

# Godzillion: Un sistema de Crowdfunding Descentralizado para Startups

Rodrigo Sainz  
rodrigo@godzillion.io

Cristóbal Pereira  
cristobal@godzillion.io

Eduardo Portugues  
eduardo@godzillion.io

www.godzillion.io  
Agosto 2017

**Abstracto.** Una arquitectura de contratos inteligentes descentralizada que opere en la máquina virtual de Ethereum como un mercado de Startup Tokens primario y secundario (global), permitiría disminuciones de fricción en el proceso de emisión, aumento de seguridad en la mantención del registro de propiedad y las transacciones de intercambio de los Startups Tokens por valor sin pasar por ningún intermediario o servidor central. Las ejecuciones y confirmaciones descentralizadas de Blockchain proporcionan parte de la solución, disminuyendo los costos de intermediación y mantención de registros, pero los principales beneficios se pierden si se requiere todavía un tercero de confianza para monitorear startups para obtener acceso a un proceso de subasta y emitir sus tokens en el mercado de crowdfunding. Proponemos una solución para el problema centralizado de selección mediante el uso de un DApp Token (GODZ) como recompensa del sistema de votación, que ofrece incentivos económicos para que las personas participen en el proceso de selección; y si los votantes lo aprueban, las Startups puedan lanzar campañas de crowdfunding. Para ser publicado y atraer a los votantes, un Startup crea un Contrato inteligente de votación y le proporciona cierta cantidad de GODZ como un incentivo económico para los votantes en lugar de pagar esa tarifa a un tercero centralizado de confianza. Esta cantidad de GODZ estará disponible para los votantes ganadores (sí o no) como recompensa después de que finalice el período de la subasta, cualquiera que sea el resultado. Debido a que la tasa interna de rendimiento de los participantes en el proceso de votación depende de la recompensa del startup y la cantidad de votos (como GODZ) en cada lado de la votación, esperamos que los participantes compitan para alcanzar un equilibrio económico en el precio de votar.

**Godzillion**, el estado del arte de un mercado descentralizado,  
donde las personas **crean, votan e intercambian** Startups.

**Índice:**

- 1. Activos Alternativos en Blockchain, ¿por qué?**
- 2. El paradigma de la etapa temprana.**
- 3. Recompensas en la votación y sin comisión sobre el valor transado.**
- 4. Godzillion, un sistema de contratos inteligentes en el Ethereum Blockchain.**
- 5. GODZ, el Token, ¿para qué?**
  - 5.1. Votación / Evaluación**
  - 5.2. Subasta / Campaña de Crowdfunding**
  - 5.3. Exchange (intercambio)**
    - 5.3.1. Mercado GODZ / ETH**
    - 5.3.2. Mercado Startups Tokens / GODZ**
    - 5.3.3. Mercado Ethereum Tokens / GODZ**
  - 5.4. Transferencias e Historia**
- 6. Estado del Arte**
- 7. Token Crowdfunding Campaign y el uso de los ingresos**
- 8. Hoja de Ruta**
- 9. Historia Corporativa**
- 10. Apéndice**

### 1. Activos Alternativos en Blockchain, ¿por qué?

Uno de los mayores desafíos para los administradores de fondos de inversión global es encontrar mayores rendimientos para sus inversiones sin renunciar a la capacidad de liquidez de dichas inversiones.

Para lograr mayores rendimientos y bajas correlaciones, la diversificación se amplía a incluir activos que se conocen como activos alternativos, donde las principales categorías incluyen fondos inmobiliarios, fondos de capital privado, fondos de capital de riesgo e inversiones en etapas iniciales.

A modo de ejemplo, Cambridge Associates calculó que los fondos de capital de riesgo de *Early Stage* rentaron, en promedio, más del 55% anual (20 años).

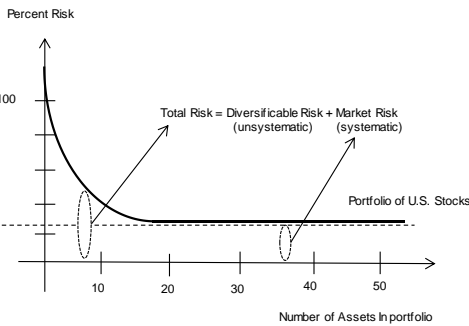
#### U.S. Venture Capital Index and Selected Benchmark Statistics

Index	1-Quarter	1-Year	3-Year	5-Year	10-Year	15-Year	20-Year	25-Year	30-Year
<b>Cambridge Associates LLC U.S. Venture Capital Index®</b>	- 3,25	4,78	19,36	14,63	10,19	4,86	30,02	25,10	18,21
<b>U.S. Venture Capital - Early Stage Index</b>	- 3,16	5,97	20,70	15,88	10,42	3,91	55,97	34,02	22,65
<b>U.S. Venture Capital - Late &amp; Expansion Stage Index</b>	- 2,59	2,57	15,25	11,29	12,16	6,83	9,99	13,03	12,40
<b>U.S. Venture Capital - Multi-Stage Index1</b>	- 3,61	3,54	18,55	13,80	9,21	6,17	11,13	14,03	12,05
Barclays Government/Credit Bond Index	3,47	1,75	2,42	4,04	4,93	5,03	5,62	6,23	6,57
Dow Jones Industrial Average Index	2,20	2,08	9,29	10,27	7,54	6,55	8,38	10,08	10,73
Dow Jones U.S. Small Cap Index	1,50	- 9,52	7,25	8,21	6,54	9,08	9,25	--	--
Dow Jones U.S. TopCap Index	1,06	0,86	11,58	11,37	7,14	6,23	8,00	--	--
Nasdaq Composite Index*	- 2,75	- 0,63	14,23	11,86	7,61	6,70	7,72	9,69	8,92
Russell 1000® Index	1,17	0,50	11,52	11,35	7,06	6,28	8,11	9,47	9,94
Russell 2000® Index	- 1,52	- 9,76	6,84	7,20	5,26	7,65	7,68	9,29	8,58
S&P 500 Index	1,35	1,78	11,82	11,58	7,01	5,99	7,98	9,28	9,93
Wilshire 5000 Total Market Index	1,18	0,24	11,26	11,01	6,95	6,62	8,04	9,40	9,75

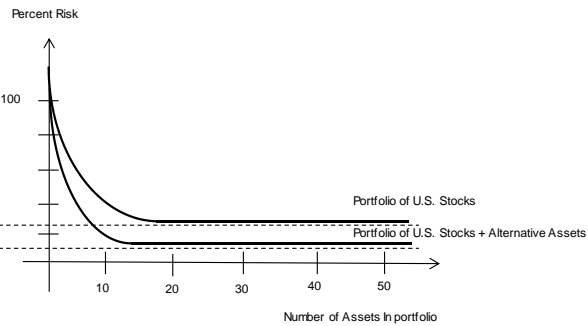
Source: <https://www.cambridgeassociates.com/benchmark/u-s-venture-capital-2016-q1/>  
Data as of March 31, 2016

Las decisiones de los administradores de fondos, de incluir o no estos activos, no están determinadas por los altos riesgos individuales de esos activos, sino que sus decisiones toman en cuenta el grado de correlación existente entre estos activos alternativos y sus activos clásicos, como las acciones y bonos de empresas o gobiernos.

Portfolio risk reduction through diversification



Portfolio risk reduction through diversification (including Alternative Assets)



Como los rendimientos de estos activos alternativos no fluctúan en sincronización con el aumento y caída de los mercados de acciones y bonos de los países industrializados (activos clásicos), sino que sus rendimientos son conducidos por diferentes factores causales, es que sus rendimientos no están correlacionados con los otros.

Esta baja correlación hace que sea posible crear portafolios que pueden generar mejores o iguales rendimientos, pero con una menor varianza total. La fórmula de la desviación estándar del retorno esperado así lo ilustra,

$$\sigma \text{ of the portfolio expected return} = \sqrt{w_{US}^2 \sigma_{US}^2 + w_{VC}^2 \sigma_{VC}^2 + 2 w_{US} w_{VC} \rho_{US-VC} \sigma_{US} \sigma_{VC}}$$

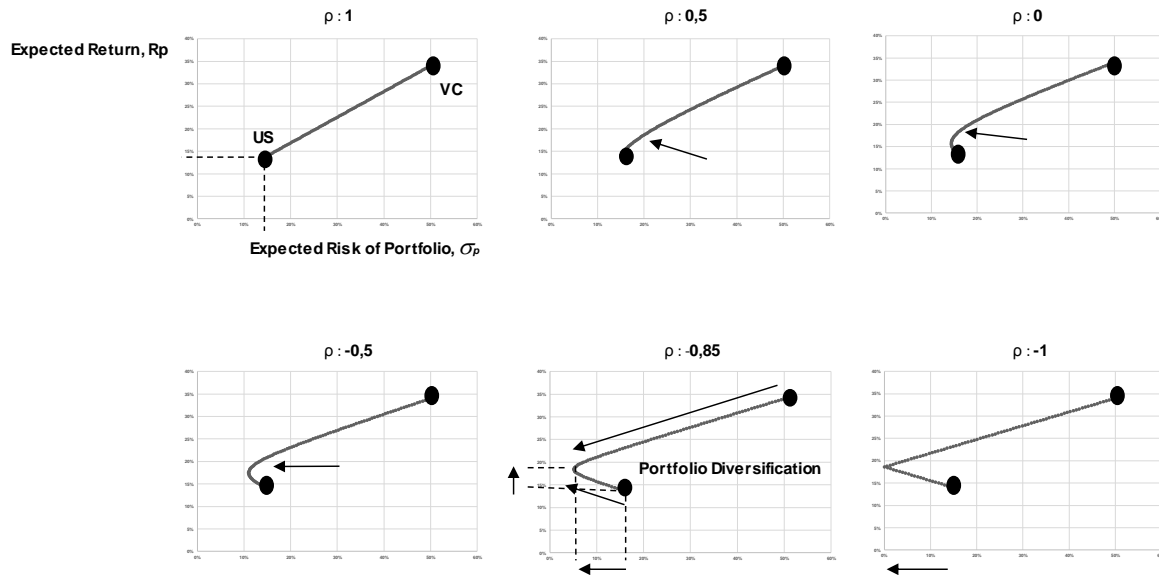
La fórmula indica que la varianza del retorno esperado de un portafolio compuesto por dos activos es el promedio ponderado de los retornos de ambos activos, ajustados por la covarianza de ambos activos. Si estos dos retornos están directamente relacionados, entonces la diversificación no reduce la varianza del rendimiento de la cartera. Pero si los rendimientos de los dos activos no están correlacionados, es posible que los retornos sean altos y estables, aún si los rendimientos de los activos individuales fueran volátiles.

Este enfoque funciona bien cuando más de dos activos se combinan para crear una cartera. Con tres o más activos, el rendimiento de la cartera puede ser mucho más estable.

En el siguiente gráfico es posible observar las ganancias por efecto diversificación, desde un coeficiente de correlación del 100% al -100%.

	Expected Return	Expected Risk ( $\sigma$ )
United States Equity Index (US)	14%	15%
Venture Capital index (VC)	34%	50%

**The Gains from Portfolio Diversification**



Se puede observar que, combinando dos activos, se puede optimizar la relación riesgo retorno. Por lo que, aunque los activos alternativos tengan una correlación baja o incluso negativa, se puede generar valor al portafolio diversificando, logrando un retorno mayor y más estable.

Esta lógica (diversificación) fue presentado por el profesor y premio nobel Harry Markowitz en 1952, la que ha sido puesta en práctica por las administradoras de inversiones. Colocar una parte de activos alternativos dentro de los portafolios clásicos es una manera de diversificar, y así mismo lograr un rendimiento esperado mucho mayor (con igual o menor riesgo).

El principal problema de esta estrategia es cuando los activos alternativos tienen poca o cero liquidez; eso coloca a los activos alternativos en una categoría menos atractiva y opaca el beneficio que pueden aportar. Esta falta de liquidez viene de la estructura convencional del mercado. No existe un mercado global agregado para los activos alternativos.

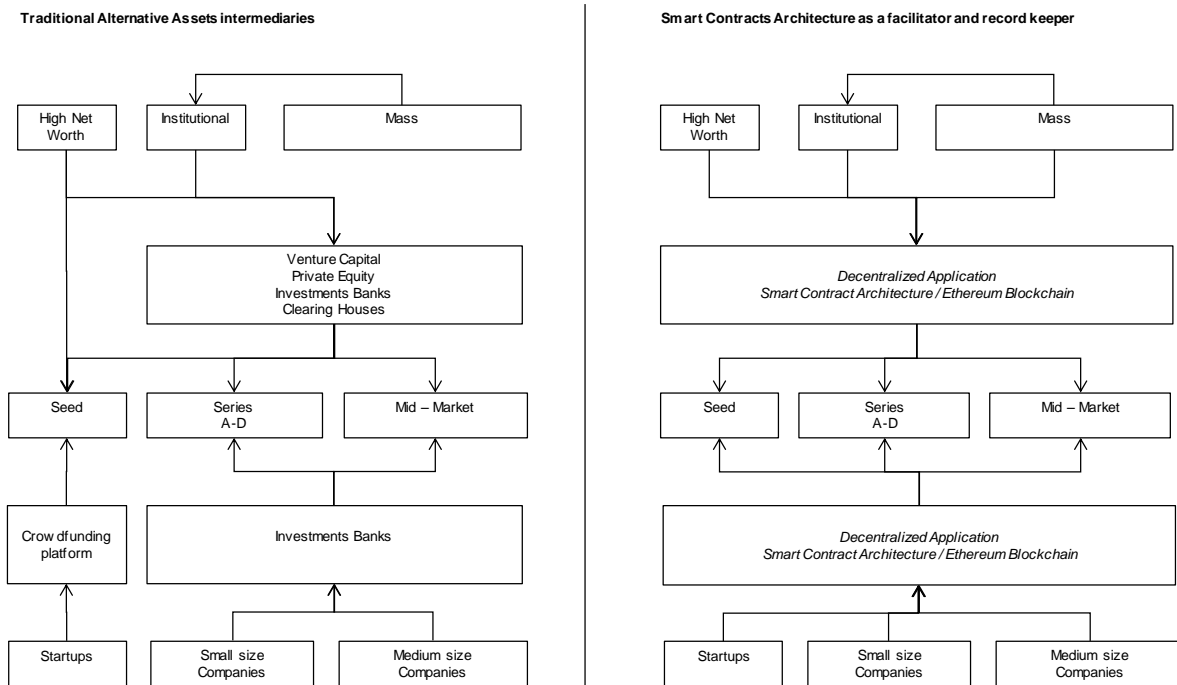
Los procesos de emisión y negociación son incómodos y arcaicos, manejados por diferentes intermediarios centralizados. Esto implica altos costos de contabilidad, custodia y auditoría, y hace que las transferencias de propiedad sean lentas y caras.

Por lo tanto, dada la estructura convencional de mercado, los activos alternativos tienen altas comisiones en los procesos de selección, emisión, seguimiento y auditoría, y pocas o nulas posibilidades de negociar (trading). Estos atributos hacen que sean inversiones con esperanza de "algún día ganar algo", sean difíciles de valorar y, por ende, menos adecuadas para muchos tipos de administradores de inversiones.

Todas estas fricciones del sistema financiero tradicional, disminuyen el rendimiento que puedan obtener los inversionistas, lo que afecta el valor de los activos alternos en el mercado secundario, y esa fricción hace más difícil que los emisores coloquen los valores en el mercado primario. En consecuencia, los inversores no pueden aprovechar las oportunidades que serían rentables si esos costos fueran más bajos y si la emisión y el comercio fuera más alineado.

**Para resolver este problema, hemos implementado (y está operando) un mercado descentralizado (global) primario y secundario de Activos Alternativos (semillas, startups y pequeñas empresas) basado en incentivos económicos y una arquitectura de contratos inteligentes, el cual opera en la máquina virtual de Ethereum. El proceso de emisión y comercialización de tokens que pueden representar la propiedad de los activos, se implementa utilizando una aplicación descentralizada (DApp). Lo llamamos Godzillion.**

La aplicación descentralizada funciona como un mecanismo para facilitar la emisión (a través del crowdfunding y el crowdfunding) y el intercambio (a través de swaps de Tokens). Opera continuamente y registra todas las transacciones en la cadena de bloques (Blockchain).



La arquitectura de los contratos inteligentes (Godzillion) que opera sobre el Blockchain de Ethereum logra varios objetivos: entrega incentivos económicos a las personas para que participen, disminuye la fricción en el proceso de emisión, aumenta la seguridad en los registros de propiedad y ejecuta intercambios de tokens por valor, sin pasar a través de un servidor intermediario o central.

Esto ayuda a remediar los inconvenientes que han impedido la aceptación de los activos alternativos en el sistema tradicional financiero. Descentraliza el incentivo económico de emisión y acelera en gran medida los engorrosos procesos de negociación y liquidación. Logra su objetivo con recompensas, menores costos de emisión y una mejor liquidez para vender y comprar activos alternativos en un mercado secundario descentralizado que opera y registra todo en Blockchain. Su diseño atrae el capital a sectores reales de la economía que pueden ofrecer altos retornos esperados sobre la inversión y con bajos grados de correlación con los activos clásicos optimizando los portafolios de los fondos de inversión.

**2. El paradigma de la etapa temprana.**

Las empresas pequeñas, para poder recaudar pequeñas cantidades de dinero y comenzar sus operaciones, deben ser evaluadas. Para ello, pagan una tarifa inicial a un banco de inversión o una empresa de crowdfunding. Si logran pasar ese primer obstáculo, deben pagar una segunda tarifa, como un porcentaje del capital recaudado en el proceso de subasta.

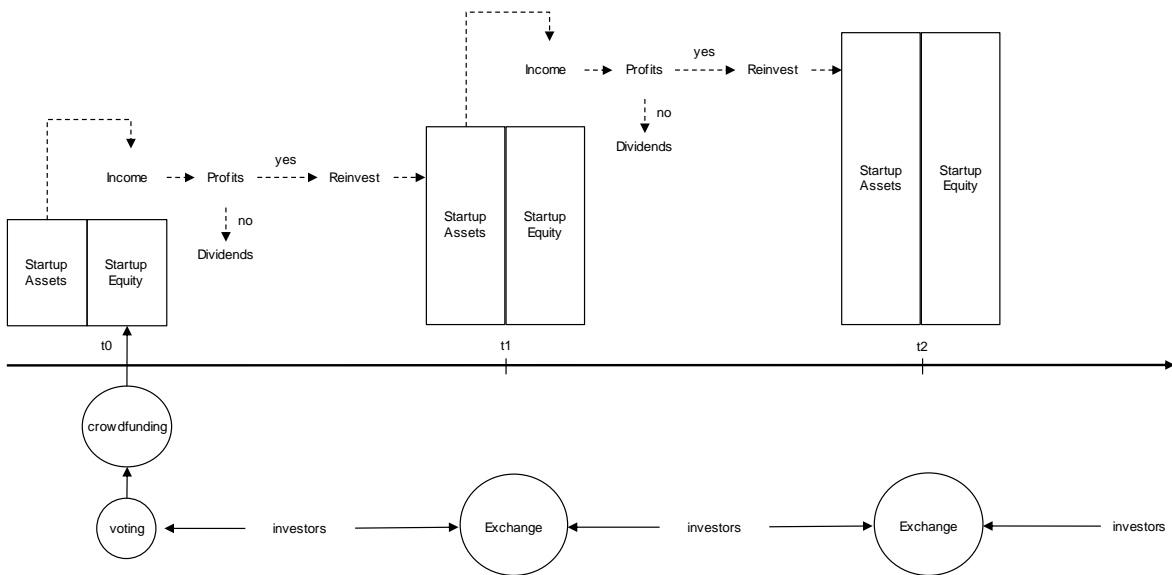
Los inversionistas de crowdfunding respaldan las emisiones de Startups y compran pequeños bloques de acciones por menos de 50 USD, pero después de este proceso primario de emisión, no hay un intercambio donde los inversores puedan negociar esos pequeños bloques de acciones a un bajo costo y con un procedimiento de liquidación y seteo de la transacción eficiente.

En los mercados actuales, con baja liquidez, un inversor en etapa de inversión temprana, no puede vender su posición, por lo que querrá recibir algunos dividendos, y mientras antes, mejor. Pero, para que una empresa pequeña crezca tan rápido como se lo permita el mercado, es aconsejable reinvertir todas las ganancias para impulsar así el crecimiento.

Por lo tanto, en la primera etapa de crecimiento, una empresa pequeña no tiene la capacidad de "devolver o pagar" a los inversores utilizando la clásica política de rendimiento de dividendos. Las empresas en etapa de arranque no deberían pagar dividendos justo después de comenzar a obtener ganancias. Después, una vez que se llena su nicho de mercado y se vuelve más maduro, ellos pueden pagar dividendos. Antes de ese momento, los inversores, en el pasado, concluían que estaban "bloqueados".

Un mercado secundario funcional para este tipo de empresas significa que las decisiones de inversión y reinversión de la startup pueden ser independientes de la necesidad de "retirar" de algunos de los primeros inversionistas, o de pagar dividendos cuando la empresa tiene buenas oportunidades para volver a invertir. Si los inversores supieran que pueden vender en cualquier momento, en lugar de pensar que necesitan mantener la inversión durante varios años antes de poder cosechar sus ganancias, nuestra plataforma sería el disruptor de ese paradigma.

Dividend yield versus expected capital gain



**Para resolver esos problemas, hemos construido un mercado primario que alinea los incentivos económicos entre todos los participantes, atrae a los votantes como una forma de evaluar qué startups podrán emitir los tokens, y un mercado secundario para estos pequeños bloques de acciones (tokens), que permiten a los inversionistas tomar decisiones independientes de una restricción del periodo de mantenimiento.**

La técnica que utiliza Godzillion es que los procesos de Votación, Emisión (campaña de crowdvoting y crowdfunding) y los intercambios de Inversores son ejecutadas utilizando una Arquitectura de Contratos Inteligentes en el Blockchain de Ethereum.

Este es un gran avance: no hay un servidor central, sino que una recompensa de votación extremadamente confiable, confirmación de órdenes y sistema de seguimiento, verificable públicamente y actualizado en tiempo real.

El resultado es una mejora cuántica para Inversores y Startups. Las propuestas de las empresas ahora son capaces de votarse de manera que se tengan los correctos incentivos económicos, y después puedan recaudar dinero con muy bajo costo de administración y transacción. Todo el proceso del paso a paso se implementa con una alta seguridad en el procedimiento de liquidación, de tal manera que los inversores puedan tener un registro verificado en todos los pasos del proceso de aumento de capital.

Usando Godzillion, los Inversores que deseen retirar dinero en cualquier momento, podrán publicar una orden ASK para sus bloques y, cuando otro inversionista paga ese precio ASK, pueden cosechar sus ganancias o pérdidas.

Con Godzillion DApp, las Startups pueden ser votadas, emitir tokens y lograr una mayor liquidez para sus primeros inversores, en el Blockchain de Ethereum, globalmente.

**3. Recompensas en la votación y sin comisión sobre el valor transado.**

Los intermediarios financieros, como los bancos o empresas de corretaje, cobran a sus clientes, en la mayoría de los casos, tarifas fijas, para luego cobrarles un porcentaje sobre el capital recaudado o comercializado, y un porcentaje de los activos bajo administración. El modelo de negocios clásico es actuar como coordinadores del proceso integral de inversiones y financiación de negocios, implementando intercambios y acuerdos entre diferentes partes del mundo y diferentes monedas, y ofreciendo diferentes libros contables privados para mantener los registros de los activos de los inversores en custodia.

En el modelo convencional de crowdfunding, los procesos son evaluar las propuestas de las Startups y las propuestas más aceptables publicarlas como subastas en sus sitios web. Estos sitios de crowdfunding atraen a los inversores, quienes deben crear cuentas en los sitios y luego ofertar en las subastas publicadas. Cuando la subasta alcanza su objetivo, el sitio de crowdfunding mueve el dinero que los inversionistas destinaron, desde las cuentas de los inversionistas a una cuenta bancaria central de la empresa dueña del sitio de crowdfunding, emiten un comprobante de compra para los inversores, registran toda la información en sus servidores y transfieren el dinero de su cuenta bancaria a la cuenta bancaria del emisor. Los registros de transferencia de este dinero son almacenados en el servidor del banco, y los registros de emisión están en el servidor de la compañía de crowdfunding.

En este enfoque tradicional, no existe un procedimiento mediante el cual, un comercio en el mercado secundario involucre estas acciones recién emitidas para que puedan registrarse y transferirse de manera rápida y segura, sino que, para realizar esta operación, el vendedor y comprador deben encontrarse físicamente entre sí, y luego deben actualizarse los registros de propiedad y pago en diferentes servidores. Este proceso puede tardar más de 5 días en liquidarse y los costos asociados pueden desalentar la emisión y la negociación.

Los intermediarios financieros que operan de acuerdo con el modelo de negocio clásico de crowdfunding reciben ingresos mediante el cobro de tarifas y honorarios iniciales sobre el capital recaudado. En algunos casos, se cobra, más del 10% del capital involucrado. Con Godzillion DApp, los pasos de la emisión e intercambio comercial cambian. Los inversionistas tienen incentivos económicos para poder ver y seleccionar las propuestas de inicio y las tarifas no se calculan como un porcentaje del capital o valor involucrado. En cambio, los costos son manejados por pasos computacionales requeridos para ejecutar la emisión o el intercambio en el Ethereum Blockchain.

		Centralized	Decentralized
<b>Process</b>  Voting / Screening ↓  Issuing ↓  Trading	Issuer	upfront fee to the crowdfunding company	reward to voters screening
	Investor	--	financial return per economic vote
	Issuer	% of the raised value to the crowdfunding company	computational steps
	Investor	% of the raised value to the crowdfunding company	computational steps
	Buyer	--	computational steps
	Seller	--	computational steps

En otras palabras, las tarifas son equivalentes a la energía y la potencia de cálculo requeridas para ejecutar una orden (suscripción o transacción), no como un porcentaje del valor involucrado en la transacción. El costo (tarifa) se basa en el cálculo de datos (medido en bytes), no en el valor que representan los datos.

**El cambio en el paradigma de cómo se evalúan las Startups y cómo se intercambia y registra el valor, implica una reducción sustancial en los costos operacionales y de implementación, aumenta el acceso de los mercados a la información, permite la transparencia del intercambio y mejora la seguridad del proceso de emisión y seteo.**



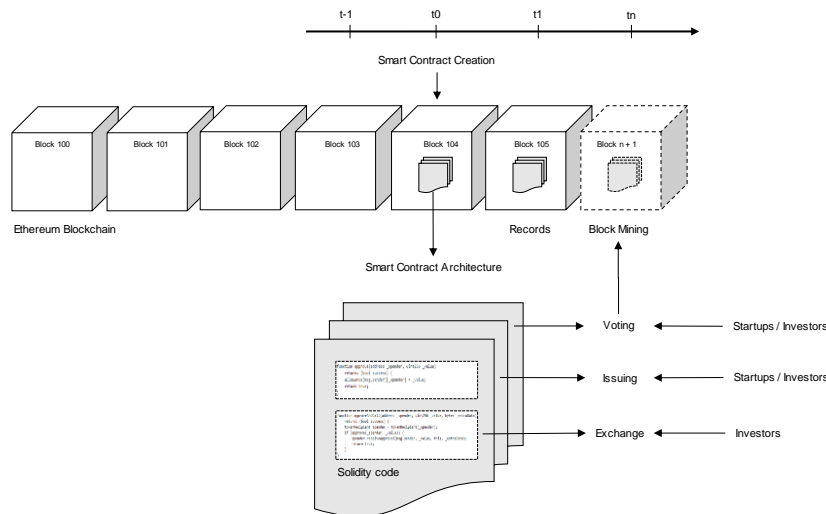
#### 4. Godzillion, un sistema de contratos inteligentes en el Ethereum Blockchain.

El protocolo Ethereum funciona como una máquina virtual descentralizada, donde se pueden implementar Contratos Inteligentes para ejecutar líneas de código predefinidas que realizan tareas específicas. La moneda criptográfica que opera como unidad de valor en el Ethereum Blockchain es el Ether (ETH) y el proceso básico de confirmación y liquidación es transferir ETH de una cuenta a otra, o a un contrato inteligente, que distribuirá el ETH de acuerdo con las instrucciones descritas en el contrato inteligente. En el protocolo de Ethereum, existe una estructura de costos definida y conocida para cada tipo de transacción que puede implementarse utilizando estos contratos y se establecen en la Máquina Virtual de Ethereum. Godzillion es una aplicación descentralizada (DApp) que se construyó como una arquitectura de contratos inteligentes en la máquina virtual de Ethereum. Por lo tanto, está codificada para ejecutar las solicitudes de los clientes, manteniendo los registros en el Ethereum Blockchain. Hoy, esas solicitudes pueden ser Transferencias, Votación, Campañas de financiación colectiva, Emisiones e Intercambio de Tokens.

Los proveedores de servicios de confirmación de transacciones (llamados mineros) son quienes validan las transacciones (en bloques) y mantienen una copia actualizada de todas las ganancias del proceso en el Blockchain. Los mineros obtienen recompensas, tanto por resolver un problema matemático difícil (Prueba de Trabajo), para validar un bloque entero de transacciones (5.0 ETH por bloque ganador), y por todo el gas gastado dentro del bloque (gas de los contratos que se ejecutaron dentro del bloque presentado por el minero ganador). Para lograr este objetivo, los mineros ejecutan computadoras (hardware) y nodos (Ethereum Virtual Machine) en todo el mundo compitiendo para confirmar las transacciones y obtener ETH como pago (hoy el proceso de confirmación se ejecuta como una prueba de trabajo; en un futuro cercano, el proceso probablemente será como una prueba de participación).

La competencia entre los mineros es en línea, y las recompensas son incentivos económicos; con esto se garantiza que haya muchos proveedores de servicios, entonces, el resultado es una Máquina de Confirmación Virtual descentralizada que es la clave de la Máquina Virtual Ethereum. La cantidad de nodos activos tiene valor, porque la capacidad de confirmar y almacenar datos ahora está completamente descentralizada. El Ethereum Blockchain no reside en un único servidor, sino que es un libro público único, almacenado en diferentes nodos en cada zona horaria, donde los mineros compiten para confirmar las transacciones y crear un nuevo Estado del Libro Mayor (sistema de contabilidad incorruptible).

Todas las transacciones de Godzillion se ejecutan, registran y almacenan en el Blockchain de Ethereum a través de la ejecución de una arquitectura de contratos inteligentes, por lo que las ejecuciones solicitan "gas" para ser procesadas y los datos se registran así en el Blockchain. Los proveedores de servicios (mineros), que confirman las transacciones ordenadas por los contratos inteligentes, actualizan el Blockchain, cobran por realizar ese servicio, y se les paga con "gas". Entonces, el "gas" que se necesita para operar un Contrato Inteligente depende de cuán compleja es la transacción, o de cuántas instrucciones incluya el Contrato Inteligente, o cuánto espacio en el Blockchain se requiera. Cuanta más complejidad haya en la solicitud, más gas se necesitará. La tarifa de "gas" no se calcula como un porcentaje del valor involucrado. En cambio, está alineado con la solicitud de tipo de ejecución que se implementará en el Ethereum Blockchain.



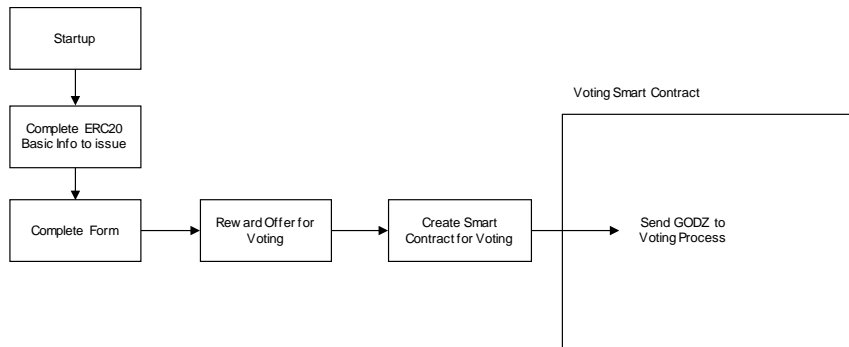
De lo anterior, para poder crear un mercado global de Startups, donde se vote, ejecute y registre en un libro de contabilidad global, el diseño de Godzillion incluye un Token que opera con la arquitectura de los contratos inteligentes. Este Token se llama GODZ y permite a los usuarios de Godzillion votar en Startups y ganar recompensas por votar, hace posible habilitar el mercado de las Startups a través de ETH y mejora el rendimiento de la ejecución de operaciones usando GODZ para intercambiar tokens de las Startups. Debido a que Godzillion es una aplicación totalmente descentralizada, todos los registros y detalles de liquidación se ejecutan de acuerdo con las prácticas de Ethereum, y se agregan a la cadena de bloques de Ethereum.

### 5. GODZ, el Token, ¿para qué?

GODZ permite a los usuarios votar si se debe permitir que una propuesta de un startup busque fondos en Godzillion; también permite a los usuarios comprar los Tokens de Activos Alternativos que se emitirán en Godzillion (Tokens de otros startups); y facilita el intercambio de esos activos. La arquitectura de los contratos inteligentes utiliza GODZ para permitir los votos, la emisión de los tokens y la distribución y control de los tokens de manera formal y estandarizada. Minimiza los costos y agiliza la ejecución, al tiempo que se registra por completo en el Ethereum Blockchain con una capacidad de seguimiento del 100%.

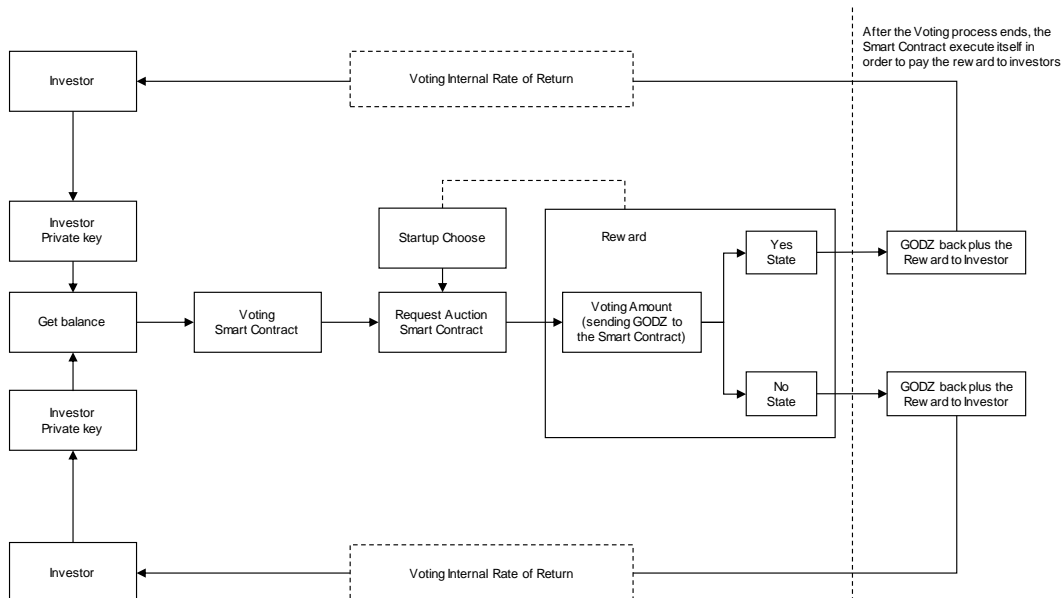
#### 5.1 Votación / Evaluación:

Los startups deben ser revisadas (votadas a favor por la multitud) para obtener acceso a un proceso de subasta y emitir sus tokens. Para ser publicado y atraer votantes, un Startup crea un Contrato inteligente de votación y le proporciona cierta cantidad de GODZ como incentivo económico para los votantes. Esta cantidad de GODZ estará disponible para los votantes ganadores (sí o no) como recompensa después de que finalice el período de la subasta, cualquiera que sea el resultado.



Utilizando Godzillion DApp, los CEOs de los Startups completan un Formulario, definen la recompensa y el monto en GODZ que se recaudará como el porcentaje del capital total de la compañía (porcentaje que se ofrecerá a los inversores a través de crowdfunding). Después de que se hayan establecido los parámetros de la emisión, se inicia el proceso de votación y los inversionistas pueden votar para que el startup pueda realizar una emisión de tokens o votar en contra para rechazar la propuesta. Los inversores pueden votar enviando a GODZ al Contrato de votación inteligente.

El procedimiento de votación es que un tenedor de GODZ estudia la propuesta publicada por el Startup y luego envía GODZ al contrato inteligente de votación. Los titulares de los tokens pueden establecer la cantidad de GODZ que envían al Contrato inteligente de votación y designar si los GODZ que envían están a favor de permitir que la propuesta avance, (un voto Sí) o en contra (un voto No). Luego, cuando el proceso de votación haya terminado, el Contrato inteligente de votación transferirá los GODZ a los titulares de los tokens que votaron. Los que votaron por el lado ganador obtienen sus GODZ más el 95% de GODZ que el Startup puso en el Contrato Smart de Votación, como una tarifa inicial por ser revisado; los titulares de tokens (los del lado perdedor) obtienen sus GODZ de vuelta más el 5% de la recompensa.





Para ilustrar en detalle cómo Godzillion implementa los procesos de votación colectiva, se presenta a continuación una parte del código del Smart Contract. Mostrando la función paso a paso. Se llama GodzStartupsVoting.

Esta primera parte del DApp establece la estructura del proceso de votación para una Startup. En el paso llamado declaración de estructura de la StartupVotingStruct dimos la estructura para almacenar los tokens en el Contrato inteligente de votación. El índice permite mantener sólo contratos inteligentes de las Startups que están en el proceso de votación.

```
function GodzStartupsVoting(address _godzContract)
{
    owner = msg.sender;
    godzContract = _godzContract;
}

struct StartupVotingStruct
{
    address startupContract;
    uint256 initialVotingDate;
    uint index;
}
```

Se requiere un Contrato inteligente de inicio del Contrato inteligente de votación y validamos que el inicio esté registrado en el administrador del Contrato de votación inteligente, para el proceso de votación. También hicimos una validación sobre el número de Startups que están en el proceso de votación.

```
function getStartupAtIndex(uint index)
public
constant
returns(address startupContract) {
    return startupVotingIndex[index];
}

function getStartupCount()
public
constant
returns(uint count) {
    return startupVotingIndex.length;
}
```

Esto es parte de la función, la que paga la recompensa a los inversores. En este caso, si la votación fue "NO", realizamos un proceso de cálculo para transferir la recompensa, dependiendo de si el "NO" fue el lado ganador de la votación o si fue el lado perdedor.

En caso de que sea el lado ganador, a los inversionistas se les paga con el 95% de la recompensa que la compañía de lanzamiento puso como incentivo para el proceso de votación. En caso de que el "NO" sea el lado perdedor, a los inversores se les paga con el 5% de la recompensa.

Después del proceso de cálculo, el contrato inteligente transfiere la recompensa a los inversores.

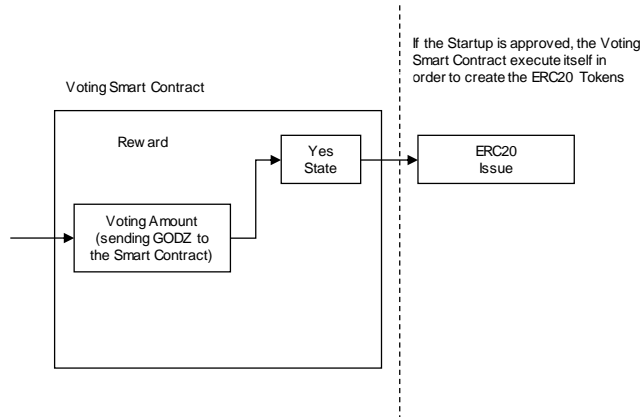
```
if (voteNo[startupContract][voter]>0)
{
    if (godzVotingTotalYes[startupContract]<godzVotingTotalNo[startupContract])
    {
        premioGodz = (GodzStartupInformation(startupContract).reward() * 95)/100;
    }
    else
    {
        premioGodz = (GodzStartupInformation(startupContract).reward() * 5)/100;
    }

    transferir = ((percent(voteNo[startupContract][voter],godzVotingTotalNo[startupContract],3) * premioGodz)*1000000000000000000)/1000;

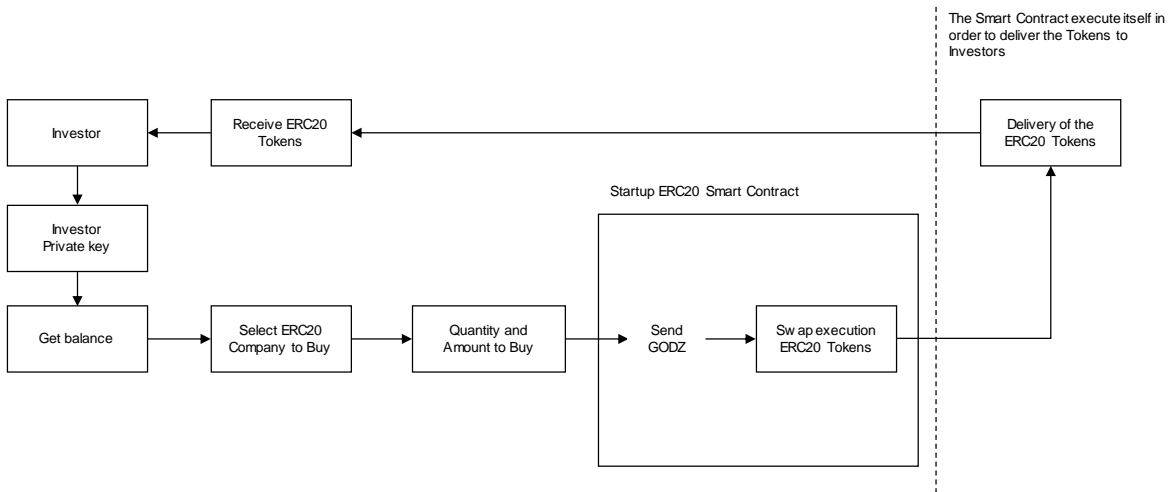
    return (voteNo[startupContract][voter], transferir, premioGodz);
}
```

### 5.2 Subasta / Campaña de Crowdfunding:

Después del proceso de selección, si se aprueba el inicio, comienza el proceso de subasta. Debido a que el proceso de subasta también se ejecuta con GODZ, se implementa y registra utilizando las invocaciones de Smart Contract. Este proceso es transparente y puede ser auditado por cualquier persona que use un explorador de blockchain.



En esta categoría, aparecerán todas las compañías que están dentro del proceso de subasta. Después, los tenedores de GODZ seleccionan las subastas a las que quieren ofertar. Este proceso implica dos resultados principales: los inversores reciben los tokens de los startups y los Startup reciben el capital que solicitaron a través de Godzillion (GODZ). Los tokens se entregan a las billeteras de los tenedores de GODZ que entraron en la subasta y los GODZ que fueron recaudados se entregan a la billetera en Ethereum de la Startup.



Para resumir, una vez que el inversor selecciona la cantidad de tokens del startup que desea ofertar y se valida el saldo de GODZ necesario para comprar esos tokens, en ese momento, cuando el inversor hace clic en Buy Tokens, envía el GODZ al contrato inteligente de la Startup y el contrato inteligente enviará automáticamente los tokens a la billetera del inversor y el GODZ a la billetera de Startup (Swap).

Este proceso de subasta terminará en el momento en que se haya vendido la cantidad total de tokens, sin importar si toma 1 día o 60 días. En ese momento, el startup tiene su convertible de GODZ en ETH y los inversionistas que quieran vender sus participaciones en acciones de Startup pueden publicar una orden de compra en el mercado descentralizado Godzillion.

Para implementar esos procesos, declaramos las variables que solicitamos como información básica para el contrato ERC20. Eso es en la forma que las Startups completan como un requisito previo para comenzar el proceso de votación.

Esta información que hemos recopilado, y que las Startups han proporcionado, es la información que utilizamos para emitir el contrato de ERC20 para el proceso de votación. Una vez que la declaración y las variables de entrada para la función que recopila la información del formulario, están completas, entonces se crea dos relaciones de vinculación: (i) entre la billetera del startup que está solicitando la votación y la información del contrato ERC20; y (ii) entre la billetera y la información sobre la información de valuación de la Startup.

Esta última información es una sección de la Parte A que dividimos para guardar toda la información del Inicio en una declaración de estructura.

```

function insertRelationStartup(
    string    _startupName,
    string    _startupSymbol,
    uint256   _startupSupply,
    uint256   _startupAmount,
    uint256   _startupReward,
    uint256   _startupPercentage,
    uint256   _startupValuation,
    uint256   _startupTokenPrice
)
{
    address _GodzStartupBasicInformation = new GodzStartupBasicInformation(
        _startupName,
        _startupSymbol,
        _startupSupply,
        _startupAmount,
        _startupReward
    );

    RelationWalletGodzStartupBasicInformation[msg.sender] = _GodzStartupBasicInformation;

    address _GodzStartupExtendedInformation = new GodzStartupExtendedInformation(
        _startupPercentage,
        _startupValuation,
        _startupTokenPrice,
        _GodzStartupBasicInformation
    );

    RelationWalletGodzStartupExtendedInformation[msg.sender] = _GodzStartupExtendedInformation;
}

```

En la sección del proceso de votación, cuando un inversor necesita revisar la información de la Startup, se realiza una solicitud al contrato inteligente que contiene la información de la Startup. Con la función `getStartupInformationPartA` se solicita la información de un determinado Startup que el inversor solicite, una vez que seleccione la empresa en el proceso de votación. Con la función de retorno, la información se implementa en el DApp para que el inversor pueda analizarla.

```

function getStartupInformationPartA(address _startupContract)
public
constant
returns (
    uint256   _startupPercentage,
    uint256   _startupValuation,
    uint256   _startupTokenPrice,
    string    _valueOfwebsite,
    string    _valueOfnameCEO,
    string    _valueOflinkedinCEO,
    string    _valueOfcountry
)
{
    return (
        StartupIssuingInfoStructs[_startupContract].startupPercentage,
        StartupIssuingInfoStructs[_startupContract].startupValuation,
        StartupIssuingInfoStructs[_startupContract].startupTokenPrice,
        valueOfwebsite[_startupContract],
        valueOfnameCEO[_startupContract],
        valueOflinkedinCEO[_startupContract],
        valueOfcountry[_startupContract]
    );
}

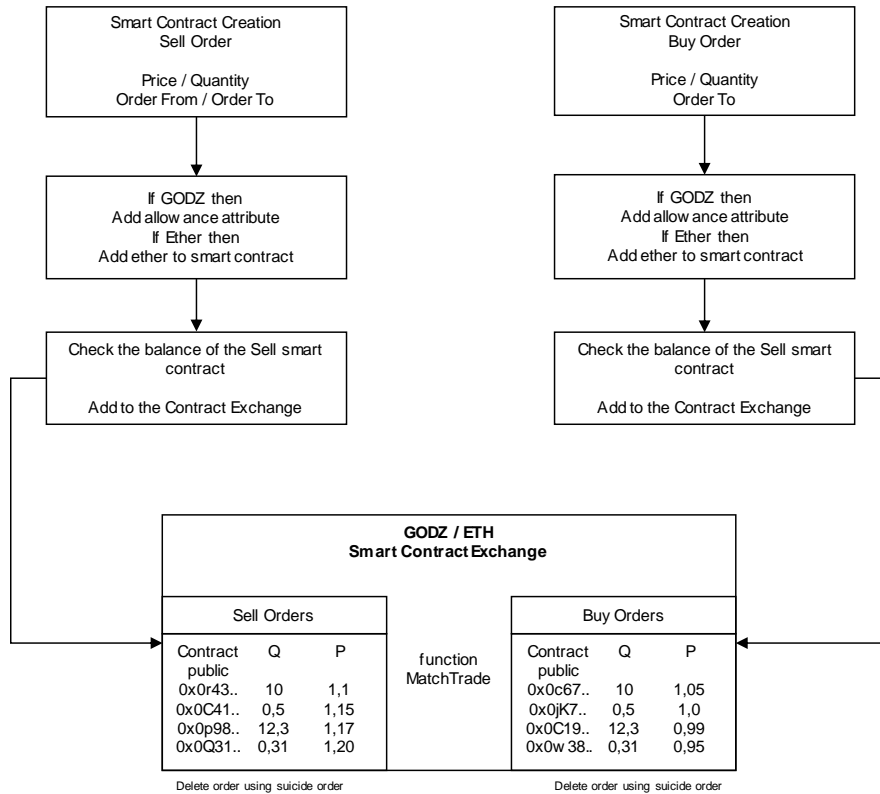
```

### 5.3 Intercambio:

GODZ pueden ser intercambiables en la DApp de Godzillion contra ETH, también intercambiable con los Tokens de un Startup específico, y también pueden ser intercambiado por Tokens emitidos en el Blockchain de Ethereum (como GNO y REP). En Godzillion hay tres mercados descentralizados: GODZ / ETH, Startups Tokens / GODZ; y Ethereum Tokens / GODZ.

### 5.3.1 Mercado GODZ / ETH:

Las personas pueden comprar GODZ utilizando ETH en el intercambio descentralizado de Godzillion. Pueden negociar ETH contra GODZ. En este intercambio, con el saldo de GODZ disponible en su billetera y el saldo de ETH, los inversores pueden publicar órdenes para comprar o vender GODZ.



Así como se describe en el siguiente código, declaramos una arquitectura de estructura de un contrato inteligente para crear el intercambio de GODZ para ETH. Las variables que se utilizan en diferentes contratos inteligentes que componen el intercambio Godzillion se declaran en esta parte del contrato. A continuación, con el mapa, organizamos la estructura de órdenes para el Contrato inteligente para una orden de Compra o Venta.

```

struct MarketOrderStruct
{
    address tokenFrom;
    address tokenTo;
    address userAddress;
    uint quantity;
    uint price;
    uint index;
}

mapping(address => MarketOrderStruct) private MarketOrderStructs;

address[] private marketOrderIndex;
    
```

Esto es parte de la función executeTrade que creamos para el intercambio. Si hay una operación, y validamos que la cantidad de una transacción es mayor que 0, se inserta la transacción en el contrato comercial del contrato inteligente, que guarda toda la información comercial ejecutada en el Exchange de Godzillion DApp.

```

if (trade(sellOrder, buyOrder, amount))
{
    if (quantity>0)
    {
        GodzExchangeTradeHistory(tradeHistory).insertTrade(tradeAddress, fecha, tipo, precioMenor, quantity, amount);
    }
}
    
```

Si la cantidad de la orden de venta es exactamente la misma de la orden de compra, entonces esta función se ejecuta. Aquí se transfiere la cantidad de tokens del vendedor al comprador, dado que existe una coincidencia entre los dos pedidos. Después de transferir los tokens, hacemos una actualización del libro de pedidos, actualizando el libro de pedidos para incluir el intercambio que se ejecutó.

```

if (MarketOrderStructs[sellOrder].quantity==MarketOrderStructs[buyOrder].quantity)
{
    Token(MarketOrderStructs[sellOrder].tokenFrom).transferFrom(
        MarketOrderStructs[sellOrder].userAddress,
        MarketOrderStructs[buyOrder].userAddress,
        MarketOrderStructs[sellOrder].quantity
    );

    marketBuyOrderGodz(buyOrder).transfer(MarketOrderStructs[sellOrder].userAddress,amount);

    marketBuyOrderGodz(buyOrder).changeQuantity(0);
    marketSellBuyOrderGodzToken(sellOrder).changeQuantity(0);

    updateOrderQuantity(buyOrder,0);
    updateOrderQuantity(sellOrder,0);

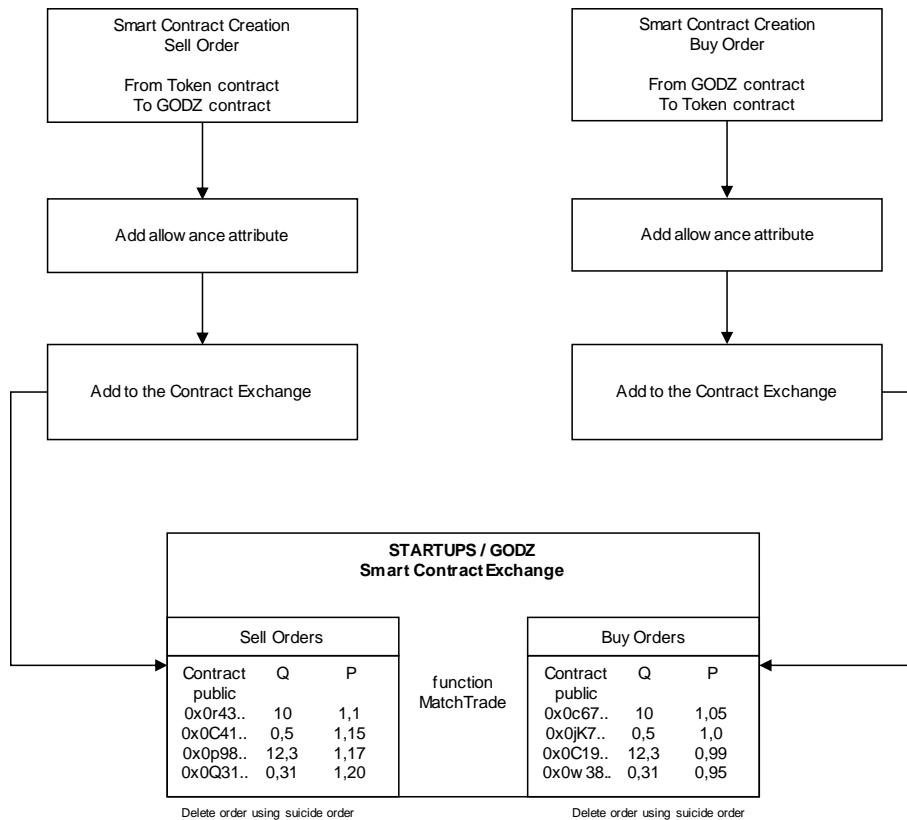
    return true;
}
    
```

**5.3.2 Mercado Tokens de Startups / GODZ:**

Cuando se haya cerrado el proceso de subasta, los tokens emitidos por la Startup estarán disponibles inmediatamente para ser intercambiados en el mercado de startups de Godzillion. Usando GODZ, los inversores pueden comprar o vender tokens de la Startup en la que han participado, o tokens de otras Startups que ya están incluidas en la lista de la DApp.

Seleccionando el token que el inversionista quiera intercambiar, tendrá acceso a la dirección pública del ERC20, y también a un botón donde se puede revisar la información de este contrato inteligente en etherscan.io.

Con el saldo de GODZ disponible en la cartera de inversores y el saldo de tokens de la empresa seleccionada, los inversores pueden publicar pedidos para comprar o vender el token del startup.





En este caso, la función que se usa para validar si la dirección del Smart Contract es una orden de compra o venta, está dado por el índice de corrección. Si el índice de corrección es 0, todavía no hay un pedido completo, como se describe en el siguiente código:

```
function isOrder(address userMarketOrderContract)
    public
    constant
    returns(bool isIndeed) {
        if (marketOrderIndex.length == 0) return false;
        return (marketOrderIndex[MarketOrderStructs[userMarketOrderContract].index] == userMarketOrderContract);
    }
```

Para la función creada para insertar un pedido de Compra o Venta en el intercambio, primero se valida si el pedido existe en nuestra plataforma. Si existe entonces se arroja.

Luego almacenamos la dirección pública de la billetera que emite la orden, junto con la cantidad, precio, del token y el índice de la orden en el Exchange de Godzillion. Con este proceso, la orden se almacena en el intercambio, a la espera de una coincidencia para intercambiar.

```
function insertOrder(
    address userMarketOrderContract,
    address userAddress,
    uint quantity,
    uint price,
    address tokenFrom,
    address tokenTo)
    public
    returns(uint index) {
        if (isOrder(userMarketOrderContract)) throw;
        MarketOrderStructs[userMarketOrderContract].userAddress = userAddress;
        MarketOrderStructs[userMarketOrderContract].quantity = quantity;
        MarketOrderStructs[userMarketOrderContract].price = price;
        MarketOrderStructs[userMarketOrderContract].tokenFrom = tokenFrom;
        MarketOrderStructs[userMarketOrderContract].tokenTo = tokenTo;
        MarketOrderStructs[userMarketOrderContract].index = marketOrderIndex.push(userMarketOrderContract) - 1;
        return marketOrderIndex.length - 1;
    }
```

Si una cantidad de una orden de compra o venta necesita actualizarse, por ejemplo: Existe una operación para comercializar esta orden, pero el vendedor o comprador no toma la cantidad total de la orden. Por ejemplo, si alguien vende 20 tokens, pero alguien sólo le compra 10, la orden de venta ahora se actualiza por 10 tokens restantes para vender.

Aquí parte de la función,

```
function updateOrderQuantity(address userMarketOrderContract, uint quantity)
    public
    returns(bool success) {
        if(!isOrder(userMarketOrderContract)) throw;
        MarketOrderStructs[userMarketOrderContract].quantity = quantity;
        return true;
    }
```

### 5.3.3 Mercado Tokens de Ethereum / GODZ

Si un Inversor no tiene ETH o GODZ en su billetera, pero tienen otros tipos de tokens que se han emitido en la Cadena de bloques de Ethereum, también se pueden intercambiar esos tokens por GODZ.

Para poder hacer esto, en esta sección los inversores deben agregar un token con la dirección pública del ERC20 (token emitido en Ethereum Blockchain), luego hacer clic en el botón Validar ERC20. La DApp buscará la información de ese ERC20 en la cadena de bloques, y mostrará el nombre asociado con ese ERC20, el suministro total de tokens, el símbolo y los decimales.

Si la información que aparece es correcta, los Inversores pueden hacer clic en el botón Agregar Token y el DApp agregará este token al Smart Contract Exchange.

Entonces, una vez que el proceso ha terminado, los inversores pueden intercambiar estos tokens con GODZ. Cuando este intercambio tiene al menos un token para negociar, los inversores pueden seleccionar el token que quiere intercambiar. Después de que el inversor selecciona, el DApp muestra la dirección pública del ERC20 y el saldo de los tokens que el inversor tiene en su billetera.

Además, se mostrará el saldo de GODZ que tienen los inversores. Con el saldo de GODZ disponible en la billetera del inversor y el saldo de los tokens añadidos, los inversores pueden emitir órdenes para comprar o vender los tokens.

Como parte del proceso de intercambio, se usa la declaración de estructura, porque diferentes contratos inteligentes interactúan para crear este intercambio. En este caso, la estructura para el intercambio de tokens, usa el mapeo para ordenar los tokens que se pueden intercambiar en el intercambio.

```

struct MarketTokenStruct
{
    address exchange;
    address trades;
    uint index;
}

mapping(address => MarketTokenStruct) private MarketTokenStructs;

address[] private marketTokenIndex;

```

Se usan `getTokens` para obtener y la historia de los intercambios de algún token que esté dentro de la lista del Exchange.

Esto se usa para obtener información de compra o venta de los contratos inteligentes o el historial de operaciones que se ejecutaron.

```

function getToken(address erc20)
    public
    constant
    returns(address exchange, address trades, uint index) {
    return (
        MarketTokenStructs[erc20].exchange,
        MarketTokenStructs[erc20].trades,
        MarketTokenStructs[erc20].index);
}

```

El índice de orden es para ordenar la solicitud de información dentro del intercambio. En la primera función, el pedido de mercado del contrato inteligente es requerido para obtener el índice del token dentro de ese contrato.

En la segunda función, se realiza un recuento de la cantidad de tokens del intercambio para un token seleccionado.

```

function getTokenAtIndex(uint index)
    public
    constant
    returns(address userMarketOrderContract) {
    return marketTokenIndex[index];
}

function getTokenCount()
    public
    constant
    returns(uint count) {
    return marketTokenIndex.length;
}

```

## 5.4 Transferencias e historia:

Con el servicio de Balance y Transferencia, los usuarios pueden revisar el saldo de GODZ, tokens y ETH de sus billeteras. Además, pueden hacer transferencias de GODZ, ETH y otros tokens de Ethereum a otras billeteras.

Para ejecutar una transferencia de tokens ERC20 a otras billeteras, el valor de la transferencia y el contractfrom se utilizan para permitir que la billetera transfiera la cantidad de tokens a la otra billetera.

La información se envía para implementar la transferencia de acuerdo con los decimales que se emitió el token, porque hay diferentes tipos de tokens listados en nuestro Exchange.

Además, los usuarios pueden revisar toda la información relacionada con los intercambios realizados en Godzillion DApp con su clave privada. Pueden revisar la fecha, el tipo de transacción (si fue una compra o una transacción de venta), el valor asociado con la operación y finalmente el token intercambiado.

Aquí parte del código,

```
window.transERC20 = function (to, value, ContractTokenFrom) {
  var contractAddress = ContractTokenFrom;

  var solidityFunction = new SolidityFunction('', _.find(abiToken, { name: 'transfer' }));

  var contractPow = web3.eth.contract(abiToken).at(ContractTokenFrom);
  var powToken = Math.pow(10, contractPow.decimals.call().toNumber());

  var valueSend = (value * powToken);

  var payloadData = solidityFunction.toPayload([to, valueSend]).data;
  var nonce = web3.eth.getTransactionCount('0x'+address);
  var nonceHex = web3.toHex(nonce);
  var gasPrice = web3.eth.gasPrice;
  var gasPriceHex = web3.toHex(gasPrice);
  var gasLimitHex = web3.toHex(4700000);
  var privateKey = returnBuffer($('#pk').val());
  var rawTx = {
    nonce: nonceHex,
    gasPrice: gasPriceHex,
    gasLimit: gasLimitHex,
    to: contractAddress,
    from: '0x'+address,
    data: payloadData
  };
  var tx = new Tx(rawTx);
  tx.sign(privateKey);
  var serializedTx = tx.serialize();
  web3.eth.sendRawTransaction('0x' + serializedTx.toString('hex'),
    function (err, hash) {
      if (!err) {
        $('#trx').val(hash);
        waitForTransactionReceipt(hash);
      }
    });
}
```

**6. Estado del Arte.**

**6.1 Votación:**

- 1) El usuario debe poner la clave privada de su billetera, para activar esta sección del DApp. Aquí, el usuario puede revisar su saldo de GODZ y el último número de Bloque del Blockchain de Ethereum.
- 2) El usuario puede seleccionar la Startup sobre la que desea votar, y con la selección de la Startup, aparecerá la recompensa que la compañía pagó para ser revisada y cuántos días quedan para que finalice este proceso de votación. El inversor puede hacer clic en el botón "Revisar y analizar para votar" para leer toda la información que la empresa completó en el formulario que rellenó en su solicitud.

### Voting Process

**1** Your Private Key

godz Balance

Block Number

[get your balance & rewards](#)

---

**2** Select the company that you want to vote

Company Information [Review and Analyze for Voting](#)

Startup reward offered for voting

How many days remaining

- 3) Aquí el inversor puede revisar cómo va el proceso de votación para la empresa seleccionada, con el porcentaje de votación Sí o No, la cantidad de GODZ en cada opción y el Retorno de votación esperado.
- 4) El inversor ahora puede poner la cantidad de GODZ que desea asignar en el proceso de votación, y luego seleccionar su opción (Sí o No), y luego hacer clic en el botón "Votar".
- 5) Finalmente, en esta tabla el inversionista puede revisar las recompensas disponibles, cuáles han sido las recompensas en el pasado, y puede solicitar y revisar la información sobre las compañías que terminaron sus procesos de votación, y ver fue el retorno a los inversionistas que participan en el proceso de votación.

### Voting Process

	YES	NO
<b>3</b> Voting Process	71%	29%
	<input type="text" value="1200.0000"/> <input type="text" value="GODZ"/>	<input type="text" value="500.0000"/> <input type="text" value="GODZ"/>
Expected Voting Return	79%	10%

**4** Voting amount

Vote  Yes  No

[Vote](#)

---

**5** Reward Available

Company	Voting Result	Voting Amount	Reward	Voting Return	
Sabores de la tierra	Yes	100.0000	79%	78.85	<a href="#">request</a>
Susana Parra	No	5.0000	1900%	95	<a href="#">request</a>

**6.2 Intercambio y emisión:**

- 1) Para activar esta sección del DApp, el inversor debe poner la clave privada de su billetera. Aquí, el inversor puede revisar su saldo de GODZ en el último Número de Bloque del Blockchain de Ethereum.
- 2) En esta tabla aparecerán todos los startups que tuvieron voto positivo, lo que significa que los inversionistas votaron Sí y con eso, el token ERC20 emitido por la compañía aparecerá en esta sección. En este caso tenemos Supa, que emitió 2.000 tokens a 1,5 GODZ por Token. Ha vendido 200 y tiene 1.800 disponibles para intercambiar. Si el inversor desea revisar el ERC20, puede hacer clic en el botón. Entonces, el inversionista solo necesita poner la cantidad de tokens que quiere comprar. El monto total aparecerá después de eso, y al hacer clic en "comprar", agregará este token a su billetera.

### Swap Service

**1** Your Private Key

godz Balance

Block Number

---

Company Name	Token Issued	Price per Token	Token Sold	Available	Review	Quantity to Buy	Amount
<b>2</b> Supa	2.000	1.5	200	1.800	<input type="button" value="ERC20 Info"/>	<input type="text"/>	<input type="text"/> <input type="button" value="buy"/>

- 3) Para que una empresa nueva emita tokens a través de Godzillion DApp, primero debe completar el formulario que se presenta a continuación. La compañía necesita tener una billetera Ethereum con saldo de ETH y GODZ para poder continuar con el proceso.
- 4) Una vez validado que la billetera de la compañía tiene un saldo de ETH y GODZ, la Startup puede luego completar el formulario con la información solicitada.

### 3 Issuing Service

If you are a company that need to raise capital, we invite you complete this form and let the investors decide

First, we need to unblock your company's wallet

Your Private Key

Ether Balance

godz Balance

Block Number

**4** Next, we invite you to complete this form

- 5) Aquí la compañía necesita completar la información relacionada con la Información Básica del ERC20, y la información relacionada con el monto a recaudar, el porcentaje de la compañía a emitir y la valoración asociada con ese porcentaje. Por defecto, los decimales son 0, porque la emisión del token estará relacionada con las acciones de la empresa.

### 5 Token Information

Name <input type="text"/>	Symbol <input type="text"/>
Amount <input type="text"/> GODZ	Total Supply <input type="text"/> Tokens
% of Shares to Issue <input type="text"/> %	Valuation <input type="text"/> GODZ
Token Price <input type="text"/> GODZ	Decimals <input type="text" value="0"/>

- 6) Después, el gerente debe completar la info básica de la compañía, como el sitio web, el nombre del gerente, la cuenta linkedin y el país.

**6** Basic Information

Website

Name of CEO

CEO's LinkedIn

Country

- 7) El último conjunto de información que la empresa necesita completar está relacionado con su negocio.
- 8) Finalmente, con toda la información completa, la empresa necesita poner la cantidad de GODZ que se ofrecerá como recompensa por el proceso de votación a los inversores que participarán. Después de completar el proceso de solicitud, la empresa debe hacer clic en "Iniciar proceso de votación"

**7** Company Information

Motivation  200 maximum characters

Opportunity  200 maximum characters

Clients  200 maximum characters

Problem Resolve  200 maximum characters

Technology Use  200 maximum characters

**8** Reward offered for voting  Amount of godz to offer

Final Notes

Before Continue, you need to hold 5 ETH at least in your company's wallet (beside the GODZ that you will going to use for incentive this votation) in order for all processes to run in the Ethereum Blockchain.

This 5 ETH will be hosting in the smart contract that will pay all processes.

If at the end of the whole process there is a balance remaining of ether, these will be returned to your wallet.

## 6.3 Intercambio:


### 6.3.1GODZ:

- 1) Para activar esta sección del DApp, el inversor debe poner la clave privada de su billetera.
- 2) Una vez que el inversor ponga su clave privada, estos campos mostrarán su saldo de ETH y GODZ. Si es su primera vez en Godzillion, el saldo de GODZ será 0.


- 3) Aquí el inversor puede revisar la información de la última operación de ETH para GODZ ejecutada en este Exchange.
- 4) Si el inversor quiere comprar GODZ, necesita revisar la información de la última operación, o ver el libro de órdenes, para colocar su orden de compra. El inversor debe poner el precio al que desea comprar, la cantidad y la cantidad total de ETH que debe incluir para colocar la orden de compra; luego puede hacer clic en el botón COMPRAR y, si la validación está ok, se realizará el pedido. En caso de que quiera vender GODZ, y tenga a GODZ en su billetera, debe hacer los mismos pasos, pero haciendo click en el botón de VENDER.

**godz Exchange**

1 Your Private Key  get your balance & update order book



2 Ether Balance



godz Balance

3

Last Price	Change	Bid	Ask	Volume
0.385000	0.1000%	0.45	0.42	428.875150

4

**SELL**

Price

Quantity

Total Amount

SELL

**BUY**

Price

Quantity

Total Amount

BUY

- 5) Aquí el inversor puede revisar el Libro de órdenes del Exchange, para analizar la oferta y la demanda, para poner su orden de compra o venta.
- 6) En esta sección, el inversor puede revisar las órdenes que ha puesto en el Exchange, según si quiere vender GODZ, o si quiere comprar GODZ. Él puede borrar esas órdenes si quiere, dado que aún no se han ejecutado.
- 7) En esta sección final, el inversor puede revisar las últimas 5 operaciones ejecutadas en este intercambio, con la fecha y hora, tipo de operación (una compra o venta), el precio, y la cantidad de la operación involucrada.

5 Order Book

SELL ORDERS			BUY ORDERS		
Q	Price	Sum	Q	Price	Sum
100.000000	0.450000	45.000000	200.000000	0.420000	84.000000
			8800.000000	0.385000	3472.000000

6 My Orders

SELL ORDERS		BUY ORDERS	
Q	Price	Q	Price

Trade History

7

Date	Type	Price	Quantity	Amount
7/7/2017 18:54:48	sell	0.385000	600.000000	231.000000
7/7/2017 18:49:25	sell	0.350000	500.000000	175.000000
7/7/2017 18:47:22	buy	0.150000	100.000000	15.000000
7/7/2017 18:30:17	sell	0.130000	10.578080	1.375150
7/7/2017 18:30:07	buy	0.130000	50.000000	6.500000

**6.3.2 Startups:**

- 1) Para activar esta sección del DApp, el inversor debe poner la clave privada de su billetera.
- 2) El inversor puede seleccionar de la lista el token de inicio que quiere intercambiar. Al seleccionar el token, verá la dirección pública ERC20 del token y también puede hacer clic en el botón "Revisar información del token" si se quiere saber más al respecto.
- 3) Una vez que el inversor ingresa su clave privada, la pantalla mostrará su saldo de GODZ y el token que seleccionó anteriormente. Si es su primera vez en Godzillion, el saldo de Token será 0.
- 4) Aquí el inversor puede revisar la información de la última operación de GODZ para Token (en este caso, SDLT) ejecutada en este Exchange.
- 5) Si el inversor quiere comprar Tokens de SDLT (para este ejemplo), necesita revisar la información de la última operación o la cartera de pedidos para poner en su orden de compra. El inversor necesita poner el precio al que desea comprar, y la cantidad total de GODZ que debe colocar en la orden de compra. Luego puede hacer clic en el botón Comprar y, si la validación es correcta, se realizará el pedido. En caso de que quiera vender Tokens de SDLT, y tenga Tokens en su billetera, debe hacer los mismos pasos, pero haga clic en el botón VENDER

**Startup Exchange**

1 Your Private Key  get your balance & update order book

2 Select Token SDLT Review Token Info

ERC20 Public Address: 0x8e16d682d7f96b90f631abdc26f101bce2879661

3 godz Balance your godz balance Token Balance your tokens balance

4

Last Price	Change	Bid	Ask	Volume
0.012580	0.0312%	0.01258	0.012	62.575000

5 SELL BUY

Price  Price

Quantity  Quantity

Total Amount  Total Amount

SELL BUY

- 6) Aquí el inversor puede revisar el Libro de órdenes para el token seleccionado específicamente en este intercambio, para analizar la Oferta y Demanda, para colocar su orden de Compra o Venta.
- 7) En esta sección, el inversor puede revisar el orden que ha colocado para este token, según si quiere Vender el Token seleccionado, o si quiere Comprar el Token seleccionado. Puede eliminar esas órdenes si lo desea porque aún no se han ejecutado.

6 Order Book

SELL ORDERS			BUY ORDERS		
Q	Price	Sum	Q	Price	Sum
87900	0.012580	1105.782000	1000	0.012000	12.000000
1000	0.121990	1227.772000			

7 My Orders

SELL ORDERS		BUY ORDERS	
Q	Price	Q	Price



- 8) En esta sección, el inversor puede revisar las últimas 5 transacciones ejecutadas para un token seleccionado en el exchange, con el día, hora, tipo de transacción (compra o venta), el precio y la cantidad de la transacción involucrada.

Trade History					
	Date	Type	Price	Quantity	Amount
8	7/7/2017 19:40:44	buy	0.012580	1000	12.580000
	7/7/2017 19:39:53	buy	0.012199	1000	12.199000
	7/7/2017 19:38:41	buy	0.012059	1000	12.059000
	7/7/2017 19:37:27	buy	0.011899	1000	11.899000
	7/7/2017 19:29:12	buy	0.012580	100	1.258000
	7/7/2017 19:24:10	buy	0.012580	1000	12.580000

### 6.3.3 Tokens en Ethereum:

- 1) Para activar esta sección del DApp, el inversor debe poner la clave privada de su billetera.
- 2) El inversor puede agregar tokens a este Exchange, si posee tokens de otro Exchange y quiere transferirlas aquí. Entonces, necesita poner la dirección pública del Token ERC20 que quiere agregar, y luego hacer clic en el botón "Validate ERC20". La información de ese Token ERC20 aparecerá en la parte inferior, con el nombre, símbolo, oferta total y decimal asociado. Si la información es correcta, el inversor puede hacer clic en "Agregar Token" y el Exchange permitirá que este Token ERC20 se comercialice, listándolo.

#### Tokens Exchange

**1** Your Private Key

[get your balance & update order book](#)

**2** Add Token

ERC20 Public Address

[Validate ERC20](#)

Name <input type="text" value="Golem Network Token"/>	Total Supply <input type="text" value="1000000000"/>
Symbol <input type="text" value="GNT"/>	Decimals <input type="text" value="18"/>

[Add Token](#)

- 3) Si el token que el inversor desea comercializar ya figura en el Exchange, solo tiene que seleccionarlo, y aparecerá la dirección pública del ERC20, y también el inversor puede revisar la información sobre este token haciendo click en el "Revisar la info del Token "

**3** Select Token  [Review Token Info](#)

ERC20 Public Address:

- 4) Una vez que el inversor ingresa su clave privada, la pantalla mostrará su saldo de GODZ y el token que seleccionó anteriormente. Si es su primera vez en Godzillion, el saldo de Token será 0.
- 5) Aquí el inversor puede revisar la información de la última operación en GODZ para cada Token (en este caso, GNT) ejecutada en este Exchange.
- 6) Si el inversor quiere comprar Tokens de GNT (para este ejemplo), necesita revisar la información de la última operación o el libro de órdenes, para poner su orden de compra. El inversor necesita poner el precio al que desea comprar, la cantidad y la cantidad total de GODZ que necesita para colocar la orden de compra. Luego puede hacer clic en el botón COMPRAR y, si la validación es correcta, se realizará el pedido. En caso de que quiera vender Tokens de GNT, y tenga Tokens en su billetera, debe hacer los mismos pasos, pero haciendo click en el botón VENDER.

The screenshot shows the Godzillion interface with the 'gz' logo and 'ERC 20' token icon. It features two balance boxes: 'godz Balance' with 'your godz balance' and 'Token Balance' with 'your tokens balance'. Below these is a market data table for ERC 20:

Last Price	Change	Bid	Ask	Volume
0.125000	0.0000%	0.125	0.123	25.000000

Below the market data are two order entry sections: 'SELL' and 'BUY'. Each section has input fields for 'Price', 'Quantity', and 'Total Amount', and a corresponding 'SELL' or 'BUY' button.

- 7) Aquí el inversor puede revisar el Libro de órdenes para el token seleccionado específicamente en este Exchange, para analizar la Oferta y Demanda, para colocar su orden de Compra o Venta.
- 8) En esta sección, el inversor puede revisar las órdenes que ha realizado para este token específico en este Exchange, de acuerdo con si desea Vender el Token seleccionado o si desea Comprar el Token seleccionado. También está la opción de eliminar esas órdenes si quiere.
- 9) En esta sección final, el inversor puede revisar las últimas 5 operaciones ejecutadas en este Exchange para el Token seleccionado, con la fecha y hora, tipo de operación (una compra o venta), el precio, y la cantidad del intercambio involucrado.

The screenshot shows the 'Order Book' section with two tables: 'SELL ORDERS' and 'BUY ORDERS'.

SELL ORDERS			BUY ORDERS		
Q	Price	Sum	Q	Price	Sum
49800.000000	0.125000	6225.000000	100.000000	0.123000	12.300000

Below the order book is the 'My Orders' section, which has two empty tables for 'SELL ORDERS' and 'BUY ORDERS' with columns for 'Q' and 'Price'.

At the bottom is the 'Trade History' section with a table showing the last five trades:

Date	Type	Price	Quantity	Amount
16/7/2017 14:03:35	buy	0.125000	100.000000	12.500000
15/7/2017 23:24:25	buy	0.125000	100.000000	12.500000

## 6.4 Transferencias e historia:

### 6.4.1 Transferencias:

- 1) Para activar esta sección del DApp, el inversor debe poner la clave privada de su billetera.
- 2) En esta sección, el inversor puede revisar el saldo de su billetera, de ETH y GODZ
- 3) En esta sección, el inversor puede revisar el saldo de todas las tokens que contiene su billetera en la DApp, y puede ver la cantidad que tiene, el último precio de su token en el intercambio y la valoración (en GODZ) de sus tokens.

- En esta sección final, el inversor puede transferir cualquiera de sus tokens, GODZ o ETH a cualquier otra billetera en el Blockchain de Ethereum. Seleccionando el tipo de valor que el inversionista quiere transferir, el inversionista solo necesita poner la dirección pública de la billetera a la que quiere transferir, poner el valor a transferir y luego hacer clic en el botón "transferir".

### Balance & Transfer Service

**1** Your Private Key  get your balance

---

**2** Wallet Balance

Account Balance 793.480282730567 ETH 98412.425 GODZ

**3**

Token	Symbol	Quantity	Last Price	Amount
Sabores de la tierra	SDLT	7000.000000	0.012580	88.060000
Susana Parra	SUPA	40.000000	12.258000	490.320000
Golem Network Token	GNT	436.700000	0.125000	54.587500

**4** Transfer

Your Public Address

Ethereum Block

Public Address To Transfer

Value to Transfer  ETH ▾

transfer

TxHash

Message

#### 6.4.2 Historial de intercambio:

- Para activar esta sección del DApp, el inversor debe poner la clave privada de su billetera.
- En esta sección, el inversor puede revisar toda la historia de sus intercambios de los diferentes intercambios que componen esta DApp. En este ejemplo, podemos revisar las operaciones de esta billetera, donde podemos ver la fecha y hora de la operación, el tipo de operación (si es una venta o compra), el valor involucrado y el token comercializado

### Trade History

**1** Your Private Key  get your history

---

**2**

Date	TxType	Value	Token
7/7/2017 18:54:48.736	Sell	600.000000	GODZ
7/7/2017 18:49:25.232	Sell	500.000000	GODZ
7/7/2017 18:30:17.591	Sell	10.578080	GODZ
16/7/2017 14:3:35.687	Buy	100.000000	Golem Network Token
15/7/2017 23:24:25.853	Buy	100.000000	Golem Network Token
7/7/2017 19:40:44.587	Buy	1000.000000	Sabores de la tierra
7/7/2017 19:39:53.85	Buy	1000.000000	Sabores de la tierra
7/7/2017 19:38:41.668	Buy	1000.000000	Sabores de la tierra
7/7/2017 19:37:27.691	Buy	1000.000000	Sabores de la tierra
7/7/2017 19:29:12.83	Buy	100.000000	Sabores de la tierra
7/7/2017 19:24:10.850	Buy	1000.000000	Sabores de la tierra

**7. Token Crowdfunding Campaign y el uso de los ingresos.**

Se crearán 300,000,000 de GODZ (Tokens ERC20) en Godzillion's Token Crowd Sale (TCS). La gente podrá comprar 210,000,000 de GODZ con ETH. El proceso se ejecutará con un contrato inteligente de intercambio, recibiendo ETH y entregando GODZ al remitente de ETH. Los compradores enviarán ETH al Contrato inteligente swap de Godzillion para recibir GODZ. Al hacerlo, recibirán Godzillion Tokens (GODZ) a razón de 160 GODZ por 1 ETH. Los detalles están aquí abajo:

	Sw ap rate
GODZ for ETH	160
ETH for GODZ	0,00625

Un participante debe enviar Ether a la cuenta del contrato inteligente después del comienzo del período de crowdfunding (especificado como número de bloque). El Crowdfunding finaliza cuando se crea el bloque final, o cuando la cantidad de ETH enviado a la cuenta alcance el máximo. La dirección de crowdfunding se anunciará en el inicio del crowdfunding a través de los siguientes canales oficiales:

- Página web del proyecto: [www.godzillion.io](http://www.godzillion.io)
- Twitter oficial: [https://twitter.com/godzillion\\_io](https://twitter.com/godzillion_io)

**Por favor, verifique dos veces la dirección antes de enviar ETH. Por razones de seguridad, recomendamos confirmar la dirección usando al menos dos fuentes diferentes. En la página web del proyecto, también encontrará una guía detallada sobre cómo participar en el crowdfunding utilizando Ethereum Wallet.**

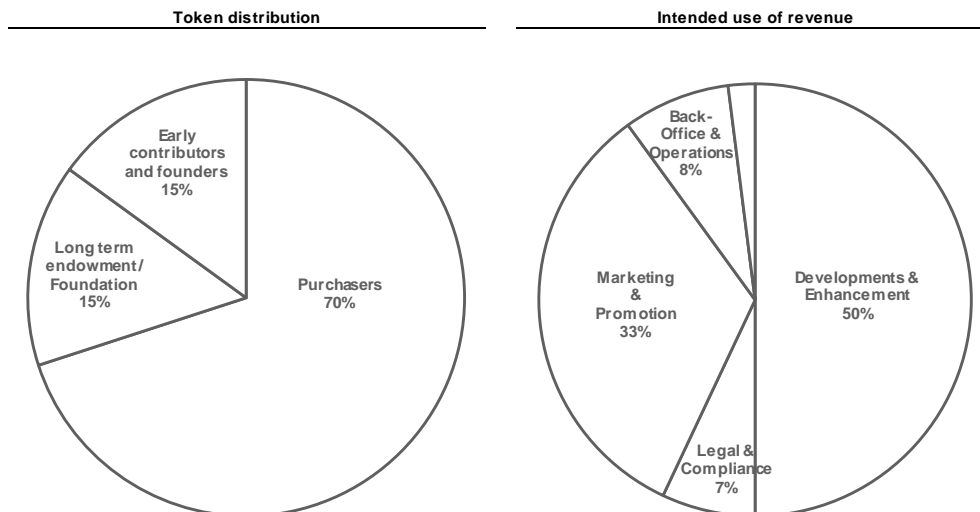
Crowd Sale se implementa como un contrato inteligente con unos pocos parámetros simples:

- Godzillion controla el contrato y la dirección a la que se enviará el ETH reunido.
- El porcentaje de tokens preasignados es del 30% (15% - Godzillion Team and Founders, 15% - Godzillion Foundation).
- StartBlock, EndBlock: estos números de bloque indican el inicio y el final del proceso de crowdfunding.

Otro grupo de 45,000,000 GODZ se destinará directamente a los primeros contribuyentes que hayan trabajado para desarrollar las ideas, implementaciones y estructuras de soporte del proyecto Godzillion hasta el momento del Crowd Sale. El tamaño de este grupo es 15% de la cantidad total de GODZ. Un segundo grupo de 45,000,000 GODZ se asignará directamente a la Fundación Godzillion para que el pago de los gastos futuros se determine a su exclusivo criterio. El tamaño de este último grupo es el 15% de la cantidad total de GODZ.

Entonces, la emisión total de GODZ (300,000,000) se distribuye de la siguiente manera: 70% para Compradores, 15% a largo plazo para la fundación Godzillion y 15% para los Colaboradores iniciales, fundadores y el equipo. Como se explica en la hoja de ruta anterior, el uso previsto de los ingresos se distribuye de la siguiente manera: 50% a Desarrollo y Mejora, 8% a Back-office y Operaciones, 33% a Marketing y Promoción, 7% a Legal y Cumplimiento y 2% a Varios y otros.

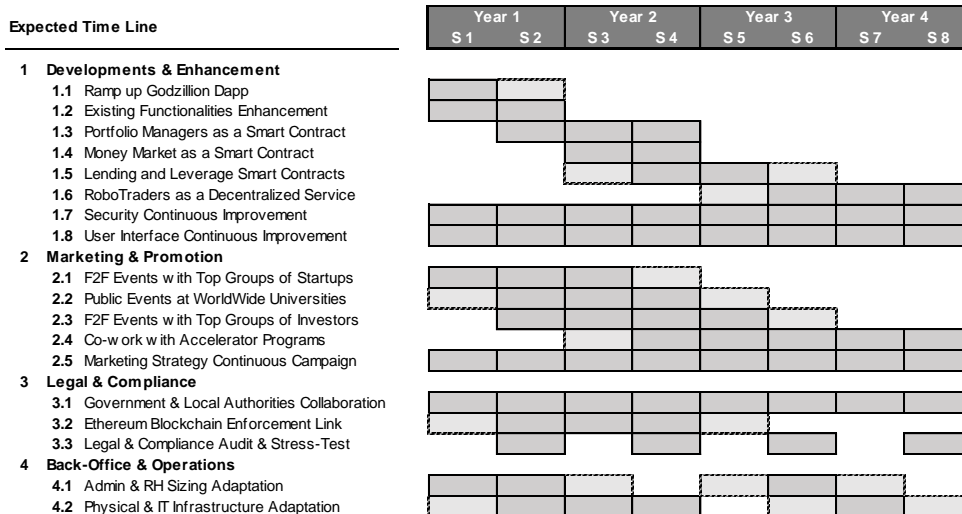
Los detalles están aquí abajo:



### 8. Hoja de Ruta.

Godzillion DApp ya está en funcionamiento y ha facilitado el financiamiento, a través de la emisión de tokens, de cuatro empresas chilenas de reciente creación. La negociación de esos tokens comenzó inmediatamente después de que se completaran los procedimientos de subasta y emisión. Para alcanzar los hitos del futuro, se ha establecido una hoja de ruta. El plan abarca cuatro ramas principales de prioridades para implementación durante el horizonte de planificación de cuatro años. Las actividades y mejoras prioritarias incluyen actividades específicas para lograr tareas de desarrollo y mejora, marketing y promoción, y tareas e hitos legales y de cumplimiento.

- **Desarrollo y mejora:** Este primer año comenzará con la aceleración de las operaciones de Godzillion DApp y la mejora de las funcionalidades que ya funcionan como la alerta de ejecución frontal. En la segunda mitad del año, una tarea importante es desarrollar un contrato inteligente que sea capaz de implementar decisiones de administración de cartera, con ajustes y controles bajo la supervisión de un gerente humano o con funciones automatizadas. En el segundo año, una tarea clave es desarrollar un contrato inteligente que pueda administrar la funcionalidad de un mercado monetario, conectando a los prestamistas a corto plazo con opciones de liquidez con tokens como garantía. El tercer año, un objetivo importante es construir un contrato inteligente que pueda implementar una función de lógica de préstamo, que supervisará e implementará un mercado de bonos para individuos y empresas nuevas. A medida que el equipo aprenda de la implementación de las mejoras y características anteriores, el desarrollo proyectado para el cuarto año es un RoboTrader descentralizado con configuraciones que pueden ser controladas por Inversores y un segundo tipo de RoboTrader con un rol de creador de mercado específico. Un tema constante durante este horizonte de cuatro años será la mejora en cuestiones de seguridad como una prioridad general, y mejoras en la experiencia de la interfaz de usuario.
- **Marketing y promoción:** Nuestro alcance a las Startups aumentará, de modo que la Startup podrá lanzar más emisiones de tokens vinculadas a acciones, utilizando la aplicación descentralizada Godzillion. Tan pronto como finalice el ICO de GODZ, los inversores mantendrán GODZ dentro de su cartera y estarán esperando que las Startups publiquen sus propuestas para que sean votadas, y los inversores estarán esperando para ofertar en las subastas en las que las Startups busquen emitir tokens. En los segundos seis meses se ampliará el alcance a las Universidades y sus Incubadoras, con el fin de propagar el uso de Godzillion DApp como un catalizador que reúne a los votantes y los startups potenciales exitosas. Y, a medida que las Startups publiquen más propuestas, obtengan más financiamiento y aumenten las transacciones de mercado secundario, Los eventos públicos ayudarán a los Grupos de Inversores a explicar el rol de GODZ, el proceso de votación y emisión, y el acceso a un mercado secundario basado en Blockchain. Después de alcanzar esos hitos iniciales, en el segundo año el enfoque involucrará la colaboración con un programa de aceleración para mejorar los lanzamientos de información y los procesos de formación de startups.
- **Legal y cumplimiento:** Con el fin de implementar el plan descrito de una manera legalmente correcta, la colaboración con diferentes oficinas de reguladores en diferentes países será un esfuerzo continuo. Con la ayuda de los abogados locales en cada país, se espera que Godzillion DApp opere en todas las facetas del espectro de crowdfunding, con reguladores y una comunidad de Crowdfunding también. Durante los segundos seis meses, con la ayuda de Abogados, el Equipo implementará una solución para vincular el proceso de emisión notarial con la emisión de registros en Blockchain. Como informe financiero continuo y política de Cumplimiento, el Proyecto Godzillion será auditado anualmente, y los informes del auditor serán publicados de acuerdo con los procedimientos estándar.
- **Back-Office y operaciones:** Después de la Crowd Sale, el equipo planea comenzar a reclutar personas calificadas para mantener la capacidad de implementación técnica y humana. Los nuevos empleados estarán principalmente en las áreas de Desarrollo y Marketing. El segundo semestre también se planea reubicar la Sede de Santiago a Londres para acercar el Proyecto al mercado europeo y brindar servicio a los muchos países que se encuentran cerca de ese centro financiero. Las actividades detalladas que están programadas para ponerse en funcionamiento durante el lapso de tiempo se ilustran en el diagrama de línea de tiempo a continuación



## 9. Historia Corporativa.

Mi futurofinanciero.com SpA es un startup cofundada entre Rodrigo Sainz y Cristóbal Pereira, ambos emprendedores chilenos. La misión de la compañía es acercar las finanzas a las personas, creando productos y servicios Fintech para que las personas logren su proceso de creación de riqueza. Desde sus primeras sesiones visionarias, su objetivo ha sido el cómo automatizar los procesos en el sector financiero para que las personas puedan ingresar al mercado financiero de manera eficiente, sin demasiados intermediarios en el proceso.

El primer problema que trataron de superar fue la fricción que las personas enfrentan cuando buscan acceder al mercado del ahorro. Para reducir esa fricción, en 2013 la compañía desarrolló una plataforma web que permite a las personas revisar diferentes tipos de productos financieros (desde ingresos fijos hasta activos alternativos) en un solo sitio. Después de lanzar la plataforma y atraer usuarios a ella durante un año, se dieron cuenta de que existe una segunda barrera para que las personas tengan acceso a la bolsa de valores o incluso al mercado de bonos. Necesitaban tener un patrimonio inicial de al menos US \$ 1,000 para abrir una cuenta con la empresa de corretaje bursátil para tener acceso al mercado e implementar órdenes.

Para resolver esta fricción, en 2015, desarrollaron una aplicación que ayuda a las personas a redondear sus gastos diarios, por lo que la decisión de gastar o ahorrar ya no era necesaria. Con esta aplicación, las personas gastan y ahorran al mismo tiempo, por lo que les resulta fácil canalizar sus pequeñas cantidades de ahorro en fondos del mercado monetario y acumular 1.000 dólares estadounidenses en un corto período de tiempo.

A fines de 2015, los cofundadores vieron el potencial de la tecnología Blockchain para reducir la fricción en los procesos financieros, la descentralización de las transferencias y el mantenimiento de registros. Después de que el Proyecto Ethereum y la lógica de los contratos inteligentes se lanzaron, comienzan a darse cuenta del enorme potencial que blockchain posee para el comercio y las finanzas corporativas. En el momento en que el equipo estaba aprendiendo acerca de esta tecnología, inmediatamente vieron su idoneidad y comenzaron a programar su primer contrato inteligente en Solidity y desplegarlo en el Ethereum Blockchain.

En el primer trimestre de 2016 codificaron su primer contrato inteligente para gestionar la emisión de bonos para empresas y particulares, con un mercado primario y un mercado secundario, ambos privados y vinculados entre sí. Después de aprender de esa experiencia, los fundadores decidieron girar 100% a la tecnología blockchain, dejando atrás la lógica centralizada, y comenzaron a construir una plataforma totalmente descentralizada como una arquitectura de contratos inteligentes en el Blockchain de Ethereum, el resultado es Godzillion.

El equipo de Godzillion trabaja en conjunto desde 2013 y son:

- **Cristobal Pereira**, cofundador y CEO, tiene una Maestría en Administración y un título de Ingeniería de la UDD, Chile.
- **Rodrigo Sainz**, cofundador y CIO, tiene un Master en Finanzas y un título de Ingeniería de la UDD, Chile.
- **Eduardo Portugues**, CTO, posee un grado en Ingeniería Computacional de DUOC, Chile.
- **Matias Pereira**, Ingeniero de Software, tiene una Maestría en Administración e Ingeniería de la UDD, Chile.

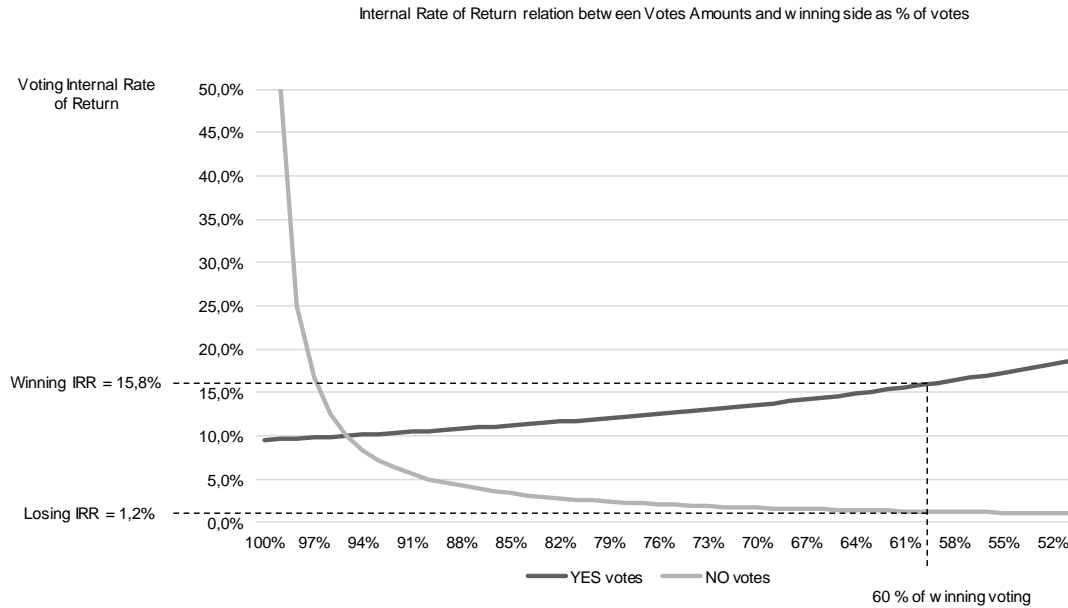
En la actualidad, el objetivo de la empresa es crear productos y servicios diseñados y programados como plataformas descentralizadas que funcionan en blockchain. Mifuturofinanciero.com SpA ha recibido informes de prensa y cobertura de radio y televisión favorables debido a la innovación en el sector financiero chileno para plataformas como mimolido y Godzillion, las dos plataformas que operan en este momento.

Los accionistas y asesores del Consejo son miembros prominentes de la comunidad financiera chilena e internacional, con títulos académicos, certificaciones profesionales y amplia experiencia en gestión de línea en intermediarios financieros chilenos tradicionales. Los asesores de la Junta son:

- **John C. Edmunds**, tiene una **D.B.A.** en Negocios Internacionales de la Escuela de Negocios de Harvard, una Maestría en Finanzas y Métodos Cuantitativos con honores de la Universidad de Boston, una Maestría en Economía de la Universidad Northeastern, y una maestría A.B. en Economía cum laude de Harvard.
- **José L. Ruiz**, es doctor en Ciencias Administrativas y Economía Aplicada por la Wharton Business School de la Universidad de Pensilvania, y tiene una Maestría en Finanzas y una Maestría en Economía, ambas de la Universidad de Chile.
- **David Diaz**, doctor en Business Intelligence, University of Manchester Business School, y Master en Finanzas de la Universidad de Chile.
- **Roberto Darrigrandi**, tiene un doctorado en Administración de Empresas de la Universidad de Liverpool y un MBA de Booth School of Business, Universidad de Chicago.

10. Apéndice.

10.1 Estructura de recompensa de votar, tasa interna de rendimiento:



10.2 Estructura de Costos, necesidad de Gas:

The fee schedule G is a tuple of 31 scalar values corresponding to the relative costs, in gas, of a number of abstract operations that a transaction may effect.

Name	Value	Description
Gzero	-	Nothing paid for operations of the set Wzero.
Gjumpdest	1	Paid for a JUMPDEST operation.
Gbase	2	Amount of gas to pay for operations of the set Wbase.
Gverylow	3	Amount of gas to pay for operations of the set Wverylow .
Gmemory	3	Paid for every additional word when expanding memory.
Gcopy	3	Partial payment for *COPY operations, multiplied by words copied, rounded up.
Gtxdatazero	4	Paid for every zero byte of data or code for a transaction.
Glow	5	Amount of gas to pay for operations of the set Wlow .
Gsha3word	6	Paid for each word (rounded up) for input data to a SHA3 operation.
Gmid	8	Amount of gas to pay for operations of the set Wmid.
Glogdata	8	Paid for each byte in a LOG operation's data.
Ghigh	10	Amount of gas to pay for operations of the set Whigh.
Gexp	10	Partial payment for an EXP operation.
Gexpbyte	10	Partial payment when multiplied by dlog256(exponent)e for the EXP operation.
Gblockhash	20	Payment for BLOCKHASH operation.
Gsha3	30	Paid for each SHA3 operation.
Gtxdatanonzero	68	Paid for every non-zero byte of data or code for a transaction.
Gsload	200	Paid for a SLOAD operation.
Gcodedeposit	200	Paid per byte for a CREATE operation to succeed in placing code into state.
Glog	375	Partial payment for a LOG operation.
Glogtopic	375	Paid for each topic of a LOG operation.
Gbalance	400	Amount of gas to pay for a BALANCE operation.
Gextcode	700	Amount of gas to pay for operations of the set Wextcode.
Gcall	700	Paid for a CALL operation.
Gcallstipend	2.300	A stipend for the called contract subtracted from Gcallvalue for a non-zero value transfer.
Greset	5.000	Paid for an SSTORE operation when the storage value's zeroness remains unchanged or is set to zero.
Gsuicide	5.000	Amount of gas to pay for a SUICIDE operation.
Gcallvalue	9.000	Paid for a non-zero value transfer as part of the CALL operation.
Rsclear	15.000	Refund given (added into refund counter) when the storage value is set to zero from non-zero.
Gsset	20.000	Paid for an SSTORE operation when the storage value is set to non-zero from zero.
Gtransaction	21.000	Paid for every transaction.
Rsuicide	24.000	Refund given (added into refund counter) for suiciding an account.
Gnew account	25.000	Paid for a CALL or SUICIDE operation which creates an account.
Gcreate	32.000	Paid for a CREATE operation.
Gtxcreate	32.000	Paid by all contract-creating transactions after the Homestead transition.