



Universidad  
Nacional  
de Loja

Portada

# Universidad Nacional de Loja

**Facultad de la Energía, las Industrias y los Recursos Naturales No  
Renovables**

**Carrera de Computación**

**Trazabilidad mediante blockchain en el sistema de ventas de obras de arte  
de la carrera de artes plásticas**

**Traceability through blockchain in the artwork sales system of the fine arts  
degree program.**

Trabajo de Integración  
Curricular, previa a la obtención  
del título de Ingeniero en  
Ciencias de la Computación.

**AUTOR:**

Alexander Adrián Vicente Jiménez

**DIRECTOR:**

Cristian Ramiro Narváez Guillen, Mg.Sc.

**Loja – Ecuador**

**2025**

## Certificación

Loja, 28 de enero del 2025

Cristian Ramiro Narvárez Guillen, Mg. Sc.

**DIRECTOR DEL TRABAJO DE INTEGRACION CURRICULAR**

### **CERTIFICO:**

Que he revisado y orientado todo el proceso de elaboración del Trabajo de Integración Curricular denominado: **Trazabilidad mediante Blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas**, previo a la obtención del título de **Ingeniero en ciencias de la computación**, de la autoría del estudiante **Alexander Adrián Vicente Jiménez** con cédula de identidad Nro. **1150841946**, una vez que el trabajo cumple con todos los requisitos exigidos por la Universidad Nacional de Loja, para el efecto, autorizo la presentación para la respectiva sustentación y defensa

-----  
Ing. Cristian Ramiro Narvárez Guillen, Mg. Sc.

**DIRECTOR DEL TRABAJO DE INTEGRACION CURRICULAR**

## **Autoría**

Yo, **Alexander Adrián Vicente Jiménez**, declaro ser autor del presente Trabajo de Integración Curricular y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos, de posibles reclamos y acciones legales, por el contenido del mismo. Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi Trabajo de Integración Curricular, en el Repositorio Digital Institucional – Biblioteca Virtual.

**Firma:**



**Cédula de identidad:** 1150841946

**Fecha:** 28 de enero del 2025

**Correo electrónico:** alexander.a.vicente@unl.edu.ec

**Teléfono:** 0992928724

**Carta de autorización por parte del autor, para consulta, reproducción parcial o total y/o publicación electrónica del texto completo, del Trabajo de Integración Curricular**

Yo, **Alexander Adrián Vicente Jiménez**, declaro ser autor del Trabajo de Integración Curricular denominado: **Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas**, como requisito para optar por el título de **Ingeniero en ciencias de la computación**, autorizo al sistema Bibliotecario de la Universidad Nacional de Loja para que, con fines académicos, muestre la producción intelectual de la Universidad, a través de la visibilidad de su contenido en el Repositorio Institucional.

Los usuarios pueden consultar el contenido de este trabajo en el Repositorio Institucional, en las redes de información del país y del exterior con las cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia del Trabajo de Integración Curricular que realice un tercero.

Para constancia de esta autorización suscribo, en la ciudad de Loja, a los veintiocho días del mes de enero de dos mil veinticinco.

**Firma:**



**Autor:** Alexander Adrián Vicente Jiménez

**Cédula de identidad:** 1150841946

**Dirección:** Malacatos, Loja Ecuador

**Correo electrónico:** alexander.a.vicente@unl.edu.ec

**Teléfono:** 0992928724

**DATOS COMPLEMENTARIOS:**

**Director del Trabajo de Integración Curricular:** Cristian Ramiro Narváez Guillen, Mg.Sc.

## **Dedicatoria**

Dedico este trabajo con todo mi cariño a mis padres y hermanos, quienes son mi mayor motivación y me brindaron todo su apoyo durante el transcurso de este arduo camino. También a mis amigos y compañeros, que fueron parte de mi trayectoria; gracias a las experiencias vividas junto a ellos, crecí tanto personal como profesionalmente. Finalmente, me lo dedico a mí mismo, reconociendo el esfuerzo, la perseverancia y la dedicación que he invertido en este proyecto, el cual representa el fruto de mi esfuerzo y la culminación de una etapa importante en mi vida.

*Alexander Adrián Vicente Jiménez*

## **Agradecimiento**

Quiero expresar mi más profundo agradecimiento a mis padres, cuyo esfuerzo, amor y dedicación han sido el pilar fundamental que me ha permitido alcanzar mis metas y superarme cada día. Extiendo también mi gratitud a todas las personas que, de una u otra manera, han sido importantes en este camino, incluidos amigos y compañeros, quienes con su apoyo y compañía hicieron este proceso más significativo.

Un agradecimiento especial al Ing. Cristian Ramiro Narváez Guillen, mi tutor, por ser una pieza clave en el éxito y desarrollo de este proyecto. Su guía, dedicación y valiosos consejos durante la realización de este trabajo han sido esenciales para alcanzar este logro.

Gracias a todos por ser mi inspiración y motivación constante en este recorrido.

*Alexander Adrián Vicente Jiménez*

## Índice de Contenido

<b>Portada</b> .....	<b>i</b>
<b>Certificación</b> .....	<b>ii</b>
<b>Autoría</b> .....	<b>iii</b>
<b>Carta de autorización</b> .....	<b>iv</b>
<b>Dedicatoria</b> .....	<b>v</b>
<b>Agradecimiento</b> .....	<b>vi</b>
<b>Índice de Contenido</b> .....	<b>vii</b>
<b>1. Título</b> .....	<b>1</b>
<b>2. Resumen</b> .....	<b>2</b>
Abstract.....	3
<b>3. Introducción</b> .....	<b>4</b>
<b>4. Marco teórico</b> .....	<b>7</b>
4.1. Antecedentes .....	7
4.2. Fundamentación Teórica.....	8
4.2.1. Aplicaciones Descentralizadas (dApps).....	8
4.2.2. Blockchain .....	9
4.2.3. Tipos de Blockchain.....	10
4.2.4. Ethereum .....	10
4.2.5. Proof of Stake (PoS).....	10
4.2.6. Ethereum Virtual Machine (EVM) .....	11
4.2.7. Gas.....	11
4.2.8. Contrato Inteligente.....	13
4.2.9. Non-Fungible Token (NFT).....	15
4.2.10. Key Performance Indicators (KPI) especializado para blockchain .....	15
4.2.11. Framework Django .....	16
4.2.12. Solidity .....	16
4.2.13. Truffle Framework.....	16
4.2.14. IPFS.....	17
4.2.15. ABCDE (Agile Block Chain DApp Engineering) .....	17
4.3. Trabajos Relacionados .....	22
<b>5. Metodología</b> .....	<b>24</b>
5.1. Área de estudio .....	24
5.2. Procedimiento .....	24
5.2.1. Objetivo 1: Integrar la tecnología Blockchain Ethereum el sistema actual de ventas de obras de arte mediante la metodología ABCDE (Agile Block Chain DApp Engineering).....	24

5.2.2.	Evaluar el porcentaje de trazabilidad en el proceso de venta de obras de arte a través del Indicador Clave de Rendimiento (KPI) de trazabilidad especializado para la industria blockchain .....	29
5.3.	Recursos.....	30
5.3.1.	Recursos Científicos .....	30
5.3.2.	Métodos Observacionales.....	31
5.3.3.	Recursos Técnicos .....	31
5.3.4.	Estándares.....	31
5.3.5.	Recursos de hardware y Software.....	32
5.3.6.	Recursos Humanos.....	33
<b>6.</b>	<b>Resultados.....</b>	<b>34</b>
6.1.	Objetivo 1: Integrar la tecnología Blockchain Ethereum en el sistema actual de ventas de obras de arte mediante la metodología ABCDE (Agile Blockchain DApp Engineering).....	34
6.1.1.	Determinar el proceso de creación de obras de arte.....	34
6.1.2.	Fase 1: Objetivo de la dapp .....	34
6.1.3.	Fase 2: Definir los actores de la dApp.....	35
6.1.4.	Requisitos e Historias de usuario .....	35
6.1.5.	Dividir el sistema en dos subsistemas.....	38
6.1.6.	Diseño del Subsistema de contratos inteligentes y de aplicaciones .....	39
6.1.7.	Diseño del contrato inteligente.....	42
6.1.8.	Codificación y prueba de los Smart Contracts. ....	42
6.1.9.	Diseño del subsistema de aplicaciones .....	44
6.1.10.	Código del subsistema de aplicaciones .....	48
6.1.11.	Pruebas Unitarias .....	48
6.1.12.	Integrar, Probar y desplegar la Dapp.....	49
6.2.	Objetivo 2: Evaluar el porcentaje de trazabilidad en el proceso de ventas de obras de arte a través del indicador clave de rendimiento KPI de trazabilidad especializado para la industria blockchain .....	53
6.2.1.	Selección de los activos.....	53
6.2.2.	Recopilación de datos.....	56
6.2.3.	Establecimiento de los Criterios de selección.....	58
6.2.4.	Cálculo del Porcentaje de Trazabilidad.....	59
6.2.5.	Interpretación de los resultados .....	60
<b>7.</b>	<b>Discusión .....</b>	<b>61</b>
7.1.	Primer objetivo: Integrar la tecnología Blockchain Ethereum en el sistema actual de ventas de obras de arte mediante la metodología ABCDE (Agile Blockchain DApp Engineering).....	61
7.2.	Segundo objetivo: Evaluar el porcentaje de trazabilidad en el proceso de ventas de obras de arte a través del indicador clave de rendimiento KPI de trazabilidad especializado para la industria blockchain .....	62

<b>8.</b>	<b>Conclusiones .....</b>	<b>64</b>
<b>9.</b>	<b>Recomendaciones .....</b>	<b>65</b>
<b>10.</b>	<b>Bibliografía .....</b>	<b>67</b>
<b>11.</b>	<b>Anexos .....</b>	<b>71</b>

## Índice de Tablas:

<b>Tabla 1.</b> Conversiones de Gwei a ETH.....	11
<b>Tabla 2.</b> Trabajos relacionados con el tema de investigación.....	22
<b>Tabla 3.</b> Recursos de Software.....	32
<b>Tabla 4.</b> Recursos Hardware .....	33
<b>Tabla 5.</b> Recursos Humano .....	33
<b>Tabla 6.</b> Actores de la dApp.....	35
<b>Tabla 7.</b> Requisitos funcionales de la dApp.....	36
<b>Tabla 8.</b> Requisitos no funcionales .....	37
<b>Tabla 9.</b> Resumen de las historias de usuario .....	37
<b>Tabla 10.</b> Iteraciones para el subsistema de contratos inteligentes.....	39
<b>Tabla 11.</b> Tabla de iteraciones del subsistema de aplicaciones .....	39
<b>Tabla 12.</b> Descripción de la estructura de carpetas del contrato inteligente .....	41
<b>Tabla 13.</b> Descripción estructura de aplicaciones.....	41
<b>Tabla 14.</b> Descripción de las funciones del contrato inteligente.....	43
<b>Tabla 15.</b> Descripción de las funcionalidades del subsistema de aplicaciones.....	48
<b>Tabla 16.</b> Casos de prueba definidos para el subsistema de aplicaciones.....	48
<b>Tabla 17.</b> Resultados obtenidos de la ejecución de los casos de prueba.....	49
<b>Tabla 18.</b> Resultados de las pruebas funcionales realizadas.....	50
<b>Tabla 19.</b> Costes del despliegue.....	52
<b>Tabla 20.</b> Eventos del contrato inteligente.....	56

## Índice de Figuras:

<b>Figura 1.</b> Estructura típica de una DApp .....	8
<b>Figura 2.</b> Arquitectura tradicional de aplicaciones web vs la arquitectura de una DApp [8] ...	9
<b>Figura 3.</b> Coste de gas de las operaciones en bytecode del contrato inteligente. ....	12
<b>Figura 4.</b> Ciclo de Vida de los Contratos Inteligentes. ....	13
<b>Figura 5.</b> Proceso de la metodología ABCDE .....	18
<b>Figura 6.</b> Oficina de docentes de la carrera de artes plasticas .....	24
<b>Figura 7.</b> Diagrama de actividades de la primera etapa del Trabajo de Integración curricular de Integración Curricular .....	25
<b>Figura 8.</b> Diagrama de actividades de la segunda etapa del desarrollo de la aplicación descentralizada.....	27
<b>Figura 9.</b> Proceso de creacion de obras de arte.....	34
<b>Figura 10.</b> Diagrama de Casos de Uso (SC-SDVOA: Smart contract para el sistema de difusion y ventas de obras de arte).....	36
<b>Figura 11.</b> Arquitectura simplificada de la dapp.....	38
<b>Figura 12.</b> Control de versiones mediante la herramienta de versionamiento Github.....	40
<b>Figura 13.</b> Estructura de carpetas del contrato inteligente.....	40
<b>Figura 14.</b> Estructura de carpetas del subsistema de aplicaciones.....	41
<b>Figura 15.</b> Diagrama de clases del contrato inteligente .....	42
<b>Figura 16.</b> Principales funciones del contrato inteligente.....	43
<b>Figura 17.</b> Ejecución de pruebas unitarias mediante la herramienta truffle .....	44
<b>Figura 18.</b> Arquitectura completa de la aplicación descentralizada .....	45
<b>Figura 19.</b> Diagrama de secuencia de la dApp .....	46
<b>Figura 20.</b> Patrón de diseño MVT (Model, Views, Templates) .....	46
<b>Figura 21.</b> Arquitectura física de la DApp.....	47
<b>Figura 22.</b> Funcionalidades del subsistema de aplicaciones.....	48
<b>Figura 23.</b> Resultado de las pruebas unitarias ejecutadas .....	49
<b>Figura 24.</b> Despliegue del contrato inteligente en la tesnet de polygon .....	51
<b>Figura 25.</b> Contrato inteligente desplegado en la tesnet Polygon zkEVM Cardona.....	52
<b>Figura 26.</b> Infraestructura de despliegue de la dApp .....	52
<b>Figura 27.</b> Proveedor Alchemy .....	53
<b>Figura 28.</b> Servicios desplegados en Render Clud .....	53

<b>Figura 29.</b> Procesos de prueba registrados en la cadena de bloques por el primer responsable .....	54
<b>Figura 30.</b> Procesos de prueba registrados en la cadena de bloques por parte del segundo responsable.....	54
<b>Figura 31.</b> Obras de arte resultado de los 13 procesos completados. ....	55
<b>Figura 32.</b> Obras de arte en su representacion como NFT en la cadena de bloques.....	55
<b>Figura 33.</b> Transacciones de compra realizadas sobre los NFT .....	56
<b>Figura 34.</b> Parte de los eventos emitidos por el contrato inteligente .....	57
<b>Figura 35.</b> Función para obtener todos los eventos emitidos por su nombre.....	57
<b>Figura 36.</b> Función para filtrar los datos en base a los criterios de selección.....	57
<b>Figura 37.</b> Descripcion de cada obra de arte rastreada con exito junto con los criterios de seleccion.....	58
<b>Figura 38.</b> Conteo de las obras registradas en la cadena de bloques y los procesos.....	58
<b>Figura 39.</b> Función para calcular el porcentaje de trazabilidad .....	59
<b>Figura 40.</b> Porcentaje de trazabilidad obtenido durante la evaluacion. ....	60

## Índice de Anexos:

<b>Anexo 1.</b> Estudio realizado por Hiscox online Art a compradoras de arte en línea .....	71
<b>Anexo 2.</b> Entrevista realizada al docente de la carrera de artes plásticas .....	72
<b>Anexo 3.</b> Diagrama de casos de uso de la plataforma de difusión y venta de obras de arte ..	75
<b>Anexo 4.</b> Especificación de requisitos IEE 830.....	76
<b>Anexo 5.</b> Historias de Usuario.....	93
<b>Anexo 6.</b> Diagrama de clases del contrato inteligente.....	96
<b>Anexo 7.</b> Código fuente del contrato inteligente .....	97
<b>Anexo 8.</b> Plan de pruebas unitarias para el contrato inteligente SDVOA .....	104
<b>Anexo 9.</b> Arquitectura del sistema de difusión y ventas de obras de arte .....	117
<b>Anexo 10.</b> Plan de pruebas unitarias para el subsistema de aplicaciones.....	118
<b>Anexo 11.</b> Plan de pruebas de Integración de la DApp.....	124
<b>Anexo 12.</b> Plan de pruebas Funcionales para la DApp .....	129
<b>Anexo 13.</b> Contrato de confidencialidad .....	149
<b>Anexo 14.</b> Certificado de traducción de resumen.....	153

## **1. Título**

**Trazabilidad mediante Blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas.**

**Traceability through blockchain in the artwork sales system of the fine arts degree program.**

## 2. Resumen

El mercado del arte en línea en los últimos años ha experimentado un crecimiento exponencial que ha permitido a los artistas exhibir y comercializar sus obras de arte a través de diversas plataformas digitales. Sin embargo, existe cierta desconfianza por parte de los compradores al adquirir piezas a través de estos medios debido a la poca información sobre su procedencia y autenticidad. Por tal motivo, se propuso implementar la tecnología blockchain en el sistema de difusión y ventas de obras de artes de la carrera de artes plásticas de la Universidad Nacional de Loja con la finalidad de proporcionar trazabilidad a cada obra de arte. Para cumplir con este objetivo, se utilizó la metodología Agile Block chain Dapp Engineering (ABCDE) que permitió mediante un proceso sistemático desarrollar los componentes necesarios como el contrato inteligente, para la integración de la tecnología en el sistema existente. Además, se llevó a cabo una evaluación del porcentaje de trazabilidad alcanzable mediante la adopción de la tecnología, utilizando el indicador clave de rendimiento (KPI) para su medición. Los resultados experimentales obtenidos indican cada obra de arte fue rastreada con éxito desde su creación hasta su venta, alcanzando un porcentaje de trazabilidad del 100 %, lo que significa, que la información registrada en la red de pruebas de Polygon zkEVM Cardona fue precisa y completa, proporcionando así un historial verificable y detallado sobre su procedencia. Finalmente, se logra determinar que la implementación de la tecnología blockchain en el sistema de difusión y ventas de obras de arte puede ofrecer una solución integral, aprovechando sus características inherentes como la descentralización, la inmutabilidad y la transparencia para proporcionar trazabilidad a las obras de arte.

**Palabras Clave:** Blockchain, Trazabilidad, NFT, obra de arte, aplicación descentralizada

## **Abstract**

In recent years, the online art market has experienced exponential growth, allowing artists to exhibit and market their works of art through a multitude of digital platforms. Nevertheless, prospective purchasers may be reluctant to acquire works through these channels due to the lack of information regarding their provenance and authenticity. In light of these considerations, it was proposed to implement blockchain technology in the system of dissemination and sales of works of art in the field of plastic arts at the National University of Loja with the objective of providing traceability to each work of art. In pursuit of our objective, we utilized the Agile Block Chain Dapp Engineering (ABCDE) methodology to systematically develop crucial components, including a smart contract, for integrating the technology into our existing system. Additionally, we evaluated to gauge the achievable level of traceability through the technology implementation, utilizing key performance indicators (KPIs) as evaluation metrics. The experimental results obtained indicate that each work of art was successfully tracked from its from its creation to its sale, reaching a percentage of 100 % traceability, which means that the information 100% traceability, which means that the information recorded in the Polygon zkEVM Cardona's Polygon zkEVM Cardona's testing network was accurate and complete, thus providing a verifiable and detailed verifiable and detailed history of its provenance. In conclusion, it has been established that incorporating blockchain technology into the state-of-the-art distribution and sales system can provide a complete solution. This is made possible by leveraging its fundamental characteristics, such as decentralization, immutability, and transparency, to ensure the trackability of art pieces.

**Keywords:** blockchain, traceability, NFT, artwork, decentralized app.

### 3. Introducción

Según el Art Trade Report de Hiscox, el mercado de arte en línea en la última década ha aumentado significativamente, en mayor parte gracias a la adopción de las plataformas digitales por parte de vendedores y compradores de arte. Estos marketplaces o galerías en línea han representado un beneficio para los artistas permitiéndoles exhibir sus obras a un público general. Sin embargo, la confianza de los compradores al adquirir obras mediante estas plataformas se ve comprometida por problemas relacionados por la poca o nula información que se proporciona sobre la procedencia y autenticidad de las obras [1].

Ante estos problemas, Hiscox propone el uso de tecnologías emergentes como la tecnología blockchain para abordar los problemas del mercado de arte en línea. Esta solución no es solo propuesta por Hiscox, diversos investigadores han centrado su estudio en el uso de la tecnología blockchain como una forma de garantizar la autenticidad, transparencia y trazabilidad de las obras de arte. Tal es el caso de Artchain, una plataforma digital diseñada mediante la tecnología blockchain para mejorar aspectos como la transparencia, trazabilidad y autenticidad de las obras de arte. Además, de proporcionar el seguimiento en tiempo real de cada obra asegurando su procedencia y propiedad. Mientras que, por otro lado, ManiaticNFT utiliza los Tokens No Fungibles (NFT) para representar, comprar y vender las obras de arte permitiendo que cada transacción sobre los NFT se registre en la cadena de bloques. Proporcionando de esta manera un historial verificable que garantiza la autenticidad y propiedad de las obras protegiéndolas así de falsificaciones. Por último, NFT Marketplace asegura la propiedad y procedencia de las obras de arte mediante el uso de los NFT y beneficiándose de las características inherentes de la blockchain como la inmutabilidad garantizando que cada NFT creado junto con su historial y propiedad no puedan ser modificados. Además, hace uso de contratos inteligentes para gestionar y automatizar diversas operaciones dentro de la plataforma como la creación, transferencia y gestión de propiedad de los NFT eliminando la necesidad de intermediarios y generando un entorno de confianza al eliminar el riesgo de un único punto de fallo asociado con los proveedores de información centralizados [1].

Como parte de un trabajo titulación en la Universidad Nacional de Loja se llevó a cabo el desarrollo e implementación de una plataforma digital para la difusión y venta de las obras de arte creadas por los estudiantes de la carrera de artes plásticas [2]. Misma que forma parte de las diversas plataformas digitales creadas con este propósito, por lo que en un futuro no lejano los problemas arraigados por el mercado del arte en línea pueden afectar negativamente

a la plataforma. Por tal motivo surgió la necesidad de implementar soluciones tempranas, tomando para ello como punto de partida y objeto de estudio el uso de la tecnología blockchain junto con los NFT para abordar los problemas mencionados.

Por lo antes mencionado, la ejecución del presente Trabajo de Integración Curricular denominado “Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas”, constituye un aporte fundamental para el mercado de arte en línea y por ende a la plataforma de difusión y ventas de obras de arte al proporcionar la capacidad de realizar un seguimiento a cada obra de arte desde su origen hasta los diferentes propietarios asegurando su procedencia y autenticidad, por consiguiente se ha abordado la siguiente pregunta de investigación “¿Qué porcentaje de trazabilidad se puede alcanzar en el proceso de venta de obras de arte al implementar la tecnología de blockchain en el sistema de venta de obras de arte de la carrera de Artes Plásticas durante el periodo 2024-2025?”

Para dar respuesta a la pregunta planteada se propuso los siguientes objetivos: Integrar la tecnología Blockchain Ethereum el sistema actual de ventas de obras de arte mediante la metodología ABCDE (Agile Block Chain DApp Engineering) y Evaluar el porcentaje de trazabilidad en el proceso de venta de obras de arte a través del Indicador Clave de Rendimiento (KPI) de trazabilidad especializado para la industria blockchain

El uso de la tecnología blockchain convirtió la plataforma de una aplicación web tradicional a una aplicación descentralizada (dApp) diseñada específicamente para permitir el seguimiento (trazabilidad) de cada obra de arte, mediante el registro de cada etapa de su proceso de creación, así como las diferentes transacciones que se realizan sobre la misma generando un historial completo y detallado que puede ser verificado por cualquier usuario. Por otro lado, mediante el uso del contrato inteligente para la gestión de cada obra mediante roles asegura que la información registrada en la blockchain sea precisa y fiable.

Entre las limitantes que surgieron durante el desarrollo de la dApp se encuentran el rediseño de la aplicación para pasar de un enfoque centralizado a un descentralizado teniendo que eliminar y rediseñar componentes para permitir la interacción entre la blockchain y la aplicación. Por otro lado, el utilizar la librería web3 represento complejidad por la forma en que se realizan las transacciones, teniendo como requisito principal la clave privada de la billetera digital (wallet) del usuario para firmar y validar las transacciones. Manejar este tipo de información privada dentro de la aplicación puede generar desconfianza por parte del usuario. Por último, encontrar una red de pruebas (testnet) apropiada fue un desafío, debido a que algunas de las redes de pruebas más conocidas han dejado de funcionar, y las que se

encuentran disponibles requieren de ciertos de requisitos para proporcionar ETH de prueba y probar el funcionamiento de la dApp.

## 4. Marco teórico

### 4.1. Antecedentes

En 2023 Hiscox presento su reporte anual sobre el estado del mercado de arte en línea, en el cual indica que el mercado de arte en línea en los últimos diez años ha crecido un 587% representando grandes oportunidades para las casas de subastas y galerías. Como se desprende del informe, gran parte de su crecimiento se debe a la incursión de los Tokens No Fungibles (NFT) y la alta actividad durante la pandemia, el reporte menciona que en 2013 solo un 38% de los compradores se animaron a adquirir obras a través de internet, mientras que para el 2023 el porcentaje alcanzo un 78%, que es un indicador de gran avance de la revolución digital en el mundo del arte [3].

Los resultados de las encuestas realizadas a distintas plataformas digitales, revela que el mercado del arte en línea aún se enfrenta a barreras que afectan significativamente a la decisión de los compradores para adquirir arte a través de internet. En términos porcentuales el reporte indica que el 70% de los compradores tienen temor a adquirir obras que no son auténticas y que no sean lo que pretenden ser, mientras que un 51% indica que la falta de información sobre la procedencia de las obras genera desconfianza. Aspecto que para 90 % de nuevos compradores y el 89% de jóvenes compradores es fundamental ya que la capacidad de rastrear la procedencia e historia de una obra de arte aumentaría su confianza de adquirir obras a través de internet. Como se desprende del informe, el uso de los NFT y la blockchain pueden ofrecer soluciones a largo plazo para abordar estas preocupaciones.

Estas soluciones ya han sido tomadas en cuenta por diversos investigadores, como es el caso de José Miguel Puccinelli quien en su estudio "El impacto de la Blockchain y los NFTs en el mundo del arte", menciona que el uso de la tecnología blockchain puede influir significativamente en la autenticidad, propiedad, y procedencia de cada obra de arte tanto física como digital. Por otro lado, Dhruv Ajay Patil y otros autores, en su investigación "An End to End Blockchain based Non-Fungible Token Platform for Buying and Selling Digital Arts [4]", destacan que en el ámbito artístico la clave de la implementación de la tecnología blockchain se encuentra en el uso de los Tokens No Fungibles para representar digitalmente diversas formas artísticas como las obras de arte para garantizar su autenticidad mediante una firma digital.

Por todo lo mencionado anteriormente, el presente Trabajo de Integración Curricular surge como una propuesta para la implementación de la tecnología blockchain en la plataforma digital de difusión y ventas de obras de arte (SIAAF Ecommerce) de la carrera de artes plásticas

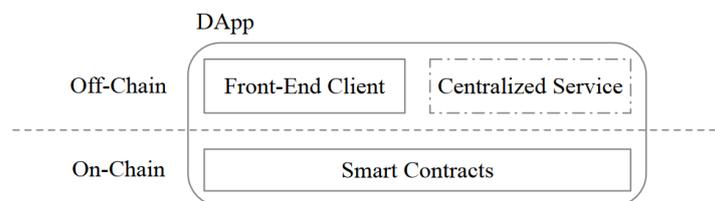
con el objetivo de proporcionar trazabilidad a cada obra de arte proporcionando un historial completo y verificable de su procedencia.

## 4.2. Fundamentación Teórica

### 4.2.1. Aplicaciones Descentralizadas (dApps)

Las aplicaciones descentralizadas o también conocidas como dApps, son aplicaciones que se ejecutan sobre una red distribuida y descentralizada como blockchain. A menudo, se construyen sobre la plataforma de Ethereum con distintos propósitos como el comercio, juegos, finanzas, redes sociales [5]. La principal diferencia con las aplicaciones tradicionales, es que no están controladas por ninguna entidad central fomentando un entorno de confianza [6].

En la **Figura 1** se presenta las capas que componen la estructura típica de una dApp, estas son: On-Chain (Dentro de la cadena) y Off-Chain (fuera de la cadena) [7] .



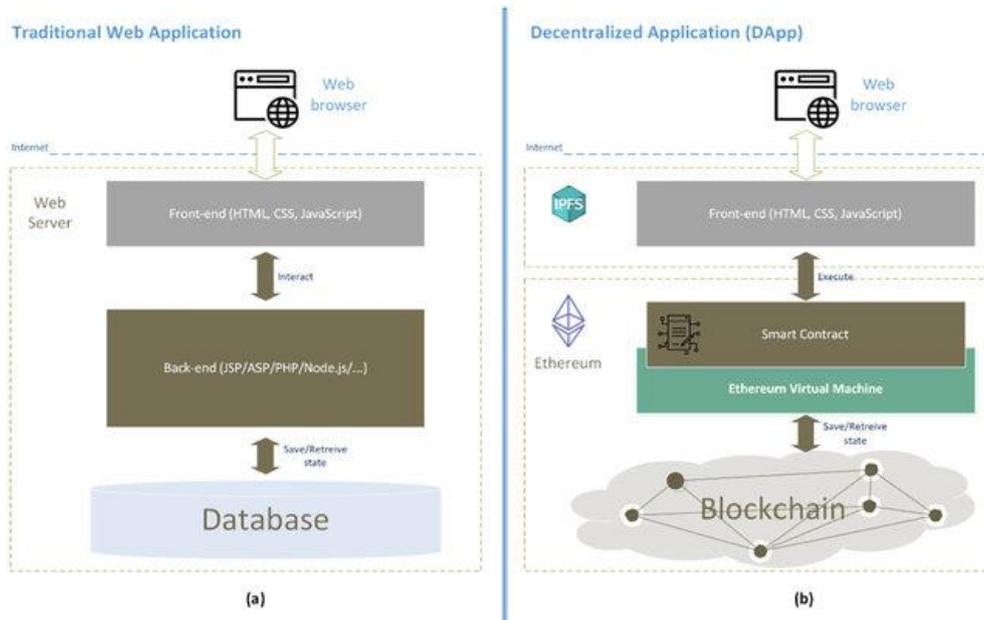
**Figura 1.** Estructura típica de una DApp

La capa inferior (On-Chain), se conforma de uno o varios Smart Contracts (Contratos Inteligentes) que son el núcleo de la dApp y tienen la función de almacenar datos cruciales en la cadena de bloques. Mientras que la capa superior (Off-Chain), se conforma de la Interfaz de Usuario para la interacción entre el usuario y la cadena de bloques, junto con un servicio descentralizado como una base de datos.

#### 4.2.1.1. Aplicación Web Tradicional vs Aplicación Descentralizada (DApp)

La mayoría de aplicaciones tradicionales se componen de una base de datos para almacenar la información, un módulo backend que interactúa con la base de datos y contiene toda la lógica de negocio y un módulo frontend con la interfaz de usuario (**Figura 2.a**). Sin embargo, esta arquitectura representa un único punto de fallo: si el servidor falla toda la aplicación se volverá inaccesible. Además, la única parte visible de la aplicación es el frontend, mientras que la base de datos permanece oculta y se vuelve susceptible a manipulaciones por parte del propietario, aumentando el riesgo de ataques de terceros o que los datos sean manipulados intencionalmente. Por otro lado, las dApps ofrecen una arquitectura descentralizada y autónoma que permite el desarrollo de aplicaciones totalmente visibles para los usuarios y garantiza la seguridad e inmutabilidad de los datos (**Figura 2.b**). En esta

arquitectura los datos se almacenan en la cadena bloques, la lógica de la aplicación se desarrolla mediante los contratos inteligentes que se registran y administran en la cadena de bloques. Los contratos inteligentes actúan como el backend de la aplicación y para poder interactuar con ellos, el usuario debe hacer uso de un nodo. El componente encargado de ejecutar el código del contrato es la EVM (Ethereum Virtual Machine). Al estar desarrolladas sobre la plataforma de Ethereum, se beneficia de la criptomoneda denominada ETH, utilizada para crear aplicaciones financieras [8].



**Figura 2.** Arquitectura tradicional de aplicaciones web vs la arquitectura de una DApp [8]

#### 4.2.2. Blockchain

La cadena de bloques, también conocida como blockchain, es un libro de contabilidad digital global que es constantemente actualizada por múltiples usuarios y se caracteriza por proporcionar se transparente e imposible de corromper [9]. El objetivo principal es eliminar la necesidad de confianza entre dos partes durante una transacción, y eliminar intermediarios que validen dicha transacción [2][10]

Su estructura se compone de una secuencia de “bloques” ordenados cronológicamente, cada bloque está conectado criptográficamente al anterior garantizando la inmutabilidad de la cadena, lo que significa que ningún elemento puede alterar el contenido de los bloques. Para registrar la información en los bloques esta debe pasar primero por un proceso de validación [11]. Cada transacción realizada sobre la cadena de bloques puede ser verificada con el tiempo garantizando la seguridad y confianza en los datos almacenados. Además, facilita la ejecución de acciones sin la necesidad de un intermediario (como un banco) reduciendo costos de transacción. [12].

Su sistema de registro distribuido y descentralizado permite la gestión de una variedad de activos como certificados, unidades de valor, derechos, propiedades, etc. Los nodos participantes de la red validan estos activos, creando un libro contable que se replica en cada nodo, proporcionando un sistema seguro y confiable para diversas aplicaciones[13] . Aunque principalmente la cadena de bloque se creó como un medio para almacenar las transacciones de Bitcoin, con el tiempo ha sido aplicada en otros sectores por su gran potencial [14].

### **4.2.3. Tipos de Blockchain**

#### **4.2.3.1. Pública**

Una red pública de blockchain se caracteriza por ser una red sin permisos, es decir cualquier usuario puede unirse y convertirse en parte de la red, crear nuevos bloques y participar en el proceso de validación de bloques. El componente principal de esta red es la minería, que combina incentivos económicos y la verificación criptográfica mediante mecanismos como Proof of Work (PoW) o Proof of Stake (PoS), para garantizar la confianza en la red. Ethereum es ejemplo de este tipo de red [15].

#### **4.2.4. Ethereum**

Ethereum es una plataforma basada en la tecnología blockchain, su propósito es permitir y facilitar a los programadores crear y desplegar aplicaciones descentralizadas (DApps) que tiene como base los Smart Contracts o contratos inteligentes[16]

La moneda nativa de Ethereum se llama Ether (ETH), utilizada para realizar transacciones y pagos simples entre los usuarios de la blockchain de Ethereum. A diferencia del Bitcoin (BTC), no solo sirve como medio de pago, sino que es el combustible para la red de Ethereum. Es decir, el Ether es necesario para desplegar y ejecutar los contratos Inteligentes. Además, es el incentivo que motiva a los desarrolladores para crear aplicaciones descentralizadas de calidad, y es la recompensa que reciben los mineros por ir agregando bloques a la cadena. El algoritmo de consenso utilizado en Ethereum es el llamado “Prueba de participación” o “Proof of Stake” [17].

#### **4.2.5. Proof of Stake (PoS)**

El algoritmo de consenso Proof of Stake (PoS), o Prueba de Participación, se basa en la idea de que los nodos que desean participar en el proceso de creación de bloques en primer lugar deben demostrar que poseen cierta cantidad de monedas o tokens. Además, deben bloquear una cantidad de sus monedas, llamado stake, en una cuenta de depósito en garantía para participar en el proceso de creación de bloques. El stake actúa como garantía de que el nodo se comportará según las reglas del protocolo [18].

#### 4.2.6. Ethereum Virtual Machine (EVM)

La Ethereum Virtual Machine (EVM), o máquina virtual de Ethereum es el entorno de ejecución de los contratos inteligentes en la red Ethereum. La EVM cumple un papel crucial en la seguridad de la red al prevenir ataques, manejando el acceso a los recursos y limitando sus acciones a la EVM [19].

La EVM, despliega contratos inteligentes a través de la interpretación de bytecode generado a partir de compilar los códigos de operación (OP\_CODES) escritos en un lenguaje programación de alto nivel denominado Solidity. El bytecode, está diseñado para ser Turing completo, que permite ejecutar prácticamente cualquier algoritmo o contrato inteligente que se le suministre, siempre y cuando exista suficiente gas disponible para completar la ejecución [20].

#### 4.2.7. Gas

Sobre la red Ethereum se puede realizar dos tipos de transacciones: transferir Ether, similar a Bitcoin, y la ejecución o creación de contratos inteligentes [21]. En la ejecución de los contratos inteligentes, cada nodo debe procesar el código asociado al contrato para determinar el resultado final de la transacción. Debido a que la EVM es un Turing Completo, puede provocar problemas de sobrecarga, bucles infinitos o fallos de ejecución por bugs en el contrato [22].

Para evitar estos riesgos, se introdujo el mecanismo de tarifa de red (network fee) o gas. El gas, se utiliza para medir el esfuerzo computacional y determinar el costo asociado a este esfuerzo en la red, en pocas palabras, el gas es necesario para llevar a cabo una transacción [23]. En la red Ethereum, el gas se expresa en giga-wei (gwei) que representa la cantidad de ETH necesaria para realizar una transacción.

En la **Tabla 1**, se presenta el valor en ETH del gas en sus diferentes unidades.

**Tabla 1.** Conversiones de Gwei a ETH

Unidad	Valor en Wei	Wei	Valor en ETH
Wei	1 wei	1	0.000000000000000001
Kwei (babbage)	1e3 wei	1,000	0,000000000000001
Mwei (lovelace)	1e6 wei	1,000,000	0,00000000001
Gwei (shannon)	1e9 wei	1,000,000,000	0,00000001
Microether(szabo)	1e12 wei	1,000,000,000,000	0,000001
miliether (finney)	1e15 wei	1,000,000,000,000,000	0,001
Ether	1e18 wei	1,000,000,000,000,000,000	1

El mecanismo de gas funciona de la siguiente manera:

1. El gas es una medida abstracta que indica la cantidad de Ethers (ETH) que deben pagarse de acuerdo con el esfuerzo [24].
2. Cuando se inicia una transacción, el remitente puede especificar dos parámetros [21]:
  - **Gas Price:** Expresado en gwei, indica la cantidad de Ether que el remitente está dispuesto a pagar por unidad de gas.
  - **Gas Limit:** Representa el máximo de gas que el remitente está dispuesto a pagar por la transacción
3. El costo real del gas producido por la transacción debe ser igual o menor al límite de gas, para dar por válida la transacción [23]. El nodo minero que lleva a cabo la transacción acumula el gas consumido, y si excede el límite, la transacción se cancela, pero el minero recibe sus honorarios por el servicio computacional prestado [22].
4. El precio mínimo en una transacción normal es de 21000 gas (0,000021 ETH [22]. Para calcular el costo de gas consumido en las transacciones donde se ejecuta código de algún contrato inteligente, se toma en cuenta el bytecode de cada operación ejecutada [21], [23]. En la **Figura 3** se presenta el coste de cada operación de bytecode.
5. Los nodos mineros ordenan las transacciones validas de mayor a menor tomando en cuenta el gas ofrecido por las transacciones. Por lo que, las transacciones con un precio de gas alto serán las primeras en incorporarse a la cadena, mientras que la de un coste menor se irán incorporando progresivamente hasta alcanzar la capacidad máxima del bloque que es de 12.5 millones de Gas.

Operation	Gas	Description
ADD/SUB	3	Arithmetic operation
MUL/DIV	5	Arithmetic operation
ADDMOD/MULMOD	8	Arithmetic operation
AND/OR/XOR	3	Bitwise logic operation
LT/GT/SLT/SGT/EQ	3	Comparison operation
POP	2	Stack operation
PUSH/DUP/SWAP	3	Stack operation
MLOAD/MSTORE	3	Memory operation
JUMP	8	Unconditional jump
JUMPI	10	Conditional jump
SLOAD	200	Storage operation
SSTORE	5,000/20,000	Storage operation
BALANCE	400	Get balance of an account
CREATE	32,000	Create a new account using CREATE
CALL	25,000	Create a new account using CALL

**Figura 3.** Coste de gas de las operaciones en bytecode del contrato inteligente.

#### 4.2.8. Contrato Inteligente

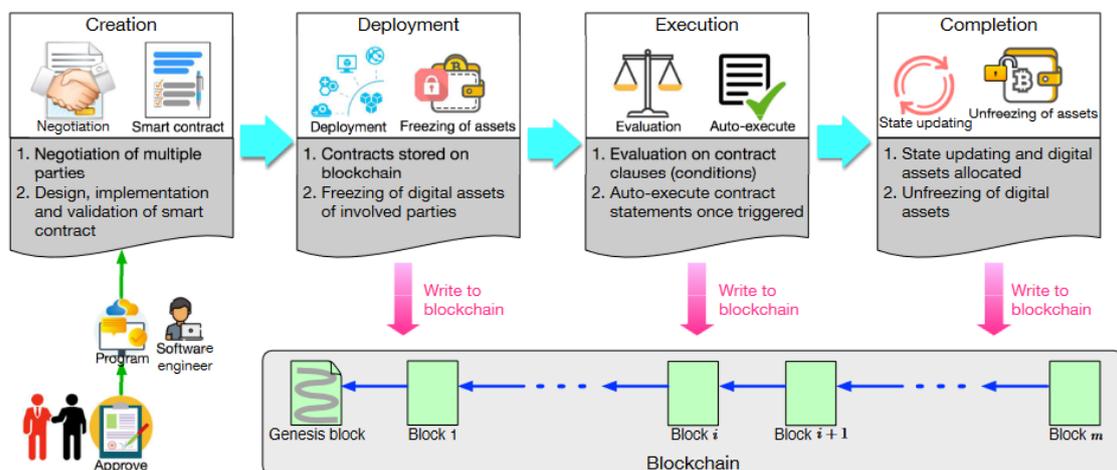
Un contrato inteligente o Smart Contract, es un programa autoejecutable que automatiza la ejecución de las acciones especificadas en un acuerdo, y a la vez se almacena, replica y actualiza en la cadena de bloques [25]. El código descrito en el contrato inteligente representa el acuerdo entre dos o más partes o puede complementar un contrato convencional en texto implementando ciertas disposiciones, como la forma de transferir dinero de una persona a otra [26].

El código ejecutara las acciones correspondientes una vez que las partes han cumplido con las condiciones predefinidas para realizar una transacción [27]. De esta manera se elimina la necesidad de intermediarios de confianza o una autoridad central que supervisen o validen la transacción [28].

##### 4.2.8.1. Ciclo de Vida de un Contrato Inteligente

Los contratos inteligentes debido a su naturaleza permiten realizar transacciones de manera segura y transparente, siendo útiles para una variedad de cosas como por ejemplo la automatización de procesos de transacción, la gestión de acuerdos entre usuarios y la transferencia de activos[29].

En la **Figura 4** se ilustra las 4 etapas del ciclo de vida de los contratos inteligentes.



**Figura 4.** Ciclo de Vida de los Contratos Inteligentes.

De manera más detallada a continuación se describen cada una de las etapas del ciclo de vida del contrato inteligente.

- **Creación del contrato inteligente:** Durante esta primera etapa las partes interesadas negocian los términos del contrato y definen las condiciones para la transferencia de activos digitales. Posterior los acuerdos escritos se convierten código mediante el uso de lenguajes de programación como solidity. Esto requiere de un diseño, implementación y validación previa [30].
- **Despliegue del contrato inteligente:** Consiste en desplegar los contratos inteligentes en la cadena de bloques. Durante el despliegue los algoritmos de consenso validan los contratos convirtiéndolos en una parte inmutable de la red, por lo que no pueden modificarse y si esto llega a ocurrir requiere que el contrato se cree nuevamente. Todas las partes pueden acceder a contratos mediante la cadena de bloques. Además, los activos digitales de ambas partes se bloquean mediante el congelamiento de las billeteras digitales se bloquean para garantizar que los activos no se transfieran mientras no se cumplan las condiciones establecidas [30].
- **Ejecución del contrato inteligente:** Una vez desplegados los contratos inteligentes y que las condiciones del contrato se cumplan, las funciones se ejecutaran automáticamente. En este punto es importante destacar que cada transacción es ejecutada y validada por los nodos mineros, es a partir de ello que los estados y transacciones confirmadas se almacenaran en la cadena de bloques [30].
- **Finalización de los contratos inteligentes:** Después de la ejecución del contrato inteligente, los nuevos estados de todas las partes involucradas se actualizan. Como consecuencia, los estados actualizados y las transacciones durante la ejecución del contrato se almacenan en la cadena de bloques. Mientras que, los activos digitales se desbloquean finalizando el ciclo de vida de los contratos inteligentes. En estas 3 últimas etapas, despliegue, ejecución y finalización se ejecutan una serie de transacciones que se almacenan en la cadena de bloques y por lo tanto requieren escribir datos en la cadena de bloques [30].

#### **4.2.9. Non-Fungible Token (NFT)**

Un Token No Fungible (NFT) consiste en un token que representa un activo digital o físico (obras de arte, fotografías, certificados, música, audio, etc.) en una cadena de bloques. Este token es criptográficamente único, indivisible, irremplazable y verificable [31]. Además, permite que el activo pueda ponerse en circulación y transferirse simplemente reasignando la cuenta del propietario.

La diferencia entre un token fungible y uno no fungible radica en su capacidad de intercambio. Un token no fungible puede ser intercambiado por una moneda, como el dólar o euro, es decir, puede ser canjeado por otro bien material o inmaterial de igual valor. En cambio, los tokens no fungibles, representan una bien inmaterial digital único creados por un autor. Estos bienes están almacenados criptográficamente en una cadena de bloques y utiliza contratos inteligentes para vincular de manera inmutable la propiedad digital y física, sirviendo como prueba de propiedad [32].

En pocas palabras, los NFT detallan el bien adquirido y crean certificados de autenticidad que le otorga a una persona la propiedad de un activo en particular. Lo certificados puede transformar las obras de arte, que suelen ser varias y difíciles de controlar, en productos únicos y rastreables [33]. Permitiendo a los artistas digitales, vender su trabajo de manera segura [34].

##### **4.2.9.1. Estándar ERC – 721**

El estándar ERC-721, permite a cualquier persona crear tokens no fungibles (NFT) en la blockchain de Ethereum. Proporciona las funcionalidades básicas para el seguimiento y transferencia de NFTs mediante su API de contratos inteligentes [35]. Registra las transacciones realizadas por los propietarios de los tokens y por terceros autorizados (corredores, billeteras o subastadores), llamados operadores. También, permite que estas aplicaciones intermediarias trabajar con cualquier NFT, incluyendo la capacidad de implementar contratos inteligentes simples y aquellos que gestionan gran cantidad de NFTs [36].

#### **4.2.10. Key Performance Indicators (KPI) especializado para blockchain**

Los indicadores clave de rendimiento o Key Performance Indicators (KPI) en inglés, hacen referencia a las métricas específicas que se usan para evaluar y medir el rendimiento y éxito de las soluciones basadas en la tecnología blockchain en una organización o industria.

##### **4.2.10.1. Trazabilidad**

Una de las métricas es la trazabilidad, que evalúa la capacidad de una solución para rastrear y registrar de manera precisa el movimiento de la información y de los activos [37].

Permite a las organizaciones mantener un registro detallado y auditable de las operaciones garantizando la responsabilidad y ayuda para minimizar los riesgos asociados con fraudes o errores.

La fórmula de la trazabilidad es la siguiente:

$$\text{Porcentaje de trazabilidad} = \frac{\text{Numero de activos rastreados con éxito}}{\text{Numero total de activos}} * 100$$

Esta fórmula expresa el porcentaje de activos rastreados con éxito en relación con el número total de activos.

#### **4.2.11. Framework Django**

Django es un marco de trabajo que esta creado sobre el lenguaje de programación Python, y trabaja bajo el concepto (DRY) “Don’t Repeat Yourself”, que incita a los desarrolladores a evitar escribir código redundante, y crear código preciso y fácil de mantener [38]. Se destaca principalmente por ofrecer a los desarrolladores una solución sólida y confiable para la creación de aplicaciones web de alto nivel modernas dinámicas, seguras y escalables [39]. El patrón arquitectónico utilizado en este marco de trabajo es el Model – View – Controller (MVC) que asegura que las responsabilidades de la aplicación se mantengan separadas y mejora la organización y mantenimiento del código permitiendo a los desarrolladores centrarse en componentes específicos de la aplicación [40].

#### **4.2.12. Solidity**

Solidity es un lenguaje de programación que mezcla convenciones de redes, lenguaje ensamblador y desarrollo web para la creación de contratos inteligentes sobre la plataforma de Ethereum [41]. Se basa en los lenguajes de programación Python, C++, JavaScript y se caracteriza por ser un lenguaje tipado estáticamente, es decir, las variables se declaran explícitamente y se determinan en tiempo de compilación, esta característica apunta a mejorar la ejecución de los contratos inteligentes en la Ethereum Virtual Machine (EVM)[42]. A través de solidity se pueden crear contratos inteligentes para el intercambio de criptomonedas, la gestión de propiedad de activos digitales como el ERC-721 y otro tipo de transacciones. No está diseñado para construir aplicaciones tradicionales, sino que es un lenguaje específicamente creado para construir aplicaciones descentralizadas [43].

#### **4.2.13. Truffle Framework**

Truffle proporciona una serie de herramientas para la creación, implementación y pruebas de los contratos inteligentes. Es compatible con solidity y es el framework más utilizado para construir dApps [44]. Entre las herramientas para la construcción de aplicaciones

descentralizadas se encuentra ganache, una red personal de Ethereum que permite la prueba de los contratos inteligentes sin necesidad de gastar Ether real [45] .

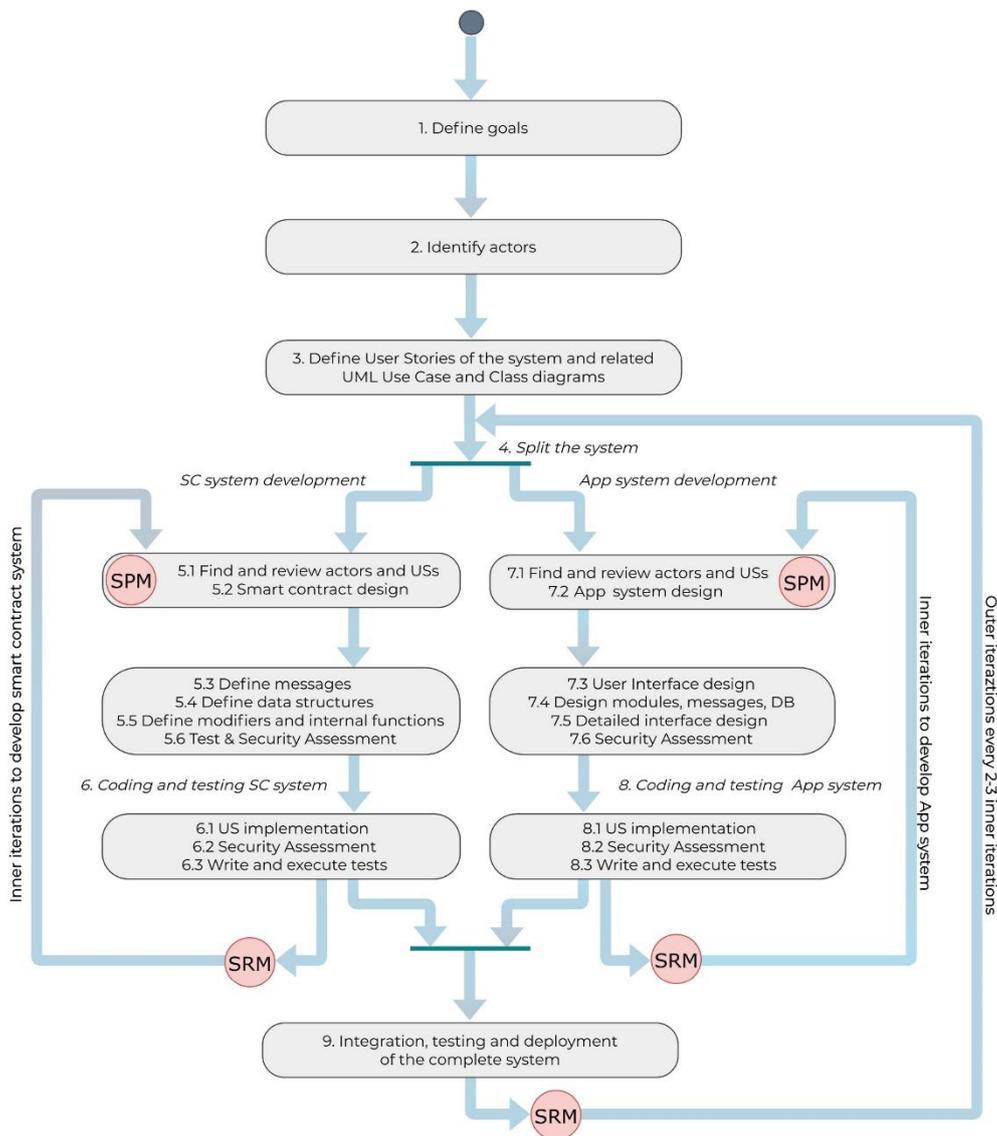
#### **4.2.14. IPFS**

El sistema de archivos interplanetario (IPFS) es un sistema de distribución de archivos creado por Juan Benet que opera sobre una red punto a punto (P2P) para gestionar el contenido en internet [46]. A diferencia del almacenamiento en servidores centralizados, IPFS divide los archivos en bloques y luego los distribuye en una red de nodos. A cada archivo almacenado le asigna un hash único dependiendo de su contenido, por lo que si el contenido cambia el hash también lo hará, garantizando la autenticidad e integridad de los datos [47]. El hash también actúa como identificador permitiendo a los usuarios acceder al contenido del archivo mediante su hash. Entre las propiedades del IPFS, se encuentra el uso de tablas hash distribuidas (DHT) que son una estructura de datos que permite buscar y encontrar eficientemente los archivos en la red descentralizada [44] .

La naturaleza de la blockchain, no le permite almacenar gran cantidad de información debido a problemas como el costo y la escalabilidad [47]. Por lo que, las dApps y otros proyectos basados en la tecnología blockchain se benefician del uso de IPFS para gestionar de manera eficiente y segura gran cantidad de información. La combinación de IPFS y la blockchain, elimina puntos de falla únicos presentes en la dependencia de servidores centralizados, mejorando la resistencia ante la censura e interrupción de servicios. Una de las aplicaciones de IPFS es el desarrollo de mercados de NFTs [48]. Los metadatos (titulo, contenido y URL de la imagen) de cada NFT son almacenados en IPFS convirtiéndose en valores de identificador de contenido (CID) permitiendo que la gestión de datos en la cadena de bloque se vuelva más eficiente [49].

#### **4.2.15. ABCDE (Agile Block Chain DApp Engineering)**

La metodología ABCDE (Agile Block Chain DApp Engineering) proporciona un enfoque para el desarrollo de aplicaciones descentralizadas (DApps). Dado que las metodologías ágiles tradicionales comprenden aquellos sistemas cuyos requisitos son cambiantes con el tiempo, ABCDE adopta este enfoque ágil para el desarrollo de la aplicación web convencional (sistema de aplicación) y los contratos inteligentes. La metodología se basa en Scrum por su simplicidad. En la Figura 5, se muestra los pasos de la metodología ABCDE.



**Figura 5.** Proceso de la metodología ABCDE

En mayor detalle, el proceso de desarrollo es el siguiente:

1. **Definir el objetivo del sistema:** Es similar al “Sprint Goal” de Scrum, se resume en 10-30 palabras el propósito del sistema y se muestra de forma visible para todo el equipo.
2. **Identificación de actores:** Consiste en identificar los actores, que son roles humanos y sistemas o dispositivos que interactúan con la DApp.
3. **Historias de Usuario:** Los requisitos del sistema se expresan como historias de usuario para seguir el enfoque ágil. En este paso se debe considerar todo el sistema dApp en desarrollo. Es útil, pero no obligatorio, utilizar un diagrama de casos para mostrar las relaciones entre los actores y los casos de uso.

#### 4. División del sistema en dos subsistemas:

Dividir el sistema implica:

- a. Los contratos inteligentes que se ejecutan en la cadena de bloques (Paso (5) a (6))
- b. El sistema de aplicaciones, que es la interfaz externa que interactúa con la cadena de bloques, facilitando la creación y envío de transacciones y supervisando los eventos generados por la ejecución de funciones en los contratos inteligentes (Paso (7) a (8))

Se debe elaborar una arquitectura de todo el sistema, resaltando que datos se deben poner en la cadena y cuales fuera de la cadena. La directriz es que los smart contracts deben gestionar datos y procesos que requieren transparencia e inmutabilidad, para que sus actores confíen en la DApp. Los datos adicionales, el procesamiento y las interfaces de usuario deben manejarse fuera de la cadena. Se utiliza el patrón “Almacenamiento de datos fuera de la cadena”, para la privacidad de aquellos datos que se deben confiar, pero no se pueden almacenar en la cadena debido a su transparencia en la cadena.

5. **Diseño de contratos inteligentes:** El diseño de contratos inteligentes requiere una serie de pasos específicos, cuando se utiliza el lenguaje Solidity. Este proceso se realiza mediante iteraciones que incluyen la codificación y entrega de incrementos de contratos inteligentes, elegidos por los requisitos del usuario para cada iteración. Estos pasos se dividen en subpasos que se consideran y se enumeran siguiendo una secuencia lógica, aunque no necesariamente deben llevarse a cabo en una secuencia en cascada.

- a. Repetir los pasos (2) y (3) para identificar actores y casos de uso, centrándose únicamente en aquellos actores que interactúan directamente con los contratos inteligentes. Si se utilizan contratos inteligentes externos, estos deben incluirse entre los actores. Además, se definen las pruebas de aceptación relacionadas con cada historia de usuario.
- b. En esta fase, se desarrolla una descripción de los contratos inteligentes (SC) que forman parte del subsistema SC. Se especifican las responsabilidades para almacenar información, realizar cálculos y trabajar con otros SC. Los SC interactúan entre sí en entornos complejos utilizando la herencia para modular características comunes. Además, se utilizan diagramas de clases UML con las adecuaciones adecuadas para

aclarar las interacciones con los Smart Contracts externos, incluidas las bibliotecas.

- c. Se define el flujo de mensajes y las transferencias de Ether entre varios Smart Contracts externos y el sistema de. Se utilizan diagramas para representar estas interacciones de manera clara y concisa, utilizamos diagramas de secuencia UML, y si es necesario, recurrimos a diagramas de estado UML para ilustrar los cambios de estado de los SC.
  - d. Describir a detalle la estructura de cada Smart contract, su interfaz externa (Interfaz Binaria de Aplicación, ABI) y los eventos relevantes que puede generar
  - e. Definir las funciones internas y privadas, así como los modificadores. Incluir funciones especiales que verifiquen las condiciones previas necesarias para que una función se ejecute de forma segura
  - f. Definir y realizar pruebas exhaustivas de seguridad. Este es un paso crucial, ya que la mayoría de los Contratos Inteligentes son críticos y gestionan fondos.
6. **Codificación y prueba del sistema de contratos inteligentes:** El sistema de SC se desarrolla y prueba de manera incremental. Las actividades de codificación y pruebas incluyen:
- a. Escriba y pruebe los contratos inteligentes de manera precisa
  - b. Realice la evaluación de la seguridad y la optimización del consumo de gas del código escrito
  - c. Escriba pruebas unitarias automatizadas (UTs) y pruebas de aceptación (ATs) para los contratos inteligentes y los módulos adicionales implementados
7. **Diseño del subsistema de interacción externa:** Este paso consiste en diseñar el sistema de aplicaciones que interactúa con los usuarios y dispositivos, envía mensajes al blockchain y puede gestionar sus propios repositorios (bases de datos y/o documentos). Esta actividad es similar al diseño de una aplicación web estándar, con la adición del blockchain como un actor que puede recibir (pero no enviar) mensajes y generar eventos. Es crucial prestar especial atención a los aspectos de seguridad, ya que a menudo los hackeos de sistemas DApps se deben a debilidades en el sistema de aplicaciones, en lugar de los contratos inteligentes (SC).

- a. Actualizar actores y casos de uso, incorporando los nuevos actores (contratos inteligentes) que interactúan con el sistema de aplicaciones.
  - b. Definir las pruebas de aceptación.
  - c. Diseñar la arquitectura incluyendo niveles de servidor y cliente, detallando el acceso al blockchain, ya sea configurando nodos propios o usando un proveedor externo/cartera estándar
  - d. Crear una interfaz de usuario responsiva para móviles y PCs, utilizando enfoques de diseño estándar como Diseño Centrado en el Uso o Diseño de Interacción para asegurar una UI efectiva y atractiva.
  - e. Descomponer el sistema en módulos, especificando interfaces y flujo de mensajes. Incluir diagramas de estado y acciones basadas en eventos de los contratos inteligentes, definir la estructura y almacenamiento de datos permanentes, y determinar qué datos se anclarán al blockchain usando almacenamiento "Off-Chain". Realizar un diseño iterativo, con incrementos de codificación y entrega, asegurando coherencia con los contratos inteligentes y utilizando diagramas UML para documentar.
- 8. Codificación y prueba del sistema de aplicaciones:** El sistema de aplicaciones se desarrolla y prueba simultáneamente con el sistema de contratos inteligentes, utilizando el mismo método de desarrollo. Cada dos o tres iteraciones, como se muestra en la **Figura 5**, los resultados de ambas ramas deben integrarse, ya que la creación de contratos inteligentes y el sistema de aplicaciones se realizan de manera iterativa. Paralelamente, se llevan a cabo las siguientes actividades.
- a. Los requisitos del sistema de aplicaciones deben implementarse de manera incremental. Esta etapa es similar a la creación de una aplicación web y forma parte del "flujo adecuado" ABCDE (ver **Figura 5**).
  - b. Realice un análisis de seguridad en el código generado para cada incremento. Para los requisitos implementados, desarrolle Pruebas Unitarias (PU) y Pruebas de Aceptación (PA) automatizadas. El conjunto de pruebas debe incluir las nuevas pruebas. Ejecute toda la suite de pruebas para asegurar que las nuevas adiciones no afecten la funcionalidad del sistema.
- 9. Integración, prueba e implementación del Sistema DApp:** Los sistemas globales previamente desarrollados deben ser desplegados en una cadena de bloques o red de prueba local para integrar los contratos inteligentes y el sistema

de aplicación. Es crucial realizar pruebas de integración para asegurar que todas las partes funcionen coherentemente. Por ejemplo, verificar que los mensajes enviados por el sistema de aplicaciones permitan que las transacciones de la cadena de bloques se validen y ejecuten correctamente, así como garantizar que los eventos activados por los contratos inteligentes sean capturados de manera adecuada por el sistema de aplicaciones.

### 4.3. Trabajos Relacionados

En Ecuador, hasta el momento, no se han realizado proyectos que utilicen la tecnología blockchain para ofrecer trazabilidad a las obras de arte. Por lo que se realizó una búsqueda en bases de datos como IEEE y ScienceDirect e indexadores como Scopus para recolectar trabajos que empleen esta tecnología con ese propósito

**Tabla 2.** Trabajos relacionados con el tema de investigación.

Titulo	Resumen	Referencia	Año
ArtChain: Blockchain- Enabled Platform for Art Marketplace	ArtChain es una plataforma basada en blockchain diseñada para la industria del arte. Utiliza la tecnología blockchain para ofrecer un historial de transacciones transparente, a prueba de manipulaciones y que protege la privacidad. Cada obra de arte se representa como un token digital en la cadena de bloques, lo que facilita el rastreo de la propiedad y las transferencias de manera inmutable. Además, utiliza contratos inteligentes actualizables para permitir correcciones de errores y mejoras funcionales en el futuro sin afectar los datos existentes. El objetivo de este estudio fue analizar técnicas basadas en blockchain para transacciones de arte digital contemporáneas. En el pasado, solo el arte físico podía tener el derecho de propiedad sobre el arte digital, pero esta nueva tecnología puede ayudar a los artistas a vender sus obras.	[50]	2019
Digital Artwork Marketplace Web Application Design using Blockchain Technology	El uso de NFT (Tokens No Fungibles) en publicaciones de arte digitales en plataformas de medios de mercado digital puede ayudar a los artistas indonesios a proteger sus derechos de autor en el futuro. La plataforma de mercado estable utiliza la tecnología blockchain Polygon Matic y un contrato inteligente de terceros de MetaMask para procesar cada transacción de forma segura y brindar seguridad y facilidad en el comercio digital.	[51]	2023

Artwork NFTs for  
Online Trading  
and Transaction  
Cancellation

El estudio analiza cómo la tecnología blockchain ha cambiado el intercambio de arte digital al permitir la creación y el comercio de tokens no fungibles (NFT), que son activos distintivos como obras de arte.

Destaca cómo los contratos inteligentes mejoran la seguridad de la representación de activos en la cadena de bloques y la automatización de acuerdos entre usuarios. La aplicación creada permite a los artistas exhibir, vender y transferir la propiedad de sus obras de arte de manera segura, permitiéndoles cancelar ventas y reembolsar a los compradores.

Además, se discuten los beneficios de blockchain, como la seguridad y la transparencia, junto con los inconvenientes, como la complejidad técnica y la volatilidad del mercado, demostrando cómo la integración está cambiando el mercado del arte digital y redefiniendo la propiedad y el intercambio de activos digitales.

[52]

2023

## 5. Metodología

En esta sección se presenta la metodología utilizada para el desarrollo del presente trabajo de integración curricular. En el punto **5.1** se encuentra el área de estudio que da contexto sobre el entorno en el que se abarco la investigación inicial. El punto **5.2** describe el procedimiento realizado para alcanzar cada uno de los objetivos planteados. Por último, los métodos, técnicas, estándares y materiales utilizado se describen en el punto **5.3**.

### 5.1. Área de estudio

La investigación inicial se llevó a cabo en la oficina de docentes la Carrera de Artes Plásticas de la Universidad Nacional de Loja ubicada en las coordenadas geográficas 4.033455, -79.200319. En este espacio de trabajo y en colaboración con el Mgs. Adolfo Aguirre se llevó a cabo una serie de reuniones, que fueron la base para el desarrollo del trabajo de integración curricular.



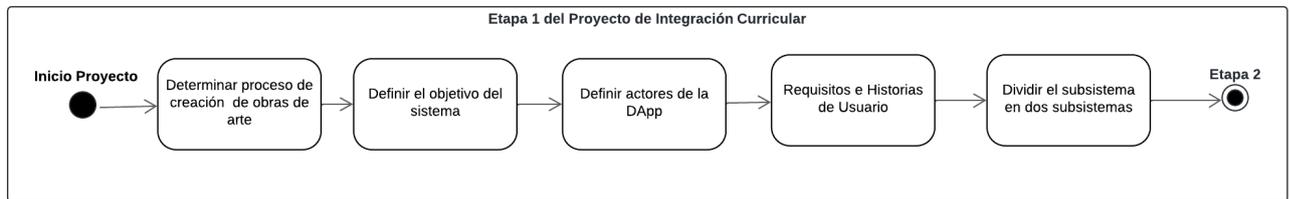
**Figura 6.** Oficina de docentes de la carrera de artes plasticas

### 5.2. Procedimiento

Para alcanzar cada uno de los objetivos se llevó a cabo una serie de actividades que se describen a continuación:

**5.2.1. Objetivo 1: Integrar la tecnología Blockchain Ethereum el sistema actual de ventas de obras de arte mediante la metodología ABCDE (Agile Block Chain DApp Engineering)**

El primer objetivo si dividió en dos etapas **Figura 7** y **Figura 8**. Cada etapa abarca las actividades de la metodología ABCDE ("Agile Blockchain DApp Engineering") descritas en el punto **4.2.15** del marco teórico.



**Figura 7.** Diagrama de actividades de la primera etapa del Trabajo de Integración curricular de Integración Curricular

- **Fase 1: Determinar el proceso de creación de obras de arte**
  - Se realizaron dos reuniones con el Mgs. Adolfo Aguirre docente de la carrera de artes plastica. Durante la primera reunión, se llevó cabo una entrevista (véase **Anexo 2**. Entrevista realizada al docente de la carrera de artes plásticas ) para conocer cómo se crean las obras de arte tanto físicas como digitales y obtener los requisitos de la dApp. En la segunda reunión, se presentó un esquema base del proceso creado a partir de la información recolectada junto con los requisitos funcionales, obteniendo retroalimentación y consideraciones para su mejora. En base a las mejoras realizadas se obtuvo el proceso final para la creación de las obras de arte el cual se muestra en el punto **6.1.1** de la sección de resultados.
- **Fase 2: Definir el objetivo del sistema.**
  - Se definió el objetivo del sistema, alineándolo con el primer objetivo específico del trabajo de integración curricular.
- **Fase 3: Definir los actores del sistema**
  - Se hizo una revisión de los actores que intervienen en el diagrama de casos de uso de la plataforma de difusión y ventas de obra arte (véase **Anexo 3**. Diagrama de casos de uso de la plataforma de difusión y venta de obras de arte), mismos que se mantuvieron para la interacción con la dApp, pero con un nuevo actor agregado, el contrato inteligente.
- **Fase 4: Requisitos e Historias de Usuario**
  - Se expresaron los requisitos funcionales de la dApp en historias de usuario (véase **Anexo 5**. Historias de Usuario) para mantener el enfoque ágil de la

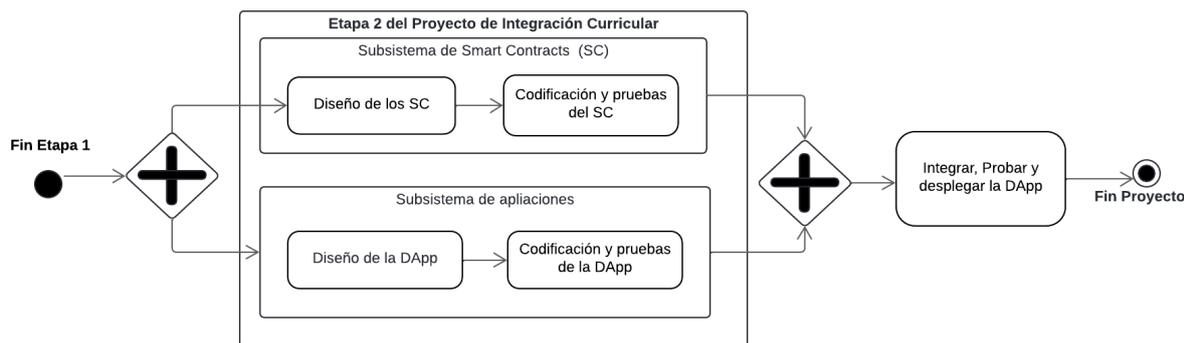
metodología, tal como se describe el **punto 6.1.4** de la sección de resultados. Los requisitos funcionales representaron las nuevas funcionalidades que permitieron dar trazabilidad las obras de arte, mismos que fueron elaborados mediante el estándar IEE 830 (véase **Anexo 4**. Especificación de requisitos IEE 830).

- **Fase 5: Dividir el sistema en dos subsistemas**

Para una visión general de los dos subsistemas se realizó una arquitectura simplificada de la dApp (ver **Figura 11** ) en la que se refleja la interacción del subsistema de aplicaciones y el de contratos inteligentes.

- El subsistema de contratos inteligentes se encarga de la interacción con la cadena de bloques y para su desarrollo se utilizó el lenguaje de programación Solidity en su versión 0.8.20, para las pruebas se utilizó Truffle en su versión 5.11.5 y como red de pruebas Ethereum local se utilizó Ganache en su versión de 6.4.2.
- El subsistema de aplicaciones representa al sistema de difusión y ventas de obras, por lo que se utilizaron las mismas herramientas con las que fue desarrollado para desarrollar las nuevas funcionales. Las herramientas tecnológicas usadas fueron Django en su versión 4.2 como servidor, HTML, CSS, JS para el frontend y postgresSQL en su versión 16.3.1 como base de datos para la gestión de los usuarios. Los componentes nuevos en este subsistema fueron la librería web3.py en su versión 6.19.0 para la interacción con el contrato inteligente. Metamask como gestor de billeteras, ingresar a la dApp y realizar las transacciones. Todo lo mencionado se encuentra en el punto **6.1.5** de la sección de Resultados

En este punto la primera etapa del desarrollo de la aplicación descentralizada fue culminada con éxito, dando paso a la segunda etapa de desarrollo cuyas actividades se ilustran en la **Figura 8**.



**Figura 8.** Diagrama de actividades de la segunda etapa del desarrollo de la aplicación descentralizada

Durante esta segunda etapa se abordaron aspectos relacionados con el diseño, codificación y prueba de los subsistemas. Además, se realizó el despliegue la dApp junto con el contrato inteligente.

A continuación, se describen las actividades realizadas durante esta etapa.

- **Fase 6: Diseño de los Smart Contracts**

- Se realizó un diagrama de clases en el que se muestra las diferentes estructuras de datos y sus distintas relaciones. La estructura de datos permitió gestionar la información de las obras de arte almacenada en la cadena de bloques. El diagrama de clases se ilustra en el punto **6.1.7** de la sección de resultados.

- **Fase 7: Codificación y prueba de los Smart Contracts**

- Se codifico el contrato inteligente utilizando el entorno de desarrollo RemixIDE junto con lenguaje de programación Solidita para la gestión de los procesos de creación de las obras de arte y el estándar ERC-721 de OppenZeppeling para la gestión de las obras de arte en su representación como NFT en la cadena de bloques (véase **Anexo 7**. Código fuente del contrato inteligente )
- Luego se llevaron a cabo pruebas unitarias para evaluar cada función del contrato inteligente y verificar que su comportamiento fuera el esperado (véase **Anexo 8**. Plan de pruebas unitarias para el contrato inteligente SDVOA). En total, se obtuvieron 18 casos de prueba, los cuales fueron automatizados mediante Truffle que permitió ejecutar y validar cada caso de prueba.

- **Fase 8: Diseño del subsistema de aplicaciones**
  - Para el diseño del subsistema de aplicaciones, en este caso se hizo un rediseño de su arquitectura con el que se integró la interacción con el contrato inteligente y a su vez con la cadena de bloques (ver **Figura 18**).
  - Luego se codificó las funcionalidades para interactuar con la cadena de bloques usando el mismo framework, con el que está desarrollado. Parte de estas funcionalidades se muestran en la **Figura 22** de del punto **6.1.9** de la sección de resultados, no se muestran todas debido a su extensión y debido a que interactúan con otros módulos que no se pueden mostrar por el contrato de confidencialidad firmado (véase **Anexo 13**. Contrato de confidencialidad)
  - Posterior se realizaron las respectivas pruebas unitarias (véase **Anexo 10**. Plan de pruebas unitarias para el subsistema de aplicaciones) a cada una de las nuevas funcionalidades desarrolladas, cuya ejecución fue automatizada mediante la librería de pruebas unitarias TestCase. Todo lo obtenido en esta fase se muestra en el punto **6.1.9** de la sección de resultados.
- **Fase 9: Integrar, Probar y desplegar la dApp**
  - Primero se realizaron las pruebas de integración entre el subsistema de aplicaciones y de contratos inteligentes (Véase **Anexo 11. Plan de pruebas de Integración de la DApp**).
  - Posterior se realizaron las pruebas funcionales (Véase **Anexo 12. Plan de pruebas Funcionales para la DApp**) para validar cada uno de los requisitos funcionales definidos y asegurar su correcto funcionamiento y cumplimiento.
  - Luego procedió a desplegar el contrato inteligente en la red de pruebas de Polygon zkEVM Cardona mediante truffle y usando alchemy como el nodo que proporciona la interacción con la cadena bloques.
  - Finalmente se realizó el despliegue de la aplicación descentralizada en los servicios de nube otorgados por Render Cloud, se crearon los servicios necesarios para el correcto funcionamiento de la dApp tal como se describe en el punto **6.1.12** de la sección de resultados

### **5.2.2. Evaluar el porcentaje de trazabilidad en el proceso de venta de obras de arte a través del Indicador Clave de Rendimiento (KPI) de trazabilidad especializado para la industria blockchain**

Para el desarrollo de este objetivo se realizó y se ejecutó una serie de pasos estructurados con el objetivo evaluar de manera eficiente el porcentaje de trazabilidad. A continuación, se describen los pasos realizados para la evaluación.

- **Identificación de activos y Selección de Activos**
  - Se realizó la identificación de los activos a ser rastreados, en este punto fue necesario definir a que se considera un activo dentro de la dApp, por la naturaleza misma de la cadena bloques para poder realizar un seguimiento o ver la trazabilidad de un activo es indispensable que este registrado en la cadena de bloques. Es por ello que en la dApp un activo se considera a cada obra de arte en su representación como NFT producto de la culminación de su proceso, esto es primordial para realizar un seguimiento sobre su proceso de creación y sus transacciones.
  - Siguiendo estas pautas se procedió a realizar un registro de 18 procesos de prueba por parte de 2 responsables (artistas), de los cuales 13 fueron completados con éxito para que se lo más parecido a un entorno real, como producto de esos 13 procesos se obtuvo 13 obras de arte en su representación como NFT y registrados en la cadena de bloques. Mismos que se seleccionaron para la evaluación.
- **Establecimiento de los criterios de selección**
  - Para realizar un seguimiento completo de la obra de arte desde su proceso de creación hasta su venta se establecieron criterios de selección necesarios para considerar que la obra de arte fue rastreada con éxito, los criterios fueron seleccionados en función a las capacidades específicas como el registro de eventos mediante los contratos inteligentes, la verificabilidad de la información en la cadena de bloques y como base la naturaleza mismo de la cadena de bloques para proporcionar trazabilidad. Los criterios definidos se presentan en el punto **6.2.2** de la sección de resultados.
- **Recopilación de datos**
  - Se procedió a recolectar los datos necesarios para la evaluación. En este punto se hizo uso de los eventos codificados en el contrato inteligente los mismos que fueron agregados tomando en cuenta la evaluación de la

trazabilidad (Para más detalle véase **Anexo 7**. Código fuente del contrato inteligente). Luego se verifico que los eventos necesarios para evaluar la trazabilidad fueran emitidos durante el registro del proceso de creación de las obras y su producto final. Para ello primero se hizo una revisión manual de los eventos o logs en la red de pruebas de Polygon zkEVM. Una vez verificado que los eventos se emitieran correctamente se desarrolló una función para recolectar todos los eventos emitidos por el contrato y filtrar los mismos en base a los criterios previamente definidos, la función permitió minimizar los errores al recolectar la información y garantizó el manejo de una gran cantidad de información de manera eficiente. Todo lo mencionado se muestra en el punto **6.2.2** de la sección de resultados

- **Cálculo del porcentaje de trazabilidad**

- Con la información necesaria para evaluar el porcentaje de trazabilidad, se procedió a usar la formula descrita en el punto **4.2.10** del marco teórico. Esto se lo realizo en la misma función que se desarrolló debido a que en la misma se obtienen las variables necesarias para calcular el porcentaje de trazabilidad. Todo lo mencionado se describe en el punto **6.2.4** de la sección de resultados

### **5.3. Recursos**

Para el desarrollo del presente Trabajo de Integración Curricular, fueron necesarios los siguientes recursos

#### **5.3.1. Recursos Científicos**

##### **5.3.1.1. Método analítico**

Este método consiste en descomponer un problema complejo en partes más pequeñas y manejables. Al desarrollar la DApp, se dividió el Trabajo de Integración curricular en objetivos específicos y sus actividades asociadas.

##### **5.3.1.2. Investigación – acción**

Consiste en una serie de etapas, que van desde la identificación de un problema hasta su ejecución de acciones concretas para abordarlo. En pocas palabras se basa en la idea de se aprende mientras se va haciendo. Este enfoque fue útil para diversas etapas del Trabajo de Integración Curricular.

##### **5.3.1.3. Replicación**

Es el proceso de repetir un estudio o experimento para verificar los resultados obtenidos. Este método permitió realizar pruebas para asegurar la consistencia de las

integraciones y modificaciones en el sistema. Además, facilito la creación de diferentes versiones del documento de Especificación de requerimientos IEEE 380. La replicación también fue útil para probar nuevas funcionalidades o ajustes, garantizando que cualquier cambio realizado mejore la fiabilidad del sistema

#### **5.3.1.4. Simulación**

La simulación permitió crear escenarios con datos ficticios para realizar las pruebas a cada uno de los Smart Contract y garantizar que su comportamiento sea el esperado. Además, este método se utilizó para simular el sistema en una testnet de blockchain y verificar la trazabilidad de las obras de arte.

### **5.3.2. Métodos Observacionales**

#### **5.3.2.1. Análisis Estático**

Este método permitió analizar el código fuente del sistema para comprender su funcionamiento y poder integrar de manera óptima las nuevas funcionalidades. Además, se utilizó para optimizar y detectar posibles errores potenciales de los Smart Contracts.

#### **5.3.2.2. Caso de Estudio**

Este enfoque permitió identificar y desarrollar soluciones adaptadas al sistema para la integración de los NFT y validar la trazabilidad de las obras de arte.

### **5.3.3. Recursos Técnicos**

#### **5.3.3.1. Entrevista**

Esta técnica tradicional fue útil para obtener información relacionada con el proceso de creación de obras de arte y determinar los requerimientos que se incorporaron en el sistema.

#### **5.3.3.2. Reuniones**

Esta técnica permitió recolectar información el proceso de creación de obras de arte y los requisitos funcionales definidos. Además de revisar los diferentes avances de este Trabajo de Integración Curricular.

### **5.3.4. Estándares**

#### **5.3.4.1. IEE 830**

Es un conjunto de recomendaciones que permitió especificar los requerimientos de la DApp

#### **5.3.4.2. ERC 721**

El estándar fue útil para la creación del contrato inteligente que otorgo todas las funcionalidades para dar trazabilidad a las obras de arte

### 5.3.5. Recursos de hardware y Software

El Trabajo de Integración Curricular se llevó a cabo con el uso de diversos recursos de software y hardware detallados en la tabla **Tabla 3** y **Tabla 4**. Estos recursos fueron necesarios para el desarrollo del Trabajo de Integración curricular y permitieron llevar a cabo diversas tareas como la creación de los diagramas de casos de uso, la arquitectura del sistema, la codificación, despliegue y prueba de los contratos inteligentes y del sistema. Así como el almacenamiento de datos y ambientes de prueba para la DApp.

#### 5.3.5.1. Recursos Software

En la tabla **Tabla 3** se muestran los recursos de software que fueron utilizados para llevar a cabo el desarrollo del Trabajo de Integración curricular, en este se incluye la descripción de cada recurso y su enlace a la web.

**Tabla 3.** Recursos de Software

Recurso	Descripción	Acceso
Lucidchar	Software para el diagramado abarca desde casos de uso, arquitectura e interfaces de usuario	<a href="https://www.lucidchart.com/pages/es">https://www.lucidchart.com/pages/es</a>
Pycharm	Entorno de desarrollo de Python para la creación de aplicaciones web	<a href="https://www.jetbrains.com/es-es/pycharm/download/">https://www.jetbrains.com/es-es/pycharm/download/</a>
Remix IDE	Entorno de desarrollo para los contratos inteligentes	<a href="https://remix.ethereum.org/">https://remix.ethereum.org/</a>
Truffle	Herramienta que permite crear, desplegar y probar los contratos inteligentes	<a href="https://archive.trufflesuite.com/truffle/">https://archive.trufflesuite.com/truffle/</a>
Ganache	Es un simulador de red Ethereum que nos permite crear una red Ethereum local en nuestro equipo	<a href="https://archive.trufflesuite.com/docs/ganache/">https://archive.trufflesuite.com/docs/ganache/</a>
Pinata IPFS	Pinata es una plataforma de desarrollo web3 emergente que ofrece la ventaja de almacenamiento escalable a través de IPFS	<a href="https://www.pinata.cloud/">https://www.pinata.cloud/</a>
Metamask	Cartera para criptomonedas y DApps en navegadores.	<a href="https://metamask.io/">https://metamask.io/</a>
Web3 py	Biblioteca Python para interactuar con Ethereum.	<a href="https://web3py.readthedocs.io/en/stable/">https://web3py.readthedocs.io/en/stable/</a>
Pytest	Framework de testing para Python	<a href="https://docs.pytest.org/en/8.2.x/">https://docs.pytest.org/en/8.2.x/</a>
Django Test case	Herramienta de Django para pruebas unitarias.	<a href="https://docs.djangoproject.com/en/5.0/topics/testing/overview/">https://docs.djangoproject.com/en/5.0/topics/testing/overview/</a>

Git Hub      Plataforma de desarrollo y control de versiones.      <https://github.com/>

---

#### 5.3.5.2. Recursos Hardware

En la tabla **Tabla 4** se muestran los recursos de Hardware que fueron utilizados para llevar a cabo el desarrollo del Trabajo de Integración curricular.

**Tabla 4.** Recursos Hardware

Recurso	Descripción
Laptop	Equipo indispensable en el desarrollo del Trabajo de Integración curricular

#### 5.3.6. Recursos Humanos

En la **Tabla 5**, se muestra las personas involucradas en diferentes etapas del desarrollo del presente Trabajo de Integración curricular.

**Tabla 5.** Recursos Humano

Nombre	Descripción
Alexander Adrián Vicente Jiménez	Estudiante a cargo del desarrollo del Trabajo de Integración curricular
Ing. Cristian Ramiro Narvárez Guillen	Docente director del Trabajo de Integración curricular de Integración Curricular
Mgsc. Adolfo Andrés Aguirre Celi	Docente de la carrera de artes plásticas

## 6. Resultados

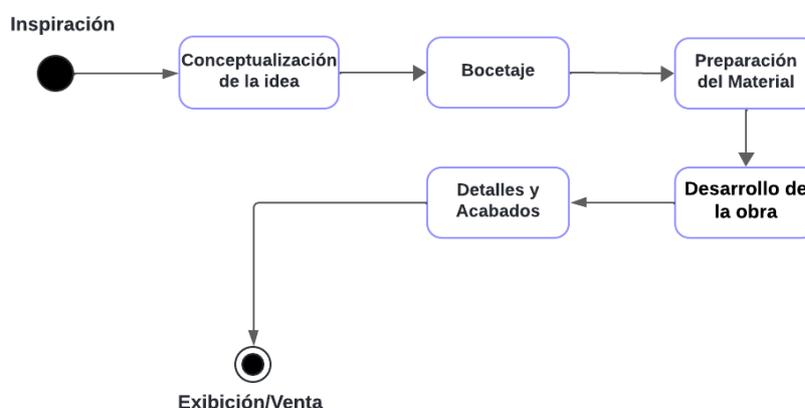
En esta sección se presentan los resultados obtenidos durante la ejecución del Trabajo de Integración curricular, el cual se desarrolló siguiendo la metodología ABCDE y cumpliendo con los objetivos específicos establecidos.

### 6.1. Objetivo 1: Integrar la tecnología Blockchain Ethereum en el sistema actual de ventas de obras de arte mediante la metodología ABCDE (Agile Blockchain DApp Engineering)

#### 6.1.1. Determinar el proceso de creación de obras de arte

En base a la entrevista realizada (Véase **Anexo 2**. Entrevista realizada al docente de la carrera de artes plásticas ) al docente de la carrera de artes plásticas, cuyo perfil se describe en la sección de **Recursos Humanos**, se identificó y definió el proceso de creación de las obras de arte que, el mismo que permitió dar trazabilidad a cada obra de arte desde su creación y delimito la información que sería almacenada en la blockchain para asegurar su inmutabilidad y transparencia.

En la **Figura 9** se ilustra el proceso de creación de obras de arte que se obtuvo mediante la entrevista y que se incorporó en la aplicación descentralizada.



**Figura 9.** Proceso de creación de obras de arte

#### 6.1.2. Fase 1: Objetivo de la dapp

El objetivo de la aplicación descentralizada (dApp) se alinea con el objetivo general del Trabajo de integración curricular el cual es: **Implementar la tecnología blockchain en el sistema de la carrera de artes visuales para dar trazabilidad a las obras de arte.**

### 6.1.3. Fase 2: Definir los actores de la dApp

En este punto los actores que interactuaron con la aplicación descentralizada se definieron en base a los actores que intervienen en el sistema de difusión y ventas de obras de artes, los cuales se encuentran en el diagrama de casos del sistema (véase **Anexo 3**. Diagrama de casos de uso de la plataforma de difusión y venta de obras de arte ) y con un nuevo actor que es el contrato inteligente.

En la **Tabla 6** se describen los actores del sistema de difusión y ventas de obras de arte junto con el nuevo actor.

**Tabla 6.** Actores de la dApp

Actores	Descripción
<b>Encargado</b>	Docente de la carrera de Artes Visuales quien registra los responsables
<b>Responsable</b>	Estudiante de la carrera de Artes Visuales a cargo de llevar todo el proceso de creación y registro de las obras de arte
<b>Cliente</b>	Usuario interesado en comprar obras de arte
<b>SC-SDVOA</b>	Contrato inteligente para almacenar la información e interactuar con lo blockchain

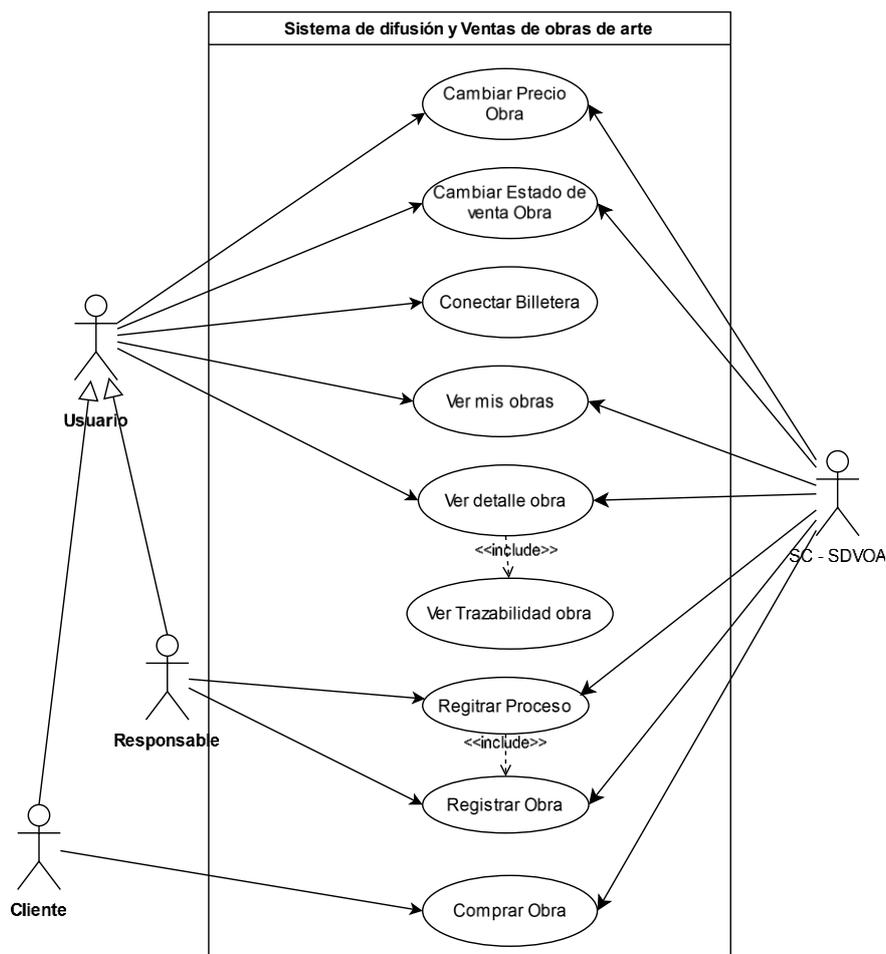
### 6.1.4. Requisitos e Historias de usuario

En esta sección, se presentan los siguientes puntos: Diagrama de casos de uso de la aplicación descentralizada, los nuevos requisitos funcionales y no funcionales extraídos del documento de especificación de requerimientos IEE830 (véase **Anexo 4**. Especificación de requisitos IEE 830) y un resumen de las historias de usuario (véase **Anexo 5**. Historias de Usuario).

#### 6.1.4.1. Diagrama de Casos de Uso

Para obtener una visión general de la interacción entre los actores y la aplicación descentralizada, se realizó un diagrama de casos de usos en el que se muestran estas interacciones.

En la **Figura 10** se ilustra el diagrama de casos de uso de la aplicación descentralizada.



**Figura 10.** Diagrama de Casos de Uso (SC-SDVOA: Smart contract para el sistema de difusión y ventas de obras de arte)

### 6.1.4.2. Requisitos Funcionales

En la **Tabla 7**, se describen los nuevos requisitos funcionales para proporcionar trazabilidad a cada obra de arte.

**Tabla 7.** Requisitos funcionales de la dApp

Identificación	Nombre	Descripción
RF01	Registrar Proceso	La dApp permitirá a un responsable registrar el proceso de creación de sus obras de arte en la cadena de bloques
RF02	Registrar Obra	La dApp permitirá al responsable registrar una obra en la cadena de bloques
RF03	Cambiar precio de una Obra	La dApp permitirá al usuario propietario de una obra modificar su precio
RF04	Comprar Obra	La dApp permitirá a los clientes comprar una obra disponible
RF05	Cambiar estado de venta de una Obra	La dApp permitirá al usuario propietario de una obra cambiar su estado de venta
RF06	Ver la trazabilidad de la obra	La dApp permitirá al usuario ver la trazabilidad de la obra, desde su proceso de creación hasta sus transacciones
RF07	Ver detalles de la Obra	La dApp permitirá ver todos los datos asociados a una obra

RF08	Conectar Billetera	La dApp permitirá a los usuarios conectar su billetera de Ethereum para poder realizar transacciones
RF09	Ver mis obras	La dApp permitirá a los usuarios ver las obras que son de su propiedad

### 6.1.4.3. Requisitos No Funcionales

En la **Tabla 8**, se describen los requisitos no funcionales para una correcta implementación y operación de la aplicación descentralizada.

**Tabla 8.** Requisitos no funcionales

Identificación	Nombre	Descripción
RNF01	Rendimiento	En una red de prueba de Ethereum, la transacción tarda entre 2 a 5 segundos; sin embargo, en un entorno real, puede tardar entre 15 segundos a 5 minutos
RNF02	Usabilidad	El sistema debe proporcionar una interfaz que permita a los usuarios visualizar la trazabilidad completa de las obras de arte.
RNF03	Seguridad	Se debe garantizar que solo los usuarios autorizados puedan realizar ciertas operaciones en el sistema

### 6.1.4.4. Historias de Usuario

En la **Tabla 9**, se muestra un resumen de las historias de usuario obtenidas a partir de los requisitos funcionales.

**Tabla 9.** Resumen de las historias de usuario

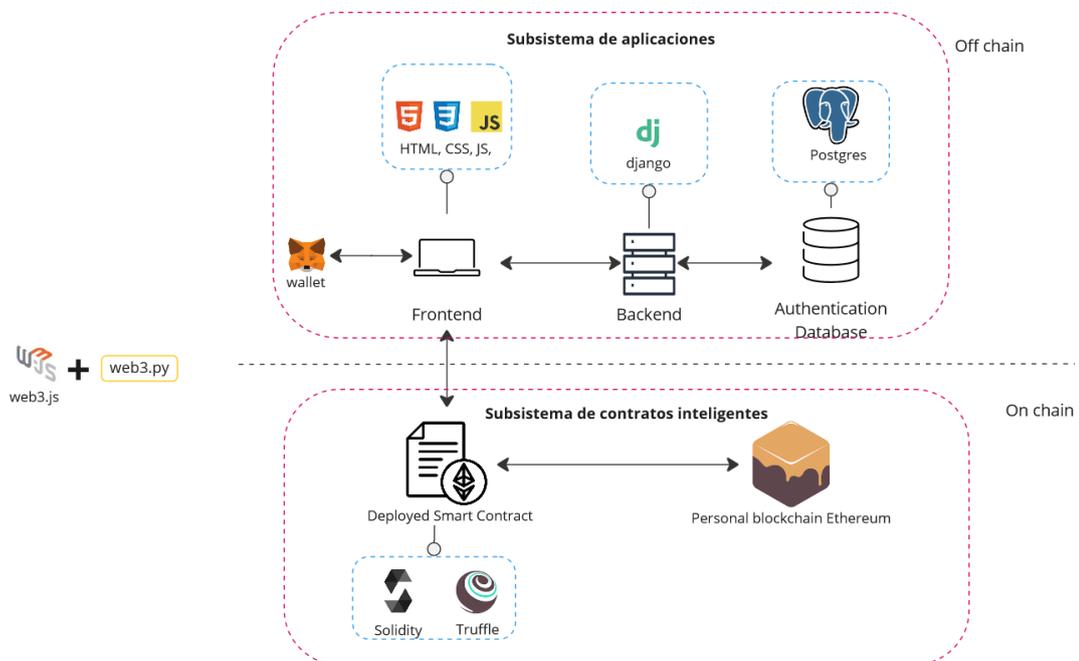
Rol	Característica / Funcionalidad	Razón / Resultado
Como un responsable	Necesito registrar el proceso de creación de una obra de arte	Con la finalidad almacenar de manera transparente la información asociada al proceso de creación
Como un responsable	Necesito convertir una obra de arte en un NFT	Con la finalidad de proporcionar una representación única y verificable de la obra
Como un Usuario	Necesito cambiar el precio de mis obras	Con la finalidad de ajustar su valor a los cambios del mercado
Como un usuario	Necesito cambiar el estado de venta de mis obras	Con la finalidad de activar o desactivar su venta
Como un usuario	Necesito ver mis obras	Con la finalidad de gestionar las obras que son de mi propiedad
Como un usuario	Necesito ver los detalles de una obra	Con la finalidad de conocer información específica de una obra
Como un Cliente	Necesito comprar una obra	Con la finalidad de adquirir la propiedad de una obra
Como un Cliente	Necesito ver la trazabilidad de una obra de arte	Con la finalidad de conocer la procedencia y autenticidad de la obra

Como un Usuario	Necesito conectar mi billetera a la Dapp	Con la finalidad de realizar transacciones dentro de la dapp
-----------------	--	--

### 6.1.5. Dividir el sistema en dos subsistemas

En este punto es necesario conocer y comprender la interacción entre el sistema de difusión y ventas de obras de arte con la cadena de bloques. Para ello se realizó una arquitectura simplificada de la aplicación descentralizada, en la que se muestra la interacción de los dos subsistemas: El de contratos inteligentes, que se ejecuta en la cadena de bloques (on-chain) y el sistema de difusión y ventas de obras de arte (off-chain).

En la **Figura 11** se ilustra la arquitectura simplificada de la aplicación descentralizada junto con los dos subsistemas y las diferentes herramientas as utilizadas para el desarrollo de cada subsistema.



**Figura 11.** Arquitectura simplificada de la dapp

El subsistema de aplicaciones, que se trata del sistema de difusión y ventas de obras de arte se compone de las tecnologías con las que fue desarrollado HTML, CSS Y JS para el frontend, Django para el backend y una base de datos postgres, cabe recalcar que la base de datos se usó para la gestión de datos de los usuarios y se incorporó Metamask para la gestión de billeteras de los usuarios y para realizar transacciones en la cadena de bloques. Para la interacción con el subsistema de contratos inteligentes, se hizo uso de la librería web3.py que permitió la interacción con el contrato inteligente desarrollado mediante solidity y desplegado mediante truffle en la red local blockchain de Ethereum Ganache.

### 6.1.6. Diseño del Subsistema de contratos inteligentes y de aplicaciones

En la presente sección se detalla la planificación realizada para el desarrollo de la aplicación descentralizada, enfocándose en el diseño del subsistema de contratos inteligentes y de aplicaciones.

#### 6.1.6.1. Iteraciones de la DApp

Para un desarrollo estructurado, se definió una serie de iteraciones específicas para cada subsistema.

En la **Tabla 10** se describen las iteraciones definidas para el desarrollo de los contratos inteligentes.

**Tabla 10.** Iteraciones para el subsistema de contratos inteligentes

Nro. Sprint	Nombre	Duración en Horas
1	Configuración Inicial	5
2	Diseño del contrato inteligente	8
3	Configuración del contrato inteligente ERC - 721	20
4	Codificación del contrato inteligente	35
5	Optimización del contrato	15
6	Compilación del contrato	3
7	Migración del contrato	5
8	Pruebas Unitarias	20

Por otro lado, en la **Tabla 11** se presentan las iteraciones para el desarrollo del subsistema de aplicaciones.

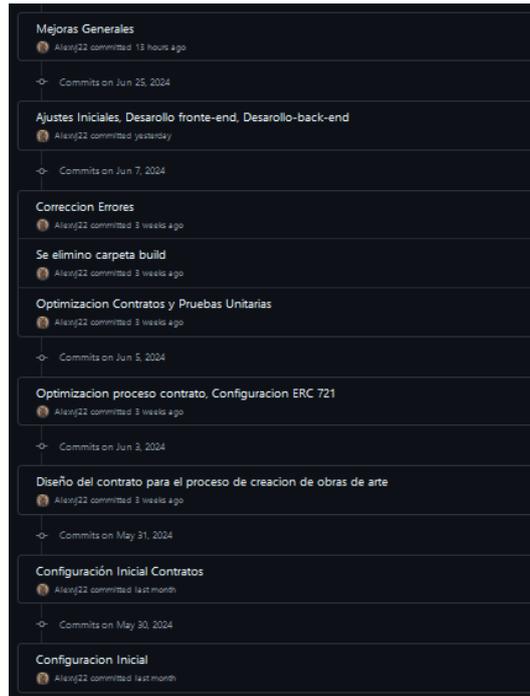
**Tabla 11.** Tabla de iteraciones del subsistema de aplicaciones

Nro Sprint	Nombre	Duración en Horas
1	Configuración del entorno	6
2	Diseño del backend	10
3	Implementación del backend	30
4	Integración con el contrato inteligente	10
5	Pruebas unitarias	20
6	Desarrollo del frontend	35
7	Pruebas de integración	15
8	Despliegue de la dapp	10

#### 6.1.6.2. Control de Versiones

Para el control de cada iteración desarrollada se utilizó la herramienta de versionamiento GitHub.

En la **Figura 12** se observan los diferentes commits realizados en la herramienta, cada commit representa las iteraciones y mejoras realizadas durante el desarrollo de los dos subsistemas.



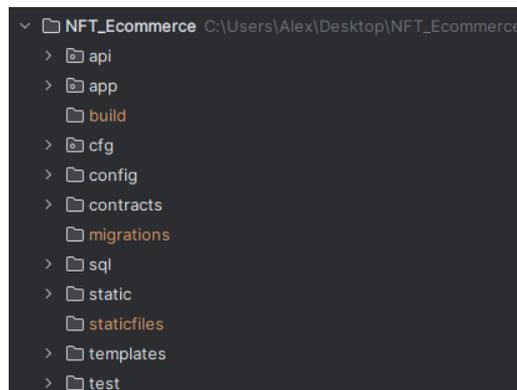
**Figura 12.** Control de versiones mediante la herramienta de versionamiento Github

### 6.1.6.3. Estructura de carpetas la Dapp

En la presente sección se detalla la estructura de carpetas tanto del subsistema de aplicaciones como el de contratos inteligentes, con la finalidad de obtener una visión general sobre cómo se organizaron los diferentes archivos y recursos de la dApp.

### 6.1.6.4. Estructura de carpetas del contrato inteligente

En la **Figura 13** se presenta la estructura de carpetas pertenecientes al subsistema de contratos inteligentes, estas son: *build*, *contracts*, *migrations* y *test* las mismas fueron generadas mediante la herramienta truffle facilitando el desarrollo, despliegue y prueba del contrato inteligente.



**Figura 13.** Estructura de carpetas del contrato inteligente

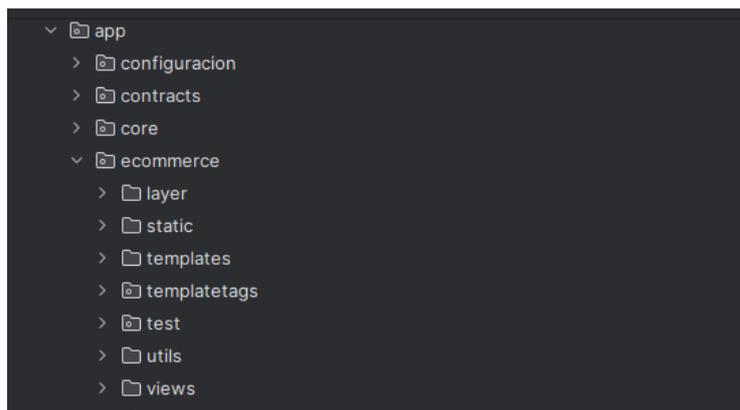
En la **Tabla 12** se describe cada carpeta del subsistema de contratos inteligentes.

**Tabla 12.** Descripción de la estructura de carpetas del contrato inteligente

<b>Carpeta</b>	<b>Descripción</b>
<b>build/</b>	Contiene el ABI y el bytecode del contrato inteligente en formato JSON para la interacción con la dApp.
<b>contracts/</b>	Contiene el contrato inteligente codificado en el lenguaje de programación Solidity
<b>migrations/</b>	Contiene el archivo de migración del contrato escrito en javascript, se usa principalmente para desplegar el contrato en la cadena de bloques.
<b>test/</b>	Contiene el script de pruebas unitarias para el contrato inteligente

### 6.1.6.5. Estructura de carpetas del subsistema de aplicaciones

El subsistema de aplicaciones (Sistema de difusión y ventas de obras de arte) mantiene la estructura de carpetas del patrón de diseño del framework Django, el mismo que se compone de: *Models, Views y Templates*. El agregado a esta estructura se encuentra en la carpeta *contracts* que contiene el archivo desarrollado mediante la librería web3 para la interacción con la cadena de bloques, tal como se muestra en la [¡Error! No se encuentra el origen de la referencia..](#)



**Figura 14.** Estructura de carpetas del subsistema de aplicaciones

En la **Tabla 13** se describe la estructura del subsistema de aplicaciones

**Tabla 13.** Descripción estructura de aplicaciones

<b>Carpeta</b>	<b>Descripción</b>
contracts/	Contiene el archivo desarrollado mediante la librería web3.py esencial para interactuar con la cadena de bloques.
Models	Contiene los modelos de datos de la dapp, se usó principalmente para la gestión de usuarios
views/	Contiene la lógica de negocio, junto con las funcionalidades de lectura sobre el contrato inteligente
templates/	Contiene todas las plantillas (Interfaces de Usuario) para la interacción entre usuario y dApp

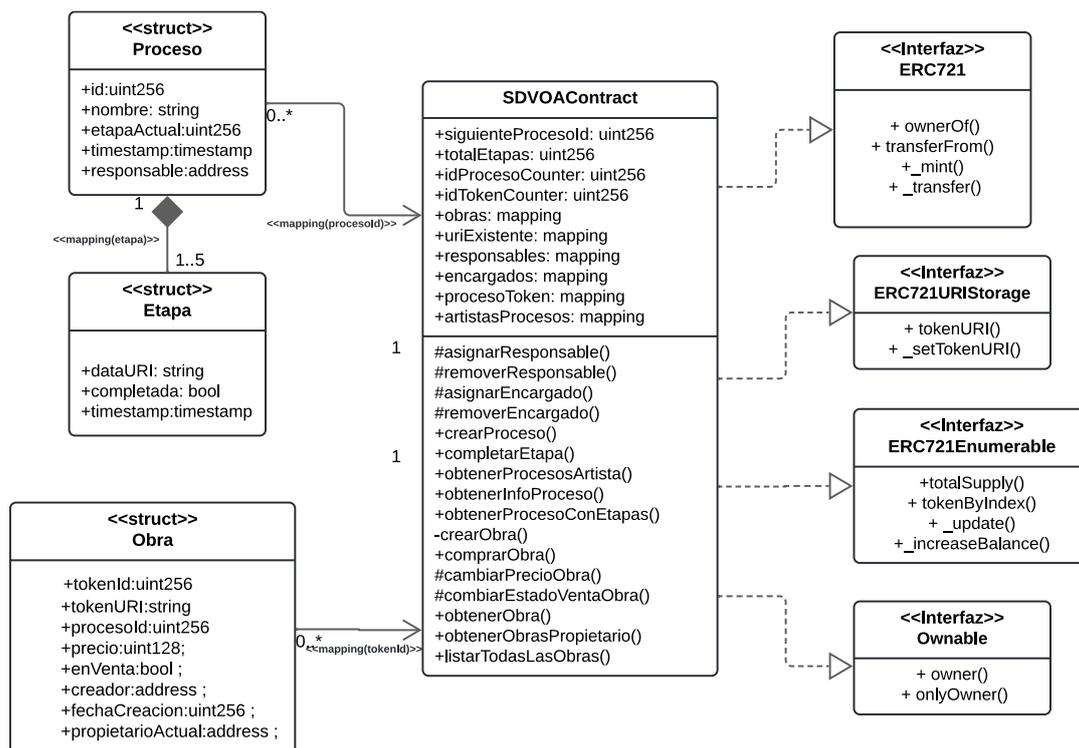
### 6.1.7. Diseño del contrato inteligente

Durante esta sección se explica el cómo se diseñó, codificó y probó el contrato inteligente utilizado para proporcionar trazabilidad a las obras de arte.

#### 6.1.7.1. Diagrama de clases

La mayor parte de la información se registra y se accede mediante el contrato inteligente es por ello que se diseñaron diferentes estructuras de datos para su manejo.

En la **Figura 15** se ilustra el diagrama de clases del contrato inteligente



**Figura 15.** Diagrama de clases del contrato inteligente

### 6.1.8. Codificación y prueba de los Smart Contracts.

El contrato desarrollado mediante el estándar ERC-721 y con el lenguaje de programación solidity se denomina SDVOAContract.sol, para un mayor detalle se recomienda revisar el **Anexo 7**. Código fuente del contrato inteligente.

En la **Figura 16** se muestra parte del código del contrato inteligente desarrollado

```

contract SDVOContract is ERC721, ERC721Enumerable, ERC721URIStorage, Ownable {
    constructor() ERC721("Coleccion de Arte", "ARTE") Ownable(msg.sender) {}
    modifier soloResponsable() {}
    modifier soloEncargado() {}
    function asignarEncargado(address _encargado) external onlyOwner {}
    function removerEncargado(address _encargado) external onlyOwner {}
    function asignarResponsable(address _responsable) external soloEncargado {}
    function removerResponsable(address _responsable) external soloEncargado {}
    function crearProceso(string memory _nombre) external soloResponsable{}
    function completarEtapa(uint256 procesoId, string memory _uriEtapa, string memory _uriToken, uint128 precio)
    external soloResponsable{}
    function obtenerProcesosDelArtista(address artista) external view returns (uint256[] memory) {}
    function obtenerInfoProceso(uint256 procesoId) external view returns (uint256 id, string memory nombre, uint256
    etapaActual, uint256 timestamp) {}
    function obtenerProcesoConEtapas(uint256 procesoId) external view returns (uint256 id, string memory nombre,
    uint256 etapaActual, Etapa[] memory etapas) {}
    function crearObra(string memory _uriToken, uint128 _precio, uint256 _idProceso) internal soloResponsable {}
    function comprarObra(uint256 idToken) external payable{}
    function cambiarPrecioObra(uint256 idToken, uint128 nuevoPrecio) external {}
    function cambiarEstadoVentaObra(uint256 idToken) external {}
    function obtenerObrasPropietario(address propietario) external view returns (Obra[] memory) {}
    function listarTodasLasObras() external view returns (Obra[] memory) {}
}

```

**Figura 16.** Principales funciones del contrato inteligente

En la **Tabla 14** para una mejor comprensión se describe cada una de las funciones.

**Tabla 14.** Descripción de las funciones del contrato inteligente

<b>Función</b>	<b>Descripción</b>
Constructor	Es el constructor del contrato, inicializa una colección denominada 'Colección de Arte' y el símbolo 'ARTE', esto es parte del estándar ERC-721 para almacenar los tokens en una colección.
soloResponsable	Modificador de acceso que restringe la ejecución de ciertas funciones únicamente a los responsables asignados.
soloEncargado	Modificador de acceso que restringe la ejecución de ciertas funciones únicamente a los encargados asignados.
asignarEncargado()	Permite al propietario del contrato asignar un nuevo encargado.
removerEncargado()	Permite al propietario del contrato remover un encargado asignado.
asignarResponsable()	Permite a una dirección con el rol de encargado asignar un nuevo responsable.
removerResponsable()	Permite a una dirección con el encargado remover a un responsable.
crearProceso()	Permite a un responsable crear un nuevo proceso con el nombre.
completarEtapa()	Permite a un responsable registrar cada etapa del proceso especificado, se proporciona la URI (metadatos) para la etapa.
obtenerProcesosDelArtista()	Devuelve una lista de IDs de los procesos asociados a un artista especificado.
obtenerInfoProceso()	Devuelve información sobre un proceso específico
obtenerProcesoConEtapas()	Devuelve las etapas de un proceso en específico
crearObra()	Permite a un responsable crear una nueva obra, especificando una URI para el token, un precio y el ID del proceso asociado.
comprarObra()	Permite a un comprador adquirir una obra
cambiarPrecioObra()	Permite al propietario de la obra cambiar el precio de una obra específica.
cambiarEstadoVentaObra()	Permite al propietario de la cambiar el estado de venta de una obra específica
obtenerObrasPropietario()	Devuelve una lista de obras propiedad de un propietario especificado.

### 6.1.8.1. Pruebas unitarias del contrato inteligente

Las pruebas unitarias se automatizaron mediante la herramienta truffle, con la que se ejecutó un total de 18 pruebas unitarias obteniendo como resultado el tiempo de ejecución de cada prueba así como su validación y aprobación. Para obtener más información acerca de las pruebas unitarias revisar el **Anexo 8**. Plan de pruebas unitarias para el contrato inteligente SDVOA.

En la **Figura 17** se muestra la ejecución de las pruebas unitarias mediante truffle.

```
Contract: SDVOAContract
Roles
  ✓ Debería asignar encargado por el propietario (120ms)
  ✓ Debería asignar responsable por el encargado (127ms)
  ✓ Debería remover encargado por el propietario (117ms)
  ✓ Debería remover responsable por el encargado (135ms)
Proceso
  ✓ Debería crear un nuevo proceso (105ms)
  ✓ Debería completar una etapa del proceso (222ms)
  ✓ Debería crear una obra cuando todas las etapas están completadas (723ms)
  ✓ Debería obtener los procesos del artista (68ms)
  ✓ Debería obtener información de un proceso (92ms)
  ✓ No debería crear una obra con un uri existente (467ms)
Obra
  ✓ Debería comprar obra (120ms)
  ✓ Debería cambiar el precio de la obra (79ms)
  ✓ Debería cambiar el estado de venta de la obra (53ms)
  ✓ Debería listar todas las obras (43ms)
  ✓ Debería obtener las obras del propietario (407ms)
  ✓ No debería comprar una obra si no se tiene fondos suficientes (453ms)
  ✓ No debería cambiar el precio de la obra si no es propietario (48ms)
  ✓ No debería cambiar el estado de venta de la obra si no es propietario (81ms)

18 passing (14s)
```

**Figura 17.** Ejecución de pruebas unitarias mediante la herramienta truffle

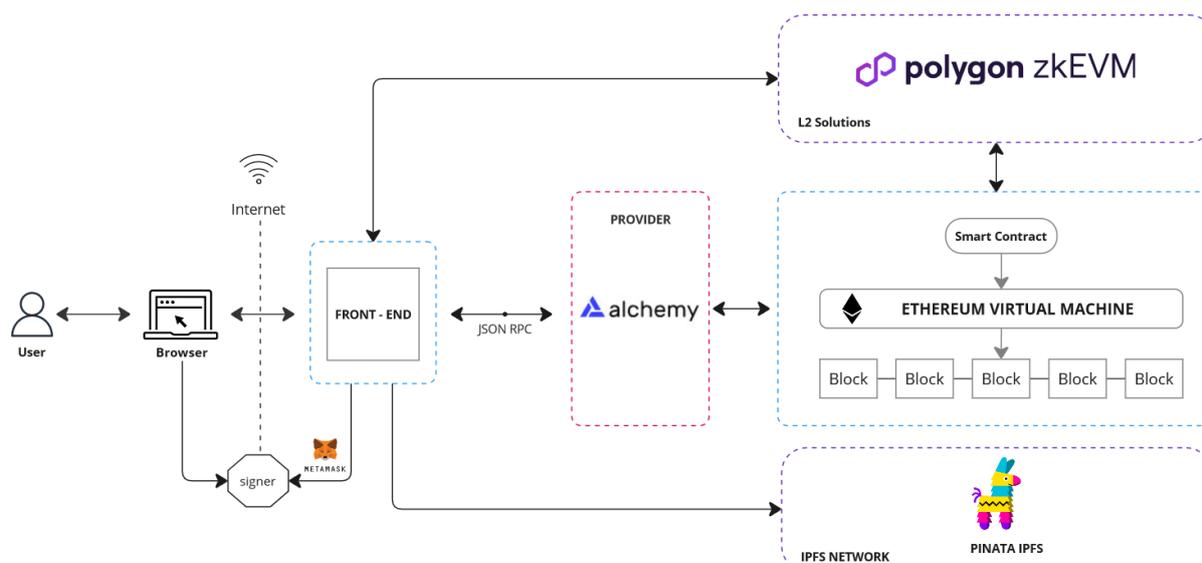
## 6.1.9. Diseño del subsistema de aplicaciones

En este punto cabe recalcar que para el diseño del subsistema de aplicaciones (sistema de difusión y ventas de obras de arte) el enfoque se centró principalmente en el desarrollo de las nuevas funcionalidades para la interacción con el contrato inteligente y a su vez con la cadena de bloques. A continuación, se presenta como se desarrolló el subsistema siguiente el enfoque mencionado.

### 6.1.9.1. Arquitectura de la aplicación descentralizada

Como se conoce una aplicación web tradicional es diferente a una aplicación descentralizada, por lo que fue necesario realizar una nueva arquitectura en la que se contemplan todos los elementos necesarios para la interacción entre el sistema de difusión y ventas de obras de arte con la cadena de bloques.

En la **Figura 18** se ilustra la arquitectura completa de la aplicación descentralizada.



**Figura 18.** Arquitectura completa de la aplicación descentralizada

La arquitectura muestra la interacción de los usuarios con la cadena de bloques través de un proveedor de terceros que es Alchemy, el cual proporciona un nodo para interactuar con el contrato inteligente desplegado en la red de prueba Polygon zkEVM. Cardona. Además, se hace uso Pinata IPFS para el almacenamiento de archivos, esto es primordial debido a que así se reducen costos al registrar información en la cadena de bloques. También se hace uso de Metamask como billetera de Ethereum para firmar y realizar cualquier transacción por parte de los usuarios.

### 6.1.9.2. Flujo de la dApp para dar trazabilidad a las obras de arte

En la **Figura 19** se presenta el diagrama de secuencia que explica el proceso completo de interacción entre la cadena de bloques y la dApp, el diagrama muestra la interrelación de los elementos de la dApp para asegurar la trazabilidad de las obras.

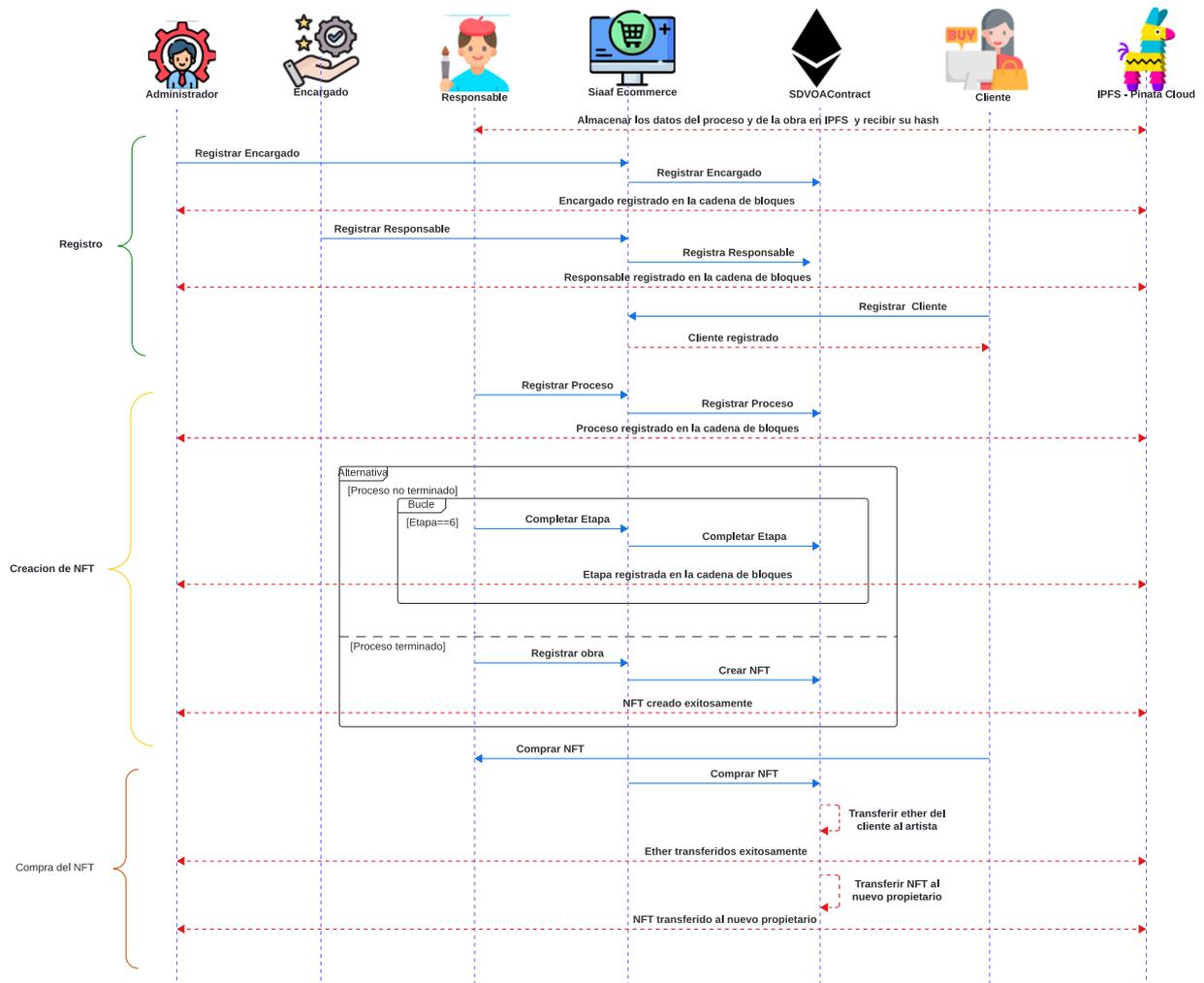


Figura 19. Diagrama de secuencia de la dApp

### 6.1.9.3. Patrón de diseño de la dapp

En este punto se rediseño el patrón de diseño original del sistema que se basa en el patron: **MVT** (Model, View, Template) propio del framework de Django para combinar las nuevas funcionalidades del sistema e interactuar con la cadena de bloques.

En la **Figura 20** se ilustra el rediseño del patrón de diseño para interacción con la cadena bloques.

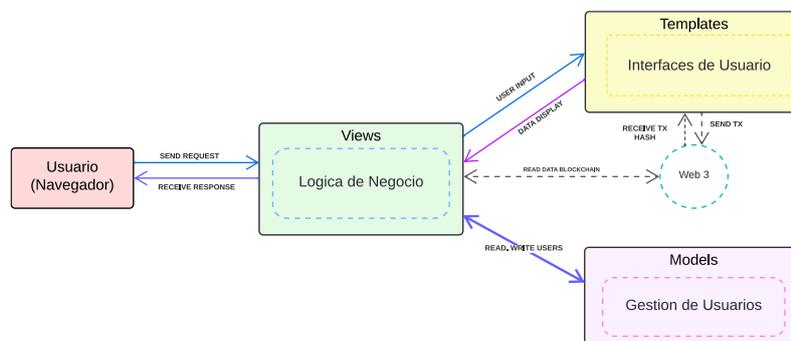


Figura 20. Patrón de diseño MVT (Model, Views, Templates)

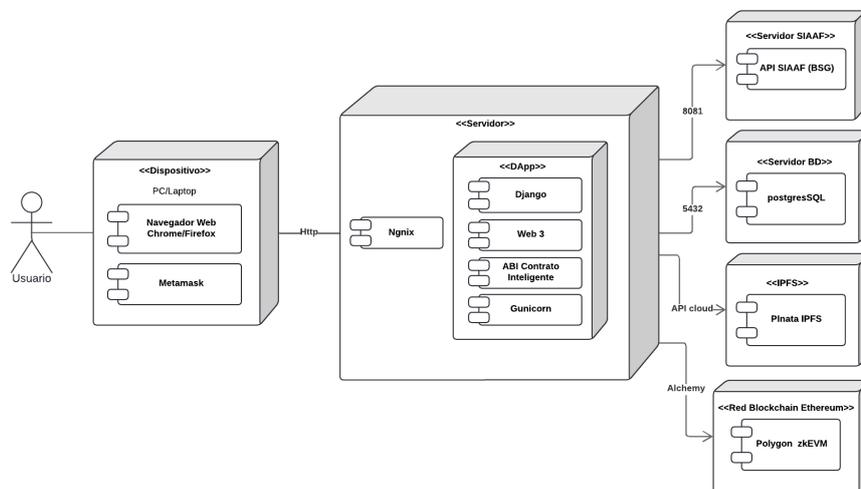
A continuación, se explica y describe cada uno de los componentes del patrón de diseño para la interacción con la cadena de bloques.

- **Model:** La dApp usa la cadena de bloques para almacenar la información, este componente se utilizó para la gestión de usuarios dentro de la dapp
- **Views:** Realiza procesos de lectura sobre la cadena de bloques mediante la librería web3 y las renderiza en las interfaces de usuario
- **Templates:** Consiste en todo el frontend de la dApp, está construido mediante como HTML, CSS, JS y Bootstrap. Además se usa metamask para realizar transacciones sobre la cadena de bloques.

#### 6.1.9.4. Arquitectura física de la dapp

Otro punto del diseño del subsistema de aplicaciones fue el rediseño de la arquitectura física del sistema para aprovechar al máximo los beneficios de la cadena de bloques, la arquitectura se enfoca en la distribución física de los nodos tomando en cuenta los nuevos componentes

En la **Figura 21** se muestra el rediseño de la arquitectura física del sistema original adaptado para la aplicación descentralizada.



**Figura 21.** Arquitectura física de la DApp

*Nota.* Adaptado de “Desarrollo de una Aplicación Web para la difusión y venta de obras de arte para la carrera de Artes Plásticas / Visuales de la Universidad Nacional de Loja” de C.E. Medina, M.A, Rojas, 2023, p, 47 (<https://dspace.unl.edu.ec/jspui/handle/123456789/28682>.)

### 6.1.10. Código del subsistema de aplicaciones

En este punto se muestran las principales funciones desarrolladas para dar trazabilidad a las obras de arte.

En la **Figura 22**, se muestran las nuevas funcionales desarrolladas

```
class ProcesoCrear(LoginRequiredMixin, UserPassesTestMixin, CreateView):...
usage new *
class ProcesoListar(LoginRequiredMixin, UserPassesTestMixin, ListView):...
usage new *
class ProcesoDetalle(LoginRequiredMixin, UserPassesTestMixin, DetailView):...
usage new *
class PasoCrear(LoginRequiredMixin, UserPassesTestMixin, CreateView):...
```

**Figura 22.** Funcionalidades del subsistema de aplicaciones

En la **Tabla 15** se describe cada una de las nuevas funcionalidades del sistema

**Tabla 15.** Descripción de las funcionalidades del subsistema de aplicaciones

Funcionalidad	Descripción
<b>CambiarPrecioToken()</b>	Actualiza el precio de un articulo
<b>ProcesoCrear()</b>	Permite a los responsables iniciar con un proceso de creación de obras de arte
<b>ProcesoListar()</b>	Muestra todos los procesos que posee el responsable
<b>ProcesoDetalle()</b>	Muestra los detalles de un proceso en especifico
<b>PasoCrear()</b>	Permite registrar un nuevo paso del proceso

### 6.1.11. Pruebas Unitarias

Para la validación de las nuevas funcionalidades se realizó un plan de pruebas (véase **Anexo 10**. Plan de pruebas unitarias para el subsistema de aplicaciones) con la finalidad de comprobar que el comportamiento de estas nuevas funcionalidades sea el adecuado.

En la **Tabla 16** se presenta un resumen de los casos de prueba definidos para validar las nuevas funcionalidades.

**Tabla 16.** Casos de prueba definidos para el subsistema de aplicaciones

ID	Caso de Prueba	Objetivo	Resultado Esperado
CP-0001	Registro de un usuario con una dirección de Ethereum	Validar que se registre un usuario con una dirección de Ethereum en el sistema	Los atributos del usuario coinciden con los valores especificados.
CP-0002	Registro Exitoso de un proceso de creación de obra de arte	Validar que se registre un proceso de creación de obra de arte correctamente	Los atributos del Proceso coinciden con los valores especificados y el estado es "INICIADO".

CP-0003	Registrar un paso de un proceso	Validar que se registre un paso correctamente	Los atributos del Paso coinciden con los valores especificados.
CP-0004	Registrar un artículo correctamente	Validar que se registre un artículo correctamente	Los atributos del artículo coinciden con los valores especificados y el estado es "PENDIENTE".

En la **Figura 23** se muestra la ejecución de pruebas automatizadas mediante Test Case.

```
(nft_ecommerce) alex@alex:~/Desktop/siaaf-ecommerce-develop$ python manage.py test
Found 14 test(s).
Creating test database for alias 'default'...
Modificar Cliente
System check identified no issues (0 silenced).
Se creo un nuevo articulo archivo con el nombre images.jpg
..Se creo un nuevo articulo con su proceso y con el titulo Obra de Arte
...Se creo un nuevo paso archivo con el nombre images.jpg
..Se creo un nuevo paso con el nombre CONCEPTUALIZACION
...Se creo un nuevo proceso con el nombre Proceso 1
..Se creo un nuevo usuario con una direccion web3
.
-----
Ran 14 tests in 0.484s
OK
Destroying test database for alias 'default'...
```

**Figura 23.** Resultado de las pruebas unitarias ejecutadas

### 6.1.12. Integrar, Probar y desplegar la Dapp

#### 6.1.12.1. Integrar la DApp

En este punto se realizó las Pruebas de Integración (véase **Anexo 11. Plan de pruebas de Integración de la DApp**) mediante las librerías Pytest y Web3, con la finalidad de validar la correcta interacción entre el subsistema de aplicaciones y el subsistema de contratos inteligentes. Pytest permitió realizar las pruebas automatizadas En la **Tabla 17** se muestran los casos de pruebas definidos para su posterior ejecución

**Tabla 17.** Resultados obtenidos de la ejecución de los casos de prueba

ID Caso de Prueba	Descripción y Funcionalidad Probada	Completado con Éxito	Observaciones
CP-001	Registrar Articulo	Sí	-
CP-002	Comprar un Articulo	Sí	-
CP-0032	Cambiar el Precio de un Articulo	Sí	-.
CP-004	Cambiar el Estado de Venta de un Articulo	Sí	-

CP-005	Obtener Historial de Transacciones	Sí	-
CP-006	Obtener Detalles de un Token	Sí	-
CP-007	Listar Todos los Articulos Disponibles	Sí	-
CP-008	Listar Articulos de un Usuario	Sí	-
CP-009	Obtener Proceso Asociado a un Articulo	Sí	-

### 6.1.12.2. Probar la DApp

En este apartado se muestran las Pruebas Funcionales realizadas a la DApp (**Véase Anexo 12. Plan de pruebas Funcionales para la DApp**) para garantizar que se cumplieron con los requisitos funcionales definidos en la **Tabla 7**.

En la **Tabla 18** se muestran los resultados obtenidos de la ejecución de las pruebas.

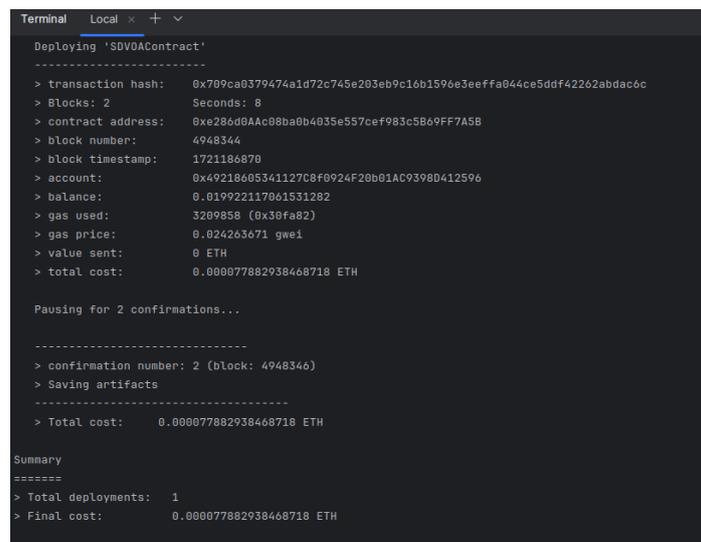
**Tabla 18.** Resultados de las pruebas funcionales realizadas

ID	Qué se probó	Requisito Funcional Asociado	Cumplido (Sí/No)	Observaciones
CP-0001	Registrar un Proceso completo de la creación de una obra de arte	RF01	Sí	Proceso registrado correctamente, se muestra mensaje de confirmación.
CP-0002	Registrar un artículo en la blockchain de Ethereum	RF02	Sí	Artículo publicado como NFT, se muestra el hash de la transacción y mensaje de confirmación.
CP-0003	Cambiar Precio de un artículo y Ver mis artículos	RF03 RF09	Sí	Precio actualizado correctamente, se muestra el hash de la transacción y mensaje de confirmación.
CP-0004	Cambiar Estado de venta de un artículo y ver mis artículos	RF04 RF09	Sí	Estado de venta actualizado correctamente, se muestra mensaje de confirmación.
CP-0005	Ver trazabilidad y detalles de artículo	RF06 RF07	Sí	Información detallada del artículo y su trazabilidad se muestran correctamente
CP-0006	Comprar Artículo	RF05	Sí	Compra realizada correctamente, se muestra el hash de la transacción y mensaje de confirmación.
CP-0007	Conectar Billetera	RF08	Sí	Billetera conectada correctamente, se muestra mensaje de confirmación y se activan funciones necesarias.

### 6.1.12.3. Desplegar la DApp

En primer lugar, se realizó el despliegue del contrato inteligente en la red de pruebas Polygon zkEVM Cardona mediante truffle y el nodo proporcionado por alchemy para interactuar con la red de pruebas. Cabe destacar que la red de pruebas solo requiere el pago del gas por cada transacción, es decir no cobra ETH adicional. Por lo que para el despliegue solo implicó el pago del gas necesario para asegurar la validez del contrato en la blockchain.

En la **Figura 24** se muestra el despliegue del contrato inteligente en la red de pruebas Polygon zk EVM cardona mediante truffle



```
Terminal Local x + v
Deploying 'SDVOAContract'
-----
> transaction hash: 0x709ca0379474a1d72c745e203eb9c16b1596e3effa044ce5ddf42262abdac6c
> blocks: 2 Seconds: 8
> contract address: 0xe286d8AAc08ba0b4035e557cef983c5869FF7A5B
> block number: 4948344
> block timestamp: 1721186870
> account: 0x4921860534112708f0924F20b01A093980412596
> balance: 0.019922117061531282
> gas used: 3209858 (0x30fa82)
> gas price: 0.024263671 gwei
> value sent: 0 ETH
> total cost: 0.000077882938468718 ETH

Pausing for 2 confirmations...

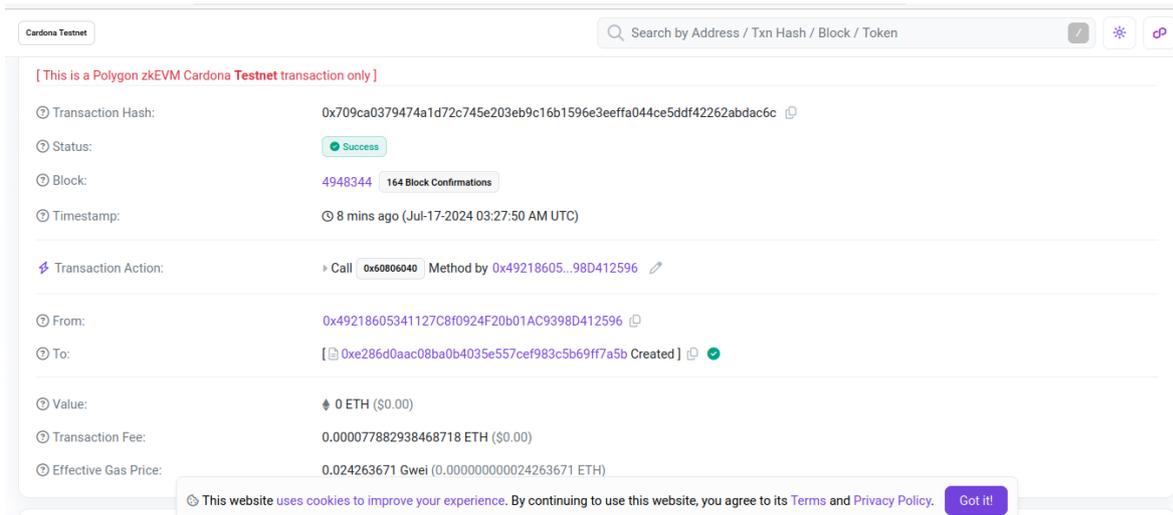
-----
> confirmation number: 2 (block: 4948346)
> Saving artifacts
-----
> Total cost: 0.000077882938468718 ETH

Summary
=====
> Total deployments: 1
> Final cost: 0.000077882938468718 ETH
```

**Figura 24.** Despliegue del contrato inteligente en la tesnet de polygon

El precio en ETH del despliegue es 0.000077882938468718 que equivale a \$ 0.26 ctvs, es importante conocer que el precio va a variar por la tasa de conversión y si el despliegue se lo hace sobre la red principal.

En la **Figura 25** se muestra detalles del contrato inteligente a través del explorador de Polygon zk EVM Cardona.



**Figura 25.** Contrato inteligente desplegado en la tesnet Polygon zkEVM Cardona

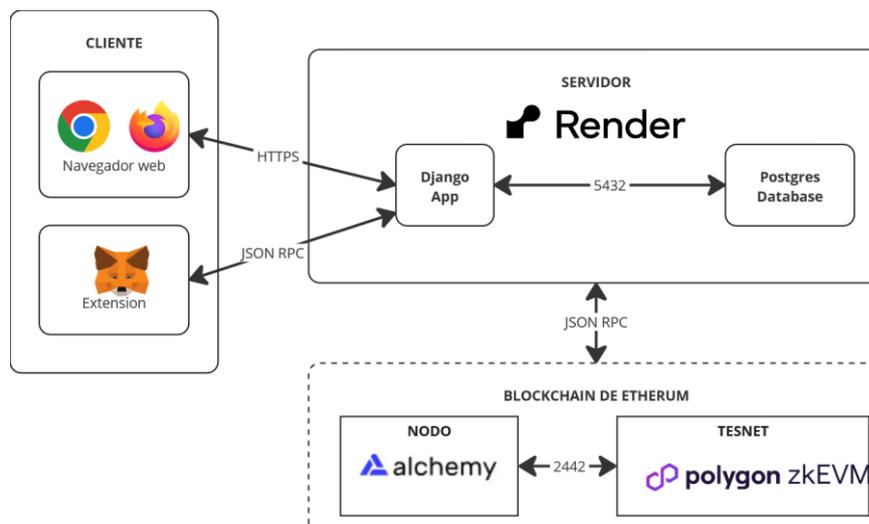
En la **Tabla 19** muestra el costo total del despliegue del contrato inteligente en la red de pruebas

**Tabla 19.** Costes del despliegue

Costo total en ETH	Costo total en dólares
0.000077882938468718	\$ 0.26 ctvs

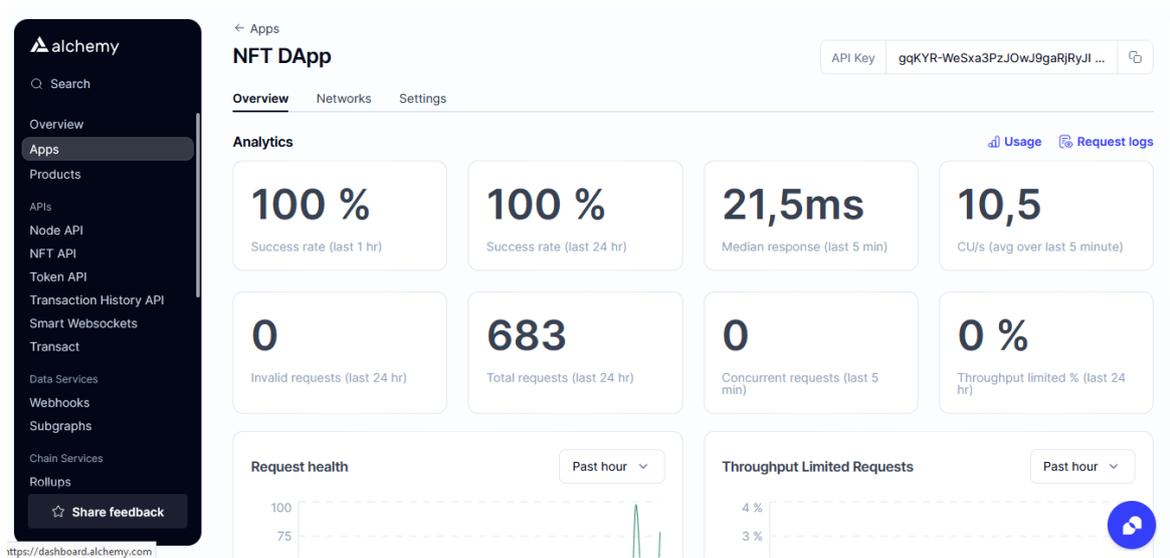
Para el despliegue de la aplicación se hizo uso de la herramienta Render Cloud, en la se despliego la imagen de Docker construida a partir de la aplicación junto con la base de datos, y para la interacción con la red de prueba Polygon zkEVM Cardona el nodo proporcionado por alchemy.

En la **Figura 26** se muestra la infraestructura de despliegue de la dApp



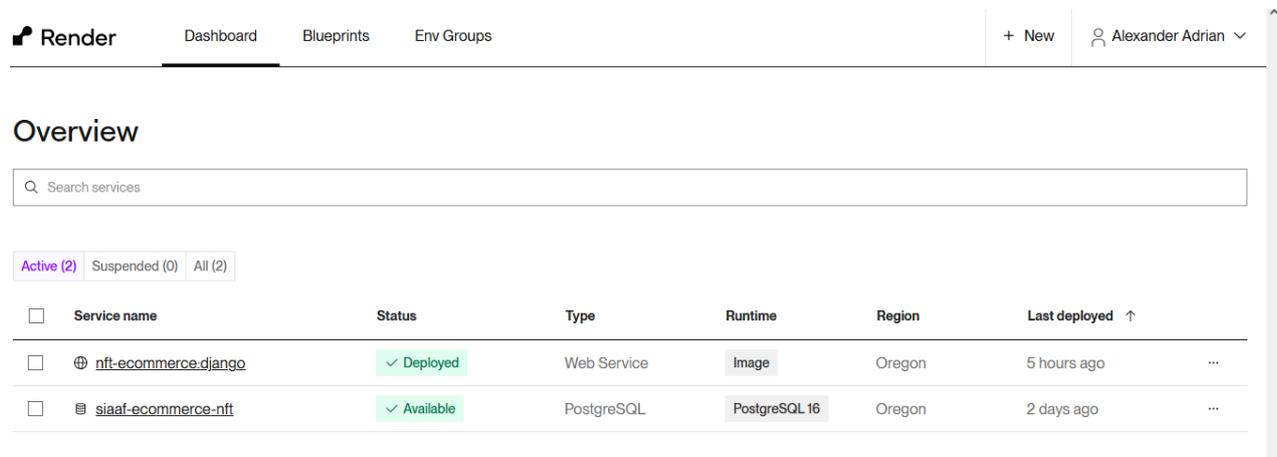
**Figura 26.** Infraestructura de despliegue de la dApp

En la **Figura 27** se muestra el proyecto creado en el proveedor alchemy, el mismo proporciona el nodo para interactuar con la red de pruebas de Polygon zk EVM



**Figura 27.** Proveedor Alchemy

En la **Figura 28** se muestran los servicios desplegados en render cloud, los servicios comprenden tanto la aplicación y la base de datos para la gestión de usuarios.



**Figura 28.** Servicios desplegados en Render Cloud

## 6.2. Objetivo 2: Evaluar el porcentaje de trazabilidad en el proceso de ventas de obras de arte a través del indicador clave de rendimiento KPI de trazabilidad especializado para la industria blockchain

### 6.2.1. Selección de los activos

En la **Figura 29** se muestran los 9 procesos de prueba registrados en la red de pruebas Polygon zk EVM Cardona por parte del primer responsable, de estos procesos en total se completaron solamente 7.

Nombre	Estado	Fecha de Creación	Acción
Proceso El Vuelo del Colibrí	FINALIZADO	2024-07-20 02:14:41	
Serenidad del Desierto	FINALIZADO	2024-07-20 02:34:02	
Campaña Toscana al Amanecer	FINALIZADO	2024-07-20 02:40:49	
Bosque Encantado	FINALIZADO	2024-08-06 21:09:14	
Proceso Horizonte Infinito	FINALIZADO	2024-08-06 21:15:53	
Proceso Reflejos del Alba	FINALIZADO	2024-08-06 21:25:19	
Proceso Metamorfosis Floral	FINALIZADO	2024-08-06 21:35:22	
Proceso Visiones Oníricas	EN PROCESO	2024-08-06 21:56:15	
Silencio Invernal	EN PROCESO	2024-08-06 22:00:02	

Mostrando registros del 1 al 9 de un total de 9 registros

Anterior 1 Siguiente

**Figura 29.** Procesos de prueba registrados en la cadena de bloques por el primer responsable

De igual manera en la **Figura 30** se presentan el resto de procesos registrados de los cuales se completaron solamente 6.

Nombre	Estado	Fecha de Creación	Acción
Proceso de Bosque Encantado en Primavera	FINALIZADO	2024-07-20 01:10:49	
Proceso Reflejos de una Ciudad Nocturna	FINALIZADO	2024-07-20 01:50:10	
Proceso Ecos del Pasado	FINALIZADO	2024-08-06 17:00:01	
Sueños Acuáticos	FINALIZADO	2024-08-06 20:20:35	
Ritmos Urbanos	FINALIZADO	2024-08-06 20:43:19	
Proceso Cielo Nocturno	INICIADO	2024-08-06 22:02:47	
Marina Tranquila	INICIADO	2024-08-06 22:03:38	
Danza del Viento	EN PROCESO	2024-08-06 22:04:36	
Proceso Reflejos Urbanos	FINALIZADO	2024-08-06 22:06:22	

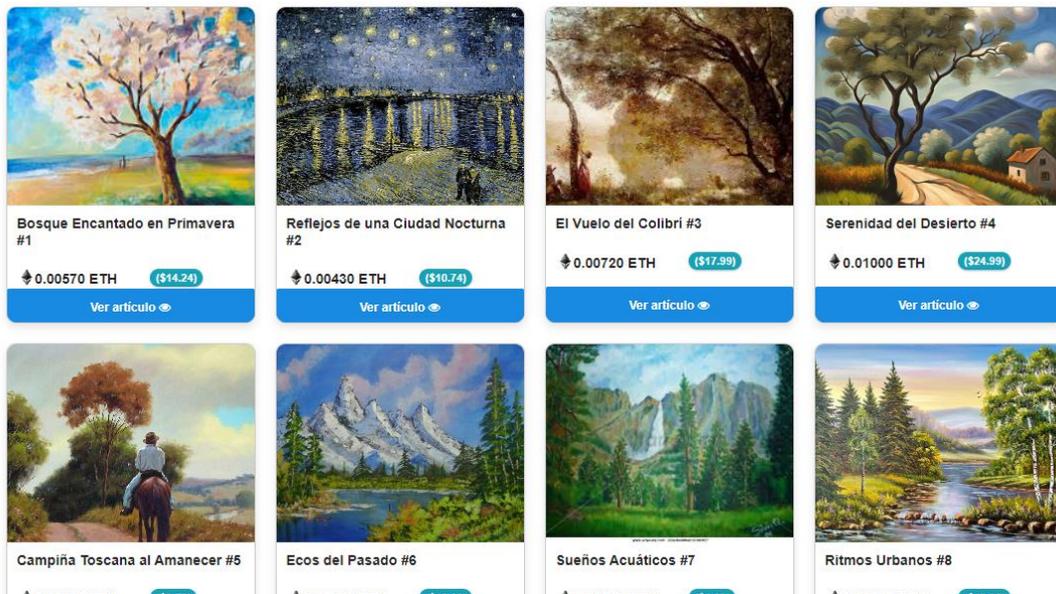
Mostrando registros del 1 al 9 de un total de 9 registros

Anterior 1 Siguiente

**Figura 30.** Procesos de prueba registrados en la cadena de bloques por parte del segundo responsable

En la **Figura 31** se muestran parte de las obras de arte en su representación como NFT resultados de los 13 procesos completados y que se encuentran registradas en la red de pruebas Polygon zk EVM Cardona <sup>1</sup>.

<sup>1</sup> [Colección de Arte NFT](#)



**Figura 31.** Obras de arte resultado de los 13 procesos completados.

En la **Figura 32** se muestran parte de las obras de arte representadas como NFT registradas en la cadena de bloques.

	<a href="#">0x8d9e85d013...</a>		<a href="#">0x2585a6c7</a>	5033071	2 days ago	<a href="#">0x00000000...00000000</a>			<a href="#">0x8a563912...Aa53bf542</a>			#5
	<a href="#">0x43f6af29d3d...</a>		<a href="#">0x2585a6c7</a>	5032641	2 days ago	<a href="#">0x00000000...00000000</a>			<a href="#">0x8a563912...Aa53bf542</a>			#4
	<a href="#">0xa56c2764cb...</a>		<a href="#">0x2585a6c7</a>	5032465	2 days ago	<a href="#">0x00000000...00000000</a>			<a href="#">0x8a563912...Aa53bf542</a>			#3
	<a href="#">0xece3b84762...</a>		<a href="#">0x2585a6c7</a>	5031883	2 days ago	<a href="#">0x00000000...00000000</a>			<a href="#">0x1c940F1c...DA36e5B8e</a>			#2
	<a href="#">0x18c30cd3cc...</a>		<a href="#">0x2585a6c7</a>	5031531	2 days ago	<a href="#">0x00000000...00000000</a>			<a href="#">0x1c940F1c...DA36e5B8e</a>			#1

**Figura 32.** Obras de arte en su representación como NFT en la cadena de bloques

Sobre cada NFT se realizaron transacciones de compra, mismas que se registraron en la cadena de bloques para proporcionar un historial completo sobre el proceso de creación y las transacciones de cada obra de arte.

En la **Figura 33** se muestran parte de las diferentes transacciones de compra sobre los NFT.

Transaction Hash	Method	Block	Age	From	To	TokenID
<a href="#">0xada3831ddc...</a>	0x3da40ee5	5033641	2 days ago	<a href="#">0x99a8E39b...D4F00513b</a>	<a href="#">0x4Cd7e880...3eF564032</a>	#4
<a href="#">0x085f7d00186...</a>	0x3da40ee5	5033628	2 days ago	<a href="#">0x99a8E39b...D4F00513b</a>	<a href="#">0x4Cd7e880...3eF564032</a>	#2
<a href="#">0x6acb1f8b6f7...</a>	0x3da40ee5	5033596	2 days ago	<a href="#">0x4Cd7e880...3eF564032</a>	<a href="#">0x99a8E39b...D4F00513b</a>	#3
<a href="#">0x9292b7cfa21...</a>	0x3da40ee5	5033584	2 days ago	<a href="#">0x4Cd7e880...3eF564032</a>	<a href="#">0x99a8E39b...D4F00513b</a>	#5
<a href="#">0xd06356f34fa...</a>	0x3da40ee5	5033573	2 days ago	<a href="#">0x4Cd7e880...3eF564032</a>	<a href="#">0x99a8E39b...D4F00513b</a>	#1
<a href="#">0x8fc994aab62...</a>	0x3da40ee5	5033449	2 days ago	<a href="#">0x8a563912...Aa53bf542</a>	<a href="#">0x99a8E39b...D4F00513b</a>	#4
<a href="#">0x9d63eb1da5...</a>	0x3da40ee5	5033441	2 days ago	<a href="#">0x1c940F1c...DA36e5B8e</a>	<a href="#">0x99a8E39b...D4F00513b</a>	#2
<a href="#">0xb94cf313ba4...</a>	0x3da40ee5	5033371	2 days ago	<a href="#">0x8a563912...Aa53bf542</a>	<a href="#">0x4Cd7e880...3eF564032</a>	#3
<a href="#">0xf628c0584eb...</a>	0x3da40ee5	5033359	2 days ago	<a href="#">0x8a563912...Aa53bf542</a>	<a href="#">0x4Cd7e880...3eF564032</a>	#5

**Figura 33.** Transacciones de compra realizadas sobre los NFT

### 6.2.2. Recopilación de datos

En la **Tabla 20** se describen los eventos del contrato inteligente usados para recolectar la información necesaria y evaluar el porcentaje de trazabilidad.

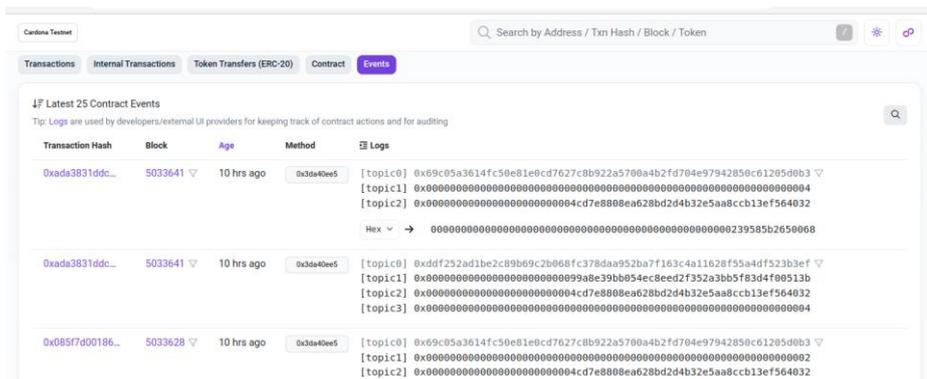
**Tabla 20.** Eventos del contrato inteligente

Evento	Descripción
ObraCreada	Evento emitido al crear una obra de arte.
ObraComprada	Evento emitido al realizar la compra de una obra de arte
ProcesoCreado	Evento emitido al crear un proceso relacionado con una obra de arte.
Etapacompletada	Evento emitido al completar una etapa en el proceso de la obra de arte.

En base a los eventos descritos, se hizo una verificación de que cada evento<sup>2</sup> fuera emitido mientras se realizaba todo el proceso de creación de obras de arte.

En la **Figura 34** se muestran parte de los eventos emitidos durante el proceso de creación de la obra de arte junto con las transacciones de compra, los eventos se encuentran registrados en la red de pruebas de Polygon zkEVM Cardona lo que permitió verificar de manera directa la emisión de esos eventos.

<sup>2</sup> [Eventos](#)



**Figura 34.** Parte de los eventos emitidos por el contrato inteligente

Para recolectar la información de los eventos emitidos se desarrollaron varias funciones en Python para automatizar este proceso. Las funciones realizan acciones de lectura sobre la red de Polygon zk EVM Cardona.

En la **Figura 35** se muestra la función desarrollada para obtener de la cadena de bloques cada uno de los eventos emitidos por su nombre.

```
def get_events(event_name):
    """Obtiene todos los eventos de un tipo específico."""
    try:
        return contract.events[event_name].create_filter(fromBlock=0, toBlock='latest').get_all_entries()
    except Exception as e:
        print(f"Error al obtener los eventos: {e}")
}
```

**Figura 35.** Función para obtener todos los eventos emitidos por su nombre

En la **Figura 36** se muestra la función para recopilar los datos de cada evento emitido y hacer un filtrado aplicando los criterios de selección previamente definidos obteniendo así aquellas obras que fueron rastreadas con éxito.

```
def filtrar_datos():
    try:
        #Recopilación de los eventos
        obras_creadas = get_events('ObraCreada')
        obras_compradas = get_events('ObraComprada')
        procesos_creados = get_events('ProcesoCreado')
        etapas_completadas = get_events('EtapasCompletada')

        total_obras = len(obras_creadas)

        #Aplicación de criterios de selección
        obras_rastreadas_exito = sum(
            any(compra['args']['idToken'] == obra['args']['idToken'] for compra in obras_compradas) and
            any(proceso['args']['procesoId'] == obra['args']['idProceso'] for proceso in procesos_creados) and
            any(etapa['args']['procesoId'] == obra['args']['idProceso'] for etapa in etapas_completadas)
            for obra in obras_creadas
        )

        obras_no_rastreadas = total_obras - obras_rastreadas_exito
```

**Figura 36.** Función para filtrar los datos en base a los criterios de selección

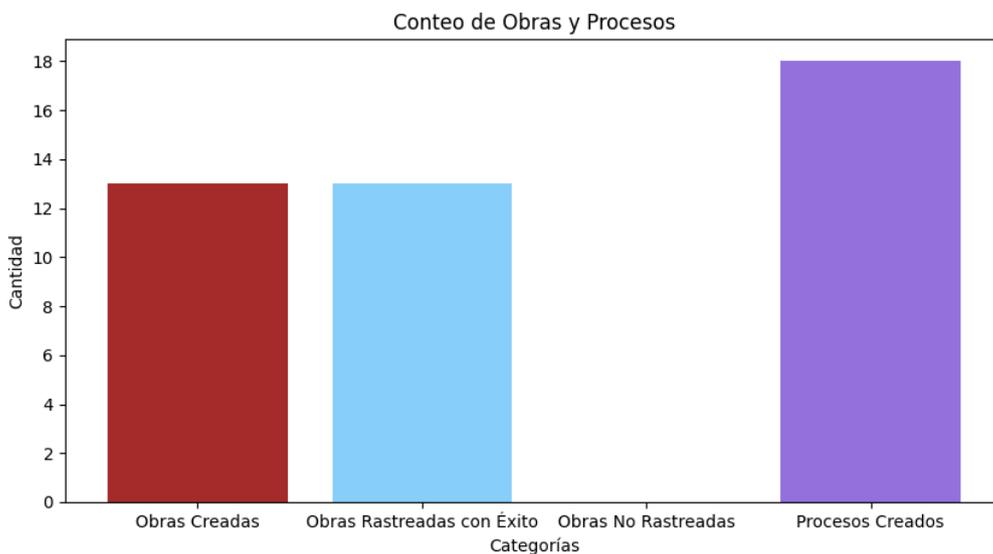
La función mostrada en la **Figura 36** recolecta todos los eventos emitidos descritos en la **Tabla 20** , para luego realizar un filtrado en base a los criterios de selección definidos y finalmente obtener aquellas obras rastreadas con éxito.

En base a todo lo realizado, como resultado en la **Figura 37** se muestran las 13 obras de arte seleccionadas para la evaluación junto con los criterios de selección La columna “id\_token” representa que la obra se encuentra registrada en la cadena de bloques, mientras que la columna “proceso\_asociado” indica que la obra está asociada a un proceso, mismo que ha sido completado satisfactoriamente como se observa en la columna “etapas\_completadas”.

id_token	comprada	num_comprada	proceso_asociado	etapas_completas	num_etapas_completas	total_etapas
1	True	2	True	True	6	6
2	True	2	True	True	6	6
3	True	2	True	True	6	6
4	True	2	True	True	6	6
5	True	3	True	True	6	6
6	True	1	True	True	6	6
7	True	1	True	True	6	6
8	True	1	True	True	6	6
9	True	1	True	True	6	6
10	True	1	True	True	6	6
11	True	1	True	True	6	6
12	True	1	True	True	6	6
13	True	1	True	True	6	6

**Figura 37.** Descripción de cada obra de arte rastreada con éxito junto con los criterios de selección

Por consecuencia, en la **Figura 38** se muestra de forma gráfica el conteo realizado internamente por las funciones aplicando todo el proceso de recolección y filtrado de datos.



**Figura 38.** Conteo de las obras registradas en la cadena de bloques y los procesos

### 6.2.3. Establecimiento de los Criterios de selección

A continuación, se describen los criterios que se consideraron para elegir aquellas obras que fueron rastreadas exitosamente. Estos criterios se basan en la naturaleza propia de la blockchain para proporcionar trazabilidad a los activos.

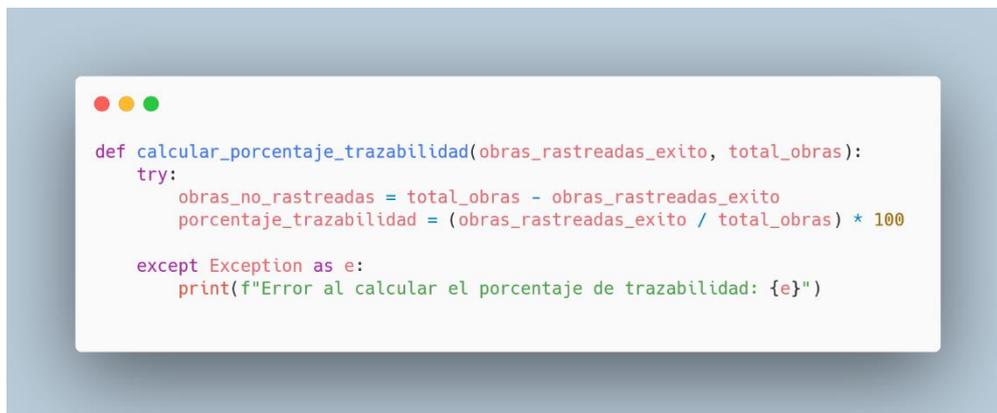
- ✓ **Registro en la cadena de bloques:** Cada obra de arte (NFT) registrada en la cadena de bloques proporciona una prueba verificable de su existencia y propiedad.
- ✓ **Proceso completo:** Cada obra de arte debe junto con su proceso de creación proporcionar un historial detallado y verificable de la creación de cada obra.
- ✓ **Transacción de Venta Registrada:** El registro de al menos una transacción de venta sobre el activo proporciona una prueba verificable de transferencia de propiedad, con el registro se demuestra también su autenticidad y proporciona un historial confiable de las transacciones

#### 6.2.4. Cálculo del Porcentaje de Trazabilidad

En este punto se presenta los resultados experimentales obtenidos al evaluar el porcentaje de trazabilidad con los datos recopilados y aplicando la fórmula descrita a continuación.

$$\text{Porcentaje de trazabilidad} = \frac{\text{Numero de activos rastreados con exito}}{\text{Numero total de activos}} * 100$$

En la **Figura 39** se muestra la función para calcular el porcentaje de trazabilidad en la que se aplica la formula y usando las variables obtenidas durante la recopilación de datos como el total de obras rastreadas con éxito y el total de obras de arte (NFT) creados.



```

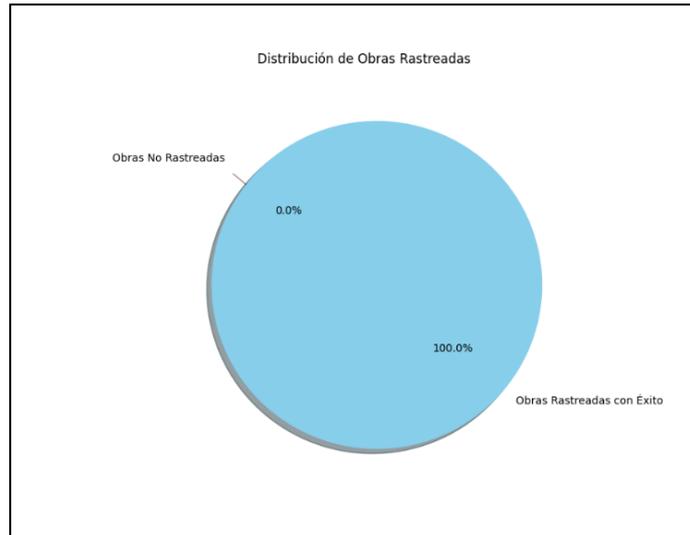
def calcular_porcentaje_trazabilidad(obras_rastreadas_exito, total_obras):
    try:
        obras_no_rastreadas = total_obras - obras_rastreadas_exito
        porcentaje_trazabilidad = (obras_rastreadas_exito / total_obras) * 100

    except Exception as e:
        print(f"Error al calcular el porcentaje de trazabilidad: {e}")

```

**Figura 39.** Función para calcular el porcentaje de trazabilidad

En la **Figura 40** se muestra el porcentaje de trazabilidad obtenido (100 %) con las 13 obras de arte seleccionadas este número, aunque pequeño, es suficiente para proporcionar una evaluación temprana de la capacidad de la dApp para proporcionar trazabilidad.



**Figura 40.** Porcentaje de trazabilidad obtenido durante la evaluación.

### **6.2.5. Interpretación de los resultados**

El porcentaje de trazabilidad del 100 % obtenido indica que de las 13 obras de arte seleccionadas para la evaluación fueron registradas con éxito, y el seguimiento sobre su ciclo de vida (desde su creación hasta su venta) fue completo. Además, refleja que cada transacción, cambio de propiedad y otro evento relacionado con las 13 obras de arte se registró de forma inmutable, asegurando que la información puede ser verificada en cualquier momento aspecto fundamental para artistas y compradores debido a que pueden verificar la procedencia de la obra.

## 7. Discusión

### 7.1. Primer objetivo: Integrar la tecnología Blockchain Ethereum en el sistema actual de ventas de obras de arte mediante la metodología ABCDE (Agile Blockchain DApp Engineering)

El uso de la tecnología blockchain para proporcionar a las obras de arte características como la unicidad, verificabilidad, trazabilidad, transparencia y dificultad para ser manipuladas, aspectos que las obras de arte tradicionales no poseen han sido abordadas por estudios previos como [50], [53], [54], [55], [56], [57]. Los estudios destacan el uso de la tecnología blockchain como una manera de garantizar la transparencia e inmutabilidad de los registros de cada obra de arte y mediante el uso de los contratos inteligentes gestionar la propiedad, autenticidad y transacciones de manera segura y transparente. Estas afirmaciones han sido corroboradas durante la ejecución del trabajo de integración curricular y por los resultados obtenidos, demostrando que la implementación de estas tecnologías en la plataforma de ventas de obras de arte de la carrera de artes plásticas permite proporcionar un historial completo, detallado y verificable de la procedencia de cada obra.

Sin embargo, a diferencia de los estudios previos mencionados (a excepto de [50] que proporciona un historial de transacciones que es público y accesible), que se centran únicamente en el uso de los NFT para rastrear la propiedad y autenticidad de las obras de arte, nuestro enfoque va mucho más allá permitiendo el seguimiento de cada obra desde la primera etapa de su creación, proporcionando un registro más completo y detallado de la procedencia de la obra, lo que no fue plenamente explorado en esas investigaciones.

Por otro lado, la metodología ABCDE demostró ser efectiva para integrar la tecnología blockchain en la plataforma de difusión y ventas de obras de arte mediante sus pasos estructurados permitieron el desarrollo del contrato inteligente utilizado para proporcionar trazabilidad a las obras y la interacción efectiva de la plataforma con la cadena de bloques. No obstante, una limitación durante su aplicación fue el diseño del subsistema de aplicaciones, debido a la necesidad de rediseñar y eliminar alguna componente del sistema original para adaptar la arquitectura descentralizada diseñada para permitir la interacción con el contrato inteligente y la vez con la cadena de bloques. Cabe mencionar que la metodología no se aplica en ninguno de los estudios previos, indicando usaron metodologías ágiles tradicionales cuyo enfoque no se centra en el desarrollo sobre la cadena de bloques, lo que puede generar problemas por la complejidad y especificidad técnica que la cadena de bloques y los contratos inteligentes requieren.

Otra limitación notable durante el desarrollo de la aplicación descentralizada fue el manejar la forma en que se realizan las transacciones, debido a la complejidad que tiene la librería web3 de Python para realizar las transacciones, ya que se necesita de las claves privadas de las billeteras de los usuarios que puede representar cierta desconfianza sobre el usuario, es por ello que la librería solo de uso para realizar acciones de lectura sobre la cadena de bloques mientras que para realizar las transacciones se utilizó Metamask que es más seguro y fácil de utilizar.

## **7.2. Segundo objetivo: Evaluar el porcentaje de trazabilidad en el proceso de ventas de obras de arte a través del indicador clave de rendimiento KPI de trazabilidad especializado para la industria blockchain**

Hasta el momento de la ejecución del Trabajo de integración curricular, no se han encontrado estudios previos que utilicen el indicador clave de rendimiento (KPI) especializado para la blockchain para medir la trazabilidad. Sin embargo, existen estudios que abordan el uso de la tecnología blockchain para la trazabilidad de las obras de arte, como es el caso de [50] que destaca que a través de la tecnología blockchain mejora la trazabilidad de cada obra al proporcionar un registro inmutable de cada transacción y cambio de propiedad asegurando que la información de cada pieza sea accesible y verificable. Y además proporciona un seguimiento en tiempo real de las obras de arte. Por otro lado [1] proporciona trazabilidad a las obras de arte mediante el uso de los NFT para registrar las transacciones en la blockchain proporcionando un acceso transparente a la información sobre la propiedad y el historial de ventas de las obras de arte. Estos estudios coinciden con los resultados experimentales obtenidos, aunque no se mencionen métricas para evaluar su efectividad.

El porcentaje obtenido de la evaluación refleja que el 100 % de las obras de arte en su representación como NFT fueron rastreadas exitosamente, es decir la información de las 13 obras seleccionadas se registró de manera inmutable y verificable en la cadena de bloques, permitiendo la recopilación y análisis de cada evento relacionado con la obra a través de los eventos emitidos por el contrato inteligente y la lectura de los datos mediante las funciones desarrolladas en Python. La naturaleza inmutable de la cadena de bloques aseguró que la información recolectada sea precisa y confiable, mientras que la automatización de la recopilación y análisis minimizó los errores que pueden ser causados por las personas, asegurando la consistencia de la evaluación. Sin embargo, este método está sujeto a limitaciones, como la dependencia de la exactitud y completitud de los eventos emitidos para garantizar la precisión del cálculo, cualquier error dentro de la aplicación descentralizada que implique el registro de datos en la cadena de bloques podría afectar los resultados. Otra

limitante significativa en la evaluación del porcentaje de trazabilidad, es la dependencia de la exactitud de la información registrada en la cadena de bloques, los responsables (artistas) deben ser transparentes en el proceso de creación de cada obra de arte para asegurar credibilidad y confianza del trabajo realizado.

## 8. Conclusiones

- A través de la implementación de la tecnología blockchain en el sistema de difusión y ventas de obras de arte, se logró obtener un porcentaje de trazabilidad del 100% al evaluar el indicador clave de rendimiento (KPI) correspondiente, lo que indica que cada etapa del proceso de creación de obras de arte junto con sus transacciones de venta se registró y documentaron en la red de pruebas Polygon zkEVM Cardona, permitiendo la validación de la procedencia y las transacciones de cada obra.
- La tecnología blockchain representa una solución integral para proporcionar trazabilidad a las obras de arte en distintas plataformas digitales. Los resultados experimentales obtenidos han demostrado que, desde el momento de la creación de las obras de arte se genera un registro completo y verificable. Además, el uso de contratos inteligentes para la gestión de cada obra de arte asegura que cada transacción se registre automáticamente, eliminando la posibilidad de que los datos sean manipulados.
- El enfoque estructurado y ágil proporcionado por la metodología ABCDE permitió el paso del sistema de difusión y ventas de obras de arte, de una aplicación web tradicional a una aplicación descentralizada, abarcando cada fase del desarrollo de manera integral y ajustando cada uno de los componentes, de tal manera que la interacción del contrato inteligente con la aplicación sea efectiva, obteniendo así un beneficio sobre la aplicación al proporcionar la capacidad de verificar la procedencia de las obras de arte.
- El uso de librerías web 3.0 para interactuar con la cadena de bloques. permitió recolectar y analizar de forma automática los datos registrados en red de pruebas Polygon zkEVM Cardona facilitando el proceso de evaluación del porcentaje de trazabilidad a través del indicador clave de rendimiento (KPI) de trazabilidad especializado para blockchain.

## 9. Recomendaciones

Una vez culminado el Trabajo de Integración Curricular se recomienda:

- Utilizar la metodología ABCDE para el desarrollo de soluciones que integren la tecnología blockchain, dado que aborda cuestiones relacionadas con el diseño de aplicaciones descentralizadas, cuestiones que las metodologías ágiles tradicionales aprendidas durante la carrera no consideran.
- Que los estudiantes de la carrera de computación interesados en el uso de tecnología blockchain, exploren, analicen y profundicen acerca del tema, ya que la experiencia y el conocimiento adquirido durante su estudio le permitirán construir una base sólida para abordar desafíos reales en el campo laboral.
- Fomentar a los estudiantes de la carrera de computación el desarrollo de sus habilidades en tecnologías que comúnmente se utilizan para el desarrollo de aplicaciones descentralizadas, como el lenguaje de programación Solidity, el uso de redes de pruebas, así como la configuración de entornos de desarrollo y despliegue de aplicaciones descentralizadas.
- Utilizar este Trabajo de Integración curricular como punto de partida para desarrollar aplicaciones descentralizadas que abarquen otras áreas como la salud, educación, registros públicos, entre otros. Al cubrir diferentes áreas, se contribuirá al aumento de investigaciones a nivel local, promoviendo la innovación y adopción de tecnologías emergentes como la blockchain.

Para futuros trabajos se recomienda.

- Buscar y adoptar soluciones para mejorar la eficiencia en el acceso a los datos almacenados en la blockchain. Como, por ejemplo, The Graph para facilitar la conexión entre aplicaciones descentralizadas y los datos almacenados en blockchain mediante un protocolo avanzado de indexación y consulta optimizando el proceso de recuperación y análisis de datos.
- Explorar el uso de tecnologías complementarias, como el almacenamiento de datos off-Chain para reducir costos y mejorar el rendimiento de las aplicaciones descentralizadas. Además, establecer un punto de equilibrio entre la transparencia proporcionada por la blockchain y la privacidad necesaria para ciertos datos sensibles.

- Explorar plataformas alternativas a Ethereum, debido a que sus tarifas de gas son altas y el trabajo sobre bloques completos afecta negativamente a la economía y experiencia del usuario. Plataformas como Polygon, pueden ofrecer una solución efectiva para optimizar el gasto en operaciones y el rendimiento de las aplicaciones descentralizadas.
- Promover la interoperabilidad con grandes mercados de como OpenSea, para conectarse a un ecosistema más amplio y establecido, facilitando el acceso a gran cantidad de usuarios y aumentando la liquidez de los activos. Además, permitiría ampliar el alcance y la visibilidad de las obras de arte al proporcionar una infraestructura sólida para su compra, venta y gestión, optimizando la experiencia del usuario al ofrecerle un acceso más fluido y conveniente a una variedad de activos.
- Explorar soluciones para la verificación de cada etapa del proceso de creación de obras de arte antes de ser registrada en la cadena de bloques y de esta manera asegurar que los datos registrados sean relevantes y precisos.

## 10. Bibliografía

- [1] M. Bal and C. Ner, “NFTracer: A Non-Fungible Token Tracking Proof-of-Concept Using Hyperledger Fabric,” ArXiv, vol. abs/1905.04795, 2019, [Online]. Available: <https://api.semanticscholar.org/CorpusID:152282694>
- [2] M. M. Cristian Eduardo and M. A. Rojas Cobos, “Desarrollar una Aplicación Web para la difusión y venta de obras de arte para la carrera de Artes Plásticas / Visuales de la Universidad Nacional de Loja,” Universidad Nacional de Loja, 2023. [Online]. Available: <https://dspace.unl.edu.ec/jspui/handle/123456789/28682>
- [3] Read Robert, “Informe de Arte Online 2023,” Feb. 2024. [Online]. Available: <https://www.hiscox.es/sites/spain/files/2023-05/Hiscox%20online%20art%20trade%20report%202023.pdf>
- [4] D. A. Patil, V. Nagwekar, M. Arunkumar, S. Datta, and A. Angelin Florence, “An End to End Blockchain based Non- Fungible Token Platform for Buying and Selling Digital Arts,” in 7th International Conference on Trends in Electronics and Informatics, ICOEI 2023 - Proceedings, 2023. doi: 10.1109/ICOEI56765.2023.10126017.
- [5] A. Garg, A. Tyagi, A. Patel, and D. Raj, “BlockChain and Decentralized Apps,” May 2023.
- [6] A. Saxena, “Enhancing Security in Decentralized Applications: Strategies and Tools for Secure (dApp) Development and Deployment,” May 2024.
- [7] W. Zhang, L. Wei, S. Li, Y. Liu, and S. C. Cheung, “Darcher: Detecting on-chain-off-chain synchronization bugs in decentralized applications,” in ESEC/FSE 2021 - Proceedings of the 29th ACM Joint Meeting European Software Engineering Conference and Symposium on the Foundations of Software Engineering, 2021. doi: 10.1145/3468264.3468546.
- [8] M. Farnaghi and A. Mansourian, “Blockchain, an enabling technology for transparent and accountable decentralized public participatory GIS,” *Cities*, vol. 105, p. 102850, May 2020, doi: 10.1016/j.cities.2020.102850.
- [9] L. Carlozo, “Why CPAs need to get a grip on blockchain,” *Journal of Accountancy*, 2017.
- [10] R. Luque Lodeiro, “Blockchain: Estado del arte, tendencias y retos,” 2020, Accessed: Apr. 24, 2024. [Online]. Available: <http://hdl.handle.net/10651/56337>
- [11] J. J. B. Pérez, A. Queiruga-Dios, V. G. Martínez, and Á. M. del Rey, “Traceability of ready-to-wear clothing through blockchain technology,” *Sustainability (Switzerland)*, vol. 12, no. 18, 2020, doi: 10.3390/su12187491.
- [12] H. D. Arrasco Arteaga, “Tecnologías Blockchain: innovación empresarial y aplicación,” UPC, Escola Superior d’Enginyeries Industrial, Aeroespacial i Audiovisual de Terrassa, Departament d’Organització d’Empreses, 2021. [Online]. Available: <http://hdl.handle.net/2117/342840>
- [13] N. Medranda Morales, and M. Arcos Argudo, *Blockchain, criptoactivos y metaverso. Una aproximación teórica*. 2023. doi: 10.17163/abyaups.6.

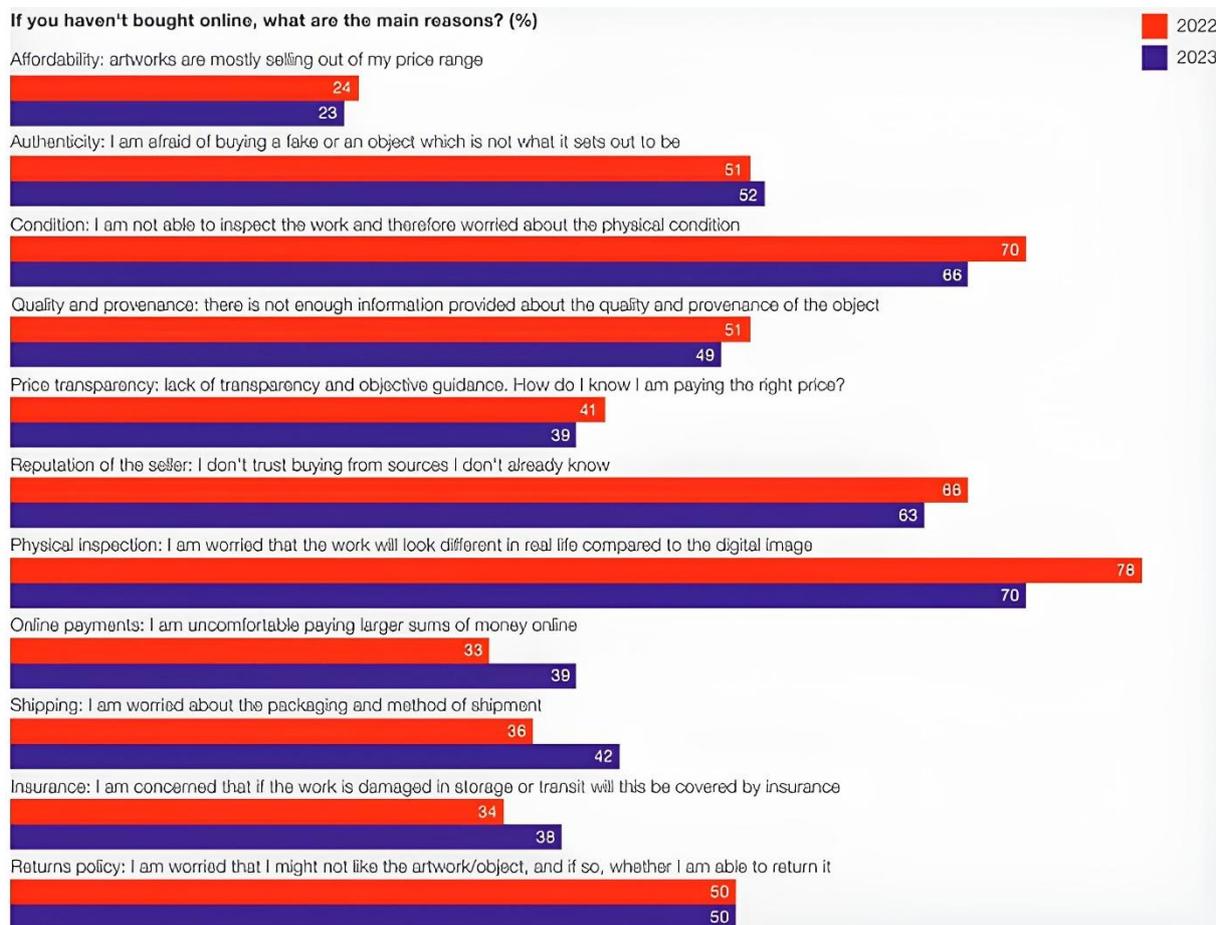
- [14] I. Gomez Lasala, "Blockchain. La revolucion en la industria," UPC, Escola Tècnica Superior d'Enginyeria Industrial de Barcelona, Departament d'Expressió Gràfica a l'Enginyeria, 2018. [Online]. Available: <http://hdl.handle.net/2117/122913>
- [15] N. Porxas and M. Conejero, "Tecnología blockchain : funcionamiento , aplicaciones y retos jurídicos relacionados," *Actualidad Jurídica Uría Menéndez*, 2018.
- [16] D. Vujičić, D. Jagodić, and S. Randić, "Blockchain technology, bitcoin, and Ethereum: A brief overview," in *2018 17th International Symposium on INFOTEH-JAHORINA, INFOTEH 2018 - Proceedings*, 2018. doi: 10.1109/INFOTEH.2018.8345547.
- [17] Victor. Miranda Palacios, "Explorando la blockchain de ethereum y el desarrollo de smartcontracts," *Sergioguillen.Com*, no. 710, 2018.
- [18] M. S. Ferdous, M. J. M. Chowdhury, M. A. Hoque, and A. Colman, "Blockchain Consensus Algorithms: A Survey," 2020.
- [19] S. Sarumathi, A. Raja, A. Kumar, A. Yadav, and F. Khan, "A Blockchain Based Decentralized NFT Marketplace," *International Journal of Advanced Research in Science, Communication and Technology*, pp. 575–581, May 2023, doi: 10.48175/IJARSCT-8545.
- [20] R. Norvill, B. Fiz Pontiveros, R. State, and A. Cullen, "Visual emulation for Ethereum's virtual machine," May 2018, pp. 1–4. doi: 10.1109/NOMS.2018.8406332.
- [21] A. Arias-Sánchez, "Estimación del coste del gas en transacciones de Ethereum mediante Deep Learning," 2022, Accessed: May 10, 2024. [Online]. Available: <http://hdl.handle.net/10609/147467>
- [22] A. Laurent, L. Brotcorne, and B. Fortz, "Transaction fees optimization in the Ethereum blockchain," *Blockchain: Research and Applications*, vol. 3, p. 100074, May 2022, doi: 10.1016/j.bcr.2022.100074.
- [23] A. Taibo Escarramán, "Seguridad en la Blockchain de Ethereum: explotación y mitigación de vulnerabilidades modernas en Smart Contracts," Madrid, May 2022. [Online]. Available: <https://oa.upm.es/70503/>
- [24] Y. Merrad, M. Habaebi, S. Toha, M. Islam, T. Gunawan, and M. Mesri Merad, "Cost-Effective Energy Demand Response Management System with a Smart Contracts-Based Optimal Power Flow Solution for Smart Grids," *Energies (Basel)*, vol. 15, May 2022, doi: 10.3390/en15124461.
- [25] I. Fetsyak, "Contratos inteligentes: análisis jurídico desde el marco legal español," *Revista Electrónica de Derecho de la Universidad de La Rioja (REDUR)*, no. 18, 2020, doi: 10.18172/redur.4898.
- [26] S. Alex Lipton, Stuart Levi, "An Introduction to Smart Contracts and Their Potential and Inherent Limitations | Insights | Skadden, Arps, Slate, Meagher & Flom LLP," *Harvard Law School Forum on Corporate Governance and Financial Regulation*, 2018.
- [27] C. D. Padilla, "La revolución blockchain y los smart contracts en el marco europeo," *Actualidad jurídica iberoamericana*, no. 16, pp. 1088–1109, 2022.
- [28] N. M. Alshahrani, M. L. M. Kiah, B. B. Zaidan, A. H. Alamoodi, and A. Saif, "A Review of Smart Contract Blockchain Based on Multi-Criteria Analysis: Challenges and Motivations," 2023. doi: 10.32604/cmc.2023.036138.

- [29] S. Wang, Y. Yuan, X. Wang, J. Li, R. Qin, and F.-Y. Wang, “An Overview of Smart Contract: Architecture, Applications, and Future Trends,” in 2018 IEEE Intelligent Vehicles Symposium (IV), 2018, pp. 108–113. doi: 10.1109/IVS.2018.8500488.
- [30] Z. Zheng et al., “An overview on smart contracts: Challenges, advances and platforms,” *Future Generation Computer Systems*, vol. 105, pp. 475–491, 2020, doi: <https://doi.org/10.1016/j.future.2019.12.019>.
- [31] F. Valeonti, A. Bikakis, M. Terras, C. Speed, A. Hudson-Smith, and K. Chalkias, “Crypto collectibles, museum funding and openGLAM: Challenges, opportunities and the potential of non-fungible tokens (NFTs),” *Applied Sciences (Switzerland)*, vol. 11, no. 21, 2021, doi: 10.3390/app11219931.
- [32] V. Armijos, “Que son los NFT, el DEFI y el Metaverso,” vol. 20000, p. <https://dialoguemos.ec/2022/06/que-son>, May 2022.
- [33] P. Fernández, S. Valera, and S. Muñoz-Viñas, “NFT y arte digital: nuevas posibilidades para el consumo, la difusión y preservación de obras de arte contemporáneo,” *Artnodes*, May 2021, doi: 10.7238/a.v0i28.386317.
- [34] B. F. Mendoza Bautista, L. F. Izquierdo Lozano, and A. C. Mendoza de los Santos, “Los NFT en el mercado electrónico y su impacto en el arte digital,” *SCIÉNDO*, vol. 25, no. 2, pp. 213–217, Jun. 2022, doi: 10.17268/sciendo.2022.026.
- [35] C. R. Romero Morocho, C. H. Flores Urgilés, J. A. Carrillo Zenteno, and D. P. Andrade Cárdenas, “Non-fungible token’s (NFT’s), Blockchain’s, oportunidades de desarrollo profesional, una mirada desde la Ingeniería de Sistemas,” *Pro Sciences: Revista de Producción, Ciencias e Investigación*, vol. 7, no. 47, pp. 36–49, Mar. 2023, [Online]. Available: <https://journalprosciences.com/index.php/ps/article/view/641>
- [36] I. D. de Figueiredo Novo, “Property-based testing of ERC-721 Ethereum smart contracts”.
- [37] Alex Ryzhkov, “Indicadores clave de rendimiento específicos de la industria para Blockchain,” Jul. 2024. [Online]. Available: <https://finmodelslab.com/es/blogs/kpi-metrics/blockchain-technology-solutions-kpi-metrics>
- [38] A. Chandiramani, “Management of Django Web Development in Python,” *Journal of Management and Service Science (JMSS)*, vol. 1, pp. 1–17, Aug. 2021, doi: 10.54060/JMSS/001.02.005.
- [39] J. Juneau, J. Baker, V. Ng, L. Soto, and F. Wierzbicki, “Web Applications With Django,” Aug. 2010, doi: 10.1007/978-1-4302-2528-7\_14.
- [40] S. Chen, S. Ahmmed, K. Lal, and C. Deming, “Django Web Development Framework: Powering the Modern Web,” *American Journal of Trade and Policy*, 2020, [Online]. Available: <https://api.semanticscholar.org/CorpusID:264578900>
- [41] C. Dannen, “Solidita Programming,” 2017, pp. 69–88. doi: 10.1007/978-1-4842-2535-6\_4.
- [42] R. Parizi, A. Singh, and A. Dehghantanha, “Smart Contract Programming Languages on Blockchains: An Empirical Evaluation of Usability and Security,” 2018, pp. 75–91. doi: 10.1007/978-3-319-94478-4\_6.
- [43] V. Tiwari, S. Singh, and V. Vadi, “Smart Contract Using Solidita (Remix -Ethereum IDE),” vol. 12, pp. 243–249, Aug. 2023, doi: 10.17148/IJARCCCE.2023.12253.

- [44] N. Sangeeta and S. Y. Nam, "Blockchain and Interplanetary File System (IPFS)-Based Data Storage System for Vehicular Networks with Keyword Search Capability," *Electronics (Basel)*, vol. 12, p. 1545, Aug. 2023, doi: 10.3390/electronics12071545.
- [45] R. Verma, N. Dhanda, and V. Nagar, "Application of Truffle Suite in a Blockchain Environment," 2023, pp. 693–702. doi: 10.1007/978-981-19-1142-2\_54.
- [46] S. Anwar, R. Tulsyan, S. Saha, and S. Sahana, "AnonChain: A Secure File Sharing Framework using IPFS Integrated Blockchain," *International Journal of Mathematical, Engineering and Management Sciences*, vol. 7, pp. 844–858, Aug. 2022, doi: 10.33889/IJMEMS.2022.7.6.053.
- [47] S. Khatal, J. Rane, D. Patel, P. Patel, and Y. Busnel, "FileShare: A Blockchain and IPFS Framework for Secure File Sharing and Data Provenance," 2021, pp. 825–833. doi: 10.1007/978-981-15-5243-4\_79.
- [48] S. Jain, "IPFS and NFTs," 2022, pp. 147–165. doi: 10.1007/978-1-4842-8975-4\_7.
- [49] N. Hung et al., "Revolutionizing Real Estate: A Blockchain, NFT, and IPFS Multi-platform Approach," 2023, pp. 68–73. doi: 10.1007/978-3-031-48316-5\_10.
- [50] Z. Wang, L. Yang, Q. Wang, D. Liu, Z. Xu, and S. Liu, "ArtChain: Blockchain-enabled platform for art marketplace," in *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 2019. doi: 10.1109/Blockchain.2019.00068.
- [51] A. W. Kusuma, A. A. Sukmandhani, and J. Ohliati, "Digital Artwork Marketplace Web Application Design using Blockchain Technology," 2023 *International Conference on Information Management and Technology (ICIMTech)*, pp. 493–498, 2023, [Online]. Available: <https://api.semanticscholar.org/CorpusID:264295017>
- [52] E. Ahmadih and N. El Madhoun, "Artwork NFTs for Online Trading and Transaction Cancellation," 2023 *Fifth International Conference on Blockchain Computing and Applications (BCCA)*, pp. 235–239, 2023, [Online]. Available: <https://api.semanticscholar.org/CorpusID:266196590>
- [53] C. H. Tantowibowo and W. C. Yau, "ArtProtect: Blockchain and NFC-based anti-counterfeit system for physical art," *IET Blockchain*, 2024, doi: 10.1049/blc2.12069.
- [54] G. Laud et al., "ManiacNFT : An Application for NFT Marketplace," *International Journal of Next-Generation Computing*, 2022, doi: 10.47164/ijngc.v13i5.916.
- [55] H. Agrawal, A. Bodhe, A. Sontakke, A. Shahane, R. Bihade, and S. Kashibai Navle, "Artcart: NFT Marketplace," *International Journal of Research Publication and Reviews*, vol. 3, no. 11, 2022.
- [56] Yogiraj Gutte, Aasit Vora, Yogesh Sharma, and Bhaskar Bhardwaj, "NFT Marketplace Based on Ethereum Blockchain," *International Journal of Advanced Research in Science, Communication and Technology*, 2022, doi: 10.48175/ijarsct-3729.
- [57] Aryan Dholi, Pratiksha Mandawade, Nisha Pagare, Kuldeep Bhangale, and Prof. Amol Nalge, "Blockchain Application Using NFT Marketplace," *International Journal of Advanced Research in Science, Communication and Technology*, 2023, doi: 10.48175/ijarsct-13162.

## 11. Anexos

### Anexo 1. Estudio realizado por Hiscox online Art a compradoras de arte en línea



**Figura 41.** Resultados del estudio realizado por Hiscox Online Art Trade.

El estudio reveló que el 70% de las personas no compran arte en línea debido al temor de que la obra se vea diferente en persona, seguido de un 66% que señaló la imposibilidad de ver la pieza antes de la compra, y un 63% que mencionó la falta de confianza en el vendedor. Sin embargo, un creciente 90% de nuevos compradores de arte y un 89% de compradores jóvenes consideran que la capacidad de rastrear con precisión la historia y procedencia de una obra de arte aumentaría su confianza en las compras en línea, lo que sugiere un cambio en las preocupaciones de los compradores hacia una mayor transparencia y autenticidad.

**Anexo 2.** Entrevista realizada al docente de la carrera de artes plásticas

**Entrevista a docente de la carrera de artes plásticas**

**Proyecto:** Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas

**Versión:** 1.0

**Fecha:** 13/04/2024

**1. ¿Cuál es el proceso para crear una obra de arte, ya sea física o digital, desde la idea hasta la finalización? ¿El proceso es el mismo para ambas o difiere?**

Para crear una obra de arte, primero hay que hacer la distinción de cuál es el estilo y también cuál es el medio técnico por materia. Ya sea físico o digital, el proceso nace de la idea del artista. Esta idea puede ser propia o acorde con algún evento específico, como un concurso. Luego, se determina el medio a utilizar, ya sea una fotografía, ilustración, escultura, pintura, etc. Si es una pintura, se define un boceto en un soporte bidimensional, se planifica el estilo, la composición y se incorpora el color. Luego, se traslada la idea al soporte definitivo, como un lienzo, se añaden detalles, texturas y acabados, y se da un título a la obra. En el caso de una obra digital, por ejemplo, una fotografía intervenida con Photoshop, se toma una fotografía y se alteran valores como matiz, saturación, luces y sombras, utilizando herramientas digitales para hacer recortes, aplicar filtros, etc. Aunque el proceso de creación difiere entre lo físico y lo digital, ambos siguen una planificación y ejecución cuidadosa acorde a sus respectivas técnicas y herramientas

**2. ¿Cuál es el proceso para vender una obra de arte?**

El proceso para vender una obra de arte puede variar. Una modalidad es a través de galerías, donde se deja la obra, se pone un precio, y el galerista añade un porcentaje y paga al artista según lo acordado. Otra modalidad es promocionar la obra en exposiciones o ferias, buscando atraer público interesado. También es posible realizar ventas directas a personas interesadas en el trabajo del artista. Además, con las tecnologías actuales, otra opción es promocionar la obra en mercados virtuales y plataformas en línea, permitiendo un alcance global. Por lo tanto, hay varios procesos y opciones para vender una obra de arte.

**3. ¿Qué criterios de calidad suelen considerarse al crear una obra de arte, ya sea en términos de originalidad, concepto o expresión?**

En términos de calidad, lo primero que se busca es observar la propuesta del creador: la esencia, las ideas y los discursos plasmados en la obra. El estilo y la forma en que se transmite también son importantes. La calidad puede variar según el estilo; en el realismo, se puede medir la representación de las figuras, mientras que en otros estilos más abstractos, la subjetividad juega un papel mayor. En general, se busca que la obra muestre una propuesta original y bien ejecutada del artista.

**4. ¿Cree que la trazabilidad de una obra de arte aumente la confianza del comprador y, por lo tanto, su valor?**

Sí, el conocimiento del proceso, el entorno y el contexto del artista puede desarrollar un vínculo con la obra y aumentar la confianza del comprador, lo cual puede contribuir a aumentar su valor. Conocer detalles sobre la creación y procedencia de la obra es valorado por muchos compradores.

**5. Si fuera comprador de una obra de arte en línea, ¿Estaría interesado en conocer detalles o ver el historial de procedencia de la obra?**

Personalmente, preferiría conocer al artista, su contexto, estilo y proceso de creación. Esto me ayudaría a desarrollar un vínculo con la obra. Aunque podría haber

excepciones en las que una obra me agradara sin interesarme por su creador, en general, consideraría importante tener un conocimiento sobre el creador y la procedencia de la obra.

**6. ¿La eliminación de intermediarios, como galerías, al vender y comprar obras de arte permite a los creadores conservar más ingresos y tener mayor control en la comercialización y distribución de sus obras? ¿Qué piensa al respecto?** Eliminar intermediarios como galerías permite a los creadores conservar más ingresos y tener mayor control sobre la comercialización y distribución de sus obras. Sin embargo, esto también implica que el artista debe encargarse del marketing y la estrategia de ventas, lo cual puede ser una dificultad. Cada modelo tiene sus responsabilidades, y el artista debe evaluar cuál le resulta más conveniente.

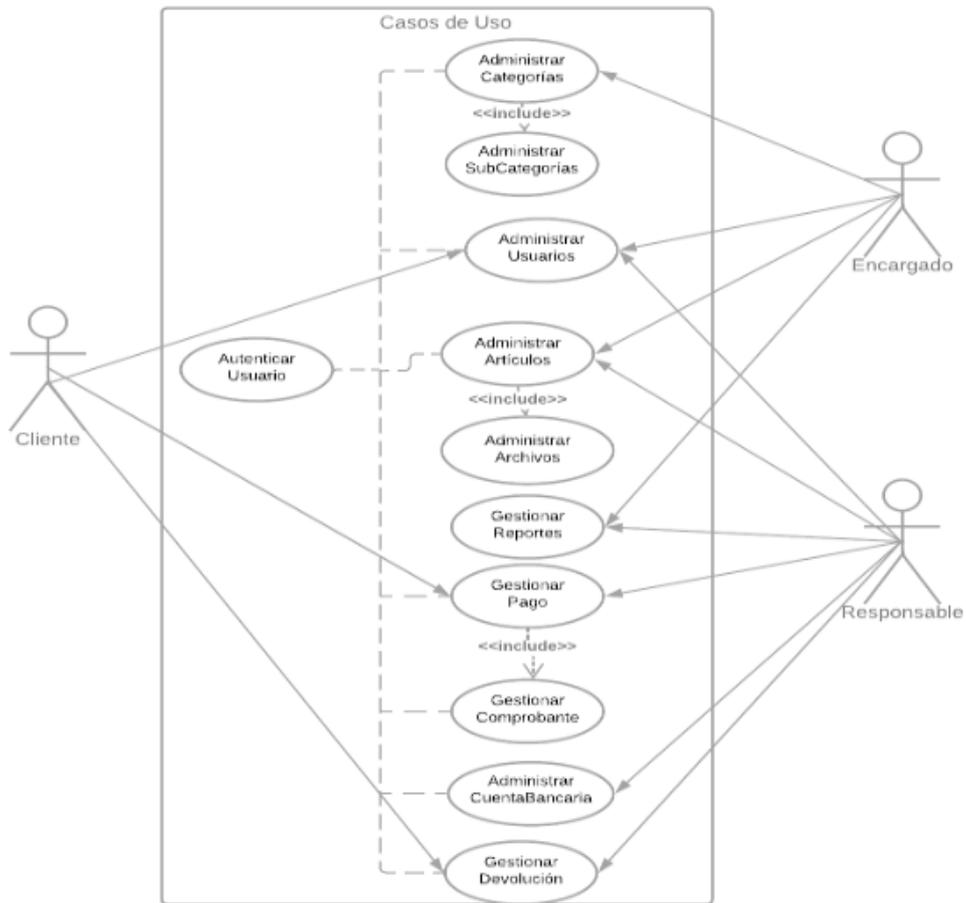
**7. ¿Cree que convertir una obra de arte, ya sea física o digital, en un NFT, pueda contribuir a la autenticidad de las mismas y por lo tanto aumentar su valor?**

La creación de NFTs puede contribuir a la autenticidad de las obras de arte, ya que ofrecen un certificado de autenticidad digital. Sin embargo, hay riesgos de especulación y precios inflados. La autenticidad de una obra digital vendrá del certificado, no de la imagen en sí, que puede ser ampliamente difundida y reinterpretada. Aunque los NFTs pueden aumentar el valor de una obra, es importante estar consciente de estos riesgos y el carácter experimental de este mercado

Para obtener más detalles de las reuniones ir a :

[https://drive.google.com/drive/folders/1OeTTi84LudRlp0e9IYRfkYdoCwyH6Yo2?usp=drive\\_link](https://drive.google.com/drive/folders/1OeTTi84LudRlp0e9IYRfkYdoCwyH6Yo2?usp=drive_link)

**Anexo 3.** Diagrama de casos de uso de la plataforma de difusión y venta de obras de arte



*Nota.* Tomado de “Desarrollo de una Aplicación Web para la difusión y venta de obras de arte para la carrera de Artes Plásticas / Visuales de la Universidad Nacional de Loja” de C.E. Medina, M.A, Rojas, 2023, p, 34 ([https://dspace.unl.edu.ec/jspui/handle/123456789/28682.](https://dspace.unl.edu.ec/jspui/handle/123456789/28682))

---

---

**Especificación de requisitos de software**

**Proyecto:** Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas

---

20/05/2024

### Ficha del documento

<b>Versión</b>	<b>Fecha</b>	<b>Cambios</b>	<b>Motivos del Cambio</b>
1.0	08/05/2023	Requisito no funcional de usabilidad (RNF02)	El enfoque debe ir a mostrar la trazabilidad al cliente
2.0	15/05/2023	Interfaces de Usuario	Mejora de las interfaces de usuario para una mejor interpretación

## **Ficha del documento 77**

### **1 Introducción 79**

- 1.1 Propósito 79
- 1.2 Alcance 79
- 1.3 Personal involucrado 80
- 1.4 Definiciones, acrónimos y abreviaturas 80
- 1.5 Referencias 81
- 1.6 Resumen 81

### **2 Descripción general 81**

- 2.1 Perspectiva del producto 81
- 2.2 Funcionalidad del producto 82
- 2.3 Características de los usuarios 82
- 2.4 Restricciones 83
- 2.5 Suposiciones y dependencias 83

### **3 Requisitos específicos 83**

- 3.1 Requisitos comunes de los interfaces 84
  - 3.1.1 Interfaces de usuario ..... 84
  - 3.1.2 Interfaces de hardware ..... 87
  - 3.1.3 Interfaces de software ..... 88
- 3.2 Requisitos funcionales 88
  - 3.2.1 Requisito funcional RF01 ..... 88
  - 3.2.1 Requisito funcional RF02 ..... 88
  - 3.2.1 Requisito funcional RF03 ..... 89
  - 3.2.1 Requisito funcional RF04 ..... 89
  - 3.2.1 Requisito funcional RF05 ..... 90
  - 3.2.1 Requisito funcional RF06 ..... 90
  - 3.2.2 Requisito funcional RF07 ..... 90
  - 3.2.3 Requisito funcional RF08 ..... 91
  - 3.2.4 Requisito funcional RF09 ..... 91
- 3.3 Requisitos no funcionales 91
  - 3.3.1 Requisitos de rendimiento ..... 91
  - 3.3.2 Usabilidad ..... 92
  - 3.3.3 Seguridad ..... 92

## 1 Introducción

En el presente documento se detalla la especificación de requisitos de Software (ERS) para implementar la tecnología blockchain en el sistema de difusión y ventas de obras de arte, proporcionando trazabilidad a las mismas. Esta especificación se elaboró según las directrices del estándar IEEE para Especificaciones de Requisitos Software ANSI/IEEE 830, 1998. La implementación de la tecnología de blockchain transformará el sistema actual en una aplicación descentralizada (Dapp), ofreciendo beneficios significativos en términos inmutabilidad y transparencia

### 1.1 Propósito

El propósito de este documento es definir los requisitos funcionales y no funcionales necesarios para dar trazabilidad a las obras de arte mediante la adopción de la tecnología blockchain y los Tokens No Fungibles (NFT) para proporcionará un registro inmutable y verificable de todas las actividades relacionadas con cada obra de arte.

### 1.2 Alcance

La especificación de requisitos refleja las necesidades y expectativas del usuario final que desea software, cuyo objetivo es dar trazabilidad a las obras de arte. El alcance incluye:

- **Registro del proceso de creación de obras de arte:** Registro del proceso de creación de cada obra de arte en la blockchain, garantizando su autenticidad desde su inicio.
- **Generación y manejo de NFTs:** Conversión de cada obra de arte en un Token No Fungible (NFT) que permita su identificación única y seguimiento.

### 1.3 Personal involucrado

Tabla 21 Personal involucrado estudiante de la carrera de computación

<b>Nombre</b>	Alexander Adrián Vicente Jiménez
<b>Rol</b>	Analista y Desarrollador de Software
<b>Categoría profesional</b>	Estudiante de carrera de computación.
<b>Responsabilidades</b>	Análisis de información, diseño y programación de la DApp
<b>Información de contacto</b>	<a href="mailto:alexander.a.vicente@unl.edu.ec">alexander.a.vicente@unl.edu.ec</a>

Tabla 22. Personal involucrado docente de la CC

<b>Nombre</b>	Cristian Ramiro Narváez Guillen
<b>Rol</b>	Director del Proyecto de Integración Curricular
<b>Categoría profesional</b>	Docente de la de la carrera de computación
<b>Responsabilidades</b>	Supervisar y asesorar en el desarrollo del Proyecto de Integración Curricular
<b>Información de contacto</b>	<a href="mailto:cristian.narvaez@unl.edu.ec">cristian.narvaez@unl.edu.ec</a>

### 1.4 Definiciones, acrónimos y abreviaturas

Tabla 23. Definiciones, acrónimos y abreviaturas

<b>Nombre</b>	<b>Descripción</b>
<b>BC</b>	Blockchain
<b>SC</b>	Smart Contract
<b>RF</b>	Requerimiento funcional
<b>RFN</b>	Requisito No Funcional
<b>SDVOA</b>	Sistema de difusión y venta de obras de arte
<b>Token / NFT</b>	Representación de la obra de arte en la red blockchain de Ethereum
<b>ETH</b>	ETH es el símbolo de la criptomoneda nativa de la red Ethereum
<b>Billetera</b>	Permite a los usuarios almacenar y gestionar sus Ether (ETH) y facilita la interacción con contratos inteligentes y aplicaciones descentralizadas (DApps) en la red Ethereum.

## 1.5 Referencias

Tabla 24. Referencias

Título del Documento	Referencia
IEEE Std 830-1998 IEEE Recommended Practice for Software	IEEE Std 830-1998 IEEE Recommended Practice for Software
Desarrollar una Aplicación Web para la difusión y venta de obras de arte para la carrera de Artes Plásticas / Visuales de la Universidad Nacional de Loja	Cristian Eduardo, M. M., & Rojas Cobos, M. A. (2023). Desarrollar una Aplicación Web para la difusión y venta de obras de arte para la carrera de Artes Plásticas / Visuales de la Universidad Nacional de Loja (Dissertation). Retrieved from <a href="https://dspace.unl.edu.ec/jspui/handle/123456789/28682">https://dspace.unl.edu.ec/jspui/handle/123456789/28682</a>

## 1.6 Resumen

El documento se divide en tres secciones. La primera sección proporciona una visión general de la funcionalidad del producto, junto con características de los usuarios y las restricciones. La segunda sección ofrece una descripción general del sistema, incluyendo interfaces de usuario, hardware y software. La tercera y última sección se enfoca en los requisitos específicos que el sistema debe cumplir y detalla su definición.

## 2 Descripción general

### 2.1 Perspectiva del producto

El sistema de difusión y venta de obras de arte, está construido sobre una aplicación web, la misma que servirá para implementar la tecnología blockchain, ofreciendo una eficiencia y rapidez los usuarios. Este enfoque aprovechará los atributos fundamentales de la blockchain, tales como transparencia, inmutabilidad, seguridad y trazabilidad asegurando que cada obra de arte cuente con un historial verificable y seguro desde su creación hasta su venta.

## 2.2 Funcionalidad del producto

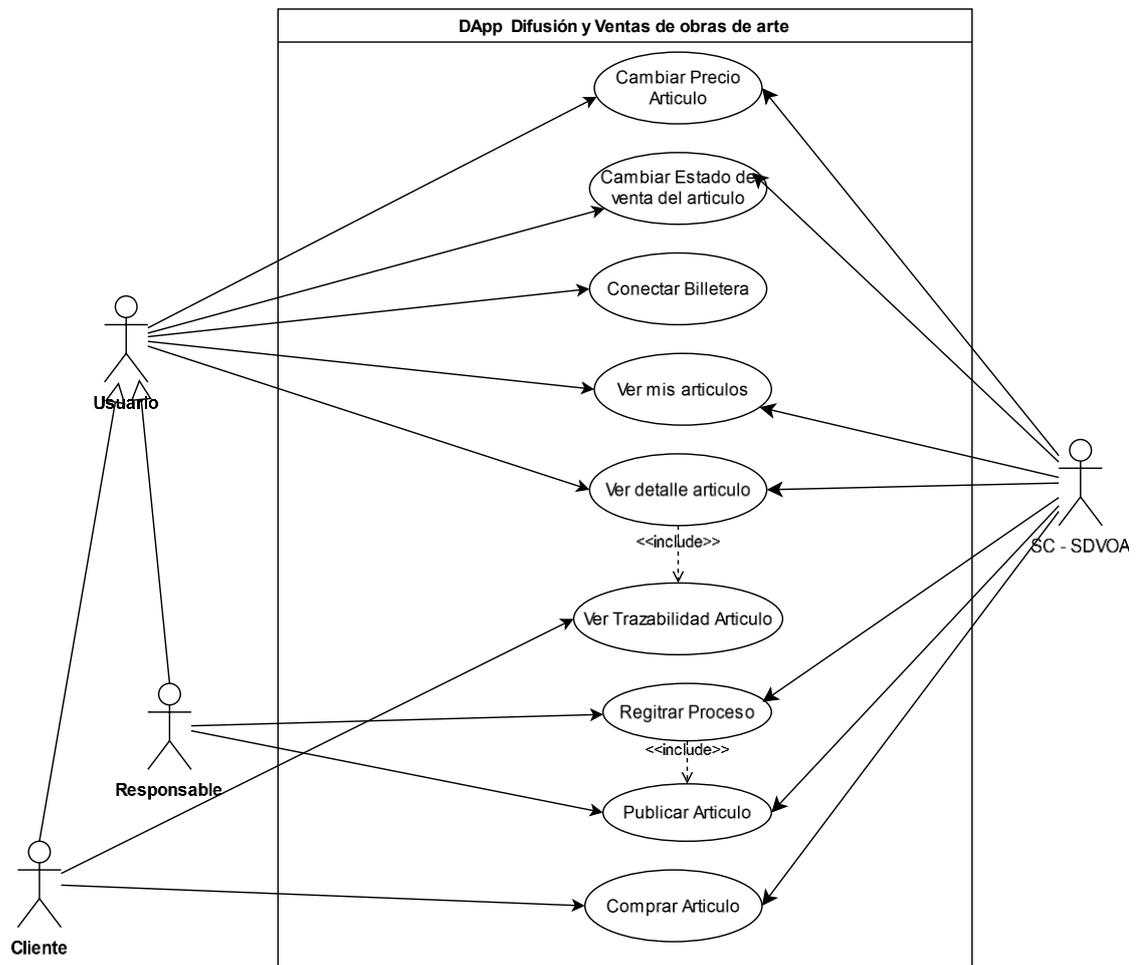


Figura 42. Roles y Actividades

## 2.3 Características de los usuarios

A continuación, se describen los roles de los usuarios que interactuarán con el SC y, a través de este, con la red de blockchain Ethereum. Se utilizaron los roles del sistema de difusión y ventas de obras de arte por su importancia en el mismo.

Tabla 25. Características usuario Responsable

<b>Responsable</b>	Usuario Responsable
<b>Formación</b>	Estudiantes de la Carrera de artes visuales.
<b>Actividades</b>	<ul style="list-style-type: none"> <li>• Registrar el proceso de creación de una obra de arte</li> <li>• Crear el token</li> </ul>

Tabla 26. Características del usuario cliente

<b>Responsable</b>	Usuario Cliente
<b>Formación</b>	Comprador de NFT's/ Interesado en las obras de arte
<b>Actividades</b>	Comprar token Visualizar la trazabilidad de cada token

Tabla 27. Características del usuario

<b>Responsable</b>	Usuario
<b>Formación</b>	N/A
<b>Actividades</b>	Conectar billetera Cambiar precio de sus tokens Cambiar estado de venta de sus tokens Ver los detalles de un token Ver los tokens que posee

Tabla 28. Características del usuario SC-SDV

<b>Responsable</b>	Usuario SC-SDV
<b>Formación</b>	N/A
<b>Actividades</b>	Interactuar con la red de blockchain de Ethereum mediante sus funcionalidades.

## 2.4 Restricciones

- Se debe emplear una metodología ágil para iteraciones rápidas y una respuesta eficiente a los cambios
- Mantener lenguajes del sistema actual (Python) y usar Solidity para crear los contratos inteligentes
- Utilizar Linux para las herramientas de desarrollo

## 2.5 Suposiciones y dependencias

- Acceso continuo a internet; restricciones pueden afectar la DApp.
- Interoperabilidad con el sistema existente

## 3 Requisitos específicos

### 3.1 Requisitos comunes de los interfaces

#### 3.1.1 Interfaces de usuario

Las interfaces de usuarios agregadas al sistema estarán compuestas por formularios, botones, imágenes, listas, campos de texto. La finalidad es facilitar el uso de las funcionalidades de la DApp

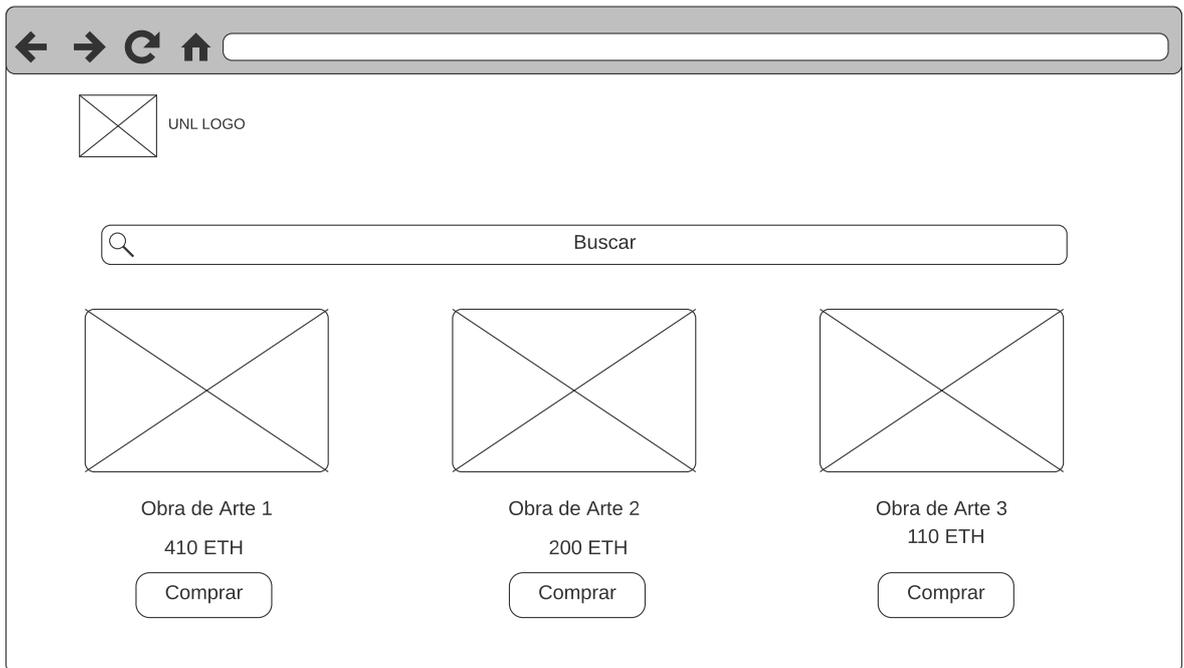
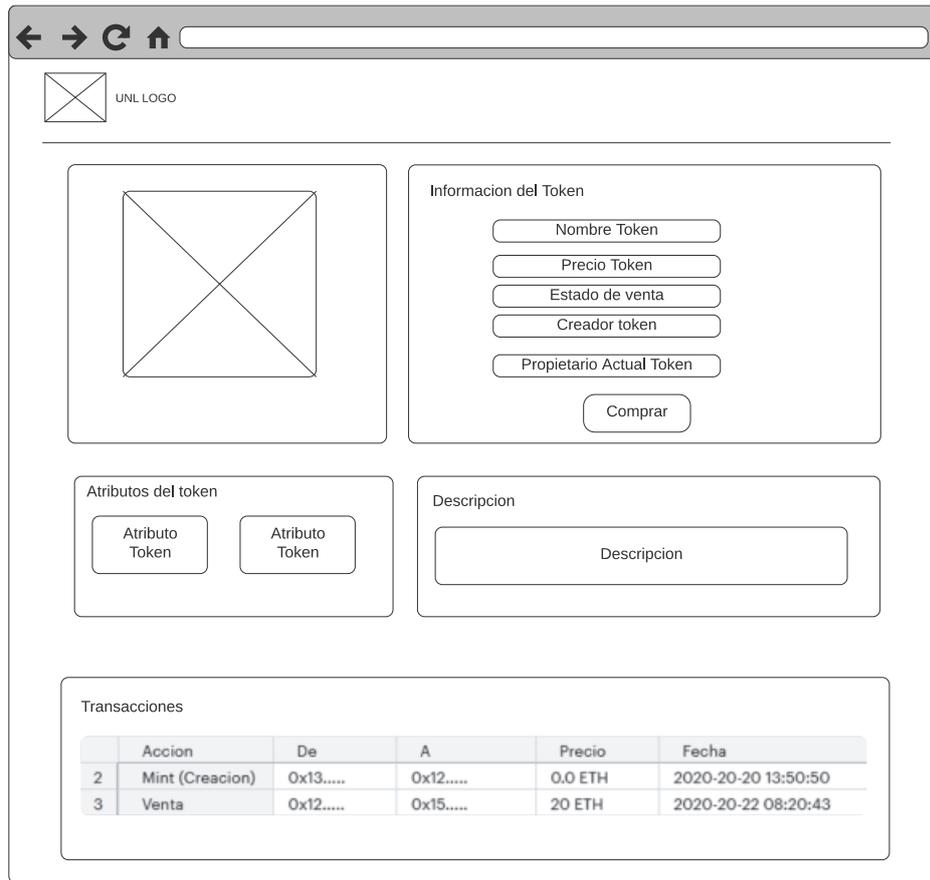


Figura 43. Interfaz de Usuario principal de la DApp

En la **Figura 43**, se muestra la interfaz principal de la DApp en la que se mostraran los tokens disponibles. El botón "Comprar" " permite a los usuarios comprar cada token.



**Figura 44.** Interfaz de Usuario para ver los detalles del token

En la **Figura 44** , se muestra la muestra la página detallada de un token. La sección principal presenta una imagen del token y debajo sus atributos. Al lado, se encuentran los detalles del token, incluyendo su nombre, precio, creador, propietario actual, junto con un botón de compra y la descripción. . Finalmente, la interfaz muestra una tabla de transacciones relacionadas con el token, detallando las acciones, direcciones involucradas, precios y fechas de las transacciones.

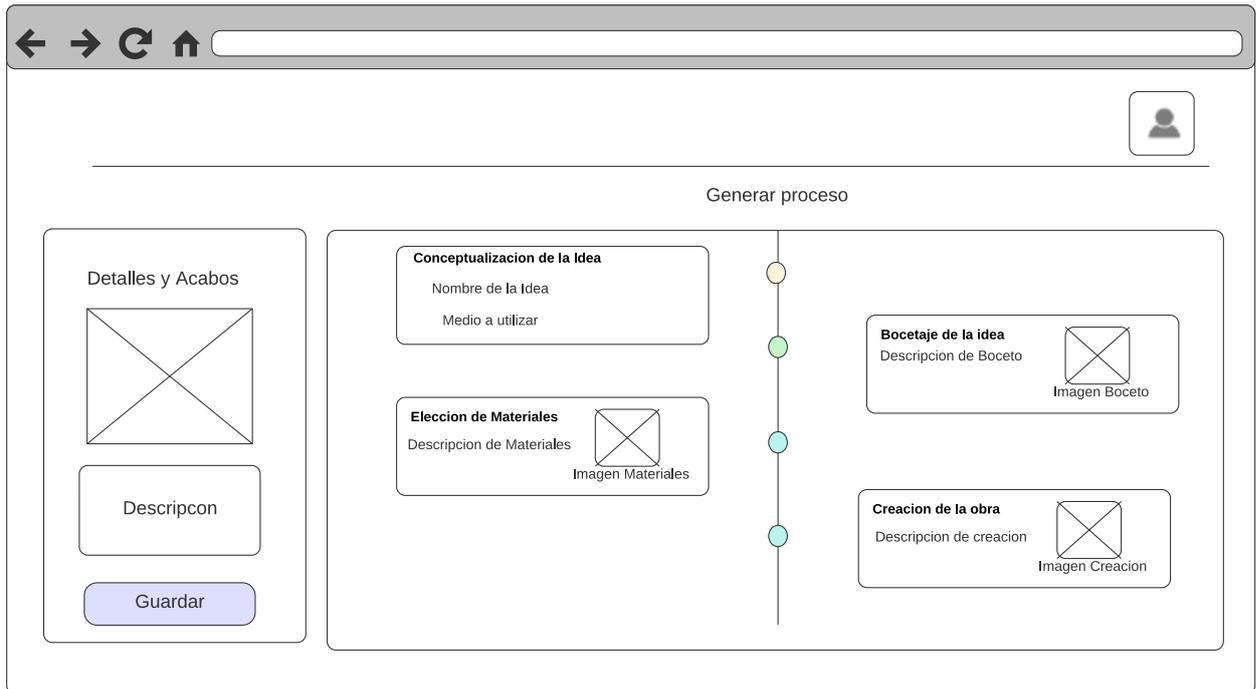


Figura 45. Interfaz de Usuario para registrar un proceso

En la **Figura 45**, se muestra la interfaz de usuario para que registre el proceso de creación de una obra de arte dentro de la DApp. Esta página permite registrar cada paso del proceso de creación de la obra de arte y datos asociados

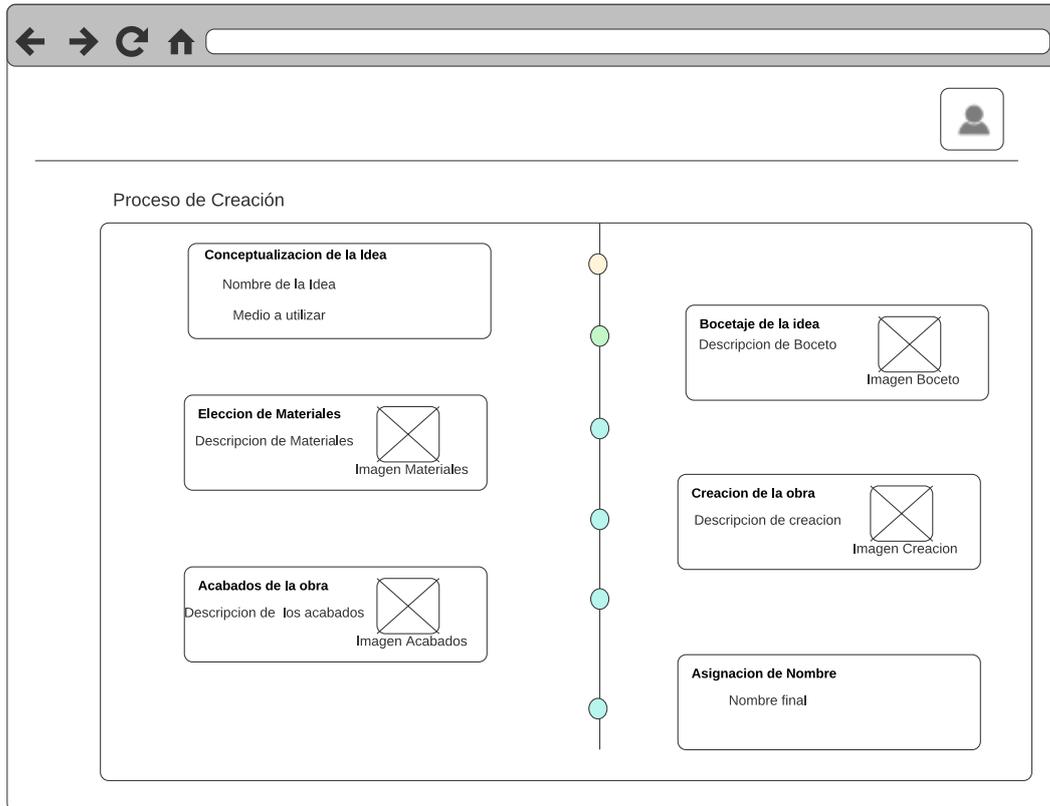


Figura 46. Interfaz de Usuario para ver el proceso de creación del token

En la **Figura 46**, se muestra la interfaz de usuario para observar el proceso de creación por el que ha pasado la obra de arte. En esta, se encuentra una línea de tiempo con cada una de las fases de proceso junto con sus datos asociados

### 3.1.2 Interfaces de hardware

Para tener una mejor interacción entre usuario y sistema, se aconseja tener dispositivos o equipos con las siguientes características.

- Laptop – Computador Personal
  - Procesador de al menos Core i3 o Pentium Dual Core.
  - Memoria RAM de 2 GB como mínimo.
  - Dispositivos de entrada/salida para facilitar la interacción del usuario con el sistema.
- Conectividad:

- Conexión a internet de alta velocidad para garantizar la comunicación fluida con la red blockchain y otros servicios externos.
- Se recomienda disponer de una conexión por cable Ethernet o Wi-Fi para garantizar un rendimiento óptimo.

### 3.1.3 Interfaces de software

- Sistema operativo Windows 7 o superior
- Navegador Web. Mozilla, chrome o edge
- Framework Django
- Web3 de python
- Metamask

## 3.2 Requisitos funcionales

### 3.2.1 Requisito funcional RF01

Tabla 29. Requisito funcional para crear un proceso de creación de obras de arte en la DApp

<b>Identificación</b>	RF01
<b>Nombre</b>	Registrar Proceso
<b>Descripción</b>	LA DApp permitirá a un responsable registrar el proceso de creación de sus obras de arte
<b>Dependencia</b>	El responsable debe estar registrado en la DApp
<b>Requisito NO funcional</b>	<ul style="list-style-type: none"> <li>• <b>RFN03</b></li> </ul>
<b>Prioridad</b>	Alta

### 3.2.1 Requisito funcional RF02

Tabla 30. Requisito funcional para crear un NFT

<b>Identificación</b>	RF02
<b>Nombre</b>	Registrar Artículo
<b>Descripción</b>	La DApp permitirá al responsable registrar un artículo en la blockchain de Ethereum
<b>Dependencia</b>	El proceso de creación de la obra de arte debe estar finalizado
	El responsable debe de tener su cuenta de metamask conectada

El responsable debe tener su cuenta asociada a la DApp

**Requisito NO funcional**

- **RFN01**
- **RFN03**

---

**Prioridad** Alta

---

### 3.2.2 Requisito funcional RF03

Tabla 31. Requisito funcional para cambiar el precio del token

---

<b>Identificación</b>	RF03
<b>Nombre</b>	Cambiar precio articulo
<b>Descripción</b>	La DApp permitirá al usuario propietario de un artículo modificar su precio
<b>Dependencia</b>	El artículo debe estar almacenado en la blockchain de Ethereum El usuario debe ser propietario del token
<b>Requisito NO funcional</b>	<ul style="list-style-type: none"><li>• <b>RFN01</b></li><li>• <b>RFN03</b></li></ul>
<b>Prioridad</b>	Alta

---

### 3.2.3 Requisito funcional RF04

Tabla 32. Requisito funcional para comprar un token

---

<b>Identificación</b>	RF04
<b>Nombre</b>	Comprar Articulo
<b>Descripción</b>	La DApp permitirá a los clientes comprar un artículo disponible
<b>Dependencia</b>	El cliente debe tener su billetera conectada a la DApp El cliente debe contar con ETH suficientes para realizar la compra
<b>Requisito NO funcional</b>	<ul style="list-style-type: none"><li>• <b>RFN01</b></li><li>• <b>RFN03</b></li></ul>
<b>Prioridad</b>	Alta

---

### 3.2.4 Requisito funcional RF05

Tabla 33. Requisito funcional para cambiar el estado de venta del token

<b>Identificación</b>	RF05
<b>Nombre</b>	Cambiar estado de venta del Articulo
<b>Descripción</b>	La DApp permitirá al usuario propietario de un artículo cambiar su estado de venta
<b>Dependencia</b>	El articulo debe estar almacenado en la blockchain de Ethereum El usuario debe ser propietario del token
<b>Requisito NO funcional</b>	<b>RFN01</b> <b>RFN03</b>
<b>Prioridad</b>	Alta

### 3.2.5 Requisito funcional RF06

Tabla 34. Requisito funcional para ver la trazabilidad de una obra de arte

<b>Identificación</b>	RF06
<b>Nombre</b>	Ver trazabilidad del Articulo
<b>Descripción</b>	La DApp permitirá al usuario ver la trazabilidad del artículo, desde su proceso de creación hasta sus transacciones
<b>Dependencia</b>	El token debe estar almacenado en la blockchain de Ethereum
<b>Requisito NO funcional</b>	<ul style="list-style-type: none"><li>• <b>RFN01</b></li><li>• <b>RFN02</b></li></ul>
<b>Prioridad</b>	Media

### 3.2.6 Requisito funcional RF07

Tabla 35. Requisito funcional para ver los detalles del token

<b>Identificación</b>	RF07
<b>Nombre</b>	Ver detalles del articulo
<b>Descripción</b>	La DApp permitirá ver todos los datos asociados de un Articulo

<b>Dependencia</b>	El articulo debe estar registrado en la blockchain de Ethereum
<b>Requisito NO funcional</b>	<ul style="list-style-type: none"> <li>• <b>RFN01</b></li> <li>• <b>RFN02</b></li> <li>• <b>RFN03</b></li> </ul>
<b>Prioridad</b>	Alta

### 3.2.7 Requisito funcional RF08

Tabla 36. Requisito funcional para conectar una billetera a la DApp

<b>Identificación</b>	RF08
<b>Nombre</b>	Conectar Billetera
<b>Descripción</b>	La DApp permitirá a los usuarios conectar su billetera de Ethereum para poder realizar transacciones
<b>Dependencia</b>	El usuario debe tener una billetera de Ethereum
<b>Requisito NO funcional</b>	<ul style="list-style-type: none"> <li>• <b>RFN03</b></li> </ul>
<b>Prioridad</b>	Alta

### 3.2.8 Requisito funcional RF09

Tabla 37. Requisito funcional para ver los tokens de un usuario

<b>Identificación</b>	RF09
<b>Nombre</b>	Ver Artículos
<b>Descripción</b>	La DApp permitirá a los usuarios ver los artículos que son de su propiedad
<b>Dependencia</b>	El usuario debe contar con tokens comprados o creados
<b>Requisito NO funcional</b>	<ul style="list-style-type: none"> <li>• <b>RFN01</b></li> <li>• <b>RFN03</b></li> </ul>
<b>Prioridad</b>	Alta

## 3.3 Requisitos no funcionales

### 3.3.1 Requisitos de rendimiento

<b>Identificación</b>	RNF01
<b>Nombre</b>	Rendimiento

<b>Descripción</b>	En una red de prueba de Ethereum, la transacción tarda entre 2 a 5 segundos; sin embargo, en un entorno real, puede tardar entre 15 segundos a 5 minutos
<b>Prioridad</b>	Alta

---

### 3.3.2 Usabilidad

<b>Identificación</b>	RNF02
<b>Nombre</b>	Usabilidad
<b>Descripción</b>	El sistema debe proporcionar una interfaz que permita a los usuarios visualizar la trazabilidad completa de las obras de arte.
<b>Prioridad</b>	Alta

---

### 3.3.3 Seguridad

<b>Identificación</b>	RNF03
<b>Nombre</b>	Seguridad
<b>Descripción</b>	Se debe garantizar que solo los usuarios autorizados puedan realizar ciertas operaciones en el sistema
<b>Prioridad</b>	Alta

---

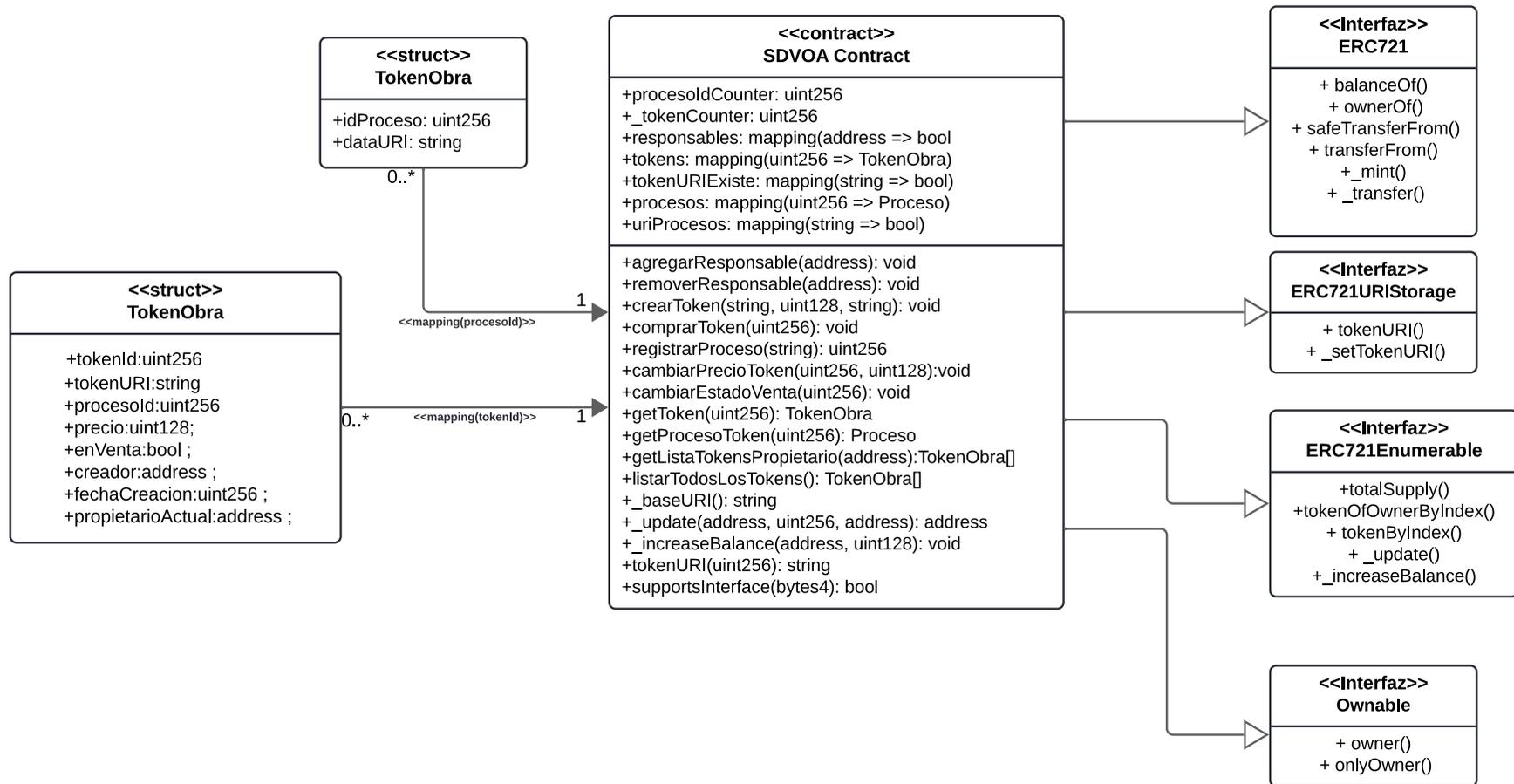
## Anexo 5. Historias de Usuario

Identificador (ID) de la Historia	Enunciado de la Historia				Criterios de Aceptación			
	Rol	Característica / Funcionalidad	Razón / Resultado	Número (#) de Escenario	Criterio de Aceptación (Título)	Contexto	Evento	Resultado / Comportamiento esperado
HU01	Como un responsable	Necesito registrar el proceso de creación de una obra de arte	Con la finalidad de tener un registro sobre todo el proceso de creación	1	Validar el registro exitoso del proceso de creación de una obra de arte	En caso que el responsable esté registrado en la DApp y vaya a registrar la obra de arte en la red blockchain de Ethereum	Cuando el responsable vaya a almacenar su obra de arte en la red blockchain de Ethereum	El sistema mediante el contrato inteligente debe almacenar el proceso en la blockchain de Ethereum
HU02	Como un responsable	Necesito registrar un artículo	Con la finalidad de proporcionar una representación única y verificable de la obra	1	Validar la creación exitosa del artículo	En caso que el proceso de creación de la obra de arte esté finalizado y se vaya a almacenar la obra en la red blockchain de Ethereum	Cuando el responsable seleccione la opción para registra el artículo	El sistema mediante el contrato inteligente debe crear el artículo y almacenarlo en la blockchain de Ethereum
				2	Validar que no se registre un artículo ya existente	En caso de que el artículo ya este registrado	Cuando el responsable seleccione la opción para registra el artículo	El contrato inteligente no debe almacenar el artículo en la red blockchain de Ethereum
				3	Validar que solo un responsable	En caso de que el responsable no este registrado	Cuando quiera registrar un artículo	El contrato inteligente no debe almacenar el artículo en la red blockchain de Ethereum

					pueda crear el artículo			
HU03	Como un Usuario	Necesito cambiar el precio de un artículo	Con la finalidad de ajustar su valor	1	Validar que el artículo cambio su precio	En caso de que el artículo este registrado en la blockchain de Ethereum	Cuando el propietario coloque el nuevo precio del artículo y presione el botón de cambiar precio	El sistema mediante el contrato inteligente debe cambiar el precio del artículo
HU04	Como un usuario	Necesito cambiar el estado de venta del artículo	Con la finalidad de activar o desactivar la venta del artículo	1	Validar que el artículo cambio su estado de venta	En caso de que el artículo este registrado en la blockchain de Ethereum	Cuando el usuario cambie el estado de venta del artículo mediante un botón	El sistema mediante el contrato inteligente debe cambiar el estado de venta del artículo
HU05	Como un usuario	Necesito ver mis artículos	Con la finalidad de gestionar los artículos que son de mi propiedad	1	Validar que se muestren los artículos del usuario	En caso de que el usuario haya comprado o creado artículos	Cuando el usuario acceda a la sección de mis artículos	El sistema mediante el contrato inteligente debe mostrar los artículos que pertenecen al usuario

HU06	Como un usuario	Necesito ver los detalles de un artículo	Con la finalidad de conocer información específica del artículo	1	Validar que se muestre la información asociada a un artículo	En caso de que existan artículos creados	Cuando el usuario quiera ver la información del artículo	El sistema mediante el contrato inteligente debe mostrar la información asociada al artículo
HU07	Como un Cliente	Necesito comprar un artículo	Con la finalidad de adquirir la propiedad de un artículo	1	Validar la compra exitosa del artículo	En caso que el cliente cuente con una billetera y ETH suficientes	Cuando el cliente seleccione el artículo y confirme la compra	El sistema mediante el contrato inteligente debe transferir el artículo al cliente y debitar los ETH correspondientes
HU08	Como un Cliente	Necesito ver la trazabilidad de una obra de arte	Con la finalidad de conocer el proceso de creación y el historial de transacciones del artículo	1	Validar la visualización de la trazabilidad	En caso que el artículo esté almacenado en la blockchain de Ethereum	Cuando el usuario vea los detalles del artículo	El sistema mediante el contrato inteligente debe obtener el proceso de creación del artículo y el historial de transacciones para mostrar la trazabilidad completa del artículo
HU09	Como un Usuario	Necesito conectar mi billetera a la DApp	Con la finalidad de realizar transacciones dentro de la DApp	1	Validar la Conexión exitosa de la billetera	En caso que el usuario tenga una billetera	Cuando el usuario seleccione la opción de conectar billetera	El sistema debe conectar la billetera del usuario a la DApp

Anexo 6. Diagrama de clases del contrato inteligente



**Anexo 7.** Código fuente del contrato inteligente

**Proyecto:** Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas

Fecha: 05/06/2024

```

// SPDX-License-Identifier: MIT
pragma solidity ^0.8.20;

import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import
"@openzeppelin/contracts/token/ERC721/extensions/ERC721Enumerable.sol";
import
"@openzeppelin/contracts/token/ERC721/extensions/ERC721URIStorage.sol";
import "@openzeppelin/contracts/access/Ownable.sol";

/**
 * @title SDVOAContract
 * @dev Contrato de token ERC721 para la gestión de una colección de arte.
 * Implementa funcionalidad de enumeración y almacenamiento URI para cada
token.
 */
contract SDVOAContract is ERC721, ERC721Enumerable, ERC721URIStorage,
Ownable {
    uint256 public procesoIdCounter;
    uint256 private _tokenCounter;
    mapping(address => bool) public responsables;
    mapping(address => bool) public encargados;

    struct TokenObra {
        uint256 tokenId;
        string tokenURI;
        uint256 procesoId;
        uint128 precio;
        bool enVenta;
        address creador;
        uint256 fechaCreacion;
        address propietarioActual;
    }

    struct Proceso {
        uint256 idProceso;
        string dataURI;
    }

    mapping(uint256 => TokenObra) public tokens;
    mapping(string => bool) public tokenURIExiste;
    mapping(uint256 => Proceso) private procesos;
    mapping(string => bool) private uriProcesos;

    event TokenCreado(uint256 indexed tokenId, string tokenURI, uint256
procesoId, uint128 precio, address indexed creador);
    event ProcesoRegistrado(uint256 indexed idProceso, string dataURI);
    event TokenComprado(uint256 indexed tokenId, address indexed comprador,
uint128 precio);
    event PrecioTokenCambiado(uint256 indexed tokenId, uint128
nuevoPrecio);
    event EstadoVentaTokenCambiado(uint256 indexed tokenId, bool enVenta);
    event ResponsableAsignado(address indexed Encargado, address indexed
nuevoResponsable);
    event EncargadoAsignado(address indexed nuevoEncargado);

    /**
     * @dev Constructor del contrato.
     */
    constructor() ERC721("Coleccion de Arte", "ARTE") Ownable(msg.sender){}

```

```

/**
 * @dev Modificador para verificar si el remitente es un responsable.
 */
modifier onlyResponsable() {
    require(responsables[msg.sender], "No tienes el rol de responsable");
    _;
}

/**
 * @dev Modificador para verificar si el remitente es el encargado.
 */
modifier onlyEncargado() {
    require(encargados[msg.sender], "No tienes el rol de encargado");
    _;
}

/**
 * @dev Asignar un nuevo encargado.
 * @param _encargado Dirección del nuevo encargado.
 */
function asignarEncargado(address _encargado) external onlyOwner {
    encargados[_encargado] = true;
    emit EncargadoAsignado(_encargado);
}

/**
 * @dev Remover un encargado.
 * @param _encargado Dirección del encargado a remover.
 */
function removerEncargado(address _encargado) external onlyOwner {
    responsables[_encargado] = false;
}

/**
 * @dev Agregar una dirección a la lista de responsables.
 * @param _responsable Dirección del responsable.
 */
function agregarResponsable(address _responsable) external
onlyEncargado {
    responsables[_responsable] = true;
    emit ResponsableAsignado(msg.sender, _responsable);
}

/**
 * @dev Remover una dirección de la lista de responsables.
 * @param _responsable Dirección del responsable.
 */
function removerResponsable(address _responsable) external
onlyEncargado {
    responsables[_responsable] = false;
}

/**
 * @dev Crear un nuevo token y un nuevo proceso asociado.
 * @param _tokenURI URI del token.
 * @param _precio Precio inicial del token.
 * @param _procesoURI URI del proceso asociado al token.

```

```

*/
function crearToken(string memory _tokenURI, uint128 _precio, string
memory _procesoURI) external onlyResponsible {
    require(msg.sender != address(0), "Direccion invalida");
    require(!tokenURIExiste[_tokenURI], "URI ya existe");

    _tokenCounter++;
    uint256 newTokenId = _tokenCounter;

    _mint(msg.sender, newTokenId);
    _setTokenURI(newTokenId, _tokenURI);

    tokenURIExiste[_tokenURI] = true;

    uint256 procesoId = registrarProceso(_procesoURI);

    tokens[newTokenId] = TokenObra({
        tokenId: newTokenId,
        tokenURI: _tokenURI,
        procesoId: procesoId,
        precio: _precio,
        enVenta: true,
        creador: msg.sender,
        fechaCreacion: block.timestamp,
        propietarioActual: msg.sender
    });

    emit TokenCreado(newTokenId, _tokenURI, procesoId, _precio,
msg.sender);
}

/**
 * @dev Comprar un token que está en venta.
 * @param tokenId ID del token a comprar.
 */
function comprarToken(uint256 tokenId) public payable {
    require(msg.sender != address(0), "Direccion invalida");

    address propietarioToken = ownerOf(tokenId);
    require(propietarioToken != address(0), "El token no tiene
propietario");
    require(propietarioToken != msg.sender, "Ya eres el propietario");

    TokenObra storage obra = tokens[tokenId];
    require(msg.value >= obra.precio, "Fondos insuficientes");
    require(obra.enVenta, "El token no esta en venta");

    _transfer(propietarioToken, msg.sender, tokenId);
    payable(propietarioToken).transfer(msg.value);

    obra.enVenta = false;
    obra.propietarioActual = msg.sender;

    emit TokenComprado(tokenId, msg.sender, obra.precio);
}

/**
 * @dev Registrar un nuevo proceso.
 * @param _dataURI URI del proceso.
 * @return ID del proceso registrado.
 */

```

```

function registrarProceso(string memory _dataURI) internal returns
(uint256) {
    require(bytes(_dataURI).length > 0, "El proceso debe contener el
URI");
    require(!uriProcesos[_dataURI], "El URI ya ha esta registrado en
otro proceso");

    uint256 newProcesoId = ++procesoIdCounter;

    procesos[newProcesoId] = Proceso({
        idProceso: newProcesoId,
        dataURI: _dataURI
    });

    uriProcesos[_dataURI] = true;

    emit ProcesoRegistrado(newProcesoId, _dataURI);

    return newProcesoId;
}

/**
 * @dev Cambiar el precio de un token.
 * @param tokenId ID del token.
 * @param nuevoPrecio Nuevo precio del token.
 */
function cambiarPrecioToken(uint256 tokenId, uint128 nuevoPrecio)
public {
    require(msg.sender != address(0), "Direccion invalida");
    require(ownerOf(tokenId) == msg.sender, "No eres el propietario");

    tokens[tokenId].precio = nuevoPrecio;

    emit PrecioTokenCambiado(tokenId, nuevoPrecio);
}

/**
 * @dev Cambiar el estado de venta de un token.
 * @param tokenId ID del token.
 */
function cambiarEstadoVenta(uint256 tokenId) public {
    require(msg.sender != address(0), "Direccion invalida");
    require(ownerOf(tokenId) == msg.sender, "No eres el propietario");

    tokens[tokenId].enVenta = !tokens[tokenId].enVenta;

    emit EstadoVentaTokenCambiado(tokenId, tokens[tokenId].enVenta);
}

/**
 * @dev Obtener los detalles de un token.
 * @param _tokenId ID del token.
 * @return Detalles del token.
 */
function getToken(uint256 _tokenId) external view returns (TokenObra
memory) {
    return tokens[_tokenId];
}

/**
 * @dev Obtener el proceso asociado a un token.

```

```

    * @param tokenId ID del token.
    * @return Detalles del proceso.
    */
    function getProcesoToken(uint256 tokenId) external view returns
(Proceso memory) {
        return procesos[tokens[tokenId].procesoId];
    }

    /**
    * @dev Obtiene una lista de tokens pertenecientes a un propietario.
    * @param propietario La dirección del propietario de los tokens.
    * @return Una matriz de estructuras TokenObra que representan los
tokens del propietario.
    */
    function getListaTokensPropietario(address propietario) external view
returns (TokenObra[] memory) {
        uint256 tokenCount = balanceOf(propietario);
        TokenObra[] memory result = new TokenObra[](tokenCount);
        uint256 index = 0;
        for (uint256 tokenId = 1; tokenId <= _tokenCounter; tokenId++) {
            if (ownerOf(tokenId) == propietario) {
                result[index] = tokens[tokenId];
                index++;
            }
        }
        return result;
    }

    /**
    * @dev Lista todos los tokens existentes.
    * @return Una matriz de estructuras TokenObra que representan todos
los tokens creados.
    */
    function listarTodosLosTokens() external view returns (TokenObra[]
memory) {
        uint256 totalTokens = totalSupply();
        TokenObra[] memory result = new TokenObra[](totalTokens);
        uint256 index = 0;
        for (uint256 tokenId = 1; tokenId <= _tokenCounter; tokenId++) {
            if (ownerOf(tokenId) != address(0)) {
                result[index] = tokens[tokenId];
                index++;
            }
        }
        return result;
    }

    /**
    * @dev Obtener la base URI para los tokens.
    * @return Base URI.
    */
    function _baseURI() internal pure override returns (string memory) {
        return "https://ipfs.io/ipfs/";
    }

    /**
    * @dev Actualizar la dirección del propietario de un token.
    * @param to Dirección del nuevo propietario.
    * @param tokenId ID del token.
    * @param auth Dirección del autorizado.

```

```

    * @return Dirección del propietario anterior.
    */
function _update(address to, uint256 tokenId, address auth)
    internal
    override(ERC721, ERC721Enumerable)
    returns (address)
{
    return super._update(to, tokenId, auth);
}

/**
 * @dev Aumentar el balance de una cuenta.
 * @param account Dirección de la cuenta.
 * @param value Valor a aumentar.
 */
function _increaseBalance(address account, uint128 value)
    internal
    override(ERC721, ERC721Enumerable)
{
    super._increaseBalance(account, value);
}

/**
 * @dev Obtener la URI de un token.
 * @param tokenId ID del token.
 * @return URI del token.
 */
function tokenURI(uint256 tokenId)
    public
    view
    override(ERC721, ERC721URIStorage)
    returns (string memory)
{
    return super.tokenURI(tokenId);
}

/**
 * @dev Verifica si el contrato soporta una interfaz específica.
 * @param interfaceId ID de la interfaz.
 * @return `true` si la interfaz es soportada, `false` de lo contrario.
 */
function supportsInterface(bytes4 interfaceId)
    public
    view
    override(ERC721, ERC721Enumerable, ERC721URIStorage)
    returns (bool)
{
    return super.supportsInterface(interfaceId);
}
}

```

**Anexo 8.** Plan de pruebas unitarias para el contrato inteligente SDVOA

**Proyecto:** Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas

Versión: 1.0

Fecha: 03/06/2024

## Hoja de control

---

<b>Organismo</b>	Universidad Nacional de Loja		
<b>Proyecto</b>	Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas		
<b>Entregable</b>	Pruebas Funcionales		
<b>Autor</b>	Alexander Vicente		
<b>Versión/Edición</b>	1.0	<b>Fecha Versión</b>	03/06/2024
<b>Aprobado por</b>	Cristian Ramiro Narváez Guillen, Mg.Sc	<b>Fecha Aprobación</b>	05/06/2024
		<b>N° Total de Páginas</b>	14

---



## 1. Introducción

En el presente documento se detalla el plan de pruebas diseñado para validar cada una de las funcionalidades del contrato inteligente SDVOAContract. Mediante el contrato se gestiona una colección de arte creado mediante el estándar ERC-721, permitiendo la creación, transferencia y gestión de obras de arte en una plataforma basada en blockchain.

Para llevar a cabo estas pruebas, se emplearán herramientas y frameworks como Truffle y OpenZeppelin, los cuales proporcionan un entorno para la creación y ejecución de pruebas unitarias y de integración. Cada prueba ha sido diseñada para cubrir un aspecto específico del contrato, asegurando así una cobertura completa y exhaustiva del contrato.

## 2. Objetivo

El principal objetivo de este plan de pruebas es comprobar que cada una de las funcionalidades se comporten según lo esperado

## 3. Alcance

Las pruebas abarcan tanto las operaciones básicas del contrato como las funcionalidades avanzadas, tales como la asignación de roles, la creación de procesos, gestión de cada obra y la gestión de pasos dentro de cada proceso.

## 4. Glosario

En esta sección se explican los términos técnicos, acrónimos o jerga utilizados en el documento con la finalidad de facilitar su comprensión.

Tabla 1. Glosario de términos

<b>Termino</b>	<b>Descripción</b>
Token	Representación de la obra de arte
ETH	ETH es la criptomoneda nativa de la red Ethereum.
Dirección de Ethereum	Una dirección de Ethereum es una cadena alfanumérica de 42 caracteres que identifica a un usuario
URI	Es una cadena de caracteres que identifica un recurso en la red
Metadata	Es información descriptiva de un token o proceso
Llamador	Es la dirección de la cuenta que realiza una llamada a un contrato inteligente.

## 5. Casos de prueba

A continuación, se describen cada caso de prueba en base a los escenarios en el punto anterior.

#### 4.1. Caso de Prueba – 0001: Validar que se asigne encargado por el propietario

Tabla 2. Caso de Prueba para validar que se asigne encargado por el propietario

<b>ID</b>	CP-0001
<b>Objetivo</b>	Validar que se asigne encargado por el propietario
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	Ninguna
<b>Entradas</b>	Dirección de Ethereum del responsable a registrar
<b>Pasos a seguir</b>	<ol style="list-style-type: none"><li>1. Llamar a la función `agregarResponsable` del contrato pasándole como parámetro la dirección Ethereum del responsable</li><li>2. Verificar que el responsable haya sido registrado exitosamente</li></ol>
<b>Resultado Esperado</b>	La dirección de Ethereum debe estar registrada con el rol de responsable

#### 4.2. Caso de Prueba – 0002: Remover un responsable del contrato

Tabla 3. Caso de prueba para remover un responsable del contrato

<b>ID</b>	CP-0002
<b>Objetivo</b>	Validar que se quite/remueva el rol de responsable a una dirección de Ethereum
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El responsable debe haber sido registrado previamente
<b>Entradas</b>	Dirección de Ethereum del responsable a remover
<b>Pasos a seguir</b>	<ol style="list-style-type: none"><li>1. Llamar a la función `removeResponsable` del contrato pasándole como parámetro la dirección Ethereum del responsable</li><li>2. Verificar que el responsable haya sido removido exitosamente</li></ol>
<b>Resultado Esperado</b>	El responsable debe removerse correctamente al contrato

#### 4.3. Caso de Prueba – 0003: Crear un token (Obra de arte)

Tabla 4. Caso de prueba validar la creación de un token

<b>ID</b>	CP-0003
<b>Objetivo</b>	Validar que se registre un token (obra de arte)
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El responsable debe haber sido registrado previamente
<b>Entradas</b>	<ul style="list-style-type: none"><li>• URI del Token: tokenURI</li><li>• Precio: 100</li></ul>

	<ul style="list-style-type: none"> <li>• URI del Proceso: procesoURI</li> <li>• Responsable: Dirección de Ethereum del responsable que crea el token.</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `crearToken` con los parámetros indicados anteriormente</li> <li>2. Verificar que el token se ha creado registrado correctamente</li> </ol>
<b>Resultado Esperado</b>	Se debe crear el token correctamente

#### 4.4. Caso de Prueba – 0004: Rechazar la creación de un token con una URI (metadata) existente

Tabla 5. Caso de prueba para validar que se rechace la creación de un token con una URI existente

<b>ID</b>	CP-0004
<b>Objetivo</b>	Validar que se rechace el registro de un token con una URI (metadata) existente
<b>Criticidad</b>	Media
<b>Precondiciones</b>	El responsable debe estar agregado y el token con la URI existente ya debe estar creado.
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• URI del Token: tokenURI</li> <li>• Precio: 100.</li> <li>• URI del Proceso: procesoURI</li> <li>• Responsable: Dirección de Ethereum del responsable que crea el token.</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `crearToken` y pasarle la URI del token existente con el resto parámetros indicados anteriormente</li> <li>2. Verificar que se rechace la creación del token</li> </ol>
<b>Resultado Esperado</b>	Se rechaza la creación del token si el URI del token ya existe

#### 4.5. Caso de Prueba – 0005: Rechazar que se cree un token con un la URI (metadata) de un proceso ya existente

Tabla 6. Caso de prueba para validar que se rechace la creación de un token con una URI de un proceso existente

<b>ID</b>	CP-0005
<b>Objetivo</b>	Validar que no se registre un token con una URI (metadata) de un proceso existente
<b>Criticidad</b>	Media
<b>Precondiciones</b>	El responsable debe estar registrado y el token con la URI del proceso existente ya debe estar creado.
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• URI del Token: tokenURI</li> <li>• Precio: 100.</li> </ul>

	<ul style="list-style-type: none"> <li>• URI del Proceso: procesoURI</li> <li>• Responsable: Dirección de Ethereum del responsable que crea el token.</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `crearToken` y pasarle la URI del proceso existente con el resto parámetros indicados anteriormente</li> <li>2. Verificar que se rechace la creación del token</li> </ol>
<b>Resultado Esperado</b>	Se rechaza la creación del token si el URI del proceso ya existe

#### 4.6. Caso de Prueba – 0006: Rechazar la creación de un token desde una dirección de Ethereum que no tenga el rol de responsable

Tabla 7. Caso de prueba para rechazar la creación de un token desde una dirección Ethereum que no tenga el rol de responsable

<b>ID</b>	CP-0006
<b>Objetivo</b>	Validar que se rechace la creación de un token, si se está creando un token desde una dirección de Ethereum que no esté registrada como responsable
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	Ninguna
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• URI del Token: tokenURI</li> <li>• Precio: 100.</li> <li>• URI del Proceso: procesoURI</li> <li>• Usuario: Dirección de Ethereum que intenta crea el token.</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `crearToken` y pasarle los parámetros desde una dirección que no tenga el rol de responsable</li> <li>2. Verificar que se rechace la creación del token</li> </ol>
<b>Resultado Esperado</b>	Se rechaza la creación del token si quien crea el token no tiene el rol de responsable

#### 4.7. Caso de Prueba – 0007: Comprar un token (obra de arte)

Tabla 8. Caso de prueba para validar la compra de un token

<b>ID</b>	CP-0007
<b>Objetivo</b>	Validar que un cliente pueda comprar un token correctamente
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El token está en venta y el comprador tiene suficientes fondos.
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• ID del Token: 1</li> <li>• Valor: 100</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `comprarToken` y pasarle como parámetros el ID del token a comprar y el monto de ETH que cuesta el token</li> </ol>

<b>Resultado Esperado</b>	2. Verificar que el token ha cambiado correctamente de propietario y los fondos se transfirieron correctamente El cliente compra el token correctamente y los fondos se transfieren al propietario anterior.
---------------------------	---

#### 4.8. Caso de Prueba – 0008: Rechazar la compra de un token que no esta en venta

Tabla 9. Caso de Prueba para rechazar la compra de un token que no está en venta

<b>ID</b>	CP-0008
<b>Objetivo</b>	Verificar que la compra de un token sea rechazada si el token no está en venta.
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El token no está en venta.
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• ID del Token: 1</li> <li>• Valor: 100</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `comprarToken` y pasarle como parámetros el ID del token que no está disponible para la venta y el monto de ETH que cuesta el token</li> <li>2. Verificar que la compra del token es rechazada.</li> </ol>
<b>Resultado Esperado</b>	Se rechaza la compra del token si no está en venta

#### 4.9. Caso de Prueba – 0009: Rechazar la compra de un token si el comprador es el propietario del token

Tabla 10. Caso de prueba para rechazar la compra de un token si el comprador es el mismo propietario

<b>ID</b>	CP-0008
<b>Objetivo</b>	Verifica que la compra de un token sea rechazada si el llamador ya es el propietario del token.
<b>Criticidad</b>	Media
<b>Precondiciones</b>	El token que se va a comprar sea del mismo propietario
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• ID del Token: 1</li> <li>• Valor: 100</li> <li>• Cliente: Dirección de Ethereum del mismo propietario del token</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `comprarToken` desde la dirección Ethereum del mismo propietario del token y pasarle como parámetros el ID del token y el monto de ETH que cuesta el token y</li> <li>2. Verificar que la compra del token es rechazada.</li> </ol>
<b>Resultado Esperado</b>	Se rechaza la compra del token si el llamador ya es el propietario.

#### 4.10. Caso de Prueba – 0010: Rechazar la compra de un token si no se tienen los fondos suficientes

Tabla 11. Caso de prueba para rechazar la compra de un token si no se tiene los fondos suficientes

<b>ID</b>	CP-0010
<b>Objetivo</b>	Verificar que la compra de un token sea rechazada si el cliente no tiene suficientes fondos
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El cliente tiene fondos insuficientes.
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• ID del Token: 1</li> <li>• Valor: 5</li> <li>• Cliente: Dirección de Ethereum del cliente que intenta comprar el token</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `comprarToken` y pasarle como parámetros el ID del token y el monto de ETH insuficientes</li> <li>2. Verificar que la compra del token es rechazada.</li> </ol>
<b>Resultado Esperado</b>	Se rechaza la compra del token si los fondos son insuficientes.

#### 4.11. Caso de Prueba – 0011: Cambiar el precio de un token

Tabla 12. Caso de prueba para validar el cambio de precio de un token por parte de su propietario

<b>ID</b>	CP-0011
<b>Objetivo</b>	Verificar que el propietario de un token pueda cambiar su precio correctamente.
<b>Criticidad</b>	Media
<b>Precondiciones</b>	El llamador es el propietario del token
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• ID del Token: 1</li> <li>• Nuevo Precio: 200</li> <li>• Propietario: Dirección de Ethereum del propietario del token</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `cambiarPrecioToken` desde la dirección del propietario y pasarle como parámetros el ID del token y su nuevo precio</li> <li>2. Verificar que el precio del token se ha cambiado correctamente.</li> </ol>
<b>Resultado Esperado</b>	El precio del token se cambia correctamente.

#### 4.12. Caso de Prueba – 0012: Rechazar el cambio de precio de un token si no se es propietario del token

Tabla 13. Caso de prueba para rechazar el cambio de precio de un token si no se es propietario del token

<b>ID</b>	CP-0012
<b>Objetivo</b>	Verificar que el cambio de precio de un token sea rechazado si el llamador no es el propietario

<b>Criticidad</b>	Media
<b>Precondiciones</b>	Ninguna
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• ID del Token: 1</li> <li>• Nuevo Precio: 200</li> <li>• Usuario: Dirección de Ethereum que no es propietario del token</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `cambiarPrecioToken` desde la dirección del usuario que no es propietario del token y pasarle como parámetros el ID del token y su nuevo precio</li> <li>2. Verificar que el cambio de precio es rechazado.</li> </ol>
<b>Resultado Esperado</b>	Se rechaza el cambio de precio si quien realiza el cambio no es propietario del token

#### 4.13. Caso de Prueba – 0013: Cambiar estado de venta de un token

Tabla 14. Caso de prueba para cambiar el estado de venta de un token por parte de su propietario

<b>ID</b>	CP-0013
<b>Objetivo</b>	Verificar que el propietario de un token pueda cambiar el estado de venta correctamente.
<b>Criticidad</b>	Media
<b>Precondiciones</b>	El llamador es el propietario del token
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• ID del Token: 1</li> <li>• Propietario: Dirección de Ethereum del propietario del token</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Llamar a la función `cambiarEstadoVenta` desde la dirección del propietario y pasarle como parámetros el ID del token que se desea cambiar su estado de venta y</li> <li>2. Verificar que el estado de venta del token se cambia correctamente.</li> </ol>
<b>Resultado Esperado</b>	El estado de venta del token se cambia correctamente.

#### 4.14. Caso de Prueba – 0014: Rechazar el cambio de estado de venta de un token si no se es propietario del token

Tabla 385. Caso de prueba para rechazar el cambio de estado de venta de un token si no se es propietario del mismo

<b>ID</b>	CP-0014
<b>Objetivo</b>	Verificar que el cambio de estado de venta de un token sea rechazado si el llamador no es el propietario
<b>Criticidad</b>	Media
<b>Precondiciones</b>	Ninguna
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• ID del Token: 1</li> <li>• Usuario: Dirección de Ethereum que no es propietario del token</li> </ul>

<b>Pasos a seguir</b>	3. Llamar a la función `cambiarEstadoVenta` desde la dirección del usuario que no es propietario del token y pasarle como parámetros el ID del token que se desea cambiar su estado de venta
<b>Resultado Esperado</b>	1. Verificar que el cambio de estado de venta es rechazado. Se rechaza el cambio de estado de venta si quien realiza el cambio no es propietario del token

#### 4.15. Caso de Prueba – 0015: Obtener la lista de tokens de un propietario

Tabla 16. Caso de prueba para validar que se obtengan correctamente los tokens de un propietario

<b>ID</b>	CP-0015
<b>Objetivo</b>	Verificar que se pueda obtener la lista de tokens de un propietario correctamente
<b>Criticidad</b>	Media
<b>Precondiciones</b>	El propietario tiene tokens creados.
<b>Entradas</b>	<ul style="list-style-type: none"> <li>Propietario: Dirección de Ethereum del propietario de los tokens</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>Llamar a la función `getListaTokensPropietario` y pasarle como parámetro la dirección del propietario</li> <li>Verificar que se obtiene la lista de tokens del propietario correctamente.</li> </ol>
<b>Resultado Esperado</b>	Verificar que se obtiene la lista de tokens correctamente.

#### 4.16. Caso de Prueba – 0016: Listar todos los tokens que se han creado correctamente

Tabla 17. Caso de prueba para verificar que se listen todos los tokens que se han creado

<b>ID</b>	CP-0015
<b>Objetivo</b>	Verificar que se pueda listar todos los tokens existentes
<b>Criticidad</b>	Media
<b>Precondiciones</b>	Hay tokens creados
<b>Entradas</b>	Ninguna
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>Llamar a la función `listarTodosLosTokens`</li> <li>Verificar que se obtiene la lista de tokens correctamente.</li> </ol>
<b>Resultado Esperado</b>	Verificar que se obtiene la lista de tokens correctamente.

## 6. Ejecución y resultados de los casos de prueba

Para llevar a cabo estas pruebas, se emplearán herramientas y frameworks como Truffle y OpenZeppelin, los cuales proporcionan un entorno robusto para la creación y ejecución de pruebas unitarias y de integración. Cada prueba ha sido diseñada para cubrir un aspecto específico del contrato, asegurando así una cobertura completa y exhaustiva del sistema.

Cada caso de prueba fue automatizado mediante la herramienta de software Truffle que permitió ejecutar y validar cada uno, obteniendo como resultado lo siguiente.

En la **Tabla** , se observan los resultados obtenidos con la ejecución de los casos de prueba

**Tabla 18.** Resultados de la ejecución de cada caso de prueba

<b>ID Caso de Prueba</b>	<b>¿OK?</b>	<b>Observación</b>
CP-0001	✓	N/A
CP-0002	✓	N/A
CP-0003	✓	N/A
CP-0004	✓	N/A
CP-0005	✓	N/A
CP-0006	✓	N/A
CP-0007	✓	N/A
CP-0008	✓	N/A
CP-0009	✓	N/A
CP-0010	✓	N/A
CP-0011	✓	N/A
CP-0012	✓	N/A
CP-0013	✓	N/A
CP-0014	✓	N/A
CP-0015	✓	N/A
CP-0016	✓	N/A

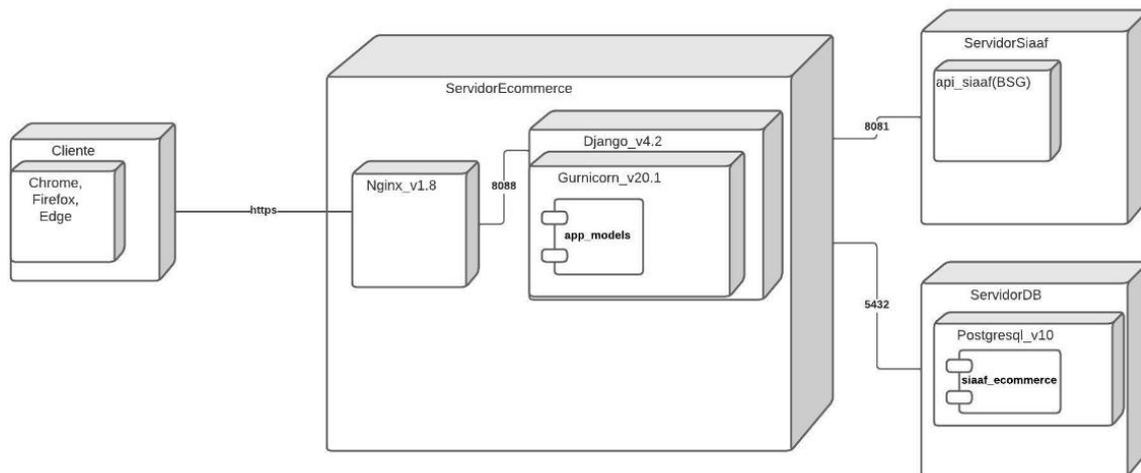
A continuación, en la Figura se muestra la ejecución de pruebas automatizadas mediante truffle.

```
Contract: SDVOAContract
  ✓ Se debe remover un responsable (44ms)
Crear Token (obra de arte)
  ✓ Se debe crear un nuevo token junto con su proceso (154ms)
  ✓ No se debe crear un token con una URI(metadatos) existente (573ms)
  ✓ No se debe crear un token con proceso URI(metadatos) existente (384ms)
  ✓ No se debe crear un token por usuario que no sea responsable (45ms)
Comprar un Token (Obra de arte)
  ✓ Se debe comprar un token (132ms)
  ✓ No debe permitir comprar un token que no está en venta (74ms)
  ✓ No debe permitir comprar un token si es el propietario
  ✓ No debe permitir comprar un token con fondos insuficientes (58ms)
Cambiar Precio del Token
  ✓ Se debería cambiar el precio de un token (64ms)
  ✓ No debe permitir cambiar el precio de un token por un usuario que no sea propietario del token
Cambiar Estado de Venta
  ✓ Se debe cambiar el estado de venta de un token (61ms)
  ✓ No debe permitir cambiar el estado de venta de un token por un usuario que no sea propietario del token
Obtener tokens de un Propietario
  ✓ Se debe obtener la lista de tokens de cada propietario
Listar todos los tokens
  ✓ Se debe listar todos los tokens existentes (47ms)

16 passing (6s)
```

Figura 2. Ejecución de las pruebas automatizadas mediante Truffle

## Anexo 9. Arquitectura del sistema de difusión y ventas de obras de arte



*Nota.* Tomado de “Desarrollo de una Aplicación Web para la difusión y venta de obras de arte para la carrera de Artes Plásticas / Visuales de la Universidad Nacional de Loja” de C.E. Medina, M.A, Rojas, 2023, p, 35 (<https://dspace.unl.edu.ec/jspui/handle/123456789/28682>.)

**Anexo 10.** Plan de pruebas unitarias para el subsistema de aplicaciones

**Proyecto:** Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas

Versión: 1.0  
Fecha: 07/08/202

## Hoja de control

<b>Organismo</b>	Universidad Nacional de Loja		
<b>Proyecto</b>	Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas		
<b>Entregable</b>	Pruebas Funcionales		
<b>Autor</b>	Alexander Vicente		
<b>Versión/Edición</b>	1.0	<b>Fecha Versión</b>	24/06/2024
<b>Aprobado por</b>	Cristian Ramiro Narváez Guillen, Mg.Sc	<b>Fecha Aprobación</b>	26/06/2024
		<b>N° Total de Páginas</b>	14

## 1. Introducción

En el presente documento se muestra el plan de pruebas unitarias definido para probar el comportamiento del subsistema de aplicaciones. Se probará cada funcionalidad nueva creada para dar trazabilidad a las obras de arte.

## 2. Objetivo

El objetivo de este plan pruebas unitarias es validar de manera aislada la implementación y comportamiento de las nuevas funcionalidades desarrolladas en el subsistema de aplicaciones para dar trazabilidad a las obras de arte.

## 3. Alcance

El plan de pruebas abarca la validación de las siguientes funcionalidades del subsistema de aplicaciones:

- Creación de usuarios con una dirección de Ethereum
- Registro un proceso de creación
- Registro de los pasos del proceso
- Creación de un artículo (obra) con su proceso

## 4. Casos de prueba

A continuación, se describe cada caso de prueba definido para validar que el comportamiento de cada funcionalidad vaya de acorde a lo esperado

### 4.1. Caso de Prueba – 0001: Registro de un usuario con una dirección de Ethereum

Tabla 39. Caso de Prueba para validar el registro exitoso de un usuario con una dirección de Ethereum

<b>ID</b>	CP-0001
<b>Objetivo</b>	Validar que se registre un usuario con una dirección de Ethereum en el sistema
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	Ninguna
<b>Entradas</b>	<ul style="list-style-type: none"><li>• web3_address: "0x1234567890abcdef1234567890abcdef12345678"</li><li>• activo: True</li><li>• nombre: "Test User"</li><li>• correo_electronico: "testuser@example.com descripcion: "Test User Description"</li><li>• force_password: False</li></ul>

	<ul style="list-style-type: none"> <li>• is_admin: True</li> <li>• token: "test_token"</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Crear un usuario con los atributos especificados</li> <li>2. Verificar que los atributos del usuario coincidan con los valores especificados.</li> </ol>
<b>Resultado Esperado</b>	Los atributos del usuario coinciden con los valores especificados.

#### 4.2. Caso de Prueba – 0002: Registro Exitoso de un proceso de creación de obra de arte

Tabla 40. Caso de prueba para remover un responsable del contrato

<b>ID</b>	CP-0002
<b>Objetivo</b>	Validar que se registre un proceso de creación de obra de arte correctamente.
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	Crear un usuario
<b>Entradas</b>	Ninguna
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Crear un Proceso asociado al usuario con el nombre "Proceso 1".</li> <li>2. Verificar que los atributos del Proceso coincidan con los valores especificados y que el estado sea "INICIADO".</li> </ol>
<b>Resultado Esperado</b>	Los atributos del Proceso coinciden con los valores especificados y el estado es "INICIADO"

#### 4.3. Caso de Prueba – 0003: Registrar un paso de un proceso

Tabla 41. Caso de prueba para validar el registro de un paso de un proceso

<b>ID</b>	CP-0003
<b>Objetivo</b>	Validar que se registre un paso correctamente.
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	Crear un usuario y un proceso asociado al usuario.
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• nombre:CONCEPTUALIZACION</li> <li>• numero: 1</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Crear un Paso asociado al proceso con los atributos especificados.</li> <li>2. Verificar que los atributos del Paso coincidan con los valores especificados.</li> </ol>
<b>Resultado Esperado</b>	Los atributos del Paso coinciden con los valores especificados.

#### 4.4. Caso de Prueba – 0004: Registrar un artículo correctamente

Tabla 42. Caso de prueba para validar que se registre un artículo exitosamente

<b>ID</b>	CP-0004
<b>Objetivo</b>	Validar que se registre un artículo correctamente.
<b>Criticidad</b>	Media
<b>Precondiciones</b>	Crear un usuario y un proceso asociado al usuario.
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• proceso: Proceso creado</li> <li>• aprobado: "aprobado@example.com"</li> <li>• precio_eth: 1.5</li> <li>• precio: 1500.0</li> <li>• titulo: "Obra de Arte"</li> <li>• descripcion: "Descripción de la obra de arte"</li> <li>• responsable: Usuario creado</li> <li>• atributos: {"color": "azul" }</li> </ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Crear un artículo con los atributos especificados.</li> <li>2. Verificar que los atributos del artículo coincidan con los valores especificados y que el estado sea "PENDIENTE".</li> </ol>
<b>Resultado Esperado</b>	Los atributos del artículo coinciden con los valores especificados y el estado es "PENDIENTE".

## 6. Ejecución y resultados de los casos de prueba

Cada caso de prueba fue automatizado mediante el framework Django y su librería de pruebas TestCase que permitió ejecutar y validar cada uno de los casos de prueba, obteniendo como resultado lo siguiente.

En la **Tabla** , se observan los resultados obtenidos con la ejecución de los casos de prueba

**Tabla 43.** Resultados de la ejecución de cada caso de prueba

<b>ID Caso de Prueba</b>	<b>¿OK?</b>	<b>Observación</b>
CP-0001	✓	N/A
CP-0002	✓	N/A
CP-0003	✓	N/A
CP-0004	✓	N/A

A continuación, en la Figura 47Figura se muestra la ejecución de pruebas automatizadas mediante Test Case.

```

(nft_ecommerce) alex@alex:~/Desktop/siaaf-ecommerce-develop$ python manage.py test
Found 14 test(s).
Creating test database for alias 'default'...
Modificar Cliente
System check identified no issues (0 silenced).
Se creo un nuevo articulo archivo con el nombre images.jpg
..Se creo un nuevo articulo con su proceso y con el titulo Obra de Arte
...Se creo un nuevo paso archivo con el nombre images.jpg
..Se creo un nuevo paso con el nombre CONCEPTUALIZACION
..Se creo un nuevo proceso con el nombre Proceso 1
..Se creo un nuevo usuario con una direccion web3
.
-----
Ran 14 tests in 0.484s

OK
Destroying test database for alias 'default'...

```

Figura 47. Resultados de la Ejecución de las pruebas unitarias automatizadas mediante Test Case

## 5. Glosario

En esta sección se explican los términos técnicos, acrónimos o jerga utilizados en el documento con la finalidad de facilitar su comprensión.

Tabla 44. Glosario de términos

<b>Termino</b>	<b>Descripción</b>
Dirección de Ethereum	Una dirección de Ethereum es una cadena alfanumérica de 42 caracteres que identifica a un usuario

## **Anexo 11.** Plan de pruebas de Integración de la DApp

**Proyecto:** Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas

Versión: 1.0

Fecha: 26/06/2024

## Hoja de control

<b>Organismo</b>	Universidad Nacional de Loja		
<b>Proyecto</b>	Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas		
<b>Entregable</b>	Pruebas Funcionales		
<b>Autor</b>	Alexander Vicente		
<b>Versión/Edición</b>	1.0	<b>Fecha Versión</b>	24/06/2024
<b>Aprobado por</b>	Cristian Ramiro Narváez Guillen, Mg.Sc	<b>Fecha Aprobación</b>	26/06/2024
		<b>N° Total de Páginas</b>	6

## 1. Introducción

Este documento describe las pruebas de integración realizadas para validar la interacción correcta entre el subsistema de aplicaciones y el subsistema de contratos inteligentes. Las pruebas se enfocan en asegurar que todas las funciones transaccionales y de consulta operen correctamente y que la integración entre la interfaz de usuario y los contratos inteligentes sea coherente y efectiva.

## 2. Objetivo

El objetivo de este plan pruebas de integración es validar la integración del subsistema de aplicaciones y subsistema de contratos inteligentes mediante la librería web3.py

## 3. Alcance

El plan de pruebas de integración abarca todas las funciones transaccionales y de consulta entre el subsistema de aplicaciones y contratos inteligentes

## 4. Casos de prueba

A continuación, se describe los casos de prueba definidos para validar la integración de los dos subsistemas.

Tabla 45. Casos de Prueba para validar la integración del subsistema de aplicaciones con el de contratos inteligentes

<b>ID Caso de Prueba</b>	<b>Descripción y Funcionalidad Probada</b>	<b>Pasos de Prueba</b>
<b>CP-001</b>	Registrar Articulo	<ol style="list-style-type: none"><li>1.Llamar a crear_token_obra con su URI, precio y el URI del proceso.</li><li>2.Verificar el estado de retorno y la existencia de la transacción.</li></ol>
<b>CP-002</b>	Comprar un Articulo	<ol style="list-style-type: none"><li>1. Llamar a comprar_token con el ID del token y el valor en wei</li><li>2. Verificar el éxito de la transacción y el uso de gas.</li></ol>
<b>CP-0032</b>	Cambiar el Precio de un Articulo	<ol style="list-style-type: none"><li>1. Llamar a cambiar_precio_token con el ID del token y el nuevo precio en wei<ol style="list-style-type: none"><li>1. 2. Verificar que no haya errores</li></ol></li></ol>
<b>CP-004</b>	Cambiar el Estado de Venta de un Articulo	<ol style="list-style-type: none"><li>1. Llamar a cambiar_estado_venta con el ID del token.</li><li>2. Verificar que no haya errores</li></ol>

<b>CP-005</b>	Obtener Historial de Transacciones	<ol style="list-style-type: none"> <li>1. Llamar a getHistorial con el ID del token</li> <li>2. Verificar que se obtenga un historial no vacío de transacciones.</li> </ol>
<b>CP-006</b>	Obtener Detalles de un Token	<ol style="list-style-type: none"> <li>1. Llamar a get_token con el ID del token.</li> <li>2. Verificar que se obtengan detalles válidos del token.</li> </ol>
<b>CP-007</b>	Listar Todos los Articulos Disponibles	<ol style="list-style-type: none"> <li>1. Llamar a listar_tokens.</li> <li>2. Verificar que se obtenga una lista no vacía de artículos</li> </ol>
<b>CP-008</b>	Listar Articulos de un Usuario	<ol style="list-style-type: none"> <li>1. Llamar a listar_tokens_usuario con la cuenta de usuario.</li> <li>2. Verificar que se obtenga una lista no vacía de tokens.</li> </ol>
<b>CP-009</b>	Obtener Proceso Asociado a un Articulo	<ol style="list-style-type: none"> <li>1. Llamar a get_proceso_token con el ID del token.</li> <li>2. Verificar que se obtenga información válida sobre el proceso asociado al token.</li> </ol>

## 5. Ejecución y resultados de los casos de prueba

En la **Figura 48**, se muestra la ejecución de los casos de prueba mediante la librería Pytest.

```

===== test session starts =====
Platform linux -- Python 3.10.12, pytest-8.2.2, pluggy-1.5.0
rootdir: /home/alex/Desktop/siaaf-ecommerce-develop
configfile: pytest.ini
plugins: web3-6.19.0
collected 12 items

app/ecommerce/test/test_integration.py ..... [100%]

===== 12 passed in 2.60s =====

```

Figura 48. Resultado de la Ejecución mediante pytest

En la **Tabla 46**, se presenta un resumen con los resultados de la ejecución de las pruebas.

Tabla 46. Resultados de la Ejecución de los casos de Prueba

ID Caso de Prueba	Completado con Éxito	Observaciones
<b>CP-001</b>	Sí	-
<b>CP-002</b>	Sí	-
<b>CP-0032</b>	Sí	-.
<b>CP-004</b>	Sí	-
<b>CP-005</b>	Sí	-
<b>CP-006</b>	Sí	-
<b>CP-007</b>	Sí	-

<b>CP-008</b>	Sí	-
<b>CP-009</b>	Sí	-

## 6. Glosario

En esta sección se explican los términos técnicos, acrónimos o jerga utilizados en el documento con la finalidad de facilitar su comprensión.

Tabla 47. Glosario de términos

<b>Termino</b>	<b>Descripción</b>
Token(s)	Representación de la obra de arte

**Anexo 12.** Plan de pruebas Funcionales para la DApp

**Proyecto:** Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas

Versión: 1.0

Fecha: 24/08/202

### Hoja de control

---

<b>Organismo</b>	Universidad Nacional de Loja		
<b>Proyecto</b>	Trazabilidad mediante blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas		
<b>Entregable</b>	Pruebas Funcionales		
<b>Autor</b>	Alexander Vicente		
<b>Versión/Edición</b>	1.0	<b>Fecha Versión</b>	24/06/2024
<b>Aprobado por</b>	Cristian Ramiro Narváez Guillen, Mg.Sc	<b>Fecha Aprobación</b>	26/06/2024
		<b>Nº Total de Páginas</b>	21

---

## 1. Introducción

En el presente documento se muestra las pruebas funcionales realizadas a la DApp, para garantizar calidad y validar que se cumplan con los requisitos definidos.

## 2. Objetivo

El objetivo principal de este plan de pruebas es generar casos de prueba que validen cada uno de los requisitos definidos, garantizando la calidad de la DApp.

## 3. Alcance

El plan de pruebas abarcará todo el sistema, centrándose en los requisitos funcionales.

## 4. Matriz de Trazabilidad: Casos de Prueba – Requisitos Funcionales

La matriz mostrada en la tabla indica la relación entre los casos de prueba y los requisitos funcionales definidos para la DApp. Esta matriz es fundamental para asegurar que todos los requisitos han sido cubiertos por los casos de prueba y para facilitar el seguimiento y control durante el proceso de pruebas.

Tabla 48. Matriz de Trazabilidad

	<b>RF01</b>	<b>RF02</b>	<b>RF03</b>	<b>RF04</b>	<b>RF05</b>	<b>RF06</b>	<b>RF07</b>	<b>RF08</b>	<b>RF09</b>
<b>CP-001</b>	X								
<b>CP-002</b>		X							
<b>CP-003</b>			X				X		X
<b>CP-004</b>					X		X		X
<b>CP-005</b>						X	X		X
<b>CP-006</b>				X			X		X
<b>CP-007</b>								X	

## 5. Glosario

En esta sección se explican los términos técnicos, acrónimos o jerga utilizados en el documento con la finalidad de facilitar su comprensión.

Tabla 49. Glosario de términos

<b>Termino</b>	<b>Descripción</b>
Metamask	Gestor de billeteras

## 6. Casos de prueba

A continuación, se describen cada caso de prueba funcionales para probar la DApp

### 6.1. Caso de Prueba – 0001: Registrar un Proceso completo de la creación de una obra de arte

**Tabla 50.** Caso de Prueba para validar el registro de un proceso completo por parte de un responsable

ID	CP-0001
<b>Objetivo</b>	Verificar que un responsable pueda registrar el proceso de creación completo de una obra de arte.
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado como responsable. El usuario debe estar en su interfaz de administración
<b>Entradas</b>	Nombre del Proceso: Prueba Datos del Paso 1: <ul style="list-style-type: none"><li>• Origen de la idea: Describir de dónde surgió la idea para la obra de arte.</li><li>• Elegir tipo de medio: Seleccionar entre "Digital" o "Físico".</li></ul> Datos del Paso 2: <ul style="list-style-type: none"><li>• Describir el estilo artístico y detalles del boceto.</li><li>• Boceto: Subir una imagen del boceto</li></ul> Datos del Paso 3: <ul style="list-style-type: none"><li>• Describir los materiales que se usarán en la obra y su aplicación.</li><li>• Materiales: Subir una imagen de los materiales</li></ul> Datos del Paso 4: <ul style="list-style-type: none"><li>• Describir el progreso de la obra.</li><li>• Creación de la obra: Subir una imagen del avance de la obra</li></ul> Datos del Paso 5: <ul style="list-style-type: none"><li>• Describir los detalles finales y acabados de la obra.</li><li>• Detalles y acabados de la obra: Subir una imagen de los detalles y acabados de la obra</li></ul> Datos de la obra: <ul style="list-style-type: none"><li>• Imagen: Subir una imagen de la obra</li><li>• Título: Colocar un título a la obra</li><li>• Descripción: Colocar una descripción a la obra</li><li>• Precio: Designar un precio a la obra</li><li>• Atributos: Colocar atributos a la obra</li></ul>
<b>Pasos a seguir</b>	<ol style="list-style-type: none"><li>1. Navegar a la página de Administración de procesos.</li><li>2. Hacer clic en el botón “Nuevo”</li><li>3. Llenar el formulario con los el nombre del proceso</li><li>4. Hacer clic en el botón "Guardar".</li></ol>

5. En la lista de procesos, hacer clic en el botón “ver proceso” del proceso recién creado
6. Hacer clic en el botón “Crear Paso”
7. Rellenar el formulario con los datos del paso 1, paso 2, paso 3, paso 4 y paso 5 respectivamente
8. Luego hacer clic sobre el botón “Finalizar Proceso”
9. Rellenar el formulario con los Datos de la obra
10. Hacer clic sobre el botón “Guardar y Enviar a Revisión”

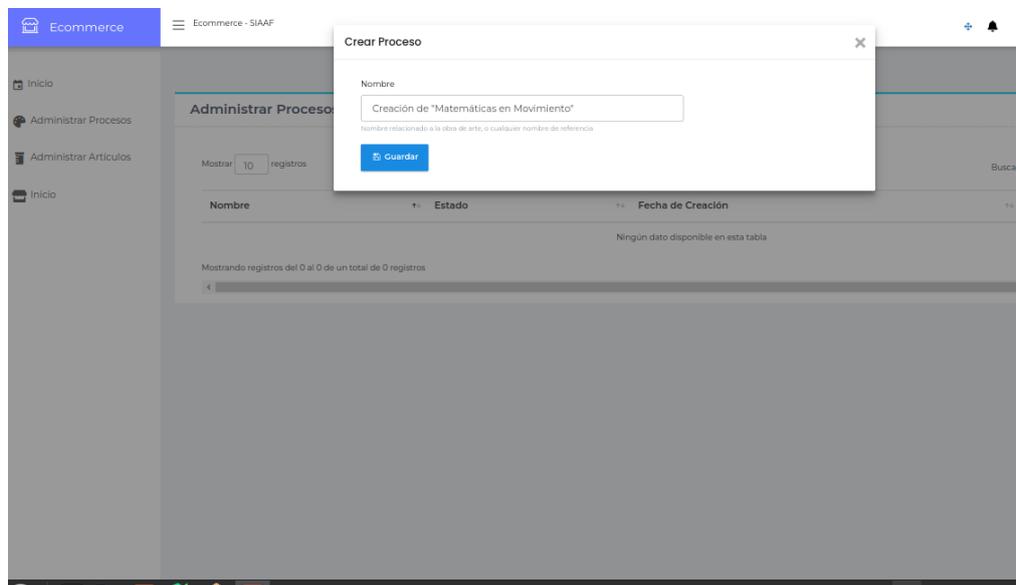
**Resultado Esperado**

Cada paso del proceso se guarda correctamente y se muestra un mensaje de confirmación.

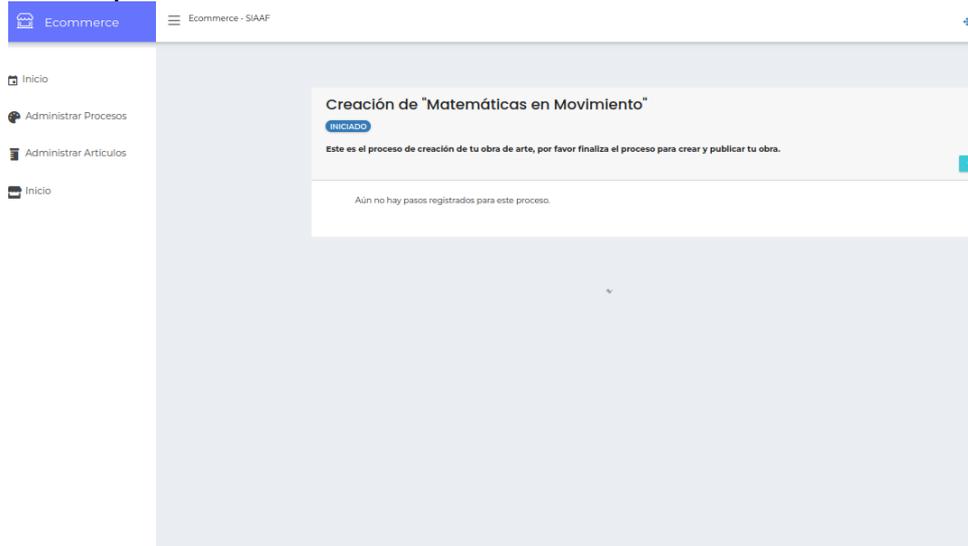
Al finalizar, el proceso completo de la creación de la obra de arte está registrado en la DApp.

**Resultado Obtenido**

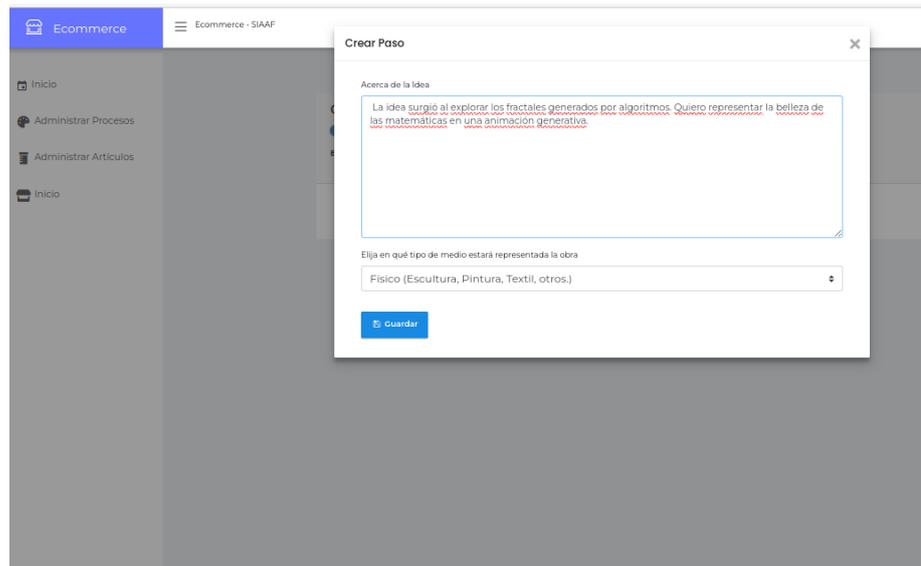
- Creación del proceso



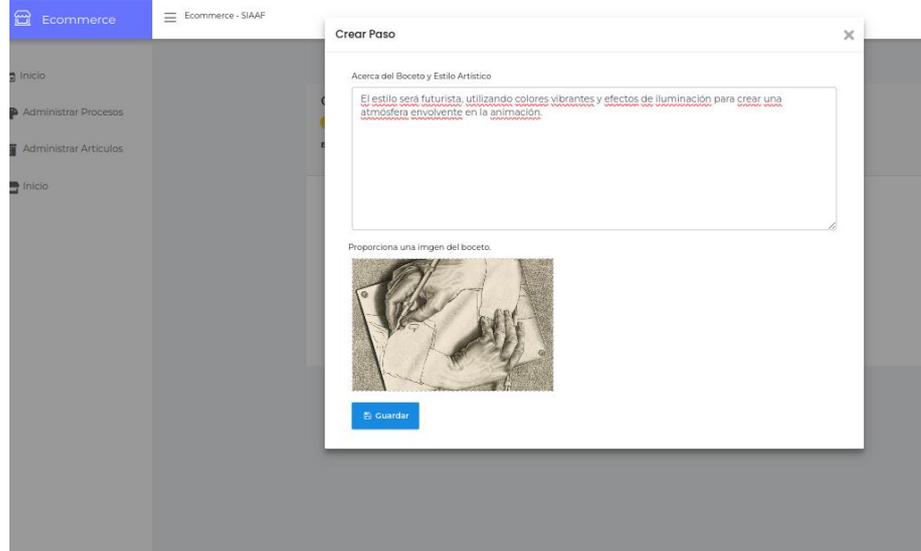
- Ver proceso



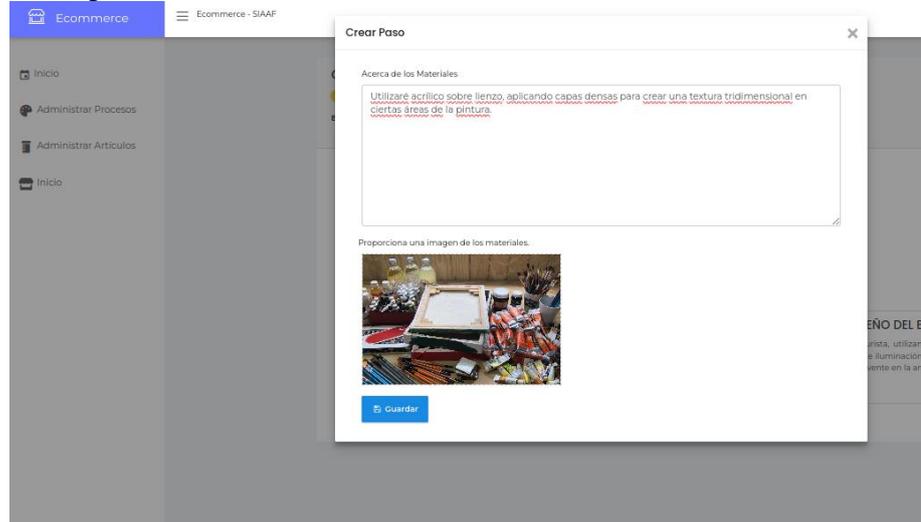
- Crear paso 1



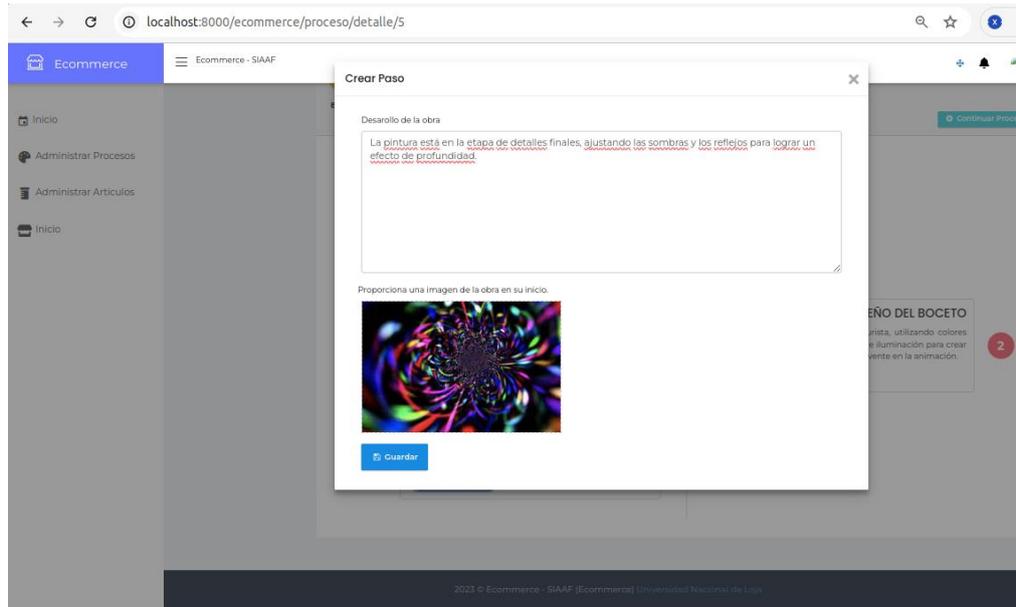
- **Crear Paso 2**



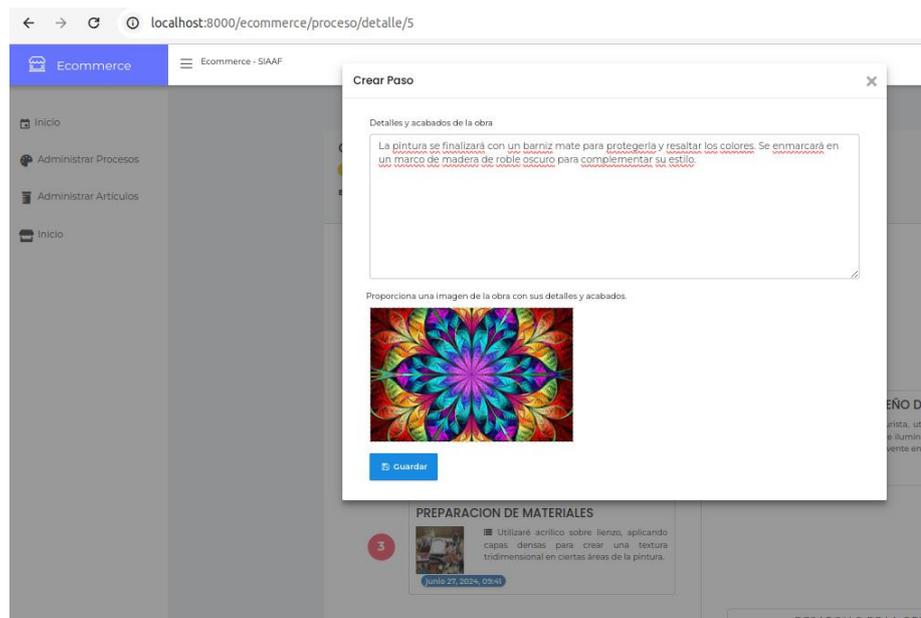
- **Crear paso 3**



- **Crear paso 4**



- Crear Paso 5



- Proceso de creación registrado

Ecommerce - SIAAF

Este es el proceso de creación de tu obra de arte, por favor finaliza el proceso para crear y publicar tu obra.

**1 CONCEPTUALIZACIÓN DE LA IDEA**

Origen de la Idea: La idea surgió al explorar los fractales generados por algoritmos. Quiero representar la belleza de las matemáticas en una animación generativa.

Medio de Representación: Físico

Junio 27, 2024, 09:36

**3 PREPARACION DE MATERIALES**

Utilizaré acrílico sobre lienzo, aplicando capas densas para crear una textura tridimensional en ciertas áreas de la pintura.

Junio 27, 2024, 09:41

**SOBRE EL DISEÑO DEL E**

El estilo será futurista, utilizaré vibrantes y efectos de iluminación para una atmósfera envolvente en la ar

Junio 27, 2024, 09:36

**DESARROLLO DE LA OBRA I**

Descripción: La pintura está repleta de detalles finales, ajustando los colores y reflejos para lograr una profundidad.

Junio 27, 2024, 09:41

- Crear artículo

Ecommerce - SIAAF

**Datos del Artículo**

Título: Matemáticas en Movimiento

Descripción: Esta obra de arte digital que explora la belleza de los fractales y la geometría matemática a través de la pintura física. Busca capturar la esencia de las matemáticas abstractas y complejas, creando una representación visual que utiliza técnicas de pintura tradicional.

Precio en dólares de la obra: 20

Precio en la obra en ETH: 0.005791452684555499

Ver estimado en ETH del precio en dólares.

Subcategoría: TOKENS NFTS

Selecciona una subcategoría

Guardar y Enviar a revisión

**Atributos**

Material Utilizado: Acrílico sobre lienzo

Coloque al menos un atributo (\*)

Estilo Artístico: Impresionista

Técnica Utilizada: Pinceladas sueltas

+ Añadir Atributo

- Artículo registrado

Administrar Artículos

Filtro por Estado: --Todos--

Mostrar 10 registros

Titulo	Precio(\$)	Estado	Creado
Matemáticas en Movimiento	20	PENDIENTE	2024-06-27 09:41

## 6.2. Caso de Prueba – 0002: Registrar un artículo en la blockchain de Ethereum

Tabla 51. Caso de prueba para validar el registro de un artículo en la blockchain de Ethereum por parte de un responsable

<b>ID</b>	CP-0002
<b>Objetivo</b>	Vverificar el registro de un artículo en la blockchain de Ethereum por parte de un responsable
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado como responsable. El responsable debe tener su cuenta de metamask conectada a la DApp El responsable debe estar en su interfaz de administración El articulo a registrar por parte del responsable debe estar aprobado
<b>Entradas</b>	Ninguno
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Navegar a la página de Administración de artículos</li> <li>2. Hacer clic en el botón “Ver artículo”</li> <li>3. Ajustar precio de la obra si es necesario (Opcional)</li> <li>4. Hacer clic en el botón "Publicar Articulo”.</li> <li>5. Hacer click en el botón “si” en la ventana de confirmación</li> <li>6. Esperar a que metamask genera la transacción</li> <li>7. Hacer clic en “Confirmar transacción” en la ventana de metamask</li> <li>8. Esperar a que se termine de la transacción</li> </ol>
<b>Resultado Esperado</b>	<p>El estado del articulo cambia su estado de APROBADO a PUBLICADO COMO NFT</p> <p>Al finalizar la transacción, se muestra el hash de la transacción</p> <p>Al finalizar, se muestra un mensaje de confirmación y el artículo se encuentra disponible en el Inicio de la tienda</p> <p>El articulo debe tener su primera transacción registrada</p> <ul style="list-style-type: none"> <li>• Detalle del articulo</li> </ul>

## Resultado Obtenido

**APROBADO**

**Imagen del Artículo**

**Información del Artículo**

Título: Matemáticas en Movimiento

Descripción: "Matemáticas en Movimiento" es una pintura al óleo que explora la belleza de los fractales y las formas matemáticas a través de la pintura física. Esta obra busca capturar la esencia de las matemáticas en una representación visual que utiliza técnicas de pintura tradicional.

Precio: \$ 20,0

Ajustar precio

Precio en ETH: 0.005791452684555499

**Publicar Artículo**

**Atributos**

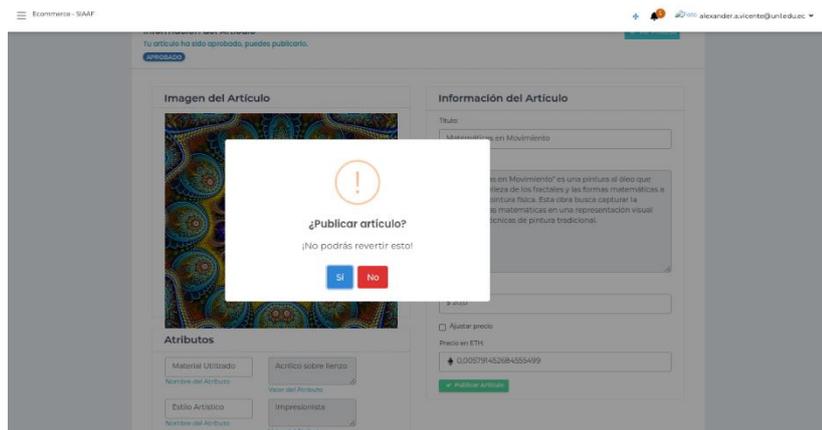
Material Utilizado: Acrílico sobre lienzo

Nombre del Atributo: Valor del Atributo

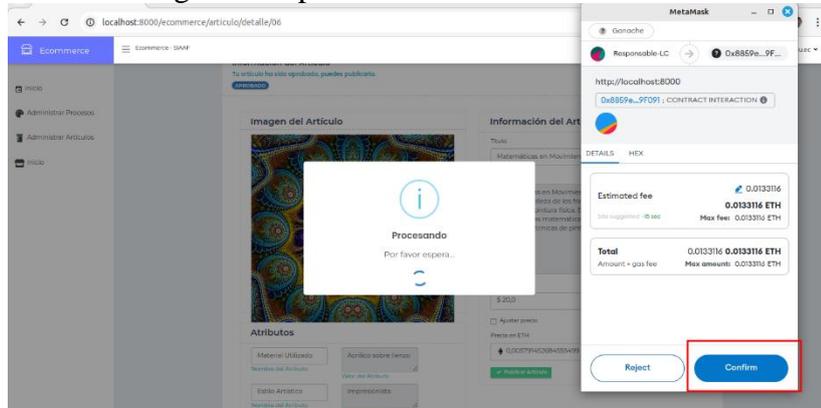
Estilo Artístico: Impresionista

Nombre del Atributo: Valor del Atributo

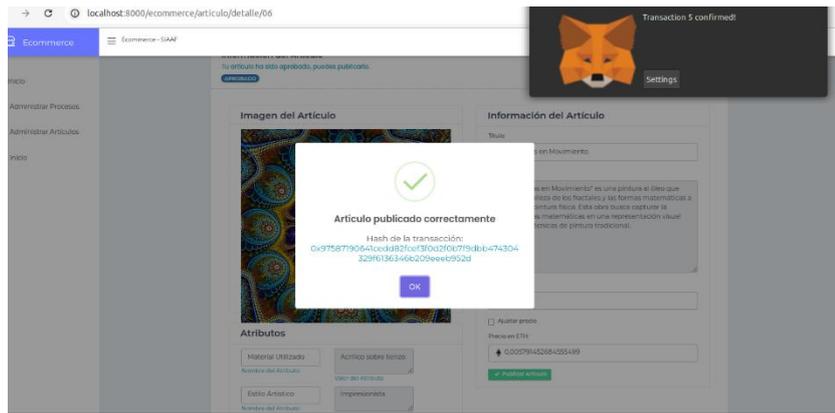
- **Publicación del artículo**



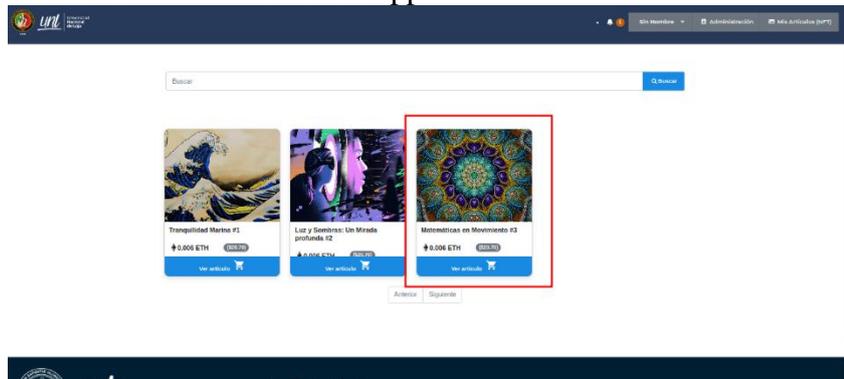
- **Transacción generada por metamask**



- **Hash de la transacción y mensaje de confirmación**



- Artículo en el inicio de la DApp



- Transacción generada por el registro del artículo

Evento	Precio	De	a	Fecha
Mint	0,0133 ETH	0x0000...0000	0xd9F8...Db7d	Hace 3 minutos

### 6.3. Caso de Prueba – 0003: Cambiar Precio de un artículo y Ver mis artículos

Tabla 52. Caso de prueba para validar el cambio de precio de un artículo por parte de su propietario y que se muestren los artículos del propietario

<b>ID</b>	CP-0003
<b>Objetivo</b>	Verificar que el propietario de un artículo pueda modificar su precio y que se muestren los artículos de su propiedad
<b>Criticidad</b>	Alta
<b>Precondicione s</b>	El usuario debe estar autenticado El usuario debe tener artículos en su propiedad El usuario debe estar en la Interfaz principal de la DApp
<b>Entradas</b>	Ninguno

## Pasos a seguir

1. Navegar a la página de “Mis artículos”
2. Hacer clic en el botón “Ver artículo”
3. Hacer clic en el botón "Cambiar Precio".
4. Ingresar el nuevo precio
5. Hacer clic en el botón "Continuar"
6. Esperar a que metamask genera la transacción
7. Hacer clic en “Confirmar transacción” en la ventana de metamask
8. Esperar a que se termine de la transacción

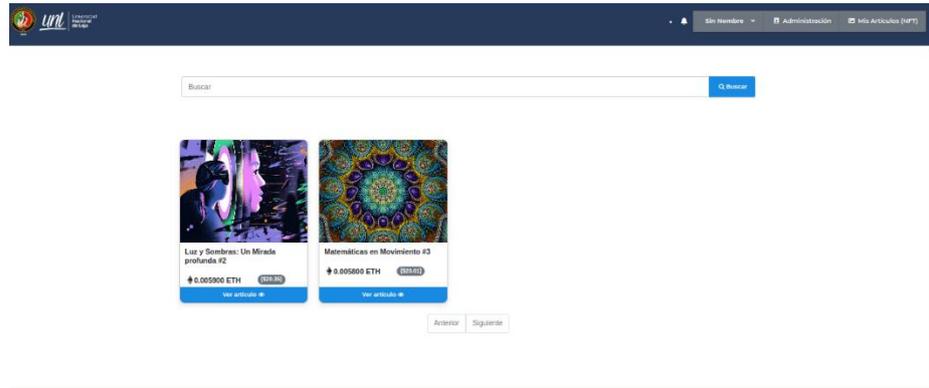
## Resultado Esperado

El precio del artículo se actualiza correctamente y se muestra un mensaje de confirmación.

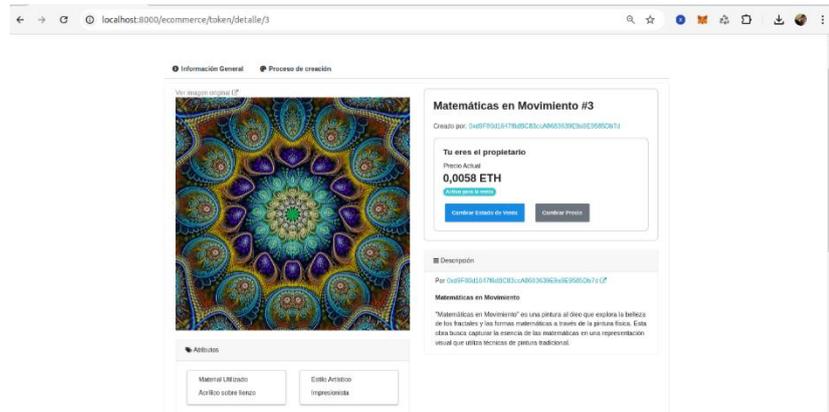
## Resultado Obtenido

Al finalizar la transacción, se muestra el hash de la transacción

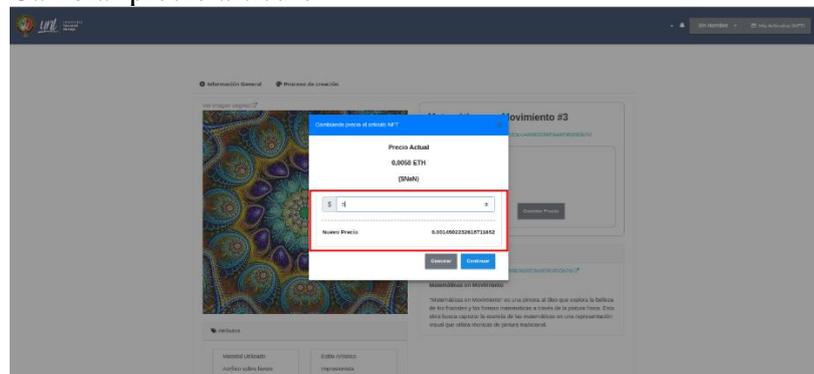
- Sección mis artículos



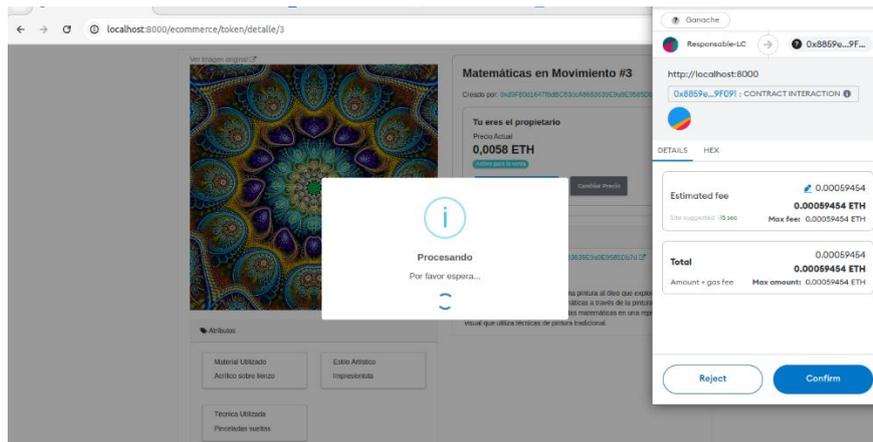
- Detalle de artículo



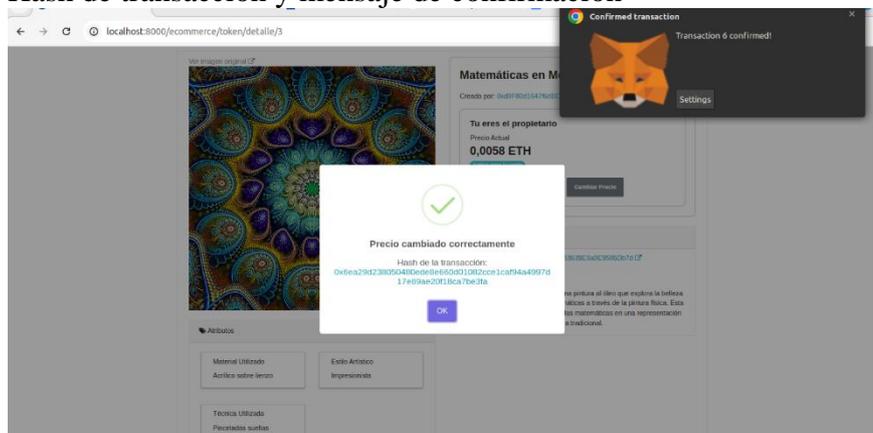
- Cambiar precio artículo



- Transacción generada



- Hash de transacción y mensaje de confirmación



#### 6.4. Caso de Prueba – 0004: Cambiar Estado de venta de un artículo y ver mis artículos

Tabla 53. Caso de prueba para validar el cambio de estado de venta de un artículo por parte de su propietario y que se muestren sus artículos

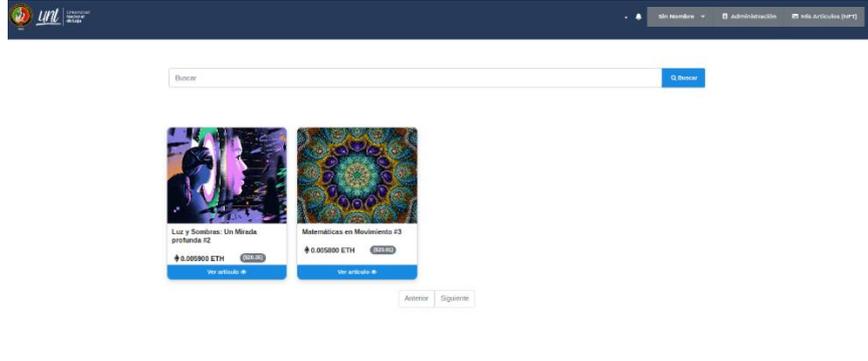
<b>ID</b>	CP-0004
<b>Objetivo</b>	Verificar que el propietario de un artículo pueda cambiar su estado de venta y que se muestren sus artículos
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado El usuario debe tener artículos en su propiedad El usuario debe estar en la Interfaz principal de la DApp
<b>Entradas</b>	Ninguno
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Navegar a la página de “Mis artículos”</li> <li>2. Hacer clic en el botón “Ver artículo”</li> <li>3. Hacer clic en el botón "Cambiar Estado de Venta".</li> <li>4. Hacer clic en el botón "Si" de la ventana de confirmacion</li> <li>5. Esperar a que metamask genera la transacción</li> </ol>

**Resultado Esperado**  
**Resultado Obtenido**

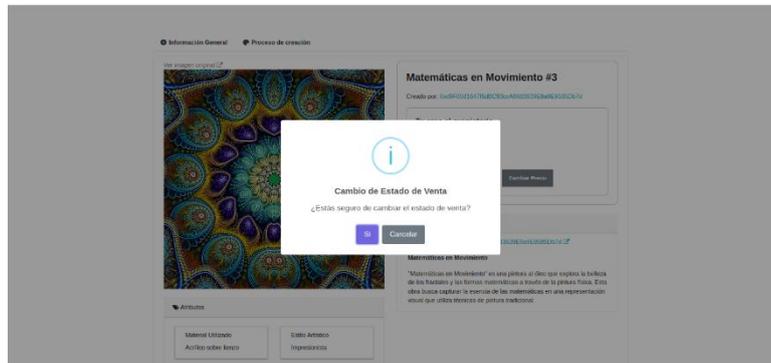
- 6. Hacer clic en “Confirmar transacción” en la ventana de metamask
- 7. Esperar a que se termine de la transacción

El estado de venta del artículo se actualiza correctamente y se muestra un mensaje de confirmación.

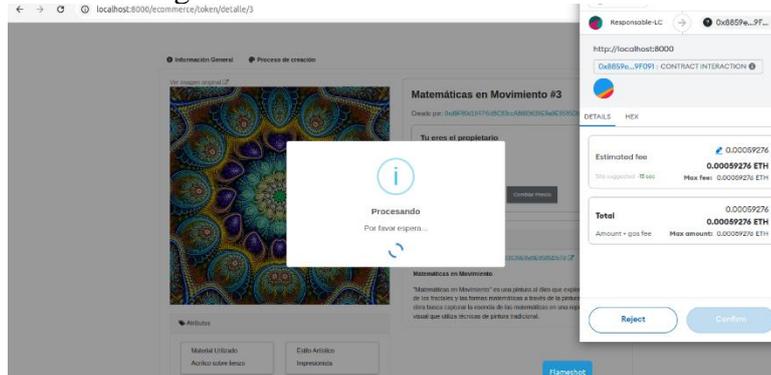
- Sección mis artículos



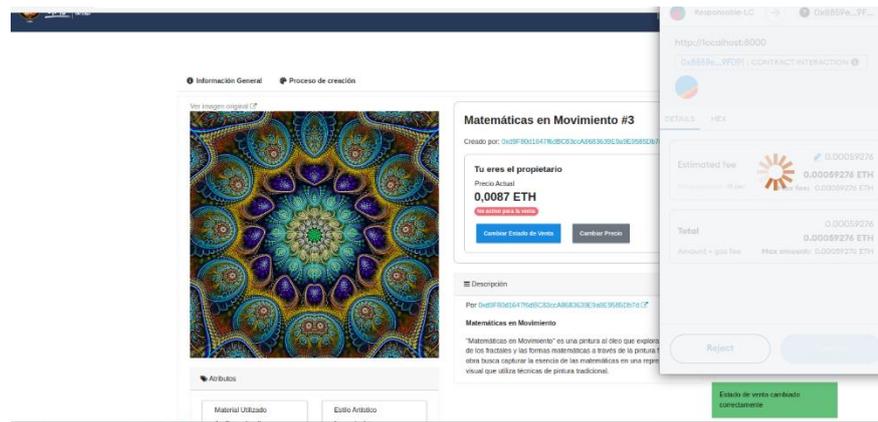
- Cambiar estado de venta



- Transacción generada



- Estado de venta del articulo cambiado

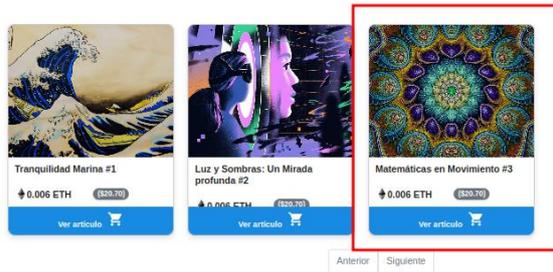


## 6.5. Caso de Prueba – 0005: Ver trazabilidad y detalles de artículo

Tabla 54. Caso de prueba para validar que se muestren los detalles y la trazabilidad del artículo

<b>ID</b>	CP-0004
<b>Objetivo</b>	Verificar que un usuario pueda ver la trazabilidad del artículo, así como todos los datos asociados del artículo.
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar en la Interfaz principal de la DApp Deben existir artículos registrados en la blockchain de Ethereum
<b>Entradas</b>	Ninguno
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Seleccionar un artículo disponible</li> <li>2. Hacer clic en el botón "Ver artículo"</li> <li>3. Revisar la información detallada del artículo, que incluye: <ul style="list-style-type: none"> <li>• Información general del artículo.</li> <li>• Historial de transacciones.</li> <li>• Proceso de creación</li> </ul> </li> <li>4. Hacer clic en "Proceso de Creación"</li> <li>5. Revisar la trazabilidad del artículo, que incluye: <ul style="list-style-type: none"> <li>• Información sobre cada etapa del proceso de creación.</li> <li>• Detalles de cada transacción realizada con el artículo.</li> </ul> </li> </ol>
<b>Resultado Esperado</b>	Se muestra correctamente toda la información detallada del artículo. La trazabilidad del artículo se muestra correctamente, detallando cada etapa del proceso de creación y todas las transacciones realizadas.
<b>Resultado Obtenido</b>	<ul style="list-style-type: none"> <li>• Artículos disponibles</li> </ul>

Q Buscar



[Anterior](#)
[Siguiente](#)

- **Detalle del artículo**

localhost:8000/ecommerce/tokens/detalle/3



**Matemáticas en Movimiento #3**

0.006 ETH (\$20.70)

Ver artículo

**Descripción**

Por 0x49f98...D67d

**Matemáticas en Movimiento**

"Matemáticas en Movimiento" es una pintura al óleo que explora la belleza de los fractales y las formas matemáticas a través de la pintura física. Esta obra busca capturar la esencia de las matemáticas en una representación visual que utiliza técnicas de pintura tradicional.

**Atributos**

Material Utilizado: Acrílico sobre lienzo

Este Artículo: Impresión

Técnica Utilizada: Pintado al óleo

Evento	Precio	De	a	Fecha
Mint	0,0133 ETH	0x0000...0000	0x49f98...D67d	Hace 3 minutos

- **Proceso de creación**

localhost:8000/ecommerce/tokens/detalle/3

Información General | **Proceso de creación**

Proceso de creación Matemáticas en Movimiento

**ASIGNACIÓN**

Nombre Asignado a la Clase: Matemáticas en Movimiento

**CONCEPTUALIZACIÓN**

Origen de la Idea: La idea surgió al explorar los fractales generados por algoritmos. Quería representar la belleza de las matemáticas en una animación generativa.

Medio de Representación: Físico

**BOCETO**

El estilo será futurista, utilizando colores vibrantes y efectos de animación para crear una atmósfera envolvente en la animación.

**MATERIALES**

Utilizaré acrílico sobre lienzo, aplicando capas densas para crear una textura tridimensional en ciertas áreas de la obra.

- **Historial de transacciones**

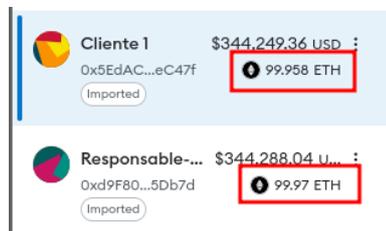
Actividad del artículo

Evento	Precio	De	a	Fecha
Mint	0,0133 ETH	0x0000...0000	0x49f98...D67d	Hace 3 minutos

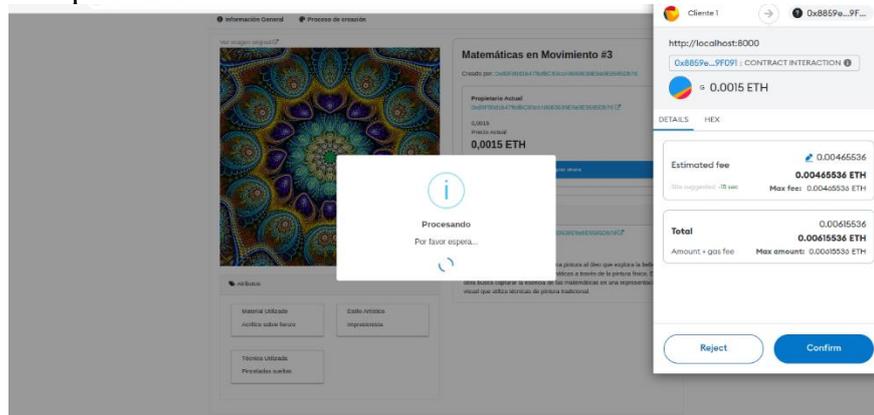
## 6.6. Caso de Prueba – 0006: Comprar Artículo

Tabla 55. Caso de prueba para validar que un cliente pueda comprar un artículo disponible

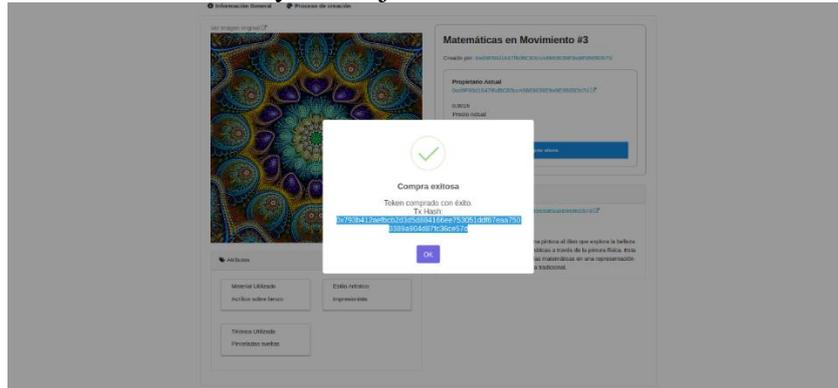
<b>ID</b>	CP-0006
<b>Objetivo</b>	Caso de prueba para validar que se muestren los detalles y la trazabilidad del artículo
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El usuario debe estar autenticado El usuario debe tener su cuenta de metamask conectada a la DApp El usuario debe estar en la interfaz principal de la DApp
<b>Entradas</b>	Ninguno
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Seleccionar un artículo disponible de la lista</li> <li>2. Hacer clic en el botón “Ver artículo”</li> <li>3. Hacer clic en el botón "Comprar".</li> <li>4. Esperar a que metamask genera la transacción</li> <li>5. Hacer clic en “Confirmar transacción” en la ventana de metamask</li> <li>6. Esperar a que se termine de la transacción</li> </ol>
<b>Resultado Esperado</b>	La compra se realiza correctamente y se muestra un mensaje de confirmación. Al finalizar la transacción, se muestra el hash de la transacción El artículo debe pasar a propiedad del cliente El artículo debe tener registrada la transacción realizada en su historial
<b>Resultado Obtenido</b>	<ul style="list-style-type: none"> <li>• Saldo en cuentas antes de la compra</li> </ul>



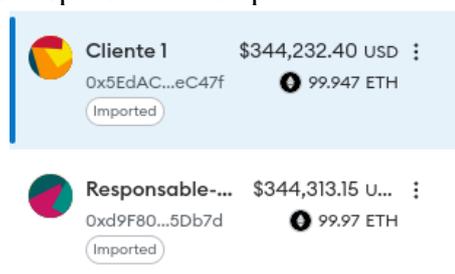
- Comprar Artículo



- Hash de transacción y mensaje de confirmación



- Saldo de cuentas después de la compra

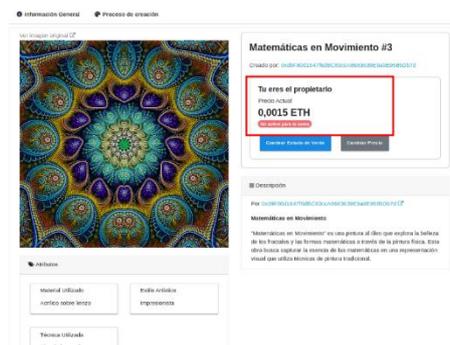


- Transacción de compra registrada en el historial

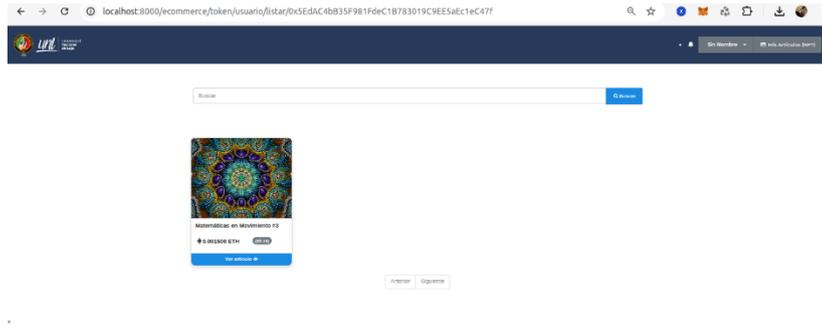
Actividad del artículo

Evento	Precio	De	a	Fecha
Mint	0,0133 ETH	0x0000...0000	0x9F8...Db7d	Hace 45 minutos
Venta	0,0047 ETH	0x9F8...Db7d	0x5EdA...C47f	Hace un momento

- Artículo transferido al cliente



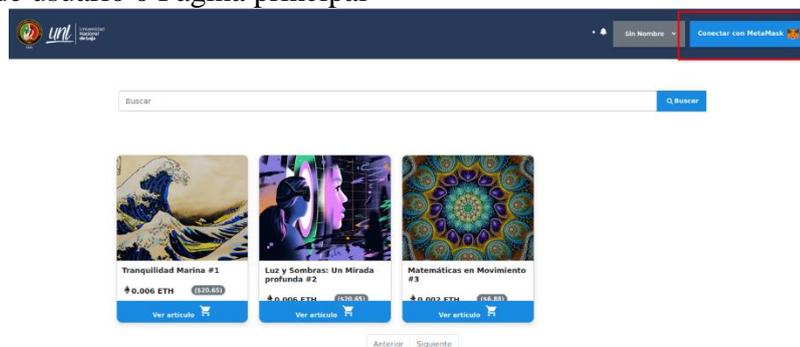
- Artículos del cliente

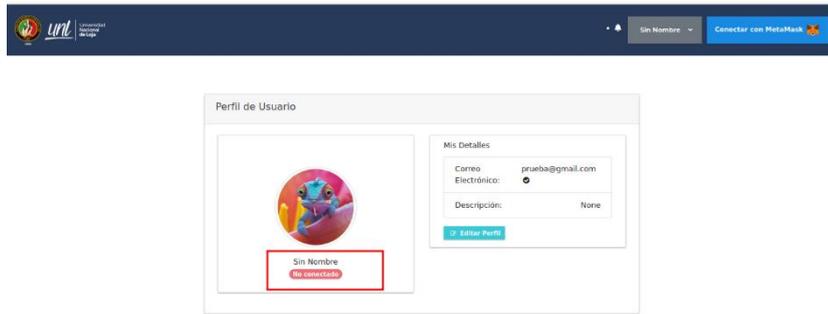


## 6.7. Caso de Prueba – 0007: Conectar Billetera

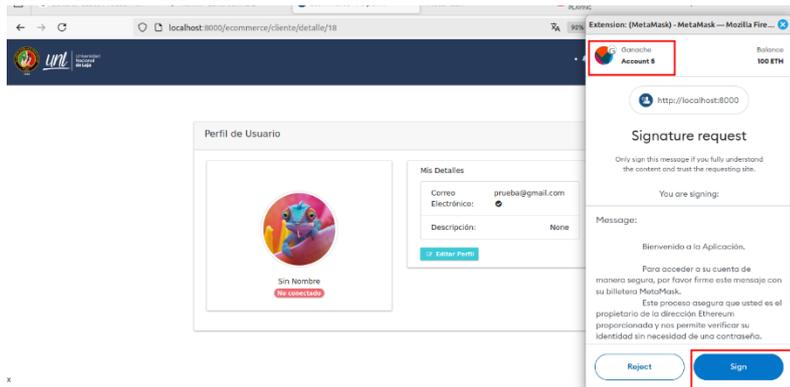
Tabla 56. Caso de prueba para validar que un usuario puede conectar su billetera de Ethereum

<b>ID</b>	CP-0007
<b>Objetivo</b>	Verificar que un usuario pueda conectar su billetera de Ethereum a la DApp mediante metamask
<b>Criticidad</b>	Alta
<b>Precondiciones</b>	El usuario no debe tener su cuenta de metamask asociada a la DApp El usuario debe estar en la intefaz principal de la DApp
<b>Entradas</b>	Ninguno
<b>Pasos a seguir</b>	<ol style="list-style-type: none"> <li>1. Hacer clic en el botón “Conectar con Metamask”</li> <li>2. Esperar a que metamask genera una solicitud de firma para autenticar el usuario</li> <li>3. Hacer clic en “Firmar” en la ventana de metamask</li> <li>4. Esperar a que se termine la validación</li> </ol>
<b>Resultado Esperado</b>	La billetera se conecta correctamente y se muestra un mensaje de confirmación El usuario queda registrado con su billetera Se activan las funciones que necesitan de la conexión con metamask
<b>Resultado Obtenido</b>	Perfil de usuario o Página principal

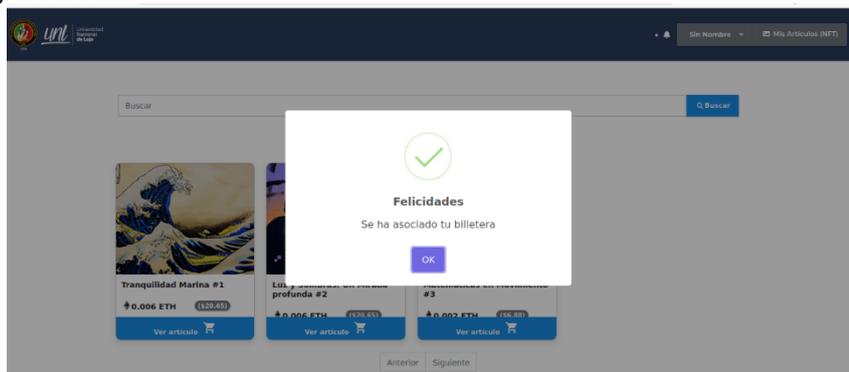




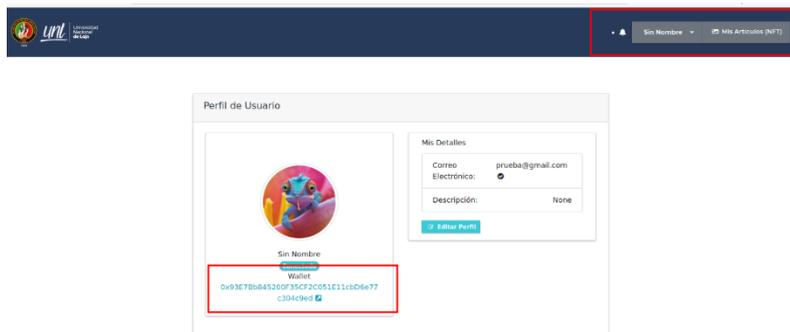
## Conexión con metamask



## Mensaje de confirmación



## Activación de funciones





## Acuerdo de confidencialidad de NO divulgación de información - Prácticas Preprofesionales y Proyectos de Titulación

Conste por el presente documento, el Acuerdo de Confidencialidad y NO divulgación de la información, que celebran por una parte la Universidad Nacional de Loja a través de la Unidad de Telecomunicaciones e Información, a quien para efectos del presente Acuerdo se denominará la

Universidad, y por otra el Sr. (a) Alexander Adrián Vicente Jiménez, perteneciente a la Carrera de Computación de la Institución Universidad Nacional de Loja perteneciente a quien en adelante se le denominará el Practicante o Tesista, de acuerdo a la situación que lo amerite.

Las partes se reconocen recíprocamente con capacidad de obligarse y al efecto suscriben el presente Acuerdo bajo las siguientes condiciones:

### DECLARACIÓN

I.- La Universidad declara que:

- a) Es una entidad que brinda servicios académicos en apego a lo dispuesto por la Ley de Educación Superior y su reglamento, disposiciones del organismo de control y demás legislación aplicable.
- b) Toda información relacionada con conocimientos técnicos; modos de trabajo adquiridos con el tiempo; tecnologías; diseños gráficos; estrategias de mercado; estrategias de competencia; procesos; distintivos (diseños, logotipos, lemas, etc.); administración de recursos materiales y humanos; datos de proveedores de bienes y servicios; cartera de socios y clientes; estadísticas y estudios de mercado; manuales de políticas y procedimientos; estatutos y reglamentos de actividad laboral, bases de datos; y, en general toda clase de datos e información electrónica, escrita o verbal, generada antes, durante y después de la firma de este Acuerdo, será considerada como propiedad intelectual de la Universidad y por tanto, es INFORMACIÓN CONFIDENCIAL que debe ser preservada y custodiada.

II.- El Practicante o Tesista declara que:

- a) Existe una relación de carácter colaborativo con la Universidad, según cartas de intención o convenios de prácticas o proyectos de titulación, debidamente legalizados;

- b) Para desempeñar las funciones dentro de sus prácticas o para la ejecución del proyecto de titulación, tendrá acceso a información privilegiada, la cual acepta guardar con escrupulosa confidencialidad.

## 7. Ognb

1859 En virtud de lo anterior, ambas partes se someten a las disposiciones siguientes:

## 8. CLÁUSULAS

PRIMERA. Ambas partes aceptan que la información señalada en la declaración 1-b), es propiedad de la Universidad y de UTI, la misma será considerada como INFORMACIÓN CONFIDENCIAL, por lo tanto, el Practicante o Tesista se obliga a custodiarla, conservarla y a no divulgarla a terceros, ya sea en forma verbal, escrita, por medios electrónicos, magnéticos, o por cualquier otro medio, directa o indirectamente.

La obligación asumida por el Pasante o Tesista mediante el presente acuerdo, permanecerá durante la vigencia del período de sus Pasantías o hasta la culminación de su proyecto de titulación, extendiéndose por tiempo indefinido luego de finalizada su vinculación colaborativa, indistintamente de las funciones que haya ocupado, dentro de la UTI.

SEGUNDA. La Universidad entregará al Practicante o Tesista los implementos de trabajo necesarios para cumplir con sus objetivos, así como las credenciales de acceso a los diferentes sistemas y/o aplicativos que requiera de acuerdo a la naturaleza de sus actividades. El nombre de usuario que se le asigne quedará registrado en todas las operaciones que realice en los sistemas y/o aplicativos a los que ingrese.

El usuario y contraseña serán remitidos al Practicante o Tesista vía correo electrónico. El cambio de contraseña, la administración y mantenimiento de las credenciales de acceso se realizará de acuerdo a las políticas y procedimientos que en materia de seguridad de la información establezca la Universidad.

TERCERA. El objetivo principal del presente Acuerdo es proteger toda información de índole financiera, comercial, técnica, laboral, académica que tenga carácter confidencial, y que se relacione con productos, servicios, procesos, proyectos, sistemas de información, nuevas tecnologías, talento humano, planificación estratégica y operativa, clientes de la Universidad.

Por tanto, las partes se comprometen a aplicar las medidas de seguridad estipuladas en la normativa interna para evitar la divulgación, reproducción, fuga o uso no autorizado de información confidencial o patentada; y, a custodiar la información en lugares de acceso limitado únicamente a personas autorizadas.

CUARTA. El Practicante o Tesista reconoce y acepta que el incumplimiento de las obligaciones contraídas en el presente Acuerdo implicará asumir las sanciones establecidas en Reglamento

Interno de la Universidad, sin perjuicio de las acciones civiles o penales que la Universidad pudiera tomar en su contra.



1859 QUINTA. Este Acuerdo deberá ser legalizado y archivado por el Director de la UT I. Una copia del Acuerdo se entregará al Practicante o Tesista y otra al Responsable de Seguridad de la Información.

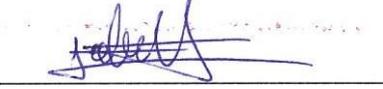
SEXTA.- El presente Acuerdo no aplicará en los siguientes casos:

- a) Por el consentimiento previo y escrito de la Universidad o de la UT I;
- b) Cuando la información confidencial haya pasado a dominio público por razones distintas al incumplimiento de las obligaciones constantes en el presente Acuerdo;
- c) Cuando exista requerimiento de autoridad competente que obligue al Practicante o Tesista a entregar la información que se encuentra a su cargo, y previo conocimiento y autorización del Director de la UT I.

SÉPTIMA. Si alguna de las estipulaciones del presente documento llegare a ser ilegal, inválida o sin vigencia, debido a modificaciones a la legislación ecuatoriana, dicha cláusula deberá excluirse, y este Acuerdo, en el alcance de lo posible y sin destruir su propósito, será ejecutado como si dicha estipulación, no hubiera hecho parte del mismo. Las restantes disposiciones aquí contenidas deberán conservar el mismo valor y efecto, sin afectación directa o indirecta, por la disposición ilegal, inválida o sin vigencia.

LAS PARTES han determinado la importancia de mantener la integridad, disponibilidad y confidencialidad de la información propiedad de la Universidad

Nacional de Loja; han leído y comprendido las estipulaciones de este Acuerdo; y, se comprometen a cumplir los términos y condiciones del mismo, para lo cual lo suscriben en Loja, a los del mes de del año 201 .

	
No. _____	Ing. Jhon A. Calderón Sanmartín
Jho _____	DIRECTOR T.I.
CI: _____	
1150841946	
In _____	
n anmartín	
PRACTICANTE ( ) TESISTA ( X )	DIRECTO

072-547252 Ext. 125  
Cluclad universitaria "Guillermo Falconi Espinosa",

Caf. àlla "S". Sector Argelia

**Anexo 14.** Certificado de traducción de resumen



**UNL**

Universidad  
Nacional  
de Loja

Loja, 12 de agosto de 2024

Lic. Pedro Geovanny Calva Jiménez

**LICENCIADO EN PEDAGOGÍA DEL IDIOMA INGLÉS**

**CERTIFICO:**

Que el resumen del Trabajo de Integración Curricular cuyo título es: **Trazabilidad mediante Blockchain en el sistema de ventas de obras de arte de la carrera de artes plásticas**, del aspirante **Alexander Adrián Vicente Jiménez**, con cédula de identidad Nro. **1150841946**, de la Carrera de Computación de la Universidad Nacional de Loja, ha sido traducido al inglés y cumple con las características propias del idioma extranjero.

Lo certifico en honor a la verdad y autorizo hacer uso del presente en lo que a sus intereses convenga.

Lic. Pedro Geovanny Calva Jiménez

**1150428496**

Nro. Reg. Senecyt: 1031-2022-2421774

**9. LICENCIADO EN PEDAGOGÍA DEL IDIOMA INGLÈS**