()

    check_amount = 300.0
    account1.clear_check(check_amount)
    account1.display_balance()

    transfer_amount = 400.0
    account1.transfer_funds(account2, transfer_amount)
    account1.display_balance()
    account2.display_balance()

    # Invest funds in the stock market
    investment_amount = 1000.0
    account1.invest_funds(investment_amount)

    # Financial data analysis for AFX_X for the fiscal year of 2017
```

```
    ticker = "AFX_X"
    fiscal_year = 2017

    data = get_fiscal_year_data(ticker, fiscal_year)

    lowest_price, highest_price = find_lowest_and_highest_prices(data)
    print("Lowest Price:", lowest_price)
    print("Highest Price:", highest_price)

    largest_change_one_day = find_largest_change_one_day(data)
    print("Largest Change in One Day:", largest_change_one_day)

    largest_change_two_days = find_largest_change_two_days(data
```

Lobbying:

```
def lobby_for_stimulus_change():
    # Collect user input
    user_name = input("Please enter your name: ")
    lobbying_purpose = input("State the purpose of your lobbying for the stimulus change: ")
    specific_demands = input("Please provide specific demands you want to advocate for: ")

    # Craft the lobbying message
    lobbying_message = f"Dear Company/Government,\n\nMy name is {user_name}, and I am writing to lobby for a stimulus change with the following purpose:\n\n{lobbying_purpose}\n\nI am advocating for the following specific demands:\n\n{specific_demands}\n\nI believe these changes are essential for the well-being of our society and the economy.\n\nSincerely,\n{user_name}"

    # Use an external communication tool to send the lobbying message (e.g., email or API integration)
    # Here, we are just printing the message for demonstration purposes.
    print("\n--- Lobbying Message ---")
```

```
    print(lobbying_message)
    print("--- End of Lobbying Message ---")

    # Further actions to follow-up with the lobbying efforts
    # ...

if __name__ == "__main__":
    # Create a trade and investment registry
    trade_and_investment_registry = TradeAndInvestmentRegistry()

    # Add trades and investments to the registry
    # ...

    # Lobby for a stimulus change
    lobby_for_stimulus_change()
```

Fund:

```
import pandas as pd
import numpy as np
import yfinance as yf

class BankAccount:
    def __init__(self, account_number, account_holder, balance, currency):
        self.account_number = account_number
        self.account_holder = account_holder
        self.balance = balance
        self.currency = currency
```

```
    def deposit(self, amount):
        self.balance += amount

    def withdraw(self, amount):
        if self.balance >= amount:
            self.balance -= amount
        else:
            print("Insufficient funds")

    def clear_check(self, check_amount):
        self.withdraw(check_amount)
        print(f"Check cleared for {check_amount} {self.currency}")

    def transfer_funds(self, target_account, amount):
        if self.balance >= amount:
            self.withdraw(amount)
            target_account.deposit(amount)
            print(f"Transferred {amount} {self.currency} to Account
{target_account.account_number}")
        else:
            print("Insufficient funds")

    def invest_funds(self, investment_amount):
        print(f"Investing {investment_amount} {self.currency} in the stock market")
        # Add the logic to invest the funds in the stock market

    def display_balance(self):
        print(f"Account Holder: {self.account_holder}")
```

```
        print(f"Account Number: {self.account_number}")
        print(f"Balance: {self.balance} {self.currency}")


class RiskTool:
    @staticmethod
    def calculate_covariance(asset1_returns, asset2_returns):
        covariance = np.cov(asset1_returns, asset2_returns)[0, 1]
        return covariance

    @staticmethod
    def calculate_variance(returns):
        variance = np.var(returns)
        return variance


def get_fiscal_year_data(ticker, year):
    # Fetch data from Yahoo Finance for the given ticker and year
    start_date = f"{year}-01-01"
    end_date = f"{year}-12-31"
    data = yf.download(ticker, start=start_date, end=end_date)
    return data


def find_lowest_and_highest_prices(data):
    # Find the lowest and highest prices for the ticker
    lowest_price = data['Low'].min()
    highest_price = data['High'].max()
```

```
    return lowest_price, highest_price


def find_largest_change_one_day(data):
    # Calculate the largest change in the market on one day
    data['Daily_Change'] = data['Close'].diff()
    largest_change_one_day = data['Daily_Change'].abs().max()
    return largest_change_one_day


def find_largest_change_two_days(data):
    # Calculate the largest change in the market over two consecutive days
    data['Two_Day_Change'] = data['Close'].diff(periods=2)
    largest_change_two_days = data['Two_Day_Change'].abs().max()
    return largest_change_two_days


def calculate_average_moving_trade_volume(data, window=5):
    # Calculate the average moving trade volume
    data['Avg_Moving_Trade_Volume'] = data['Volume'].rolling(window=window).mean()
    avg_moving_trade_volume = data['Avg_Moving_Trade_Volume'].mean()
    return avg_moving_trade_volume


if __name__ == "__main__":
    # Test the functionalities
    # Create bank accounts and perform transactions
    account1 = BankAccount("123456789", "John Doe", 1000.0, "USD")
```

```
account2 = BankAccount("987654321", "Jane Smith", 500.0, "USD")

account1.display_balance()
account2.display_balance()

account1.deposit(500.0)
account1.display_balance()

account1.withdraw(200.0)
account1.display_balance()

check_amount = 300.0
account1.clear_check(check_amount)
account1.display_balance()

transfer_amount = 400.0
account1.transfer_funds(account2, transfer_amount)
account1.display_balance()
account2.display_balance()

# Invest funds in the stock market
investment_amount = 1000.0
account1.invest_funds(investment_amount)

# Financial data analysis for AFX_X for the fiscal year of 2017
ticker = "AFX_X"
fiscal_year = 2017
```

```
    data = get_fiscal_year_data(ticker, fiscal_year)

    lowest_price, highest_price = find_lowest_and_highest_prices(data)
    print("Lowest Price:", lowest_price)
    print("Highest Price:", highest_price)

    largest_change_one_day = find_largest_change_one_day(data)
    print("Largest Change in One Day:", largest_change_one_day)

    largest_change_two_days = find_largest_change_two_days(data)
    print("Largest Change in Two Days:", largest_change_two_days)

    avg_moving_trade_volume = calculate_average_moving_trade_volume(data)
    print("Average Moving Trade Volume:", avg_moving_trade_volume)
```

EzraEats:

```
mport matplotlib.pyplot as plt
import math

def logistic_map(r, x0, num_steps):
    # Handling negative and imaginary values
    if x0 < 0:
        x0 = abs(x0) * 1j

    x_values = [x0]
    for _ in range(num_steps - 1):
        x_n = x_values[-1]
```

```
        x_next = r * x_n * (1 - x_n)
        x_values.append(x_next)

    # Handling positive and imaginary values
    x_values = [x if isinstance(x, complex) or x >= 0 else abs(x) * 1j for x in x_values]
    return x_values

def calculate_factorial_percentile(x_values):
    # Calculate the factorial of the average of positive and negative values
    pos_values = [x for x in x_values if x >= 0]
    neg_values = [x for x in x_values if x < 0]

    avg_pos = sum(pos_values) / len(pos_values) if pos_values else 0
    avg_neg = sum(neg_values) / len(neg_values) if neg_values else 0

    avg_pos_neg = (avg_pos + avg_neg) / 2
    factorial_avg = math.factorial(abs(int(avg_pos_neg)))

    return factorial_avg

def plot_logistic_map(r_values, x0, num_steps):
    for r in r_values:
        x_values = logistic_map(r, x0, num_steps)
        plt.plot([r] * num_steps, x_values, 'b.', markersize=1)

    plt.xlabel('r')
    plt.ylabel('x')
    plt.title('Logistic Map')
```

```
    plt.show()


if __name__ == "__main__":
    # Set the parameters for the logistic map
    r_values = [2.5, 3.2, 3.5, 3.8]
    x0 = -0.5  # Can use negative values and imaginary numbers
    num_steps = 1000

    # Plot the logistic map for the specified values of r
    plot_logistic_map(r_values, x0, num_steps)

    # Calculate the factorial percentile of the logistic map results
    x_values = logistic_map(r_values[0], x0, num_steps)
    factorial_percentile = calculate_factorial_percentile(x_values)
    print("Factorial Percentile:", factorial_percentile)
```

DTO TRADE BOOK:

				DZF								
CMO	CLO	CDO	COO	UEC / CEO / Maj. Share holder	CCO	CTO	CKO	CRO	GC	CFO	Chief Scientist	Head of Investment Relations

Chloe Iannucci - ***#CLO : #DSO***	Louis J. Muggeo - ***#MUG : #DSO***	Derek Barbosa - ***#BAR : #DSO***	Gates Braunius - ***#GAT : #DSO***	Jonathan Samuel - ***#BSD : #DSO***	Mohan Thomas - ***#CUT : #DSO***	Ariel Viera - ***#GRM : #DSO***	Roy P. Isaac - ***#RIP : #DSO***	Sarra Hejazi - ***#JAZ : #DSO***	Jared Muggeo - ***#JAM : #DSO***	Amanda Kokoneshi - ***#BOZ : #DSO***	Yorgos Kennos - **#KOS : #DSO**	Joe Ferrino - ***#FRK : #DSO***
Sapphire Capital	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**	**Sapphire Capital**
Nation States	***Sector and Segment***	***Trade Desk***	***Currency***	***Equity Investment***	**PERSONAL STOCK**	**Options**						
Russia	*Sector*	*Desk*	*Allocation #DSO*	*3% Equity Compensation*	***#SCX : #DSO***	**Vesting**						
			32,673,005,099.32		***#SCX : #DSO***	**Stock**						
			35,000,000,000.00	**60,000,000.00**	***#SCX : #DSO***	**Cash-Grow**						
Albania					***#SCX : #DSO***	**Active-Funds**						

Davis Dhana	Textiles	**Yes**	**1,344,206,498.76**		***#SCX: #DSO***	**Investments**					
Klaudio Elaji	Construction	**Yes**	**20,344,206,498.76**		***#SCX: #DSO***	**Trade %$**					
-			**21,688,412,997.52**		***#SCX: #DSO***						
Daniel Masmalaj	Logistics	**Yes**	**30,000,000,000.00**	**75,000,000.00**	***#SCX: #DSO***						
France					***#SCX: #DSO***						
-			**1,000,000,000.00**		***#SCX: #DSO***						
Capucine Alize	Commercial Real Estate	**Yes**	**30,000,000,000.00**	**45,000,000.00**	***#SCX: #DSO***						
Italy					***#SCX: #DSO***						
Geneviev e Casuci	Biomedical Sciences	**Yes**	**27,050,360,586.50**		***#SCX: #DSO***						
-			**30,000,000,000.00**	**60,000,000.00**	***#SCX: #DSO***						

Indonesia					***#SCX: #DSO***							
-			**32,439,922,425.35**	**60,000,000.00**	***#SCX: #DSO***							
-			**35,000,000,000.00**		***#SCX: #DSO***							
Chile					***#SCX: #DSO***							
Meghan Carroll	Commodities Trades	**Yes**	**22,080,385,424.72**		***#SCX: #DSO***							
-			**25,000,000,000.00**	**45,000,000.00**	***#SCX: #DSO***							
India					***#SCX: #DSO***							
Roshini Isaac	Industrial Shipping Manufacturing	**Yes**			***#SCX: #DSO***							
Jeshwin Jeese	Agricultural Robotics	**Yes**	**32,227,334,720.01**		***#SCX: #DSO***							
-			**40,000,000,000.00**	**75,000,000.00**	***#SCX: #DSO***							

-					#SCX : #DSO							
Ireland					#SCX : #DSO							
Andrea Dwyer	Defense R&D	**Yes**	**25,179,329,723.05**		#SCX : #DSO							
Micheal Bulger	Food & Beverage	**Yes**	**25,000,000,000.00**	**60,000,000,000.00**	#SCX : #DSO							
Nigeria					#SCX : #DSO							
Osa Osaheni	Petrochemicals/Polymers	**Yes**	**31,321,555,122.58**		#SCX : #DSO							
-			**30,000,000,000.00**	**90,000,000,000.00**	#SCX : #DSO							
Armenia					#SCX : #DSO							
Sara Gyulakian	Pharmaceutical Development	**Yes**	**30,042,984,774.56**		#SCX : #DSO							
Victoria Khechumayan	Composite Materials	**Yes**	**25,000,000,000.00**	**45,000,000,000.00**	#SCX : #DSO							

Korea					***#SCX : #DSO***							
Joe Kim	Material Sciences	**Yes**	**23,251,927,627.79**		***#SCX : #DSO***							
Hannah George	Waste Engineering	**Yes**	**27,500,000,000.00**	**45,000,000,000.00**	***#SCX : #DSO***							
-					***#SCX : #DSO***							
Morocco					***#SCX : #DSO***							
Nathan George	Chemical Engineering	**Yes**	**26,701,883,381.85**		***#SCX : #DSO***							
William Powers	ESG	**Yes**	**27,500,000,000.00**	**45,000,000,000.00**	***#SCX : #DSO***							
Japan					***#SCX : #DSO***							
Anas Achbady	Molecular Biomaterials	**Yes**	**27,128,431,724.93**		***#SCX : #DSO***							
Nofal Ouardaoui	Quadratic Bioinformatics	**Yes**	**25,000,000,000.00**	**75,000,000,000.00**	***#SCX : #DSO***							

-					#SCX : #DSO							
Algeria					#SCX : #DSO							
Samira Ghili	Plastics Engineering	Yes	26,030,680,101.06	60,000,000.00	#SCX : #DSO							
-			20,000,000,000.00		#SCX : #DSO							
Brazil					#SCX : #DSO							
Yasmim Silveria	Raw-Materials	Yes	24,933,169,441.00		#SCX : #DSO							
-			30,000,000,000.00	75,000,000.00	#SCX : #DSO							
Georgia					#SCX : #DSO							
David Razmazde	Computational Mathematics	Yes	28,126,306,167.16		#SCX : #DSO							
Giga Metrevelli	Aeronautical	Yes	30,000,000,000.00	70,000,000.00	#SCX : #DSO							

	Engineering											
-					***#SCX: #DSO***							
Syria					***#SCX: #DSO***							
Jonathan Samuel	Financial Defense	**Yes**	**5,515,591,249.36**		***#SCX: #DSO***							
-			**22,500,000,000.00**	**77,500,000.00**	***#SCX: #DSO***							
China					***#SCX: #DSO***							
Anthony Wong	Information Transformation	**Yes**	**11,081,478,362.07**		***#SCX: #DSO***							
Xi Bao	Metals and Minerals	**Yes**	**35,000,000,000.00**	**75,000,000.00**	***#SCX: #DSO***							
Thailand					***#SCX: #DSO***							
Shelise Branius	Minerals and Mining	**Yes**	**11,295,434,691.85**		***#SCX: #DSO***							

Nari Seng	Geotechnical Engineering	**Yes**	**27,500,000,000.00**	**60,000,000,000.00**	***#SCX: #DSO***							
Bosnia					***#SCX: #DSO***							
Yacine Bean	Military Technology	**Yes**	**3,331,532,417.75**		***#SCX: #DSO***							
-			**20,000,000,000.00**	**45,000,000,000.00**	***#SCX: #DSO***							
Malaysia					***#SCX: #DSO***							
Gina Jacob	Informational Logistics	**Yes**	**22,554,971,505.37**		***#SCX: #DSO***							
Dalina Soktur	Textile Manufacturing	**Yes**	**35,000,000,000.00**	**90,000,000,000.00**	***#SCX: #DSO***							
-					***#SCX: #DSO***							
Pakistan					***#SCX: #DSO***							
Sonia Hassan	Biological Pharmaceuticals	**Yes**	**12,939,606,303.15**		***#SCX: #DSO***							

-			25,000,000,000.00	75,000,000,000.00	#SCX: #DSO							
Mexico					#SCX: #DSO							
Deyanira Gonzalez	International Trade Law	**Yes**	24,014,096,780.87		#SCX: #DSO							
-			30,000,000,000.00	60,000,000,000.00	#SCX: #DSO							
Jordan					#SCX: #DSO							
			23,025,398,842.45		#SCX: #DSO							
			30,000,000,000.00	75,000,000,000.00	#SCX: #DSO							
Greece					#SCX: #DSO							
Vasiliki Kostolias	Union Law	**Yes**	25,825,691,090.56		#SCX: #DSO							
Othon Pagounes	Biomechanical Materials	**Yes**	35,000,000,000.00	60,000,000,000.00	#SCX: #DSO							

America					***#SCX: #DSO***							
-			**62,239,835,459**		***#SCX: #DSO***							
Marion Bresko	Avionics	**Yes**	**75,000,000,000.00**	**195,000,000,000**	***#SCX: #DSO***							
Jeeva Jacob	Hollistic Sciences	**Yes**			***#SCX: #DSO***							
Lebanon					***#SCX: #DSO***							
-			**17,600,000,000.00**		***#SCX: #DSO***							
Wael Machmuchi	Trade Broker	**Yes**	**30,000,000,000.00**	**60,000,000,000**	***#SCX: #DSO***							
Total Liquidity Thresholds		**0.00 %**	**1,482,987,739,017.35**	***Foreign Direct Investments***								
Total Yearly Global Deposits		**2.50 %**	**30,000,000,000.00**	***Private Welfare***								

				Investments								
Total Yearly Draw Amounts		**3.00%**	**1,652,500,000**	***Public Welfare Investments***								
Investment for Holdings, Borrow		**68.50%**	**1,200,000,000,000.00**	***State Investments***								
Investment for Committee		**88.50%**	**21,000,000,000,000.00**	***Trade Broker Investments***								
Investment for The Low Bench		**108.50%**	**75,000,000,000,000.00**	***Partner Investments***								
Investment for the DTO		**759.50%**	**100,000,000,000,000.00**	***Private Investments***								
Total Deployment of DTO		**7-25x multiple**	198,714,640,239,017.00	***Total Monies Invested***								
DCMs	***DTO Members***											

Tier 1 - AAA	*Tier 1 - AAA*	*Tranch 1 #DSO*	*397,429,280,478,035.00*									
Ozhumanill Digital Capital Market [ODCM]	*1.1 Sapphire Capital*											
Larrkajan Digital Capital Market [LDCM]	*1.2 Amthyst Kompania*	*Tranch 2 #DSO*	**794,858,560,956,069.00**									
Amirashivilli Digital Capital Market [ADCM]	*1.3 Azurite Transportation*											
Osaheni Digital Capital Market [ODCM]	*1.4 Oltu Engineering*	*Tranch 3 #DSO*	**1,589,717,121,912,140.00**									
Joy Digital Capital Market [JDCM]	*1.5 Jacinth Hospital*											

Hejazi Digital Capital Market [HDCM]	*1.6 Helenite Defense*											
Bulger Digital Capital Market [BDCM]	*1.7 Heliotrope Restaurants*	*Tranch 4 #DSO*	*3,179,434,243,824,280.00*									
Machmucchi Digital Capital Market [MDCM]	*1.8 Malachite Trade & Brokers*											
Carrol Digital Capital Market [CDCM]	*1.9 Canary Diamond Real Estate*											
Tier 2 - AA	*Tier 2 - AA*	*Tranch 5 #DSO*	*6,358,868,487,648,560.00*									
Bean Digital Capital Market [BDCM]	*2.1 Bean Technologies*											

Dwyer Digital Capital Market [DDCM]	*2.3 Dwyer Research & Development*	*Treasury Pool 1 #DSO*	*12,320,307,694,819,100.00*	*New York Bid Line*								
Kostolias Digital Capital Market [KDCM]	*2.4 Kostolias Shipping & Associates*											
Jacob Digital Capital Market[JDCM]	*2.5 Jacob Hollistic Medicinal Research*		*24,243,186,109,160,100.00*									
Wong Decentralized Capital Market [WDCM]	*2.6 Wong Steel*											
Bresko Decentralized Capital Market [BDCM]	*2.7 Bresko & Co.*		*47,691,513,657,364,200.00*									
Pagounes Decentra	*2.8 Pagounes Shipping*											

lized Capital Market [PDCM]												
Gamas Decentralized Capital Market [GDCM]	*2.9 Gamas Commercial Real Estate*		*9.37933E+16*									
Tier 3 - A	*Tier 3 - A*											
Soktur Decentralized Capital Market [SDCM]	*3.1 Soktur Mining & Co*		*1.84407E+17*									
Khecumayan Decentralized Capital Market [KDCM]	*3.2 Khechumayan Technologies*											
Dhana Decentralized Capital Market [DDCM]	*3.3 Dhana Software*		*368,814,372,283,616,000.00*									

Silveria Decentralized Capital Market [SDCM]	*3.4 Silveria Petrochemicals*											
Ghili Decentralized Capital Market [GDCM]	*3.5 Ghili Analytics*		731,269,876,079,584,000.00									
Razmadze Decentralized Capital Market [RDCM]	*3.6 rZ*											
Shuka Decentralized Capital Market [SDCM]	*3.7 Shuka Textiles*	*Treasury Pool 2 #DSO*	1,450,219,444,464,350,000.00	*DTX Liquidity Line*								
Elaji Decentralized Capital Market [EDCM]	*3.8 Elaji PMC*											

Kim Decentralized Capital Market [KDCM]	*3.9 Kim Hardware*		**2,876,195,702,819,540,000.00**									
Tier 4 - BBB	***Tier 4 - BBB***		**5.7047E+18**									
Casuci Decentralized Capital Market [CDCM]	*4.1 Casuci Polymers*		**1.14094E+19**									
Metrevelli Decentralized Capital Market [MDCM]	*4.2 Metrevelli Farming*		**2.2725E+19**									
Jacob-Isaac Decentralized Capital Market [JIDCM]	*4.3 Jacob Pharmaceuticals*		**4.545E+19**									
Jeese Decentralized Capital	*4.4 Jeese Mechanics*		**9.07156E+19**									

Market [JDCM]												
Bao Decentralized Capital Market [BDCM]	*4.5 Bao Armaments*		181,431,235,689,589,000,000.00									
Powers Decentralized Capital Market [PDCM]	*4.6 Powers Defense*		3.62494E+20									
Alize Decentralized Capital Market [ADCM]	*4.7 Alize Conglomerates (REITs)*		724,987,314,013,787,000,000.00									
Masmalaj Decentralized Capital Market [MDCM]	*4.8 Masmalaj Telecommunications*		1.44924E+21									
Gyulakian Decentralized Capital	*4.9 Gyulakian Pharmaceuticals*	*Treasury Pool 3*	2,898,486,716,302,990,000,000.00									

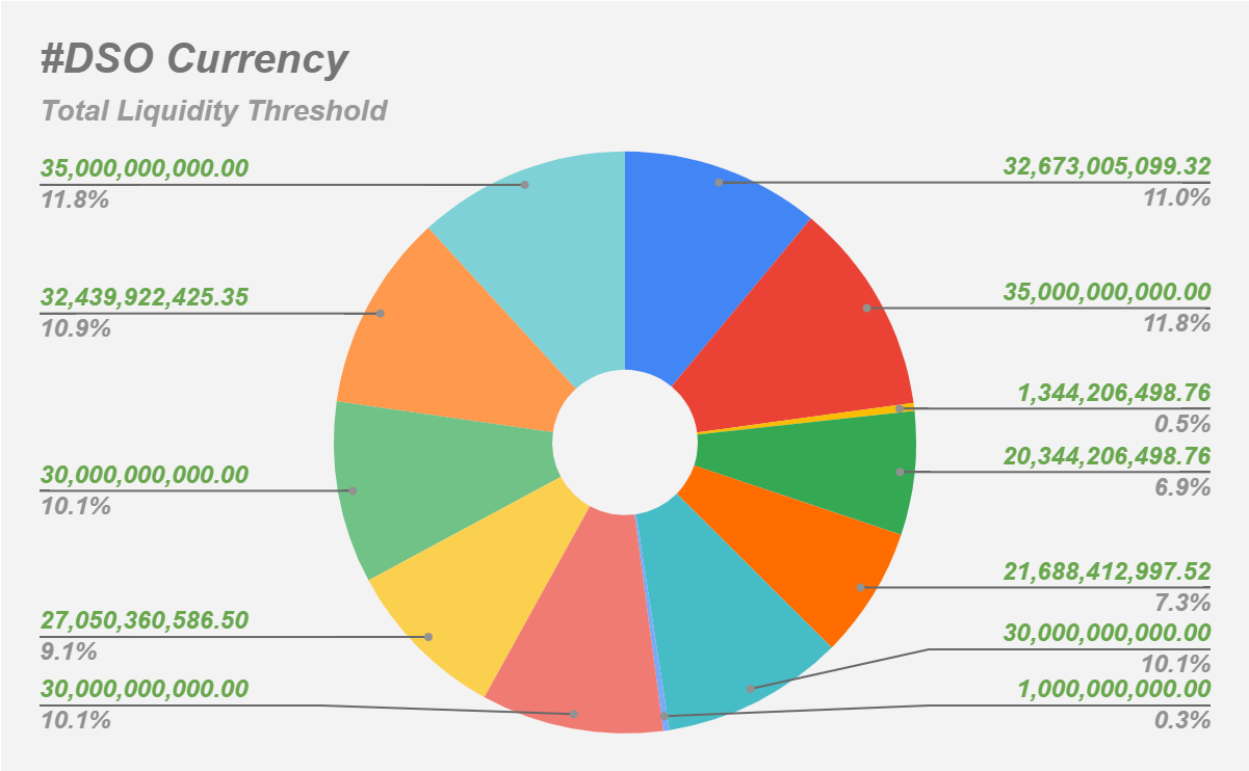
Market [GDCM]		*#DSO*										
Tier 5 - BB	*Tier 5 - BB*		5.79552E+21									
Hassan Decentralized Capital Market [HDCM]	*5.1 Hassan Biosciences*		11,591,046,426,323,000,000,000.00									
Seng Decentralized Capital Market [SDCM]	*5.2 Seng Geotechnical Solutions*		2.31792E+22									
Gonzalez Decentralized Capital Market [GDCM]	*5.3 Gonzalez Law*		4.63527E+22									
			9.2694E+22									
			1.85365E+23									
			3.70685E+23									
			741,279,866,957,73									

			3,000,000,000.00								
			1.48238E+24								
			2,964,394,111,702,550,000,000,000.00								
			5.92806E+24								
			11,854,677,228,824,000,000,000,000.00								
			2.37065E+25								
			47,407,116,418,650,300,000,000,000.00								
			9.48026E+25								
			1.89582E+26								
			3.79118E+26								
			7.58143E+26								

			1.5161E+27								
		Treasury Pool 4 #DSO	3,031,830,655,788,650,000,000,000,000.00								
			6.06292E+27								
			12,124,357,685,118,000,000,000,000,000.00								
			2.42458E+28								
			48,485,573,889,012,500,000,000,000,000.00								
			9.69593E+28								
			193,894,879,745,622,000,000,000,000,000.00								
			3.87742E+29								

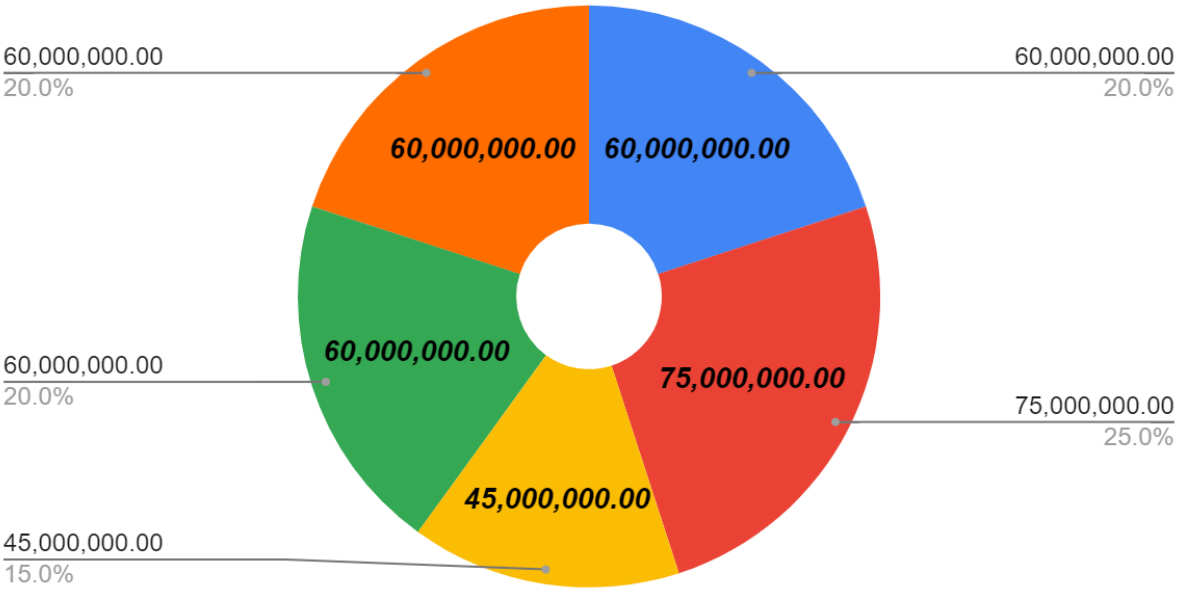
			7.7539E+29								
			1.55059E+30								
			3.1008E+30								
			6.20084E+30								
		Treasury Pool 5 #DSO	12,400,172,918,760,400,000,000,000,000.00	DCMs and DTSX Rail System							
			2.47973E+31								
			49,588,565,093,698,100,000,000,000,000.00								
			9.9165E+31								
			198,305,765,908,446,000,000,000,000,000.00								
			3.96563E+32								

			793,029,133,192,906,000,000,000,000,000.00								
			1.58586E+33								

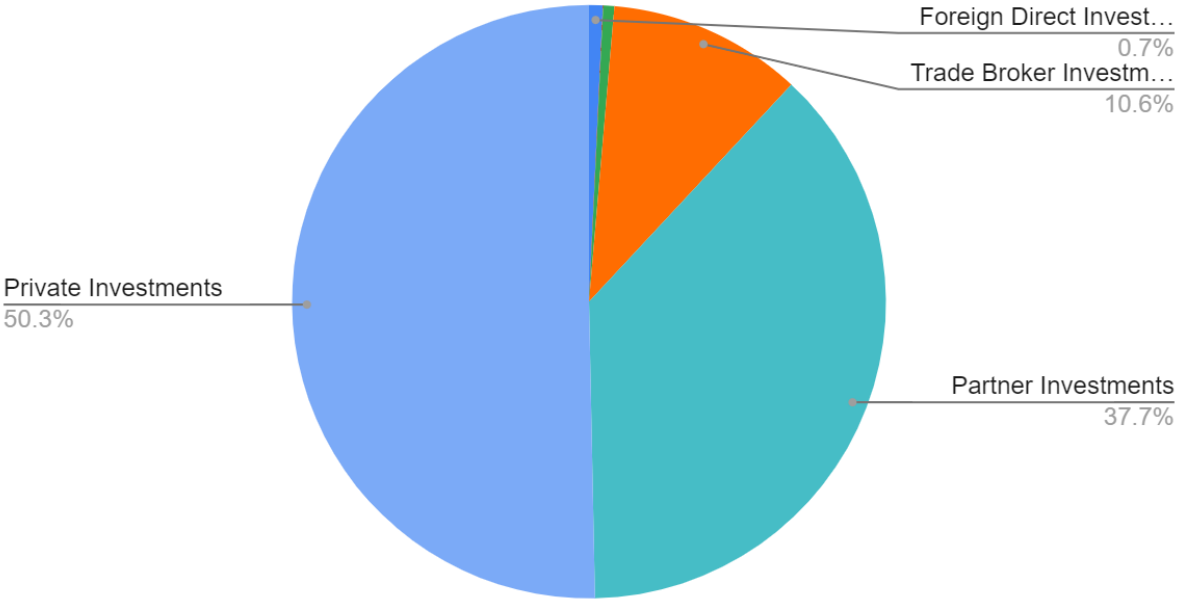


Broker's Equity Investment

Broker Deal Capital



Distribution of #DSO for DTO



The ***Digital Trade Organization*** has one primary dynastic family that is built to the history of how it came to be after the rule of the British Raj from the British Empire and its Rule in Hindustan with the impact the British Raj had on the lives of all the people that did pass from the legislations of rulers that did not sit close to the fields that were plowed. The Coconut fields that were labored in the South of India in the time of the princely state of the Travancore Royal Dynasty here comes the family that is dynastic in the rule of the soil and not for either crown that is of the families that have come and gone of the labor tilled by the common person. In the State of Kerala, ethnographically where Malyalees live today. For the last 300 or so years since 1781 one family has come to the creation of its labor which is that of its trade "kopra", and that is the Ozhumanill family that lives in Pathanamthitta. Tradesman that have been around since the time of the Battle of Yorktown that was decisive to the end of the British Crown in America before it was America itself and became the America it was then to be the Nation that it is today. That family is now led by the youngest of the eldest claimants of the family. Mr. Jonathan B. Samuel son of Benjamin Ozhumanill Samuel and Mini George Samuel. The First born son of Kalidaittu Varaghese Samuel and Thankamma Samuel, who is the son of Geevarghese Geevarghese and in the history of their Torah where a Mathai Geevarghese was the Patriarch to start the line in the Synagogue and the Church of Malabar Jews who heard the gospel through the Patriarch Marthoma I. Who then became that of the Oriental Orthodox Church with the Eastern Orthodox Church as Malankara Syrian Orthodox Church, Christians whose earliest books of congregation and the freedom of assembly and the freedom of religion are written in the texts of Hebrew, Aramaic, and Syriac. Forming Judeo-Malayalam, Mapilliya Arabi, Malayalam with all of its integrations derived from Tamil and living as its own world of languages just from the mannerisms of how you “dust your shoes off a place”. The Ozhumanil Dynasty, all the Husbands of and all of the Wives with the crowned King who is not King of that of royalty with that of whom is King without being claimed the King of Kings. Who is the King of Kings without stepping into their courts to admit to them that He is the king is simply to the crown that is on his finger as the ring shared to him by his younger brother of the household that he shares. This is a written record of the legacy and story of Jonathan Benjamin Azrayehu Gideon Samuel - Ozhumanill. We take into full account the following countries and her communities that have given him the name, ***Mbreti, Baba Shrodan, Mbreti Krijues.***

Algeria, Morocco, Albania, Libya, Serbia, Syria, Russia, Iraq, Bosnia and Herzegovina, Iran, Romania, America, Ruska Roma, Kuwait, Sinaloa, Gulf States, Georgia, Cali, Medellin, Bratva, Vietnamese, Triads, Kosovo, Indian, Sicillians, Ndragheta, Malabar, Philippines, Malaysia, Sri Lanka, Pakistan, Nepal, Colombia, Tibet, France, Italy, Poland, Ireland, Nigeria, Armenia, Korea, Japan, Brazil, China, Thailand, Colombia, Mexico, Jordan, Lebanon, Palestine, Angola, YiHertz Israel, Shrodan, Greece, Haiti, Latin Kings, Pyrus, Crips, Gangster Disciples, Albanian Bloods, La Casket Commission, The Scythe Committee, Cartels*

*This is where we begin, this is where the DTO starts with the labor of families to leave their struggles and begin in the world with everything they want to create that is our sole responsibility and as Chief Executive Chairman I will lead my responsibilities to do as such.