PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ

FACULTAD DE CIENCIAS E INGENIERÍA

Sistema de voto electrónico basado en blockchain

Tesis Para optar por el Título de Ingeniero Informático que presenta el bachiller:

SEBASTIAN ANDRES SANCHEZ HERRERA

20143071

Asesor:

Dr. LUIS ALBERTO FLORES GARCIA

Lima, agosto de 2021

Dedicatoria

A Dios, por cuidar y guiarme en mi cada momento de mi vida.

A mis padres Sebastián Sanchez y Nancy Herrera, y a mi abuela Nancy Paico; por brindarme su apoyo incondicional en mi etapa universitaria.

A mi asesor Luis Flores, por guiarme en mi formación profesional.

Resumen

En todo proceso electoral, la seguridad de las elecciones es un factor que siempre se trata de proteger y aún más cuando hay tecnología de por medio. En tal sentido, los sistemas de voto electrónico se vienen utilizando desde el siglo XIX para automatizar algún proceso interno en cualquier proceso electoral. Actualmente, existen diversos sistemas de voto electrónico que han traído grandes beneficios a los procesos electorales, entre ellos ahorro ecológico, eficiencia en el conteo de votos y accesibilidad a los electores que se encuentran en el extranjero.

Sin embargo, a la par del crecimiento de la inserción de la tecnología dentro de los procesos electorales también ha crecido las vulnerabilidades y ataques informáticos a dichos sistemas. Dichas vulnerabilidades se han reflejado en el bajo nivel de seguridad que poseen los sistemas de voto electrónico, las cuales este proyecto las ha agrupado en 3 categorías: la información centralizada y no accesible para los actores en cada fase del proceso electoral que abarca un sistema de voto electrónico, la ausencia de mecanismos que permiten la verificación de la integridad de los datos y la falta de cumplimiento de estándares legales y técnicos en el desarrollo de un sistema de voto electrónico.

Para poder afrontar dichas deficiencias se presenta la tecnología *blockchain* y los contratos inteligentes, los cuales serán las principales herramientas que por su estructura descentralizada e inmutable permiten proponer una solución.

Este proyecto propone el análisis, diseño e implementación de un sistema de voto electrónico para procesos electorales bajo estándares legales y técnicos que brinden transparencia y robustez en las fases de preparación, registro, votación, emisión de voto, escrutinio y auditoría aplicado a las elecciones generales en el Perú.

Tabla de Contenido

Índice de Tablas

Índice de Figuras

Capítulo 1. Generalidades

1.1. Problemática

1.1.1 Descripción

El voto electrónico es un concepto que se viene utilizando desde el siglo XIX. Esto empezó desde la automatización de recuento con el uso de tarjetas perforadas hasta los sistemas actuales con escaneo óptico (Rial, 2004). Es preciso mencionar que la tecnología en el proceso de votación electrónica se ha subdivido en dos modalidades: el voto electrónico presencial y el voto electrónico remoto o no presencial. Mientras que la modalidad presencial requiere que el elector se acerque a un local de votación, la modalidad remota utiliza los beneficios del internet y la telemática para realizar el proceso de votación en un entorno no controlado que el elector decida conveniente (Oficina Nacional de Procesos Electorales [ONPE], 2017).

El uso del voto electrónico ha traído consigo grandes beneficios respecto al sistema tradicional en el proceso electoral como la reducción de costos que trae el ahorro ecológico y mayor eficiencia en el conteo de votos (Panizo, 2007). Otros beneficios del uso del voto electrónico se vinculan a la accesibilidad de los electores que se encuentran en el extranjero o que tienen difícil acceso a los colegios electorales lo que conlleva a la posibilidad del aumento de la participación de la población en el proceso electoral (Goodman, 2017).

Debido a los grandes beneficios que trae consigo el voto electrónico, países como la India, Venezuela y Brasil han optado por el uso de las MVE en todo su sistema electoral con el fin de una mejora a la organización y administración de las elecciones en la totalidad de su territorio (Rijo, 2020). Otros países han realizado ensayos y pruebas piloto y han decidido optar por la inserción del voto electrónico dentro de alguna actividad del proceso electoral ya sea en elecciones municipales o estatales. Estados Unidos insertó el voto electrónico remoto por internet en estados como Florida en 2008 y Virginia en 2010. Por otro lado, Suiza implantó el

voto electrónico remoto desde el 2005 y en sus últimas elecciones ha sido utilizado por casi la mitad de los electores (Europa Press, 2019).

Sin embargo, la votación electrónica presenta desventajas que se refleja con el tema de la inseguridad y desconfianza por la seguridad del voto realizado (ONPE, 2013, p.110). El presente proyecto de tesis evidencia las posibles causas del bajo nivel de seguridad que poseen los sistemas de voto electrónico, en especial los sistemas de voto electrónico remoto para procesos electorales, y las ha agrupado en tres categorías: la información centralizada y no accesible para los actores en cada fase del proceso electoral que abarca un sistema de voto electrónico, la ausencia de mecanismos que permiten la verificación de la integridad de los datos y la falta de cumplimiento de estándares legales y técnicos en el desarrollo de un sistema de voto electrónico.

En primer lugar, no existe un sistema de votación electrónica de código abierto que gestione la información descentralizada y sea accesible para los actores en cada fase del proceso electoral que abarca un sistema de voto electrónico y que no muestre resultados parciales. En la actualidad, la mayoría de las soluciones de votación electrónica son sistemas centralizados en las cuales el administrador del sistema posee total acceso a la información y es la única entidad encargada de la gestión de los datos (Navarrete et al., 2019). Esto genera que los sistemas de votación electrónica carezcan de transparencia hacia el elector, es decir, el elector no tiene acceso a la información necesaria para la verificación de que su voto ha sido contado, por lo que solo tiene que limitarse a confiar en la auditoría post-elección (Matile et al., 2019). El caso más citado para este problema es el de Alemania. El 2009, el Tribunal Constitucional declaró inconstitucional el uso del voto electrónico puesto que el software utilizado no era accesible ni controlado por el público elector. Además, sostuvo que, de implementarse estos tipos de sistemas, es recomendable el respaldo del papel para el elector. Además, el sistema de votación electrónica no solo debe ser accesible por todos los actores, sino entendida, no a

detalle, pero sí lo necesario para que el elector pueda tener plena confianza en el sistema (Aguerre, 2017).

En segundo lugar, hay gran ausencia de mecanismos que permitan la verificación de la integridad de los datos durante un proceso electoral. Usualmente los mecanismos criptográficos de datos para sistemas de votación electrónica no suelen ser robustos, esto es, por el hecho de presentar técnicas de cifrado muy deficientes (Nardi y Maenza, 2017). En el año 2008, Finlandia utilizó el voto electrónico como prueba piloto en elecciones municipales sin embargo el 2009 la Suprema Corte anuló dichas votaciones. Esto se debe a que la auditoría realizada por el ministerio de Justicia sostuvo que el sistema no poseía las cualidades para ser auditado y los datos registrados tenían riesgo potencial de ser intervenidos por software malicioso (Vaha, 2009). Por otro lado, en las elecciones presidenciales del 2012 en Venezuela, el intruso informático llamado HACK521, demostró que los sistemas de votación eran sensibles a ataques troyanos y a la manipulación de los datos (Toalombo, 2016, p. 33). Así mismo, en el año 2019 el sistema de voto electrónico ruso basado en blockchain fue comprometido debido a que el esquema de cifrado era débil. Esto se debió a que la longitud de claves era muy menor a 256 bits (Aguilar, 2019).

Por último, algunos sistemas de voto electrónico carecen de aplicación de estándares legales y técnicos en las fases de emisión, escrutinio y auditoría. Irlanda utilizó el voto electrónico desde el año 2000 y se consideró que tenía un buen funcionamiento debido a que sus estándares para medir su efectividad tenían un grado de aceptación alto por quienes la usaban, sin embargo, años después se filtró información sobre las vulnerabilidades que esta presentó en el conteo de votos (Feldman et al., 2006). Pese a ello, el gobierno irlandés invirtió para el mantenimiento del sistema, pero en el 2009 el ministro Gormley anunció la renuncia al voto electrónico, debido al gran costo de mantenimiento que implicó y que todavía implicaría (Aguerre, 2017). Respecto a estándares legales se refiere, Holanda renunció al voto electrónico

el 2008 debido a que se consideró que el sistema de voto electrónico utilizado poseía irregularidades que lo volvían ilegal. Además, en el informe de la Comisión Consultiva del Proceso Electoral "*Voting with confidence*" (2007) se estableció que todo proceso electoral debe verificar la transparencia, verificabilidad, equidad, elegibilidad para votar, sufragio libre, y accesibilidad. Los sistemas de voto electrónico deben seguir estándares técnicos que aseguren una arquitectura robusta. Estonia fue el primer país en insertar el voto electrónico por Internet en el año 2005. En el año 2014, el gobierno de Estonia designó una comisión para analizar su sistema de votación la cual concluyó que la arquitectura del sistema era adecuada para 10 años atrás, pero que en la actualidad era obsoleta (Springal et al., 2014).

En base a las causas mencionadas en párrafos anteriores, el presente proyecto de fin de carrera plantea proponer una alternativa para la siguiente problemática: "El bajo nivel de seguridad de los sistemas de voto electrónico remoto para procesos electorales". Adicionalmente se ha podido identificar que dicha problemática trae como consecuencia niveles de insatisfacción perjudicando la confianza de los electores sobre los sistemas de voto electrónico (Moura, 2017). Además, estos sistemas vulnerables pueden ser intervenidos y alterados durante y post- proceso de emisión de votos por ciberataques. Esto se debe a que las vulnerabilidades de los sistemas de votación electrónico incrementan la escalabilidad de la amenaza tanto interna como externa al sistema (Montes et al., 2016). Por lo tanto, se concluye que dicha problemática genera el rechazo al sistema de voto electrónico, posibles sistemas corruptos, vulnerables a ataques informáticos internos, externos, así como la desconfianza de los electores y altos gastos en mantenimiento para la continuidad de su uso (Lauer, 2014).

1.1.2 Problema seleccionado

En base a lo descrito en la anterior subsección se puede concluir que la información centralizada y no accesible para todos los actores en todas las fases del proceso electoral que

abarca un sistema de voto electrónico, la ausencia de mecanismos que verifiquen la integridad de los datos y la ausencia del cumplimiento de estándares legales y técnicos en el desarrollo de un sistema de voto electrónico son las causas del bajo nivel de seguridad en las fases de votación, emisión, escrutinio y auditoría de los sistemas de votación electrónica para un proceso electoral, lo cual es la problemática que el presente proyecto de fin de carrera propone solucionar.

1.2 Objetivos

1.2.1 Objetivo general

Respecto al problema general identificado, el presente proyecto de fin de carrera tiene como objetivo general el análisis, diseño e implementación de un sistema de voto electrónico para procesos electorales bajo estándares legales y técnicos que brinden transparencia y robustez en las fases de preparación, registro, votación, emisión de voto, escrutinio y auditoría. Cabe mencionar que el sistema a desarrollar es una alternativa de voto electrónico y la unidad de estudio sobre la cual se aplicará este objetivo será las elecciones generales en el Perú.

1.2.2 Objetivos específicos

O 1. Implementar un sistema de voto electrónico de código abierto que gestione la información del proceso electoral de forma descentralizada para los actores del proceso electoral.

O 2. Implementar un algoritmo de cifrado que provea la integridad de los datos.

O 3. Utilizar estándares legales y técnicos en las fases de emisión, escrutinio y auditoría.

1.2.3 Resultados esperados

O 1. Implementar un sistema de voto electrónico de código abierto que gestione la información del proceso electoral de forma descentralizada para los actores del proceso electoral.

R1. Implementación de una aplicación web mediante un contrato inteligente para elecciones generales en el Perú.

R2. Implementación de un módulo de emisión de voto con el uso de una red de blockchain.

R3. Implementación de un módulo de escrutinio de votos en una blockchain.

R4. Creación de un repositorio con el código fuente del software de acceso libre para la comunidad.

O 2. Implementar un algoritmo de cifrado que provea la integridad de los datos.

R5. Implementación de un módulo de cifrado que resguarde la integridad de los datos mediante cifrado probabilístico asimétrico parcialmente homomórfico El Gamal.

O 3. Utilizar estándares legales y técnicos en las fases de emisión, escrutinio y auditoría.

R6. Catálogo de requerimientos basados en los estándares de los sistemas de votación electrónica propuesta por el Consejo Europeo y la ley Orgánica de elecciones N° 26 859.

R7. Implementación de funcionalidades que permitan auditar al sistema.

R8. Implementación de un módulo que permita la verificación individual de los votos.

1.2.4 Mapeo de objetivos, resultados y verificación

Objetivo: Implementar un sistema de voto electrónico de código abierto que gestione la información del proceso electoral de forma descentralizada para los actores del proceso electoral.		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**
R1. Implementación de una aplicación web mediante un contrato inteligente para elecciones generales en el Perú.	- Código fuente del sistema elaborado.	-Pruebas unitarias y de integración realizadas por el tesista al 100% de aprobación al concluir el proyecto.
R2. Implementación de un módulo de emisión de voto con el uso de una red de blockchain.	- Código fuente del software elaborado. -Comandos del terminal de blockchain Ethereum.	-Pruebas de trazabilidad de los nodos de la blockchain al 100% de aprobación. -Pruebas de integración al 100% de aprobación en la segunda iteración.
R3. Implementación de un módulo de escrutinio de votos en una blockchain.	- Código fuente del software elaborado -Comandos del terminal de blockchain Ethereum.	-Pruebas de integración al 100% de aprobación después de implementación del módulo en la tercera iteración.
R4. Creación de un repositorio con el código fuente del software de	-Archivos con el código fuente en versión digital.	-Pruebas de acceso al repositorio y existencia de un archivo manual y licencia dentro del mismo para poder utilizar el software.

acceso libre para la comunidad.		

Objetivo: Implementar un algoritmo de cifrado que provea la integridad de los datos		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**
R5. Implementación de un módulo que resguarde la integridad de los datos mediante cifrado probabilístico asimétrico parcialmente homomórfico.	-Código fuente -Software elaborado	-Pruebas de funcionalidad del algoritmo al 100% de aprobación después de implementación del módulo en la segunda iteración.

Objetivo: Utilizar estándares legales y técnicos en las fases de emisión, escrutinio y auditoría.		
Resultado	**Medio de verificación**	**Indicador objetivamente verificable**
R6. Catálogo de requerimientos basados en los estándares del Consejo Europeo y la ley Orgánica de elecciones N° 26 859.	-Documento de catálogo de requerimientos validado.	-Validación por un experto al 100% de aprobación en la primera iteración.
R7. Implementación de un módulo que permita la	- Código fuente del software elaborado.	-Pruebas de integración al 100% de aprobación al segundo sprint

verificación individual de los votos.		-Pruebas de acceso de información al 100% de aprobación en la segunda iteración.
R8. Implementación de funcionalidades que permitan auditar al sistema.	- Código fuente del software elaborado. - Comandos del terminal de blockchain de Ethereum	- Pruebas de trazabilidad de los nodos de la blockchain al 100% de aprobación en la tercera iteración.

1.3 Métodos y Procedimientos

En la presente sección se brindará el detalle de cada herramienta, método y procedimiento que se usó para cumplir con los objetivos específicos.

i. Resumen

Resultados esperados	**Herramientas**	**Métodos**
R1: Implementación de una aplicación web mediante un contrato inteligente para elecciones generales en el Perú.	- Vista Front-end: NodeJs, React JS, JavaScript, Html, Bootstrap - Interacción con el blockchain: Ethereum, Solidity, Truffle, Ganache, Metamask - Versiones: GitLab, Drive - Diagramas: Balsamiq, Diagramas.net	PMBOK Guide Metodología Iterativa UML
R2: Implementación de un módulo de emisión de voto	- Vista Front-end: NodeJs, React JS, JavaScript, Html, Bootstrap	PMBOK Guide Metodología Iterativa

con el uso de una red de blockchain	- Interacción con el blockchain: Ethereum, Solidity, Truffle, Ganache, Metamask - Versiones: GitLab, Drive - Diagramas: Balsamiq, Diagramas.net	UML
R3: Implementación de un módulo de escrutinio de votos en una blockchain	- Vista Front-end: NodeJs, React JS, JavaScript, Html, Bootstrap - Interacción con el blockchain: Ethereum, Solidity, Truffle, Ganache, Metamask - Versiones: GitLab, Drive - Diagramas: Balsamiq, Diagramas.net	PMBOK Guide Metodología Iterativa UML
R4: Accesibilidad a una copia de los archivos fuentes del sistema a través de un repositorio de código.	- Versiones: GitLab	
R5: Implementación de un módulo que resguarde la integridad de los datos mediante cifrado	- Interacción con el blockchain: Ethereum, Solidity, Truffle, Ganache, Metamask - Versiones: GitLab, Drive - Diagramas: Diagramas.net	PMBOK Guide Metodología Iterativa

probabilístico asimétrico parcialmente homomórfico.		
R6: Catálogo de requerimientos basados en los estándares del Consejo Europeo	- Versiones: Drive - Estándar: Documento de recomendación del comité de ministros del Consejo de Europa a los estados miembros sobre los estándares legales, procedimentales y técnicos de los sistemas de votación electrónica.	PMBOK Guide Metodología Iterativa
R7: Implementación de funcionalidades que permitan auditar al sistema.	- Vista Front-end: NodeJs, React JS, JavaScript, Html, Bootstrap - Interacción con el blockchain: Ethereum, Solidity, Truffle, Ganache, Metamask - Versiones: GitLab, Drive - Diagramas: Balsamiq, Diagramas.net	PMBOK Guide UML Metodología Iterativa
R8: Implementación de un módulo que permita la verificación individual de los votos.	- Vista Front-end: NodeJs, React JS, JavaScript, Html, Bootstrap - Interacción con el blockchain: Ethereum, Solidity, Truffle, Ganache, Metamask	PMBOK Guide UML Metodología Iterativa

	- Versiones: GitLab, Drive - Diagramas: Diagramas.net	

ii. Herramientas

A continuación, se explica en más detalle las herramientas que se usaron para obtener los resultados esperados para el presente proyecto.

- **NodeJs**

  NodeJs es un entorno de ejecución de JavaScript orientado a eventos asíncronos que son ejecutados en el lado del servidor. El objetivo de NodeJs es desarrollar aplicaciones escalables (Nodejs, s.f.).

- **JavaScript**

  JavaScript es un lenguaje de programación multiparadigma con la finalidad de implementar funcionalidades a páginas web (JavaScript.info, 2020). Los programas en este lenguaje se llaman scripts y estos se pueden escribir directamente en el HTML de una página web y ejecutarse automáticamente a medida que se carga en la página (JavaScript.info, 2020). Los beneficios que posee JavaScript es la integración completa con HTML y CSS, y además brinda soporte para los principales navegadores (JavaScript.info, 2020). Se ha elegido usar este lenguaje debido a que es de código abierto y hay varias comunidades que ayudan a solucionar las diferentes dudas o errores que pueden ir apareciendo.

- **React JS**

  React es una biblioteca de JavaScript para la creación de interfaces de usuario. En el presente proyecto se usará React para el diseño del *front-end* puesto que es una herramienta que brinda facilidades para la creación de componentes HTML reutilizables de manera eficiente (React, s.f.).

- **Blockchain**

La tecnología blockchain se define como una plataforma distribuida punto a punto (P2P) para transacciones transparentes sin necesidad de un intermediario confiable, la cual almacena transacciones (Dogo et al., 2018).

Según Hanifatunnisa & Rahardjo. (2017) la tecnología blockchain posee las siguientes características:

- Es descentralizada, toda la grabación de los datos está disponible para todos los nodos. Por ende, no hay necesidad de autoridades centrales.
- Utiliza algoritmos de consenso que son ejecutados por diferentes participantes para la inserción de nuevos datos en los bloques.
- Utiliza criptografía y firmas digitales para demostrar la identidad de un usuario. Es decir, cada transacción se basa en la identidad criptográfica, la cual es teóricamente anónima.
- Posee un mecanismo complejo para evitar la alteración de los registros almacenados.
- Poseen una marca en el tiempo "*time stamp*" para realizar la trazabilidad y verificar la información.
- Verificabilidad e integridad: Cada bloque se verifica y se agrega a la blockchain.

Adicionalmente existen tres tipos de blockchain: Público, privado, híbrida y como servicio (Hanifatunnisa & Rahardjo, 2017).

- Públicos: Cadenas de bloques sin permiso, el cual permite el acceso a cualquier persona y no existen administradores.
- Privados: Cadenas de bloques con permiso, el control lo ejerce una única entidad que se encarga de mantener la cadena.

- Híbrida: Son aquellas que producen grandes transacciones y tienen reglas las cuales filtran el acceso para entidades específicas.

En tal sentido, en el presente proyecto se ha visto a bien usar una blockchain pública para mostrar la mayor transparencia en cada fase del proceso de votación electrónica. Así mismo, se aprovecharán las características que posee la blockchain para cumplir las propiedades que debe tener un sistema de votación electrónico.

- **Ethereum**

Ethereum es una plataforma de código abierto basada en la tecnología de blockchain, la cual permite la ejecución de contratos inteligentes. Ethereum proporciona una máquina virtual descentralizada para la ejecución de scripts utilizando una red internacional de nodos públicos (Ethereum, 2020). Así mismo, para ejecutar transacciones en la red de Ethereum utilizan una unidad denominada “Gas”. Asimismo, el gas tiene un valor monetario en unidades de ether, la cual es la criptomoneda de Ethereum. Ethereum utiliza el concepto de gas debido a la volatilidad del cambio del precio del ether, por lo que el costo del gas para ejecutar una transacción determinada será la misma sin importar el valor actual del ether (Miethereum, s.f.).

- **Contrato inteligente**

Los contratos inteligentes son programas ejecutables que rigen las reglas del comportamiento de una cuenta dentro de Ethereum (Solidity, 2020).

- **Solidity**

Solidity es un lenguaje de programación de alto nivel orientado a objetos con el fin de implementar contratos inteligentes. Adicionalmente se utilizará este lenguaje de programación puesto que es el más usado para la implementación de contratos inteligentes (Solidity, 2020).

- **Truffle**

Truffle es un *framework* de desarrollo completo de código abierto muy utilizado para el desarrollo de contratos inteligentes para Ethereum. Además, cuenta con manuales en diferentes repositorios de Github (Miethereum, s.f.).

- **Ganache**

Ganache forma parte de la suite Truffle suite. Ganache es una blockchain personal para el desarrollo de Dapps (aplicaciones descentralizadas) en Ethereum y Corda, es decir, Ganache permite recrear entornos de blockchain localmente para probar contratos inteligentes (Trufflesuite, s.f.). Se utilizará Ganache como herramienta para probar el proyecto durante todo el ciclo de vida de desarrollo previo al despliegue en la blockchain.

- **Metamask**

Metamask es un plugin que sirve como puente entre Dapps y el navegador web sin comprometer la seguridad. Las Dapps son aplicaciones descentralizadas que se ejecutan sobre blockchain (Miethereum, s.f.). Se eligió usar Metamask puesto que es compatible con el framework Truffle.

- **Visual Studio Code**

El editor de código fuente de Visual Studio Code fue desarrollado por Microsoft (Miethereum, s.f.). Es una herramienta que incluye soporte para depurar proyectos y manejar versiones mediante Git; además posee una extensión para permitir el desarrollo de contratos inteligentes (Miethereum, s.f.).

- **Git**

Git es un software de código abierto de control de versiones usado generalmente para la organización de proyectos entre programadores. Los objetivos de Git son la integridad de datos, soporte de flujos no distribuidos y no lineales. Este sistema permite gestionar las versiones del código dentro de “ramas” las cuales son independientes entre sí (Git, s.f.).

Se ha decidido utilizar Git para el manejo de versiones del proyecto con el fin de realizar un trabajo ordenado. Se utilizará una nueva rama para cada funcionalidad que se desee agregar a la rama principal. Además, antes de agregar una rama auxiliar a la rama principal se realizarán pruebas unitarias para verificar el correcto funcionamiento, y posterior a la agregación de la rama se realizarán pruebas de integración.

- **Diagramas.net**

Diagramas.net es una aplicación gratuita ofrecida por Google Drive la cual permite graficar diagramas de flujo: UML, DER, Organigramas, modelos de procesos de negocio, etc. (Gsuite, s.f.).

iii. Métodos

- **PMBOK GUIDE (Project Management Body of Knowledge)**

PMBOK GUIDE es una guía de buenas prácticas y lineamientos para la gestión, administración y desarrollo de proyectos (PMI, 2016). El PMBOK GUIDE define 5 grupos de procesos en los que incluye 47 procesos estándares para el desarrollo de cualquier proyecto. Estos son:

- **Procesos de inicio:** En esta fase se define el nuevo proyecto y se busca la aprobación para la inicialización de las siguientes fases. En el presente proyecto se buscó la aprobación del profesor del curso, asesor de tesis y del comité de tesis. El resultado de este proceso fue la aprobación del proyecto para iniciar los siguientes procesos.
- **Proceso de planificación:** Fase que define el objetivo, alcance y las limitaciones del proyecto, así como el catálogo de requerimientos que tiene el proyecto. Un resultado de esta fase es el Anexo B, el cual contiene el plan del proyecto.
- **Procesos de ejecución:** En este grupo de procesos se ejecutan las actividades definidas en el proyecto.

- **Procesos de Control y monitorización:** Se definen procesos para la supervisión del desempeño del proyecto.
- **Procesos de Cierre:** Es la última fase del proyecto, y define el cierre del proyecto en su totalidad y el nivel de aceptación respecto al resultado final del proyecto. Para esta fase final se entregará toda la documentación requerida al momento de la sustentación de tesis.

Así mismo, la metodología que se utilizará es Iterativo. Por lo que se han definido 4 iteraciones para la realización del proyecto. Adicionalmente, en cada iteración se entregarán documentos entregados en versiones anteriores en caso estos hayan sido actualizados.

- **UML (Unified Modeling Language)**

UML es un lenguaje de modelado para sistemas de software. UML ayuda a especificar, visualizar y documentar modelos de sistemas de software. Estos modelos pueden ser de estructura y diseño con el fin de alcanzar todos los requisitos. La flexibilidad de UML permite modelar, a su vez, aplicaciones distribuidas que se basan en casi cualquier middleware en el mercado (UML, s.f.). En tal sentido, se utilizará UML para modelar determinados diagramas de estructura y diseño del presente proyecto de Tesis.

Capítulo 2. Marco Legal/Regulatorio/Conceptual/otros

2.1 Introducción

El objetivo del presente capítulo es presentar el marco conceptual y legal que permitirán el mejor entendimiento referente a procesos electorales y por ende a la problemática definida en el presente proyecto. Para una mayor comprensión, se introducirá un ejemplo al final de cada concepto contextualizado a la problemática.

2.2 Desarrollo del marco

i. Proceso electoral

"Se entiende por proceso electoral al conjunto de acciones ordenadas por etapas, previstas en la Constitución y en las leyes electorales, dirigidas por los organismos electorales para la realización de las elecciones y consultas populares" (Jurado Nacional de Elecciones [JNE], s.f., p. 1).

En la figura 1 se presenta los procesos electorales que se realizan en el Perú definidos por la ley Orgánica de elecciones del Perú establecida en la ley N°26859 (2019).

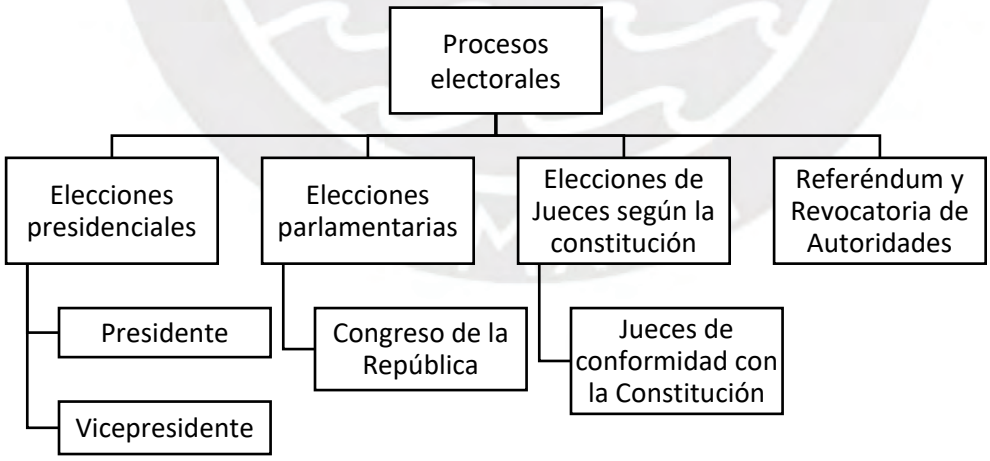


Figura 1. Tipos de procesos electorales en el Perú. (Elaboración propia)

ii. Votar

La Real Academia Española [RAE] define la palabra votar como: “Decir su dictamen en una reunión o cuerpo deliberante, o en una elección de personas.” (2019). Según la Declaración Universal de Derechos Humanos [DUDH], la base de la autoridad del gobierno se expresa mediante elecciones periódicas que se realizan por sufragio universal mediante procedimientos de votación libre (1948, art. 21). Lo anterior se manifiesta en relación directa con el caso peruano, el cual declara que “El voto es personal, libre, igual y secreto.” (Ley N°26859. art. 7).

iii. Cifra repartidora

La cifra repartidora es el procedimiento matemático utilizado para la distribución de escaños de forma proporcional a la votación obtenida cuando en una elección existen más de un participante por partido político. Este procedimiento consiste en dividir el número total de votos obtenidos de cada partido político entre una constante denominada “cifra repartidora” con el fin de obtener el número de vacantes proporcional a los votos obtenidos de cada partido político. Así mismo, este procedimiento se aplica en el Perú desde el año 1963 (JNE, 2005).

iv. Modalidades de votación

La figura 2 evidencia las diferentes modalidades de votación agrupadas por el uso de tecnología.

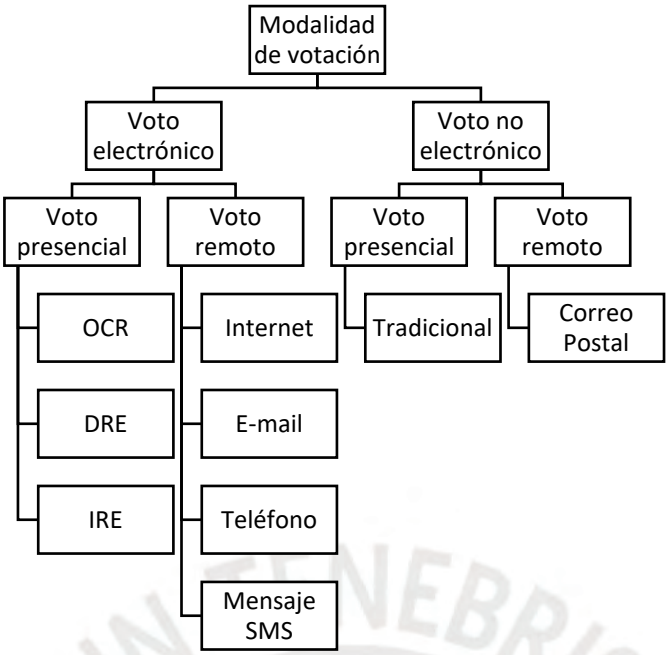


Figura 2. Modalidades de votación. (Elaboración propia)

Puesto que el enfoque del presente proyecto de tesis es el voto electrónico, solo se detalla las características que poseen las diferentes modalidades de este tipo.

- **Voto electrónico**

Según el Instituto Internacional para la Democracia y la Asistencia Electoral [IDEA], se define como voto electrónico a aquellos procesos de votación que insertan el uso de las TIC para el registro, emisión o conteo de los votos (2014). El proceso de voto electrónico se subdivide en tres pasos: creación del voto, resguardo anónimo del voto, conteo de los votos. La etapa de la creación del voto designa que el elector crea el voto al emitirlo. El resguardo anónimo es resguardar la privacidad del voto con otros votos para anonimizar. Y el conteo de votos es la etapa en la cual se dispone el tiempo para realizar el escrutinio de los votos resguardados (Montes et al., 2016).

Voto electrónico presencial

Según la ONPE (2017), los sistemas de votación electrónica presencial son aquellos que permiten automatizar los procesos electorales en ambientes y sistemas administrados por

el gobierno. El uso del voto electrónico ha traído consigo grandes beneficios en los países que lo han implementado, un claro ejemplo es Brasil. “En Brasil el voto electrónico presencial ha favorecido la participación de las personas ciegas al incorporar a la urna electrónica caracteres en Braille” (Presno, 2016, p. 291).

Voto electrónico presencial OCR

En esta modalidad, el voto es realizado manualmente; sin embargo, en la fase de conteo de votos se realiza mediante técnicas de OCR (*Optical Character Recognition*), es decir, utiliza un escáner óptico para leer las boletas marcadas y contar los resultados (*Ace Project*, s.f.).

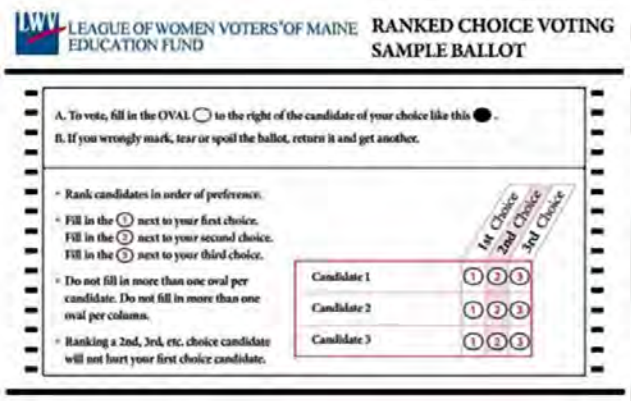


Figura 3. Ejemplo de voto OCR. (Concord Monitor, 2017)

Voto electrónico presencial DRE

Se define como el uso de máquinas de votación DRE (*Direct Recording Electronic*) las cuales realizan la emisión y el conteo de los votos en una sola máquina. Sin embargo, la mayoría de los DRE no proporcionan registros de auditoría en papel (Montes et al., 2016).

Voto electrónico presencial IRE

Esta modalidad realiza el uso de máquinas de votación IRE (*Indirect Recording Electronic*) las cuales separan la emisión del voto de su conteo. En las IRE el elector después de emitir recibe un token o boleta la cual deposita en una urna para su posterior conteo (Jones, 2010).

Figura 4. Ejemplo de máquina DRE implementado con el sistema vot.ar. (Vot.ar, s.f.)

- **Voto electrónico remoto**

Según la ONPE (2017), todo sistema de votación electrónica es remoto cuando permite al elector sufragar mediante el uso del internet basándose en altos estándares de seguridad.

Voto electrónico remoto por internet

El grupo Ad Hoc (2004) definió el voto electrónico remoto por internet como aquel que se produce en un entorno remoto no controlado mediante un dispositivo móvil u ordenador que está conectado a internet. En el año 2012, México aplicó el voto por Internet para ciudadanos residentes en el extranjero (Salvador, 2015).

Voto electrónico remoto por teléfono

Según Tubella y Vilaseca (2005), existe otra alternativa en el voto electrónico el cual se emite el voto vía teléfono a través de tonalidades de marcado o por mensaje de texto.

Voto electrónico remoto por e-mail

Esta modalidad de voto utiliza el correo electrónico como medio de votación, esta modalidad utiliza un cifrado simétrico donde básicamente el elector renuncia al secreto de su voto (Puiggali et al., 2007).

v. Tipos de votos

La figura 5. presenta los tipos de votos que se pueden emitir durante un proceso de votación.

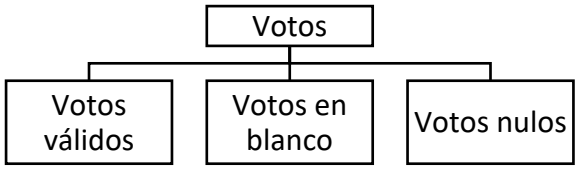


Figura 5. Tipos de voto. (Elaboración propia)

Según la ley N°26859 del gobierno peruano existen tres tipos de votos: los votos válidos, votos nulos, votos en blanco. Los votos válidos son todos aquellos emitidos después de deducir los votos en blanco y nulos. Los votos nulos son aquellos que el elector marcó más de un símbolo o lleven algún identificador del elector en el voto. Por último, los votos en blanco son todos aquellos que no tienen ningún símbolo marcado (2019).

Por otro lado, según la ONPE existen 4 tipos de listas para procesos de votación (s.f.):

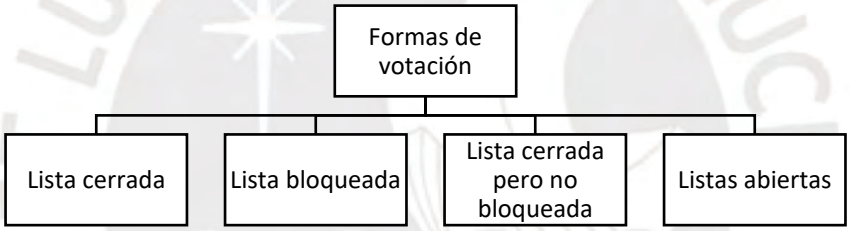


Figura 6. Formas de voto. (Elaboración propia)

Lista cerrada

El elector emite su voto por candidatos que solo están en la lista sin poder introducir candidatos distintos.

Lista bloqueada

El orden predefinido por los partidos políticos no puede ser alterado. En Latinoamérica, esta forma de elecciones se da en Colombia, Costa Rica, Venezuela, Paraguay, entre otros.

Lista cerrada pero no bloqueada

Aquí se introduce el concepto de voto preferencial. El elector emite su voto por una lista del partido, pero a su vez, puede modificar el orden total o parcial asignándole un orden numérico. Este caso se presenta en Panamá, Brasil y Perú.

Lista abierta

Se permite al elector emitir su voto por candidatos de diferentes partidos. Asimismo, el elector puede elegir su orden de preferencia. Este tipo de votación es común en Suiza, Senado español y Luxemburgo.

vi. Fases del proceso electoral en un sistema de voto electrónico remoto

El 2015, la Fundación U.S. Vote define un proceso electoral sobre un sistema de voto electrónico remoto en 6 fases: Preparación, Registro, Votación, Emisión del voto, escrutinio y auditoría (*Vote Foundation*, 2015).

Preparación

Esta fase abarca el diseño de los votos, inscripciones de candidatos, envío de información previa a los electores que lo requieran, formación de las mesas electorales, la logística, el escrutinio y la organización de los recursos disponibles.

Registro

El registro es una fase con múltiples opciones de registro. Estos pueden ser mediante códigos QR asociados (Calvo, 2019), recepción de códigos vía correo postal, correo electrónico (Polys, s.f.), códigos mediante SMS o reconocimiento biométrico (Tivi, 2016).

Votación

La presente fase en la cual el elector ingresa sus preferencias sobre los candidatos a escoger. En esta fase, por lo general, el voto se encripta por una clave pública y privada de la elección (Voatz, 2019).

Emisión del voto

Etapa en la que las autoridades electorales reciben los votos (*Vote Foundation*, 2015).

Escrutinio

Según Matarrita (2012), miembro del Tribunal Supremo Electoral (TSE) de Honduras, el proceso de escrutinio se define como la etapa, del proceso electoral, en la cual los organismos electorales designados se encargan de calificar y cuantificar los votos emitidos, con el fin de regular el resultado de las elecciones. “En el Acta de Escrutinio debe registrarse la siguiente información: a) Número de votos obtenidos por cada lista de candidatos u opción según sea el caso. b) Número de votos nulos. c) Número de votos en blanco. d) Horas en que empezó y concluyó el escrutinio. e) Reclamaciones u observaciones formuladas por los Personeros, así como las resoluciones de la Mesa. Y f) Nombres, números de Documento Nacional de Identificación y firmas de los Miembros de la Mesa y Personeros que deseen suscribirla” (Ley N°26 859. art. 178).

Auditoría

Un proceso de votación es auditable cuando existen procedimientos que puedan verificar el registro del voto del elector y que el proceso es auditable antes, durante y después del proceso de votación (Panizo et al., 2007). En las elecciones del 2005, Alemania implementó un sistema de voto electrónico, sin embargo, el 2009 el Tribunal Constitucional lo declaró inconstitucional ya que el ciudadano promedio no poseía los conocimientos técnicos para que pudiese auditar su voto (Ariel, 2015).

vii. Transparencia en el voto electrónico

“La transparencia es un principio clave para elecciones creíbles. Un proceso electoral transparente es aquel en el que cada paso está abierto al escrutinio de las partes interesadas (partidos políticos, observadores electorales y electores por igual), que pueden verificar independientemente que el proceso se realice de acuerdo con los procedimientos y que no se

hayan producido irregularidades.” (Instituto Nacional de Democracia [NDI], 2013). En tal sentido, para que haya una transparencia en el sistema de votación electrónico, este debe poseer verificación de extremo a extremo.

viii. Verificación E2E

Según la fundación US Vote un sistema de voto electrónico con verificación E2E es aquel sistema en el que durante todo el proceso electoral se produce un resultado que coincide con las intenciones del elector la cual se pueden evidenciar mediante objetivos que poseen tres características (*Vote Foundation,* 2015). Estas tres características son:

Emitido según lo previsto: Es la demanda de garantizar que la emisión del voto utilice comunicaciones seguras y demás mecanismos para garantizar que el *malware* y *hackers* no puedan cambiar el voto.

Registrado como se ha emitido: Es la exigencia de que el propio sistema electoral interprete de forma correcta el voto emitido.

Contado como se ha registrado: Es la demanda al proceso para que el conteo sea preciso.

Respecto al ámbito nacional, “El personero legal ante un Jurado Electoral Especial está facultado para presentar cualquier recurso o impugnación al Jurado correspondiente, en relación con algún acto que ponga en duda la transparencia electoral. Dicha impugnación debe estar debidamente sustentada.” (Ley N°26 859. art. 142).

ix. Estándares de seguridad

Los estándares son aquellos requerimientos que permiten establecer, implementar, operar, monitorear, revisar, mantener y mejorar un sistema informático. La ISO (*International Organization for Standardization*) es una organización encargada de promover el desarrollo de estándares internacionales, como la ISO 27001, la cual expresa normas y estándares sobre los

sistemas de gestión de seguridad de la información (Fiscalización Informática del Voto Electrónico. Guía para la actuación profesional. Argentina. Tinta Libre Ediciones. et al., 2015).

x. Integridad de datos

Según la Asociación de Auditoría y Control de Sistemas de Información [ISACA], la integridad de los datos se define como la precisión, consistencia y validez de los datos almacenados dentro de una base de datos durante todo el ciclo de su vida. Además, la integridad de datos es considerada una medida de calidad debido a que proporciona mecanismos de validación (2011).

Figura 7. La integridad de datos. (Tecnologías-información., s.f.)

Durante un proceso electoral el envío de resultados debe incluir canales de comunicación seguros que puedan validar la integridad y autenticidad de los datos transmitidos (Hernández, 2011). El año 2002, se filtró información del Ministerio del Interior de Irlanda el cual aseguraba que la integridad de los datos del proceso electoral implementado con voto electrónico no estaba garantizada (Aguerre, 2017).

xi. Fraude electoral

“Se define al fraude electoral como el recurso a acciones clandestinas para alterar los resultados electorales (…) tanto los actos descaradamente coercitivos como las irregularidades de la votación tienen un carácter fraudulento porque pueden influir en los resultados de la elección” (Lehoucq, 2007, p. 2-3).

Existen diversos casos de fraude electoral a lo largo de la historia del voto electrónico, uno de ellos es la votación municipal de la provincia de Adelante. “La Fiscalía de Sevilla investigará una denuncia por la presunta compra de votos en Albaida del Aljarafe. Según la denuncia del grupo municipal de Adelante, existen indicios de que en este municipio sevillano se cometió un fraude en el voto por correo, inusualmente alto en comparación con la media provincial y nacional” (Nestor, 2019).

xii. Criptografía

“La Criptografía es una rama de las matemáticas que, al orientarse al mundo de los mensajes digitales, proporciona las herramientas idóneas para solucionar los problemas relacionados con la autenticidad y la confiabilidad” (Mendvil, 1999, p. 1). En el año 2004, la empresa VoteHere patentó un sistema de voto electrónico que empleaba credenciales basadas en criptografía de claves públicas que más tarde fue empleado en votaciones locales en Reino Unido en el 2008 (Hernández, 2008).

Capítulo 3. Estado del Arte

3.1 Introducción

Este capítulo presenta detalladamente cómo se realizó la aplicación de la revisión sistemática basada en la metodología propuesta por Kitchenhan (2007), la cual, expresa que una revisión sistemática es una forma de identificar, evaluar e interpretar una investigación relevante a partir de una pregunta principal. Dicho proceso consta de tres etapas: la planificación, realización y reporte de la revisión.

3.2 Objetivos de revisión

Esta revisión sistemática tiene como objetivo responder la siguiente pregunta de investigación principal: ¿Cómo ha impactado el uso de la tecnología blockchain en el sistema de voto electrónico?

3.3 Preguntas de revisión

Según el objetivo principal definido previamente se han planteado las siguientes preguntas específicas:

- P1. ¿Qué características debería poseer un sistema de voto electrónico?
- P2. ¿Qué aspectos de seguridad debe contemplar un sistema de voto electrónico?
- P3. ¿Cuál es la utilidad de la tecnología blockchain en la votación electrónica?
- P4. ¿Cuáles son los desafíos que se enfrenta al utilizar blockchain?

La lista de los términos usados para resolver las preguntas de investigación es: *e-voting, electronic vote, blockchain, decentralized network, smart contract, voter confidence, secure, security, democracy, features*.

3.4 Estrategia de búsqueda

3.4.1 Motores de búsqueda a usar

Los motores de búsqueda y las bases de datos consultadas fueron IEEE Xplore, Scopus, ProQuest y Google Scholar. Se utilizará dichos motores debido a que están orientados a temas de tecnología e ingeniería. Siendo los tres primeros motores de búsqueda en inglés y Google Scholar como motor de búsqueda para documentos en español e inglés.

3.4.2 Cadenas de búsqueda a usar

En base a la combinación de la lista de términos y el uso de conectores lógicos AND y OR y se obtuvo una lista de cadenas generales básicas de búsqueda (ver Tabla 1).

Tabla 1. Cadenas generales básicas de búsqueda. (Elaboración propia)

Cadenas generales básicas de búsqueda	
C1	*“e-voting" OR "electronic vote" OR "electronic voting"*
C2	*“voter confidence” OR “secure” OR “security” OR “democracy” OR “features”*
C3	*“decentralized network” OR “decentralized technology” OR “smart contract” OR “blockchain”*

En base a las cadenas parciales anteriores se ha generado una cadena resultante para cada par de preguntas.

P1 y P2: “C1 AND C2 AND NOT C3” y

P3 y P4: “C1 AND C2 AND C3”.

Se ha reajustado cada cadena de búsqueda resultante de acuerdo con la sintaxis de cada motor de búsqueda.

Las cadenas de búsqueda para P1 y P2:

ProQuest:

("e-voting" OR "electronic vote" OR "electronic voting") AND ti("e-voting" OR "electronic vote" OR "electronic voting") AND ("voter confidence" OR "secure" OR "security" OR "democracy" OR “features”)

Scopus:

TITLE ("e-voting" OR "electronic vote" OR "electronic voting") AND KEY ("e-voting" OR "electronic vote" OR "electronic voting") AND NOT ("blockchain" OR " smart contract") AND ABS(“features” AND “e-voting”)

IEEE Xplore:

((((("All Metadata":"secure" OR "democracy") AND "Author Keywords":":"e-voting" OR "electronic voting" OR "vote" OR "e-vote") NOT "All Metadata":"bitcoin" or “iot” or "blockchain" or "smart contract") AND "Document Title":"e-voting" OR "e-vote" OR "electronic voting")

Google Scholar:

"voting" or “features” or “secure” -data -mining -iot -blockchain

Las cadenas de búsqueda para P3 y P4:

ProQuest:

ti("blockchain" OR "smart contract") AND ti("e-voting" OR "electronic vote" OR "electronic voting") AND ("decentralized network" OR "decentralized technology" OR "smart contract" OR "blockchain") NOT ("bitcoin" or "iot" or "data mining" or "cryptocurrency")

Scopus:

TITLE ("e-voting" OR "electronic vote" OR "electronic voting" OR "blockchain" OR "smart contract") AND KEY ("e-voting" OR "electronic vote" OR "electronic voting") AND KEY ("blockchain" OR "smart contract")

IEEE Xplore:

(((((("All Metadata":"decentralized network" OR "decentralized technology” OR "secure" OR "democracy") NOT "Author Keywords":"bitcoin" or “iot”) AND "Author Keywords":"e-voting" OR "electronic voting" OR "vote" OR "e-vote") AND "Author

Keywords":"blockchain") AND "Document Title":"e-voting" OR "e-vote" OR "electronic voting" AND "blockchain")

Google Scholar:

"Blockchain" and "voting" or "risk" -data -mining -iot -health or “smart contract”.

3.4.3 Documentos encontrados

Posterior al proceso de búsqueda con la cadena resultante generada, cada base de datos arrojó una lista de documentos (ver Tabla 2).

Tabla 2. Resultados de la búsqueda. (Elaboración propia)

Fuente	**Resultados**	**Criterios de inclusión/ exclusión**	**Repetidos**	**Después de revisar el *abstract***	**Seleccionados**
IEEE Xplore	70	37	11	22	16
Scopus	123	40	9	16	8
Proquest	320	11	9	8	13
Google Scholar	120	32	0	18	8
Total	633	120	29	64	45

3.4.4 Criterios de inclusión/exclusión

Los siguientes criterios se definieron con el fin de identificar documentos que guarden relación directa y actualizada sobre las preguntas de investigación.

1. Los documentos escritos en inglés y español.
2. Los documentos cuya fecha de publicación se encuentre en el rango de temporalidad de 2015-2020.

Los criterios de exclusión que se presentan permiten utilizar documentos con aporte científico confiables.

1. Los documentos que posean menos de 3 páginas debido a que contendrá información escasa o redundante de otros documentos.

2. Los documentos cuyo título no posean relación con el tema de estudio.
3. Los documentos duplicados.
4. Los documentos considerados literatura gris.
5. Los documentos que son redundantes del mismo autor.
6. Los documentos que no hayan sido citados.

Después de realizar los criterios de inclusión y exclusión, de eliminar los documentos repetidos y de revisar el resumen se ha obtenido una lista de 25 documentos primarios, las cuales darán respuesta a cada una de las preguntas de la revisión sistemática (ver Tabla 1.2). A continuación, se presentan los 6 primeros documentos de la lista de estudios primarios, los restantes estarán ubicados en el Anexo A. Se ha utilizado el estándar APA como estándar de referencia bibliográfica.

3.5 Formulario de extracción de datos

El formulario de extracción de datos registró tanto información general como detalles que describen su aporte a las preguntas específicas de la revisión sistemática. Dicho formulario contiene información de la lista de estudios primarios, así como fuentes secundarias encontradas en la referencia de los documentos de la lista de estudios primarios. El formulario se definió con cada campo de la Tabla 3.

Tabla 3. Campos de formulario de extracción. (Elaboración propia)

Campo	**Descripción**	**RQ**
ID	EP [número]. P.ej: F001	General
Título		General
Tipo de documento	Revista, Artículo, *Conference paper* o capítulo de libro	General
Autor(es)		General
Año de publicación		General
Título de la publicación	Nombre de la revista, congreso o libro	General
Motor de búsqueda		

Características del voto electrónico	¿Qué características debería poseer un sistema de voto electrónico?	P1
Aspectos de seguridad	¿Qué aspectos de seguridad debe contemplar un sistema de voto electrónico?	P2
Uso del blockchain	¿Cuál es la utilidad de la tecnología blockchain en la votación electrónica?	P3
Limitaciones del blockchain	¿Cuáles son los desafíos que se enfrenta al utilizar blockchain?	P4
Ámbito de origen geográfico	Origen geográfico sobre el cual se realizó el estudio	General
Comentario	Mención breve de puntos importantes que ayudarán a responder las preguntas de la revisión sistemática	

El formulario de extracción completo con la literatura revisada está ubicado en el siguiente enlace:

https://docs.google.com/spreadsheets/d/14a4GBpcpEkdg568fe3w7amQXAztNwSlEUWS-wcpxL7Y/edit#gid=0

3.6 Resultados de la revisión

3.6.1 Respuesta a pregunta P1

¿Qué características debería poseer un sistema de voto electrónico?

Los sistemas de voto electrónico necesitan satisfacer las siguientes características como mínimo para ser aplicadas:

- El sistema debe garantizar la integridad de los votos (Bulut, 2019).
- El sistema debe permitir que solo los electores emitan su voto (Faou, 2019). Por lo que el sistema no debería registrar votos de individuos externos al proceso de votación (Hjálmarsson et al., 2018).
- El sistema debe estar disponible durante todo el proceso de votación (Faou, 2019). Esto se refiere a que debe soportar que muchos electores emitan su voto de forma simultánea.
- El sistema debe ser justo, es decir, ningún resultado parcial de los votos debe ser publicado hasta que finalice el proceso de votación (Faou, 2019).

- El sistema debe garantizar el anónimo del elector (Faou, 2019). Otros autores, como Miguel Montes (2016), llaman a esta propiedad el “Reaseguro individual”, es decir, el sistema no permite revelar la identidad del elector. Esta característica generará seguridad del elector.
- El sistema debe permitir verificación individual, es decir, cada elector puede verificar que su voto ha sido emitido y contado correctamente (Chaieb et al., 2018).
- El sistema debe permitir la verificación universal. Cualquier interesado en el proceso electoral puede verificar que la cantidad del resultado final corresponde a la suma de votos (Chaieb et al., 2018).
- El sistema debe proporcionar un nivel de robustez alto (Faou, 2019). Es decir, el sistema debe poseer una arquitectura sólida que le permita ser resistente a ataques informáticos durante todo el proceso de votación.
- El sistema debe poseer un nivel de transparencia alto, es decir, debe ser capaz de ser estudiado por diferentes entidades. (Montes et al., 2016).
- El sistema debe permitir la auditoría no electrónica. Esta característica permitirá al elector verificar su voto y auditar de manera aleatoria una urna en comparación con el conteo electrónico. Así mismo, el sistema debe pasar por un proceso de homologación con el fin de que verifique su correcto funcionamiento (Montes et al., 2018).
- El sistema debe poseer protección contra lectura no autorizada. Esto es, poseer una protección hacia los datos contra ataques externos o internos (Montes et al., 2016).
- El sistema debe garantizar que no se permita la coerción. Esto permite que el elector tenga la libertad de emitir su voto sin que nadie lo esté coaccionando (Faou, 2019). Hjálmarsson et al. Afirman que un sistema electoral no debe permitir la votación forzada (2018).

3.6.2 Respuesta a pregunta P2

¿Qué aspectos de seguridad debe contemplar un sistema de voto electrónico?

Respecto a las características mencionadas en la anterior pregunta, se puede identificar los siguientes aspectos de seguridad para un sistema de voto electrónico:

- Los sistemas de voto electrónico deben poseer alto nivel de privacidad, y en especial los sistemas de voto electrónico no presencial, debido a que se desarrollan en un entorno no controlado. Según el autor Yeregui (2018), la privacidad de los sistemas de voto electrónico se categoriza en tres tipos:
    - **Privacidad del voto:** El voto del elector no debe ser revelado a nadie.
    - **Ausencia de recibo:** El sistema de votación electrónica no debe probar por quien se realizó el voto.
    - **Resistencia a la coerción:** Un elector no debería colaborar en un acto de coerción para obtener alguna información de su voto.
- Los sistemas de voto electrónico deben tener alto nivel de poseer resistencia al fraude. Inclusive, este aspecto fue la razón por la cual algunos países como Austria, Bélgica y Estados Unidos migraron de la votación en papeletas a votación electrónica (Faour, 2019).
- El sistema de voto electrónico debe poseer protección de la integridad de los datos. Así mismo, como menciona Gao et al. (2019), desde que se realizan los procesos de votación se ha querido mantener la integridad de los datos, al minimizar la posibilidad que estos puedan ser alterados, es por ello por lo que cada vez se emplean herramientas criptográficas de mayor complejidad computacional.
- El sistema de voto electrónico debe mantener el anonimato la identidad de los electores, pero no la trazabilidad sobre su voto (Bulut et al., 2019). Es decir, se busca tener seguimiento sobre los votos, para ver si han sido contados, si no han sido modificados,

sin revelar la identidad del elector. En este aspecto también son útiles los métodos criptográficos, como por ejemplo el uso de una doble llave (*public-private key*) o protocolos homomórficos (Navarrete, 2019).

- El sistema de voto electrónico solo debe permitir emitir su voto a electores para realizar el derecho a voto (Sheer, 2018). Este aspecto abarca gran importancia, ya que lleva a que las arquitecturas a implementar posean comunicaciones, previas al proceso de emisión de votos, con sistemas del gobierno para validar a los electores sin revelar su identidad.
- Por último, permitir ver el resultado de los votos cuando el proceso electoral haya concluido con el fin de que no influencie a ningún elector es un aspecto que aumenta la confianza del elector (Zhang et al., 2019). Este aspecto va relacionado con la auditoría de los votos después de haber concluido el proceso de emisión de votos con el fin de mostrar transparencia en el proceso electoral (Pérez et al., 2018).

Añadiendo a lo anterior, para que un sistema de voto electrónico se considere seguro debe basarse en estándares que permitan verificar los requisitos necesarios que este debe presentar. El Consejo de Europa (CE) fue la primera organización internacional en elaborar la primera propuesta integral de estándares referente al voto electrónico (CE, 2004). Estos estándares que brindan robustez a un sistema de voto electrónico están divididos en tres tipos: Estándares legales, estándares procedimentales, y estándares técnicos.

1. Estándares legales
    - Sufragio universal
    - Sufragio igualitario
    - Sufragio libre
    - Transparencia
    - Verificación y control

- Fiabilidad y seguridad

2. Estándares procedimentales
    - Convocatoria
    - Electores
    - Candidatos
    - Emisión del voto
    - Resultados Auditoría
3. Estándares técnicos
    - Accesibilidad
    - Interoperabilidad
    - Sistemas operativos
    - Seguridad
    - Auditoría
    - Certificación

3.6.3 Respuesta a pregunta P3

¿Cuál es la utilidad de la tecnología blockchain en la votación electrónica?

La característica más resaltante sobre la tecnología blockchain es que permite brindar una base de datos encriptado y distribuida a los sistemas de votación, lo cual incrementa la seguridad de los sistemas de votación electrónica puesto que es una red descentralizada y permite mayor participación del elector y quitar participación de terceros (Sudharsan et al., 2019). Por otro lado, la tecnología blockchain ha permitido que la información de los votos quede registrada y sea pública para todos los actores que intervienen en este proceso y que además garantiza que una vez registrada la información esta no pueda ser modificada (Faour, 2019).

El enfoque de la tecnología blockchain ha ido evolucionando y en la actualidad el más reciente cambio de la blockchain ha recibido el nombre de “Blockchain 3.0”. Esta nueva versión permite la creación de los famosos “contratos inteligentes” (Dogo et al., 2018). Los contratos inteligentes pueden especificar reglas, las cuales brindan seguridad al sistema de votación electrónica. Existen plataformas que permiten la creación de dichos contratos inteligentes sobre una red blockchain, la más conocida es Ethereum (Dhulavvagol et al., 2020). Por otro lado, cabe mencionar que la aplicación del blockchain en el voto electrónico no solo se ha enfocado a procesos electorales, sino que existen diversas aplicaciones a organizaciones privadas, religiosas y pequeñas comunidades (Khandelwal, 2019).

3.6.4 Respuesta a pregunta P4

¿Cuáles son los desafíos que se enfrenta al utilizar blockchain?

La escalabilidad es un factor que desafía el uso de blockchain. Al querer implementar una solución de voto electrónico con tecnología blockchain, se debe tener en consideración que se va a realizar sobre toda una nación y por ende debe ser capaz de que esta propuesta soporte millones de ingresos de votos, que estarían representadas por cada transacción en la blockchain (Abuidris et al., 2019). Muchos estudios afrontan dicho problema al utilizar una plataforma privada blockchain o híbrida en vez de una red pública con el fin de que la red sea exclusiva para el sistema y genere mayor velocidad de tráfico de datos del que se pudiera en una red pública (Hjálmarsson et al., 2018). Sin embargo, ello afecta la característica de ser una red descentralizada y la convierte a ser parcialmente centralizada (Košt’ál, 2019).

Otro desafío que enfrenta el uso del blockchain es que, si se realiza sobre una red pública ya existente, el costo de implementar dicho sistema dependerá netamente del precio de la criptomoneda que utiliza dicha red pública, en el caso de utilizar la red Ethereum dependerá del ether ; sin embargo, el coste de implementar un contrato inteligente en Ethereum se mide

por una unidad llamada gas la cual equivale a una cantidad fija de ethers para que su coste no se vea afectado por la fluctuación del costo del ether (Lai, 2018).

Un gran desafío que podría enfrentar el uso del blockchain es la computación cuántica, la cual eleva exponencialmente la capacidad para realizar cálculos operativos de un computador. Esto se debe, a que los conceptos de criptografías (encriptación de datos) están basados en modelos matemáticos que requieren de una alta capacidad computacional para poder desencriptarlos. Algunos estudios hacen frente a dicho problema alterando la participación de la criptografía en la seguridad por algoritmos basados en códigos, lo cual minimiza el aporte de la capacidad matemática en la seguridad de los datos (Gao et al., 2019).

Por último, al igual que todo sistema de votación electrónico, el más grande desafío que enfrenta el uso de esta nueva tecnología blockchain es la aceptación del público y la confianza de los electores (Kshetri y Voas, 2018).

3.7 Productos de blockchain en el mercado

Adicionalmente a la revisión sistemática, se presentará una lista de los productos de voto electrónico en el mercado que usan blockchain.

- **Agora**

  Plataforma creada por el Instituto Federal Tecnológico de Suiza el 2015. Esta plataforma basada en blockchain permite la votación remota en línea desde cualquier dispositivo (Agora, 2017). Utilizado por primera vez en Sierra Leona el 2018 (Calvo, 2019).

- **Follow My Vote**

  Plataforma que brinda votación en línea basado en blockchain que permite auditar la urna y muestra el proceso de votación en tiempo real. Utiliza una webcam como parte de las credenciales para iniciar sesión de forma remota (FollowMyVote, s.f.).

- **Polys**

Plataforma basada en los contratos inteligentes que brinda la máquina virtual de Ethereum. Tiene versión gratuita con algunas limitaciones y es flexible para adaptarse a cualquier tipo de votación (Calvo, 2019). Esta plataforma está basada en votación móvil en línea y como método para iniciar sesión utiliza códigos enviados a los correos electrónicos de los electores (Polys, s.f.).

- **Polyas**

  Empresa que utilizó blockchain para ofrecer sistemas de votación electrónica. Polyas ofrece un sistema con un nivel de seguridad, el cual fue certificado por la Oficina Federal Alemana de Información de Seguridad el año 2016. Esta plataforma es utilizada en Estados Unidos. Permite realizar monitoreo del sistema en tiempo real (Polyas, s.f.).

- **Voatz**

  Creada el 2015 después de participar y ganar en SXSW Hackathon. Utilizado el 2018 por el Senado de Virginia Occidental para votación electrónica remota por celular para militares estadounidenses autorizados que vivían en el extranjero (Voatz, s.f.). La plataforma utiliza como credenciales una foto del rostro del elector tomada desde la misma aplicación y una prueba biométrica como la huella digital o la exploración de retina (Calvo, 2019).

- **Coinstack**

  En el año 2016, la empresa Blocko.io creó la plataforma Coinstack basada en la blockchain de Bitcoin. Esta plataforma presenta compatibilidad con los contratos inteligentes de Ethereum. Utilizada por primera vez para elección de proyectos comunitarios en Gyeonggi-do, Corea del Sur (Calvo, 2019).

- **Bobak**

  Fue creada por la empresa emergente británica Monax y se caracteriza por ser un sistema de votación “multijurisdiccional”, es decir, que se puede acceder desde

cualquier parte del mundo. Basada en los contratos inteligentes que provee la red distribuida de Ethereum (Calvo, 2019).

- **Secure vote**

  Plataforma que ofrece voto electrónico basado en Bitcoin desarrollado por la empresa emergente australiana XO.1. Se puede acceder a *Secure Vote* desde cualquier Smartphone o máquinas de votación conectadas a la red (Calvo, 2019).

- **Tivi**

  Sistema diseñado por la empresa Smartmatic. Es una plataforma en línea con autenticación biométrica mediante una foto del rostro. Tivi asegura la elegibilidad y proporciona diferentes técnicas de autenticación mediante cifrados (Marwa Chaieb et al., 2018).

A continuación, se presenta un cuadro comparativo de las propuestas existentes en el mercado detallado en la tabla 4. Sin embargo, no se ha tomado en consideración los productos Bobak y Coinstack, debido a la poca información publicada acerca de esta plataforma. Adicionalmente, aquellas celdas de la tabla 1.4 en donde no se haya podido encontrar información se ha escrito las siglas NI para indicar que no hay información relevante y confiable para determinar dicho campo.

Tabla 4. Cuadro comparativo de los productos existentes que utilizan *blockchain*. (Elaboración propia)

	Estándares técnicos							**Estándares legales**					**Integridad**		**Modalidad**		**Accesibilidad**
	Elegibilidad	Autenticación	Modificar el voto	Robustez	Protección de la coerción	Seguridad	Auditable	Anonimato	Justo	Verificación individual	Verificación universal	Privacidad del voto	Consistencia	Emisión de recibo	Presencial	Remoto en línea	Código abierto
Agora	✓	✓	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	X	X	✓	X
Follow My Vote	✓	✓	✓	X	✓	✓	✓	✓	X	✓	✓	✓	X	X	X	✓	✓
Polys	✓	✓	X	✓	NI	✓	✓	✓	✓	✓	NI	✓	✓	✓	✓	✓	X
Polyas	✓	✓	X	✓	X	✓	✓	✓	✓	✓	✓	✓	✓	X	X	✓	X
Voatz	✓	✓	✓	✓	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	X	✓	X
Secure vote	✓	✓	NI	✓	NI	NI	✓	✓	NI	✓	✓	NI	✓	NI	✓	✓	✓
Tivi	✓	✓	X	NI	X	✓	X	✓	NI	✓	✓	NI	✓	X	X	✓	X

3.8 Conclusiones

Debido a la característica descentralizada de la blockchain y el uso de redes P2P la ha llevado a convertirse en una tecnología que podría tener aplicaciones en varios campos de la ciencia, particularmente nos enfocaremos en su uso en sistemas de votación electrónica. La votación electrónica ha estado en constante lucha con los aspectos de seguridad que esta debe poseer para tener una recepción positiva por parte de los electores. El uso de la blockchain ha permitido afrontar estos aspectos de seguridad de manera favorable debido a su estructura descentralizada y distribuida. Sin embargo, esta tecnología presenta algunos desafíos para llevar su implementación a votos presidenciales. Pese a dichos desafíos existen muchos estudios que han diseñado soluciones de votación electrónica con blockchain enfocadas a atacar desafíos específicos.

Por último, la solución del presente proyecto plantea cubrir los vacíos de seguridad e integridad de los datos que presentan los sistemas actuales de voto electrónico que no utilizan blockchain aplicado a las elecciones generales del Perú. Respecto a las soluciones que usan blockchain, el presente proyecto de tesis aborda la accesibilidad hacia el código, así como, el cumplimento de estándares legales y técnicos presentados por el Consejo Europeo.

Capítulo 4. Implementar un sistema de voto electrónico de código abierto que gestione la información del proceso electoral de forma descentralizada para los actores del proceso electoral

4.1 Introducción

El presente capítulo evidencia el cumplimiento del objetivo 1, el cual es “Implementar un sistema de voto electrónico de código abierto que gestione la información del proceso electoral de forma descentralizada para los actores del proceso electoral”. El cumplimiento de este objetivo se evidencia con la presentación a detalle de los primeros 4 resultados alcanzados. Para ello, se va a describir los métodos, procedimientos, pasos y dificultades que se presentaron para obtenerlos.

El primer resultado alcanzado consiste en la implementación de una aplicación web mediante un contrato inteligente donde se explota la tecnología blockchain, y Solidity para la implementación de Dapps. El segundo, se enfoca en el módulo de emisión de votos, la cual es la fase fundamental en el proceso de votación. El tercero, está enfocado en la siguiente etapa, la cual es el módulo de escrutinio. Como último resultado es brindar el producto final a la comunidad mediante un repositorio público con el de GitLab.

Finalmente, se termina con una sección de discusión acerca de los resultados alcanzados y las características que se deben tener en consideración en caso se quiera utilizar otro público objetivo.

4.2 Resultados alcanzados

R1. Implementación de una aplicación web mediante un contrato inteligente para elecciones generales en el Perú.

a) Descripción

Este resultado alcanzado permite manejar la información del proceso electoral, en específico elecciones generales las cuales incluyen elecciones para presidente,

vicepresidente y congresales, de forma descentralizada gracias al uso de la tecnología blockchain. Así mismo, toda la documentación se encuentra en el Anexo D del presente documento.

b) Métodos y herramientas utilizadas

Para obtener la implementación de una aplicación web, y en general para cualquier software, es de vital importancia la documentación con el fin de tener una visión más clara de lo que se obtiene como resultado final. En tal sentido, se optó por desarrollar tres documentos de diseño: Casos de uso, diagrama de actividades y el diagrama de arquitectura. Así mismo, estos documentos de diseño se realizaron por cada módulo presente en los demás resultados alcanzados.

En primer lugar, los casos de uso permiten entender cómo un actor interactúa con el sistema. En segundo lugar, el diagrama de actividades permite plasmar las actividades y el flujo de cada proceso que se realiza en el sistema. Por último, el diagrama de arquitectura es vital para poder tener claro las tecnologías que se van a utilizar durante del desarrollo del sistema. Así mismo, por el hecho de utilizar una tecnología nueva como lo es blockchain, es importante tener claro cómo va a ser la arquitectura del sistema planteado y todos los componentes que interactúan. En tal sentido, la arquitectura propuesta para el sistema a implementar es la siguiente.

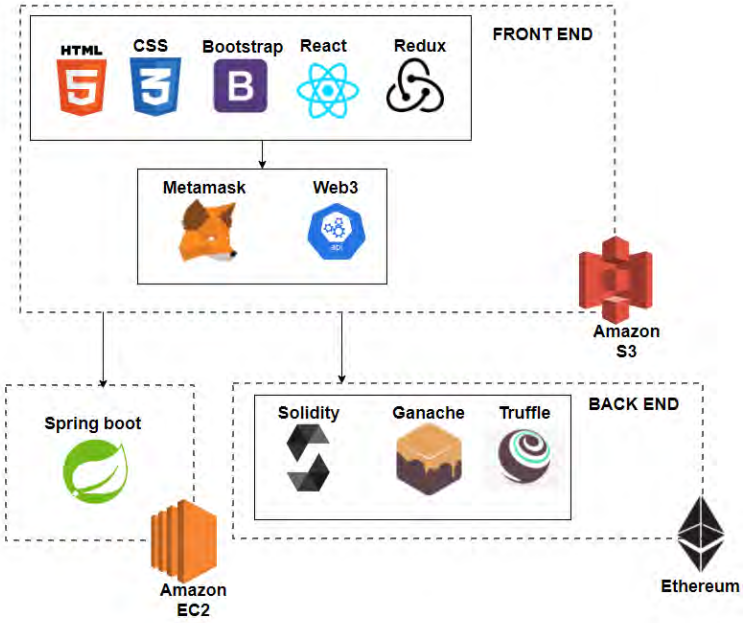


Figura 8. Arquitectura del sistema a implementar. (Elaboración propia)

Como se puede apreciar en la figura 8 la vista del cliente, la cual se identifica como el Front-end utiliza React como *framework* y demás herramientas que se muestra. Así mismo, para la interacción con la *billetera* se está usando Metamask y web3 para acceder al contrato inteligente implementado en Solidity. Adicional a ello, se encuentra el Back-end el cual utiliza Ganache para simular una blockchain local y se utilizó el *framework* de Truffle. Por último, se encuentra el servicio de notificación vía correo electrónico implementado en Spring boot.

Adicionalmente, se desarrolló el prototipo del sistema en la aplicación Balsamiq, ya que esta aplicación permite crear prototipos como bosquejo con el fin de enfocarse en la funcionalidad del sistema y no tanto en el diseño. El prototipo se encuentra en el Anexo D.

A continuación, se presentan algunas imágenes del prototipado acerca de la fase de la creación del proceso electoral.

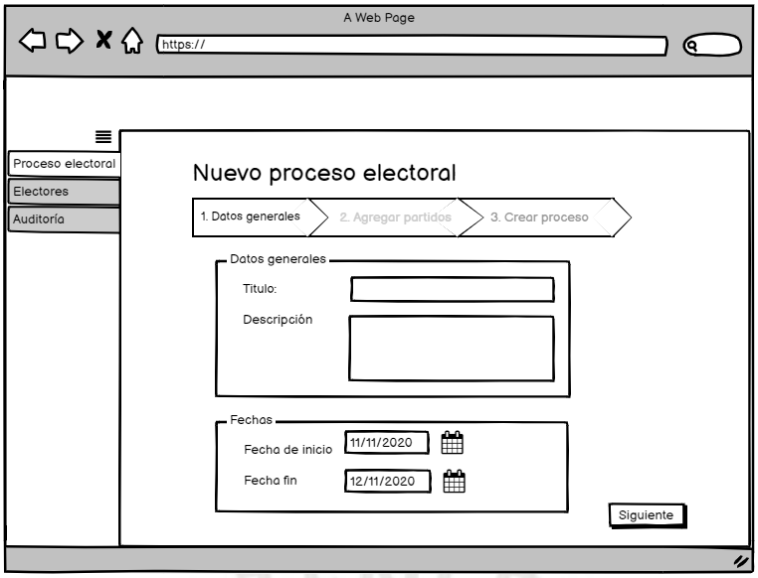


Figura 9. Paso 1 para la creación de un proceso electoral. (Elaboración propia)

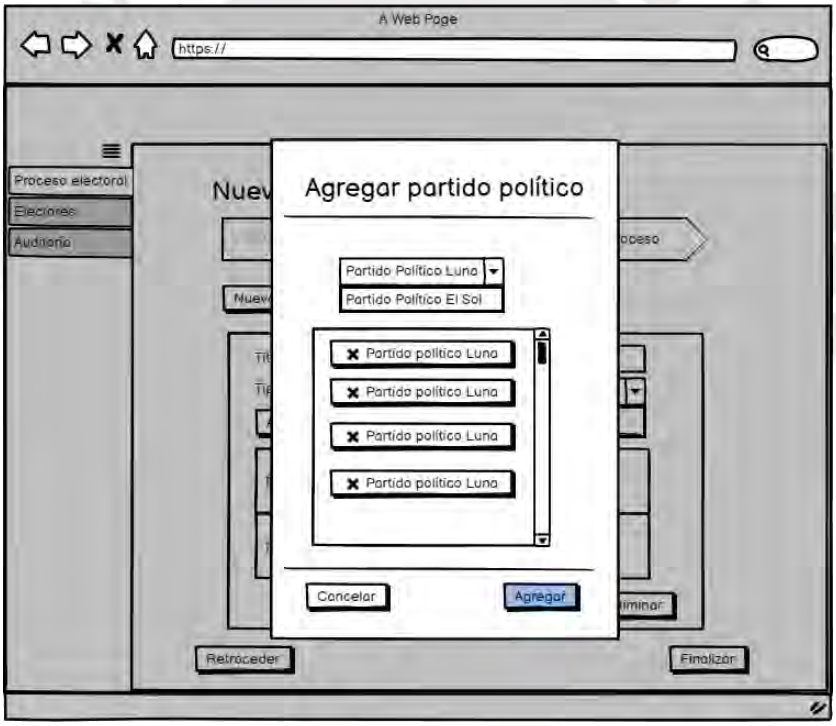


Figura 10. Paso 2 para la creación de un proceso electoral. (Elaboración propia)

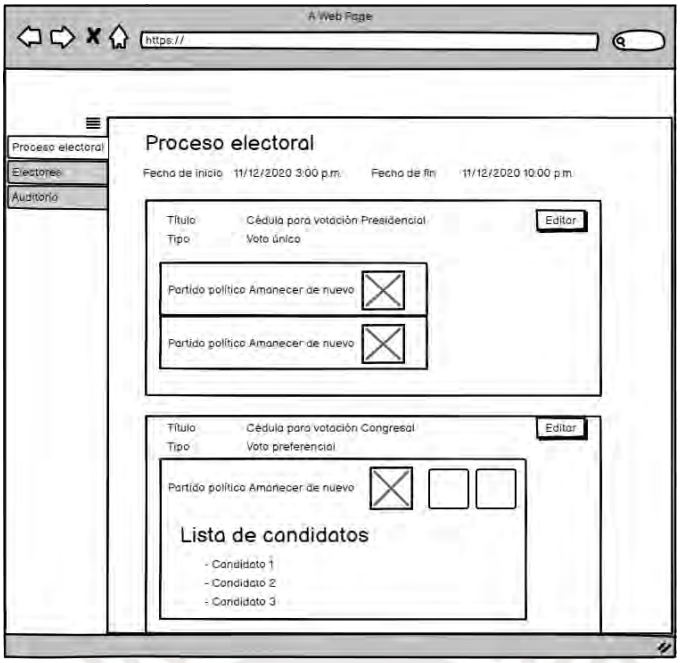


Figura 11. Resumen del proceso electoral creado. (Elaboración propia)

Los obstáculos que se presentaron para este resultado alcanzado fueron que Solidity es una herramienta que no permite el uso de clases sino de Contratos y estructuras por lo que interacción entre la vista web y el contrato inteligente eran mediante servicios con datos primitivos del Solidity, llámese arrays, string y uint (enteros sin signo). Los beneficios que presentó Solidity, fueron que mediante "*modifiers*" permite restringir quien pueda cambiar el estado del contrato inteligente, en este caso, solo el creador del contrato inteligente puede actualizarlo. A continuación, se presentan las funciones que se utilizaron para la creación del proceso electoral y se evidencia como la funciones de "addPoliticalParties" puede ser ejecutados por la clave privada creadora del contrato inteligente mediante el *modifier* onlyOwner, el cual es implementado en otra sección del código:

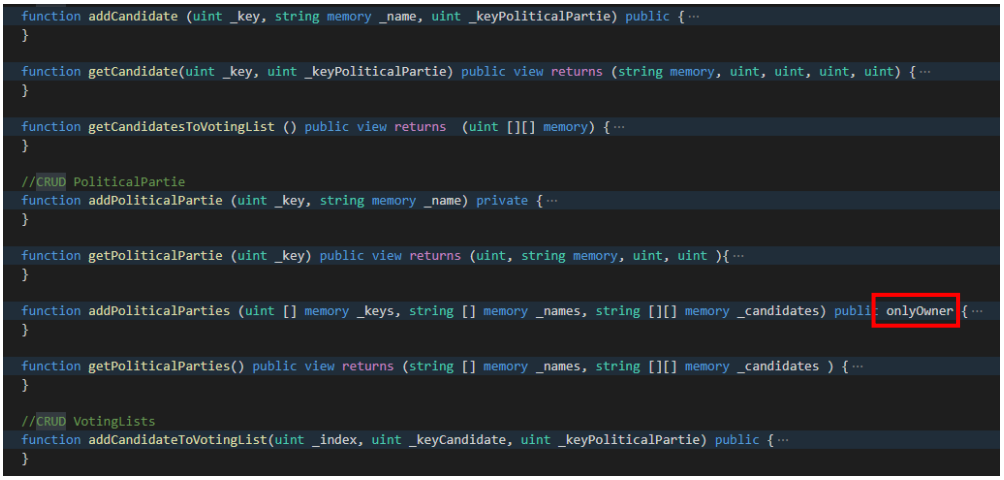


Figura 12. Imagen de algunas funciones implementadas en el contrato inteligente. (Elaboración propia)

Para más detalle de la creación del proceso electoral puede revisar el código fuente del módulo ubicado en el repositorio de GitLab, el cual puede ser accedido mediante el siguiente enlace: https://gitlab.com/srbastian.sash/tesis2-e-voting-system. Así mismo, el sistema maneja el registro de electores y notificación a los votantes sobre el registro y participación del proceso electoral electrónico no presencial creado en el sistema. Este módulo de registro de electores tiene la funcionalidad de registrar las cuentas de los electores a la blockchain, la cual tiene recargada una determinada cantidad de ethers, la moneda electrónica de la red Ethereum, para poder interactuar con el sistema del presente proyecto de tesis. Adicionalmente, cabe mencionar que queda bajo la responsabilidad del elector utilizar dichos ethers para la emisión del voto y no para otra transacción con otras dApps. Para este módulo, hubo dificultades que se presentaron al momento de implementarlo, puesto que en la red de blockchain no se permite enviar correos. En tal sentido, la solución fue implementar un servicio en Spring el cual es desplegado en una EC2 de Amazon para la creación de credenciales y envío de correo con toda la información. Cabe mencionar que el único rol que puede crear otros roles es el administrador, por lo que existe dos interfaces gráficas para la

creación de roles, una para el auditor y otra para el elector las cuales se detallan en las siguientes secciones.

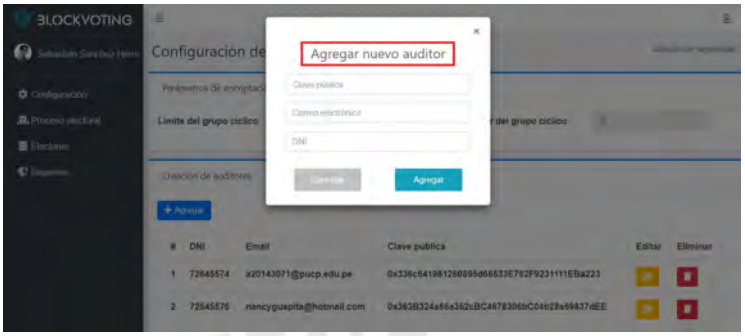


Figura 13. Interfaz gráfica de la creación del rol auditor. (Elaboración propia)

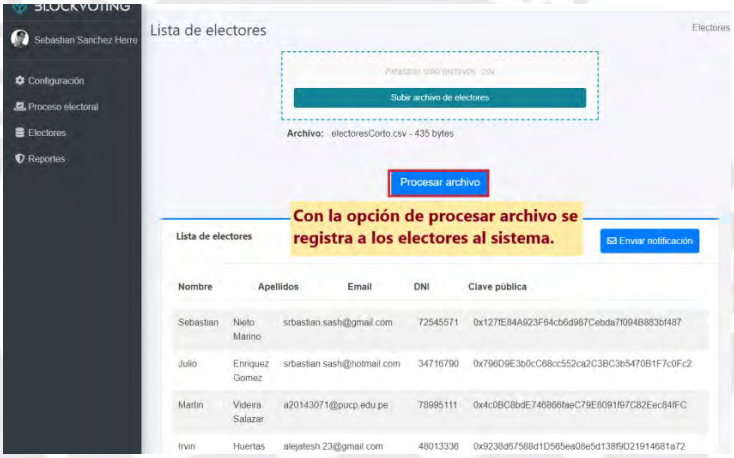


Figura 14. Interfaz gráfica de la creación del rol elector. (Elaboración propia)

Finalmente, el indicador objetivamente verificable para este resultado alcanzado son las pruebas unitarias e integrales del presente módulo. Dichas pruebas se encuentran en el Anexo E: Pruebas unitarias e integrales.

R2. Implementación de un módulo de emisión de voto con el uso de una red de blockchain.

a) Descripción

Este resultado alcanzado permite brindar la funcionalidad de emisión de votos en el proceso electoral para los electores. Esto permite que todo elector pueda escoger su candidato de preferencia y emitir su voto dentro del sistema.

b) Métodos y herramientas utilizadas

Al igual que todos los módulos presentados en los demás resultados alcanzados, la primera herramienta utilizada para el desarrollo de este módulo son los documentos de diseño y de prototipado, por lo que se han generado diagramas de casos de uso, de actividades, de arquitectura y el prototipado. Así mismo, toda la documentación se encuentra en el Anexo D del presente documento. A continuación, se presenta el diagrama de actividades de la etapa de emisión del voto.

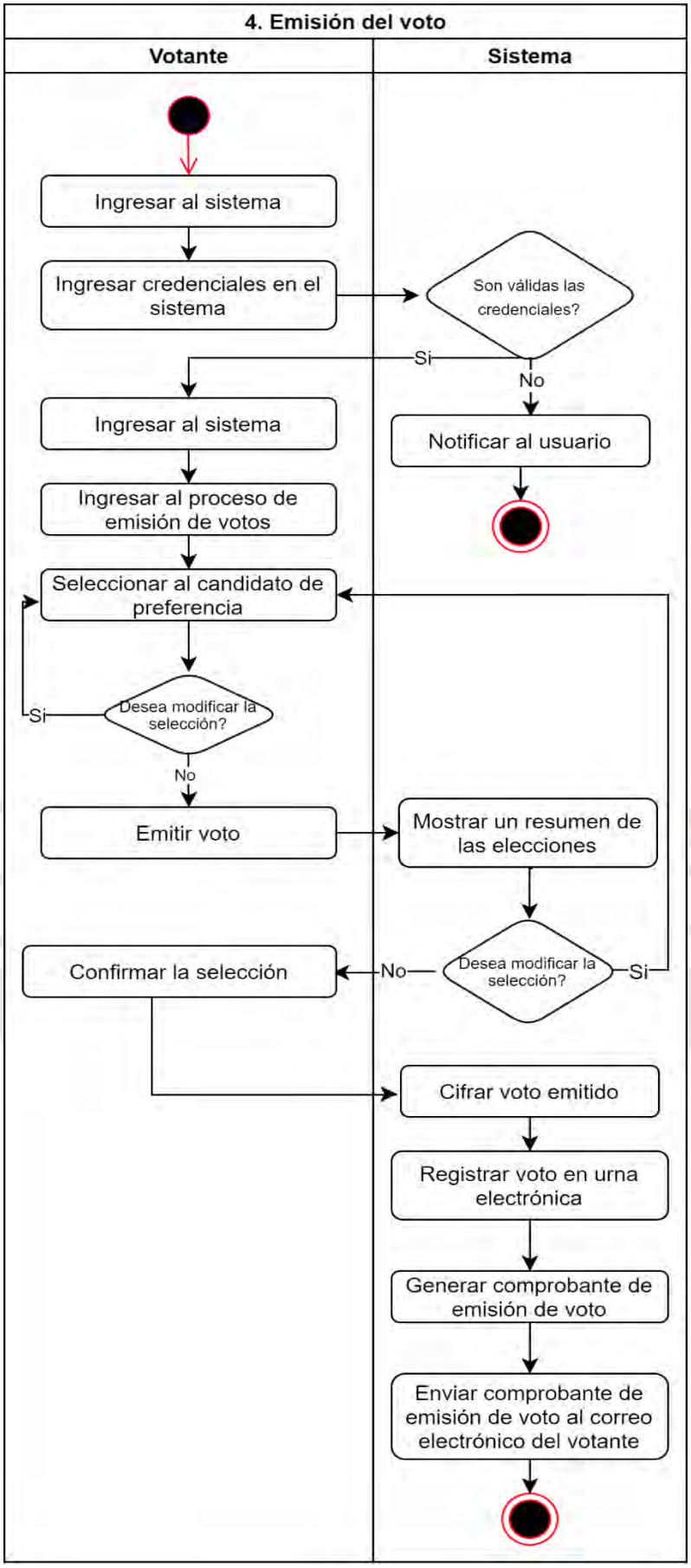


Figura 15. Diagrama de actividades del proceso de emisión de votos. (Elaboración propia)

Así mismo, este módulo interactúa con el usuario que posee rol de elector mediante una interfaz gráfica la cual posee una vista informativa y otra que simula las cédulas de votación en la que el elector puede emitir su voto por el candidato o partido político de su preferencia.

Figura 16. Interfaz gráfica de inicio del sistema para el elector. (Elaboración propia)

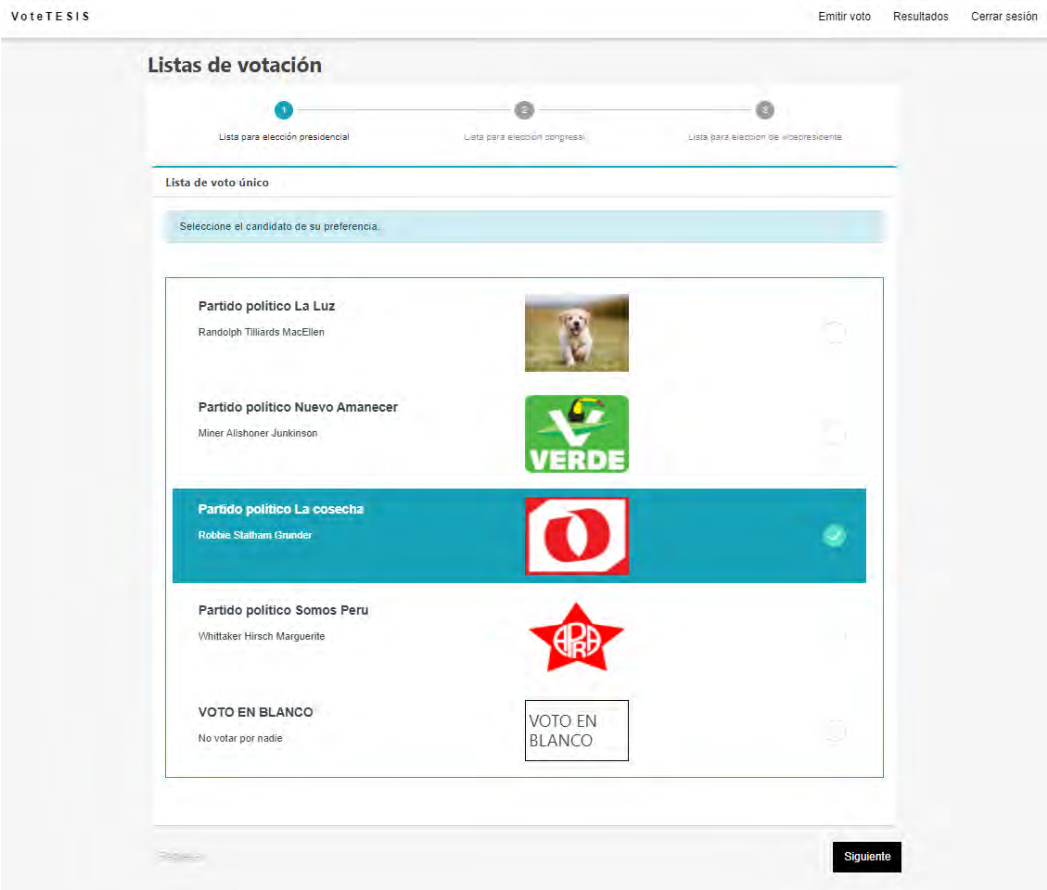


Figura 17. Interfaz gráfica de la cédula de votación del sistema para el elector. (Elaboración propia)

Por otro lado, el código fuente del presente módulo se encuentra en el repositorio de Gitlab, el cual puede ser accedido desde el siguiente enlace: https://gitlab.com/srbastian.sash/tesis2-e-voting-system.

La definición del presente módulo contempla los tipos de votos permitidos para las elecciones generales según la ley orgánica del Perú, por lo que está permitido el voto válido y el voto en blanco. Cabe mencionar que por la característica propia del sistema no se está contemplando el voto nulo debido a que por su definición no aplica en el voto electrónico no presencial. Así mismo, se cuenta con una sección de Resumen de selección antes de emitir el voto, vista donde se visualiza el resumen de los candidatos escogidos.

Otra consideración es que el sistema asume que el voto es voto en blanco en la cédula que el elector no hayas seleccionado ningún candidato. En esta sección es donde

el elector utiliza los ethers de su billetera electrónica para emitir su voto, los cuales se le depositaron previo al proceso electoral. Este procedimiento de uso de ethers se realiza mediante la herramienta Metamask la cual le permite al elector generar una transacción con el sistema y pagar con ella mediante su clave pública que se encuentra en su billetera y con el saldo de ethers que posee. A continuación, se muestra una imagen de cómo se registra la billetera al sistema.

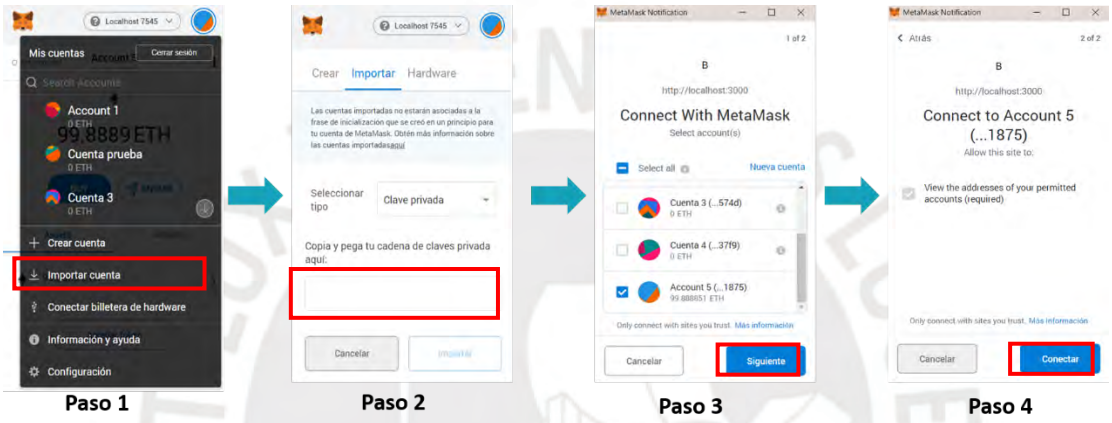


Figura 18. Registro de la billetera en el sistema por parte del elector. (Elaboración propia)

Finalmente, el indicador objetivamente verificable para este resultado alcanzado son las pruebas unitarias e integrales del presente módulo. Dichas pruebas se encuentran en el Anexo E: Pruebas unitarias e integrales.

R3. Implementación de un módulo de escrutinio de votos en una blockchain.

a) Descripción

Este resultado alcanzado permite brindar la funcionalidad de escrutinio de votos en el proceso electoral, pero con la finalidad que todo el proceso de escrutinio sea realizado y almacenado en la blockchain. Así mismo, toda la documentación se encuentra en el Anexo D del presente documento.

b) Métodos y herramientas utilizadas

Al igual que todos los módulos presentados en los demás resultados alcanzados, la primera herramienta utilizada para el desarrollo de este módulo son los documentos de diseño y de prototipado. Para esto, se han generado diagramas de casos de uso, de actividades, y de arquitectura.

El presente resultado alcanzado está relacionado con el algoritmo de encriptación del sistema debido a que recibe el conteo final que el algoritmo devuelve y lo procesa debido al tipo de lista de votación que se tenga. En el caso más sencillo es la elección de presidente y vicepresidente, pero respecto a las listas de votación de elección de congresistas se utiliza el concepto de cifra repartidora. Para esta sección no existe interfaz gráfica puesto que solo es el procesamiento de los datos obtenidos por el algoritmo de encriptación. Cabe mencionar que este proceso es activado por el administrador del sistema una vez haya concluido el proceso electoral.

Para la presente sección, se tuvo que realizar la comprensión del concepto del cálculo de la cifra repartidora aplicada en los procesos de votación en el gobierno peruano. Para un mayor entendimiento y aplicación de dicho cálculo se presenta dos casos que ejemplifiquen los dos tipos de Listas de votación utilizadas: candidato único y candidatos múltiples.

El caso para la ejemplificación es el siguiente:

> Se tiene un proceso electoral con dos listas de votación, una con la opción de candidato único y la otra con candidatos múltiples y los resultados obtenidos del algoritmo de encriptación fueron los siguientes. Así mismo, para la lista de votación 2 se están considerando que solo existen 6 curules disponibles.

Tabla 5. Lista de votación del tipo de “único candidato” para el caso ejemplificación. (Elaboración propia)

Lista de votación 1: Único candidato	
Partido político	**# de votos**
A	800
B	300
C	1 200
D	500

Tabla 6. Lista de votación del tipo de "múltiple candidato" para el caso ejemplificación. (Elaboración propia)

Lista de votación 2: Múltiples candidatos		
Partido político	**# de candidatos**	**# de votos**
A	10	800
B	8	300
C	20	1 200
D	12	500

En primer lugar, se desarrolla la lista de votación obtenida de la tabla 5 debido a que es el proceso más sencillo. En tal sentido se puede observar que el partido A tiene el mayor número de votos por lo que el partido A sería el ganador con su único candidato que este haya presentado.

En segundo lugar, se desarrolla la lista de votación obtenida de la tabla 6. Para ello se realiza el siguiente procedimiento.

- **Paso 0:** Se considera como paso 0 filtrar todas las listas que superen la vaya del 5%, la cual es una regla dentro de las elecciones congresales del Perú (ONPE, 2011). Para el caso propuesto todos los votos de los partidos políticos superan el 5% del total (> 86 votos).

- **Paso 1:** Ordenar la lista de votación de mayor a menor de acuerdo con los resultados obtenidos.

Tabla 7. Paso 1 del método de la cifra repartidora. (Elaboración propia)

Partido político	# de votos
C	1 200
A	800
D	500
B	300

- **Paso 2:** Dividir el número de votación de cada partido entre el número de curules que se desea cubrir, en este caso 6. Y escoger los 6 números más grandes de la tabla.

Tabla 8. Paso 2 del método de la cifra repartidora. (Elaboración propia)

	C	A	D	B
:1	1 200 (1)	800 (2)	500 (4)	300
:2	600 (3)	400 (5)	250	150
:3	400 (6)	266.667	166.667	100
:4	300	200	125	75
:5	240	160	100	60
:6	200	133.333	83.333	50

En este punto, se observa que el partido político C ha obtenido 3 escaños, el partido político A ha obtenido 2 escaños, el partido político D ha obtenido 1 escaño y el partido político B no ha obtenido ningún escaño.

- **Paso 3:** Obtener la cifra repartidora que consiste en tomar la última cifra o cociente a la que recibió un escaño, en este caso 400.

CIFRA REPARTIDORA → 400

- **Paso 4:** Dividir la votación obtenida de cada partido político que alcanzo un escaño entre la cifra repartidora (400).
    - **Partido C:** 1 200/ 400 = 3
    - **Partido A:** 800/400 = 2
    - **Partido D:** 500/400 = 1.25

- **Paso 5:** Ajustar los resultados a su valor entero menor.
    - **Partido C:** 3 congresistas
    - **Partido A:** 2 congresistas
    - **Partido D:** 1 congresista

Finalmente, el indicador objetivamente verificable para este resultado alcanzado son las pruebas unitarias e integrales del presente módulo. Dichas pruebas se encuentran en el Anexo F: Pruebas de funcionalidad del módulo de escrutinio de votos.

R4. Creación de un repositorio con el código fuente del software de acceso libre para la comunidad.

a) Descripción

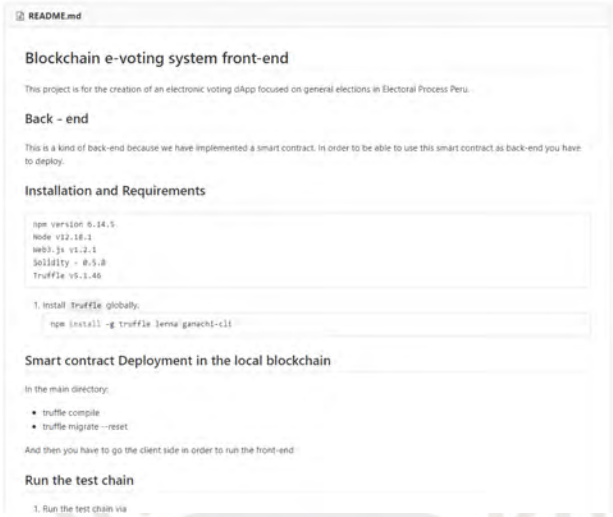
Este resultado alcanzado permite brindar la confianza al votante para que pueda acceder al código fuente del sistema del proceso electoral y validarlo. Así mismo, al ser de código abierto es más accesible para la comunidad y puede ser adaptado para el voto que se desee realizar en cualquier entidad.

b) Métodos y herramientas utilizadas

En este caso las herramientas a utilizar fueron el Gitlab el cual permite guardar el proyecto en el repositorio. El proyecto no solo se limita a ser accesible y descargado por la comunidad, sino que cualquier usuario puede realizar un *pull request* y el administrador del repositorio, en este caso mi persona, puede aceptarlos o rechazarlos. Así mismo, algunos desafíos han sido los comentarios al realizar *commits* en el repositorio ya que estos comentarios tienen que reflejar cual es el cambio que se ha realizado a la versión anterior.

Adicionalmente posee un *readme*, el cual expresa a detalle cómo utilizar el código fuente, como están estructuradas las carpetas y las versiones de las

herramientas que se utilizaron para su creación. A continuación, se muestra el *readme* del proyecto.


README.md

Blockchain e-voting system front-end

This project is for the creation of an electronic voting dApp focused on general elections in Electoral Process Peru.

Back - end

This is a kind of back-end because we have implemented a smart contract. In order to be able to use this smart contract as back-end you have to deploy.

Installation and Requirements

```
npm version 6.14.5
Node v12.18.1
Web3.js v1.2.1
Solidity - 0.5.8
Truffle v5.1.46
```

1. Install `Truffle` globally.

```
npm install -g truffle ]enna ganachi-cli
```

Smart contract Deployment in the local blockchain

In the main directory:

- truffle compile
- truffle migrate --reset

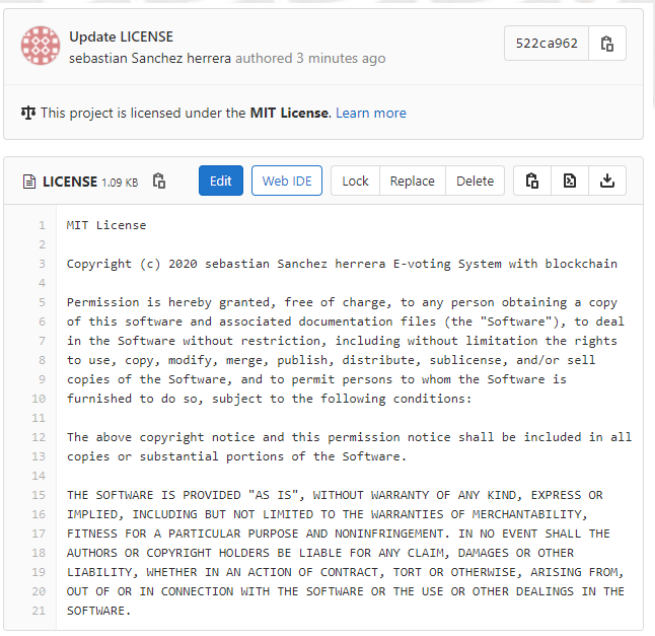
And then you have to go the client side in order to run the front-end

Run the test chain

1. Run the test chain via

Figura 19. *README.md* del repositorio. (Elaboración propia)

Finalmente, el proyecto posee un documento de licencia el cual valida que puede ser usado por la comunidad libremente. La figura 20 muestra una imagen que evidencia la licencia del proyecto.


Update LICENSE
sebastian Sanchez herrera authored 3 minutes ago
522ca962

This project is licensed under the **MIT License**. Learn more

LICENSE 1.09 KB | Edit | Web IDE | Lock | Replace | Delete

```
MIT License

Copyright (c) 2020 sebastian Sanchez herrera E-voting System with blockchain

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

Figura 20. *LICENSE* del repositorio. (Elaboración propia)

El acceso al repositorio está disponible mediante el siguiente enlace: https://gitlab.com/srbastian.sash/tesis2-e-voting-system.

4.3 Discusión

El presente proyecto se implementa como un prototipo orientado a validar los beneficios y factibilidad técnica del uso de blockchain en proceso electoral. Las consideraciones a tener en cuenta para el objetivo de implementar un sistema de voto electrónico de código abierto que gestione la información del proceso electoral de forma descentralizada para los actores del proceso electoral son:

En primer lugar, para poder utilizar el sistema en un proceso electoral se tienen ciertas consideraciones iniciales, las cuales no existen actualmente en el Perú. El primer requisito, es que todo elector que desee participar del uso de este sistema debe poseer una billetera electrónica, el cual se menciona en el presente capítulo usado para emitir el voto. Debido a ello, se asume que el elector debió haber solicitado participar en el proceso electoral mediante la gestión de adquirir un DNI electrónico. Es decir, al momento de adquirir el DNI electrónico, se le crea al elector una billetera electrónica, la cual es útil para participar en los procesos electorales que se creen en el presente sistema.

En segundo lugar, debido a que el sistema está enfocado en brindar seguridad al proceso de votación, este se basa en los pasos fundamentales que permitan obtener dicho objetivo. Sin embargo, esto puede ser más escalable en el sentido de contar con otros módulos de información al público o secciones que permitan simular pruebas en las cuales los electores pueden emitir votos ficticios previo al evento del voto electrónico, ello con el fin de familiarizarse con el sistema.

Por último, cabe mencionar que este tipo de aplicaciones descentralizadas para voto electrónico ya se está utilizando en Europa en varios países con un gran alcance a su población (Martínez, 2019).

Capítulo 5. Implementar un algoritmo de cifrado que provea la integridad de los datos

5.1 Introducción

El presente capítulo tiene como objetivo presentar un único resultado alcanzado y evidenciar los métodos, procedimientos y pasos que se realizan para obtenerlos. Este resultado es brindar un algoritmo de cifrado de votos con el fin de dar robustez a la seguridad del sistema. Cabe mencionar que este módulo es un proceso interno del sistema y no presenta vista gráfica.

Finalmente, se termina con una sección de discusión acerca del resultado alcanzado y las características que se deben tener en consideración en caso se quiera utilizar para otro público objetivo.

5.2 Resultados alcanzados

R5. Implementación de un módulo de cifrado que resguarde la integridad de los datos mediante cifrado probabilístico asimétrico parcialmente homomórfico El Gamal.

a) Descripción

Para incrementar el nivel de seguridad del sistema se utiliza el cifrado parcialmente homomórfico ElGamal el cual permite encriptar los votos y realizar el escrutinio sobre los votos encriptados y posteriormente desencriptar el resultado final mediante la operación realizada sobre ellos, la implementación del algoritmo se realiza en 4 etapas durante todo el proceso electoral. Así mismo, se realiza una pequeña prueba con el fin de mostrar el correcto uso del algoritmo.

b) Métodos y herramientas utilizadas

Los métodos utilizados para este resultado esperado están basados en la implementación del algoritmo criptografico ElGamal propuesta en el artículo *“Design and Implementation of Secure Remote e-Voting System Using Homomorphic Encryption”* (Jabbar & Alsaad, 2017). En tal sentido, el presente

módulo interactúa con los módulos de escrutinio, auditoría, y verificación individual por lo que para su implementación se utilizan 4 etapas, estas son:

- etapa de configuración,
- etapa de registro,
- etapa de votación y
- etapa de escrutinio.

Para un mejor entendimiento del algoritmo utilizado se simula un caso de ejemplo, el cual es desarrollado al final de cada etapa detallada en el algoritmo.

En tal sentido, se presenta el siguiente caso base para el ejemplo demostrativo:

- cantidad de electores = 3
- cantidad de partidos políticos = 4
- cantidad de autoridades de escrutinio = 2

En primer lugar, se evidencia un pseudocódigo del algoritmo ElGamal, con el fin de entender el algoritmo utilizado, para ello es necesario tener conocimientos de grupos cíclicos, puesto que es la base para la encriptación.

Tabla 9. Traducido de figura *ElGamal cryptosystem pseudocode*. (Jabbar & Alsaad, 2017)

Generación de la clave
q ← número primo largo x ← número que pertenece al grupo cíclico $G = < Z_q^*, X > \wedge$ x $\in [1, q - 1]$ g ← número raíz que será el generador del grupo cíclico $G = < Z_q^*, X >$ y ← g^x mod q clave pública ← función(q,y,q) clave privada ← x
Encriptación
r ← número aleatorio que pertenece al grupo cíclico $G =< Z_q^*, X > \wedge$ x $\in [1, q - 1]$ $C_1 = g^r \ mod \ q$ $C_2 = (p.y^r) \ mod \ q$ //p es un texto plano
Desencriptación
$P = [C_2(C_1^x)^{-1}]$ mod q

1. Etapa de configuración

Para un mayor entendimiento del algoritmo a realizar se presenta el siguiente diccionario de términos usados en código.

Tabla 10. Tabla de variables para la encriptación en la etapa de configuración. (Elaboración propia)

Variable	Descripción
q	**atributo "q_upper_limit"** del algoritmo
G	**atributo"g_ generator"** generadora del grupo cíclico
nTallyAuthorities	**atributo "nTally_authorities"** número de auditores o autoridades para el conteo del proceso electoral.
x_private_key[i]	número primo escogido por el i-ésimo auditor para ser asignada como su clave privada
y_public_key[i] = $g\_generator^{x\_private\_key[i]}$	clave pública generada a través de la clave privada del auditor o autoridad de escrutinio.

Para esta primera etapa de la configuración se realiza la generación de la clave del algoritmo. Esta fase tiene tres funciones importantes: "save_Q_G_values", "addNewTallyAuthoritie", "setPublicKeyAndPrivateKeyEncryption" y "createCipherTextVoters". Para la primera función, el administrador del sistema registrará los valores de q_upper_limit, g_ generator.

```
//Set q and g values
function save_Q_G_values(uint256 q, uint256 g) public payable onlyOwner {
    q_upper_limit = q;
    g_generator = g;
}

function setPublicKeyAndPrivateKeyEncryption(
    string memory _dni,
    uint256 _private_key
) public payable returns (uint256) {
    if (funciones.compareStrings(tally_authorities[_dni].dni, _dni)) {
        tally_authorities[_dni].x_private_key = _private_key;
        tally_authorities[_dni].y_public_key = g_generator**_private_key;

        return tally_authorities[_dni].y_public_key;
    } else {
        return 0;
    }
}
```

Figura 20. Imagen del código fuente de la etapa de configuración. (Elaboración propia)

Respecto al caso base del ejemplo demostrativo los valores configurados de q y g son:

- g_generator = 5
- q_upper_limit = 13
- nTally_authorities = 2

La segunda función “setPublicKeyAndPrivateKeyEncryption”, es una acción realizada por cada auditor del sistema creado por el administrador del sistema. Esta función permite asignar una clave pública y privada a cada auditor del sistema, la cual es necesaria para la desencriptación del resultado final del proceso electoral. Respecto al ejemplo demostrativo se presentan las claves públicas y privadas de encriptación ingresada por las dos autoridades de escrutinio.

Tabla 11. Tabla de claves públicas y privadas de las autoridades de escrutinio para el proceso de encriptación. (Elaboración propia)

i	**Autoridad**	**x_private_key[i]**	**y_publick_key[i]**
1	Autoridad 1	7	$5^7 = 78125$
2	Autoridad 2	11	$5^{11} = 48828125$

Así mismo, se muestra la vista de la interfaz sobre la cual el administrador del sistema registra los datos para la configuración y las “autoridades de escrutinio” que hacen la vez de auditor durante todo el proceso electoral.

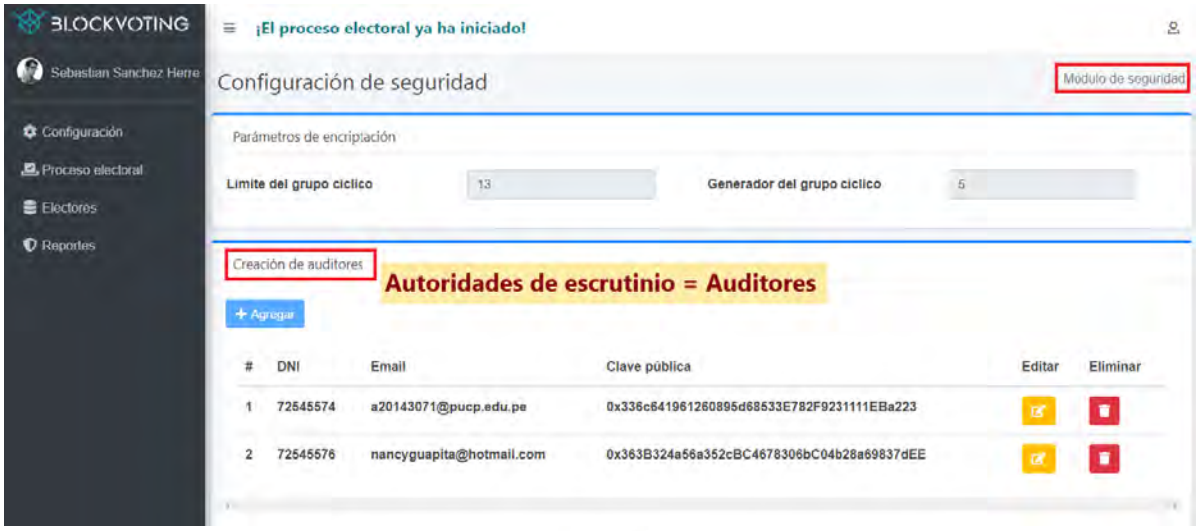


Figura 21. Imagen de la interfaz gráfica sobre la cual se registra los parámetros para la encriptación. (Elaboración propia)

2. Etapa de registro

La segunda etapa pertenece a la fase de encriptación del algoritmo ElGamal y consiste en generar un valor encriptado que contenga los votos emitidos del elector.

Tabla 12. Tabla de variables para la encriptación en la etapa de registro. (Elaboración propia)

Variable	**Descripción**
$r_i = a_{i,1} + a_{i,2} + \ldots + a_{i,n}$; $donde\ a_{i,j} \in \{0,1\}$.	**atributo *"reference"*,** es un número entero aleatorio con el total de numero de bits igual al número de partidos políticos.
y = número aleatorio $\in Z_q^*$	**atributo "y"** número entero aleatorio perteneciente al i-ésimo auditor la cual será su clave privada.
$A_{i,j} = g^{y_{i,j}}$	***atributo "ciphertext*_A"** para la encriptación del voto emitido por elector.
$B_{i,j} = \begin{cases} g.\left(\Pi_{t=1}^{\text{nTally\_authorities}} y_t\right)^{y_{i,j}}; si\ a_{i,j} = 0 \\ g^{-1}.\left(\Pi_{t=1}^{\text{nTally\_authorities}} y_t\right)^{y_{i,j}}; si\ a_{i,i} = 1 \end{cases}$	***atributo "ciphertext*_B"** para la encriptación del voto emitido por elector.

Esta etapa de registro es realizada una vez iniciado el proceso electoral, debido a que se generan los "*ciphertexts*" para poder realizar el conteo de los votos

encriptados. Así mismo, se agrega el atributo "*reference*" que posee cada elector con el cual se firma los votos a fin de mantenerlos en secreto. A continuación, se pone en evidencia el código fuente de la implementación de la presente etapa.

```
//Step 2: Registration: Set the ciphertext for each Voter
function createCiphertextVoters() private onlyOwner {
    y = new uint256[][](cantVoterLists);
    ciphertext_A = new uint256[][](cantVoterLists);
    ciphertext_B = new uint256[][](cantVoterLists);

    for (uint256 pos = 0; pos < cantVoterLists; pos++) {
        //reference ri for each voter
        uint256 _auxCiphertext = randModulus( cantPoliticalParties, keyVoterList[pos] );
        _reference[pos] = toBinaryString( _auxCiphertext, 2**cantPoliticalParties - 1 );

        //create y value
        y[pos] = new uint256[](cantPoliticalParties);

        //Create ciphertext A
        ciphertext_A[pos] = new uint256[](cantPoliticalParties);

        //Create ciphertext B
        ciphertext_B[pos] = new uint256[](cantPoliticalParties);

        for (uint256 posCand = 0; posCand < cantPoliticalParties; posCand++) {
            y[pos][posCand] =
                uint256(
                    keccak256(abi.encodePacked(posCand, _reference[pos], block.timestamp, block.difficulty, msg.sender))
                ) % q_upper_limit;
            ciphertext_A[pos][posCand] = g_generator**y[pos][posCand];

            ciphertext_B[pos][posCand] = 1;
            for (uint256 posB = 0; posB < nTally_authorities; posB++) {
                ciphertext_B[pos][posCand] =
                    ciphertext_B[pos][posCand] *
                    tally_authorities[keyTallyAuthoritiesList[posB]]
                        .y_public_key;
            }
            ciphertext_B[pos][posCand] =
                ciphertext_B[pos][posCand]**y[pos][posCand];
            if (compareStrings(getSlice(_reference[pos], posCand), "0")) {
                ciphertext_B[pos][posCand] = ciphertext_B[pos][posCand] * g_generator;
            } else {
                ciphertext_B[pos][posCand] = ciphertext_B[pos][posCand] / g_generator;
            }
        }
        voters[keyVoterList[pos]].referenceValue = _reference[pos];
        voters[keyVoterList[pos]].ciphertext_A = ciphertext_A[pos];
        voters[keyVoterList[pos]].ciphertext_B = ciphertext_B[pos];
    }
}
```

Figura 22. Código fuente de la creación de *ciphertexts* (A, B) para la encriptación de los votos. (Elaboración propia)

Respecto al ejemplo demostrativo se presentan las claves públicas y privadas de encriptación ingresada por las dos autoridades de escrutinio al aplicar las ecuaciones de la tabla 7.

- $r_1 = (0,0,0,0), r_2 = (0,0,1,0), r_3 = (0,0,1,1)$

- $y_{1,1} = 5;\ \ y_{1,2} = 10;\ \ y_{1,3} = 3;\ \ y_{1,4} = 5$
- $y_{2,1} = 12;\ \ y_{2,2} = 8;\ \ y_{2,3} = 4;\ \ y_{2,4} = 12$
- $y_{3,1} = 4;\ y_{3,2} = 7;\ \ y_{3,3} = 4;\ \ y_{3,4} = 6$

Así mismo, los valores de los textos cifrados de “A” generados para cada elector son:

- Elector 1:
  - $A_{1,1} = 3125;\ A_{1,2} = 9765625;\ \ A_{1,3} = 125;\ A_{1,4} = 3125$
- Elector 2:
  - $A_{2,1} = 244140625;\ A_{2,2} = 390625;\ \ A_{2,3} = 625;\ A_{2,4} = 244140625$
- Elector 3:
  - $A_{3,1} = 625;\ A_{3,2} = 78125;\ \ A_{3,3} = 625;\ A_{3,4} = 15625$

Para el caso de los textos cifrados “B” solo se mostrará el generado para el primer elector:

- $B_{1,1}$ = 40389678347315804437080502542478654959268169477581977844238281 25
- $B_{1,2}$ = 17200292317904156532503109672098612811350018480773336435266628927180757121653
- $B_{1,3}$ = 27755575615628913510590791702270507812 5

- $B_{1,4}$ = 40389678347315804437080502542478654959268169477581977844238281 25

3. Etapa de votación

En esta etapa se registra el voto del candidato, esto sucede una vez que el elector haya ingresado al sistema y seleccionado la opción de emitir su voto. El voto es registrado en la *blockchain* y encriptado con el atributo *reference* propio del elector. Una vez firmado el voto este es registrado en la blockchain. Para encriptar el voto se utiliza la conversión de la tabla 13.

Tabla 13. Tabla que determina el valor de β de cada voto. Traducido de *Table 2 Voter guide to determine β*. (Jabbar & Alsaad, 2017)

Referencia	Voto	β
0	Si	1
1	Si	-1
0	No	-1
1	No	1

La forma de registrar el voto es de la siguiente forma: Si el atributo *reference* del elector es (0,1,1,0) asumiendo que existen solo 4 partidos políticos. Y asumiendo que en una de las cédulas el elector escogió al 3er partido político, entonces su β será (-1,1,-1,-1). Este atributo es el que se registra en la blockchain y es a partir de este punto en donde el valor registrado del voto no evidencia lo que el votante a elegido por lo que se cumple la característica del voto secreto propuesto en el catálogo de requerimientos. Respecto a esta variable, β, se hace mayor énfasis en la sección 6.R8 la cual es clave para la verificación individual del votante.

Por otra parte, respecto al uso de blockchain se está aprovechando la característica de los eventos, la cual permite mantener un log de registros de los votos emitidos. Este log que genera la blockchain registra la clave pública de la billetera del elector y su dni asociado con el fin de mantener un registro de los electores que interactuaron con el sistema. Así mismo, se hace mayor énfasis sobre este punto en la sección 6.R7 la cual se detalla todas las funcionalidades de auditoría que posee el sistema.

Respecto al ejemplo demostrativo se mostrará el valor del atributo β de cada elector asumiendo que cada elector inicio su sesión en el sistema y emitió su voto electrónico. Se sabe además que:

- El primer elector eligió al partido político 1
- El segundo elector eligió al partido político 2
- El tercer elector eligió al partido político 1

Tabla 14. Tabla que determina el valor del atributo β de cada voto. (Elaboración propia)

Elector	r_i				Voto emitido				β			
E_1	0	0	0	0	Si	No	No	No	1	-1	-1	-1
E_2	0	0	1	0	No	Si	No	No	-1	1	1	-1
E_3	0	0	1	1	Si	No	No	No	1	-1	1	1

4. Etapa de escrutinio

Tabla 15. Tabla de variables para la encriptación en la etapa de escrutinio. (Elaboración propia)

Variable	**Descripción**
$X_{T,j} = \prod_{i=1}^{n_v} A_{i,j}{}^{\beta_{i,j}}$	Ecuación (a) para el escrutinio de los votos encriptados realizada por la autoridad de escrutinio.
$Y_{T,j} = \prod_{i=1}^{n_v} B_{i,j}{}^{\beta_{i,j}}$	Ecuación (b) para el escrutinio de los votos encriptados realizada por la autoridad de escrutinio.

$eq_j = Y_{T,j}.\prod_{i=1}^{n_T} X_{i,j}{}^{-1}$ $= \prod_{i=1}^{n_T} g^{\beta_{i,j}(-1)^{a_{i,j}}}$ $= g^{y_j - n_j}$	ecuación (c) para el escrutinio de los votos encriptados realizada por el administrador.
$g^{2y_j - n_v} = g^{y_j - n_j}$ $z_j = \frac{ln(g^{y_j - n_j})}{ln(g)}$ $n_j = n_v - y_j$	Ecuación (d) para la desencriptación del resultado final realizado por el administrador.

La presente etapa tiene como finalidad el escrutinio de los votos, la cual hace uso de la propiedad homomórfica del esquema de encriptación ElGamal mencionado en esta sección. En este punto, las autoridades de escrutinio utilizarán sus claves públicas para la etapa del conteo siguiendo las ecuaciones (a) y (b) propuestas en la tabla 8.

Una vez que las autoridades de escrutinio realicen las ecuaciones mencionadas, estos valores son enviados al administrador del sistema el cual ejecuta la ecuación (c) donde y_j y n_j son valores booleanos que representan Si o No, respectivamente por el partido político C_j. Estos valores son obtenidos por la ecuación (d) lo que genera los resultados finales de las elecciones. Sin embargo, dichos resultados solo devuelven el conteo final para cada cédula, por lo que son procesados y publicados en el sistema en el módulo de escrutinio, el cual está enfocado al procesamiento de los resultados finales de los votos y generación de reportes.

Respecto al ejemplo demostrativo una vez que las autoridades de escrutinio hayan ejecutado la ecuación (a) y la ecuación (b) se obtiene los siguientes valores para la autoridad 1 con clave privada de encriptación 7:

- $X_{1,1}$ = 0.000000000000000000000209715200000000006
- $X_{1,2}$ = -5787529170519.066
- $X_{1,3}$ = 0.000000000000000000000000291038304567337 03
- $X_{1,4}$ = 18820.065978765804

Posterior a ello, el administrador del sistema ejecuta la ecuación (c) y (d) y se obtienen los siguientes resultados:

- Votos por el partido político 1: #SI=2 y #NO=1
- Votos por el partido político 2: #SI=1 y #NO=2
- Votos por el partido político 3: #SI=0 y #NO=3
- Votos por el partido político 4: #SI=0 y #NO=3

Finalmente, el indicador objetivamente verificable para este resultado alcanzado es la prueba de funcionalidad del algoritmo donde se detalla a profundidad el caso ejemplo demostrativo de la presente sección mediante la herramienta Remix, la cual permite utilizar la tecnología blockchain mediante una interfaz gráfica utilizada para evidenciar los resultados y cálculos parciales que se utilizarán. Dichas pruebas se encuentran en el Anexo E: Pruebas unitarias e integrales del algoritmo de encriptación.

5.3 Discusión

Para el resultado alcanzado mencionado en esta sección se utilizó un algoritmo que si bien es muy utilizado para procesos de votación se ajustó a las características propias del proceso electoral peruano como, por ejemplo, la del voto secreto y la característica de no tener voto múltiple, pese a que el algoritmo lo soporta. Así mismo, otras consideraciones que se tuvieron

es el de ejecutar la etapa de escrutinio del algoritmo por cada cédula registrada en el sistema, aunque esto se encuentra acotado a un máximo de 4 cédulas por proceso electoral. Adicionalmente, no se está considerando la característica de que en caso existan dos partidos políticos que lleguen a tener un puntaje alto y cercano se evalué la segunda vuelta, es decir, el algoritmo solo devuelve el conteo final en una *data* bruta, es por ello por lo que se mencionó que el módulo de escrutinio es el encargado de procesar dicha *data*. Finalmente, con el uso del algoritmo de encriptación se permite incrementar el nivel de seguridad del registro de los votos ya que existe un nivel de procesamiento computacional de los votos registrados.

Capítulo 6. Utilizar estándares legales y técnicos en las fases de emisión, escrutinio y auditoría

6.1 Introducción

El presente capítulo evidencia el cumplimiento del objetivo 3, el cual es “Utilizar estándares legales y técnicos en las fases de emisión, escrutinio y auditoria”. El cumplimiento de este objetivo se evidencia con la presentación a detalle de los últimos 3 resultados alcanzados. Por lo tanto, se muestran evidencias de los métodos, procedimientos, pasos y dificultades que se presentaron para obtenerlos. Así mismo, estos están basados en estándares legales y técnicos puesto que el cumplimiento de estos determina dichas características que debe poseer el sistema para ser considerado seguro; estas características se ven evidenciadas en las fases de emisión, escrutinio y auditoría por lo que los resultados presentes están enfocados en dichas etapas.

El primer resultado alcanzado consiste en la creación del catálogo de requerimientos el cual se basa en los estándares propuestos por el Consejo Europeo y la ley Orgánica de elecciones N° 26 859. El segundo, se enfoca en la creación de un módulo que permita la auditabilidad del proceso electoral. la cual es esencial para evidenciar la seguridad y fiabilidad del proceso electoral. El tercero, está enfocado en la característica de la verificación individual del sistema.

Finalmente, se termina con una sección de discusión acerca de los resultados alcanzados y las características que se deben tener en consideración en caso se quiera utilizar otro público objetivo.

6.2 Resultados alcanzados

R6. Catálogo de requerimientos basados en los estándares de los sistemas de votación electrónica propuesta por el Consejo Europeo y la ley Orgánica de elecciones N° 26 859.

a) Descripción

Es el documento que contiene el catálogo de requerimientos para el sistema propuesto el cual está basado en Estándares de los sistemas de votación electrónica propuestos por el Consejo Europeo y la ley orgánica de elecciones N° 26 859. El presente documento se encuentra en el Anexo C del presente documento.

b) Métodos y herramientas utilizadas

Para determinar las funcionalidades que va a tener el sistema propuesto se realizó la creación del catálogo de requerimientos. En tal sentido, para determinar dichas funcionalidades y plasmarlas en el catálogo se utilizó los dos documentos mencionados en la descripción: los estándares de los sistemas de votación electrónica propuestos por el Consejo Europeo y la ley orgánica de elecciones N° 26 859. Se utilizó dos documentos como base para la creación del catálogo de requerimientos, puesto que el primero da un enfoque de características de un sistema a nivel mundial, mientras que el segundo, la ley Orgánica de elecciones, detalla las particularidades que posee el voto en nuestra nación, esto con el fin de brindarle mayor alcance y flexibilidad al sistema proporcionado. Así mismo, al finalizar la versión preliminar del documento se detallaron 34 requerimientos. Posterior a ello, se validó el documento con la Licenciada Angela Quispe Medina, experta en procesos electorales de la ONPE, la cual ha participado en la creación de 6 procesos electorales. Dicho documento se perfeccionó y validó en su versión actualizada, la cual cuenta con 34 requerimientos. Por último, como evidencia de la validación del

documento se generó un Acta de reunión con la experta en procesos electorales, así como el video de la reunión que tuvo lugar en la plataforma de *Google Meet.*

Adicionalmente, se realizó una segunda validación del catálogo de requerimientos con el magister Gerardo Enrique Salazar Lara especialista en Transformación digital y actualmente es consultor para la ONPE para las elecciones generales del 11 de abril del próximo año. En dicha validación se ajustaron los temas del proceso de voto electrónico no presencial, así como la eliminación de los roles de personero en el local de votación, y el personal encargado del área de digitación de los votos, y miembros de mesa. Sin embargo, existen los auditores y personeros que se encuentran en la sede central de la ONPE que van a revisar los reportes que genera el sistema tales como el listado de electores, resultados de escrutinio, listado de votos válidos y votos en blanco con el fin de tener una correcta auditoria post proceso electoral.

A continuación, se presenta una versión resumida del catálogo de requerimientos la cual omite la prioridad del requerimiento, el módulo al cual pertenece y las observaciones que se detallaron para cada uno de ellos. Sin embargo, la versión completa del catálogo de requerimientos incluyendo la prioridad y observaciones sobre este se encuentra en el Anexo C.

Tabla 16. Catálogo de requerimientos resumido. (Elaboración propia)

CÓDIGO	REQUISITO
PRE001	El sistema no debe permitir que un elector pueda emitir su voto antes de empezar la etapa de votación.
PRE002	El administrador debe poder crear un proceso electoral.
PRE003	El administrador debe poder registrar los candidatos.
PRE004	El administrador debe configurar el proceso electoral para que se permita crear cédulas de voto único y/o voto para congresistas y voto para parlamento andino seleccionados en el caso que sea necesario.
PRE005	El sistema debe mostrar toda la lista de los candidatos permitidos.

REG001	El administrador debe ingresar un archivo .csv que contendrá la lista de electores que participaran del proceso electoral.
REG002	El sistema debe procesar el archivo que contenga la lista de los electores y registrar la billetera electrónica del elector al sistema. Así mismo se notificará esta acción al elector vía correo electrónico.
EMI001	El sistema debe permitir ingresar al elector a la plataforma para realizar el proceso de elección.
EMI003	El sistema debe permitir al elector seleccionar al candidato de su preferencia.
EMI004	El sistema debe mostrar un mensaje al elector que solo puede introducir una papeleta en la urna electrónica.
EMI005	El sistema debe mostrar todos los partidos políticos en las cédulas manejando el mismo tipo de fuente, color y tamaño con el fin de no influenciar la intención del voto del elector hacia un partido político.
EMI006	El sistema debe impedir que una vez que el elector haya emitido su voto, éste pueda modificarse.
EMI007	El elector puede emitir votos en blanco.
EMI008	El elector puede modificar su voto en todo momento antes de la emisión definitiva del mismo.
EMI009	El sistema debe mostrar un resumen de las cédulas con las selecciones realizadas por el elector antes de que este sea emitido.
VER001	El sistema debe emitir un comprobante el cual indique que el elector a emitido su voto.
VER002	El sistema notificará al elector vía correo electrónico que su voto ha sido emitido.
VER003	El comprobante emitido no debe mostrar información sobre el voto emitido.
ESC001	El proceso electoral debe iniciar y finalizar de forma manual por el administrador del sistema.
ESC002	El sistema debe poseer una urna electrónica en donde se almacenen los votos registrados para su posterior escrutinio.
ESC003	El sistema debe garantizar que los votos contenidos en la urna electrónica y los votos que se escrutan son, y seguirán siendo anónimos.
ESC004	El sistema debe escrutar todo voto depositado en la urna electrónica.
ESC005	El sistema debe revelar el número de votos emitidos solo cuando se proceda al cierre de la urna electrónica.
ESC006	El sistema debe realizar el proceso de escrutinio una vez finaliza la etapa de emisión de votos.
CIF001	El sistema preservará la confidencialidad de los votos.
CIF002	El sistema encriptará cada voto que haya sido emitido y lo guardará en la urna electrónica.
CIF003	El sistema desencriptará los resultados finales una vez haya terminado de realizar el conteo de los votos.
AUD001	El administrador y el auditor son los únicos que deben poder visualizar los reportes de sufragio y escrutinio.

AUD002	El sistema mantendrá un control de auditoria respecto a usuario y hora donde se registre quien ha realizado cada modificación.
AUD003	El sistema debe impedir que la sección de auditoria sea modificada. así mismo, esta sección solo será visible para el administrador del sistema y el director de mesa presidencial.
AUD004	El sistema generará un acta de sufragio el cual detallará la hora en la que se emitió cada voto, la cuenta asociada a dicho voto, y el voto encriptado.
AUD005	El sistema generará un acta de escrutinio que detalle el número de votos en blanco, votos válidos, así como la cantidad de votos por cada partido electoral, la hora de inicio del escrutinio y la hora final.
AUD006	La sección de auditoria salvaguardará el anonimato de los electores en todo momento.
AUD007	El sistema web puede ser utilizado en cualquier sistema operativo.

R7. Implementación de funcionalidades que permitan auditar al sistema.

a) Descripción

La auditabilidad es una característica fundamental para un proceso electoral electrónico. En tal sentido, el presente resultado permite brindar un medio por el cual se pueda auditar el proceso electoral durante el sufragio y el escrutinio. Así mismo, toda la documentación se encuentra en el Anexo D del presente documento.

b) Métodos y herramientas utilizadas

Para obtener el presente resultado se ha comenzado con el desarrollo de los diagramas de diseño: casos de uso, actividades y arquitectura; así como el prototipado. Estos documentos han sido desarrollados de la misma forma que los demás módulos presentes en los demás resultados esperados. Adicionalmente, para este módulo se manejan dos roles en el sistema, el primero es el administrador el cual solo accede solo puede visualizar la información del módulo de reportes, y el otro es un rol el cual es accedido por la sede principal de la ONPE, que, para aplicación del presente proyecto, es el rol auditor, quien puede ver el resultado del conteo y el reporte de escrutinio generado en el sistema. Cabe mencionar que para

esta parte se generaron graficas estadísticas que muestren los resultados finales del proceso electoral. Este módulo posee tanto vista para el usuario como manejo de lógica la cual se encuentra en el contrato inteligente. Adicionalmente, se le asigna el rol de auditor a la autoridad de escrutinio, es decir, el auditor es aquel que participa activamente en el módulo de encriptación del proceso electoral.

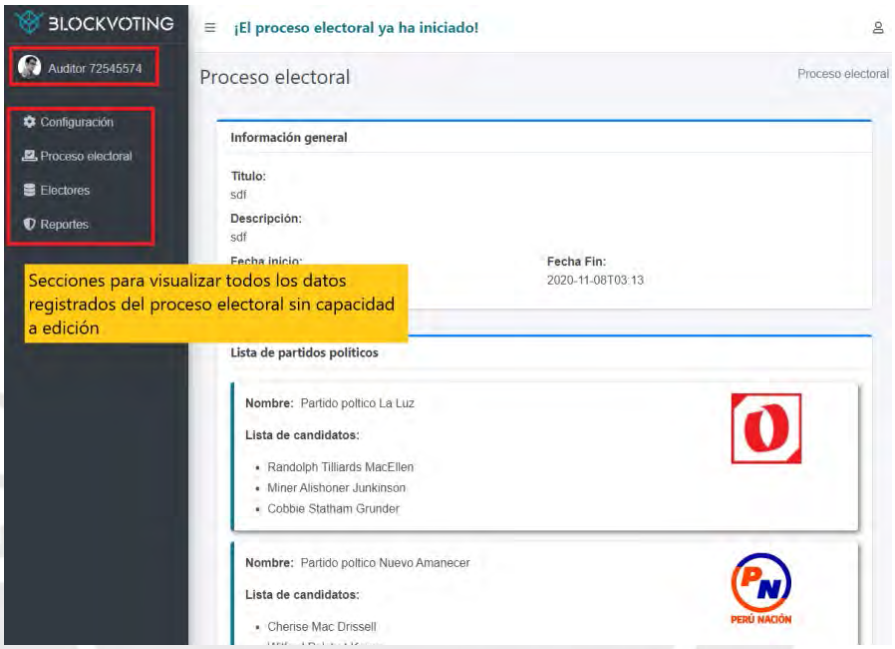


Figura 23. Imagen de interfaz gráfica de la sección principal de un auditor. (Elaboración propia)

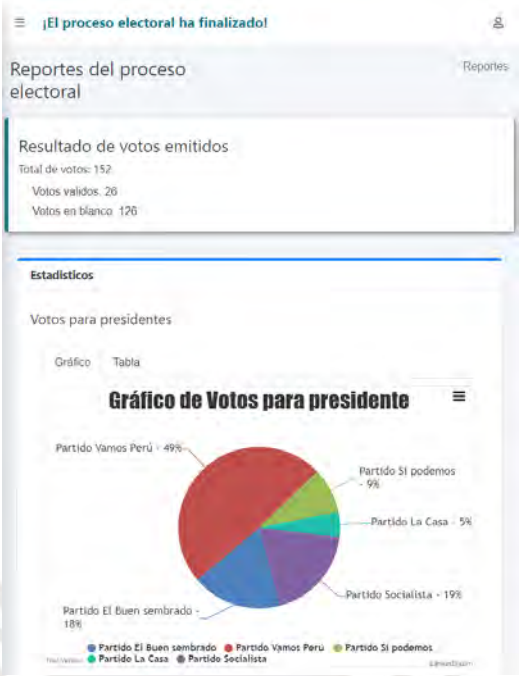


Figura 24. Imagen de interfaz gráfica del reporte de elecciones. (Elaboración propia)

Así mismo, cabe mencionar que el detalle mencionado de la capacidad de auditoría del sistema mencionada en esta sección puede ser realizada por cualquier entidad ya que se utilizan interfaces gráficas usables para el usuario auditor. Sin embargo, se puede manejar un nivel de auditoría en un siguiente nivel. Esto es, poniendo el escenario donde el auditor posea conocimientos de blockchain. Las funciones de consulta sobre las variables declaradas en el contrato inteligente utilizadas en el sistema están diseñadas para que sean accedidas por el administrador del sistema y cada entidad que el agregue como auditor y/o autoridad de conteo. Sin embargo, respecto a las variables que son utilizadas como claves privadas solo son accesibles por el mismo dueño de dicha clave y no otra entidad.

Finalmente, para este segundo escenario se está haciendo el uso de eventos de los contratos inteligentes para la emisión de los votos por parte de los electores. Es decir, cada vez que el elector emite su voto se genera un evento en la blockchain, la cual registra un file log de la transacción realizada, numero de bloque generado,

timestamp, y demás *metadata* que sirve para poder tener seguimiento sobre ella. De dicha forma se puede saber cuándo un elector emitió su voto, y en casos de seguridad cuando este sufrió alguna modificación.

```
event votedEvent(address _addressCandidate, string dni);

function emitVote(string memory  dni, uint [] memory _vote) public{
    emit votedEvent(msg.sender, _dni);
    if (msg.sender == voters[_dni].publicKey) {
        int[][] memory voteEncrypted = new int[][](cantVotingLists);
        voters[_dni].castVote = true;

        for (uint256 posVL = 0; posVL < cantVotingLists; posVL++) {
            voteEncrypted[posVL] = new int[](cantPoliticalParties);

            for (uint256 posPP = 0; posPP < cantPoliticalParties; posPP++) {
                if (compareStrings(getSlice(_reference[posVL], posPP), "0")) {
                    if (_vote[posVL] == posPP){ //YES
                        voteEncrypted[posVL][posPP] = 1;
                    }else {//NO
                        voteEncrypted[posVL][posPP] = -1;
                    }
                }else {  // = "1"
                    if (_vote[posVL] == posPP){ //YES
                        voteEncrypted[posVL][posPP] = -1;
                    }else { //NO
                        voteEncrypted[posVL][posPP] = 1;
                    }
                }
            }
            voters[_dni].voteEncrypted = voteEncrypted;
        }
    }
}
```

Figura 25. Código fuente del uso de eventos en un contrato inteligente. (Elaboración propia)

Finalmente, el indicador objetivamente verificable para este resultado alcanzado son las pruebas unitarias e integrales del módulo de auditoría. Dichas pruebas se encuentran en el Anexo E: Pruebas unitarias e integrales.

R8. Implementación de un módulo que permita la verificación individual de los votos.

a) Descripción

Otra de las características que debe contar el voto electrónico y es de prioridad exigible es presentar un medio por el cual el elector pueda conseguir la verificación individual del voto que ha emitido por lo que este resultado alcanzado tiene como finalidad la implementación de un módulo que permita la verificación individual de

los votos. Así mismo, toda la documentación se encuentra en el Anexo D del presente documento.

b) Métodos y herramientas utilizadas

Para iniciar con la implementación del presente módulo se partió del diseño, es por ello por lo que se han desarrollado los diagramas de casos de uso, actividades, arquitectura y el prototipado. Adicionalmente, este módulo maneja tanto una interfaz gráfica para el usuario y posee la lógica de negocio desarrollada en el contrato inteligente.

Adicionalmente, cabe mencionar que este resultado alcanzado está relacionado con el algoritmo de encriptación que usa el sistema. Debido a que en la sección 5 se detalló el uso del algoritmo entonces en esta sección se utiliza el atributo $\boldsymbol{\beta}$, el cual simboliza el voto encriptado que se generó al elector como resultado de emitir su voto. Dicho voto encriptado se utiliza como identificador del elector para corroborar su voto emitido. Este identificador es enviado por correo al elector una vez haya emitido su voto. Así mismo, es publicado en el panel del sistema, donde el elector puede ver que su voto ha sido emitido. Es importante resaltar que como el sistema maneja varias listas de votación, entonces se genera un identificador de voto encriptado por cada lista de votación.

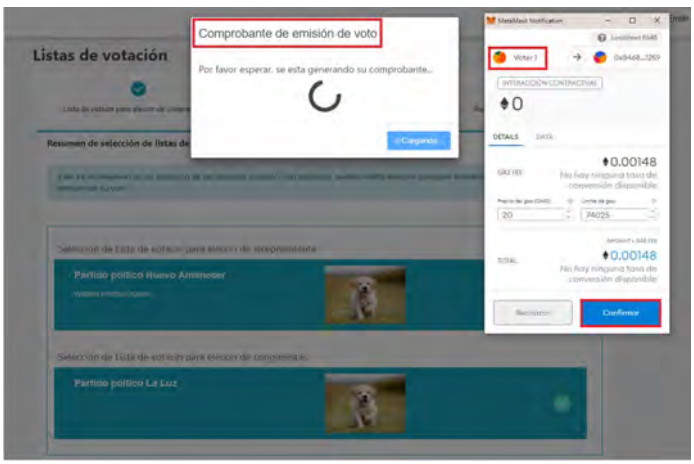


Figura 26. Imagen de la vista gráfica de la emisión del voto por parte del elector. (Elaboración propia)

El aspecto de seguridad que involucra dicho identificador es que, si algún parámetro que se utiliza en el algoritmo de encriptación es modificado, este identificador también sería alterado, por lo cual se puede validar llevar un control de la no alteración del voto.

Figura 27. Imagen de la vista gráfica del panel del elector con los identificadores generados por cada lista de votación. (Elaboración propia)

Finalmente, el indicador objetivamente verificable para este resultado alcanzado son las pruebas unitarias e integrales del módulo de verificación individual. Dichas pruebas se encuentran en el Anexo E: Pruebas unitarias e integrales. Cabe mencionar que el código del presente módulo se encuentra en el repositorio de Gitlab en el siguiente enlace: https://gitlab.com/srbastian.sash/tesis2-e-voting-system.

6.3 Discusión

Los resultados alcanzados presentes en la sección 6 se puede observar que intervienen en diferentes etapas del proceso electoral que brinda el sistema. Así mismo, como ya se mencionó en secciones anteriores, el sistema a implementar está enfocado en el gobierno electrónico a nivel nacional, es decir, para elecciones generales, es por ello por lo que se adaptó el catálogo de requerimientos enfocado a dicho nivel de gobierno electrónico. Adicionalmente, el sistema maneja la entidad de auditor y autoridad de escrutinio con el mismo rol, con el fin de que el auditor tenga participación durante todo el proceso electoral y pueda visualizar el desarrollo de este en todo momento. Como se detalló en la sección 6.2R6 el auditor hace el rol de personero en la sede central de la ONPE con el fin de visualizar en todo momento como se va desenvolviendo el proceso electoral y poder reunir todos los reportes que el sistema genera. Respecto al uso de las ventajas que posee blockchain se está usando el concepto de eventos en el contrato inteligente para la auditoria de los votos emitidos, pero pueden utilizarse más eventos en otras etapas del proceso electoral, esto depende del nivel de auditoria que se desee manejar en el sistema. Así mismo, se podría incrementar funcionalidades en el sistema, como el de exportar los reportes generados a archivos .xlsx, .pdf, etc. Finalmente, respecto a la verificación individual también se podría crear mayores funcionalidades. Por ejemplo, existen sistemas de voto electrónico que poseen un pizarrón virtual donde se publican todos los códigos asignados a los votos de los electores. Es decir, cualquier elector puede visualizar el código de votación de otro elector; sin embargo, como este código en una secuencia de números encriptados permanece la cualidad del voto secreto.

Capítulo 7. Conclusiones y trabajos futuros

7.1 Conclusiones

En esta sección se presentan las conclusiones obtenidas al conseguir los resultados esperados de los objetivos específicos, así como las conclusiones generales del proyecto de tesis.

El primer objetivo específico fue implementar un sistema de voto electrónico de código abierto que gestione la información del proceso electoral de forma descentralizada para los actores del proceso electoral. En base a la arquitectura diseñada resultante de las fases de análisis y diseño, se consiguió implementar un sistema de voto electrónico para elecciones generales en el Perú compuesto por una capa front-end en React Js y una capa de back-end en la tecnología blockchain con un servicio de envío de correos desarrollado en Spring. Una vez escogida la arquitectura del sistema a utilizar, se definieron tres módulos que permitieron conseguir este primer objetivo específico. Estos tres módulos fueron el módulo de emisión de votos, escrutinio de los votos y mantener un repositorio del proyecto con acceso libre para la comunidad. En tal sentido, se pudo definir toda la configuración del proceso electoral y la forma de interacción con el sistema mediante billeteras electrónicas que se basan en la red de Ethereum y se detallaron el uso de las transacciones en la aplicación descentralizada que permitían mantener un control de auditoria sobre el sistema. Así mismo, el concepto de cifra repartidora fue vital para el flujo del proceso de escrutinio ya que dentro del alcance se encontraba lo que es lista de votación con candidatos múltiples aplicado a elecciones congresales. Adicionalmente, cabe mencionar que el uso de "*modifiers*" fue de gran utilidad debido a que permitía validar que billetera podía interactuar con el sistema.

Posterior a ello, el segundo objetivo específico fue implementar un algoritmo de cifrado que provea la integridad de los datos, en la cual se utilizó el algoritmo de cifrado ElGamal. Una característica muy importante para incrementar la seguridad que se quiso agregar en el sistema

propuesto. Este tipo de algoritmo es muy utilizado en sistemas de procesos electorales y posee muchas variaciones. Para el uso del presente algoritmo se tuvo que crear la existencia de un nuevo rol, el de “autoridad de escrutinio”, la cual cumple una función principal en el sistema para la encriptación y desencriptación del conteo de votos. Se concluye que el algoritmo a utilizar agrega seguridad al sistema y juntamente con el uso de la blockchain permite darle al sistema una seguridad E2E ya que se tiene un control de auditoria de movimientos en el sistema y además la información de los votos registradas en la blockchain se encuentran encriptados. Cabe mencionar que se pueden utilizar “n” autoridades de escrutinio y mientras más se incremente más robusto será la encriptación de los votos, sin embargo; está el otro punto en contra el cual es la cantidad de recursos computacional que le toma al sistema procesar el algoritmo. Este último punto, es fundamental considerarlo ya que debido a que mientras más procesamiento computacional se realice más monedas electrónicas se utiliza en el sistema y esto puede incrementar en gran manera los costos del uso del sistema.

El tercer objetivo específico fue utilizar estándares legales y técnicos en las fases de emisión, escrutinio y auditoría. En tal sentido, un resultado alcanzado fundamental fue la creación del catálogo de requerimientos, el cual constituye la base para la definición del nivel de seguridad que posee el sistema. Es por ello, que el catálogo de requerimientos sufrió tres versiones a lo largo del proyecto de tesis. Así mismo, se implementó el módulo de verificación individual de los votos y funcionalidades que permitan auditar el sistema. Con estas últimas características se pudo incrementar y finalizar el nivel de seguridad del sistema propuesto. En este objetivo más enfocado al control de cambios registrados en el sistema lo que permitían un nivel alto de auditoria del sistema por parte del elector como por parte del auditor.

Finalmente, el nivel de gobierno electrónico en el cual se implementó el sistema fue las elecciones generales para procesos electores en el Perú. En tal sentido, el alcance del proyecto es a nivel nacional; sin embargo, es factible poder implementarlo en un gobierno electrónico

distrital, donde se debería tener en consideración el lugar de residencia del elector, o en un nivel de gobierno electrónico de colegio de profesionales en el Perú en donde el uso de sistema de voto electrónico no presencial es más utilizado. Inclusive se podría agregar mayor nivel en la fase de configuración del proceso electoral para aceptar otros tipos de procesos electorales, tales como el referéndum.

7.2 Trabajos futuros

Se proponen los siguientes trabajos futuros que podrán extender las funcionalidades del sistema de voto electrónico:

- Implementar un módulo de segunda vuelta para las elecciones presidenciales
- Implementar un módulo de capacitación y tutoriales de emisión de votos, la cual es una característica obligatoria para sistemas de votación electrónica según los estándares del Consejo Europeo de procesos de voto electrónico.
- Implementar una base de datos relacional para una primera etapa del proceso electoral en la cual se puedan realizar “n” modificaciones sobre el sistema. Lo que se busca con esto, es que las modificaciones o actualizaciones de los datos en la etapa de configuración del sistema no generen costos adicionales sobre el sistema, lo cual pasa con el uso de blockchain. Posterior a ello, se pasaría a la segunda etapa donde se validaría toda la *data* que se encuentra en el sistema con el JNE y una vez aprobado se registraría en la blockchain. A partir de este punto, solo se interactuaría con la blockchain, con el fin de reducir el número de transacciones en el sistema y con ello reducir costos.

Referencias

Abolhasan, M., Wysocki, T., & Dutkiewicz, E. (2004). A review of routing protocols for mobile ad hoc networks. Ad hoc networks, 2(1), 1-22.

Abuidris, Y., Kumar, R., & Wenyong, W. (2019). A survey of blockchain based on e-voting systems. Paper presented at the ACM International Conference Proceeding Series, 99-104. Doi:10.1145/3376044.3376060 Retrieved from www.scopus.com

ABC (2019, Abril). Ir a votar se va a acabar: el <<blockchain>> permite que lo hagas con el móvil desde el salón de tu casa. https://www.abc.es/tecnologia/redes/abci-votar-acabar-blockchain-permite-hagas-movil-desde-salon-casa-201904220258_noticia.html

Agora. (2018). Agora Whitepaper. Agora, 46.

Aguerre, T. (2017). Voto electrónico : un debate entre lo seguro y lo moderno. 37–53.

Agbesi, S., & Asante, G. (2019). Electronic voting recording system based on blockchain technology. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/CMI48017.2019.8962142

Aguilar, Ricardo (2109). Logran romper el esquema de cifrado del sistema de votación ruso basado en blockchain. Auditor Informático Héctor Teodoro Hernández, P. (n.d.). VOTO ELECTRÓNICO Estándares, Seguridad y Confidencialidad.

Andina. (2020, Mayo). Coronavirus: Conoce las últimas medidas dictadas por el Gobierno. Medidas para la ampliación del distanciamiento social obligatorio.https://andina.pe/agencia/noticia-coronavirus-conoce-ultimas-medidas-dictadas-por-gobierno-796496.aspx

Atá Roghnaithe, V. L. Secrecy, Accuracy and Testing of the Chosen Electronic Voting System.

Arroyo, M. A. M. (2013). Escrutinio y recuento de votos en el ordenamiento jurídico costarricense. Revista de Derecho Electoral, (15), 4.

Braghin, C., Cimato, S., Cominesi, S. R., Damiani, E., & Mauri, L. (2019). Towards blockchain-based E-Voting Systems doi:10.1007/978-3-030-36691-9_24 Retrieved from www.scopus.com

Bulut, R., Kantarci, A., Keskin, S., & Bahtiyar, S. (2019). Blockchain-based electronic voting system for elections in turkey. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/UBMK.2019.8907102

B, S., V, R. T., Krishna M P, ,Nidhish, J, B. R., Surya, A. M., & Alagappan, D. M. (2019). Secured electronic voting system using the concepts of blockchain. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/IEMCON.2019.8936310

Calvo,M. (6 de febrero, 2019). 6 sistemas electorales que usan el potencial de la blockchain. BolckChain Services. www.blockchainservices.es/novedades/6-sistemas-electorales-que-usan-el-potencial-de-la-blockchain/

Chaieb, M., Yousfi, S., Lafourcade, P., & Robbana, R. (2018, October). Verify-your-vote: a verifiable blockchain-based online voting protocol. In European, Mediterranean, and Middle Eastern Conference on Information Systems (pp. 16-30). Springer, Cham.

Chaparro, E. A. (2015). El Sistema De Voto Electrónico De La Ciudad De Buenos Aires : Una " Solución " En Busca De.

Complementaria, D., & Ley, D. (2006). Ley Orgánica de Elecciones LEY No 26859. 26859.

Cucho Espinoza, M. A. (2014). Buenas Practicas En Voto Electrónico En America: Reflexiones y lecciones desde los estándares electrónicos internacionales. http://www.corteidh.or.cr/tablas/r34627.pdf

Del Blanco, D. Y. M. (2018). Ciberseguridad aplicada a la e-democracia: análisis criptográfico y desarrollo de una metodología práctica de evaluación para sistemas de voto electrónico remoto y su aplicación a las soluciones más relevantes (Doctoral dissertation, Universidad de León).

Dhulavvagol, P. M., Bhajantri, V. H., & Totad, S. G. (2020). Blockchain Ethereum Clients Performance Analysis Considering E-Voting Application. Procedia Computer Science, 167, 2506-2515.

Dogo, E. M., Nwulu, N. I., Olaniyi, O. M., Aigbavboa, C. O., & Nkonyana, T. (2018). Blockchain 3.0: Towards a Secure Ballotcoin Democracy through a Digitized Public Ledger in Developing Countries. I-manager's Journal on Digital Signal Processing, 6(2), 24-35.

El Comercio (2020, Junio). El voto preferencial sigue en debate: estos son los proyectos que buscan su eliminación. https://elcomercio.pe/politica/el-voto-preferencial-sigue-en-debate-estos-son-los-proyectos-que-buscan-su-eliminacion-paridad-y-alternancia-noticia/

Elecciones, J. N. D. E. (2005). Historia de los procesos electorales en el Perú Nota introductoria.

Electorales, O. (2014). Proceso Electoral. Wani Revista Del Caribe Nicaragüense, 0(23), 13–17.

Ethereum (2020) Aprende sobre Ethereum. https://ethereum.org/es/learn/

Europe., C. of. (2010). E-voting handbook: Key steps in the implementation of e-enabled elections.

Faour, N. (2019). Transparent E-voting dApp based on waves blockchain and RIDE language. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/REDUNDANCY48165.2019.9003336

Gao, S., Zheng, D., Guo, R., Jing, C., & Hu, C. (2019). An Anti-Quantum E-Voting Protocol in Blockchain With Audit Function. IEEE Access, 7, 115304-115316.

Gañán, C. H. (2009). 'Scalable Multi-Source Video Streaming Application over Peer-to-Peer Networks'. UPC, Dept. Enginyeria Telematica, September.

Garcia-font, V., & Rif, H. (n.d.). Votaciones Blockchain. 257–262.

Git (s.f.). About. https://git-scm.com/about

Goodman,N. (2017). Online Voting: A Path Forward for Federal Elections: I of Cánada. https://www.canada.ca/en/democratic-institutions/services/reports/online-voting-path-forward-federal-elections.html

Goldsmith, B. (n.d.). Electronic Voting and Counting Technologies Chapter 2 . 2 : Building the counting Lead Authors System for E-voting or E-.

Gsuite (s.f.). Marketplace. Diagramas.net. https://gsuite.google.com/marketplace/app/diagramsnet/671128082532

Hanifatunnisa, R., & Rahardjo, B. (2017, October). Blockchain based e-voting recording system design. In 2017 11th International Conference on Telecommunication Systems Services and Applications (TSSA) (pp. 1-6). IEEE.

Hjálmarsson, F. Þ., Hreiðarsson, G. K., Hamdaqa, M., & Hjálmtýsson, G. (2018, July). Blockchain-based e-voting system. In 2018 IEEE 11th International Conference on Cloud Computing (CLOUD) (pp. 983-986). IEEE.

Hernández, H. (2011). Fiscalización Informática del Voto Electrónico. Guía para la actuación profesional. Argentina. Tinta Libre Ediciones.. Guía para la actuación profesional. Argentina. Tinta Libre Ediciones. et al., 2015

IDEA Internacional. (2012). Una introducción al voto electrónico: Consideraciones esenciales.

ISACA. (2011). Data Integrity - Information Security's Poor Relation: Isaca Journal Archives. https://www.isaca.org/resources/isaca-journal/past-issues/2011/data-integrity-information-security-s-poor-relation

Jain, H., Oak, R., & Bansal, J. (2019, January). Towards Developing a Secure and Robust Solution for E-Voting using Blockchain. In 2019 International Conference on Nascent Technologies in Engineering (ICNTE) (pp. 1-6). IEEE.

JavaScript.info (2020, Febrary). An introduction to JavaScript. What is JavaScript?. https://javascript.info/intro

JNE. (2005). Proceso Electoral. https://dnef.jne.gob.pe/zonaescolar/material/7-proceso-electoral.pdf

JNE. (2005). La cifra repartidora en los procesos electorales en el Perú.

https://portal.jne.gob.pe/portal_documentos/files/informacioninstitucional/escuelaelectoral/Martes%20Electorales%20-%20Exposiciones/ee2005/cifraRepartidora.pdf

Jones. D. (2010). The Sailau E-Voting System in Direct Democracy: Progress and Pitfalls of Election Technology, IFES Election Technology Series

Jurado Nacional Electoral. (s.f). El voto preferencial. doi:https://portal.jne.gob.pe/portal_documentos/files/informacioninstitucional/escuelaelectoral/Martes%20Electorales%20-%20Exposiciones/ee2007/mar_14ago2007.pdf

Khandelwal, A. (2019). Blockchain implimentation on E-voting system. Paper presented at the Proceedings of the International Conference on Intelligent Sustainable Systems, ICISS 2019, 385-388. Doi:10.1109/ISS1.2019.8907951 Retrieved from www.scopus.com

Kitchenham, B., & Charters, S. (2016). Guidelines for performing systematic literature reviews in software engineering. Cs. Auckland. Ac. Nz. 2007.

Kohno, Tadayoshi, Adam Stubblefield, Aviel Rubin y Dan Wallach (2004). «Analysis of an Electronic Voting System», IEEE Symposium on Security and Privacy 2004. IEEE Computer Society Press.

Košt'ál, K., Bencel, R., Ries, M., & Kotuliak, I. (2019, October). Blockchain E-Voting Done Right: Privacy and Transparency with Public Blockchain. In 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS) (pp. 592-595). IEEE.

Kshetri, N., & Voas, J. (2018). Blockchain-enabled E-voting. IEEE Software, 35(4), 95-99. Doi:10.1109/MS.2018.2801546

Lai, W. J., Hsieh, Y. C., Hsueh, C. W., & Wu, J. L. (2018, August). Date: a decentralized, anonymous, and transparent e-voting system. In 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN) (pp. 24-29). IEEE.

Lauer, T. W. (2004). The Risk of e-Voting. Electronic Journal of E-Government, 2(April), 177–186. http://www.ejeg.com/volume-2/volume2-issue3/v2-i3-art4-abstract.htm

Lehoucq, F. (2007). ¿Qué es el fraude electoral? Su naturaleza, sus causas y consecuencias. Revista Mexicana de Sociología, 69(1), 1–38. https://doi.org/10.22201/iis.01882503p.2007.001.6082

Linera, M. Á. P. (2016). Premisas para la introducción del voto electrónico en la legislación electoral española. Revista de estudios políticos, (173), 277-304.

Matile, R., Rodrigues, B., Scheid, E., & Stiller, B. (2019, May). CaIV: Cast-as-Intended Verifiability in Blockchain-based Voting. In 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) (pp. 24-28). IEEE.

Mata, F. J., Matarrita, R., & Pinto, C. (2012). Assessing computer education in Costa Rica: Results of a supply and demand study of ICT human resources. CLEI Electronic Journal, 15(1), 6-6.

Mendívil, I. (n.d.). El ABC de los Documentos Electrónicos Seguros.

Miethereum (s.f.). Solidity. Smart contracts. https://www.miethereum.com/smart-contracts/solidity/#toc1

Montes, M., Penazzi, D., & Wolovick, N. (2016). Consideraciones Sobre El Voto Electrónico. 10° Simposio Sobre Informática En El Estado, 297–307. http://sedici.unlp.edu.ar/bitstream/handle/10915/58355/Documento_completo.PDF-PDFA.pdf?sequence=1&isAllowed=y%0Ahttp://45jaiio.sadio.org.ar/sites/default/files/SIE-27.PDF

MSc, I. F. (2011). Gestión de la Planificación de los Riesgos del Proyecto. Doi:http://www.ucipfg.com/Repositorio/MAP/MAPD-10/BLOQUE-ACADEMICO/Unidad-2/Gestion_de_la_Planificacion_de_los_Riesgos_del_Proyecto_Tema-04.pdf

National Democracy Institute . (december 17, 2013). Transparency. https://www.ndi.org/e-voting-guide/transparency

Nardi, J. L., Lopapa, A., Zitelli, L., & Vázquez, A. (2017). Análisis de Riesgos , Vulnerabilidades y Propuestas de Auditoría sobre Sistemas de Voto Electrónico. November.

Navarrete, M., Huancas, R., Diaz, P., & Rivadeneira, M. (2019). Blockchain electronic vote system. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/CHILECON47746.2019.8988084

Nestor,C. (2019). El alto porcentaje de voto por correo despertó las sospechas de fraude electoral en Albaida del Aljarafe. Eldiario.es. https://www.eldiario.es/andalucia/sevilla/Adelante-Andalucia-Albaida-Aljarafe-porcentaje_0_956105034.html

NodeJs (s.f.). Acerca de NodeJs. https://nodejs.org/es/about/

ONPE (2011). Elecciones de congresistas de la República. https://www.web.onpe.gob.pe/modElecciones/Procesos/EPA-2011/Informacion/materiales/Gu%C3%ADa_Congreso.pdf

ONPE (2013). Voto electrónico No Presencial. Aproximaciones desde las experiencias internacionales y el caso peruano. [Archivo PDF]. https://www.web.onpe.gob.pe/modEducacion/Publicaciones/L-0087.pdf

ONPE (2017). ¿Qué es el voto electrónico no presencial?:Oficina Nacional de Procesos Electorales. https://www.web.onpe.gob.pe/modAsistenciaTecnica/elecciones-ciudadanas/instituciones/CAL2017/que-es-venp

Pandey, A., Bhasi, M., & Chandrasekaran, K. (2019). VoteChain: A blockchain based E-voting system. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/GCAT47503.2019.8978295

Pacheco, S. L. (2015). Hacia el voto electrónico en la práctica electoral mexiquense: consideraciones elementales. Apuntes Electorales, 14(52), 51–81.

Panizo Alonso, L. (2007). Aspectos tecnológicos del voto electrónico. 60. https://doi.org/9972-695-11-5

Perez, A. J., & Ceesay, E. N. (2018, July). Improving End-to-End Verifiable Voting Systems with Blockchain Technologies. In 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (pp. 1108-1115). IEEE.

Polys (s.f.). Polys features https://polys.me/

Process, E., & Commission, A. (n.d.). Voting with confidence. September 2007.

Puiggali, J., & Morales-Rocha, V. (2007, October). Remote voting schemes: a comparative analysis. In International Conference on E-Voting and Identity (pp. 16-28). Springer, Berlin, Heidelberg.

Quirós,F. (26 de abril, 2019). Empresa española crea plataforma basada en blockchain para garantizar el voto seguro. Cointelegraph en español. https://es.cointelegraph.com/news/spanish-company-creates-platform-based-on-blockchain-to-guarantee-secure-vote

RAE. (2019). Definición de votar. https://del.rae.es/votar

React (s.f.). Documentación. https://es.reactjs.org/docs/getting-started.html

Remmert, M. (2004). Towards European Standards on Electronic Voting. Proceedings of the 1st Conference on Electronic Voting, 13–16.

Rial, J.(2004) Posibilidades y límites de voto electrónico. https://www.web.onpe.gob.pe/modEducacion/Publicaciones/L-0026.pdf#page=77

Rijo, R.(17 de febrero,2020). Muchos países decidieron abandonar el voto electrónico.ElCaribe. https://www.elcaribe.com.do/2020/02/17/muchos-paises-decidieron-abandonar-el-voto-electronico/#

Santoro del Campo, A. (2006). Voto electrónico. In Rev (Issue 19).

SCRUM Guide. (s.f.). What is SCRUM? https://www.scrumguides.org/

Sheer Hardwick, F., Gioulis, A., Naeem Akram, R., & Markantonakis, K. (2018). E-Voting with Blockchain: An E-Voting Protocol with Decentralisation and Voter Privacy. arXiv preprint arXiv:1805.10258.

Solidity (2020). Solidityv0.6.10. Language documentation. https://solidity.readthedocs.io/en/v0.6.10/

Springall, D., Finkenauer, T., Durumeric, Z., Kitcat, J., Hursti, H., MacAlpine, M., & Halderman, J. A. (2014). Security analysis of the estonian internet voting system. Proceedings of the ACM Conference on Computer and

Communications Security, May, 703–715. https://doi.org/10.1145/2660267.2660315

Tenorio, H. A. (2004). Elecciones. Chasqui, 11(1), 107. https://doi.org/10.2307/29739741

Torres, A. (2015). Algunas reflexiones sobre el voto electrónico. Diario La Nación, 11. https://www.lanacion.com.ar/tecnologia/algunas-reflexiones-sobre-el-voto-electronico-nid1809389

Truffle Suite (s.f.) Documentación. https://www.trufflesuite.com/docs

Tubella, I. & Jordi V. (2005). Sociedad del conocimiento. Cómo cambia el mundo ante nuestros ojos. Barcelona: Editorial uoc.

UML (s.f.). What is UML?. https://www.uml.org/what-is-uml.htm

U.S. Vote Foundation (2015, July). The futuro of voting. End-to-end verifiable voting. Galois.

Vähä-Sipilä, A. (2009). A Report on the Finnish E-Voting Pilot. November, 1–16. https://www.verifiedvoting.org/wp-content/uploads/2014/09/Finland-2008-EFFI-Report.pdf

Vijayalakshmi, V., & Vimal, S. (2019, March). A Novel P2P based System with Blockchain for Secured Voting Scheme. In 2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM) (Vol. 1, pp. 153-156). IEEE.

Vilamala, R., & Josep, M. (2008). Ocho dudas razonables sobre la necesidad del voto electrónico. Idp, 1(6), 1–1.

Voatz(s.f). What is thisn internet voting?.Voatz. https://voatz.com/faq.html

Voto electronico por internet y riesgos para la democracia (I).pdf. (n.d.).

Vo-Cao-Thuy, L., Cao-Minh, K., Dang-Le-Bao, C., & Nguyen, T. A. (2019, March). Votereum: An Ethereum-Based E-Voting System. In 2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF) (pp. 1-6). IEEE.

Wang, K.-H., Mondal, S. K., Chan, K., & Xie, X. (2017). A Review of Contemporary E-voting: Requirements, Technology, Systems and Usability. Ubiquitous International, 1(1), 31–47. http://www.ikelab.net/dspr-pdf/vol1-1/dspr-paper3.pdf

Xukai,Z. (2017). Transparent, Auditable, and Stepwise Verifiable Online E-Voting Enabling an Open and Fair Election

Yi, H. (2019). Securing e-voting based on blockchain in P2P network. EURASIP Journal on Wireless Communications and Networking, 2019(1), 1-9.

Zaghloul, E., Li, T., & Ren, J. (2020, February). Anonymous and Coercion-Resistant Distributed Electronic Voting. In 2020 International Conference on Computing, Networking and Communications (ICNC) (pp. 389-393). IEEE.

Zhang, S., Wang, L., & Xiong, H. (2019). Chaintegrity: Blockchain-enabled large-scale e-voting system with robustness and universal verifiability. International Journal of Information Security, doi:10.1007/s10207-019-00465-8

&NA; (1955). Interim Report. AJN, American Journal of Nursing, 55(9), 1125. https://doi.org/10.1097/00000446-195509000-00033

Anexos

Anexo A: Listado de estudios primarios

Abuidris, Y., Kumar, R., & Wenyong, W. (2019). A survey of blockchain based on e-voting systems. Paper presented at the ACM International Conference Proceeding Series, 99-104. Doi:10.1145/3376044.3376060 Retrieved from www.scopus.com

Agbesi, S., & Asante, G. (2019). Electronic voting recording system based on blockchain technology. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/CMI48017.2019.8962142

B, S., V, R. T., Krishna M P, ,Nidhish, J, B. R., Surya, A. M., & Alagappan, D. M. (2019). Secured electronic voting system using the concepts of blockchain. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/IEMCON.2019.8936310

Braghin, C., Cimato, S., Cominesi, S. R., Damiani, E., & Mauri, L. (2019). Towards blockchain-based E-Voting Systems doi:10.1007/978-3-030-36691-9_24 Retrieved from www.scopus.com

Bulut, R., Kantarci, A., Keskin, S., & Bahtiyar, S. (2019). Blockchain-based electronic voting system for elections in turkey. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/UBMK.2019.8907102

Dogo, E. M., Nwulu, N. I., Olaniyi, O. M., Aigbavboa, C. O., & Nkonyana, T. (2018). Blockchain 3.0: Towards a Secure Ballotcoin Democracy through a Digitized Public Ledger in Developing Countries. I-manager's Journal on Digital Signal Processing, 6(2), 24-35.

Dhulavvagol, P. M., Bhajantri, V. H., & Totad, S. G. (2020). Blockchain Ethereum Clients Performance Analysis Considering E-Voting Application. Procedia Computer Science, 167, 2506-2515.

Faour, N. (2019). Transparent E-voting dApp based on waves blockchain and RIDE language. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/REDUNDANCY48165.2019.9003336

Gao, S., Zheng, D., Guo, R., Jing, C., & Hu, C. (2019). An Anti-Quantum E-Voting Protocol in Blockchain With Audit Function. IEEE Access, 7, 115304-115316.

Jain, H., Oak, R., & Bansal, J. (2019, January). Towards Developing a Secure and Robust Solution for E-Voting using Blockchain. In 2019 International Conference on Nascent Technologies in Engineering (ICNTE) (pp. 1-6). IEEE.

Jabbar, I., & Alsaad, S. N. (2017). Design and Implementation of Secure Remote e-Voting System Using Homomorphic Encryption. IJ Network Security, 19(5), 694-703.

Khandelwal, A. (2019). Blockchain implimentation on E-voting system. Paper presented at the Proceedings of the International Conference on Intelligent Sustainable Systems, ICISS 2019, 385-388. Doi:10.1109/ISS1.2019.8907951 Retrieved from www.scopus.com

Kshetri, N., & Voas, J. (2018). Blockchain-enabled E-voting. IEEE Software, 35(4), 95-99. Doi:10.1109/MS.2018.2801546

Košťál, K., Bencel, R., Ries, M., & Kotuliak, I. (2019, October). Blockchain E-Voting Done Right: Privacy and Transparency with Public Blockchain. In 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS) (pp. 592-595). IEEE.

Lai, W. J., Hsieh, Y. C., Hsueh, C. W., & Wu, J. L. (2018, August). Date: a decentralized, anonymous, and transparent e-voting system. In 2018 1st IEEE International Conference on Hot Information-Centric Networking (HotICN) (pp. 24-29). IEEE.

Matile, R., Rodrigues, B., Scheid, E., & Stiller, B. (2019, May). CaIV: Cast-as-Intended Verifiability in Blockchain-based Voting. In 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC) (pp. 24-28). IEEE.

Navarrete, M., Huancas, R., Diaz, P., & Rivadeneira, M. (2019). Blockchain electronic vote system. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/CHILECON47746.2019.8988084

Pandey, A., Bhasi, M., & Chandrasekaran, K. (2019). VoteChain: A blockchain based E-voting system. Piscataway: The Institute of Electrical and Electronics Engineers, Inc. (IEEE). Doi:http://dx.doi.org.ezproxybib.pucp.edu.pe:2048/10.1109/GCAT47503.2019.8978295

Perez, A. J., & Ceesay, E. N. (2018, July). Improving End-to-End Verifiable Voting Systems with Blockchain Technologies. In 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) (pp. 1108-1115). IEEE.

Sheer Hardwick, F., Gioulis, A., Naeem Akram, R., & Markantonakis, K. (2018). E-Voting with Blockchain: An E-Voting Protocol with Decentralisation and Voter Privacy. arXiv preprint arXiv:1805.10258.

Tecnologias de Información. Integridad de datos. [Figura]. Recuperado de: https://www.tecnologias-informacion.com/integridaddatos.html

Vijayalakshmi, V., & Vimal, S. (2019, March). A Novel P2P based System with Blockchain for Secured Voting Scheme. In 2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM) (Vol. 1, pp. 153-156). IEEE.

Vo-Cao-Thuy, L., Cao-Minh, K., Dang-Le-Bao, C., & Nguyen, T. A. (2019, March). Votereum: An Ethereum-Based E-Voting System. In 2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF) (pp. 1-6). IEEE.

Yi, H. (2019). Securing e-voting based on blockchain in P2P network. EURASIP Journal on Wireless Communications and Networking, 2019(1), 1-9.

Zhang, S., Wang, L., & Xiong, H. (2019). Chaintegrity: Blockchain-enabled large-scale e-voting system with robustness and universal verifiability. International Journal of Information Security, doi:10.1007/s10207-019-00465-8

Zaghloul, E., Li, T., & Ren, J. (2020, February). Anonymous and Coercion-Resistant Distributed Electronic Voting. In 2020 International Conference on Computing, Networking and Communications (ICNC) (pp. 389-393). IEEE.

Anexo B: Plan de Proyecto

- **Justificación**

El presente proyecto de fin de carrera tiene como propósito desarrollar un sistema de voto electrónico basado en blockchain, el cual es una alternativa a los sistemas existentes. El presente proyecto es una alternativa que brinde transparencia y robustez en las fases de preparación, registro, votación, emisión de voto, escrutinio y auditoría siguiendo estándares legales y técnicos para sistemas de voto electrónico brindados por el Consejo Europeo. Así mismo, la implementación del presente sistema puede contribuir a incrementar la confianza en las TICs para procesos electorales. También se propone que esta solución sea de código libre para que cualquier institución pueda acceder a ella y utilizarlo como herramienta democrática. Por otro lado, la unidad de estudio sobre la cual se validará la aplicación del sistema será las elecciones generales del Perú, la cual permite crear procesos electorales para elecciones de presidente, vicepresidente, congresistas de la república y el parlamento andino.

Adicionalmente, cabe resaltar que debido a la coyuntura actual hay una serie de restricciones emitidas por el gobierno, entre las cuales resalta la prohibición de eventos o actividades que generen aglomeración de personas (Andina, 2020). En tal sentido, la presente solución es una alternativa que facilita a todas las instituciones que necesiten realizar eventos de votación y no cuenten con las herramientas tecnológicas para realizarlo de manera remota.

- **Viabilidad**

Para el análisis de viabilidad se han considerado tres factores:

- **Viabilidad Económica:** Todas las herramientas planteadas en el capítulo 1.3 del presente documento, las cuales son indispensables para la implementación del proyecto, son gratuitas e inclusive algunas son de código abierto.

- **Viabilidad Técnica:** Se posee conocimiento y experiencia recopilada durante la carrera acerca de programación web mediante el lenguaje de programación JavaScript. En cuanto a la tecnología blockchain Ethereum apareció en el 2015 y no se posee amplios conocimientos; sin embargo, el lenguaje de programación que utiliza es muy parecido al de JavaScript y cuenta con gran cantidad de documentación lo cual disminuye la curva de aprendizaje. Así mismo, respecto a las pruebas que son necesarias la tecnología Blockchain presenta medios (comandos) por el cual se pueden realizar pruebas a su red. Adicionalmente, para el proceso de implementación existen diversas herramientas que permiten su desarrollo y prueba en una blockchain local del computador.
- **Viabilidad Temporal:** Basándose en el plan de proyecto y el cronograma de actividades se establece que el proyecto se culminará en diciembre del 2020.

- **Alcance**

El objetivo del presente proyecto es la implementación de un sistema de voto electrónico. Al igual que los sistemas de voto electrónico que existen en el mercado, esta solución permite usuarios con dos diferentes roles: Elector y Administrador. El proyecto se basará en estándares técnicos y legales para votación electrónica presentados por el Consejo Europeo y la ley Orgánica de elecciones N° 26 859.

Así mismo, el sistema abordará todas las fases de un voto electrónico: preparación, registro de electores, votación, emisión del voto, escrutinio y auditoría. En tal sentido, el sistema se implementará para que sea funcional tanto a nivel de gobierno de elecciones generales como elecciones locales (como elecciones distritales) y a su vez este será validado por expertos y por el asesor de tesis. Por ende, el resultado del presente proyecto cuenta con los siguientes módulos:

- **Módulo de preparación del proceso electoral:** Este módulo permite iniciar el proceso de votación electrónica. Con ello, se abordará el subproceso de “diseño de voto”. En el proceso de diseño se definirá el tipo de voto que se realizará, debido a que el sistema permite dos tipos de votos: voto único y voto múltiple el cual está enfocado a la elección de congresistas y el parlamento andino.

- **Módulo de registro de electores:** Este módulo verifica el registro de los electores al sistema de voto electrónico. Se recibirá una lista de electores y se registrará al elector admitido. La fase concluye con la notificación del registro satisfactorio del elector en el sistema vía correo electrónico.

- **Módulo de emisión de votos:** Este módulo abarca las fases de votación y emisión de voto de un proceso de votación electrónica. El módulo permite al elector registrar el (los) candidato(s) seleccionados. Así mismo, antes de emitir el voto, se muestran un resumen de las selecciones del elector para que el elector verifique su voto. Posterior a ello, el módulo registrará el candidato seleccionado por el elector en la blockchain.

- **Módulo de cifrado:** Este módulo pertenece a la fase de emisión de votos por lo cual es activado por el módulo de emisión de votos y se encarga de encriptar cada voto mediante el cifrado ElGamal.

- **Módulo de verificación individual:** Este módulo genera un código de verificación el cual se le envía por correo electrónico al elector con el fin de que este pueda cumplir la propiedad de verificación individual del elector.

- **Módulo de escrutinio:** Este módulo pertenece a la fase del escrutinio de los votos. Así mismo, este módulo realizará el conteo de los votos y los publicará en la blockchain. El presente módulo no brindará información sobre el conteo parcial de votos, sino hasta acabar el proceso de votación con el fin de cumplir la característica de justicia de un voto.
- **Módulo de auditoría:** Este módulo pertenece a la fase de auditoría. Así mismo, este módulo brinda los medios para que exista verificación universal del sistema por parte de cualquier entidad interesada en el sistema. Esto lo realizará mediante el acceso a la blockchain y obtener información que brinde trazabilidad de los bloques.

Adicionalmente el sistema a desarrollar tiene como objetivo específico permitir la accesibilidad a su código por lo que se diseñará como un software de código abierto. Finalmente, para la presente solución se asumirá que el elector usará un dispositivo seguro libre de virus para la fase de emisión de voto. Así mismo, el sistema no permitirá la modificación del voto.

- **Restricciones**

  Se procederá a especificar las limitaciones del presente proyecto de tesis.

  - La participación de expertos para validar los resultados del proyecto.
  - Las pruebas de cada módulo, así como las pruebas integrales solo se realizarán en entornos locales que simulan una red de blockchain.
  - Debido a que el sistema se piensa ejecutar en la red blockchain de Ethereum se necesita utilizar determinada cantidad de gas para poder desplegar el contrato inteligente.

- **Identificación de los riesgos del proyecto**

  - **Escala de medición de riesgos:**

Tabla 17. Escala de probabilidad

Escala de probabilidad				
Calificación	1.1 – 0.3	0.4 – 0.6	0.7 – 0.9	1
Interpretación	Baja	Media	Alta	Hecho

Tabla 18. Escala de impacto

Escala de impacto	
Calificación	**Interpretación**
1	Fracaso del proyecto
0.9	Proyecto retrasado en 40%
0.8	Proyecto retrasado entre 30% y 40%
0.7	Proyecto retrasado entre 20% y 30%
0.6	Proyecto retrasado entre 10% y 20%
0.5	Proyecto retrasado entre 1% y 10%
0.4	Reducción importante de las reservas de tiempo y costo
0.3	Reducción media de las reservas de tiempo y costo
0.2	Reducción pequeña de las reservas de tiempo y costo
0.1	Muy poco impacto

Tabla 19. Escala de riesgo

Escala de riesgo	
Color	**Interpretación**
Condición Verde	Riesgo bajo
Condición Amarillo	Riesgo moderado
Condición Rojo	Riesgo alto

Tabla 19. Matriz Probabilidad x Impacto

	Riesgo = Probabilidad x Impacto										
Probabi	1	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
	0.9	0.09	0.18	0.27	0.36	0.45	0.54	0.63	0.72	0.81	0.90
	0.8	0.08	0.16	0.24	0.32	0.40	0.48	0.56	0.64	0.72	0.80
	0.7	0.07	0.14	0.21	0.28	0.35	0.42	0.49	0.56	0.63	0.70

	0.6	0.06	0.12	0.18	0.24	0.30	0.36	0.42	0.48	0.54	0.60
	0.5	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
	0.4	0.04	0.08	0.12	0.16	0.20	0.24	0.28	0.32	0.36	0.40
	0.3	0.03	0.06	0.09	0.12	0.15	0.18	0.21	0.24	0.27	0.30
	0.2	0.02	0.04	0.06	0.08	0.10	0.12	0.14	0.16	0.18	0.20
	0.1	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.10
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
		Impacto									

- **Riesgos identificados**

Los riesgos identificados se han agrupado en 5 categorías:

A. Elaboración de la planificación

B. Organización y Gestión

C. Ambiente y estructura de desarrollo

D. Requisitos

E. Diseño e implementación

F. Personal

Tabla 20. Identificación de riesgos del proyecto.

ID	Riesgo	Causas
A. Elaboración de la planificación		
R01	Mala estimación de los tiempos planificados para cada actividad	- No se consideró el tiempo de la curva de aprendizaje. - No se consideró el tiempo requerido para cada tarea.
R02	La planificación del cronograma no incluye las tareas necesarias	- El alcance del proyecto no está correctamente definido.
R03	Cambio de la fecha final del entregable	- El comité de tesis modifica el cronograma del curso.
R04	Retraso en el cumplimiento de algunas tareas puede generar retraso en las siguientes	- Estimación equivocada de costos y esfuerzos. - Inadecuado levantamiento de necesidades. - Errores en el diseño de la solución.
B. Organización y Gestión		
R05	Las tareas encargadas a terceros, como la validación de expertos, necesitan más tiempo del esperado	- Expertos identificados no disponibles - No se agendó el tiempo de los expertos

C. Ambiente y estructura de desarrollo		
R06	Los espacios están disponibles, pero no son los adecuados	- Falla del internet, energía eléctrica - Mal funcionamiento del computador usado.
R07	Alta curva de aprendizaje	- Aprender una nueva tecnología - No hubo consideración de la curva de aprendizaje cuando se desarrolló el cronograma del proyecto - Falta de documentación de las tecnologías a usar
D. Requisitos		
R08	Modificación del alcance	- Mala gestión de tiempo - Inadecuada definición del alcance inicial.
E. Diseño e Implementación		
R09	Diseño simple no cubre las cuestiones principales	- Falta de validación del diseño con los requisitos. - Mala interpretación de la lectura de los requisitos
R10	Diseño complejo que exige contar con complicaciones innecesarias e improductivas en la implementación	- Falta de validación del diseño con los requisitos. - Mala interpretación de la lectura de los requisitos
F. Personal		
R11	Caída en la productividad del desarrollo del proyecto por problemas de salud.	- Fatiga - Emergencias médicas y/o familiares

- **Niveles de severidad y acciones a tomar**

Tabla 21. Niveles de severidad y acciones a tomar.

ID	Probabilidad	Impacto	Severidad	Plan de mitigación	Plan de contingencia
R01	0.2	0.4	0.08	- Realizar un correcto cronograma del proyecto	- Reajustar el cronograma para las actividades posteriores e incrementar

					el tiempo de dedicación al proyecto.
R02	0.2	0.7	0.14	- Definir correctamente el alcance del proyecto.	- Reajustar el cronograma para las actividades posteriores.
R03	0.1	0.8	0.08	- Mantener una comunicación fluida con el comité de tesis para estar informado de las actualizaciones.	- Modificar el plazo de las actividades por realizar en el cronograma.
R04	0.3	0.6	0.18	- Dedicar el tiempo suficiente para realizar una correcta estimación de costos y esfuerzos. - Levantar correctamente un catálogo de requisitos que cumpla todo el alcance definido.	- Pactar nuevas fechas de entrega mediante reuniones virtuales. - Actualizar el cronograma del proyecto.
R05	0.5	0.6	0.30	- Identificar expertos con anticipación. - Contar con una lista de expertos adicionales en caso uno de los escogidos no esté disponible.	- Pactar fechas de reuniones virtuales.
R06	0.3	0.8	0.24	- Contar con instalaciones alternativas en caso haya falla del internet y/o energía eléctrica - Contar con otro computador como respaldo	- Utilizar instalaciones alternativas para continuar con el desarrollo de las actividades definidas.
R07	0.5	0.5	0.25	- Conseguir capacitación con anticipación como cursos libres.	- Ampliar las horas de capacitación y reajustar los tiempos de desarrollo.
R08	0.1	0.8	0.08	- Mala gestión de tiempo - Inadecuada definición del alcance inicial.	- Pactar reunión para negociar el alcance del proyecto.
R09	0.3	0.7	0.21	- Realizar una correcta interpretación de los requisitos. - Realizar una correcta validación del diseño con los requisitos.	- Pactar reunión para negociar el alcance del proyecto.
R10	0.3	0.7	0.21	- Realizar una correcta interpretación de los requisitos. - Realizar una correcta validación del diseño con los requisitos.	- Pactar reunión para negociar el alcance del proyecto.

R11	0.5	0.4	0.20	- Estimar un tiempo de holgura en caso se materialicen incidentes que no permitan realizar el proyecto con normalidad.	- Reprogramar fechas de entregas considerando la holgura que se planificó.

- **Estructura de descomposición del trabajo (EDT)**
  - **Diagrama de EDT**

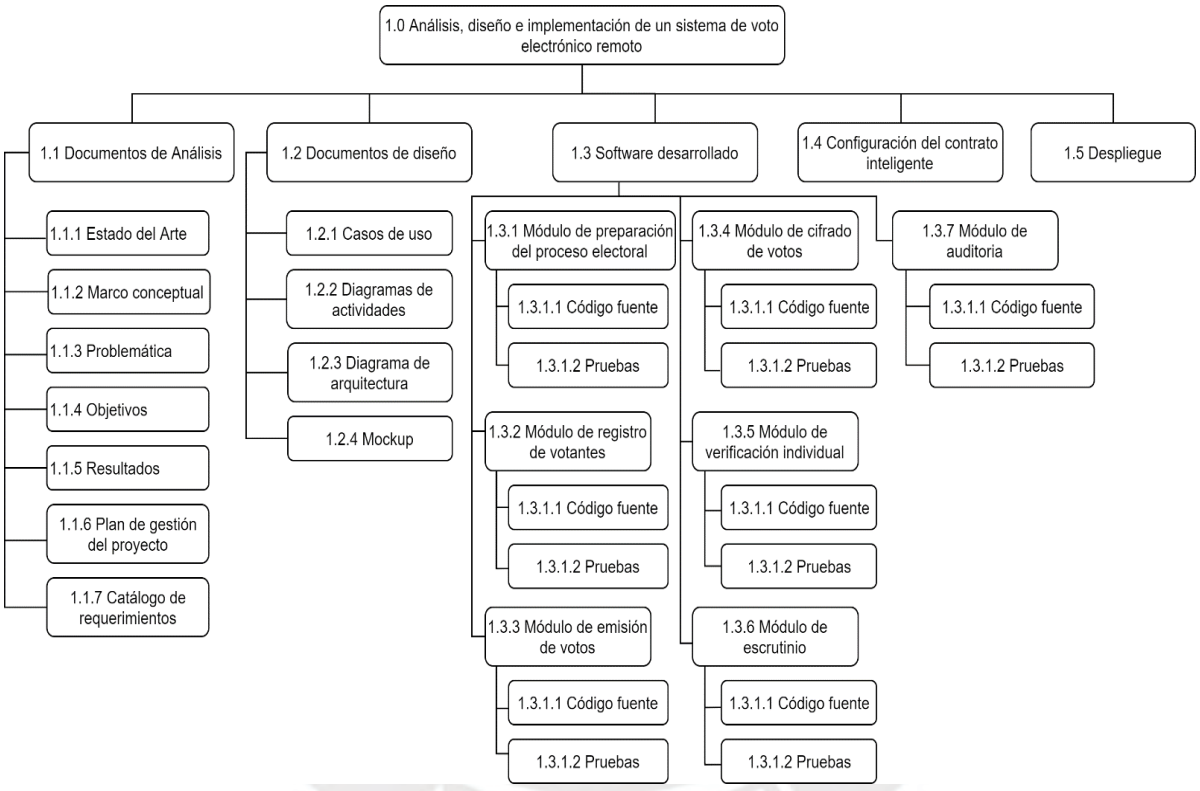


Figura 28. Estructura de descomposición del trabajo para el proyecto. (Elaboración propia)

  - **Descripción del EDT**

A continuación, se detalla los componentes del EDT mediante la descripción de cada paquete, documentos a entregar y el criterio de aceptación.

Componente	Descripción
Código del paquete de trabajo	EDT1110
Descripción del paquete de trabajo	Realizar una revisión sistemática sobre votación electrónica

Entregable(s)	- Formulario de extracción de datos - Capítulo 3: Estado del arte

Componente	Descripción
Código del paquete de trabajo	EDT1120
Descripción del paquete de trabajo	Marco conceptual y legal de los términos indispensables para entender la problemática y el estado del arte.
Entregable(s)	- Capítulo 2: Marco legal y Conceptual

Componente	Descripción
Código del paquete de trabajo	EDT1130
Descripción del paquete de trabajo	Desarrollo de la descripción de la problemática que el proyecto piensa resolver.
Entregable(s)	- Capítulo 1: Problemática

Componente	Descripción
Código del paquete de trabajo	EDT1140
Descripción del paquete de trabajo	Descripción de los objetivos, tanto general como objetivos específicos, que se plantea conseguir en el proyecto.
Entregable(s)	- Capítulos 1.2.1 y 1.2.2

Componente	Descripción
Código del paquete de trabajo	EDT1150
Descripción del paquete de trabajo	Descripción de los resultados esperados del proyecto, así como, su verificación mediante indicadores.
Entregable(s)	- Capítulos 1.2.3 y 1.2.4

Componente	Descripción
Código del paquete de trabajo	EDT1160
Descripción del paquete de trabajo	Elaboración del plan del proyecto. Este incluye la justificación, viabilidad y alcance del proyecto.
Entregable(s)	- Plan del proyecto - Gestión de riesgos - EDT - Cronograma del proyecto - Lista de recursos - Costeo del proyecto

Componente	Descripción
Código del paquete de trabajo	EDT1170
Descripción del paquete de trabajo	Lista del catálogo de requerimientos funcionales y no funcionales del proyecto
Entregable(s)	- Lista de requerimientos funcionales - Lista de requerimientos no funcionales

Componente	Descripción
Código del paquete de trabajo	EDT1210
Descripción del paquete de trabajo	Elaboración de los casos de uso para la implementación del sistema
Entregable(s)	- Casos de uso

Componente	Descripción
Código del paquete de trabajo	EDT1220
Descripción del paquete de trabajo	Elaboración de diagrama de actividades para la implementación del sistema
Entregable(s)	- Diagrama de actividades

Componente	Descripción
Código del paquete de trabajo	EDT1230
Descripción del paquete de trabajo	Elaboración de diagrama de arquitectura para la implementación del sistema
Entregable(s)	- Diagrama de arquitectura

Componente	Descripción
Código del paquete de trabajo	EDT1240
Descripción del paquete de trabajo	Elaboración del mockup del sistema a implementar
Entregable(s)	- Mockups

Componente	Descripción
Código del paquete de trabajo	EDT1311
Descripción del paquete de trabajo	Desarrollo e implementación del módulo de preparación del proceso electoral. Así mismo, se actualizará el repositorio online para incluir el siguiente módulo de preparación del proceso electoral.
Entregable(s)	- Código fuente del módulo de preparación del proceso electoral - Actualización del repositorio en GitLab

Componente	Descripción
Código del paquete de trabajo	EDT132
Descripción del paquete de trabajo	Se realizarán las pruebas correspondientes al módulo de preparación del proceso electoral.
Entregable(s)	- Reporte de pruebas

Componente	Descripción
Código del paquete de trabajo	EDT1321
Descripción del paquete de trabajo	Desarrollo e implementación del módulo de registro de electores. Así mismo, se actualizará el repositorio online para incluir el módulo de registro de electores.
Entregable(s)	- Código fuente del módulo de registro de electores. - Actualización del repositorio en GitLab

Componente	Descripción
Código del paquete de trabajo	EDT1322
Descripción del paquete de trabajo	Se realizarán las pruebas correspondientes al módulo de registro de electores.
Entregable(s)	- Reporte de pruebas

Componente	Descripción
Código del paquete de trabajo	EDT1331
Descripción del paquete de trabajo	Desarrollo e implementación del módulo de emisión de votos. Así mismo, se actualizará el repositorio online para incluir el módulo de emisión de votos.
Entregable(s)	- Código fuente del módulo de emisión de votos - Actualización del repositorio en GitLab

Componente	Descripción
Código del paquete de trabajo	EDT1332
Descripción del paquete de trabajo	Se realizarán las pruebas correspondientes al módulo de emisión de votos.
Entregable(s)	- Reporte de pruebas

Componente	Descripción
Código del paquete de trabajo	EDT1341
Descripción del paquete de trabajo	Desarrollo e implementación del módulo de cifrado de votos. Así mismo, se actualizará el

	repositorio online para incluir el módulo de cifrado de votos.
Entregable(s)	- Código fuente del módulo de cifrado de votos - Actualización del repositorio en GitLab

Componente	Descripción
Código del paquete de trabajo	EDT1342
Descripción del paquete de trabajo	Se realizarán las pruebas correspondientes al módulo de cifrado de votos.
Entregable(s)	- Reporte de pruebas

Componente	Descripción
Código del paquete de trabajo	EDT1351
Descripción del paquete de trabajo	Desarrollo e implementación del módulo de verificación individual. Así mismo, se actualizará el repositorio online para incluir el módulo de verificación individual.
Entregable(s)	- Código fuente del módulo de verificación individual - Actualización del repositorio en GitLab

Componente	Descripción
Código del paquete de trabajo	EDT1352
Descripción del paquete de trabajo	Se realizarán las pruebas correspondientes al módulo de verificación individual.
Entregable(s)	- Reporte de pruebas

Componente	Descripción
Código del paquete de trabajo	EDT1361
Descripción del paquete de trabajo	Desarrollo e implementación del módulo de escrutinio. Así mismo, se actualizará el repositorio online para incluir el siguiente módulo.
Entregable(s)	- Código fuente del módulo de escrutinio - Actualización del repositorio en GitLab

Componente	Descripción
Código del paquete de trabajo	EDT1362
Descripción del paquete de trabajo	Se realizarán las pruebas correspondientes al módulo de escrutinio.
Entregable(s)	- Reporte de pruebas

Componente	Descripción

Código del paquete de trabajo	EDT1371
Descripción del paquete de trabajo	Desarrollo e implementación del módulo de auditoría. Así mismo, se actualizará el repositorio online para incluir el siguiente módulo.
Entregable(s)	- Código fuente del módulo de auditoría - Actualización del repositorio en GitLab

Componente	Descripción
Código del paquete de trabajo	EDT1372
Descripción del paquete de trabajo	Se realizarán las pruebas correspondientes al módulo de auditoria.
Entregable(s)	- Reporte de pruebas

Componente	Descripción
Código del paquete de trabajo	EDT1400
Descripción del paquete de trabajo	Desarrollo de la configuración del contrato inteligente para interactuar con la blockchain. Así mismo, se actualizará el repositorio online para incluir el contrato inteligente.
Entregable(s)	- Archivo fuente del contrato inteligente

Componente	Descripción
Código del paquete de trabajo	EDT1500
Descripción del paquete de trabajo	Despliegue del proyecto final y reporte del despliegue. Así mismo, contieneel repositorio actualizado con todos los módulos implementados.
Entregable(s)	- Software completo - Reporte del despliegue - Repositorio con el código fuente

- **Lista de tareas**

ID	Nombre	Duración estimada (Horas)	Esfuerzo estimado (Horas-persona)	Costo estimado (PEN)
	Proyecto de Tesis	**376,40**	**393,90**	**6092,20**
	1.1 Documentos de análisis	**122,50**	**133,50**	**2251,50**
	1.1.1 Estado del arte	**46**	**47**	**838,00**
T01	- Investigar en motores de base de datos	20	20	260,00

T02	- Crear el formulario de extracción de datos	24	24	312,00
T03	- Reunión con el asesor y profesores del curso	2	3	266,00
	1.1.2 Marco conceptual	**20**	**20**	**260,00**
T04	- Investigar conceptos en internet	16	16	208,00
T05	- Leer leyes orgánicas y metodologías	4	4	52,00
	1.1.3 Problemática	**26**	**30**	**506**
T06	- Investigar la problemática en diversos artículos	16	16	208
T07	- Elaborar árbol de problemas	3	3	39
T08	- Reunión con el asesor y profesores del curso	2	6	194
T09	- Desarrollar la descripción de la problemática	5	5	65
	1.1.4 Objetivos	**6**	**8**	**136**
T10	- Definir el objetivo general del proyecto	2	2	24
T11	- Definir los objetivos específicos	3	3	39
T12	- Reunión con el asesor y profesores del curso	1	3	73
	1.1.5 Resultados	**6**	**7**	**138**
T13	- Definir los resultados esperados	2	2	26
T14	- Crear el mapeo de objetivos, resultados y verificación	3	3	39
T15	- Reunión con el asesor y profesores del curso	1	2	73
	1.1.6 Plan de proyecto	**13,5**	**14,5**	**235,5**
T16	- Elaborar el alcance del proyecto	1	1	13
T17	- Definir la justificación y el objetivo del proyecto	1	1	13
T18	- Definir las restricciones del proyecto	1	1	13
T19	- Definir la viabilidad del proyecto	1	1	13
T20	- Elaborar el plan de gestión de riesgos	1	1	13
T21	- Elaborar el EDT	2,5	2,5	32,5
T22	- Elaborar el cronograma del proyecto	2	2	26
T23	- Elaborar la lista de actividades	1,5	1,5	19,5
T24	- Elaborar la lista de recursos	1	1	13
T25	- Elaborar el coste del proyecto	1	1	13
T26	- Reunión con el asesor y profesores del curso	0,5	1,5	66,5
	1.1.7 Catálogo de requerimientos	**5**	**7**	**138**
T27	- Elaborar la lista de requerimientos funcionales	2,5	2,5	32,5
T28	- Elaborar la lista de requerimientos no funcionales	1	1	13
T29	- Validar la lista de requerimientos con un experto	1	2	26
T30	- Reunión con el asesor y profesores del curso	0,5	1,5	66,5
	1.2 Documentos de diseño	**10,5**	**9**	**196,5**
T31	- Reunión con el asesor para validar los documentos	3	1,5	99

	1.2.1 Casos de uso	**1,5**	**1,5**	**19,5**
T32	- Elaborar los casos de uso para el sistema	1,5	1,5	19,5
	1.2.2 Diagrama de actividades	**1,5**	**1,5**	**19,5**
T33	- Elaborar el diagrama de actividades para el sistema	1,5	1,5	19,5
	1.2.3 Diagrama de arquitectura	**1,5**	**1,5**	**19,5**
T34	- Elaborar el diagrama de arquitectura para el sistema	1,5	1,5	19,5
	1.2.4 Mockup	**3**	**3**	**39**
T35	- Elaborar los mockups para el sistema	2	2	26
T36	- Probar el flujo con una persona externa al proyecto	1	1	13
	1.3 Software desarrollado	**230,1**	**234,1**	**3377,3**
	1.3.1 Módulo de preparación de proceso electoral	**38,3**	**30,3**	**393,9**
	1.3.1.1 Código fuente	**20,3**	**20,3**	**263,9**
T38	- Implementar módulo de preparación de proceso electoral	20	20	260
T39	- Actualizar el repositorio de GitLab	0,3	0,3	3,9
	1.3.1.2 Pruebas	**10**	**10**	**130**
T40	- Realizar pruebas unitarias e integrales	10	10	130
	1.3.2 Módulo de registro de electores	**30,3**	**30,3**	**393,9**
	1.3.2.1 Código fuente	**20,3**	**20,3**	**263,9**
T41	- Implementar módulo de registro de electores	20	20	260
T42	- Actualizar el repositorio de GitLab	0,3	0,3	3,9
	1.3.2.2 Pruebas	**10**	**10**	**130**
T43	- Realizar pruebas unitarias e integrales	10	10	130
T44	- Reunión con el asesor para validar los documentos	2	4	146
	1.3.3 Módulo de emisión de votos	**30,3**	**30,3**	**393,9**
	1.3.3.1 Código fuente	**20,3**	**20,3**	**263,9**
T45	- Implementar módulo de emisión de votos	20	20	260
T46	- Actualizar el repositorio de GitLab	0,3	0,3	3,9
	1.3.3.2 Pruebas	**10**	**10**	**130**
T47	- Realizar pruebas unitarias e integrales	10	10	130
	1.3.4. Módulo de cifrado	**40,3**	**40,3**	**523,9**
	1.3.4.1 Código fuente	**30,3**	**30,3**	**393,9**
T49	- Implementar módulo de cifrado de votos	30	30	390
T50	- Actualizar el repositorio de GitLab	0,3	0,3	3,9
	1.3.4.2 Pruebas	**10**	**10**	**130**
T51	- Realizar pruebas unitarias e integrales	10	10	130
T52	- Reunión con el asesor para validar los documentos	2	4	146
	1.3.5. Módulo de verificación individual	**30,3**	**30,3**	**393,9**
	1.3.5.1 Código fuente	**20,3**	**20,3**	**263,9**

T53	- Implementar módulo de verificación individual	20	20	260
T54	- Actualizar el repositorio de GitLab	0,3	0,3	3,9
	1.3.5.2 Pruebas	**10**	**10**	**130**
T55	- Realizar pruebas unitarias e integrales	10	10	130
	1.3.6 Módulo de escrutinio	**30,3**	**30,3**	**393,9**
	1.3.6.1 Código fuente	**20,3**	**20,3**	**263,9**
T56	- Implementar módulo de escrutinio	20	20	260
T57	- Actualizar el repositorio de GitLab	0,3	0,3	3,9
	1.3.6.2 Pruebas	**10**	**10**	**130**
T58	- Realizar pruebas unitarias e integrales	10	10	130
T59	- Reunión con el asesor para validar los documentos	2	4	146
	1.3.7 Módulo de auditoría	**30,3**	**30,3**	**393,9**
	1.3.7.1 Código fuente	**20,3**	**20,3**	**263,9**
T60	- Implementar módulo de auditoría	20	20	260
T61	- Actualizar el repositorio de GitLab	0,3	0,3	3,9
	13.7.1 Pruebas	**10**	**10**	**130**
T62	- Realizar pruebas unitarias e integrales	10	10	130
T63	- Reunión con el asesor para validar los documentos	2	4	52
	1.4 Configuración del contrato inteligente	**6,3**	**8,3**	**201,9**
T64	- Configurar el contrato inteligente para despliegue	4	4	52
T65	- Reunión con el asesor para validar los documentos	2	4	146
T66	- Actualizar el repositorio de GitLab	0,3	0,3	3,9
	1.5 Despliegue	**7**	**9**	**65**
T67	- Desplegar el proyecto en la blockchain	3	3	39
T68	- Elaborar el reporte de despliegue	2	2	26
T69	- Reunión con el asesor para validar los documentos	2	4	146

- **Cronograma del proyecto**

ID	Nombre	Fecha Inicio	Fecha Fin	Dependencias
	Proyecto de Tesis	**1/4/2020**	**25/9/2020**	
	1.1 Documentos de análisis	**1/4/2020**	**4/9/2020**	
	1.1.1 Estado del arte	**1/4/2020**	**20/4/2020**	
T01	- Investigar en motores de base de datos	1/4/2020	15/4/2020	
T02	- Crear el formulario de extracción de datos	15/4/2020	20/4/2020	T01
T03	- Reunión con el asesor y profesores del curso	20/4/2020	20/4/2020	T02

	1.1.2 Marco conceptual	**21/4/2020**	**3/5/2020**	
T04	- Investigar conceptos en internet	21/4/2020	1/5/2020	T01
T05	- Leer leyes orgánicas y metodologías	1/5/2020	3/5/2020	T01
	1.1.3 Problemática	**3/5/2020**	**9/5/2020**	
T06	- Investigar la problemática en diversos artículos	3/5/2020	7/5/2020	T03
T07	- Elaborar árbol de problemas	7/5/2020	9/5/2020	T06
T08	- Reunión con el asesor y profesores del curso	9/5/2020	9/5/2020	T05
T09	- Desarrollar la descripción de la problemática	9/5/2020	11/5/2020	T07
	1.1.4 Objetivos	**21/5/2020**	**1/6/2020**	
T10	- Definir el objetivo general del proyecto	21/5/2020	25/5/2020	T09
T11	- Definir los objetivos específicos	25/5/2020	29/5/2020	T10
T12	- Reunión con el asesor y profesores del curso	29/5/2020	1/6/2020	T09
	1.1.5 Resultados	**11/6/2020**	**21/6/2020**	
T13	- Definir los resultados esperados	11/6/2020	14/6/2020	T12
T14	- Crear el mapeo de objetivos, resultados y verificación	14/6/2020	18/6/2020	T13
T15	- Reunión con el asesor y profesores del curso	21/6/2020	21/6/2020	T12
	1.1.6 Plan de proyecto	**21/6/2020**	**10/7/2020**	
T16	- Elaborar el alcance del proyecto	21/6/2020	23/6/2020	T15
T17	- Definir la justificación y el objetivo del proyecto	21/6/2020	21/6/2020	T16
T18	- Definir las restricciones del proyecto	21/6/2020	21/6/2020	T17
T19	- Definir la viabilidad del proyecto	24/6/2020	24/6/2020	T18
T20	- Elaborar el plan de gestión de riesgos	25/6/2020	25/6/2020	T19
T21	- Elaborar el EDT	29/6/2020	1/7/2020	T20
T22	- Elaborar el cronograma del proyecto	4/7/2020	5/7/2020	T21
T23	- Elaborar la lista de actividades	5/7/2020	5/7/2020	T22
T24	- Elaborar la lista de recursos	8/7/2020	10/7/2020	T23
T25	- Elaborar el coste del proyecto	8/7/2020	8/7/2020	T24
T26	- Reunión con el asesor y profesores del curso	8/7/2020	10/7/2020	T15
	1.1.7 Catálogo de requerimientos	**24/8/2020**	**4/9/2020**	
T27	- Elaborar la lista de requerimientos funcionales	24/8/2020	24/8/2020	T26
T28	- Elaborar la lista de requerimientos no funcionales	24/8/2020	24/8/2020	T27
T29	- Validar la lista de requerimientos con un experto	1/9/2020	1/9/2020	T28
T30	- Reunión con el asesor y profesores del curso	3/9/2020	4/9/2020	T28
	1.2 Documentos de diseño	**4/9/2020**	**10/9/2020**	
T31	- Reunión con el asesor para validar los documentos	10/9/2020	10/9/2020	T36
	1.2.1 Casos de uso	**25/8/2020**	**26/8/2020**	
T32	- Elaborar los casos de uso para el sistema	25/8/2020	26/8/2020	T30
	1.2.2 Diagrama de actividades	**26/8/2020**	**28/8/2020**	

T33	- Elaborar el diagrama de actividades para el sistema	26/8/2020	28/8/2020	T31
	1.2.3 Diagrama de arquitectura	**26/8/2020**	**28/8/2020**	
T34	- Elaborar el diagrama de arquitectura para el sistema	26/8/2020	28/8/2020	T32
	1.2.4 Mockup	**26/8/2020**	**5/9/2020**	
T35	- Elaborar los mockups para el sistema	26/8/2020	5/9/2020	T33
T36	- Probar el flujo con una persona externa al proyecto	7/9/2020	9/9/2020	T34
	1.3 Software desarrollado	**9/9/2020**	**6/11/2020**	
	1.3.1 Módulo de preparación de proceso electoral	**9/9/2020**	**25/9/2020**	
	1.3.1.1 Código fuente	**9/9/2020**	**24/9/2020**	
T38	- Implementar módulo de preparación de proceso electoral	9/9/2020	24/9/2020	T35
T39	- Actualizar el repositorio de GitLab	9/9/2020	24/9/2020	T37
	1.3.1.2 Pruebas	**24/9/2020**	**25/9/2020**	**T38**
T40	- Realizar pruebas unitarias e integrales	24/9/2020	25/9/2020	T38
	1.3.2 Módulo de registro de electores	**9/9/2020**	**25/9/2020**	
	1.3.2.1 Código fuente	**9/9/2020**	**24/9/2020**	
T41	- Implementar módulo de registro de electores	9/9/2020	24/9/2020	T35
T42	- Actualizar el repositorio de GitLab	9/9/2020	24/9/2020	T41
	1.3.2.2 Pruebas	**24/9/2020**	**25/9/2020**	
T43	- Realizar pruebas unitarias e integrales	24/9/2020	25/9/2020	T41
T44	- Reunión con el asesor para validar los documentos	25/9/2020	25/9/2020	T43,T40
	1.3.3 Módulo de emisión de votos	**9/9/2020**	**2/10/2020**	
	1.3.3.1 Código fuente	**9/9/2020**	**1/10/2020**	
T45	- Implementar módulo de emisión de votos	9/9/2020	1/10/2020	T35, T43
T46	- Actualizar el repositorio de GitLab	9/9/2020	1/10/2020	T45
	1.3.3.2 Pruebas	**1/10/2020**	**2/10/2020**	
T47	- Realizar pruebas unitarias e integrales	1/10/2020	2/10/2020	T45
	1.3.4. Módulo de cifrado	**16/9/2020**	**9/10/2020**	
	1.3.4.1 Código fuente	**16/9/2020**	**8/10/2020**	
T49	- Implementar módulo de cifrado de votos	16/9/2020	8/10/2020	T35, T45
T50	- Actualizar el repositorio de GitLab	16/9/2020	8/10/2020	T49
	1.3.4.2 Pruebas	**8/10/2020**	**9/10/2020**	
T51	- Realizar pruebas unitarias e integrales	8/10/2020	9/10/2020	T35, T48
T52	- Reunión con el asesor para validar los documentos	9/10/2020	9/10/2020	T51
	1.3.5. Módulo de verificación individual	**16/9/2020**	**9/10/2020**	
	1.3.5.1 Código fuente	**16/9/2020**	**8/10/2020**	
T53	- Implementar módulo de verificación individual	16/9/2020	8/10/2020	T35, T50
T54	- Actualizar el repositorio de GitLab	16/9/2020	8/10/2020	T53
	1.3.5.2 Pruebas	**8/10/2020**	**9/10/2020**	
T55	- Realizar pruebas unitarias e integrales	8/10/2020	9/10/2020	T53
	1.3.6 Módulo de escrutinio	**10/10/2020**	**24/10/2020**	
	1.3.6.1 Código fuente	**10/10/2020**	**23/10/2020**	
T56	- Implementar módulo de escrutinio	10/10/2020	23/10/2020	T35, T53
T57	- Actualizar el repositorio de GitLab	10/10/2020	23/10/2020	T56
	1.3.6.2 Pruebas	**23/10/2020**	**24/10/2020**	
T58	- Realizar pruebas unitarias e integrales	23/10/2020	24/10/2020	T56
T59	- Reunión con el asesor para validar los documentos	24/10/2020	27/10/2020	T58
	1.3.7 Módulo de auditoría	**24/10/2020**	**6/11/2020**	

	1.3.7.1 Código fuente	**24/10/2020**	**5/11/2020**	
T60	- Implementar módulo de auditoria	24/10/2020	5/11/2020	T35, T58
T61	- Actualizar el repositorio de GitLab	24/10/2020	5/11/2020	T57
	13.7.1 Pruebas	**5/11/2020**	**6/11/2020**	
T62	- Realizar pruebas unitarias e integrales	5/11/2020	6/11/2020	T60
T63	- Reunión con el asesor para validar los documentos	6/11/2020	6/11/2020	T62
	1.4 Configuración del contrato inteligente	**6/11/2020**	**7/11/2020**	
T64	- Configurar el contrato inteligente para despliegue	6/11/2020	7/11/2020	T63
T65	- Reunión con el asesor para validar los documentos	7/11/2020	7/11/2020	T64
T66	- Actualizar el repositorio de GitLab	6/11/2020	7/11/2020	T64
	1.5 Despliegue	**7/11/2020**	**12/11/2020**	
T67	- Desplegar el proyecto en la blockchain	7/11/2020	8/11/2020	T64
T68	- Elaborar el reporte de despliegue	8/11/2020	10/11/2020	T67
T69	- Reunión con el asesor para validar los documentos	10/11/2020	12/11/2020	T68

- **Lista de recursos**

	Descripción	**Cantidad**	**Oportunidad de uso**
1.	**Involucrados**		
1.1.	- Tesista	1	Elaborar el proyecto de Tesis
1.2.	- Asesor de tesis	1	Asesorar al tesista en conocimientos técnicos.
1.3.	- Profesores del curso	3	Enseñar al tesista la metodología de investigación
1.4.	- Experto en procesos electorales	1	Validar documentos elaborados por el tesista.
1.5.	- Jurados de exposición	2	Calificar la presentación del proyecto de tesis
2.	**Materiales**		
2.1.	- No aplica	---	---
3.	**Estándares**		
3.1.	- ISO/IEC 12207	---	Durante el desarrollo del SW.
3.2.	- Estándares legales, procedimentales y técnicos de los sistemas de votación electrónica propuestas con el Consejo de Europa.	---	Durante el desarrollo del SW.
4.	**Equipamiento**		
4.1.	- Laptop		Para el desarrollo de la tesis, tanto documentación como para la programación.
4.2.	- Energía eléctrica	---	Para proveer luz al tesista
4.3.	- Servicio de internet	---	Para poder conectarse a Internet
5.	**Herramientas requeridas**		
5.1.	- NodeJS	---	Para desarrollar el front-end del sistema.
5.2.	- JavaScript	---	Para implementar el sistema
5.3.	- React JS	---	Para ser usado como framework en el front-end
5.4.	- Ethereum	---	Para desplegar el sistema
5.5.	- Solidity	---	Para desarrollar el contrato inteligente
5.6.	- Truffle	---	Para desarrollar el contrato inteligente
5.7.	- Ganache	---	Para realizar pruebas en un blockchain local
5.8.	- Metamask	---	Para interactuar entre la Dapp y Ethereum
5.9.	- Visual Studio Code	---	Para ser usado como editor de texto en el desarrollo de los módulos
5.10.	- Git	---	Para manejar las versiones del proyecto. Así como una copia de seguridad en internet
5.11.	- Diagramas net	---	Para realizar los diagramas de diseño y análisis

- **Costeo del Proyecto**

Ítem	Descripción	Unidad	Cantidad	Valor Unidad (S/.)	Monto Parcial (S/.)	Monto Total (S/.)
0	**Costo total del proyecto**	**---**	**---**	**---**	**---**	**11,260**
1.	**Estudiantes o tesistas**	**---**	**---**	**---**	**---**	**6,080.00**
1.1	Tesista (*)	Horas	380	16	6,080	
2.	**Otros participantes**	**---**	**---**	**---**	**---**	**4,830**
2.1	Asesor de Tesis	Horas	45	60	2,700	
2.2	Profesor del curso	Horas	30	60	1,800	
2.3	Jurados de exposición del curso	Horas	2	60	120	
2.4	Experto en Procesos electorales	Horas	3	70	210	
3.	**Servidores**	**---**	**---**	**---**	**---**	**350**
2.1	Máquina virtual blockchain Ethereum	---	1	300	300	
2.2	Servidor AWS Educate para front-end	---	1	50	50	

(*) La energía eléctrica, laptop y el uso de internet se agregar al monto de valor unidad al tesista como S/. 3.00 /Hora.

Anexo C: Catálogo de requerimientos

A continuación, se presenta el catálogo de requerimientos.

N°	CÓDIGO	REQUISITO	MÓDULO	PRIORIDAD
1	PRE001	El sistema no debe permitir que un elector pueda emitir su voto antes de empezar la etapa de votación.	MÓDULO DE PREPARACIÓN DEL PROCESO ELECTORAL	Exigible
2	PRE002	El administrador debe poder crear un proceso electoral.	MÓDULO DE PREPARACIÓN DEL PROCESO ELECTORAL	Exigible
3	PRE003	El administrador debe poder registrar los candidatos.	MÓDULO DE PREPARACIÓN DEL PROCESO ELECTORAL	Exigible
4	PRE004	El administrador debe configurar el proceso electoral para que se permita crear cédulas de voto único y/o voto para congresistas y voto para parlamento andino seleccionados en el caso que sea necesario.	MÓDULO DE PREPARACIÓN DEL PROCESO ELECTORAL	Exigible
5	PRE005	El sistema debe mostrar toda la lista de los candidatos permitidos.	MÓDULO DE PREPARACIÓN DEL PROCESO ELECTORAL	Exigible
6	REG001	El administrador debe ingresar un archivo .csv que contendrá la lista de electores que participaran del proceso electoral.	MÓDULO DE REGISTRO DE ELECTORES	Exigible
7	REG002	El sistema debe procesar el archivo que contenga la lista de los electores y registrar la billetera electrónica del elector al sistema. Así mismo se	MÓDULO DE REGISTRO DE ELECTORES	Exigible

		notificará esta acción al elector vía correo electrónico.		
8	EMI001	El sistema debe permitir ingresar al elector a la plataforma para realizar el proceso de elección.	MÓDULO DE EMISIÓN DE VOTOS	Exigible
9	EMI003	El sistema debe permitir al elector seleccionar al candidato de su preferencia.	MÓDULO DE EMISIÓN DE VOTOS	Exigible
10	EMI004	El sistema debe mostrar un mensaje al elector que solo puede introducir una papeleta en la urna electrónica.	MÓDULO DE EMISIÓN DE VOTOS	Exigible
11	EMI005	El sistema debe mostrar todos los partidos políticos en las cédulas manejando el mismo tipo de fuente, color y tamaño con el fin de no influenciar la intención del voto del elector hacia un partido político.	MÓDULO DE EMISIÓN DE VOTOS	Exigible
12	EMI006	El sistema debe impedir que una vez que el elector haya emitido su voto, éste pueda modificarse.	MÓDULO DE EMISIÓN DE VOTOS	Exigible
13	EMI007	El elector puede emitir votos en blanco.	MÓDULO DE EMISIÓN DE VOTOS	Exigible
14	EMI008	El elector puede modificar su voto en todo momento antes de la emisión definitiva del mismo.	MÓDULO DE EMISIÓN DE VOTOS	Exigible
15	EMI009	El sistema debe mostrar un resumen de las cédulas con las selecciones realizadas por el elector antes de que este sea emitido.	MÓDULO DE EMISIÓN DE VOTOS	Exigible
16	VER001	El sistema debe emitir un comprobante el cual indique que el elector a emitido su voto.	MÓDULO DE VERIFICACIÓN INDIVIDUAL	Exigible
17	VER002	El sistema notificará al elector vía correo electrónico que su voto ha sido emitido.	MÓDULO DE VERIFICACIÓN INDIVIDUAL	Exigible
18	VER003	El comprobante emitido no debe mostrar información sobre el voto emitido.	MÓDULO DE VERIFICACIÓN INDIVIDUAL	Exigible
19	ESC001	El proceso electoral debe iniciar y finalizar de forma manual por el administrador del sistema.		
20	ESC002	El sistema debe poseer una urna electrónica en donde se almacenen los votos registrados para su posterior escrutinio.	MÓDULO DE ESCRUTINIO	Exigible
21	ESC003	El sistema debe garantizar que los votos contenidos en la urna electrónica y los votos que se escrutan son, y seguirán siendo anónimos.	MÓDULO DE ESCRUTINIO	Exigible
22	ESC004	El sistema debe escrutar todo voto depositado en la urna electrónica.	MÓDULO DE ESCRUTINIO	Exigible
23	ESC005	El sistema debe revelar el número de votos emitidos solo cuando se proceda al cierre de la urna electrónica.	MÓDULO DE ESCRUTINIO	Exigible
24	ESC006	El sistema debe realizar el proceso de escrutinio una vez finaliza la etapa de emisión de votos.	MÓDULO DE ESCRUTINIO	Exigible
25	CIF001	El sistema preservará la confidencialidad de los votos.	MÓDULO DE CIFRADO	Exigible
26	CIF002	El sistema encriptará cada voto que haya sido emitido y lo guardará en la urna electrónica.	MÓDULO DE CIFRADO	Exigible

27	CIF003	El sistema desencriptará los resultados finales una vez haya terminado de realizar el conteo de los votos.	MÓDULO DE CIFRADO	Exigible
28	AUD001	El administrador y el auditor son los únicos que deben poder visualizar los reportes de sufragio y escrutinio.	MÓDULO DE AUDITORIA	Exigible
29	AUD002	El sistema mantendrá un control de auditoria respecto a usuario y hora donde se registre quien ha realizado cada modificación.	MÓDULO DE AUDITORIA	Exigible
30	AUD003	El sistema debe impedir que la sección de auditoria sea modificada. así mismo, esta sección solo será visible para el administrador del sistema y el director de mesa presidencial.	MÓDULO DE AUDITORIA	Exigible
31	AUD004	El sistema generará un acta de sufragio el cual detallará la hora en la que se emitió cada voto, la cuenta asociada a dicho voto, y el voto encriptado.	MÓDULO DE AUDITORIA	Exigible
32	AUD005	El sistema generará un acta de escrutinio que detalle el número de votos en blanco, votos válidos, así como la cantidad de votos por cada partido electoral, la hora de inicio del escrutinio y la hora final.	MÓDULO DE AUDITORIA	Exigible
33	AUD006	La sección de auditoria salvaguardará el anonimato de los electores en todo momento.	MÓDULO DE AUDITORIA	Exigible
34	AUD007	El sistema web puede ser utilizado en cualquier sistema operativo.	MÓDULO DE AUDITORIA	Exigible

El catálogo de requerimientos y sus versiones se encuentra en el siguiente enlace:

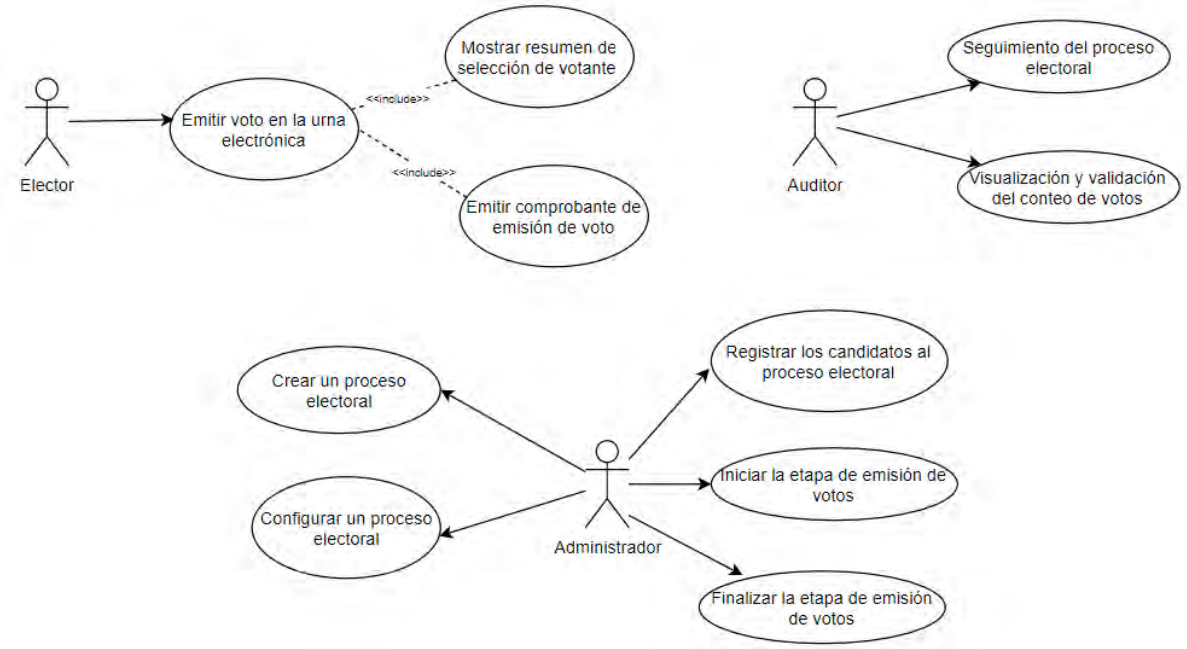
https://drive.google.com/drive/folders/11GMn6odbgoMaslQRnwhDSbVFFfJUqz_3

Así mismo, el documento de acta de reunión con la experta de la ONPE y el video de la reunión como prueba de la validación del documento se encuentran en el siguiente enlace:

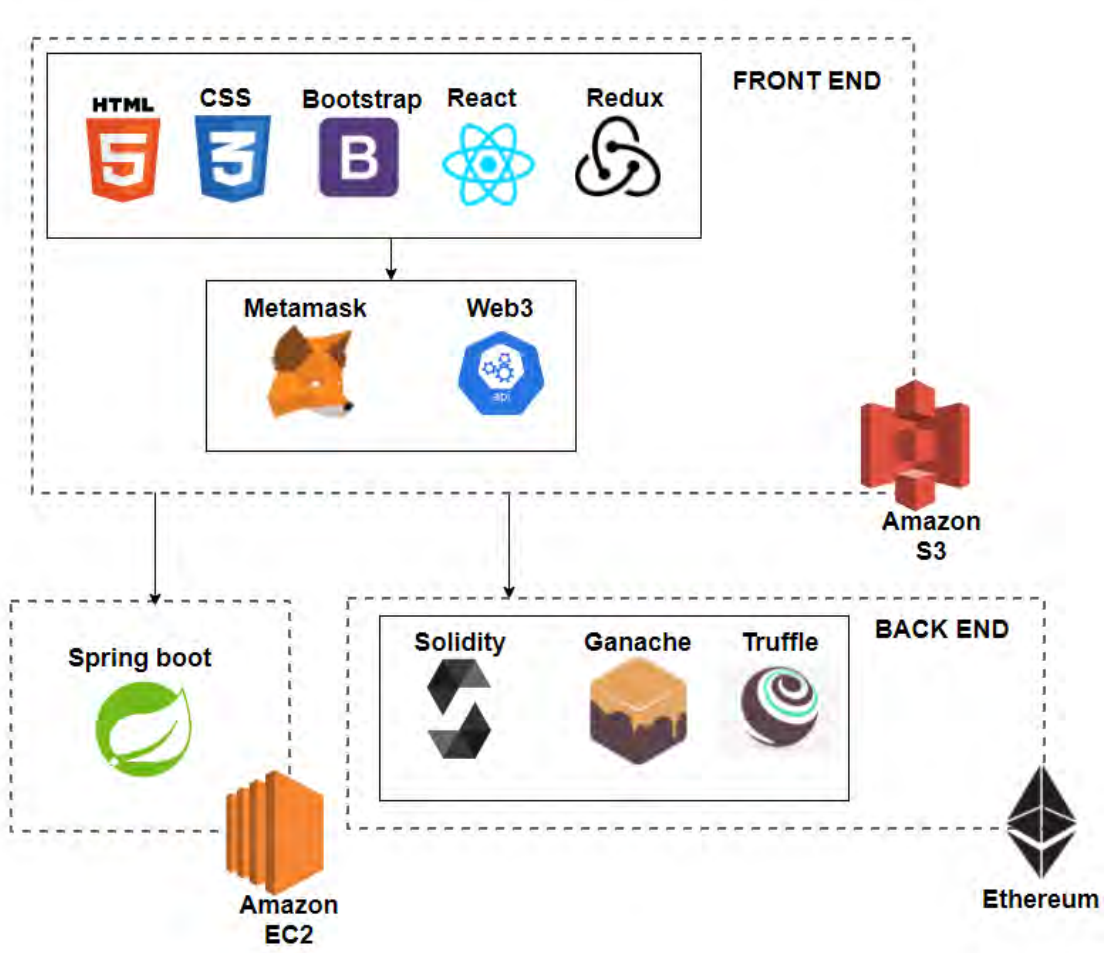
https://drive.google.com/drive/folders/11GMn6odbgoMaslQRnwhDSbVFFfJUqz_3

Anexo D: Documentos de diseño y prototipado

A continuación, se presenta el diagrama de casos de uso.



En el documento de diseño se explican a detalle los actores, funcionalidades y el detalle de cada caso de uso. Así mismo, también se presenta el diagrama de arquitectura.



Así mismo, para ver a más detalle los diagramas de diseño presentados puede acceder al siguiente enlace con el nombre de “Diagramas de diseño y arquitectura.pdf”:

https://drive.google.com/drive/folders/1h3lVo7V3jZIsztdgFO5oDbWOE-tXZTNZ

Por último, el prototipado en su versión final y completo se encuentra en el siguiente enlace con el nombre de “Prototipo.pdf”:

https://drive.google.com/drive/u/0/folders/1h3lVo7V3jZIsztdgFO5oDbWOE-tXZTNZ

Anexo E: Pruebas unitarias e integrales

A. Pruebas para Resultado alcanzado 1

A continuación, se muestra la tabla de casos de prueba en donde se encontrarán pruebas unitarias e integrales para el módulo de creación del proceso electoral y registro de electores.

Plan de Pruebas para módulo de creación del proceso electoral		
Código	**Componente**	**Descripción de lo que se probará**
CP01	Inicio de sesión del Administrador	Se probará iniciar sesión como administrador en el sistema
CP02	Formulario de Datos generales	Se probará el llenado del formulario de datos generales de la creación del proceso electoral
CP03	Formulario de Partidos políticos y candidatos	Se probará el llenado del formulario de creación de partidos políticos y sus candidatos que participarán en el proceso electoral
CP04	Formulario de Listas de votación	Se probará el llenado del formulario de creación de listas de votación para el proceso electoral
CP05	Formulario de Editar Datos generales	Se probará el llenado del formulario de creación de listas de votación para el proceso electoral
CP06	Formulario de Editar Partidos políticos y candidatos	Se probará el llenado del formulario de creación de listas de votación para el proceso electoral
CP07	Formulario de Editar Listas de votación	Se probará el llenado del formulario de creación de listas de votación para el proceso electoral
CP08	Iniciar proceso electoral	Se probará el inicio del proceso electoral

Plan de Pruebas para el módulo de registro de electores		
Código	**Componente**	**Descripción de lo que se probará**
CP01	Subir archivo de electores	Se probará subir el archivo csv de contiene la lista de electores para el proceso electoral.
CP02	Procesar archivo	Se probará la carga del archivo al sistema y procesamiento de su data.
CP03	Generación de credenciales	Se probará generar credenciales para cada usuario y su envío de credenciales a su correo.

Así mismo, para ver a más detalle de las pruebas realizadas presentados puede acceder al siguiente enlace:

https://drive.google.com/drive/u/0/folders/11vvuMnJS6DSWtWI_PEF4W2LDCUbAJQ2Z

B. Pruebas para Resultado alcanzado 2

A continuación, se muestra la tabla de casos de prueba en donde se encontrarán pruebas unitarias e integrales para el módulo de emisión de votos.

Plan de Pruebas para el módulo de emisión de votos		
Código	**Componente**	**Descripción de lo que se probará**
CP01	Inicio de sesión del elector	Se probará iniciar sesión como elector en el sistema
CP02	Registrar selección de candidatos de preferencia	Se probará el registro de las selecciones de candidatos en el proceso de emisión del voto.

Así mismo, para ver a más detalle de las pruebas realizadas puede acceder al siguiente enlace:

https://drive.google.com/drive/u/0/folders/1XWFT8ioR973HwnBFXiczhOsdsvytKAwM

C. Pruebas para Resultado alcanzado 5

A continuación, se muestra la tabla de casos de prueba en donde se encontrarán pruebas unitarias e integrales para el módulo de encriptación de votos.

Plan de Pruebas para el algoritmo de encriptación de votos		
Código	**Componente**	**Descripción de lo que se probará**
CP01	Formulario de registro de autoridades de escrutinio	Se probará iniciar sesión como administrador en el sistema
CP02	Formulario de registro de generación de clave privada.	Se probará la generación de claves privadas por parte de cada autoridad de escrutinio

Así mismo, para ver a más detalle de las pruebas realizadas puede acceder al siguiente enlace con el nombre de “Pruebas del algoritmo de encriptación del proceso electoral.pdf”:

https://drive.google.com/drive/u/0/folders/1ajXUDGVTxPSSQ4N87_oUmEemEGQhdtj-

Finalmente, para la realización de las pruebas de funcionalidad del algoritmo se desarrolló una prueba en Remix el cual esta detallado en el documento “Pruebas de funcionalidad del algoritmo de encriptación.pdf”: https://drive.google.com/drive/u/0/folders/1ajXUDGVTxPSSQ4N87_oUmEemEGQhdtj-

D. Pruebas para Resultado alcanzado 7

A continuación, se muestra la tabla de casos de prueba en donde se encontrarán pruebas unitarias e integrales de funcionalidades que permitan la auditoría.

Plan de Pruebas para el módulo de auditoría		
Código	**Componente**	**Descripción de lo que se probará**
CP01	Inicio de sesión del Auditor	Se probará iniciar sesión como auditor en el sistema
CP02	Visualización de datos de la creación del proceso electoral	Se probará el acceso a la información del proceso electoral
CP03	Visualización de los reportes	Se probará el acceso de los reportes finales del proceso electoral

Así mismo, para ver a más detalle de las pruebas realizadas puede acceder al siguiente enlace con el nombre de “Pruebas módulo de auditoría.pdf”: https://drive.google.com/drive/u/0/folders/1JezXQ0dYxNbUjsZaRv_n8BmL6qz2XflY

E. Pruebas para Resultado alcanzado 8

A continuación, se muestra la tabla de casos de prueba en donde se encontrarán pruebas unitarias e integrales para el módulo de verificación individual.

Plan de Pruebas para el módulo de verificación individual		
Código	**Componente**	**Descripción de lo que se probará**

CP01	Emisión de comprobante de voto emitido	Se probará el envío del comprobante de emitir voto al correo electrónico.
CP02	Validar que ya no puede emitir su voto por segunda vez	Se probará que se denegará el acceso a emitir su voto de nuevo, más bien se le mostrará su comprobante de emisión.

Así mismo, para ver a más detalle de las pruebas realizadas puede acceder al siguiente enlace con el nombre “Pruebas módulo de verificación de los votos.pdf”: https://drive.google.com/drive/u/0/folders/18ZVK0mAKGeQdIDSYzEa1J9hgDTqnKUMi

Anexo F: Pruebas de funcionalidad del módulo de escrutinio de votos

Para acceder a la prueba de funcionalidad del módulo de escrutinio se puede realizar mediante el siguiente enlace con el nombre “Pruebas de funcionalidad del módulo de escrutinio.pdf”: https://drive.google.com/drive/folders/1UHYOfNRtMsSqdIGbx80YHDMyu1_7wJ_T