UNIVERSIDAD DE ANTIOQUIA

Desarrollo de un sistema basado en blockchain para la plataforma Witcash

Autor(es)

Carlos Andrés García Montoya
Esteban Restrepo Restrepo

Universidad de Antioquia

Facultad de Ingeniería, Departamento de Ingeniería de Sistemas

Medellín, Colombia

2021

Desarrollo de un sistema basado en blockchain para la plataforma Witcash.

Carlos Andrés García Montoya
Esteban Restrepo Restrepo

Tesis o trabajo de investigación presentada(o) como requisito parcial para optar al título de:

Ingeniero de Sistemas

Asesores (a):

Diego José Luis Votía Valderrama, Doctor (phD)

Héctor Orlando Sanchez, Magister (MSc)

Universidad de Antioquia

Facultad de ingeniería, Departamento de ingeniería de sistemas.

Medellín, Colombia

2021.

TÍTULO

Desarrollo de un sistema basado en blockchain para la plataforma Witcash.

Resumen

Al momento de iniciar el proyecto, encontramos un sistema en producción con una arquitectura e infraestructura poco eficientes, que presentaba fallos y lentitud al usuario final. Dado esto abordamos el reto de analizar, desarrollar y desplegar un sistema financiero basado en bitcoin. Para esto tuvimos sesiones de análisis y diseño en las que se replantearon los requisitos del negocio, la arquitectura del sistema y la infraestructura a utilizar para soportar el sistema. Posteriormente iniciamos el desarrollo backend de la lógica del negocio, en la que se crearon los modelos de base de datos y se expusieron los servicios requeridos por el front para dar usabilidad a la aplicación. Una vez concluida la fase de lógica del negocio, continuamos desarrollando, configurando y desplegando los componentes de comunicación con la blockchain, que permitieron al sistema tener una comunicación directa con la red de bitcoin para la creación y seguimiento de cuentas, consulta de balances, obtención y envío de transacciones, etc. En paralelo a esto configuramos ambientes de **test** y **producción** en servidores CentOS 8 alojados en Google Cloud; para el servidor de test se implementó integración y despliegue continuo haciendo uso de la herramienta que Gitlab ofrece para ello.

Introducción

Blockchain es una de las tecnologías emergentes de la cuarta revolución industrial [1], la cual tiene como objetivo principal almacenar información de manera secuencial y distribuida, con el fin de obtener un sistema que no necesite de intermediarios o terceros de confianza que garanticen la integridad de la información.
Podemos entender Blockchain como una red de computadores con una base de datos distribuida, donde todos los participantes tienen un registro sincronizado de la información. Es decir, pasamos de un modelo centralizado, donde un ente controla la información, a uno descentralizado, donde la información se distribuye entre participantes de la red.
Cada blockchain, define un método de consenso, que debe ser cumplido por cualquier miembro con intenciones de añadir información, garantizando que no se puedan monopolizar los contenidos. Por medio de dicho consenso, los integrantes determinan cuál es el estado de la base de datos donde van a converger y descartan posibles alteraciones por actores malintencionados.
En general, su potencial e importancia se debe en gran parte a las ventajas de utilizar una base de datos distribuida, que en algunos casos puede entenderse como un libro de contabilidad compartido, con un gran número de actores interesados en mantener su integridad.

Esta tecnología cada vez toma más fuerza, por ejemplo en los mercados financieros. Donde la actividad se basa en la realización de transferencias, movimientos de fondos y existe un gran interés por la seguridad y privacidad de la información, así como por mantener un registro de los movimientos que sea accesible y confiable para los participantes. Es aquí donde la tecnología blockchain puede aportar gran valor al negocio, donde no solo elimina

las ineficiencias inherentes a los sistemas actuales de pago [3], sino que los mejora, reduciendo riesgos, fraudes y costos transaccionales; también aumenta en gran medida la redundancia de la información [4].

Bitcoin[2] es el primer caso de uso real de una blockchain, la primera criptomoneda y sin duda una solución financiera que marcó un antes y un después en el mundo transaccional, sin embargo en un inicio no era fácil para una persona sin conocimientos técnicos interactuar con la red, a partir de esto surgieron múltiples soluciones denominadas billeteras, que permiten a las personas recibir y enviar bitcoin de manera fácil e intuitiva.
Witcash, es una aplicación móvil, que permite la transferencia, custodia, y monetización de Bitcoin, además de gestión de contactos, manejo y administración de múltiples llaves públicas y privadas, entre otros.

Actualmente el backend, la base de datos y la infraestructura del proyecto presentan grandes problemas de diseño y eficiencia. El backend, inicialmente desarrollado en el lenguaje de programación .NET fue diseñado con muy malas prácticas de desarrollo de software, la base de datos SQL, alojada en un servidor privado sobre la infraestructura de google cloud, presenta grandes problemas en el modelamiento y normalización de datos, y la infraestructura se encuentra desplegada en un servidor Windows poco eficiente, con malos diseños como el almacenamiento de logs en la herramienta bloc de notas, entre otros.

Debido a esto surge la necesidad de construir nuevamente todo el sistema descrito anteriormente, con el fin de reducir los errores presentados en producción, mejorar los tiempos de respuesta de la aplicación, aumentar la seguridad y brindar una mejor experiencia a los usuarios.

Objetivos

General:

- Implementar el backend, la base de datos y la infraestructura que permita establecer conexión con la blockchain de bitcoin para la plataforma Witcash.

Específicos:

- Desplegar la infraestructura en un servidor CentOS 8 y alojarlo en Google Cloud.
- Diseñar y desarrollar el backend usando las tecnologías NodeJs y Express mediante el framework NestJs.
- Diseñar el modelo de base de datos tipo NoSQL y desplegarlo en un cluster de MongoDB .
- Hacer uso de la herramienta para CI/CD para integración y despliegue continuo que ofrece Gitlab.
- Desplegar un componente de NBXplorer y sincronizarlo de manera segura con un nodo de Bitcoin Core.

Marco Teórico

Las transferencias de dinero se vieron revolucionadas con la llegada del internet y con ello la facilidad para realizar pagos a grandes distancias. En un principio, los pagos digitales dependían casi exclusivamente de terceros de confianza como las instituciones financieras, que a pesar de ser un sistema de pagos efectivo, está sujeto a la intervención de entes reguladores como bancos y gobiernos ocasionando un incremento en el costo transaccional debido al costo de las mediaciones. Este problema fue abordado por Satoshi Nakamoto en su artículo Bitcoin: un sistema de dinero electrónico peer-to-peer [5], en el que propone un sistema financiero basado en criptografía en lugar de confianza, con grandes bondades como la capacidad de realizar transacciones seguras sin necesidad de intermediarios, bajas comisiones para grandes transferencias, etc. Dicha propuesta dio lugar al desarrollo del primer caso de éxito de la tecnología blockchain.

Blockchain es básicamente un sistema con el cual se pueden hacer transacciones seguras entre personas en todo el mundo sin necesidad de intermediarios. Se trata de una gran base de datos en la que muchos nodos guardan una copia de la información. Además, blockchain basa la certificación de la información en el consenso, es decir, si todos los participantes de la red tienen la misma información, significa que esa información es verdadera.

A diferencia de los sistemas tradicionales, blockchain está basado en infraestructuras distribuidas y descentralizadas (Figura 1.0.0), lo que le permite ganar ciertas ventajas, como lo son:

- Prescindir de una entidad centralizada que pudiera manipular, alterar o censurar la información almacenada.
- Todos los nodos son independientes, lo que asegura que la caída de cualquiera de estos no causará pérdidas de información ni inconvenientes en la comunicación de los demás nodos. Esto le proporciona alta disponibilidad y redundancia al sistema.

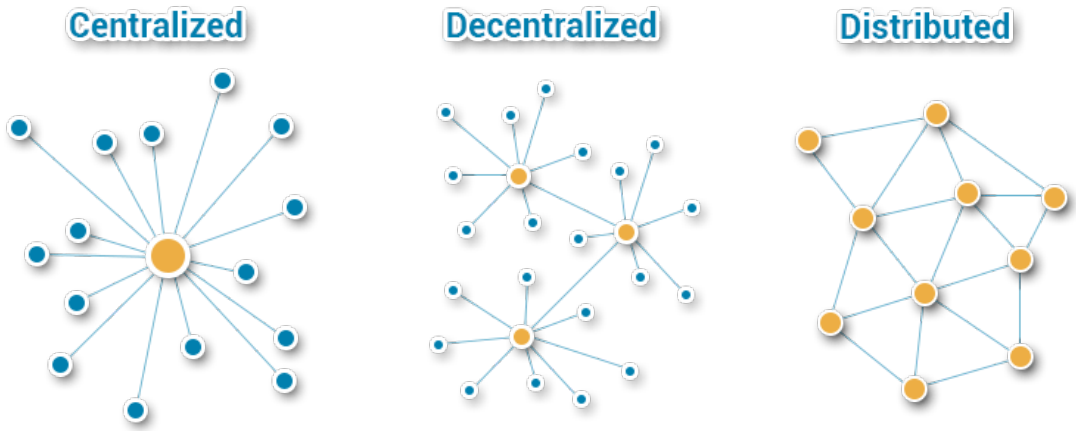

Figura 1.0.0 Fuente: https://bienestarmutuo.org/centralizado-descentralizado-distribuido-lo-horizontal-a-vertical/

El nombre de “Blockchain” hace referencia a que la información se almacena en cadenas de bloques, cada nodo contiene exactamente la misma cadena de bloques que los otros nodos. En Bitcoin, una transacción se refiere a una transferencia de valor, y los bloques están compuestos por un conjunto de dichas transacciones, además de otros metadatos importantes (Figura 2.0) como un identificador único, una estampa de tiempo, el hash[8] del bloque anterior, el nonce, entre otros.

El identificador de cada bloque se genera pasando toda la información del bloque incluyendo el resultado del hash[8] del bloque anterior y los demás metadatos por una función hash[8], logrando obtener un identificador único que contiene la información del bloque correspondiente y de su bloque anterior, formando de esta manera una cadena de bloques que no se puede alterar, ya que cualquier modificación en la información de alguno de los bloques no coincidiría con el hash del bloque en cuestión y tampoco con sus sucesores rompiendo de esta manera la cadena.

En la figura 2.0.0 podemos observar cómo se relacionan los bloques por medio de sus hash correspondientes y el de sus antecesores.

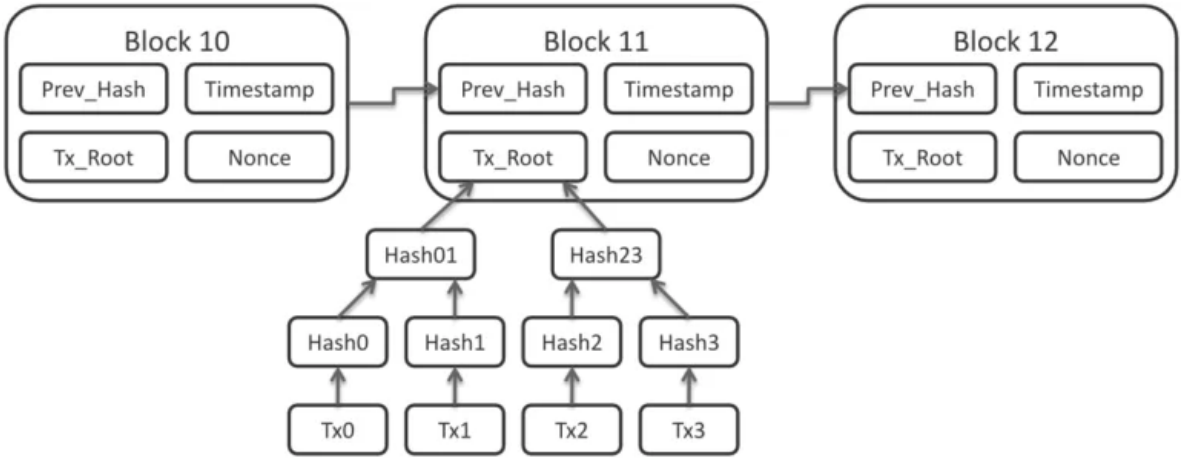

Figura 2.0.0 Fuente:
https://www.digikey.com/es/articles/build-security-into-blockchain-applications-part-1

En Bitcoin, el proceso en el que un nuevo bloque es agregado a la cadena de bloques se conoce como minería, y se da con un protocolo de consenso entre los nodos llamado prueba de trabajo, que consiste en un problema aleatorio, al cual los nodos tienen que encontrarle una solución buscando un número que resuelva dicha aleatoriedad, para esto requieren hacer uso de un poder computacional muy alto y así poder competir contra otros mineros. Una vez se obtiene la solución, el nodo emite su respuesta al resto de la red para su validación. En caso de que sea correcta, el bloque pasa a formar parte de la cadena y el nodo es recompensado con un monto de bitcoins.

L a criptografía es una parte esencial de todas las plataformas basadas en blockchain, y se utiliza para garantizar la integridad de los mensajes y/o transacciones creados en el

protocolo. Específicamente en Bitcoin, la creación de billeteras y la firma de transacciones, son unos de los procesos más importantes, y dependen en gran medida de la criptografía de clave pública (Figura 3.0). El protocolo de Bitcoin utiliza un algoritmo de firma digital llamado ECDSA para crear un nuevo conjunto de claves privada y pública. La clave pública se utiliza para verificar la autenticidad de una transacción, y también se usa con una función hash para crear la dirección pública que los usuarios de Bitcoin usan para enviar y recibir fondos. La clave privada se debe mantener en secreto, ya que se utiliza en el firmado de una transacción digital para asegurar que el origen de la transacción sea legítimo. Este sistema de claves ayuda por lo tanto a garantizar la autenticidad e integridad de las transacciones confiando en técnicas criptográficas avanzadas como se mencionó anteriormente. La principal función de una billetera de bitcoin, es administrar las claves ECDSA de sus usuarios, permitiéndoles el envío y recepción de bitcoin de una manera fácil e intuitiva.

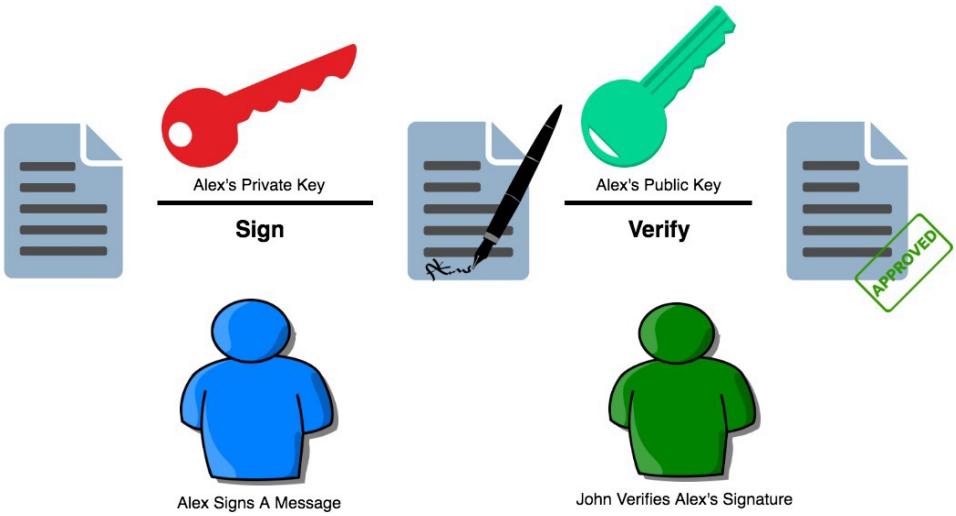

Figura 3.0 Fuente:
https://www.mycryptopedia.com/public-key-private-key-explained/

Todos los sistemas de información poseen vulnerabilidades que pueden llegar a ser muy significativas para la integridad del sistema. Bitcoin por su parte, está sujeto a un ataque llamado “Ataque del 51%”, y aunque para la robustez de la red (Figura 4.0) que se ha logrado construir a lo largo del tiempo puede ser muy complejo ejecutarlo, se puede producir en el momento en que una entidad pueda controlar el 51% del poder computacional de la red, es decir, dispondría de más de la de la mitad de la capacidad de procesamiento de la red, y a su vez, más participantes para las “votaciones” del consenso que el resto.

Una vez una entidad haya logrado controlar más del 51% de la red, podría llegar a utilizar esa ventaja para realizar operaciones, tales como:

- Revertir transacciones y producir un doble gasto de Bitcoins
- Evitar las confirmaciones de transacciones que deberían validarse de forma normal
- Evitar que los demás mineros puedan minar bloques válidos

Como se puede apreciar en la imagen de la figura 4.0.0, actualmente el mayor pool de minería de bitcoins posee un 18.7% de poder computacional con respecto al resto de la red, lo cual hace que la complejidad del Ataque del 51% sea muy elevada.

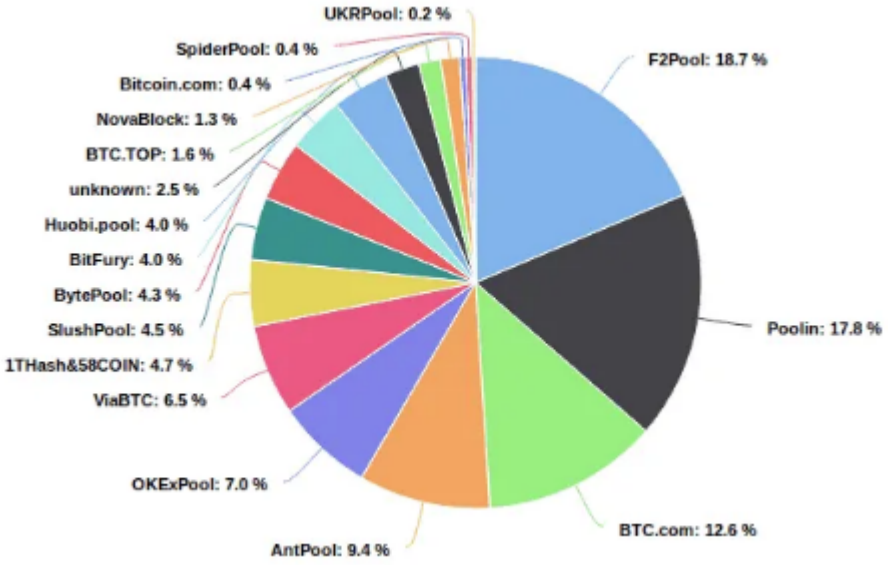

Figura 4.0.0 Fuente: https://www.blockchain.com/es/pools

Metodología

El desarrollo del proyecto se abordará en ciclos de tiempo con entregas de software funcional, compuestos por planificación, análisis de requisitos, diseño, codificación y pruebas. Para cumplir con este enfoque se planea trabajar bajo el marco de trabajo SCRUM que permite continuamente mejorar el producto, el equipo y el entorno de trabajo gracias a sus Roles, Eventos, Artefactos y Reglas asociadas [6]. Para llevar a cabo dicho marco de trabajo de manera exitosa, se debe enmarcar cada ciclo de tiempo dentro de un sprint, que es es un período de tiempo, para este proyecto de dos semanas, durante el cual se crea un incremento de producto "Terminado" utilizable y potencialmente desplegable. Durante cada sprint se deben llevar a cabo ciertas ceremonias. Inicialmente se lleva a cabo el sprint planning, reunión en la que se planifica el trabajo a realizar durante el sprint, por medio de historias de usuario que deben ser puntuadas por el equipo de desarrollo. Una vez comenzado el Sprint se debe cumplir con la daily, reunión diaria de 15 minutos máximo en la que el equipo de desarrollo planea el trabajo a realizar en las siguientes 24 horas, también se expone el resultado de las 24 horas previas y se notifica al equipo dificultades o impedimentos para la ejecución de tareas. Una vez finalizado el tiempo asignado para cada Sprint, se debe realizar una ceremonia llamada Review, la cual consta de una revisión de las funcionalidades desarrolladas en el Sprint finalizado. Por último se realiza una retrospectiva en la cual los miembros

En al figura 3.0.0 se observa un diagrama ilustrativo de la metodología anteriormente descrita.

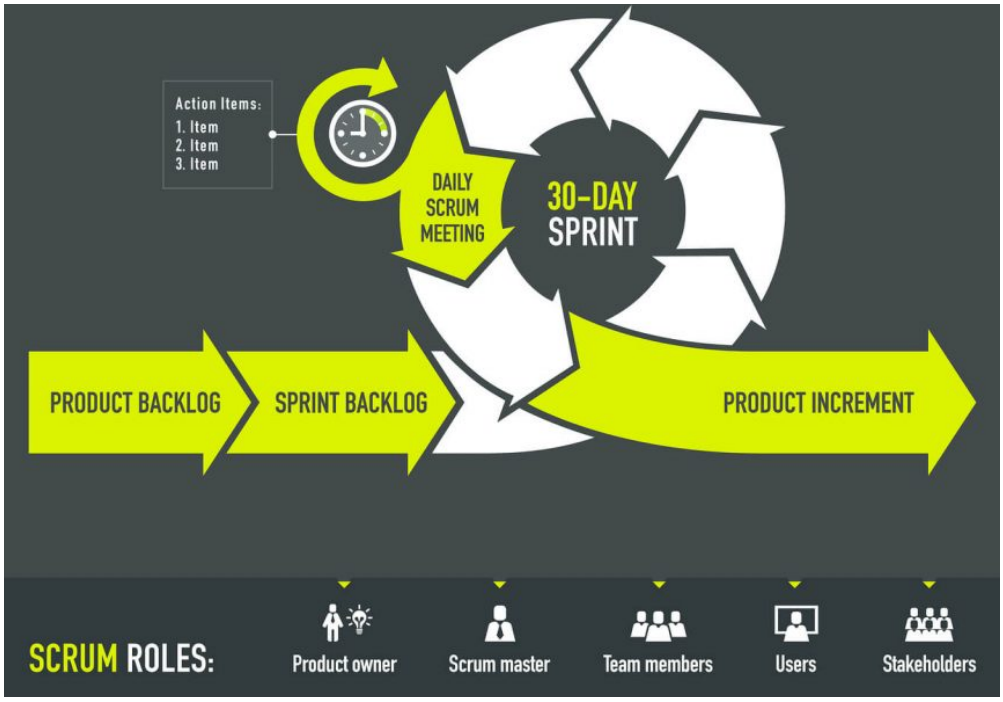

Figura 3.0.0 Fuente: https://www.digite.com/es/agile/que-es-scrum/

Resultados y análisis

- Las reuniones del equipo de desarrollo para el análisis de requisitos permitieron la identificación y priorización de historias de usuario.
- La planificación en sprints permitió llevar a cabo un trabajo ordenado y una medición del avance real del proyecto.
- De acuerdo con lo observado, se infiere que un software de calidad permite optimizar procesos a las empresas y evitar cambios y gastos a futuro.
- Las competencias de desarrollo de los miembros del equipo permitieron llevar a cabo los compromisos pactados y entregar una nueva versión de la aplicación.
- Las pruebas constantes realizadas a la aplicación sirvieron para garantizar el correcto funcionamiento del sistema.
- La empresa Innventa software ejecutó un análisis del estado actual en el que se encontraba la aplicación witcash, lo que permitió identificar las necesidades que presentaba la aplicación para ofrecer un servicio de calidad a sus usuarios.
- La configuración de despliegue e integración continuos facilitó al equipo de desarrollo probar nuevas funcionalidades en diferentes ambientes.
- La comunicación directa con la blockchain de bitcoin, por medio de herramientas como nbxplorer y bitcoin core permitió eliminar la dependencia de apis de terceros que generaban lentitud.
- Optimizar el cálculo de tarifas para envío de transacciones permitió reducir los costos y tiempo de espera.

1. Requisitos funcionales y no funcionales

Los requisitos desarrollados en la ejecución de este proyecto son los que se mencionan a continuación:

Funcionales:

- El sistema le permitirá a los usuarios crear una billetera de bitcoin y le entregará su correspondiente llave privada.
- El sistema le dará la opción al usuario de importar una llave privada en caso de que lo requiera.
- Al momento de crear o importar una llave privada, el usuario deberá poder crear un pin de seguridad, el cual se necesitará para poder enviar transacciones de bitcoin
- Al ingresar a la aplicación, el sistema le permitirá al usuario visualizar el balance y las transacciones de bitcoins de su billetera
- El usuario podrá crear bolsillos independientes dentro de su billetera de bitcoin, estos se generan de su misma llave privada
- El usuario podrá visualizar la dirección de su billetera para poder recibir bitcoin.
- El usuario podrá visualizar el detalle de sus transacciones, y en caso de necesitarlo, tendrá la opción de redirigir a blockchain info.
- El sistema le permitirá al usuario enviar bitcoin a otra dirección, esta dirección debe ser válida y puede ser de Witcash u otra externa.
- Para mayor facilidad, el usuario podrá crear contactos dentro de la aplicación, a los cuales les podrá agregar un nombre y una dirección de Bitcoin
- El usuario podrá ver el precio del Bitcoin dentro de la aplicación y su correspondiente valor en varias divisas (COP, USD, EUR, GBP BRL)
- El sistema le permitirá al usuario cambiar bitcoin por COP en un módulo denominado ‘witmoney’ y recibir dicho monto en una cuenta bancaria, para lograr esto el usuario deberá registrar de manera exitosa su información personal básica y financiera.
- El usuario podrá inscribir y administrar cuentas bancarias a las cuales se le podrá depositar el cambio realizado en el módulo de witmoney.
- El sistema le permitirá al usuario visualizar un histórico de ‘witmoneys’ realizados con su estado asociado
- El sistema le permitirá al usuario registrar su información personal básica y financiera en un módulo denominado KYC.
- El sistema le notificará dentro de la aplicación al usuario cuando haya finalizado la revisión del proceso de KYC, cuando se haya revisado la información de una inscripción de cuenta bancaria o cuando se haya realizado el depósito de un witmoney. Estas notificaciones están internacionalizadas en inglés o español.
- En caso de ser necesario, el usuario podrá hacer un backup de su llave privada

- Si el usuario requiere, podrá configurar datos biométricos (Footprint en Android - FaceId en iOS) como parámetro de seguridad al ingresar a la aplicación

No Funcionales:

- Los servicios web de la aplicación se deben documentar en swagger
- La base de datos del sistema debe ser de tipo NoSQL
- Los puertos de los servidores del sistema deben estar protegidos mediante firewall
- El sistema debe tener un nodo propio de bitcoin core para la comunicación con la blockchain.
- El sistema debe usar nbxplorer como explorador de bloques.
- La aplicación debe estar disponible en app store y play store.
- Los servicios del backend deben estar protegidos siguiendo las recomendaciones del OWASP, usando parámetros como timestamp y firmas HMAC en cada petición.
- La comunicación entre la aplicación y el backend debe ser por medio del protocolo https.
- La infraestructura debe desplegarse en máquinas virtuales de google cloud con un sistema operativo CentOS 8
- La alta disponibilidad de la información de la base de datos deberá estar garantizada por medio de un clúster de Mongodb Atlas
- El backend debe estar implementado con tecnologías modernas como NodeJs, y debe ser desarrollado utilizando buenas prácticas.

2. Diseño y Arquitectura de la solución

A continuación se presenta diagrama de despliegue de la figura 4.0.0, el cual describe de manera gráfica el sistema desarrollado y sus diferentes componentes:

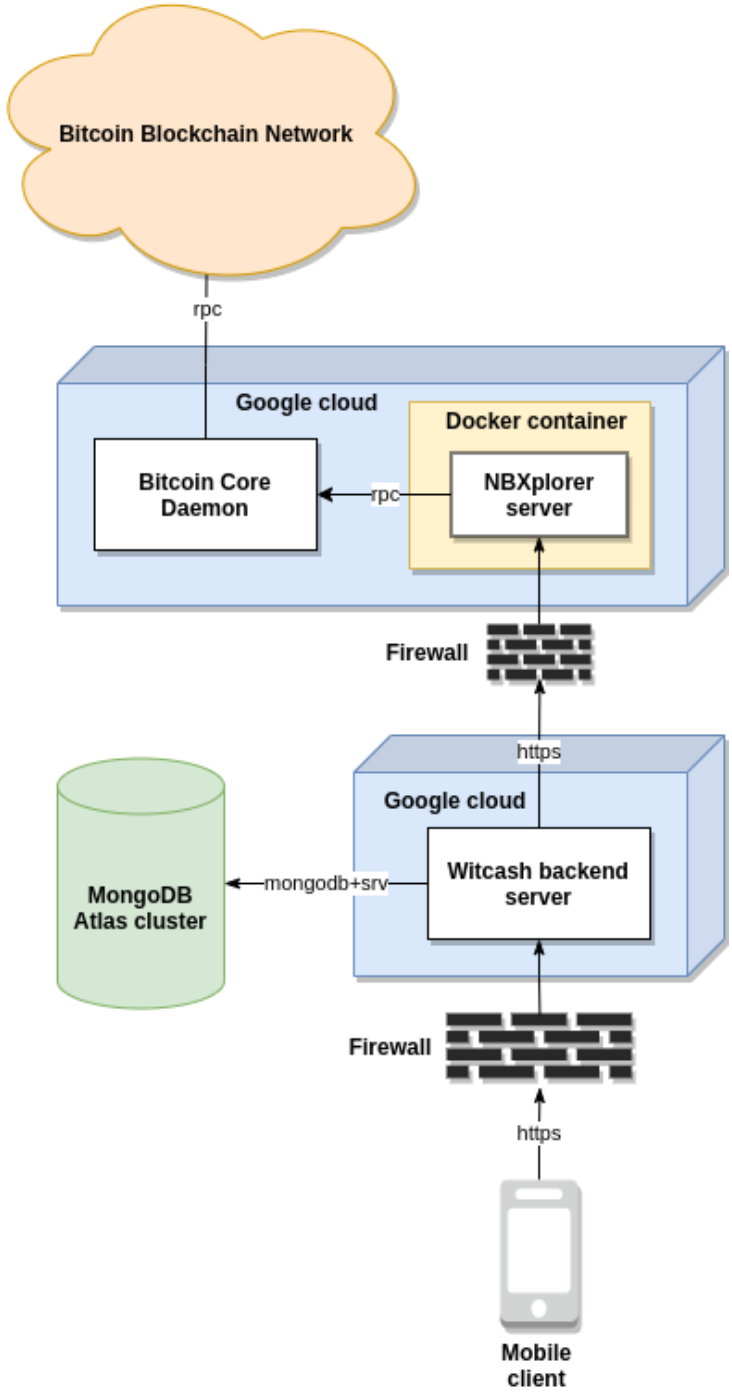

Figura 4.0.0

- Bitcoin Core Daemon: Este componente es Open Source y contiene una copia exacta y sincronizada de la información de la blockchain de bitcoin. Su principal función es la de proveer autonomía al sistema al momento de requerir información de la blockchain, de esta manera el sistema no depende de servicios de terceros. La comunicación con ese componente se hace por medio de rpc.

- NBXplorer server: Es otra herramienta Open Source desarrollada en .NET, la cual es utilizada como un explorador de la blockchain. Es decir, toda la información que requiere el sistema relacionada con la blockchain de bitcoin, es solicitada a este componente por medio de un API Restful, y este a su vez, realiza la búsqueda en el componente de bitcoin core. Como se muestra en el diagrama, este componente está encapsulado dentro de un contenedor de Docker para garantizar su alta disponibilidad. Este componente se comunica con el demonio de bitcoin core por medio de rpc.
- Google Cloud: Es la plataforma donde Google ofrece una variedad de servicios basados en la nube. En este componente es donde se encuentran desplegados los servidores de la aplicación. Los servidores del sistema fueron protegidos con varias técnicas de seguridad informática, una de ellas es la protección de los puertos por medio de firewalls, como se observa en el diagrama de la figura 4.0.0.
- MongoDB Atlas cluster: Es un clúster de bases de datos no relacional, donde se almacena la información propia del sistema.
- Witcash backend server: Es el backend de la aplicación y el core del sistema, contiene toda la lógica del negocio, expone los servicios web necesarios a la aplicación móvil para su correcto funcionamiento, accede a la base de datos para guardar u obtener información del sistema y establece comunicación con NBXplorer cuando se requiere información de la blockchain
- Bitcoin blockchain network: Este componente es la red blockchain de bitcoin y el sistema accede a este por medio de la sincronización con el demonio de bitcoin core.
- Mobile client: Es el frontend de la aplicación desarrollado en Ionic, este es el componente con el que interactúa el usuario final, se despliega en las tiendas Appstore y Play store para poder ser descargado por los usuarios. Este componente no entró dentro del alcance de esta práctica empresarial, por lo cual fue desarrollado simultáneamente por otro equipo de desarrollo de la organización.

3. Implementación

APIs Restful del backend

Como se indicó anteriormente, la lógica del negocio está contenida dentro del componente witcash backend server, dentro del cual se expusieron 54 servicios web, los cuales se encuentran modularizados según su propósito. A continuación se muestran los módulos más relevantes y la respectiva documentación de sus servicios web.

Módulo Wallet (Figura 5.1.0): Contiene toda la lógica de creación de billeteras de bitcoin, a su vez, cada billetera puede contener bolsillos y además contiene los servicios necesarios para recuperar la billetera en caso de necesitarlo.

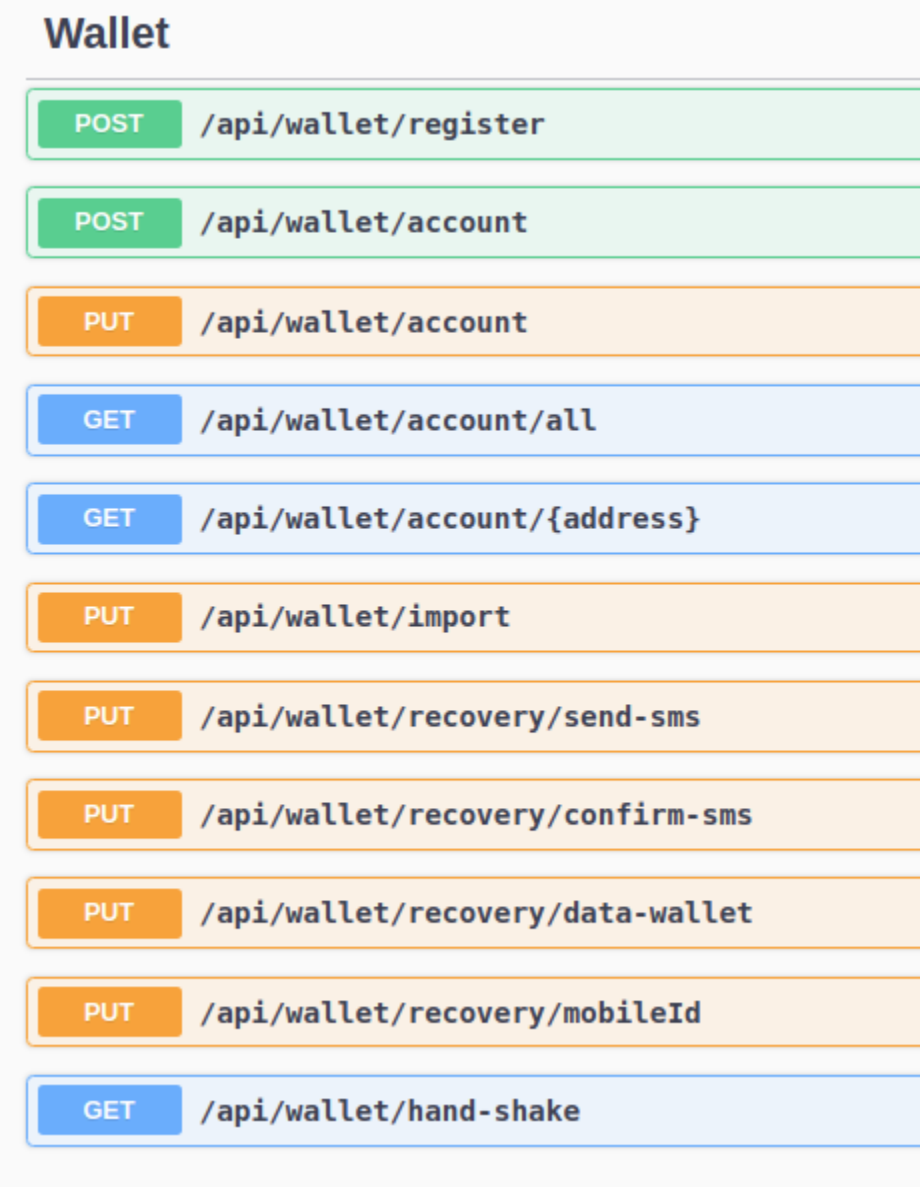

Figura 5.1.0

Módulo de usuario (Figura 5.2.0): Este módulo contiene el manejo del pin de seguridad de los usuarios. Este pin es utilizado como parámetro de seguridad a la hroa de enviar bitcoins desde una billetera

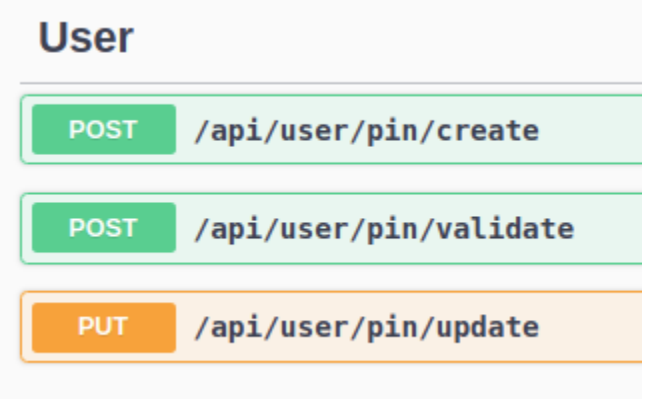

Figura 5.2.0

Módulo Bitcoin (Figura 5.3.0): Este módulo hace referencia a todos los servicios web que necesita la aplicación móvil para interactuar con la información de la blockchain, tales como visualizar el balance de bitcoin de una billetera, obtener las transacciones realizadas por el usuario, visualizar el detalle de una transacción, enviar transacciones, etc.

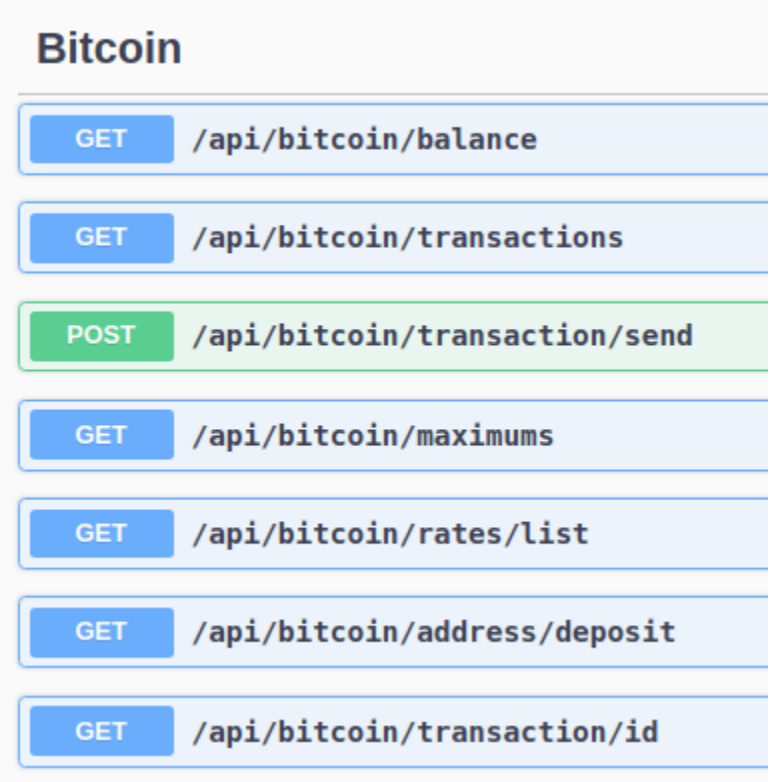

Figura 5.3.0

Módulo contactos (Figura 5.4.0): La aplicación le permite a los usuarios almacenar, editar y eliminar contactos para enviar bitcoins a personas recurrentes de una manera más fácil y rápida, este módulo se encarga de ese funcionamiento.

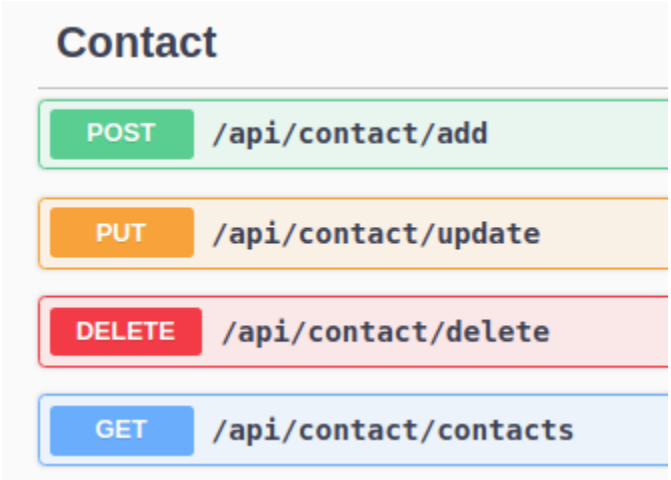

Figura 5.4.0

Módulo KYC (Figura 5.5.0): Al tratarse de una aplicación financiera, se requiere que los usuarios realicen un proceso llamado Know Your Customer, el cual consiste en recolectar la información personal y financiera básica de los usuarios para poder ofrecerles a éstos, servicios que involucren transacciones bancarias.

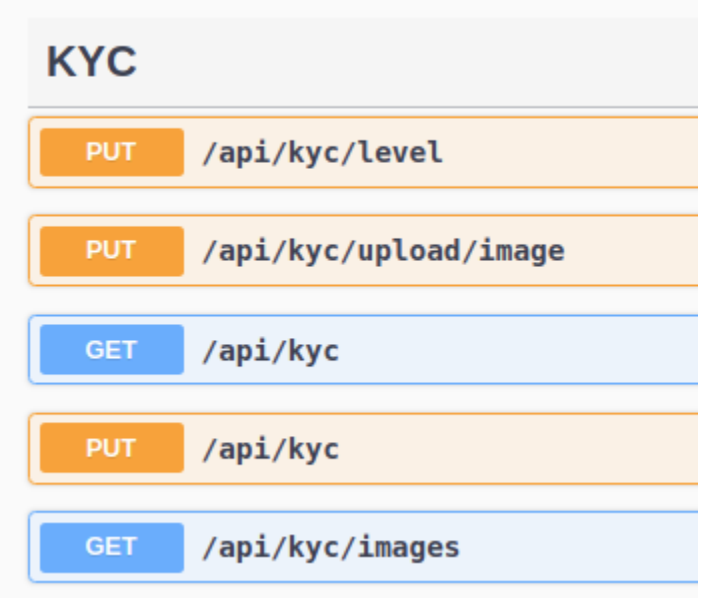

Figura 5.5.0

Módulo Witmoney (Figura 5.6.0): Witmoney es un servicio en el que los usuarios que hayan completado el proceso de KYC exitosamente y que hayan posteriormente inscrito cuentas bancarias, puedan monetizar sus bitcoins a pesos colombianos. El valor equivalente en pesos colombianos a los bitcoins monetizados menos una pequeña comisión del sistema, es depositado en la cuenta bancaria del usuario.

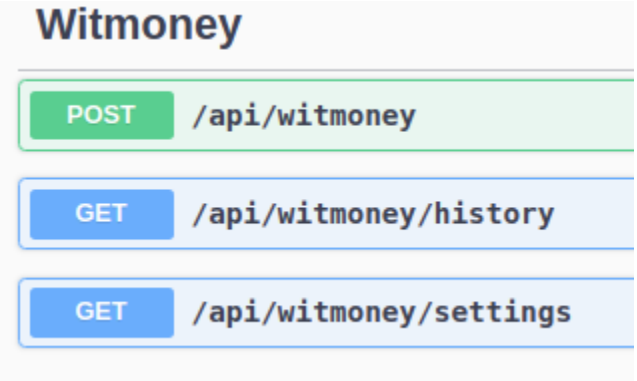

Figura 5.6.0

Módulo Bank Account (Figura 5.7.0): Este módulo, le permite a los usuarios registrar los datos de sus cuentas bancarias una vez completen de manera exitosa el proceso de KYC. Estas cuentas bancarias se utilizan para recibir los bitcoins monetizados en el módulo de Witmoney.

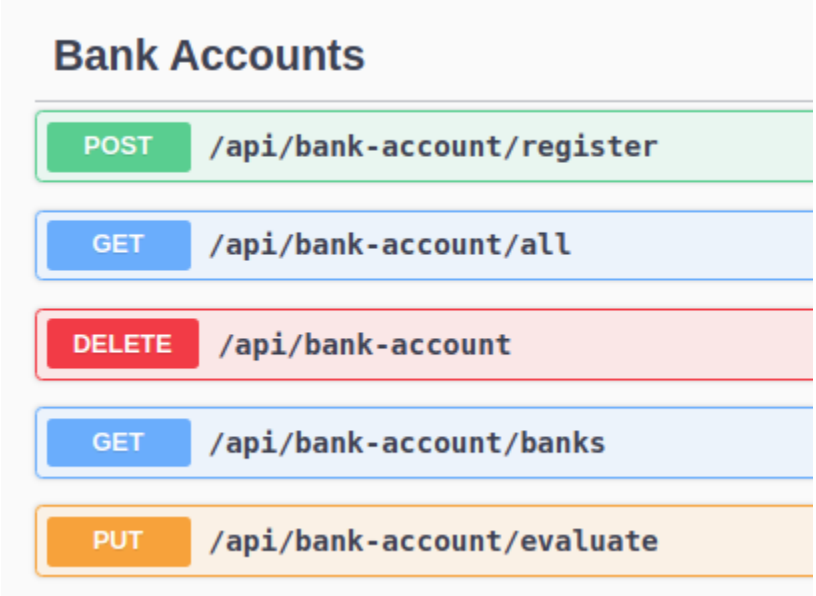

Figura 5.7.0

Servidores

En la Figura 6.0.0 podemos observar las diferentes instancias de compute engine que se tienen alojadas en google cloud, en las que se realizó el despliegue de los diferentes componentes y ambientes del sistema. Además en la Figura 6.1.0 podemos observar los detalles de de la instancia witcash-backoffice-production, donde se observa que se cuenta con un sistema operativo CentOS 8.

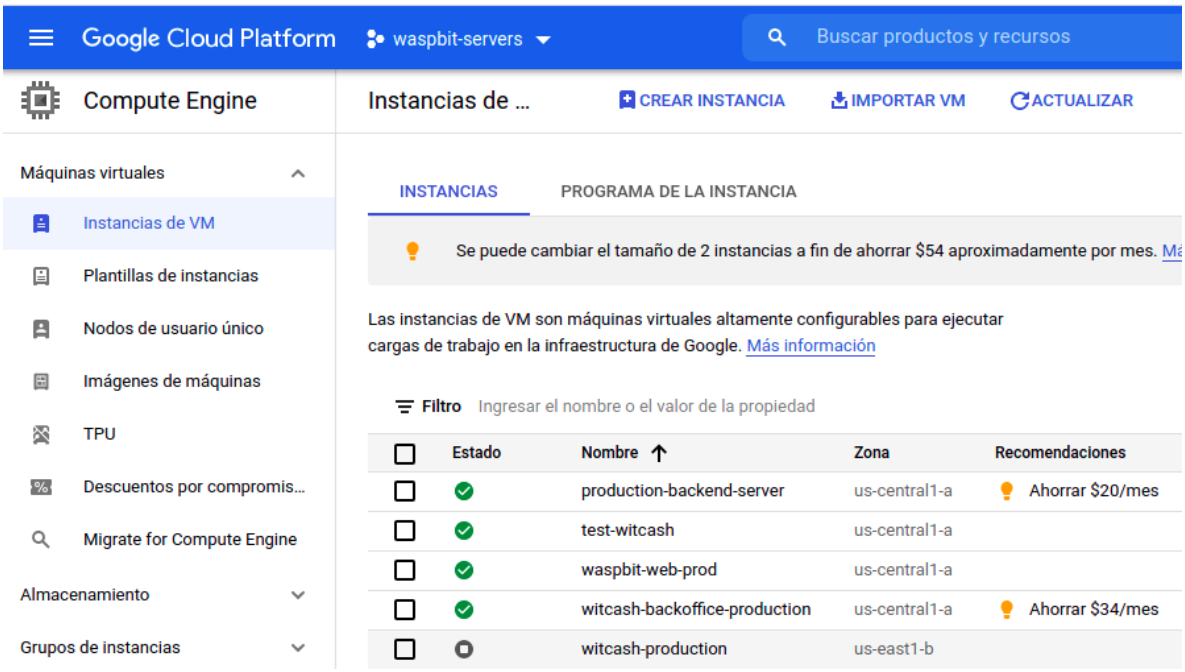

Figura 6.0.0

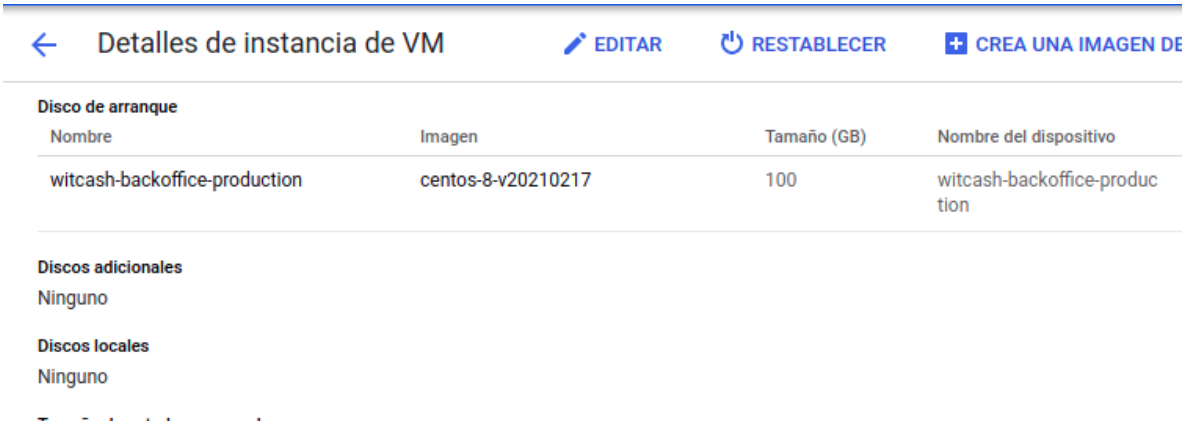

Figura 6.1.0

También podemos observar (Figura 6.1.2) que en el ambiente de test se utilizó la herramienta pm2 [10] para desplegar los proyectos 'backoffice-front', 'backoffice-service', 'wallet-service'

```
carlos@asus-pc:~$ ssh devops@test.witcash.io
devops@test.witcash.io's password:
Last failed login: Sun Jul 11 19:07:39 UTC 2021 from 120.239.196.55 on ssh:notty
There was 1 failed login attempt since the last successful login.
Last login: Sat Jul 10 17:00:41 2021 from 179.12.52.145
manpath: can't set the locale; make sure $LC_* and $LANG are correct
[devops@test-witcash ~]$ pm2 list
```

id	name	namespace	version	mode	pid	uptime	↺	status
57	backoffice-front	default	N/A	fork	1076590	4D	0	online
56	backoffice-service	default	0.0.1	fork	862409	12D	0	online
51	wallet-service	default	0.0.1	fork	796266	12D	0	online

```
[devops@test-witcash ~]$
```

Figura 6.1.2

Clúster de base de datos MongoDB

En la Figura 7.0.0 se observa que se tienen dos proyectos en Atlas, uno para producción y otro para test, cada uno contiene un cluster de mongoDB, con 3 nodos de réplicas como se puede observar en la figura 7.1.0

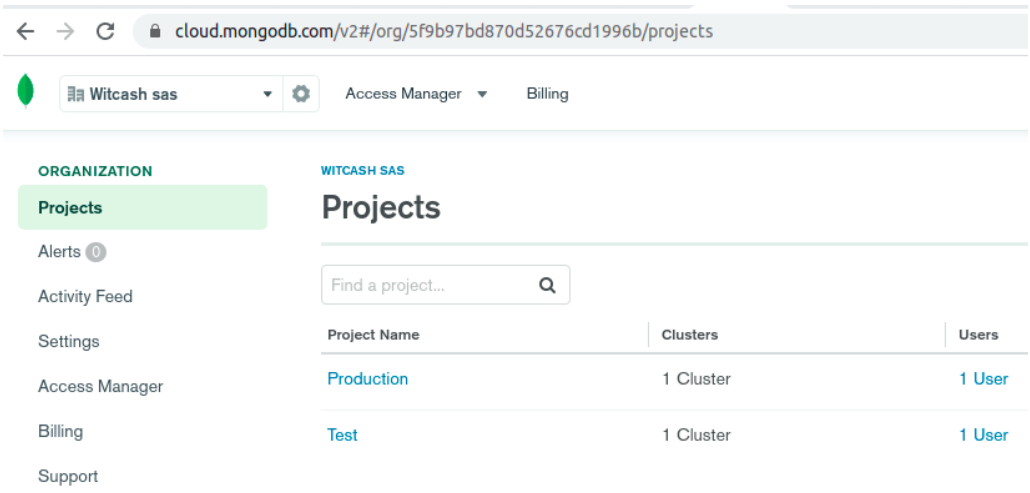

Figura 7.0.0

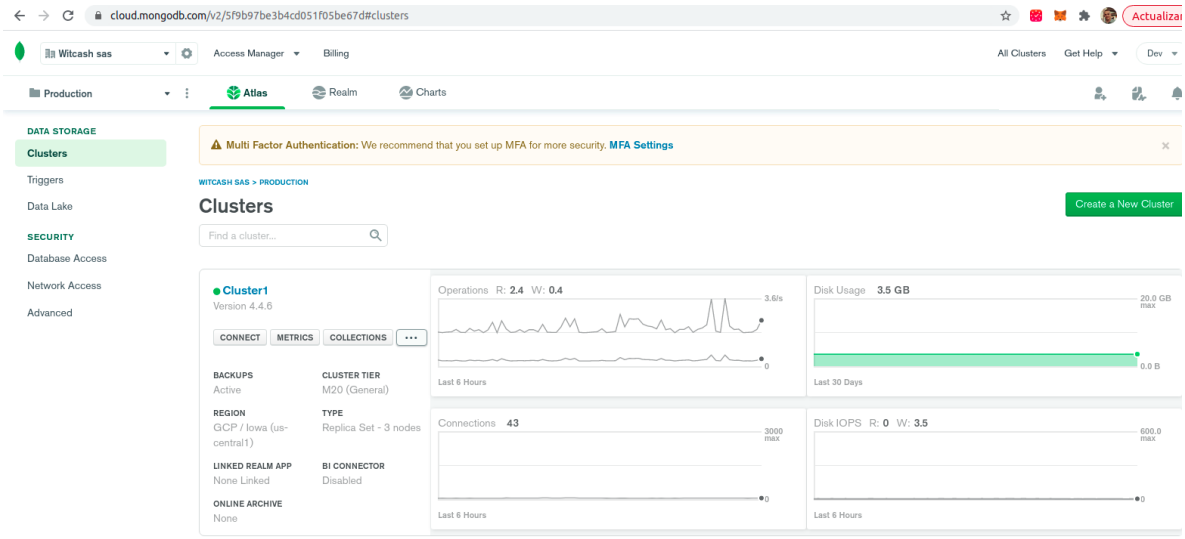

Figura 7.1.0

CI/CD Devops

En la Figura 8.0.0 se puede observar el proyecto witcash backend wallet alojado en gitlab y algunos de los pipelines ejecutados en el proceso de integración y despliegue continuo

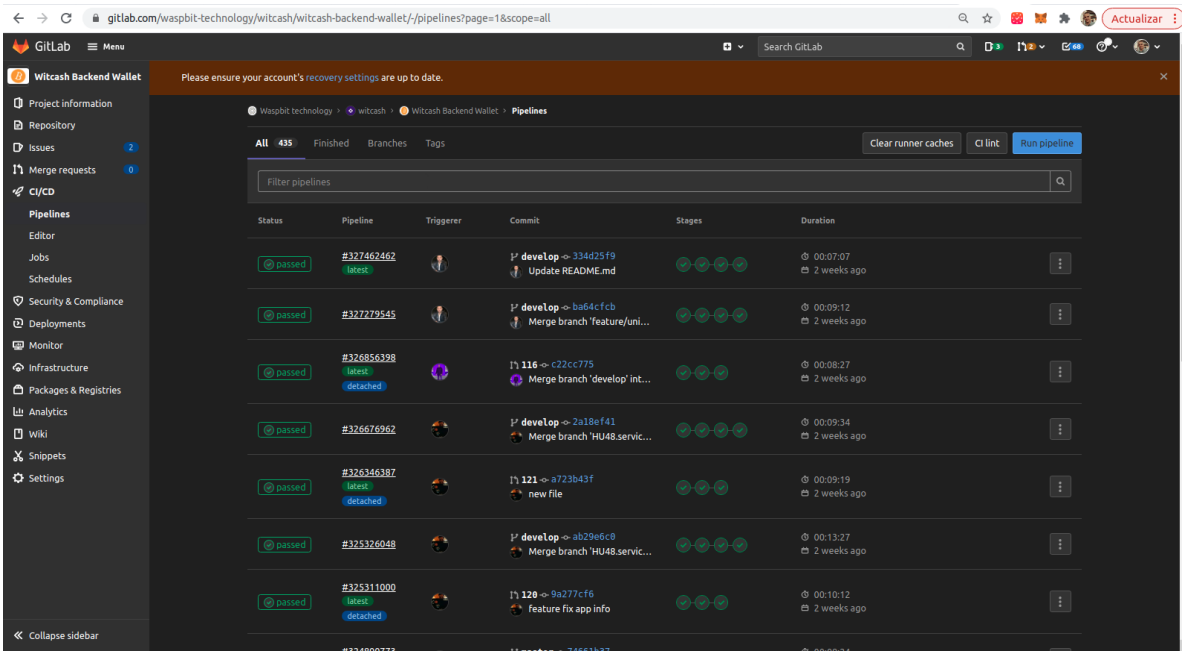

Figura 8.0.0

Servidor de NBXplorer y Bitcoin core Daemon

Estos dos componentes hacen parte de una instancia “bitcoin-node” diferente a la del backend del sistema de google cloud como se muestra en la figura 9.0.0.

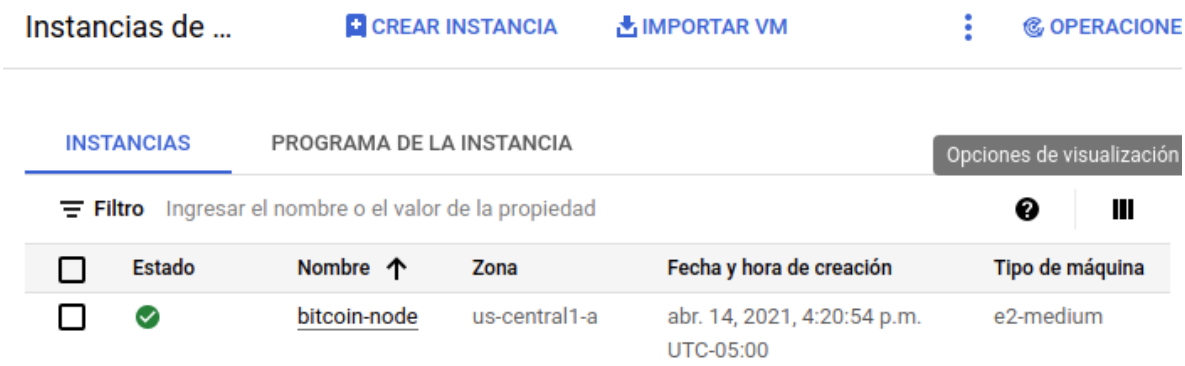

Figura 9.0.0

En la figura 9.1.0 se muestra una captura de pantalla del servidor de NBXplorer, que como se ha mencionado anteriormente, se encarga de la comunicación con el nodo de bitcoin del sistema por medio de rpc.

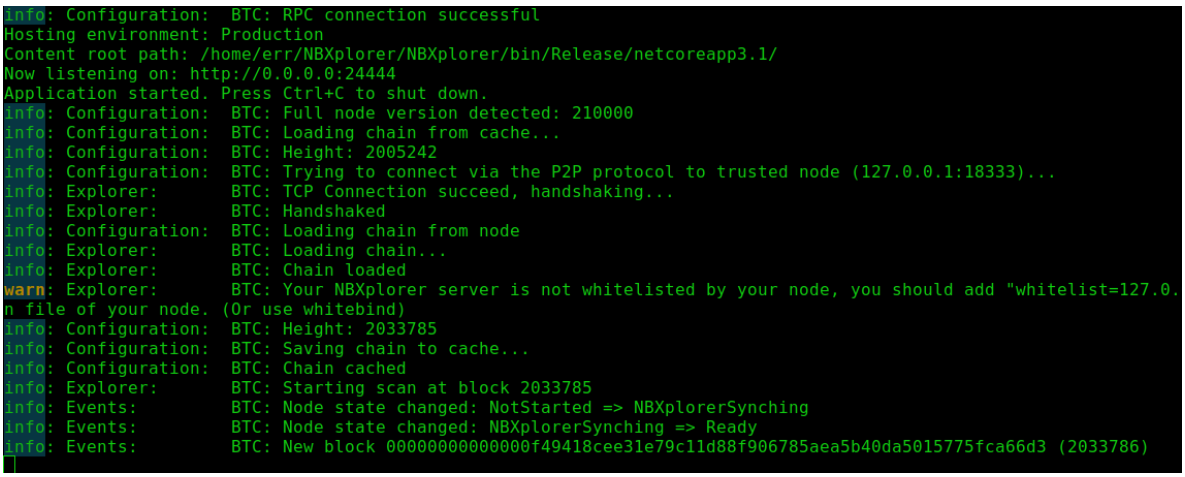

Figura 9.1.0

En la imagen anterior se evidencia que la conexión con bitcoin core por medio de RPC fue exitosa, y que además se encuentra totalmente sincronizado. En la última línea de la imagen se ve que mientras se tomaba captura de la imagen, el nodo se sincronizó con un nuevo bloque de la blockchain.

Interfaz gráfica

A continuación se adjuntan imágenes con la descripción de cada módulo dentro de la aplicación.

- **Registro** (Figura 10.0.0): tenemos la opción de crear una nueva billetera.

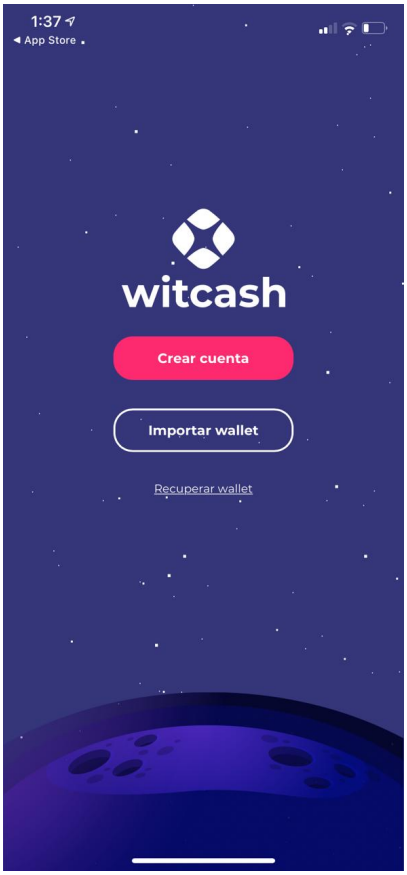

Figura 10.0.0

Cuando un usuario crea una nueva billetera puede observar 12 palabras (Figura 10.0.1) que corresponden a la llave privada que le dan completo control sobre los bitcoin que se reciban en dicha billetera.

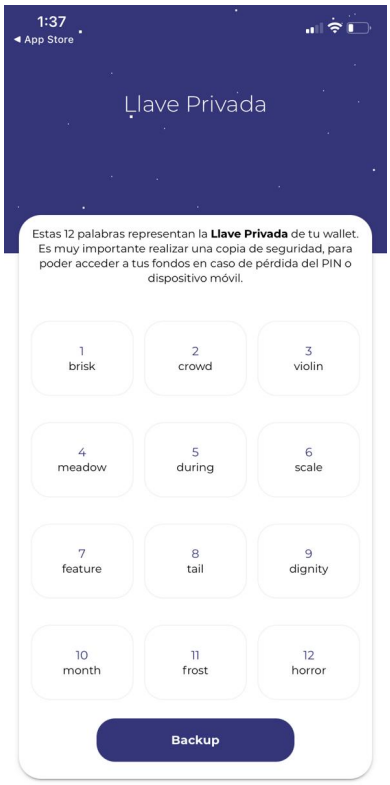

Figura 10.0.1

- **Inicio** (Figura 10.1.0): se observan los módulos activos de la aplicación:
    - **Witwallet**: Módulo que permite visualizar historial de transacciones, consultar balance, enviar transacciones y obtener dirección de bitcoin para recibir transferencias.
    - **Witmoney**: Módulo que permite intercambiar bitcoin por pesos colombianos, inscribir y administrar cuentas bancarias para recibir los pagos y consultar el historial de intercambios realizados.

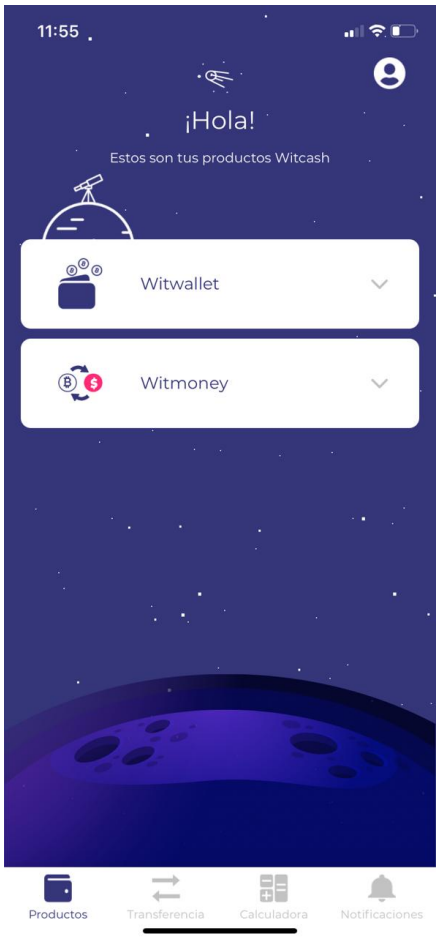

Figura 10.1.0

- **Witwallet** (Figura 10.2.0): Al ingresar a este módulo se puede observar el balance en bitcoin y dólares, también se tiene la opción para enviar transacciones y obtener dirección para recibir bitcoin.

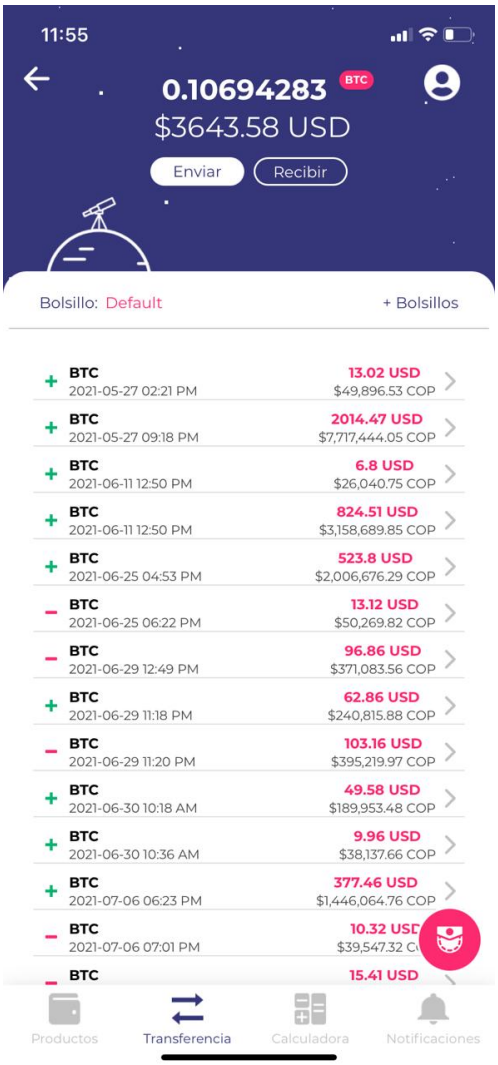

Figura 10.2.0

- **Enviar** (Figura 10.2.1): Al seleccionar la opción enviar, veremos el siguiente modal,en el que debemos ingresar la dirección de destino, la cantidad a enviar (En bitcoin o dólares) y seleccionar la prioridad con la que queremos que la red de bitcoin procese la transacción.

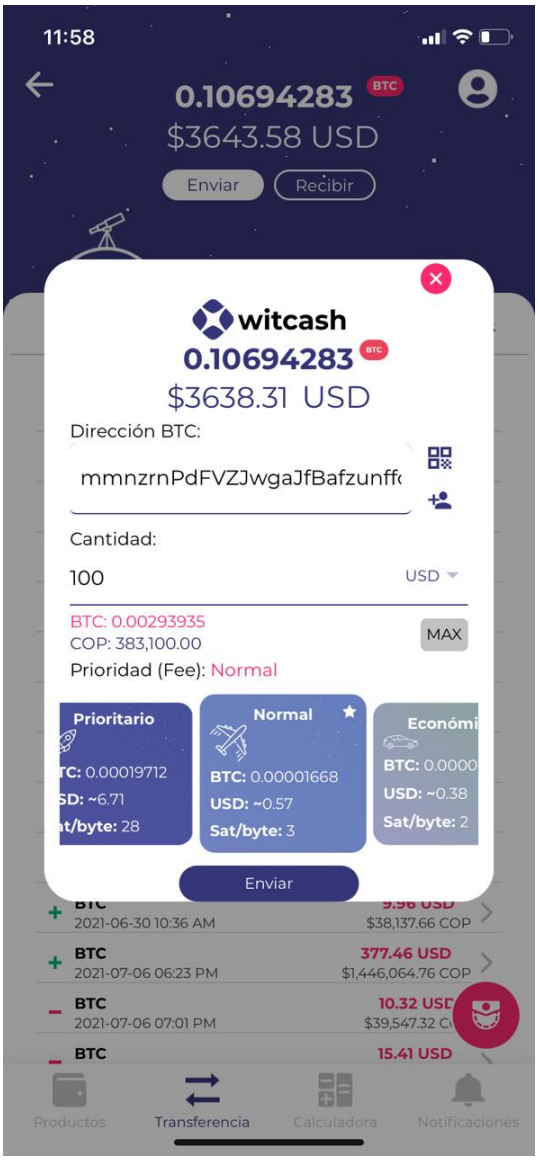

Figura 10.2.1

- **ingresar pin** (Figura 10.2.2): Al presionar el botón enviar la aplicación solicita ingresar el pin de seguridad, si este es correcto se procesa el envío de la transacción.

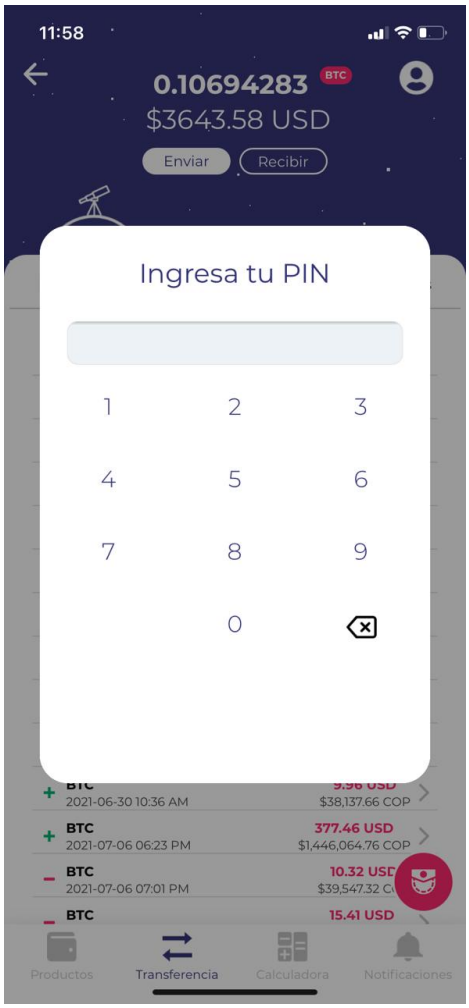

Figura 10.2.2

- **Recibir** (Figura 10.2.3): Se despliega un modal con la dirección de bitcoin que debe ser compartida para recibir fondos, se tiene la opción de copiar en el portapapeles y también se puede observar un QR que puede ser escaneado desde otro celular con la dirección de bitcoin.

Figura 10.2.3

- **Witmoney** (Figura 10.3.0): Desde el inicio, si ingresamos por la opción witmoney, podremos observar el balance, los límites que pueden ser procesados por medio de un ‘witmoney’ y 3 secciones que serían: servicio, historial y cuentas. Por defecto se ingresa a la opción servicio.

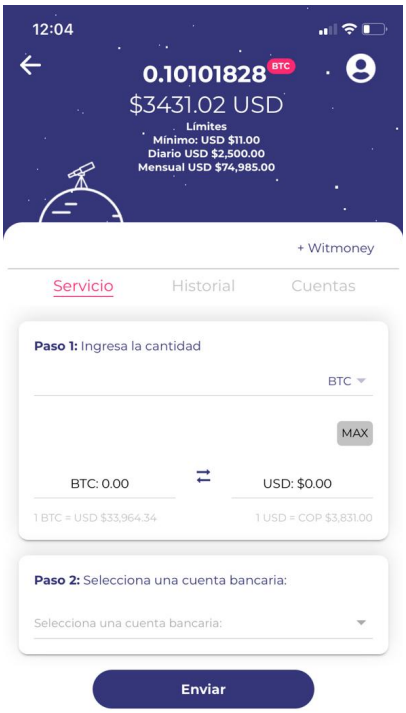

Figura 10.3.0

- **Servicio** (Figura 10.3.1): En la sección servicio tenemos la opción de intercambiar bitcoin por pesos colombianos. En este paso se debe ingresar en bitcoin o dólares la cantidad a intercambiar, además se debe seleccionar la cuenta bancaria en la que se van a depositar los fondos. Posteriormente se observa la comisión que se debe pagar a los mineros de bitcoin, la tarifa que cobra witcash por el intercambio y el total en COP que el usuario va a recibir en su cuenta bancaria. Para finalizar el usuario debe presionar el botón de envío e ingresar el pin de seguridad.

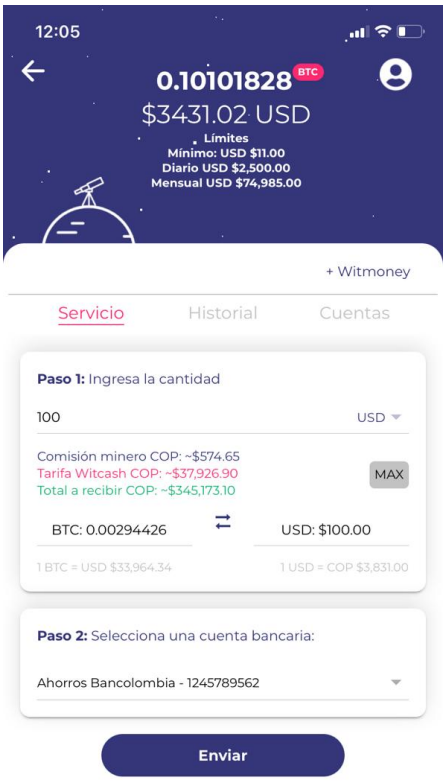

Figura 10.3.1

- **Historial** (Figura 10.3.2): En esta sección observamos el histórico de ‘witmoney’ realizados por el usuario, además se tiene la opción de ver el detalle de cada uno (Figura 2.3.3).

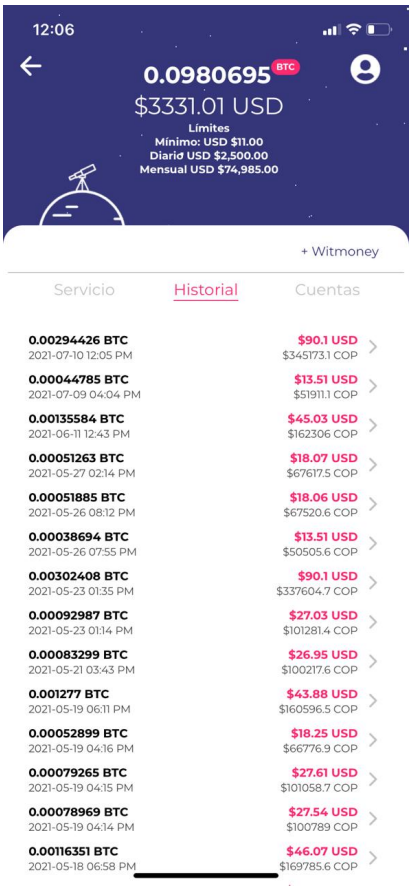
12:06
0.0980695 BTC
$3331.01 USD
Límites
Mínimo: USD $11.00
Diario USD $2,500.00
Mensual USD $74,985.00
+ Witmoney
Servicio
Historial
Cuentas
0.00294426 BTC
2021-07-10 12:05 PM
$90.1 USD
$345173.1 COP
0.00044785 BTC
2021-07-09 04:04 PM
$13.51 USD
$51911.1 COP
0.00135584 BTC
2021-06-11 12:43 PM
$45.03 USD
$162306 COP
0.00051263 BTC
2021-05-27 02:14 PM
$18.07 USD
$67617.5 COP
0.00051885 BTC
2021-05-26 08:12 PM
$18.06 USD
$67520.6 COP
0.00038694 BTC
2021-05-26 07:55 PM
$13.51 USD
$50505.6 COP
0.00302408 BTC
2021-05-23 01:35 PM
$90.1 USD
$337604.7 COP
0.00092987 BTC
2021-05-23 01:14 PM
$27.03 USD
$101281.4 COP
0.00083299 BTC
2021-05-21 03:43 PM
$26.95 USD
$100217.6 COP
0.001277 BTC
2021-05-19 06:11 PM
$43.88 USD
$160596.5 COP
0.00052899 BTC
2021-05-19 04:16 PM
$18.25 USD
$66776.9 COP
0.00079265 BTC
2021-05-19 04:15 PM
$27.61 USD
$101058.7 COP
0.00078969 BTC
2021-05-19 04:14 PM
$27.54 USD
$100789 COP
0.00116351 BTC
2021-05-18 06:58 PM
$46.07 USD
$169785.6 COP

Figura 10.3.2

Figura 10.3.3

- **Cuentas** (Figura 10.3.4): Sección para consultar, eliminar o agregar una nueva cuenta bancaria.

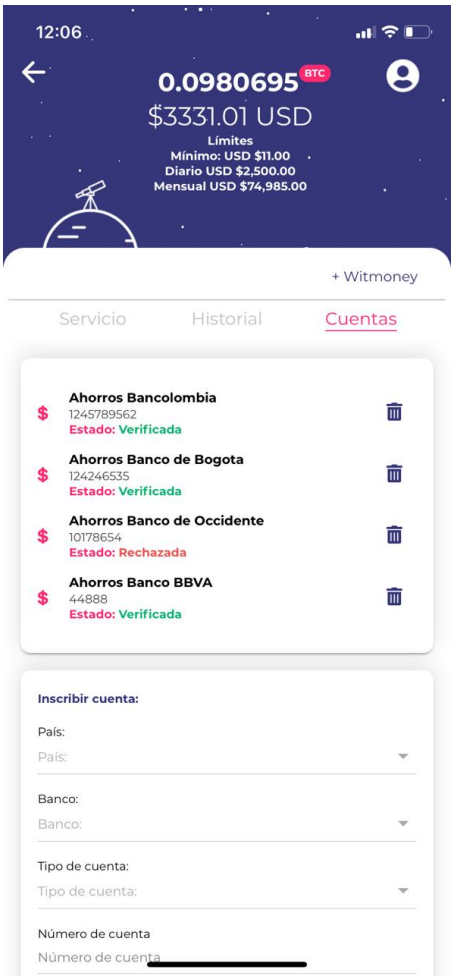

Figura 10.3.4

- **Calculadora** Figura(10.4.0) En esta pantalla el usuario puede comparar diferentes cantidades de bitcoin con las divisas: USD, EUR, COP, GBP, BRL. Debe seleccionar las dos divisas que quiere observar

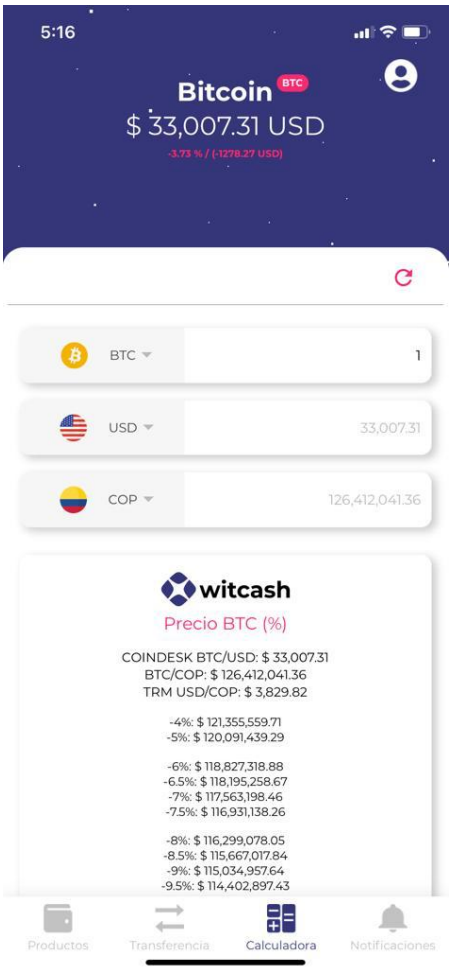

Figura 10.4.0

- **Notificaciones** Figura(10.4.0): La aplicación presentará un listado de notificaciones al usuario, al presionar clic se visualizará el detalle de la notificación Figura(10.4.1).

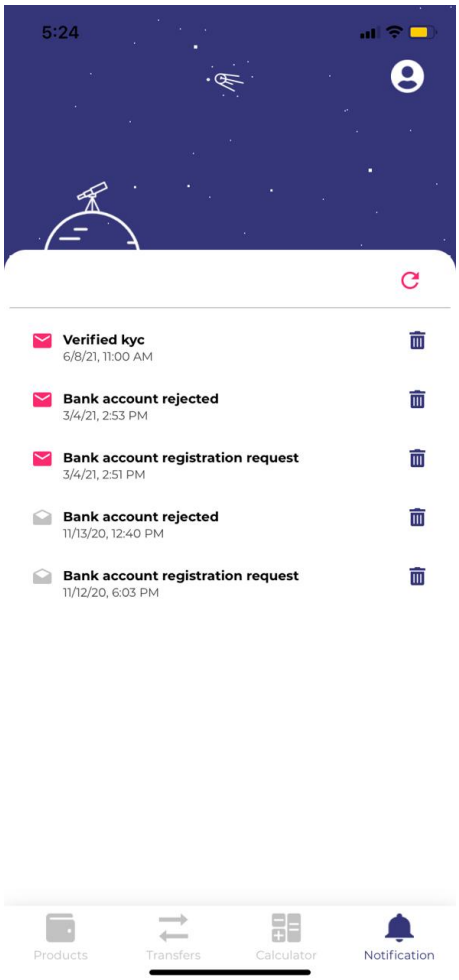
5:24
Verified kyc
6/8/21, 11:00 AM
Bank account rejected
3/4/21, 2:53 PM
Bank account registration request
3/4/21, 2:51 PM
Bank account rejected
11/13/20, 12:40 PM
Bank account registration request
11/12/20, 6:03 PM
Products
Transfers
Calculator
Notification

Figura 10.4.0

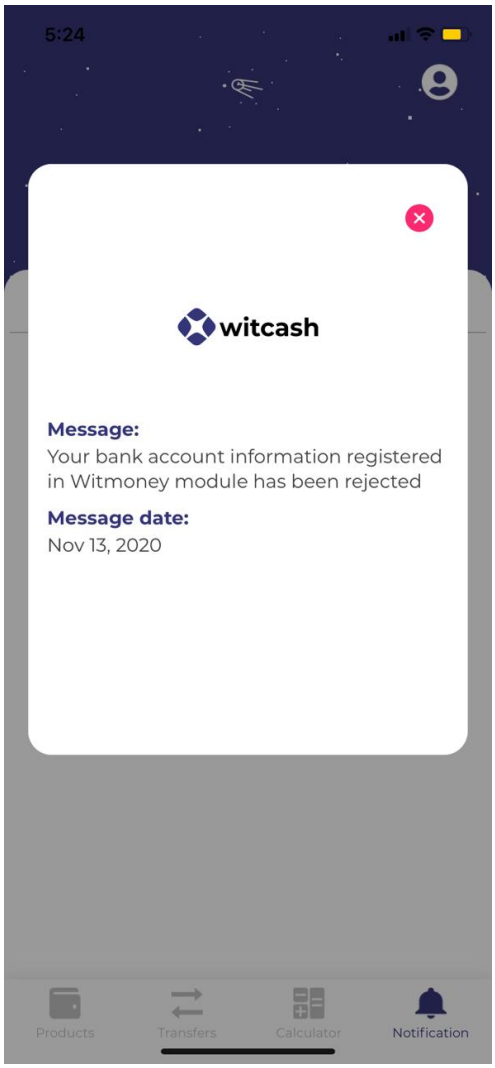

Figura 10.4.1

4. Pruebas realizadas

A continuación se presenta información de gráficas tomadas del backoffice de la aplicación del ambiente de **pruebas.**

1. Transacciones de witwallet.

   En esta gráfica podemos observar la cantidad en bitcoin que se ha transferido en el módulo y la comisión en bitcoin generada para la compañía, se puede observar que el flujo de transacciones es mucho mayor en el mes de Abril donde se inició la fase de pruebas y disminuyen hasta julio. Esta gráfica corresponde al año 2021.

Figura 11.0.0

2. Transacciones de witmoney.

En esta gráfica (Figura 11.1.0) se puede observar el monto total transferido en el módulo de witmoney, la comisión obtenida por la compañía, el número total de transacciones aprobadas, pendientes y rechazadas. También se puede observar un flujo más alto de transacciones en los meses de pruebas

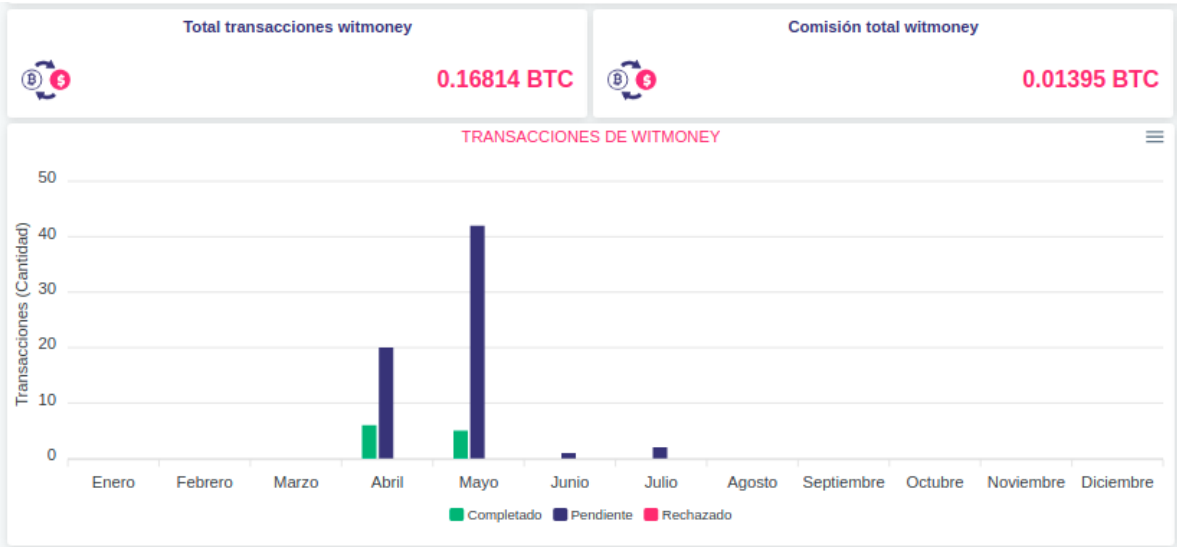

Figura 11.1.0

Uno de los principales focos de la aplicación con los cuales se pretende competir en el mercado es la seguridad y la rapidez, por lo cual se realizó una encuesta a los 10 stakeholders del proyecto, en el cual evaluaron la seguridad, la velocidad, la satisfacción al usar la aplicación y los bugs encontrados. A continuación se anexan los análisis de dicha evaluación.

Como se puede apreciar en la figura 11.2.0, el nivel de satisfacción de los stakeholders en términos generales fue bueno, ya que en una escala de 1 a 5 el mayor porcentaje obtuvo la puntuación 4.

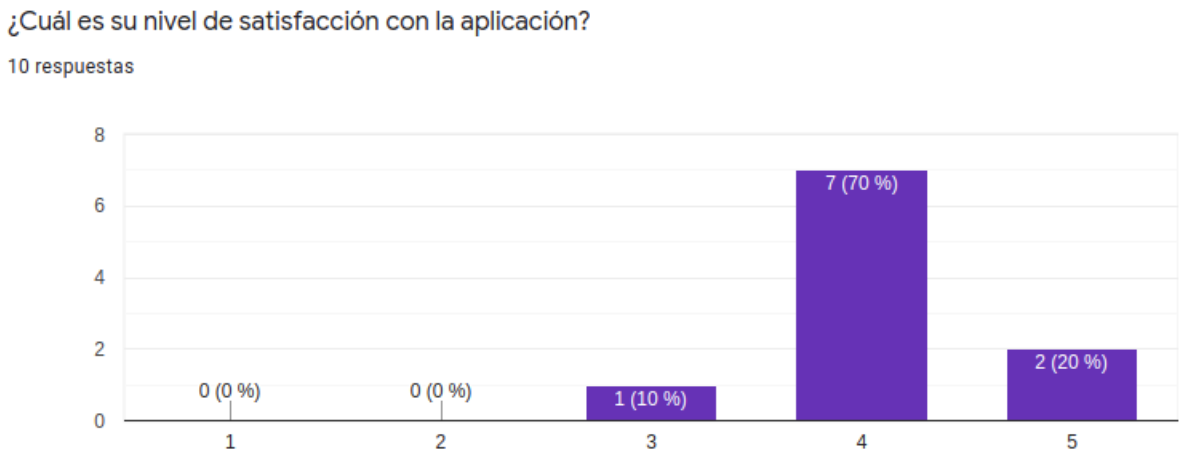

Figura 11.2.0

La gráfica (figura 11.3.0), indica que el funcionamiento de la aplicación en términos de velocidad es bastante bueno, porque como se puede apreciar, todas las respuestas fueron 4 o 5.

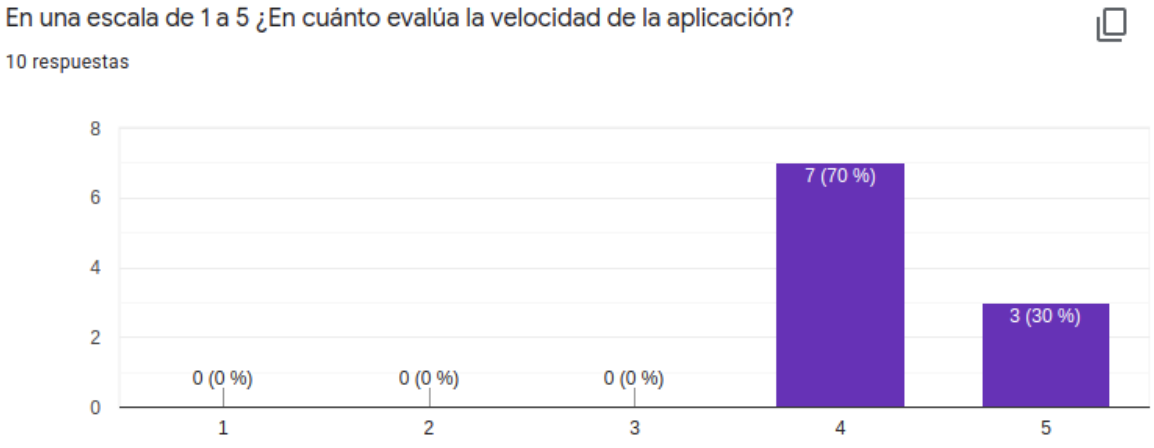

Figura 11.3.0

En cuanto a la seguridad de la aplicación, se puede encontrar en la figura 5.2 que casi la mitad de los encuestados votaron por 3, lo que puede indicar que hay aspectos dentro de la aplicación que se pueden mejorar para tratar de subir la confianza de los usuarios finales a la hora de usar la aplicación.

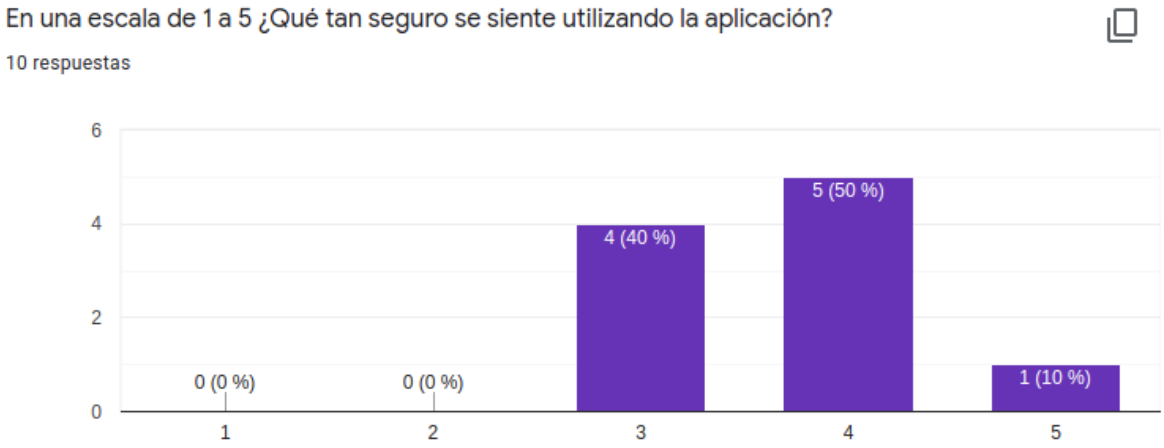

Figura 11.4.0

El porcentaje de fallas encontradas en la figura 11.5.0 dentro de la aplicación es bastante alto, por lo cual se debe indagar un poco más y prestar atención a la próxima pregunta del formulario para saber si los bugs son significativos e influyen en el correcto funcionamiento de la aplicación, o si son bugs de forma, que pueden tener una prioridad baja.

Figura 11.5.0

Analizando las respuestas (figura 11.6.0) de los bugs, se puede evidenciar que los bugs no son mayores, es decir, no afectan el flujo principal de la aplicación, por lo cual se deben solucionar pero tienen una prioridad baja.

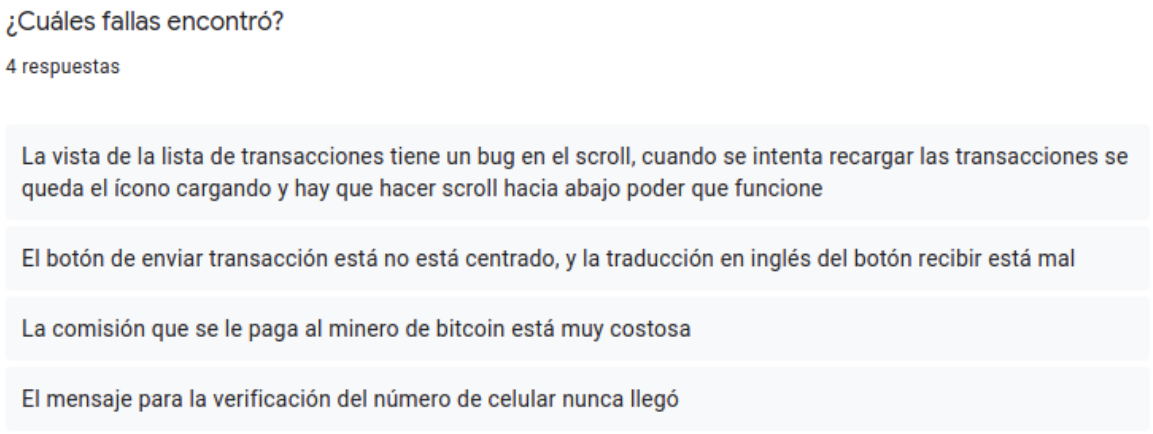

Figura 11.6.0

La última pregunta (Figura 11.7.0) nos puede orientar un poco hacia los próximos pasos que puede tomar la aplicación en las próximas versiones, ya que las sugerencias recibidas fueron muy relevantes pero requieren un esfuerzo de desarrollo bastante elevado. Analizando las respuestas, se puede evidenciar que los usuarios requieren tener más criptomonedas dentro de la aplicación, tales como

ethereum y tether, y además requieren que la aplicación haga más énfasis en la custodia de la clave privada.

Figura 11.7.0

Conclusiones

La revisión de los resultados obtenidos por medio de gráficas, funcionalidad de la aplicación y encuestas sobre experiencia de usuario, nos permiten concluir que construir nuevamente el componente backend del sistema fue una decisión acertada, ya que se redujeron tiempos de espera, costos y bugs presentados en producción. También se mejoró considerablemente la administración de la infraestructura al cambiar a CentOS 8 el sistema operativo del servidor, en el que se implementó integración y despliegue continuo, optimizando los tiempos de desarrollo y pruebas.

Además, tomar la decisión de usar un nodo propio de bitcoin core, brindó autonomía al cliente, ya que permite prescindir del uso de terceros para la comunicación con la blockchain y así reducir tiempos de espera y costos administrativos. A partir de los datos expuestos anteriormente se puede concluir que el trabajo realizado permitirá al cliente brindar una mejor experiencia a sus usuarios.

Referencias Bibliográficas

[1] Learn me bitcoin, Blog, 2015, 2021, Walker, G. https://learnmeabitcoin.com/

[2] ¿Qué es la Cuarta Revolución Industrial?. (2018). Recuperado 12 de julio de 2021, de Salesforce website:
https://www.salesforce.com/mx/blog/2018/4/Que-es-la-Cuarta-Revolucion-Industrial.html

[3] Morgado, J. (2016) La importancia del 'blockchain' para los servicios financieros. Recuperado 12 de julio de 2021, de Expansión website:
https://www.expansion.com/economia-digital/protagonistas/2016/10/26/58061701ca47412f138b4652.html

[4] Parrondo, L. (2018). Tecnología blockchain, una nueva era para la empresa. Recuperado 12 de julio de 2021, de Books.google website:
https://books.google.com.co/books?hl=es&lr=&id=f7SIDwAAQBAJ&oi=fnd&pg=PA11&dq=que+es+la+blockchain+de+ethereum+&ots=L4wNV_he-j&sig=yZDCI5snBH_f1vPa7ETx-X3yyLE&redir_esc=y#v=onepage&q=que%20es%20la%20blockchain%20de%20ethereum&f=false

[5] Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. https://bitcoin.org/bitcoin.pdf

[6] Schwaber, K. Sutherlandm, J. (2017) La Guía Definitiva de Scrum: Las Reglas del Juego
https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Spanish-European.pdf

[7] OWASP. REST Security Cheat Sheet:
https://cheatsheetseries.owasp.org/cheatsheets/REST_Security_Cheat_Sheet.html

[8] NBXplorer, Repositorio, 2019, 2021, Garage, D. https://github.com/dgarage/NBXplorer

[9] Preneel, B. (1994), Cryptographic hash functions. Eur. Trans. Telecomm., 5: 431-448. https://doi.org/10.1002/ett.4460050406

[10] Strzelewicz, A. (2014). pm2 (Nº de versión 5.1.0). Paris: Unitech:
https://pm2.keymetrics.io/docs/usage/pm2-doc-single-page/.