

---

This is the **published version** of the bachelor thesis:

Ferrando Ruiz, Joel; Pérez Martínez, Andreu, dir. Transformación de moneda de crédito a criptomoneda. 2022. (958 Enginyeria Informàtica)

---

This version is available at <https://ddd.uab.cat/record/264163>

under the terms of the  license

# Transformación de moneda de crédito a criptomoneda

Joel Ferrando Ruiz

**Resumen—** El mundo de los eSports sigue en crecimiento y en paralelo la tecnología Blockchain sigue revolucionando muchas áreas y explorando nuevas posibilidades. El sector de los videojuegos ya comienza a experimentar algunas funcionalidades de esta nueva tecnología, aprovechando lo mejor de ambas para satisfacer las necesidades de los jugadores. Con esta finalidad, se pretende aprovechar un sistema de organización de eventos eSports ya creado y transformarlo en un nuevo producto, empleando las nuevas posibilidades que nos puede ofrecer la Blockchain y las criptomonedas. En este proyecto podremos conocer el desarrollo de inicio a final de todas las partes necesarias para llevar a cabo una transformación de moneda de crédito a criptomoneda y beneficiarse de este cambio para, a partir de un problema real de poco volumen de usuarios en una aplicación web, solucionar el problema creando un nuevo sistema innovador a través de unir conceptos de eSports y Blockchain.

**Palabras clave—** deportes-electrónicos blockchain videojuegos token BSC cartera play-to-earn business-plan ethereum contrato-inteligente

**Abstract—** The world of eSports continues growing and in parallel Blockchain technology continues revolutionizing many areas and exploring new possibilities. The video game sector is already beginning to experiment with some of the functionalities of this new technology, taking advantage of the best of both to meet the needs of players. With this purpose, we intend to take advantage of an eSports event organization system already created and transform it into a new product, using the new technologies that Blockchain and cryptocurrencies can offer us. In this project we will know the development from start to finish of all the necessary parts to carry out a transformation from credit currency to cryptocurrency and benefit from this change to, from a real problem of low volume of users in a web application, solve the problem and create a new innovative system combining concepts of eSports and Blockchain.

**Keywords—** eSports blockchain videogames token BSC wallet play-to-earn business-plan ethereum smart-contract

## 1 INTRODUCCIÓN

**L**OS deportes electrónicos siguen en expansión gracias a las nuevas franquicias de videojuegos, la expansión de los eSports en móvil y al gran crecimiento en los mercados emergentes. Con estas nuevas características se espera que la audiencia de los eSports siga creciendo al +8,7 % anual [1].

Por otra parte, las empresas de Blockchain y criptomone-

das, recaudaron un récord de 30 mil millones de dólares de fondos de capital riesgo. Aún con las caídas recientes del mercado de criptomonedas, los fondos de inversión siguen apostando e invirtiendo en todo tipo de proyectos relacionados con Web 3.0, NFTs, Fan tokens... [2]

Juntando estos dos sectores, cada vez más videojuegos apuestan por crear NFTs donde los usuarios puedan venderlos entre los demás jugadores. Además de recompensar con tokens el tiempo que se destina jugando a sus videojuegos, originando que los jugadores pasen más tiempo en el ecosistema y tengan una valoración más positiva.

Se quiere explorar una vía de unir estos dos mundos y crear un ecosistema que abarque lo mejor de ambos para solucionar un problema real.

- E-mail de contacto: joelferrando27@gmail.com
- Mención realizada: Enginyeria de Computadors
- Trabajo tutorizado por: Andreu Pérez
- Curso 2021/22

## 1.1. Foozle y su problema a resolver

Foozle es una startup y aplicación web de eventos de eSports que permite crear, organizar y jugar a tus videojuegos favoritos y ganar recompensas. [3]

Fue creada en 2020 por 3 socios estudiantes de Ingeniería Informática. La aplicación cuenta con una trayectoria de un año en el mercado y dos de desarrollo.

El mayor problema a resolver se trata del poco volumen de jugadores y visitas en nuestra web. Basándonos en un modelo de negocio típico de este tipo de aplicaciones, que se basa en un modelo de publicidad personalizada, este poco volumen ocasiona que sea poco atractivo para las marcas el hecho de querer publicitarse en nuestra web. Además de contar con el hándicap de ser una startup con poco recorrido y experiencia en el mercado.

La mayoría de los jugadores buscan un premio atractivo, un evento de prestigio o de máxima competición y que sus marcas o influencers favoritos patrocinen ese mismo evento. Al depender de los ingresos de las marcas y no poder organizar esos tipos de eventos, conseguir un gran volumen de personas es inviable actualmente.

## 1.2. Propuesta de valor

Para resolver el problema actual y diferenciarnos de la competencia, se propone la creación de un token para dar valor a nuestro producto y poder:

- Unir la gente de videojuegos con la gente del mundo cripto, ampliando la cuota de mercado a la que nos dirigimos.
- Obtener ingresos de otras vías distintas a los anuncios personalizados como:
  - Tasas en las transacciones convencionales y Marketplace
  - Venta de productos exclusivos en nuestro Marketplace
- Conseguir un ecosistema play-to-earn para cualquier videojuego del mercado.

También se espera que al ser la primera aplicación de organización de eventos de eSports que recompense a los jugadores mediante tokens por jugar, independientemente de la posición, a cualquier videojuego y no a un juego específico con un sistema play-to-earn, se cree un gran interés tanto de jugadores, empresas, influencers, inversores...

## 2 ESTADO DEL ARTE

Aunque el auge de los eSports es muy reciente, la organización de torneos de videojuegos ya existe desde hace mucho tiempo. Con este gran crecimiento, muchas plataformas han decidido organizar este tipo de eventos y hacerse un hueco en el mercado, pudiendo encontrar diferentes plataformas organizadoras de eventos de eSports que se quieren diferenciar en diferentes ámbitos, ya sea desde procesos automáticos de torneos implementando APIs externas que facilitan los 'publishers' de videojuegos, a la posibilidad de crear torneos personalizados como usuario, hasta ser el mayor organizador de eventos de un videojuego en concreto.

Sabemos que todas ellas tienen en común lo mismo, se basan en un modelo de negocio de anuncios y aprovechan toda esa gran masa de jugadores para producir beneficios. También conocemos que trabajan con moneda real, como puede ser el euro, o con una moneda de crédito que solo es utilizable en su propia plataforma.

El paso que aún ninguna empresa de organización de eSports, si nos referimos a las que tengan una trayectoria amplia y un lugar definido en el mercado, no ha dado aún, es la implementación de pagos con criptomonedas a los jugadores de su plataforma.

Por otra parte, hasta hace pocos años los videojuegos no recompensaban a sus usuarios de forma monetaria, hasta que aparecieron los juegos play-to-earn, donde los desarrolladores del videojuego pagan a los usuarios que jueguen a sus juegos. Los requisitos de este pago vienen dados por ellos y pueden ser desde jugar un número determinado de partidas a quedar en las mejores posiciones de su ranking. El pago suele ser realizado en NFTs o tokens.

El juego de referencia actual es Axie Infinity que tuvo un pico de 2.7 millones de jugadores activos [4]. Con el gran éxito de Axie, no han tardado en aparecer nuevas variantes de videojuegos con este nuevo sistema al mercado, donde los jugadores concuerdan en que quieren ser beneficiados de alguna forma por pasar tanto tiempo jugando al videojuego. Sin embargo, otra gran parte de los usuarios coincide en otro factor: Los videojuegos play-to-earn actuales no son divertidos, aún así jugamos porque nos pagan con tokens.

## 3 OBJETIVOS

Se propone dividir el proyecto en dos ámbitos diferentes: el ámbito técnico y el empresarial. En el ámbito técnico se investigarán y analizarán los procesos necesarios para la creación de un token cripto en una blockchain existente e implementar un sistema de carteras en nuestra aplicación donde se puedan realizar transacciones.

Por parte del ámbito empresarial, se buscará crear un ecosistema donde el token tenga sentido y pueda realmente ser utilizado y después poder mostrar toda esa información a agentes externos interesados en el proyecto.

Los objetivos definidos son:

- Creación del token
- Sistema de carteras
- Integración Token - Cartera
- Funcionamiento de las transacciones del token
- Business Plan con simulación de flujo de tokens
- Whitepaper

## 4 METODOLOGÍA

Durante todo el proyecto se ha trabajado con una metodología SCRUM [6]. Esta metodología consistía en definir 'sprints' de dos semanas donde se realizaba una reunión entre alumno y profesor para mostrar las tareas definidas en la reunión anterior y plantear las nuevas tareas para la siguiente reunión. Por parte de alumno, también se ha llevado un control de:

	Ethereum	BSC
Consenso	PoS(Eth 2.0)	PoSA
Marca	Ninguna marca	Binance
Centralización	- centralizado	+ centralizado
Salida funcional	2015	2020
Costo Trans.	+ costosas	- baratas
Vel. Trans.	+ lentas	+ rápidas
DApps	+3000	+800

TABLA 1: DIFERENCIAS ENTRE ETHEREUM Y BSC

- Definición y descripción de las tareas realizadas
- Control de horas de las tareas desarrolladas
- Bibliografía de las fuentes utilizadas
- Documentación de las tareas y procedimientos

Esta estrategia nos ha permitido mantener un trabajo constante, pudiendo variar en los objetivos planteados al inicio del proyecto, así como en las tareas a realizar durante la realización de este.

## 5 ÁMBITO TÉCNICO

### 5.1. ¿Qué necesitamos?

Antes de iniciar con la creación de la criptomoneda, cabe conocer que funcionalidad queremos darle a nuestro token. A continuación se muestran los motivos por los cuales se ha decidido crear un token dentro de una blockchain ya existente:

- Necesidad de utilizar contratos inteligentes para poder crear un contrato entre usuarios y Foozle donde, por ejemplo, los premios de los torneos sean entregados automáticamente y cualquier usuario pueda visibilizar que los pagos se han realizado correctamente.
- Ahorro importante en el tiempo de desarrollo al no tener que crear una Blockchain desde cero.
- Seguridad que aporta a la plataforma una Blockchain ya existente y probada en el mercado.

Una vez tenemos claros que funcionalidades queremos que desarrolle nuestro token, se ha decidido por dos blockchains que aportan todas nuestras necesidades. En este caso se trata de BSC y ETH.

### 5.2. Ethereum o BSC

Para elegir una de estas dos opciones discutiremos sobre los conceptos que hemos creido más importantes. Podemos revisar algunos de los puntos en la tabla 1.

#### 5.2.1. Centralización

En PoS de Ethereum, los validadores harán staking con ETH para tener la oportunidad de producir bloques. Otros validadores revisarán el bloque y comprobarán que sea correcto. Si alguien produce un bloque que contiene transacciones falsas, correrá el riesgo de perder todas sus monedas en staking. Luego, los validadores reciben recompensas

por los bloques exitosos y por cualquier certificación que realicen.[7]

En PoA, los creadores de bloques se conocen como validadores. En otras palabras, los validadores son preaprobados y seleccionados por Binance, como indica en su documentación. Para ser aceptados, deben verificar que son personas reales e invertir dinero para demostrar su compromiso a largo plazo. En este modelo, Binance tiene el control total de la blockchain. Además, ellos deciden quién se convierte en validador, y también eliminan validadores a su discreción. Sobre todo, esto requiere que los usuarios confíen en que Binance se comportará en su mejor interés. Asimismo, si Binance decide alterar cualquier característica de la cadena o del ecosistema, tiene el poder de hacerlo.

Esto puede ser útil en caso de que se llegue al caso de que se comprometa la integridad de nuestra aplicación y se produzca una salida de tokens no intencionada. En ese caso, Binance puede revertir las transacciones de los tokens hacia nuestra aplicación de nuevo y retornar los activos a los afectados.

#### 5.2.2. Transacciones

El problema actual de ETH es que debido al gran crecimiento que ha tenido no lo ha podido soportar, causando una congestión en su red y provocando transacciones más costosas y lentas. Con la actualización de Eth 2.0, Ethereum promete transacciones más baratas y rápidas pero aún sigue en desarrollo.

Si nos fijamos en las transacciones diarias, hay una diferencia aún mayor entre las dos. En BSC, es más rápido y rentable para los usuarios mover sus fondos e interactuar con los smart contracts. Podemos ver en la figura (1) que el pico de BSC es de alrededor de 16 millones de transacciones diarias y en su estado actual es de más de cuatro millones. Por otro lado, Ethereum nunca ha superado los 1,75

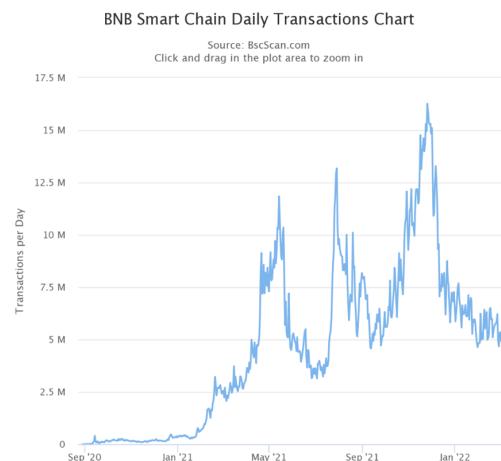


Fig. 1: Transacciones Diarias en la BSC

millones de transacciones diarias (2). Para los usuarios que necesitan mover sus fondos con regularidad, BSC parece ser la opción más popular.

### 5.3. Tokens y billeteras

Las aplicaciones descentralizadas y los tokens que funcionan en la Binance Smart Chain se ajustan a la máquina

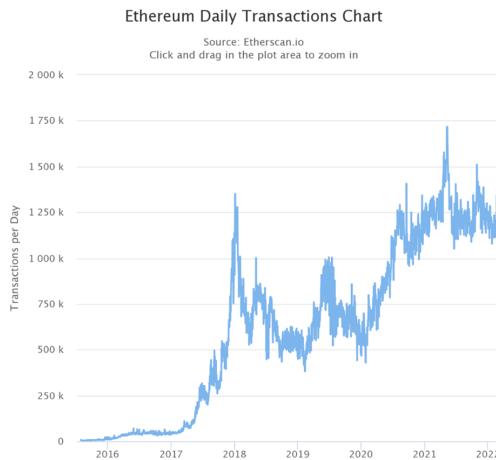


Fig. 2: Transacciones Diarias en Ethereum

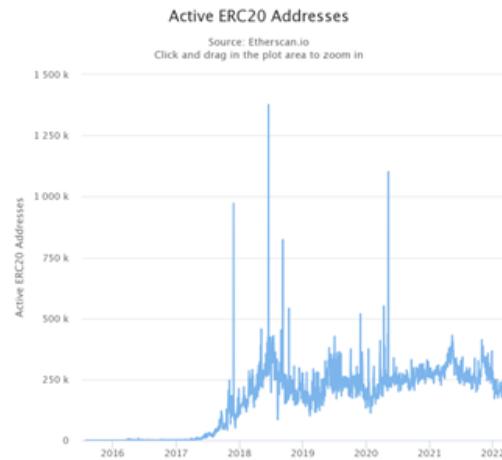


Fig. 4: Carteras ERC20 Activas

virtual de Ethereum (EVM). No solo eso, sino que las direcciones de los monederos públicos son las mismas en ambas blockchains. Al mismo tiempo, hay incluso proyectos de cross-chain que funcionan en ambas redes. [7]

Las transacciones diarias también deben verse en el contexto de direcciones activas. Actualmente, BSC tiene un mayor número de usuarios que realizan más transacciones en promedio. En las figuras (3) y (4) podemos observar el número de carteras activas. En la BSC se ha logrado un pico de 5,5 millones de carteras activas con un promedio de 1,5 millones actualmente, mientras que en las carteras de Ethereum el máximo ha sido de 1,4 millones con un promedio actual de 250 mil carteras activas.



Fig. 3: Carteras BEP-20 Activas

### 5.3.1. Apps

Ethereum es la blockchain donde existen los juegos más populares de blockchain, aunque algunos proyectos de BSC son bastante parecidos a los juegos de moda play-to-earn como CryptoKitties y Axie Infinity, no han conseguido una audiencia tan grande como los juegos clásicos de Ethereum. Aun así, cada vez se están creando más proyectos en la BSC relacionados con Blockchain y Binance está apostando por los más prometedores.

### 5.3.2. Elección

Ambas plataformas son muy parecidas y las diferencias no son tan grandes como para decantarnos por una u otra. Al final, se ha definido BSC ya que ofrece transacciones más rápidas y rentables para los usuarios y esto aporta una ventaja a los usuarios cuando quieran mover sus fondos y poder interactuar con smart contracts. Además, analizando como crear un token en las distintas Blockchains, existe más información y de más calidad en la forma de cómo crear un token en la BSC, cuestión que influye en la facilidad de construcción del proyecto. Asimismo, está la posibilidad de aplicar en el Binance Labs y Binance Launchpad donde apoyan proyectos cripto relacionadas con su propia blockchain.

## 5.4. Creación del token

Para la creación de un token deberemos desarrollar un contrato inteligente y desplegarlo. Para desplegarlo se ha elegido la herramienta Remix. Remix es un entorno de desarrollo de Ethereum que nos ofrece la posibilidad de compilar, desplegar y administrar smart contracts. En el código del smart contract definiremos algunos de los parámetros más básicos pero necesarios como:

- **totalSupply:** Cantidad de monedas que existen actualmente y que están en circulación o bloquedas de alguna manera.
- **name:** Nombre que tendrá nuestra criptomonedra.
- **symbol:** Siglas de nuestra criptomonedra.

Además, deberemos desarrollar las diferentes funciones que queremos utilizar con nuestro token:

- **allowance:** Nos permite dar permisos a la cuenta designada para transferir cierta cantidad de tokens.
- **transfer:** Transfiere tokens desde la cartera oficial a la cartera designada.
- **transferFrom:** Transfiere desde una cartera emisora hacia otra cartera receptora la cantidad de tokens deseados.

- **getBalanceOf:** Obtiene la cantidad de tokens de la cartera que se le facilita.

El Apéndice 1 nos muestra que estructura tendría el smart contract desarrollado para crear el token. Una vez el smart contract esté finalizado, este podrá ser compilado en la BSC. Para poder comprobar todo tipo de transacciones y gozar de BNB de forma gratuita, la infraestructura ha sido realizada en la BSC de test. También finalizamos transfiriéndole todos los tokens a una wallet donde mantendremos y gestionaremos todos los tokens como entidad (5).

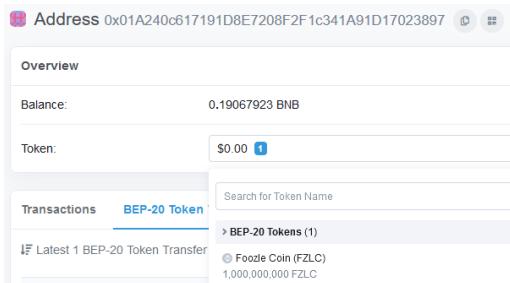


Fig. 5: Cartera principal con todos los FZLC

## 5.5. Sistema de carteras

Se han propuesto diferentes tipos de integración de wallets a desarrollar en nuestra web. A continuación, se explicarán que ventajas y desventajas se han tenido en consideración para elegir una ellas:

1. **Wallets y tokens ficticios:** Esta opción asignaría una wallet ficticia a cada usuario donde los tokens dados también serían ficticios. Las retiradas de tokens de los usuarios se gestionarían desde una wallet general gestionada por nosotros. La ventaja consiste en tener un sistema más centralizado y tener que desarrollar menos código para el correcto funcionamiento. El problema que hace INVIABLE esta opción reside en el caso en que se desee añadir la funcionalidad de que los usuarios quieran enviar a las carteras de Foozle sus propios tokens para poder comprar en el marketplace. Una vez envíen sus tokens a nuestra dirección (nuestra wallet general), no podremos saber de quién es cada transacción debido a que no conocemos a quien pertenece la cartera emisora y, por consiguiente, no podremos “devolverle” los tokens a su cartera de Foozle.
2. **Metamask:** La opción de integrar nuestro sistema con Metamask es viable y se considera una buena opción. Existe la posibilidad de enviar y recibir tokens en nuestra web de modo que los usuarios puedan comprar productos en nuestra tienda. Además, se cuenta con la seguridad de contar con un equipo que lleva años de experiencia en el mundo Blockchain y en consecuencia hace nuestro sistema más seguro. El problema existe en que se busca la mayor facilidad para los usuarios. Esto nos deja en una situación de tener que crear tutoriales y existir una complejidad añadida al tener que instalar Metamask y gestionarlo como usuario. Desde Foozle buscamos que los usuarios tengan las mínimas dificultades posibles y la interacción entre transacciones sea la mínima posible.

3. **Wallets nativas:** En esta opción se crea una cartera en la Blockchain y se asigna a la cuenta Foozle del usuario una vez se registra. Con esto ganamos que pueda recibir tokens añadiendo solo la dirección pública desde el lugar desde donde se desea enviar y que el usuario solo tenga que copiar una clave. La desventaja radica en el mayor intervalo de tiempo de desarrollo de código para el funcionamiento interno de las transacciones y realizar un sistema más seguro.

Analizando las ventajas y desventajas de las tres opciones, se ha optado por realizar la tercera opción de wallets nativas para el sistema de wallets y transacciones del token. Esto también nos servirá para conocer como funcionan los diferentes tipos de Blockchain y como podemos trabajar con ellas y añadir nuevas criptomonedas o tokens como métodos de pago.

## 5.6. Integración

Para la integración de carteras y transacciones con BNB y nuestro token en la BSC se ha optado por web3js. Web3js es una colección de bibliotecas de JavaScript, que te permiten interactuar con un nodo de Ethereum local o remoto mediante HTTP, IPC o WebSocket [8].

En Ethereum, los nodos proporcionan interfaces de bajo nivel para que los usuarios envíen transacciones. Las transacciones pueden ser recibidas por un nodo a través de una interfaz JSON RPC.

La interfaz JSON RPC no sólo se ocupa de las transacciones, sino también de otras interacciones, como la obtención de información sobre el estado de la red. Los usuarios o las aplicaciones pueden enviar invocaciones JSON RPC directamente a un nodo, formando la estructura de datos JSON necesaria y enviándola a la interfaz expuesta.

Ya que BSC y Ethereum son compatibles y comparten muchas de las mismas funcionalidades podremos trabajar de la misma forma.

### 5.6.1. Base de datos

En la tabla de Usuarios necesitaremos guardar dos nuevos campos:

- Clave pública: Identificación de la cuenta
- Clave privada: Autenticación del usuario

Estos dos campos son necesarios para poder identificar a cada usuario y que puedan transaccionar tokens dentro de nuestra aplicación junto al depósito y retirada de tokens. También crearemos la Tabla Transacciones para tener un control de hacia que carteras y que cantidad de tokens se envian fuera de nuestra aplicación, almacenando:

- Dirección emisor
- Dirección receptor
- Cantidad

### 5.6.2. Funciones

A continuación, se hace una breve introducción de las funciones que se están usando actualmente para poder interactuar con la blockchain de Binance y poder realizar transacciones con tokens en nuestra aplicación web:

- **create:** Función necesaria para poder crear una wallet en la BSC. Nos devuelve un objeto que contiene la clave pública y privada.
- **signTransactions:** Función necesaria para firmar las transacciones y poder enviar tokens de un usuario a otro usuario.
- **getBalance:** Obtiene el balance de cada cuenta de nuestros usuarios para poder mostrarle por pantalla de cuantos tokens dispone.

En el apéndice A.2 podemos observar la estructura de las funciones utilizadas y para más información podemos revisar la documentación oficial de web3js [8].

## 6 ÁMBITO EMPRESARIAL

### 6.1. Business Plan

El business Plan se ha creado con la intención de crear un plan para ver el futuro del proyecto, donde nos guiará en las decisiones que tomemos y nos proporcionará una mayor claridad sobre los distintos ámbitos que tendremos que tener en cuenta. En este, se describirán todos los detalles necesarios para poder seguir adelante con el proyecto, definiendo desde nuestra estrategia comercial hasta donde queremos llegar en un futuro.

Complementariamente, se ha añadido un documento de simulación donde podemos encontrar los diferentes módulos que se encontrarán en la aplicación y como van a trabajar. La descripción de estos y su flujo de tokens nos permitirá ajustar los parámetros para encontrar un sistema equilibrado. Además, se añadirán estos conceptos en el business plan para añadir las entradas y salidas de tokens o capital.

El business plan sigue una previsión de cuatro años vista (hasta 2025). Se crea inicialmente definiendo los usuarios activos esperados y la previsión de crecimiento mensual en nuestra aplicación. Seguidamente, podemos encontrar todas las variables que aportarán ingresos:

- **Ventas marketplace:** Ventas de todo tipo de producto por parte de Foozle.
- **Anuncios:** Fuente principal de los ingresos. El modelo de negocio se basa principalmente en la venta de posiciones estratégicas de anuncios a marcas que deseen publicitarse.
- **Comisiones:** Todo tipo de transacción en la aplicación tiene una comisión, y esta se destina al mantenimiento del ecosistema del token.

Posteriormente, se añaden los gastos variables:

- **Retiradas:** Los usuarios que deseen retirar activos desde Foozle a una plataforma externa.
- **Productos:** Compra de productos que luego se venderán en el marketplace.
- **Marketing:** Coste de atraer nuevos clientes y mantenerlos en la aplicación. Hay que tener en cuenta de donde proviene el usuario. Conseguir clientes europeos o estadounidenses cuestan más que usuarios latinoamericanos.

Después, sumamos los gastos fijos donde se incluye desde sueldos de los trabajadores hasta lo necesario para mantener la aplicación: servidores, licencias, oficinas...

Una vez todos los campos han sido indicados, observamos en la figura (6) que empezamos a obtener beneficios en el tercer año y el punto de equilibrio del proyecto se cumple entre el segundo y tercer año. Necesitando de una financiación en los compases iniciales de la vida del proyecto para poder ejecutar todas las estrategias planteadas por la empresa.

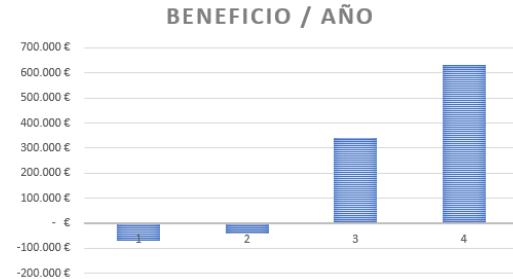


Fig. 6: Beneficios

## 7 RESULTADOS

Una vez la integración ha sido realizada, podemos observar los resultados de las transacciones. Empezaremos siguiendo el flujo que seguiría la aplicación. Una vez un jugador se registre mediante el formulario de Registro, se le asignará una dirección pública y una clave privada (7).

public_key	private_key
0x3FC0504ff1e55a167a150fd6acf6fe371319b75d	0x505390d2d79fecc9387b0e977db098617342ab90c...
0x1549f371569585326538c7a35a0469767355	0x197d5a6b304f33b1e09c25f50a45902a4f1d434eb...
0xC8Ac92304AB0f40905a45f15d114438a3fB62	0x40dec09455da394c6395280d418f347fe78709ae...
0x25878b40742035107888214460A495a1ec989	0x928488b5c698aacf77e988deeb248fb6bb7d530a3...

Fig. 7: Asignación de clave pública y privada en la BD.

Una vez registrado podemos comprobar en la web BscScan como la cartera se acaba de crear y no dispone de balance.

### 7.1. Pago de torneo

Para comprobar el pago de nuestra cartera principal, que gestiona todos los tokens de la aplicación, hacia el ganador de un torneo, simularemos que este se ha realizado con éxito, donde el premio era de 0.01 BNBs y 50 mil FZLC.

Mediante las funciones descritas en los módulos anteriores podemos ver en la figura (8) y (9) como se ejecutan ambas transacciones con el emisor y receptor definidos. El control de estas transacciones se puede revisar en la BSC de test. Una vez han sido confirmadas, el ganador ya dispone de sus tokens (10).

From:	0x1a240c617191d8e7208f2f1c341a91d17023897
To:	0x3fc084ff1e53a167a150fd6acf6fe371319b75d
Value:	0.01 BNB (\$0.00)

Fig. 8: Transacción de BNB de la cartera principal a la cartera del usuario ganador.



Fig. 9: Transacción de FZLC de la cartera principal a la cartera del usuario ganador.

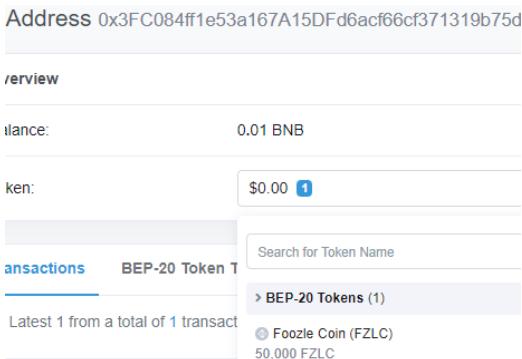


Fig. 10: Balance de la cartera del usuario ganador

## 7.2. Transacción de usuario a usuario

Otro de los objetivos del proyecto era poder realizar transacciones entre dos usuarios independientes.

En el módulo de enviar criptomonedas podemos observar nuestro balance actual, podemos introducir a quien queremos enviar los tokens y que cantidad. Tendremos que tener en cuenta que en caso de querer enviar tokens, deberemos tener un pequeño porcentaje de BNB para poder pagar el gas de la transacción (11).

### General

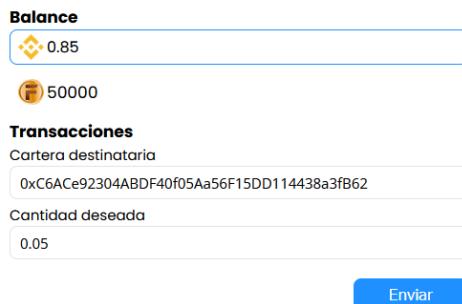


Fig. 11: Módulo para enviar tokens entre usuarios

Una vez ejecutemos la transacción observamos a través de las figuras (12), (13) y (14) como el usuario envía a otro usuario una cantidad de 0.05 BNB .



Fig. 12: Transacción de 0.05 BNBs

## 8 CONCLUSIONES

Una vez finalizado el proyecto, podemos sacar varias conclusiones:

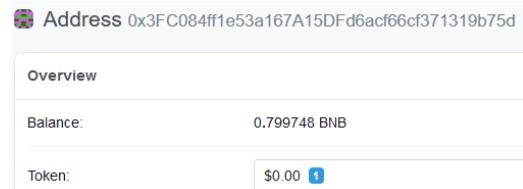


Fig. 13: Cartera del emisor con el balance de BNB actualizado.

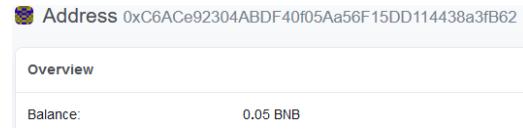


Fig. 14: Cartera del receptor con el balance de BNB actualizado.

- Debido a un problema inicial de falta de usuarios en nuestra aplicación, se han analizado y propuesto varias soluciones para abordar el problema. Gracias a esto, se ha implementado un sistema cripto para solucionar el problema.
- Analizando el Business Plan, podemos concluir que el proyecto es rentable en el futuro, pero cabe destacar varios problemas que hay que solucionar:
  - El uso de algunos videojuegos en eventos de eSports requiere de una licencia. Esta licencia se obtiene por la empresa desarrolladora de ese videojuego y suelen venir con restricciones en la cantidad máxima de premios, cantidad máxima de torneos que puedes realizar por mes...
  - Varias empresas de videojuegos bloquean el uso de su videojuego en proyectos relacionados con Blockchain y criptomonedas.
- Se ha aprendido a analizar que requisitos son necesarios para crear un token y elegir una Blockchain acorde a estos. También, como interaccionan los tokens y su propia Blockchain y por último, a como integrar todos los conceptos anteriores para poder realizar transacciones entre usuarios.
- Creemos que el mercado aún no está preparado para aceptar este proyecto. Aun así, estamos seguros de que dentro de poco tiempo, con la aceptación de proyectos Blockchain y cripto conjuntamente con proyectos gaming, nuestra idea de plataforma podrá ser lanzada.
- Debido a nuestra concepción del futuro sobre nuestro proyecto, todos los conceptos aprendidos durante la realización de este trabajo nos permitirán seguir trabajando en futuras implementaciones para seguir mejorando nuestra propuesta de valor.

## 9 FUTURAS IMPLEMENTACIONES

El proyecto aún es muy amplio y contempla un abanico muy amplio de posibles implementaciones. Como próximas implementaciones para hacer aún más viable el proyecto se contempla lo siguiente:

- Posibilidad de trabajar con empresas de videojuegos que quieran integrar un sistema de pago con criptomonedas en su juego uniendo ambas plataformas.
- Acuerdo con empresas pequeñas y medianas de videojuegos que busquen conseguir nuevos jugadores a cambio de la licencia de uso de su videojuego para poder organizar eventos.
- Búsqueda de financiación para poder costear los costes del proyecto y conseguir el crecimiento descrito en el Business Plan.

## AGRADECIMIENTOS

Agradecer a mis padres por el continuo apoyo durante la carrera universitaria. A mi tutor Andreu por su ayuda, seguimiento y visión sobre el proyecto. También agradecer a David y Adrián, socios de Foozle, por poder debatir sobre las funcionalidades del token y como enfocar de la forma más realista el proyecto. Por último, a Rafa, Borja y Arturo, interesados en el proyecto, que nos han dado su visión desde una perspectiva más amplia y real de mercado y saber de todos los posibles problemas que se desconocían sobre el proyecto.

## REFERENCIAS

- [1] Global Esports Live Streaming Market Report. <https://newzoo.com/products/reports/global-esports-live-streaming-market-report>
- [2] Morgan Stanley Says Record Crypto Venture Capital Investment to Slow. <https://www.coindesk.com/business/2022/05/31/morgan-stanley-says-record-crypto-venture-capital-investment-to-slow/>
- [3] Foozle - Crea, organiza y juega torneos. <https://foozle.gg/inicio>
- [4] Axie Infinity Was Losing Gamers Even Before Record Crypto Hack. <https://www.bloomberg.com/news/articles/2022-04-04/axie-infinity-was-losing-gamers-even-before-record-crypto-hack>
- [5] Remix - Ethereum IDE community. <https://remix-project.org/>
- [6] Professional Scrum Master™ II — Scrum.org. <https://www.scrum.org/resources/what-is-scrum>
- [7] Diferencias entre Ethereum y Binance Smart Chain <https://academy.binance.com/es/articles/binance-smart-chain-vs-ethereum-what-s-the-difference>
- [8] Web3.js - Ethereum JavaScript API. <https://web3js.readthedocs.io/en/v1.7.3/>
- [9] Binance Chain Docs. <https://docs.binance.org/smart-chain/wallet.html>
- [10] BscScan. <https://testnet.bscscan.com/>

## GLOSARIO

- Cadena de bloques / (Blockchain)** Libro compartido e immutable que facilita el proceso de registro de transacciones y de seguimiento de activos en una red de negocios
- Cartera / (Wallet)** Una cartera es un dispositivo o programa utilizado para la transferencia y el almacenamiento de criptomonedas
- Clave privada / (Private Key)** Clave necesaria para firmar digitalmente las transacciones. No debe compartirse nunca y debe guardarse siempre de forma segura. Si se accede a la clave privada, los atacantes pueden retirar todos los tokens de la cartera. Si se pierde la clave privada de una cartera, todos los tokens que se hayan enviado o almacenado en esa dirección se pierden permanentemente
- Clave pública / (Public Key)** Dirección del receptor. La dirección de la cartera puede ser compartida y mostrada libremente. Para enviar una transacción se necesita saber esta dirección
- Contratos inteligentes / (Smart contracts)** Programas almacenados en una cadena de bloques que se ejecutan cuando se cumplen unas condiciones pre-determinadas. Suelen utilizarse para automatizar la ejecución de un acuerdo sin que intervenga ningún intermediario
- Deportes electrónicos / (eSports)** Competiciones de videojuegos que se han convertido en eventos de gran popularidad. Por lo general, son competiciones de videojuegos multijugador entre jugadores profesionales
- JSON-RPC** JSON RPC es un formato de codificación textual que permite a los procesos en ejecución recibir datos
- Metamask** Software de criptomonedas que es instalado como extensión de un navegador web. Permite a los usuarios almacenar y administrar claves, transmitir transacciones, enviar y recibir criptomonedas y tokens
- NFT** Tipo de token criptográfico que representa un activo único. Funcionan como pruebas verificables de autenticidad y propiedad dentro de una red Blockchain
- play-to-earn** Juegos que permiten a los usuarios ganar criptomonedas y NFTs que pueden venderse en el mercado
- Tienda / (Marketplace)** Lugar donde los usuarios pueden comprar, intercambiar y vender productos gaming, NFTs, tarjetas digitales, etc
- Token** Activo digital que se ejecuta sobre la cadena de bloques de otra criptomonedra
- Web 3.0** Se describe como la posible siguiente versión de Internet basada en Blockchain donde los datos se interconectarán de manera descentralizada, a diferencia de Web 2.0 donde los datos se almacenan principalmente en servidores centralizados

**Whitepaper** Documento que define el proyecto, explicando cuál es el problema que se quiere resolver, cómo se resolverá, la financiación y la cantidad de token que se dispondrán. Además de una hoja de ruta de la implementación del proyecto, entre otros detalles destacables

## SIGLAS

**BSC** Binance Smart Chain

**ETH** Ethereum

**PoA** Proof of Authority

**PoS** Proof of Stake

## APÉNDICE

### A.1. Smart Contract

```

uint public totalSupply = 1000000000 * 10 ** 18;
string public name = "Foolz Coin";
string public symbol = "FZLC";
uint public decimals = 18;

function balanceOf(address owner) public view returns(uint) {
    return balances[owner];
}

function transferFrom(address from, address to, uint value) public returns(bool) {
    require(balanceOf(from) >= value, 'balance too low');
    require(allowance[from][msg.sender] >= value, 'allowance too low');
    balances[to] += value;
    balances[from] -= value;
    emit Transfer(from, to, value);
    return true;
}

```

### A.2. Funciones web3

```

web3.eth.accounts.create();
> {
    address: "0xb8CE9ab6943e0eCED004cDe8e3bBed6568B2Fa01",
    privateKey: "0x348ce564d427a3311b6536bbcff9390d69395b06ed6c486954e971d960fe8709",
    signTransaction: function(tx){...},
    sign: function(data){...},
    encrypt: function(password){...}
}

web3.eth.getBalance("0x407d73d8a49eeb85d32cf465507dd71d507100c1")
.then(console.log);

web3.eth.accounts.signTransaction({
    to: '0xF0109fC8DF283027b6285cc889F5aA624EaC1F55',
    value: '1000000000',
    gas: 2000000
}, '0x4c0883a69102937d6231471b5dbb6204fe5129617082792ae468d01a3f362318')
.then(console.log);

```