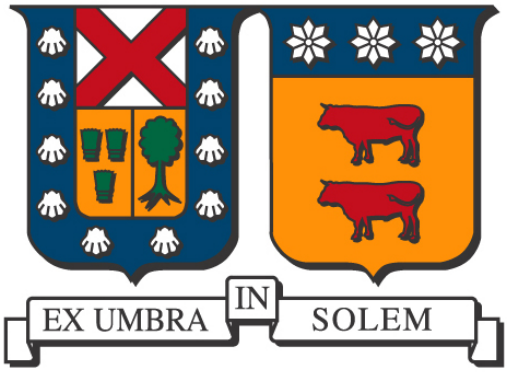
2022-03

DESARROLLO BACKEND DE PLATAFORMA WEB PARA EMISIO N Y VERIFICACION DE CERTIFICACIONES ACADEMICAS, USANDO TECNOLOGIA BLOCKCHAIN

TRONCOSO GARGIOLI, RODOLFO ALEJANDRO

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE ELECTRÓNICA
VALPARAÍSO - CHILE



"DESARROLLO BACKEND DE PLATAFORMA WEB PARA EMISIÓN Y VERIFICACIÓN DE CERTIFICACIONES ACADÉMICAS, USANDO TECNOLOGÍA BLOCKCHAIN"

RODOLFO ALEJANDRO TRONCOSO GARGIOLI

MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE INGENIERO CIVIL TELEMÁTICO

PROFESOR GUÍA: JOSÉ MANUEL MARTINEZ VERDUGO

PROFESOR CORREFERENTE: GABRIEL TORRES YARZA

Marzo - 2022

Agradecimientos

A todo el que lea:

Soy un tipo con suerte. El tipo de suerte que hace que te cruces con las personas correctas, de haber nacido de la mujer correcta, de haberse echado los -muchos- ramos correctos, de ser amigo -valga la redundancia- de los amigos correctos y de haber confiado -y ser digno de confianza- de las personas correctas.

Para Telemática:

Si viniste a la U solo a estudiar, te recomiendo que te eches un par de ramos y aprendas a disfrutar el proceso. Si ya te va bien, y lo estás pasando súper, espero que pronto te sientas desafiad@. Un reloj solo puede ser más preciso, en la medida que tenga más entropia, por favor, ¡Expándete! Hay gente que vale la pena conocer y, que si te atreves, de seguro encontrarás en ellos algo más que un mero favor:

María Ibacache, "Mari". Para mi usted es quien hace que las cosas pasen en nuestra carrera, gracias por ser y hacer mucho más que eso. Por las palabras de aliento, por la compañia, por todo aquello que ha hecho por más de alguno de nosotros :)

José Manuel, tuve la suerte de conocerte como estudiante. Gracias por tanta paciencia, tino y apoyo en el desarrollo de este trabajo. Aún más agradecido estoy por ayudarme a entender qué es la universidad y qué es *morir con las botas puestas.*

Gabriel Torres, "Profe Gabo" Usted es el epítome de la reinvención profesional. Si alguna vez debo dedicarme a algo *nah que ver con mi pregrado* seguiré fielmente su ejemplo. Gracias por sentirse orgulloso de cada uno de los que hemos sido sus alumnos.

A mis amigos:

Me costó un mundo entrar a la carrera y universidad que quería y... Me costó el triple salir por la puerta ancha. Nunca fui lo que se espera de un buen estudiante, pero como soy un tipo con suerte, tengo la bendición de tener amigos y aliados que hicieron esto posible. Es a ellos quienes quiero saludar en primer lugar, y en el orden en que los conocí:

Don Javier Martínez, "Zander" ¡Cuántos carretes innecesarios y conversaciones de todo tipo hubo ahí! Viajes a Temuco, reflexiones de vida y jornadas de estudio que pasamos.

Don Manuel Sánchez, Sr. Sánchez "Mi mejor hombre". Más allá de que me hayas ayudado enseñándome con infinita paciencia más de algún ramo y haber vivido juntos, es muy bacán seguir viendo cómo rompes con el statu quo y cómo hasta el día de hoy, nuestra amistad crece.

Don David Tapia, Sr. Tapia "Mi secuaz". Compañero de batallas, depositario de mi confianza y mi partner. Gracias por apañar en negocios, darnos ánimo y estar presentes cuando mutuamente lo hemos necesitado. Nos quedan aún más episodios por compartir y gran parte de la seguridad que tengo en vivirlos es porque sé que estaremos ahí para apoyarnos.

Don Lutís González, eres el Ingeniero que más me inspira profesionalmente. Tu calidad humana y compañerismo han hecho muy fácil construir la amistad que nos ha traído tantos momentos y espacios de muy alta calidad.

Fabricio Rosales, ojalá algún día aprenda a ser tan metódico y enfocado como tú. Mientras tanto, qué bacán conocer y compartir con alguien que me ha enseñado tanto del mundo y sus sutilezas... Además de campos.

Mauricio Aros, Mr. Rings. El m3n con la disposición y paciencia de oro. Tengo menos de la mitad de cada una de estas características que tanto te destacan, qué bacán es tener de quién aprender bien esto.

A mi familia:

Mandi, me faltan palabras para expresar lo afortunado que me siento de ser hijo de una mujer que hizo frente a criarnos, prácticamente sola y en un país extranjero, a mi hermana y a mi. Me emociona poder felicitarte por el resultado que conseguiste al poner tanto cariño y cuidado en que la Cafla y yo pudiéramos desarrollarnos y convertirnos en personas que aporten a la sociedad. Decir que este título que hoy obtengo es gracias a ti me parece injusto, pues a ti te debo la vida, mamá.

Cafla, mi máxima compañera de aventuras, juegos y peripecias. Somos nosotros, el apoyo que nos compartimos y nuestras iniciativas, las que le han dado forma a nuestras vidas. Es contigo que aprendí a emprender y también a contar con el apoyo de mis socios y compañeros. No sé cómo, ni quién, y aún menos sé, cuándo partió eso. De lo que estoy seguro es que todo lo aprendí contigo, hermana.

Camilo, hace poco más de 17 años atrás contigo aprendí qué es la amistad, la lealtad y el carácter. Todos valores que además de haberme ayudado a sacar adelante esta carrera, también me han ayudado a lograr muchas de las metas que me he propuesto. ¡Gracias por tanto!

José, te admiro por ser el epítome de vivir el día a día. Además de eso, tienes un lugar especial en mi corazón por ser lo más cercano a una figura paterna presente, por ser el sostén y apoyo de mi mamá, y bueno, también por soportar tanta travesura que de niño cometí.

A los demás et. al

Gracias a los que estuvieron y aportaron pues también son parte de este logro. Respecto a los que no, gracias a ustedes **me es muy fácil** distinguir y reconocer a las personas que quiero mantener en mi vida. Tanto para compartir alegrías, como también, penas.

A mi mismo:

Mismo, sé que haces y das lo mejor de ti en cada cosa que emprendes. Gracias por creer, confiar, por insistir y persistir, por echarle pa'lante en toda circunstancia, y por siempre, hacerlo con fe, cariño y esperanza.

Resumen

En el contexto del Programa de Memorias Multidisciplinarias 2019 [1], de la Universidad Técnica Federico Santa María, la empresa IT Axxion SpA ha propuesto el desafío: ¿Cómo podemos entregar confianza a privados e instituciones, respecto a los logros académicos y formativos que hemos entregado u obtenido?.

En la presente memoria se muestra el diseño, desarrollo e implementación de un sistema web de arquitectura híbrida -centralizada y decentralizada- que de manera segura y persistente permite la gestión, administración y emisión de certificaciones académicas verificables y con incapacidad de repudio, donde interactúan tanto usuarios beneficiados por la certificación, como instituciones y terceros verificadores de dichos certificados.

Para lograr lo anterior se ha desarrollado una aplicación web donde el rol de coordinadores académicos y responsables académicos operando conjuntamente pueden, sin intervención de terceros, emitir certificados confiables y fácilmente trazables.

Palabras clave: Certificación, Blockchain [2], Aplicación web, Sistema de Gestión, Ingeniería de Software, IPFS [3], Contrato inteligente [4], Ethereum [5].

Abstract

In the context of the 2019 Multidisciplinary Memories Program, of the Federico Santa María Technical University, the company IT Axxion SpA has proposed the challenge: How can we give confidence to private institutions, regarding the academic and training achievements that we have delivered or obtained? .

This report shows the design, development and implementation of a hybrid architecture web system -centralized and decentralized- that in a secure and persistent way allows the management, administration and issuance of verifiable academic certifications and with the incapacity of repudiation, where they interact both users benefited from the certification, such as institutions and third party verifiers of said certificates.

To achieve the above, a web application has been developed where the role of academic coordinators and academic managers operating can also, without the intervention of third parties, issue reliable and easily traceable certificates.

Keywords: Certification, Blockchain, Web Application, Management System, Software Engineering, IPFS, Smart Contract, Ethereum.

Glosario

- **ABI**: Interfaz entre aplicación y Binario (*Application Binary Interface*, en inglés).
- **API**: Interfaz de Programación de Aplicaciones (*Application Program Interface*, en inglés).
- **Backend**: Sistema responsable del procesamiento de la información.
- **Blockchain**: Estructura de datos cuya información se agrupa en bloques en una red distribuida (*Cadena de bloques* en español).
- **BPMN**: Modelo y notación de procesos de negocios (*Business Process Model and Notation* en inglés).
- **Contratos Inteligentes**: Abstracción que hace referencia a lógica incorporada en el blockchain. También son conocidos en inglés como *Smart Contracts.*
- **CRUD**: Crear, Leer, Actualizar, Eliminar (*Create, Read, Update, Delete*, en inglés).
- **dApps**: Aplicaciones descentralizadas (*Descentralized Applications* en inglés).
- **Framework**: conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.
- **Frontend**: Sistema responsable de la interfaz de usuario.
- **Hash/Hashing**: Función que al recibir un argumento de entrada, produce como respuesta un texto de largo fijo y finito. Es muy sensible a variaciones pequeñas, por ejemplo a espacios y puntos.
- **HTTPS**: Protocolo seguro de transferencia de Hipertexto (*HyperText Transfer Protocol Secure*).
- **I+D+i**: Investigación, desarrollo e innovación.
- **IPFS**: Sistema Interplanetario de archivos (*Interplanetary File System*, en inglés).
- **JSON**: Notación de Objeto JavaScript (*JavaScript Object Notation*, en inglés).
- **Let's Encrypt**: Autoridad Certificadora gratuita para comunicación TSL.
- **Metamask**: Cliente que permite interactuar con la red de Ethereum.
- **Mobile first**: Convención de diseño priorizando la plena funcionalidad desde dispositivos móviles.
- **Mockup**: Diseño no funcional con carácter demostrativo

- **MVC**: Modelo Vista Controlador (*Model View Controller*, en inglés).
- **MVVM**: Modelo Vista Vista-modelo (*Model View Viewmodel*, en inglés).
- **Node**: Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor (pero no limitándose a ello) basado en el lenguaje de programación JavaScript, asíncrono.
- **ORM**: Mapeo Objeto-Relacional (*Object Relational Mapping*, en inglés).
- **REST**: Transferencia de Estado Representacional (*REpresentational State Transfer*, en inglés).
- **RFC**: Petición de comentarios (*Request For Comments*, en inglés).
- **SLA**: Acuerdo de nivel de servicio (*Service Level Agreement* en inglés).
- **SSL**: Capa de puertos seguros (*Secure Sockets Layer*, en inglés).
- **TLS**: Capa de Transportes de Seguridad (*Transport Layer Secure*, en inglés).
- **Trustify**: Plataforma de gestión de identidades, cursos y emisión de documentos para certificar logros académicos.
- **UI**: Interfaz de Usuario (*User Interface*, en inglés).
- **UML**: Lenguaje Unificado de Modelado (*Unified Modeling Language*, en inglés).
- **UI/UX**: Interfaz de usuario - Experiencia de usuario (User Interface/User Experience).
- **UX**: Experiencia de Usuario (*User Experience*, en inglés).
- **VPS**: Servidor virtual privado (*Virtual Private Server* en inglés).

Índice general

1. Introducción **1**

1.1. Propósito 1

1.2. Alcance 2

1.3. Definición del problema 2

1.3.1. Antecedentes de IT Axxion SpA 3

1.4. Hipótesis 3

1.5. Objetivos 4

1.5.1. Objetivo general 4

1.5.2. Objetivos específicos 4

1.6. Estructura de la memoria 4

2. Marco Teórico **5**

2.1. Soluciones actuales 5

2.1.1. Enfoque tradicional 5

2.1.2. Enfoque basado en Tecnologías de la información: 6

2.1.3. Enfoque utilizado por la contraparte: 7

2.2. Marco Legal 8

3. Estado del Arte y Solución **9**

3.1. Estado del Arte 9

3.1.1. Blockcert Wallet 9

3.1.2. OpenCerts 10

3.2. Propuesta de solución 10

3.3. Debilidades y amenazas 12

4. Definición **13**

4.1. Identificación de requerimientos 13

4.1.1. Requerimientos funcionales 14
4.1.2. Requerimientos no funcionales 14
4.2. Patrones de software 15
4.3. Interacción con dispositivos 15
4.4. Estándares y buenas prácticas 16
4.4.1. Arquitectura de Diseño 16
4.4.2. Descripción de componentes tecnológicos 17
4.4.3. Protocolos de comunicación 18
4.4.4. Control de versiones 19

5. Diseño de la solución **21**
5.1. Diagrama de flujo de datos 21
5.2. Descripción de módulos 23
5.3. Arquitectura del sistema 25
5.4. Modelo de datos 27
5.5. Diseño de Interfaces 28
5.6. Diagrama de casos de uso 28
5.7. Contrato inteligente 29
5.7.1. Contrato Trustify 29
5.8. Flujo de emisión de documento 32

6. Validación **35**
6.1. Validación de DigitalBank Transformación Digital 35
6.2. Análisis de seguridad 36
6.3. Análisis de rendimiento 36

7. Resultados **39**
7.1. Entorno de trabajo 39
7.1.1. Entorno de desarrollo 39
7.1.2. Entorno de pruebas pre-productivas 39
7.2. Implementación de API 40
7.2.1. Cursos 41
7.2.2. Certificado 42
7.2.3. Instructor y Coordinador 43
7.2.4. Misceláneo 44
7.3. Implementación de vistas 44

7.3.1. Panel principal Coordinador 44
7.3.2. Panel principal Instructor 45
7.3.3. Cursos 46
7.3.4. Proceso de Firma Instructor 47
7.3.5. Proceso firma Coordinador 49
7.4. Registro de Instructor 50
7.5. Ingreso a la plataforma 52
7.6. Documentos 52
7.7. Diagramas de secuencias 53
7.7.1. Creación de Instructor 54
7.7.2. Flujo de firma de documentos 54
7.7.3. Verificación de documento 55

8. Conclusiones **57**
8.1. Conclusiones generales 57
8.2. Conclusiones específicas 58
8.3. Trabajo futuro 58

Anexos **61**
A.1. Mockups de las interfaces 61

Índice de figuras

2.1. Proceso de certificación manual 6
2.2. Proceso de certificación basado en tecnología 7

3.1. Esquema de interacciones. 11

5.1. Diagrama de flujo de datos del sistema 22
5.2. Diagrama de Arquitectura - Trustify 25
5.3. Modelo de Datos - Trustify 27
5.4. Casos de uso usuarios Trustify 28
5.5. Diagrama BPMN de proceso de firma y emisión de documentos 33

7.1. Diagrama de secuencia API REST 40
7.2. Vista del tablero principal Instructor Trustify 45
7.3. Vista del tablero principal Coordinador Trustify 45
7.4. Vista de Cursos 46
7.5. Vista Creación de Cursos 46
7.6. Vista Información de Curso 47
7.7. Vista de Cursos Coordinador 47
7.8. Vista Subir información de alumnos al Curso 48
7.9. Vista Información Subida de alumnos al Curso 48
7.10. Vista de firma de información 49
7.11. Vista de opciones Coordinador 49
7.12. Vista de siguientes eventos 50
7.13. Vista de Instructores 50
7.14. Vista crear un Instructor 51
7.15. Vista crear contraseña 51
7.16. Vista crear contraseña con wallet 52

7.17. Vista de interfaz de ingreso 52
7.18. Vista de interfaz de ingreso 53
7.19. Vista de interfaz de ingreso 53
7.20. diagrama de secuencia Creación de un instructor 54
7.21. diagrama de secuencia Proceso de firma 55
7.22. diagrama de secuencia Verificación de documento 56

8.1. Logo del servicio Trustify 58
A.2. Ingreso 61
A.3. Panel de Administración 62
A.4. Ver Coordinadores/Instructores 62
A.5. Crear Coordinador/Instructor 63
A.6. Crear clave de acceso 63
A.7. Ver cursos 64
A.8. Crear un curso 64
A.9. Opciones de documentos Instructor 65
A.10. Subir información del curso 65
A.11. Firmar información del curso por Instructor 66
A.12. Opciones de documentos Coordinador 66
A.13. Cursos pendientes de firmar por Coordinador 67
A.14. Firmar cursos por Coordinador 67
A.15. Lista de documentos emitidos 68
A.16. Verificar documento 68

Índice de tablas

6.1. Tabla de resultado encuesta de validación 35
6.2. Tabla de resultado de mediciones de la aplicación 37

7.1. Tabla de rutas para Cursos 41
7.2. Tabla de rutas para administrar productos 42
7.3. Tabla de rutas para administración de usuarios 43
7.4. Tabla de rutas misceláneas 44

Capítulo 1

Introducción

El Programa de Memorias Multidisciplinarias de la Universidad Técnica Federico Santa María, tiene como propósito recibir desafíos de empresas e instituciones que permitan a sus estudiantes resolver un problema o necesidad concreta de un tercero. Es bajo este contexto que IT Axxion SpA, presentó el desafío "¿Cómo podemos entregar confianza a privados e instituciones, respecto a los logros académicos y formativos que hemos entregado u obtenido?"

La propuesta de solución consiste en una diseñar e implementar una plataforma web que permita gestionar la emisión y validación de certificaciones académicas. Para ello, se considera el uso de tecnologías que permitan almacenar información de manera persistente y procedimientos de verificación de emisores.

1.1. Propósito

Proveer trazabilidad, certeza, persistencia y verificabilidad sobre los reconocimientos académicos entregados u obtenidos. Para lograrlo, se llevará a cabo una investigación de cómo en la actualidad se solucionan problemáticas similares a este, además de hacer un repaso en el estado del arte de las tecnologías disponibles para ello.

Una vez completada la fase de investigación, se tiene contemplado desarrollar una plataforma web que permita articular las tecnologías que permitirían satisfacer el desafío planteado.

La globalización, el actual alcance que tiene internet y la facilidad de acceso a las nuevas tecnologías, han permitido que todo producto o servicio se encuentre a un "*click de distancia*". Esto sumado a las diferentes corrientes de transformación digital que buscan, además de optimizar procesos de negocio; disminuir intermediaciones, usar menos papel y ofrecer una mejor experiencia de cara al usuario/cliente final.

Considerando el contexto tecnológico en el que se plantea el desafío, sumado al uso de las tecnologías existentes que potencian la entrega y gestión confiable a lo largo de los cualquier proceso bien definido, es que a lo largo del desarrollo de esta memoria se diseñará una plataforma que facilite la emisión, validación y verificación de documentos académicos entregados por parte de las instituciones educativas, haciendo énfasis en el hecho de que dichos reconocimientos deben ser perdurables en el tiempo y tan únicos como sea posible.

1.2. Alcance

El alcance de este trabajo de memoria considera el diseño, desarrollo e implementación de una solución al desafío propuesto por la empresa IT Axxion SpA. El alcance del diseño de la solución considera; la identificación y levantamiento de requerimientos, definición de la arquitectura del sistema, diseño de interfaces, diseño de modelo de datos y consideraciones de seguridad. El desarrollo implica el despliegue de un prototipo plenamente funcional.

La planificación para el cumplimiento de este objetivo, considera una ventana de tiempo de ocho (8) meses corridos, el periodo se inicia en Abril 2019 y culmina en Diciembre de 2019. El equipo a cargo del proyecto a decidido denominarse *Trustify* y está compuesto por:

- David Tapia Otarola: Encargado desarrollo e integración de front-end, diseñador de modelo de datos y confiabilidad del sistema.
- Rodolfo Troncoso Gargioli: Responsable de la toma de requerimientos funcionales, diseño e implementación de arquitectura backend y comunicación con el mandante del desafío.

Sin perjuicio de lo anterior, las personas anteriormente mencionadas trabajaran de forma conjunta, aportando en la mejor manera posible de acuerdo a su experiencia al desarrollo de esta plataforma. Adicionalmente, a lo largo de todo el proyecto se contará con el apoyo estratégico y operativo de la empresa IT Axxion SpA.

1.3. Definición del problema

El desafío propuesto por la empresa IT Axxion SpA: *¿Cómo podemos entregar confianza a privados e instituciones, respecto a los logros académicos y formativos que hemos entregado u obtenido?* Para lograr esto, se llevan a cabo reuniones de trabajo tanto con el solicitante, como también con posibles clientes y usuarios de la plataforma. Producto de las cuales se identifica como relevante el que entre sus características el diseño de la plataforma contemple lo siguiente:

- Ser una herramienta de fácil uso.
- Operable con elementos que los usuarios le sean familiares.
- Debe contar con un sistema de almacenamiento seguro y distribuido de los datos relevantes que componen el certificado.

Como punto adicional, se distingue la necesidad de realizar diagnóstico de cómo actualmente se realiza la gestión y emisión de reconocimientos académicos, para identificar y evaluando las ventajas y debilidades que presentar el esquema de trabajo puramente manual o, si se prefiere, de escritorio.

1.3.1. Antecedentes de IT Axxion SpA

La empresa Integración Desarrollo y Asesoría Tecnológica Axxion SpA, en adelante IT Axxion SpA es una empresa de desarrollo de software, la cual a la fecha del levantamiento del desafío que motivó esta memoria contaba con tres años en el mercado y con clientes estables. Es bajo el contexto antes mencionado que, luego de detectar una oportunidad de mercado la empresa propone este desafío.

1.4. Hipótesis

El desafío propuesto busca generar una fuente de información íntegra y con alta capacidad de ser consumida, pues la disponibilidad de esta es clave para establecer relaciones de confianza entre partes que -al no necesariamente conocerse entre sí- se valen de un documento probatorio como garantía.

Adicionalmente, se busca minimizar el tiempo que toma el proceso de generación de certificados y posterior acreditación por parte de autoridades académicas, además del cumplimiento del proceso burocrático necesario para disponibilizar la documentación ante cualquier solicitante, todo lo anterior, potenciado por la mitigación de riesgos ante pérdida de documentos y alteraciones, maliciosas o no, que los documentos puedan sufrir en el proceso.

Con lo anterior como referencia es que el foco estará dado por la problemática y los procesos conocidos en la generación de estos documentos por parte de las instituciones educacionales, definiendo la hipótesis, para efectos de este trabajo, como **Desarrollar un sistema de base tecnológico para la generación de documentos de certificación académica, que cuente el uso de firma digital por parte de las autoridades académicas pertinentes y que almacene toda información relevante de forma persistente, además de permitir la verificación de dicha documentación**.

1.5. Objetivos

1.5.1. Objetivo general

Implementar Una plataforma web que, basada en la segmentación de perfiles, permita: certificar, emitir y validar logros académicos obtenidos por estudiantes.

1.5.2. Objetivos específicos

- **Diseñar** un modelo de datos que permita mantener coherencia y persistencia de los datos que componen lo expuesto en los logros académicos.
- **Integrar** la solución tecnológica a un proceso de negocio compuesto por varios actores que en la actualidad interactúan mediante un proceso puramente manual.
- **Implementar** una solución que satisfaga plenamente el principal requerimiento funcional, siendo este; la emisión segura y persistente de certificados de avalen la obtención de reconocimientos académicos.
- **Desarrollar** las interfaces que permitan la interacción de los usuarios con el sistema.

1.6. Estructura de la memoria

El presente escrito está organizado según la siguiente estructura:

- **Capítulo 2. Estado del arte**. Muestra las alternativas y soluciones actuales disponibles, tanto de índole nacional como internacional, además de la propuesta de solución.
- **Capítulo 3. Definición**. Muestra y repaso de bases técnicas sobre las cuales se desarrollará la plataforma.
- **Capítulo 4. Diseño de la solución**. Apartado enfocado en el diseño conceptual y definición de modelos que permitan comprender el entorno el desafío y problema.
- **Capítulo 5. Desarrollo de la solución**. Se presenta la implementación de los componentes lógicos utilizados en la solución, junto con las interfaces de usuario.
- **Capítulo 6. Validación**. Se analiza si lo desarrollado cumple con las expectativas esperadas de parte de los actores principales participantes del desafío, y con los estándares actuales del internet.
- **Capítulo 7. Conclusiones**. Muestra si fue posible cumplir la hipótesis, junto con el trabajo a futuro a seguir con la plataforma.

Capítulo 2

Marco Teórico

La confianza es requisito esencial e indispensable para la ejecución de cualquier acción entre entidades o personas que se relacionan laboral o comercialmente. Por lo que contar con una única fuente de verdad, confiable, neutra para los participantes y, sobre todo incorruptible facilita cualquier tipo de relación entre partes.

Dentro del contexto de negocios, es normal que al momento de evaluar la contratación de alguien para realizar alguna tarea particular, se espere que como mínimo cuente con las habilidades necesarias para desempeñar de manera adecuada el cargo para el cual será contratado, lo que normalmente se verifica a través de la exigencia de ciertos títulos o credenciales académicas que actúan como garantes de que, quien postula, cuenta con el conocimiento mínimo necesario. Normalmente, dichas credenciales son entregadas como prueba tangible del cumplimiento de requisitos mínimos exigidos por una institución académica competente, y se valida en a través de la autorización, emisión y firma por parte de autoridades académicas reconocidas.

2.1. Soluciones actuales

La forma en que se lleva a cabo el proceso de certificación, si bien puede variar entre distintas instituciones, a nivel abstracto es siempre el mismo proceso. Esto se puede evidenciar gracias a la existencia e interacción de los mismos actores: Un estudiante, uno o más administrativos de apoyo a la gestión y uno o más responsables académicos.

2.1.1. Enfoque tradicional

La linea base, sobre la cual se realizará este trabajo, es de naturaleza puramente manual. De hecho, a la fecha de la realización de este escrito, la Universidad Técnica Federico Santa Mafia sigue este patrón.

El cual, gráficamente, se expresa de la figura 2.1

Figura 2.1: Proceso de certificación manual

De izquierda a derecha, se tiene:

1. **Responsable académico:** Persona encargada de certificar el cumplimiento satisfactorio de requerimientos mínimos, por parte del estudiante, para obtener el reconocimiento académico que da origen a la certificación. Algunos ejemplos de quienes ejercen este rol son: Profesor, relator, evaluador, etc.

2. **Administrativo académico:** Toda persona que contribuye al manejo de la documentación relevante para la ejecución del proceso, algunos ejemplos son: Estafetas, Secretarias/os, Profesionales de apoyo a la gestión, etc.

3. **Director académico:** Es la abstracción de la máxima autoridad dentro del proceso de certificación educativa. Representa a la institución en sí y además es quien, en última instancia, realiza el visado al proceso para su sana concreción. Dentro del contexto universitario, esta definición recae en el cargo de Rector.

2.1.2. Enfoque basado en Tecnologías de la información:

La mayor diferencia respecto al enfoque puramente manual, consiste en el uso de tecnologías de información para realizar un proceso que, si bien en su esencia es igual al anterior, genera ahorros de tiempo y disminución en el uso de papel según lo expuesto en la Figura 2.2.

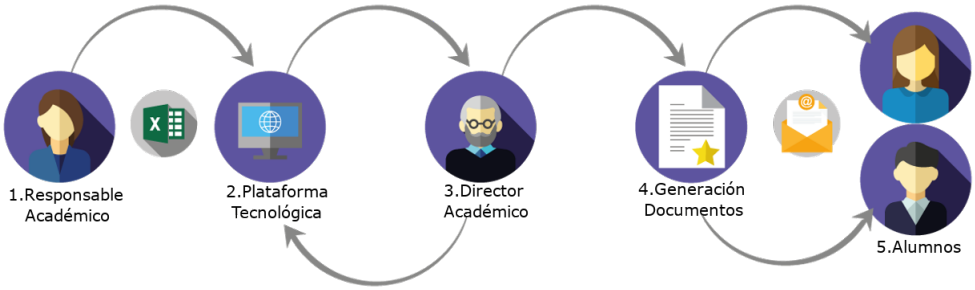


Figura 2.2: Proceso de certificación basado en tecnología

1. **Responsable Académico:** Resposanble de cargar la información al sistema de los alumnos/receptores de los documentos.
2. **Plataforma Tecnológica:** Sistema en el cual se ingresa la información a validar y se disponibiliza a los distintos actores.
3. **Director Académico:** Responsable de acreditar y validar la emisión de documentos a nombre de la institución.
4. **Generación Documentos:** Sistema que genera los documentos en base a la información entregada por el responsable.
5. **Alumnos:** Conjunto de personas receptores de documentos que validan una situación concreta con la institución.

De acuerdo con la información recabada a la fecha del cierre de este escrito, la forma en que se emiten certificaciones académicas en las instituciones educacionales y prestadores de servicios de capacitación consultados, consiste en un híbrido entre los dos sistemas anteriormente mencionados, esto porque si bien existen procesos automatizados -carentes de toda intervención humana- estos son complementados por procesos puramente manuales a lo largo de la emisión, administración y formalización del documento. Lo que forzosamente conlleva un alto grado de intervención humana directa y con ello un aumento en la exposición a riesgos operacionales y posibles intervenciones de terceros.

2.1.3. Enfoque utilizado por la contraparte:

Se deja como antecedente que, si bien la contraparte proponente de esta solución es una empresa de consultoría y desarrollo de software, ellos cuentan con al menos un cliente que necesita una solución que permita entregar a sus alumnos un certificado digital que sirva como testimonio de haber cursado capacitaciones.

En Septiembre de 2017 se implementó una iniciativa de capacitaciones en transformación digital enfocada en la industria financiera Latinoamericana. El primer curso se dictó en Chile y el enfoque de dicho curso, fue ayudar a entender la importancia de tecnologías de la información y cómo éstas impactan el desarrollo del mercado. Posterior a dicha instancia, se crearon y dictaron distintos cursos y ciclos de capacitación para colaboradores de distintas instituciones financieras de Latinoamérica. La manera en que se certificaba la asistencia y aprobación de dichos cursos, consistía en imprimir los *diplomas* y firmar a mano cada uno de ellos, para luego enviar por correo el diploma a cada participante. El procedimiento anterior, si bien cumplía con un protocolo tradicional y permitía entregar solemnidad al proceso, ante el aumento de la demanda y el desarrollo del negocio, la operación descrita se volvió insostenible. Por lo que se hizo necesario contar con alguna herramienta tecnológica que apoyase una mejor ejecución del proceso.

2.2. Marco Legal

En Chile, el consentimiento expresado a través de firma electrónica está regulado en la ley N° 19.799 "*Sobre documentos electrónicos, firma electrónica y servicios de certificación de dicha firma*" [6] promulgada el 25 de Marzo de 2002 y con una última actualización el 09 de Enero de 2014 contenida en la ley N° 20.720 [7], define una firma digital "como cualquier clase de sonido, símbolo o proceso electrónico que permita al receptor de un documento electrónico identificar, al menos formalmente, a su autor" [8] esto permite emitir y certificar documentos de forma digital cumpliendo requisitos específicos para contener un proceso de firma electrónica, simple o avanzada, completo y válido ante la ley. Entre ellas:

- Un código de identificación único del certificado.
- Identificación del prestador de servicio de certificación.
- Los datos de la identidad del titular.
- Su plazo de vigencia.

Con lo anterior, y entendiendo que para efectos de este trabajo no es necesario contar con una certificación ante el Ministerio de Economía Chileno, se aplicará un proceso de firma electrónica simple, la cual vinculada a un proceso de onboarding mediante registro por correo electrónico institucional, permitirá implementar procesos de identificación, verificación y validación para la emisión de diplomas y certificados.

Capítulo 3

Estado del Arte y Propuesta de Solución

3.1. Estado del Arte

En el capítulo anterior se presentaron los requisitos y funcionalidades que debe disponer la plataforma a desarrollar para realizar la emisión, certificación y validación de documentos en un contexto académico. En esta sección, se mencionan algunos desarrollos para procesos de certificación y firmas de documentos similares a las desarrolladas en este documento.

3.1.1. Blockcert Wallet

En el 2017 el Instituto de Tecnología de Massachusetts (MIT) le permitió a 111 estudiantes contar con sus certificados tanto en forma física, como digital usando sus smartphones. La aplicación llamada *Blockcert Wallet* [9] la cual usando la tecnología blockchain, permite de manera segura:

- Tomar control por parte del estudiante de su propio certificado
- Verificar diplomas de otros estudiantes
- Compartir diplomas con terceros
- Contar con una herramienta vinculada directamente al gestor de identidades del MIT

En términos generales, el funcionamiento de esta herramienta consistió en generar un par digital para cada certificado físico emitido. A partir de la experiencia, la Oficina de Registro ha ampliado el programa piloto de diploma digital para incluir una cohorte de estudiantes que se graduaron en septiembre de ese

mismo año. Mientras que en el largo plazo se contempla explorar la posibilidad de ampliar el alcance de esta iniciativa a otros programas educacionales del MIT, entre ellos: MIT Professional Education, Kaufman Teaching Certificate Program y Bernard M. Gordon-MIT Engineering Leadership Program.

3.1.2. OpenCerts

El ministerio de educación de Singapur anunció la implementación de un sistema de certificación de grados académicos utilizando blockchain como parte fundamental para la validación y firma de los documentos, este proyecto conocido como OpenCerts [10], permite que todo certificado gestionado por la plataforma, cuente con un identificador único y verificable. El flujo de funcionamiento considera tres pasos:

1. Al momento de emitir un certificado en OpenCerts, este genera un código único que permite etiquetar la información contenida en dicho certificado. Tanto la información del certificado, como el código en sí, son almacenados en el blockchain.

2. Al momento de abrir un archivo de extensión ".opencert" en el sitio https://www.opencerts.io/, el archivo provisto será comparado con lo almacenado en el blockchain.

3. De manera automatizada, se lleva a cabo un proceso de comparación entre el archivo a certificar y lo entregado por el usuario. Se verifica autenticidad cuando el certificado se muestra en la plataforma.

Como dato extra al proceso, es importante mencionar que OpenCerts es una plataforma de código abierto y que el blockchain usado en su implementación, corresponde a la red pública de Ethereum.

3.2. Propuesta de solución

Luego de entrar en contacto con la empresa proponente del desafío, se acordó enunciar este de la siguiente manera: *¿Cómo podemos entregar confianza a privados e instituciones respecto a los logros académicos y formativos que hemos entregado u obtenido?* Con el propósito de dar una respuesta, se lleva a cabo un proceso de investigación, del cual se logra determinar tecnologías que presentan características técnicas útiles para entregar persistencia, capacidad de no-repudio y seguridad criptográfica tanto a la información como a los procesos que den lugar al tratamiento de la misma. Al respecto, se determina que la tecnología blockchain, un sistema de almacenamiento decentralizado y una interfaz web robusta permitirían cumplir con las características relevantes.

A nivel funcional, se identifican cuatro actores principales:

1. Responsable académico:
   - Es quien inicia el proceso al cargar la planilla de curso y evaluaciones.
   - Proporciona primera firma de información.
2. Director Académico:
   - Máximo responsable de la institución
   - Proporciona segunda firma de la documentación cargada por el responsable académico y con ello se emite la certificación.
3. Usuarios-Alumnos:
   - Reciben la certificación.
   - Disponibilizan certificación para dar testimonio.
4. Usuarios-otros:
   - Consultan, verifican y descargan certificaciones emitidas por la plataforma.

Teniendo en cuenta lo anterior, en la figura 3.1 se presenta un esquema del sistema propuesto:

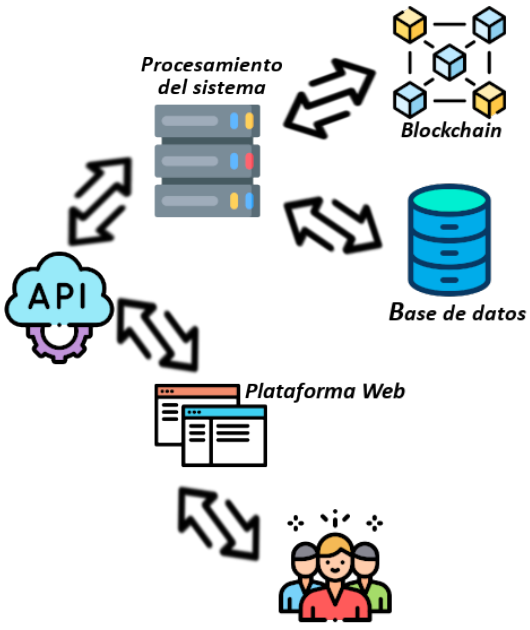


Figura 3.1: Esquema de interacciones.

El sistema contempla la interacción de los usuarios con una plataforma web, la cual se comunica con un sistema de gestión de requerimientos (backend) mediante API. Es gracias al backend que, dependiendo de la calidad y tipo de requerimiento y a través de validaciones lógicas se determina la información a desplegar, pudiendo ser extraída tanto desde la red pública de Ethereum, como de la base de datos del sistema. En cuanto a funcionalidades, se tiene:

- Registro y enrolamiento de responsables académicos.
- Creación y administración de cursos, talleres, títulos y grados según perfil de usario dentro de la plataforma.
- Carga masiva de información relativa a alumnos y asistentes a cursos, talleres, diplomados o grados académicos.
- Proceso de firma y aprobación de información por parte del director académico.
- Emisión de certificados con firma digital de las autoridades responsables y competentes.
- Reconstrucción, visualización y verificación de certificados emitidos a través de portal web de acceso público.
- Almacenamiento persistente y decentralizado de la información contenida de todo documento emitido.

3.3. Debilidades y amenazas

Respecto a las debilidades del sistema propuesto, se distinguen las siguientes:

- Para su uso y adopción, tanto los posibles usuarios finales como las instituciones, deben contar con una alta adopción tecnológica.
- La tecnología blockchain está en evolución constante. Lo que si bien representa una oportunidad, a la vez genera incertidumbre respecto al desempeño y desarrollo futuro.
- La emisión de cada certificado, depende de contar con saldo disponible en la criptomoneda Ether. Dicho instrumento al poseer un precio en extremo volátil puede encarecer la emisión de certificados al punto en que sean económicamente inviables.

Capítulo 4

Definición

Este capítulo sienta las bases técnicas sobre las cuales se desarrollará la plataforma, definiendo los requerimientos a cumplir, el patrón de arquitectura elegido, la manera en que la plataforma va a interactuar con los actores definidos y, por último, los estándares y buenas prácticas a seguir.

4.1. Identificación de requerimientos

El levantamiento de requerimientos se logró gracias a múltiples y periódicas reuniones con la contraparte, en las que se definieron distintas problemáticas a solucionar tanto a nivel administrativo como operacional.

En términos generales, lo más importante es reducir la incidencia de errores humanos, aumentar la certeza del beneficiario de la certificación y, por último, que terceros podrán validar la existencia y veracidad de los documentos que reciban. Adicionalmente se validaron y ajustaron los perfiles mencionados en el punto 2.2. Todo lo anterior decantó en procedimientos funcionales alineados con las necesidades del cliente y procesos lógicos útiles, entre ellos:

- Flujo de información y firmado de la misma
- Modalidad y formato de carga de información relativa a la certificación.
- Forma de creación y entrega del certificado
- Interacción de terceros ajenos a la plataforma

Con los puntos anteriores, se definen:

4.1.1. Requerimientos funcionales

- **Permitir acceso y uso del sistema con funcionalidad segmentada al rol de cada usuario**. Debe existir al menos una instancia de acceso a la plataforma con credenciales establecidas por un correo institucional y una contraseña personal. Cada usuario tendrá accesos a las funcionalidades mínimas que requiere para la ejecución de su trabajo y alineadas con su respectivo rol.
- **Permitir la creación y administración de cursos, talleres, títulos y grados**. Todo alumno receptor de algún documento emitido a través de la plataforma debe estar asociado a un curso o plan de estudios determinado, a su vez, estos deben poder ser gestionados.
- **Almacenamiento, disponibilización y recuperación de certificados y/o documentos emitidos, a todo evento**. Contar con un método de almacenamiento persistente e independiente de lo que la institución pueda proveer y que garantice la fácil consulta de estos y segura reconstrucción.
- **Validación identitaria a todo actor relevante al proceso de certificación y emisión de documentos**. Los usuarios que tienen la facultad para participar en el proceso de emisión de documentos contarán con 2 instancias de autenticación independientes entre si, la primera será para entrar al sistema y la segunda en el momento de realizar la firma digital.
- **Validación de certificados de forma ágil y unívoca para cualquier parte interesada, incluido y no limitado, a actores externos a la plataforma**. Los documentos tienen asociados un identificador único que el cual al ser consultado en la plataforma, inicia un proceso de recuperación de información almacenada en la red distribuida, lo que reconstruye y despliega el documento para el usuario.

4.1.2. Requerimientos no funcionales

- **El sistema debe ser confiable en términos de seguridad de la información:** La plataforma cuenta con certificado SSL brindado por *Let's Encrypt* [11], una Autoridad Certificadora (CA) que brinda de forma gratuita, automatizada y abierta certificados digitales para que sitios web puedan funcionar bajo el protocolo HTTPS. Esto permitirá que exista verificación a nivel de control de acceso y confidencialidad.
- **El sistema debe ser intuitivo para el usuario:** La construcción lógica y gráfica del sistema es de acuerdo a flujos definidos junto al cliente y sigue convenciones de UI/UX validadas por un asesor competente.
- **Debe ser accesible tanto en navegadores de escritorio como en navegadores móviles:** La plataforma cuenta con un desarrollo orientado a las prácticas ceñidas por la corriente *Mobile First* [12],

esto permite construir un aplicativo completamente funcional y accesible no solo para el computador, sino también para dispositivos móviles.

- **Requiere un almacenamiento redundate de la información y documentación emitida:** Se desarrolla un sistema de conexión a una red decentralizada de almacenamiento de información.

- **El sistema debe estar disponible siempre que el cliente lo requiera y funcionar de forma correcta**. Para manejar adecuadamente esto, el sistema es sustentado en una arquitectura híbrida, donde la lógica se encuentra igualmente replicada en al menos 6.498 nodos Ethereum [13], mientras que la información almacenada está distribuida entre 9.879 nodos IPFS [3].

4.2. Patrones de software

Para mantener consistencia y orden en el software a desarrollar, se tienen en cuenta tanto patrones de diseño, como de arquitectura de software. Lo que permite velar por el sano proceso de desarrollo y mantenimiento de código. Al respecto se tiene en cuenta, como mínimo, lo siguiente:

- **Patrón M.V.C.(*Modelo Vista Controlador*)** [14]. Siendo parte de una arquitectura basada en capas [15], es una de las más comunes como estándar *de facto*. En esta arquitectura, los datos se definen en el Modelo, se procesan en el Controlador, y el usuario interactúa con ellos mediante la Vista.

- **Patrón M.V.V.M. (*Modelo Vista - Vista Modelo*)**. Al igual que el patrón M.V.C., el patrón M.V.V.M. utiliza distintas capas para separar la información [16], generando una mayor interacción y fluidez entre la interfaz y el usuario.

- **REST**. La arquitectura REST define los elementos de la web como recursos [17], permitiendo las operaciones básicas de éstos (CRUD) mediante los métodos HTTP (*POST*, *GET*, *PUT*, *DELETE*) plenamente estandarizados y definidos.

4.3. Interacción con dispositivos

A continuación se definen los componentes que se requieren para la interacción de los administradores y clientes con el sistema, acorde a lo recabado por el equipo *Trustify*.

- **Navegador Web**. Debido a que la propuesta de solución contempla el desarrollo de una aplicación web, es de vital importancia un navegador web con acceso a Internet. De esta forma, tanto administradores como clientes y terceros pueden interactuar con la plataforma.

- **Metamask** [19]. Elemento requerido para la segunda instancia de verificación de identidad, habilita la posibilidad de firmar la información de manera digital. Es un complemento de navegador que al conectarse con un nodo ethereum, permite interactuar con dApps [20] de la red ethereum, representando a un actor por medio del par clave pública - privada. Para efectos de esta plataforma, la dApp es el contrato inteligente que contiene las lógicas de firma y emisión.

4.4. Estándares y buenas prácticas

Para la comunicación entre el equipo técnico a cargo del presente trabajo, se definen a continuación los siguientes estándares y buenas prácticas.

4.4.1. Arquitectura de Diseño

El diseño de la arquitectura para la plataforma consta de 5 componentes principales:

- **Almacenamiento Distribuido:** Sistema de almacenamiento que replica y distribuye la información a lo largo de componentes geográficamente separados.

- **BackEnd:** Unidad de procesamiento, manejo de escritura y lectura a la base de datos. Posibilita interacciones con el smart contract, como también la creación de los documentos usando como generador el lenguaje descriptivo. Corre en el servidor. LaTeX.

- **Base de datos:** Almacena información de carácter modificable. Debido a su naturaleza, permite tratar la información en los pasos previos a la escritura en el contrato inteligente e incorporación al blockchain.

- **Blockchain:** Sistema decentralizado, resiliente y altamente persistente. Se caracteriza por la inalterabilidad de la información y lógicas almacenadas en él, razón por la cual es en este componente donde se encuentra la lógica para autorizar y gestionar perfiles de firma de documentos, punteros a certificados emitidos y firmantes autorizados en la plataforma. Lo anterior está completamente embebido en este sistema a través de un contrato inteligente.

- **FrontEnd:** Componente encargado de la visualización, despliegue e interacción entre la información y el usuario final. Es ejecutado en la máquina del usuario, específicamente en el navegador web.

4.4.2. Descripción de componentes tecnológicos

Almacenamiento distribuido

El sistema encargado de gestionar, administrar e implementar esta característica es IPFS. Dicho sistema se caracteriza por presentar una excelente compatibilidad con ethereum, ser decentralizado y además de código libre. Al cierre de este escrito, la información es distribuida a lo largo de 9.789 nodos interdependientes entre sí.

BackEnd

Para el desarrollo del backend se escogió usar el framework AdonisJS, dicha decisión se tomó teniendo presente la simplicidad que brinda para la gestión adecuada de threads, permitiendo crear procesos de firma, emisión certificados y envío de certificados emitidos. Adicionalmente, se tuvo en cuenta la facilidad para generar invocaciones asíncronas de forma nativa y la alta compatibilidad documentada con Web3 [21] (sistema de conexión y comunicación con la red blockchain de Ethereum).

Base de datos

Se escogió una base de datos relacional, específicamente MySQL 5.7, esto por las relaciones lógico-jerárquicas que existen entre los datos y los procedimientos que se cargan en la plataforma.

Blockchain

Blockchain, desde el punto de vista más básico, es una estructura de datos similar a la una lista simplemente enlazada, cuya principal diferencia es que en un blockchain, solo es posible agregar bloques de información, siendo imposible eliminarlos. A nivel de implementación de las tecnologías, existen dos grandes grupos:

- **Blockchain Públicos:** Se caracterizan por el hecho de que cualquier computador puede ser parte de la red, sin la necesidad de contar con ningún tipo de autorización para ser parte de la red. La principal ventaja de este tipo de blockchain radica en los mecanismos de incentivo para conectar computadores a la red lo que aumenta la decentralización propia del sistema. Como debilidad, se distingue el bajo rendimiento transaccional, en el caso de ethereum en torno a las 15 transacciones por segundo.
- **Blockchain Privados:** : También conocidos como consorcios, son redes blockchain para las cuales se debe contar con algún tipo de validación para participar y, para mayor diferencias respecto a las redes públicas, el rendimiento transaccional es mayor, existiendo implementaciones capaces de alcanzar las 25.000 transacciones por segundo.

Considerando lo antes mencionado, lo que sumado al hecho de que es importante la permanencia de la información, para llevar a cabo este desarrollo se escogió la red pública de Ethereum. Entre las principales razones se cuentan:

- Gran comunidad Open Source, evidenciada en más de 50.000 estrellas y 19.000 forks en GitHub
- Soporte a ejecución de lógicas Turing completas, por medio de contratos inteligentes.
- Globalmente distribuido y replicado en 6.498 nodos.
- Lógica decentralizada, la que permite sustentar la persistencia de la información y tolerancia a fallas sistémicas.

Se espera que durante el año 2022, se genere una actualización importante a la red. Teniendo el potencial de alcanzar 100.000 transacciones por segundo y reducir el gasto de energía eléctrica en un 99 %.

FrontEnd

En el caso del frontend, la elección fue VueJs, principalmente por la extensa y detallada documentación disponible de esta herramienta. También se tuvo en cuenta la excelente compatibilidad que presenta con los distintos navegadores, la curva de adopción con respecto a otros frameworks y su modular forma de construcción, completamente basada en componentes, lo que facilita la encapsulación de funcionalidades de la plataforma, además entregar mayor flexibilidad de cara al escalamiento de la plataforma, tanto a nivel nuevas funcionalidades como mejora de las existentes.

4.4.3. Protocolos de comunicación

Una vez definidos la interfaz de usuario y la lógica de negocios, solo queda definir cómo se ha de llevar a cabo la forma de comunicación, escogiéndose lo siguiente:

- **API - RESTful**
  Dentro del desarrollo web, una *API* Permite definir el formato de peticiones y respuestas. Las peticiones son recibidas por los controladores en la capa de negocios y entregan respuestas que son recibidas por la interfaz de usuario.
- **JSON / RFC 8259**
  El estándar *JSON* [22] describe la sintaxis para el intercambio de información. Dicho estándar se utiliza como formato para el envío y recepción de datos del sistema.

4.4.4. Control de versiones

Para una estructuración y gestión del código fuente desarrollado por el equipo Trustify, se utilizará GIT, a través de *GitHub*.

Capítulo 5

Diseño de la solución

Este capítulo describe los aspectos de diseño técnico para el prototipo de la solución, apoyándose diagramas de formato *UML* [23].

5.1. Diagrama de flujo de datos

La siguiente figura explicita los principales módulos del sistema y la circulación de los flujos de información. Cada módulo es referenciado por medio de su nombre corto o abreviatura.

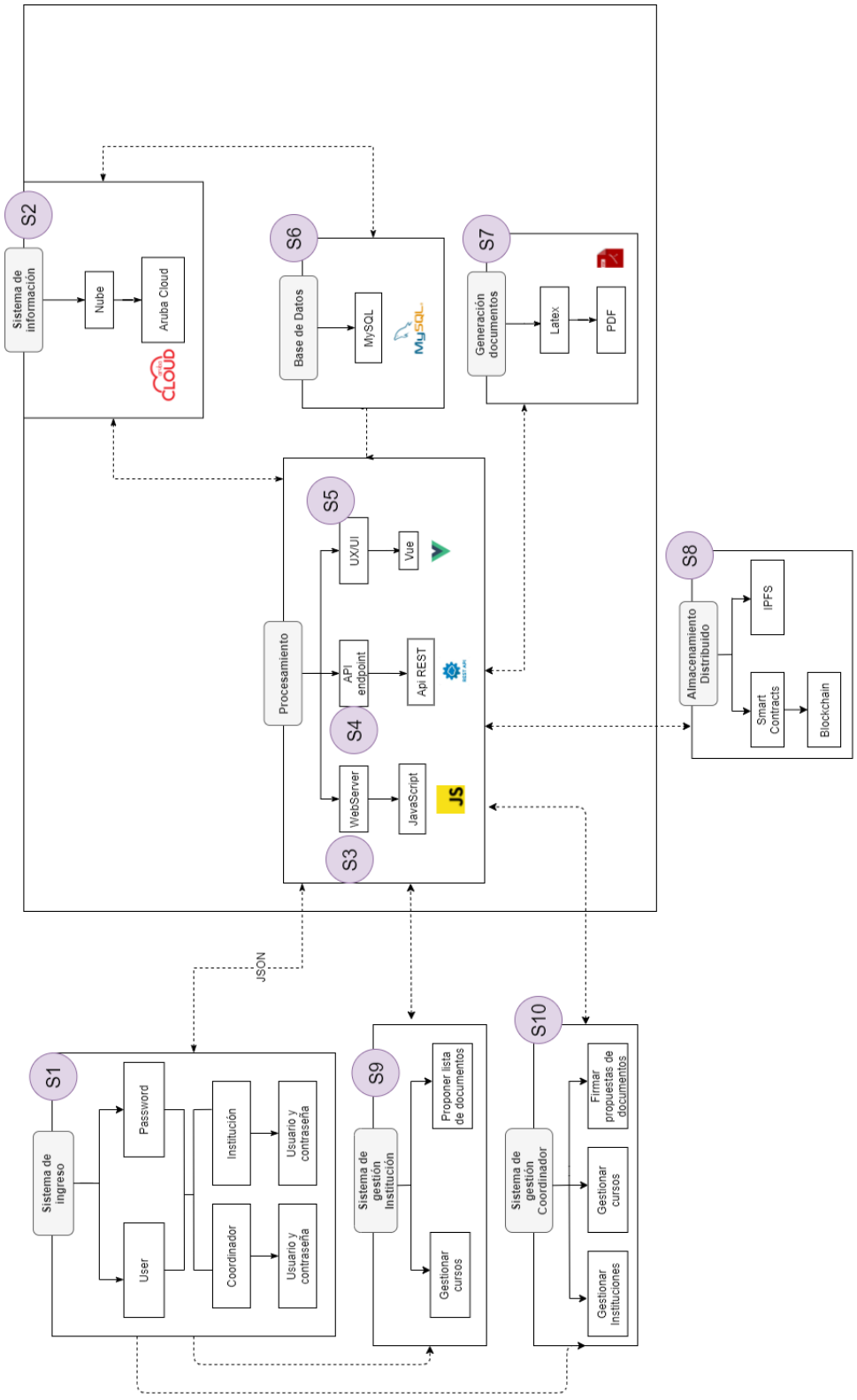


Figura 5.1: Diagrama de flujo de datos del sistema

5.2. Descripción de módulos

A continuación se describen los componentes mostrados en 5.1. Además de mencionar su funcionalidad y interacción con otros componentes y el sistema.

- **Módulo 1: Sistema de Ingreso (S1)**: Componente que, además de autenticar a los usuarios, verifica el estado *activo* que debe tener la cuenta en la plataforma. También concede acceso de acuerdo al perfil de las credenciales provistas. Este componente consta de dos partes; la primera, desarrollada en el Front-End para la interacción usuario - sistema y; la segunda parte en el Back-End, encargada del manejo, validación y consulta del proceso de login.

- **Módulo 2: Sistema de Información (S2)**: Representa la capa de despliegue a nivel de infraestructura del sistema, permite la conexión a internet y dispone recursos físicos que hacen posible el uso y acceso a la plataforma.

- **Módulo 3: Servidor Web (S3)**: Se encarga de gestionar las transacciones de los usuarios autenticados, procesándolas según su perfil -coordinador e institución instructora- y externos, permitiendo o impidiendo la realización de acciones para cada una de las partes. El servidor web se conecta directamente con el Sistema de Información y la Base de Datos, llevando a cabo toda la lógica descrita por el producto.

- **Módulo 4: Conexión API (S4)**: La API es la encargada de ofrecer puntos de conexión compatible entre sistemas, posibilitando una interacción ordenada y una integración tecnológica completa. Dicha conexión no solo actúa como puente entre el Servidor Web y la Interfaz de Usuario (UX/UI), sino también provee una interfaz adecuada para el sistema en su conjunto.

- **Módulo 5: Interfaz de Usuario UI/UX (S5)**: Capa de presentación hacia el usuario, se encarga de mostrar en forma gráfica las opciones disponibles a realizar dentro de la plataforma o servicio. Se comunica con el Servidor Web y la conexión API.

- **Módulo 6: Base de Datos (S6)**: Encargada de almacenar toda la información con respecto al sistema y estado del mismo. Contiene credenciales de acceso al sistema, coordinadores inscritos, las instituciones instructoras inscritas, los roles y permisos asociados a cada uno, la información de cursos, alumnos y documentación emitida. Además almacena la dirección donde están guardados los documentos generados para tener acceso rápido a ellos cuando sean requeridos.

- **Módulo 7: Generación de Documentos (S7)**: Servicio de generación dinámica de los documentos a emitir. Se conecta con el Servidor Web para recibir la información que se debe incluir en el documento o certificado, compila el documento LaTeX, generando un PDF, permitiendo la descarga del mismo.

- **Módulo 8: Almacenamiento Distribuido (S8)**: El módulo de almacenamiento distribuido contiene el contrato inteligente, componente en el que se realiza la verificación de identidad de los usuarios con facultad de firmar digitalmente los documentos, también contiene las reglas que controlan el flujo de firma de los documentos y las lógicas del almacenamiento del resultado de la función de Hashing en la red distribuida de Ethereum. Por último almacena un puntero a la información firmada por los usuarios competentes y almacenada en la red IPFS para asegurar la persistencia.

- **Módulo 9: Sistema de Gestión de Institución (S9)**: Este módulo permite al responsable de la institución gestionar los cursos o instancias de enseñanza, informar la lista de alumnos con sus respectivas calificaciones y firmar la información subida para ser presentada al director académico.

- **Módulo 10: Sistema de Gestión Coordinador (S10)**: Este módulo permite al coordinador académico ver los cursos que contienen información adjuntada por los responsables académicos, permite visualizar la lista de alumnos y sus respectivas calificaciones y proporcionar la respectiva firma para habilitar el proceso de emisión y almacenamiento de documentos.

5.3. Arquitectura del sistema

Considera un esquema híbrido entre sistemas centralizados y decentralizados. Pues, a nivel de pre-procesamiendo, toda la información se mantiene centralizada, mientras que la información resultante como producto de procesos, lógicas de negocio, firmado y almacenamiento persistente es hecho en ethereum e IPFS. Esto se muestra en la figura 5.2:

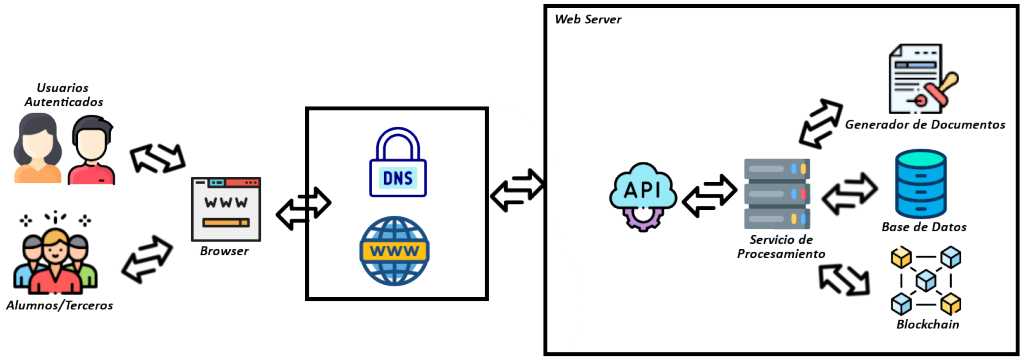


Figura 5.2: Diagrama de Arquitectura - Trustify

De la figura anterior, y repasando de izquierda a derecha, se tiene:

- Los usuarios, tanto autenticados como alumnos/terceros, pueden interactuar con la plataforma por medio de un navegador web que se ejecute tanto en un smartphone como en un computador. Sin embargo, para llevar a cabo el proceso de firma y emisión de certificados, lo recomendable es que sea desde computador. Al ser una plataforma web, es necesario contar con conexión a internet.

- Respecto al sistema "Web Server" la forma en que se comunica el navegador web, con el "Servicio de Procesamiento" es a través de API-Rest. El resto de los componentes interactúan de la misma forma y se representan, en términos simplificados, alojados una infraestructura física o lógica, llamada "Servicio de procesamiento".

- Todo documento que sea generado por el sistema como comprobante de la finalización de un proceso de firma y autorización y emisión es realizado por el generador de Documentos. Los documentos reconstruidos a través de la consulta de un tercero por su código único también son gestionados por este componente del sistema Trustity.

- La base de datos almacena toda la información temporal del proceso de firma y emisión de documentos, además de toda la información correspondiente a los usuarios autenticados en el sistema.

- Una vez realizado de forma satisfactoria el proceso de firma y emisión de documentos la información que contienen es almacenada en IPFS, aprovechando la estructura que posee como servicio decentralizado de distribución y almacenamiento de archivos, mientras que la referencia a dicho archivo es alojada en el contrato inteligente, permitiendo la validación de la misma al ser comparada.

5.4. Modelo de datos

La forma en que se encuentran organizados los datos locales, y la manera en que se relacionan entre ellos, se describen en la siguiente figura 5.3. E diagrama de datos está enfocado en el manejo de información para la autenticación de usuario con sus respectivos roles, los cuales les dan las facultades solicitadas para actuar en el proceso de firma y emisión de documentos. También almacena la información temporal de los documentos a emitir mientras se realizan las firmas por parte de los responsables.

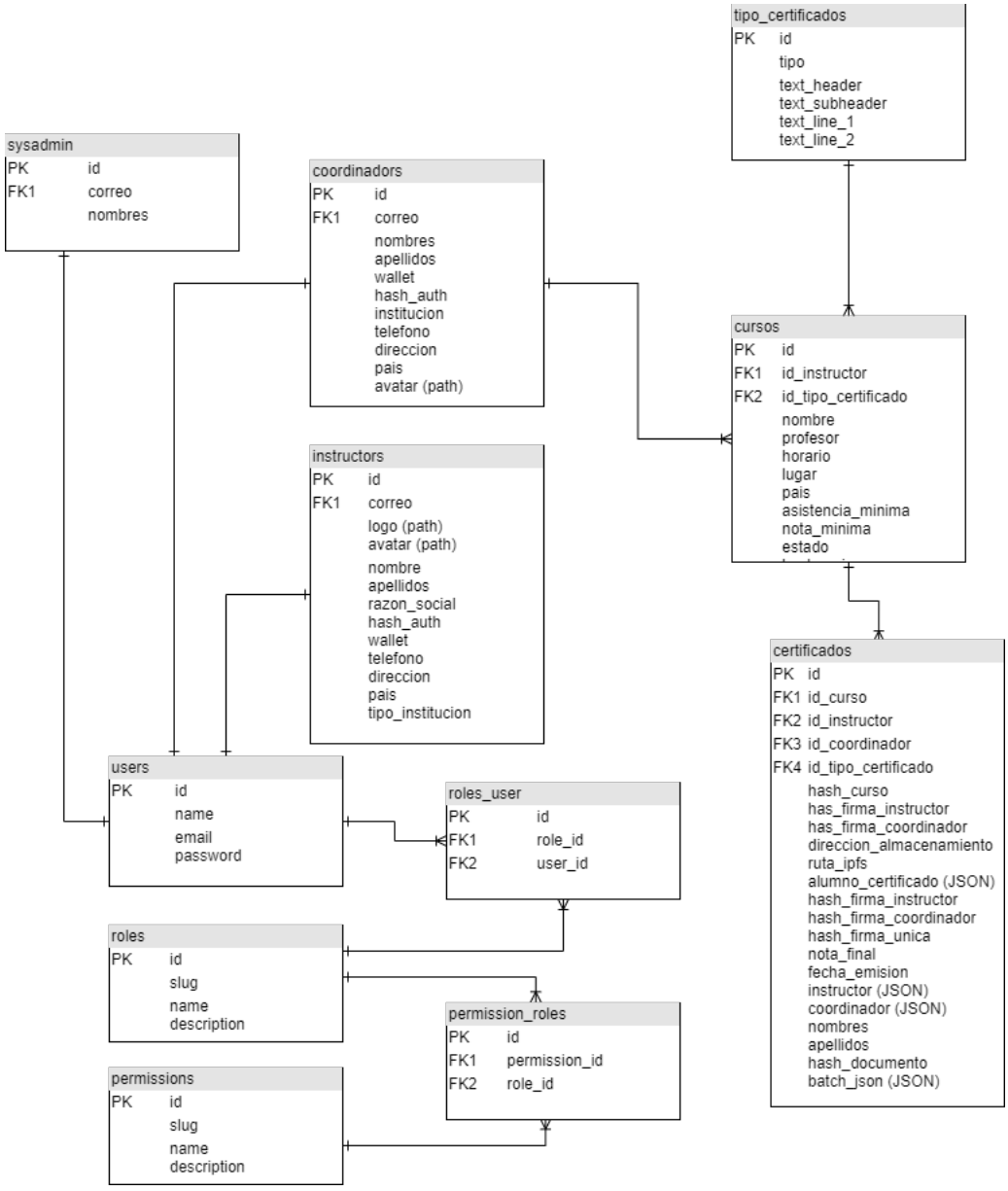


Figura 5.3: Modelo de Datos - Trustify

5.5. Diseño de Interfaces

Los *mockups* de los diseños para la interfaces, se encuentran dentro del Anexo A.1, desde la página 61. Éstos diseños corresponden a las vistas generales de la aplicación a desarrollar, abarcando en gran medida los requerimientos funcionales descritos en el Capítulo 4.

5.6. Diagrama de casos de uso

Se presenta una visión general de las acciones que se pueden realizar en la plataforma Trustify, tanto para usuarios con acceso al sistema interno de administración y gestión, como también para los alumnos receptores de los documentos y terceros que requieran validar la información de los documentos recibidos y que fueron emitidos por la plataforma. En la figura 5.4 se definen las acciones que pueden realizar los 3 tipos de actores que pueden interactuar con el sistema, repasando el hecho de que los usuarios coordinadores e instructores son responsables de llevar a cabo el proceso de firma y emisión de documentos; mientras que los alumnos o terceros solo pueden consultar la existencia de un documento ya emitido.

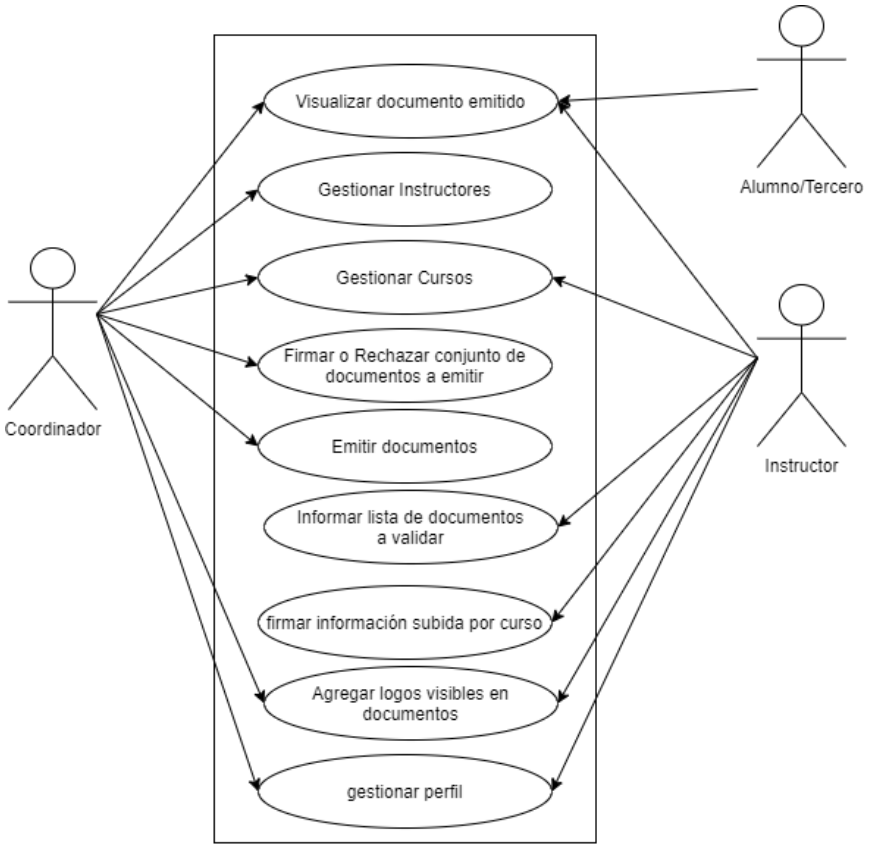


Figura 5.4: Casos de uso usuarios Trustify

5.7. Contrato inteligente

Un blockchain segunda generación actúa como una máquina virtual -similar a la de Java- capaz de ejecutar instrucciones lógicas y almacenar datos. Por su parte, un contrato inteligente (en inglés Smart contract) es un programa que está almacenado en un blockchain y, al igual que cualquier otro programa computacional, ejecuta rutinas y acciones destinadas a realizar tareas o comprobaciones al momento en que las condiciones de ejecución se cumplen. El uso típico de este tipo de programas es para automatizar ejecuciones de acuerdos entre participantes que esperan obtener alguna señal de respuesta de parte del sistema, todo en un ambiente donde el único intermediario es una tecnología.

En el caso de la red pública de Ethereum, la ejecución lógica puede ser paga o gratuita. El factor diferenciador entre ambos casos, radica en si la lógica resguardada en el contrato inteligente almacena o lee información en el blockchain, siendo pagadas aquellas ejecuciones que escriben. El pago de las mismas se hace a través de Ether, la criptomoneda de la red pública de Ethereum.

Actualmente, la emisión de un solo certificado gestionado por medio de la plataforma Trustify tiene un costo de 32.936 unidades de GAS. Haciendo las conversiones pertinentes y considerando que el precio por unidad de GAS al momento de es momento de escribir esta memoria es de ETH 0.000000029, nos da un costo aproximado de *CLP 2.313* [18] por certificado emitido.

5.7.1. Contrato Trustify

Para lograr la emisión según las reglas de negocio,se desarrolló un contrato inteligente en Solidity [24]. Dicho contrato se vale de una dirección que ejecuta acciones de manera automatizada, denominada **autosigner**. A continuación se muestra el prototipo de cada función ejecutada en el contrato inteligente.

```
1  pragma solidity ^0.5.0;
2
3  contract Trustify {
4
5      /******************
6       dA* Variables Globales
7       ******************/
8      address public owner; // Variable que indica la dirección dueña del contrato.
9
10     bool public onPause; // Variable lógica que indica si el contrato ha sido pausado
           o no.
11     address public pauser; // Variable que indica dirección que puede pausar el
           contrato.
12     dA dirección "autosigner".
13
```

```
14    mapping (address => uint8) coordinators; // Diccionario que indica si es valida o
          no una dirección de coordinador.
15    mapping (address => uint256) coordinatorToListID; // Diccionario que relaciona a
          un coordinador con las instituciones que maneja.
16    mapping (uint256 => mapping (address => uint8)) institutions; // Diccionario que
          relaciona los id institucionales con las direcciones y sus correspondientes
          estados de validez
17
18    uint256 public currentListID; //
19
20    // Constants
21    uint8 constant _NOT_AUTHORIZED_ = 0; // Constante que indica si una direccion no
          ha sido autorizada como coordinador
22    uint8 constant _AUTHORIZED_     = 1; // Constante que indica si una direccion ha
          sido autorizada como coordinador firmante
23    uint8 constant _WAS_AUTHORIZED_ = 2; // Constante que indica si una direccion fue
          autorizada como coodinador firmante. En este estado, el coordinador no puede
          firmar, pero indica validez en firmas antiguas. Además al cambiar esta, un
          coordinador puede ser re-autorizado.
24
25    /******************
26     * Eventos
27     ******************/
28    event transferedOwnership(address _from, address _to); // Avisa la transferencia
          de dominio del contrato.
29    event changedPauser(address _from, address _to);// Avisa la transferencia de la
          facultad de pausar el contrato.
30    event authorizedCoordinator(address _coordinatorWallet, uint256 currentListID);//
          Avisa la autorización de un coordinador.
31    event deauthorizedCoordinator(address _coordinatorWallet, uint256 currentListID);
          // Avisa la desautorización de un coordinador.
32    event authorizedInstitution(address _coordinatorWallet, address _institutionWallet
          );// Avisa la autorización de una nueva institución.
33    event deauthorizedInstitution(address _coordinatorWallet, address
          _institutionWallet);// Avisa la desautorización de una institución
34
35    event assignedAutoSigner(address _autosignerWallet);
36    // Avisa la asignación de la dirección autosigner
37    event unassignedAutoSigner(address _autosignerWallet);U
38     // Avisa la desasignación de la dirección autosigner
39
40    //
          ----------------------------------------------------------------------------------------------------
```

```
41      event newCertificate(address _coordinator, address _institution, string
            _ipfsAddress); // Avisa la emisión de un nuevo certificado, señalando la
            dirección de coordinador, la dirección de institución y la dirección de IPFS
            del certificado.
42      //
            ----------------------------------------------------------------------------------------------------


43  }
```

5.8. Flujo de emisión de documento

La lógica principal del sistema está contenida en los pasos a seguir y requisitos a cumplir para determinar que un documento fue válidamente emitido. Para esto se considera la interacción que los distintos usuarios del sistema tendrán con la plataforma y cómo esta interacción incide en que este documento pase sus fase de validación hasta que se le entrega, vía correo electrónico, a los alumnos receptores. En un primer paso instructor o el coordinador debe crear las instancias de enseñanza denominadas como *cursos*. Estos cursos contienen una lista de alumnos enrolados en él y, una vez finalizado, se informa a la plataforma las calificaciones y, bajo los criterios definidos en la creación del curso, se categoriza a los alumnos como aprobados o no aprobados. Luego de subir esta información el instructor debe proveer su firma correspondiente con su wallet autenticándose con la herramienta Metamask. Ya finalizado esto se le informa al coordinador que tiene cursos con información de alumnos para revisar y emitir o rechazar la emisión de los respectivos documentos. Luego de contar con la firma del coordinador, se guarda la metadata en IPFS y, a su vez, en el respectivo bloque del blockchain se resguarda la referencia de dicha metadata, logrando almacenar la información en forma persistente. Por último, mediante correo electrónico se le hace llegar a los alumnos una *URL* que permite descargar en formato PDF el respectivo certificado.

En cuanto al uso, los alumnos cuentan con un certificado fácilmente compartible, mientras que un tercero -cualquiera- puede verificar de manera pública, segura y sencilla la autenticidad del certificado. Todo a través de un proceso de reconstrucción del documento en base a la información que almacenada en IPFS y referenciada en el blockchain por medio del contrato inteligente.

A continuación, en la figura 5.5 se muestra el flujo lógico de la solución. Cabe mencionar que este flujo condensa las lógicas implementadas tanto en el backend de la Trustify como en el contrato inteligente en si.

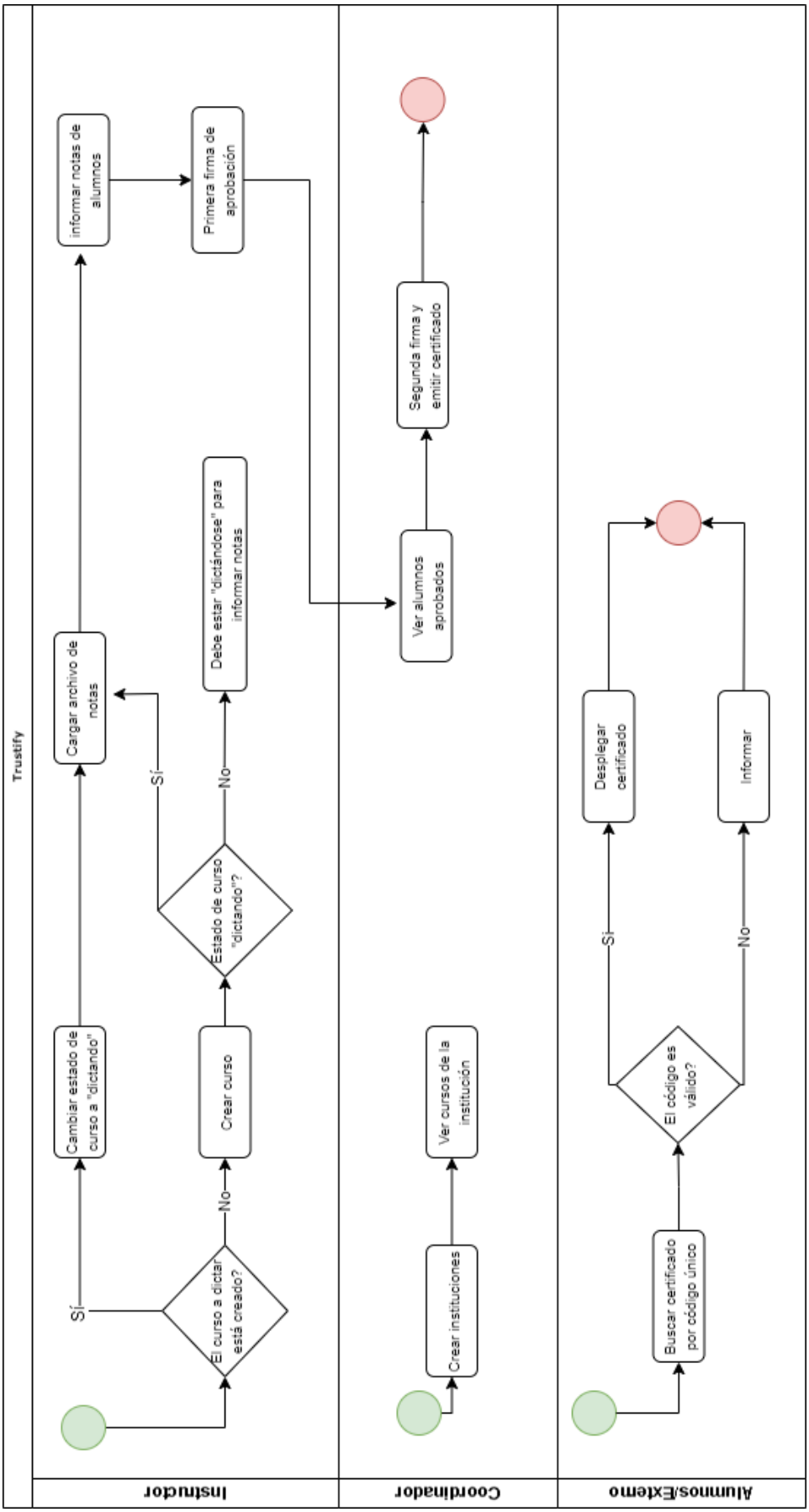


Figura 5.5: Diagrama BPMN de proceso de firma y emisión de documentos

Capítulo 6

Validación

El presente capítulo presenta el resultado obtenido de la implementación, frente a la contraparte que implementó el uso de la plataforma, además de resultados obtenidos frente a pruebas de seguridad y desempeño.

6.1. Validación de DigitalBank Transformación Digital

La validación realizada por el cliente DigitalBankTD, consistió principalmente en el *feedback* otorgado por los principales usuarios que actuaron como Coordinador e Instructor respectivamente. Durante un periodo de 2 meses, que incluyeron un acotado periodo de enseñanza y ambientación en la plataforma. En dichas reuniones se probó el funcionamiento de la plataforma.
Se presentó el prototipo funcional de la plataforma, mostrando en detalle el desarrollo realizado tanto el descrito en la presente memoria. Para finalizar el desafío y la solución propuesta, se presenta una encuesta de validación y usabilidad, a los administradores y usuarios de DigitalBankTD.

Criterio	Evaluación (1: Muy malo - 5: Excelente)
Usabilidad de las funciones	4
Rapidez de la aplicación	4
Facilidad de uso	4
Uso de Metamask para firmar	3

Tabla 6.1: Tabla de resultado encuesta de validación

Se puede concluir que, en términos de uso, la contraparte ha quedado satisfecha con el trabajo realizado. Se ha mostrado gran interés en mantener el servicio y pasarlo de piloto o marcha blanca a producción con sus respectivos acuerdos económicos y ajustes requeridos, detallándose en la sección Trabajo Futuro del Capítulo 8.

6.2. Análisis de seguridad

Uno de los requerimientos no funcionales, solicitados por la contraparte para el desarrollo del proyecto, es que el sistema debe operar bajo un protocolo seguro. Para la validación de dicho requerimiento, el sistema se ha sometido a múltiples análisis de seguridad, los cuales evalúan los principales parámetros requeridos por los sitios web actuales, con el fin de evitar el riesgo de sufrir ataques malintencionados.

El análisis realizado por Qualys, Inc, califica el sitio web a partir del tipo de certificado de seguridad proporcionado, el soporte a protocolos para la conexión, los tipos de cifrado disponibles y la fuerza que tienen éstos. Por consiguiente, la evaluación general otorgada por Qualys, Inc ha sido **A+**, la nota máxima obtenible. Esto refleja las siguientes características:

- El certificado proporcionado por la aplicación es válido.
- El certificado fue otorgado por una institución de confianza (*Let's Encrypt Authority X3*).
- La forma de conectarse es compatible con la gran mayoría de algoritmos disponibles.
- El protocolo de conexión cubre las principales vulnerabilidades conocidas.

6.3. Análisis de rendimiento

Una sección que es importante en la fase de validación, es el rendimiento y el desempeño ofrecido por la plataforma para el funcionamiento de los requerimientos solicitados por la contraparte. Para medir de forma cuantitativa el tiempo de respuesta de la aplicación, se utilizó una herramienta de auditoría llamada *Google Lighthouse* [26], la cual genera un informe resumido de los principales problemas que pueden ocasionar que una plataforma web tenga un desempeño no óptimo.

Usando dicha herramienta, se revisano 2 posibles escenarios:

- En el primero, se visita el sitio web desde un computador de escritorio, sin limitar la velocidad de descarga, y sin limitar la velocidad de procesamiento.
- Para el segundo, el sitio web es consumido desde un teléfono celular, modelo *Huawei P30*, con una velocidad de descarga simulada de una red celular 4G, y castigando la potencia de procesado 4 veces referente al primer escenario.

De las mediciones obtenidas, se hace un promedio simple. Los resultados son expuestos en tabla 6.2:

Criterio	Resultado
Tiempo en dibujar el primer texto o imagen	5.2 segundos
Tiempo en dibujar el contenido principal de la imagen	5.2 segundos
Tiempo en mostrar contenido	6.8 segundos
Tiempo para poder interactuar con el sitio	5.5 segundos
Tiempo de respuesta aplicación - entrada de datos	0.015 segundos
Tiempo de respuesta cliente - servidor	0.415 segundos
Tiempo de ejecución de javascript	0.75 segundos
Tamaño de la página web y sus contenidos (*payload*)	986 kilobytes

Tabla 6.2: Tabla de resultado de mediciones de la aplicación

Considerando que el sistema está construido como una aplicación de única página (*Single Page Application*, en inglés), se puede concluir que:

- La primera vez que la aplicación se cargue en un dispositivo, habrán aproximadamente 5 segundos de espera antes de que comiencen a aparecer los contenidos de la sección. Esto puede generar repercusiones negativas en la experiencia de usuario, que no tendrá conocimiento si es que *"la página está mala"* o *"la página está cargando"*. La solución recomendada por la herramienta de análisis es crear una pantalla de carga, y descargar los contenidos de forma diferida (*Lazy Loading*, en inglés), de esta manera, se mantiene al tanto al usuario de la inicialización de la plataforma. Luego de la primera vez que la aplicación es cargada, las siguientes ocasiones tarda en promedio 2.36 segundos, debido a que los recursos se almacenan en caché.

- El tiempo promedio de interacción entre el cliente y el servidor es de 0.415 segundos, lo que mejora la experiencia de usuario, ya que la interacción con la *API* optimiza el tiempo de petición, enfocándose en entregar los recursos solicitados, en vez de tener que volver a generar una vista.

- Usando los datos recolectados del análisis, y comparándolos con el último reporte anual de Google [25], se puede concluir que el tiempo de carga se encuentra por debajo del promedio en Latinoamérica (13.5 segundos), pero por el encima del tiempo objetivo (menor a 3 segundos), esto se traduce como un sentimiento del usuario descrito como *"es lento, pero tolerable"*. Por otra parte, el tamaño de la aplicación se encuentra por debajo del promedio de Latinoamérica (3.4 MB) y dentro del objetivo recomendado (menor a 1 MB)

Capítulo 7

Resultados

En este capítulo se muestra el desarrollo e implementación de la lógica de negocios de la solución, junto con la descripción de cada uno de los módulos que componen al sistema.

7.1. Entorno de trabajo

El presente proyecto ha sido seccionado en 2 ambientes distintos, enfocado en una metodología de desarrollo ágil [28], y acorde a los requerimientos no funcionales mencionados en el Capítulo 4.

7.1.1. Entorno de desarrollo

El entorno de desarrollo que se utilizó es un híbrido entre un ambiente de contenedores con Docker, donde se implementó el módulo de base de datos con *MySQL 8* y el módulo de Redis. Adicionalmente se utilizó el ambiente nativo de cada computador personal con el software *NodeJs 10.7.0*, *NPM 6.** y *Vue CLI 2.**.

7.1.2. Entorno de pruebas pre-productivas

Con fines demostrativos y de prototipaje, el sistema se encuentra alojado en servidores *Cloud* de *Aruba* [27], empresa operativa en Italia, cuyo negocio consiste en ofrecer soluciones *IaaS* (Infrastructure as a Service, en inglés). Dentro de las principales ventajas, cuenta con:

- Interfaz de usuario sencilla para asignación extra de recursos y escalabilidad de *hardware*
- *SLA* de un 99.80 %
- Precios asequibles, habiendo *VPS* desde EUR 2,79 mensuales.

- Templates de despliegue rápido, en este caso, se escogió: *Ubuntu 16.04*, *NodeJs 10.7.0*, *MySQL 5.7.25* y *Apache 2.4.18*

En cuanto a hardware, para el desarrollo y despliegue de la plataforma, se configuró una *VPS* con las siguientes características:

- Disco duro *SSD* 512 [GB]
- Memoria RAM de 8[GB]
- Procesador virtual de 8 [vCPU]

7.2. Implementación de API

Parte importante del desarrollo de la plataforma se enfoca en que la *API REST* provea un correcto manejo de roles y autenticación del coordinador o instructor, y que a nivel de *FrontEnd* solo se muestre la información correspondiente al rol designado. El siguiente diagrama de secuencias *UML* muestra el comportamiento base para todas las acciones que lleva a cabo este componente:

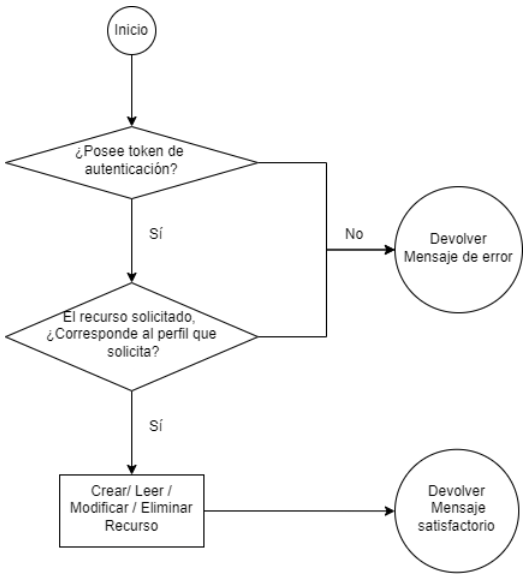


Figura 7.1: Diagrama de secuencia API REST

Las rutas están divididas entre las que son accesibles por el Coordinador, por el Instructor y las que son accesibles por cualquier usuario de forma pública. Para tener una organización de las rutas de la *API*, éstas deben comenzar con el prefijo `/api`. Por otro lado, para hacer efectiva la autenticación bajo arquitectura *REST*, en cada petición *HTTP* debe incluir el parámetro `Authorization`, el cual es un

código alfanumérico generado y almacenado en la base de datos. Este código es utilizado en la interfaz de usuario por *Vue.Js*.

7.2.1. Cursos

La siguiente tabla muestra el método *HTTP*, la ruta, y la acción que realiza la *API* para un Curso.

Método	Ruta	Acción	Usuario
GET	/curso	Retorna todas los cursos asociados a un usuario	Instructor, Coordinador
GET	/curso/:id	Retorna el curso solicitado	Instructor
POST	/curso	Crea un nuevo curso	Instructor
PUT	/curso/:id	Modifica el curso solicitado	Instructor
POST	/curso/estado/:id	Modifica exclusivamente el estado de un curso	Instructor
POST	/curso/finalizar/:id	Modifica el estado de un curso en estado FINALIZADO y sube la información a validar	Instructor
DELETE	/curso/:id	Elimina el curso solicitado	Instructor
GET	/curso/:id/pendientes	Retorna la lista de alumnos pendientes de aprobación por parte del coordinador	Coordinador
GET	/pendiente/cursos	Retorna todos los cursos en estado FINALIZADO y sin firma por parte del coordinador	Coordinador

Tabla 7.1: Tabla de rutas para Cursos

Donde el parámetro `:id` corresponde al identificador único del Curso. Los cursos solo pueden ser finalizados por el Instructor y cuando lo hacen informan la lista de alumnos con sus respectivas calificaciones para la evaluación y aprobación del coordinador. Los estados de los cursos siguen un orden lineal partiendo por el estado de Inscripciones, el siguiente estado es Cancelado o Dictando, donde solo estos últimos pueden ser finalizados; pudiendo todos volver al estado primero de Inscripciones.
Cuando el coordinador rechaza el conjunto de alumnos informados por el instructor el curso pasa nuevamente al estado Dictando para que el responsable adjunte la información correcta y así reiniciar el proceso de firma.

7.2.2. Certificado

La *API* de certificados corresponde las acciones que realiza la *API* para el certificado.

Método	Ruta	Acción	usuario
GET	/certificado-i/buscar	Retorna los certificados asociados a un nombre de alumno	Instructor
GET	/certificado-i/firmados	Retorna todos los certificados firmados por el instructor	Instructor
GET	/certificado-i/ultimos-firmados	Retorna los ultimos 20 certificados firmados por el instructor	Instructor
GET	/tipo_certificado	Retorna todos los tipo certificados	Coordinador, Instructor
GET	/tipo_certificado/:id	Retorna el tipo certificado solicitado	Coordinador, Instructor
POST	/certificado/firmar	Firma un conjunto de documentos	Coordinador
POST	/certificado/rechazar	Rechaza un conjunto de documentos presentados por el Instructor	Coordinador
GET	/certificado/buscar	Retorna los documentos asociados a un nombre de alumno	Coordinador
GET	/certificado/firmados	Retorna todos los documentos firmados y emitidos	Coordinador
GET	/certificado/ultimos-firmados	Retorna los últimos 20 documentos firmados y emitidos	Coordinador
GET	/certificado/ver/:hash	Retorna el documento en formato PDF correspondiente al Hash de verificación	Público general
GET	/certificado/thumbnail/:hash	Retorna una imagen del documento	Público general
GET	/certificado/hash/:hash	Retorna el la información del documento consultado por el Hash	Público general
GET	/certificado/reconstruct/:hash	Retorna la información del documento consultado por el Hash después de pasar el proceso de extracción desde la red distribuida y reconstrucción de archivo	Público general

Tabla 7.2: Tabla de rutas para administrar productos

Es importante notar que el campo `hash` corresponde al identificador único del certificado o documento.

7.2.3. Instructor y Coordinador

Para la administración de Coordinadores e Instructores, se proporcionan los siguientes puntos de *API*, que se detallan en la tabla a continuación.

Método	Ruta	Acción	Usuario
POST	/register/instructor	Crea un nuevo instructor en la base de datos	Coordinador
GET	/instructor/verify_hash	Verifica si el Hash recibido en el correo electrónico es válido para completar la creación de cuenta	Instructor, Externos
POST	/instructor/completar	Crea contraseña y añade wallet de Ethereum al perfil	Instructor
GET	/coordinador/verify_hash	Verifica si el Hash recibido en el correo electrónico es válido para completar la creación de cuenta	Coordinador, Externos
POST	/coordinador/completar	Crea contraseña y añade wallet de Ethereum al perfil	Coordinador

Tabla 7.3: Tabla de rutas para administración de usuarios

Los coordinadores solo pueden ser creados por el administrador de la plataforma. Cuando se crean Coordinadores e Instructores se envía un correo electrónico informando la creación del perfil y solicitando ingresar a un enlace que permite completar la información para terminar con el registro. Es en este momento en que se requiere informar la dirección de la wallet correspondiente al perfil, pues una vez verificada la autorización de la wallet informada, el sistema habilita el proceso de firma. La abstracción aquí mencionada, permite hacer una equivalente entre la wallet y la identidad del usuario que puede generar alguna transacción a través del contrato inteligente.

7.2.4. Misceláneo

Además de las rutas anteriormente mencionadas, han sido implementadas otras rutas *API* que se encargan de administrar otros aspectos de la plataforma, descritos a continuación.

Método	Ruta	Acción	Usuario
POST	/login	Ingreso al sistema con las credenciales de Correo Electrónico y Contraseña	Público general
POST	/parse-xlsx	Recibe un formato de Excel definido y retorna la información contenida en formato JSON	Instructor
GET	/perfil	Obtiene la información de perfil del usuario registrado	Coordinador, Instructor
POST	/perfil	Modifica la información del perfil de usuario	Coordinador, Instructor
POST	/avatar	Modifica la imagen de perfil del usuario	Coordinador, Instructor
DELETE	/avatar	Elimina la imagen de perfil del usuario	Coordinador, Instructor
GET	/estadisticas/certificados	Retorna las estadísticas de la emisión de documentos	Coordinador
GET	/estadisticas/metricas	Retorna las métricas relevantes del flujo de emisión de documentos	Coordinador
GET	/estadisticas-i/certificados	Retorna las estadísticas de la emisión de documentos	Instructor
GET	/estadisticas-i/metricas	Retorna las métricas relevantes del flujo de emisión de documentos	Instructor
POST	/logo	Ingresa el logo a agregar en los documentos, Instructor	
DELETE	/logo	Elimina el logo asociado al instructor	Instructor

Tabla 7.4: Tabla de rutas misceláneas

7.3. Implementación de vistas

A continuación, se muestran las principales vistas que fueron implementadas en el sistema, junto con una breve descripción.

7.3.1. Panel principal Coordinador

Este panel permite permite al Coordinador conocer de forma resumida la información crítica de los documentos emitidos, cantidad de cursos creados por este instructor, cantidad de documentos firmados en la última fecha y el total de cursos que están en el estado de Dictando con la posibilidad de Finalizar en el tiempo correspondiente.

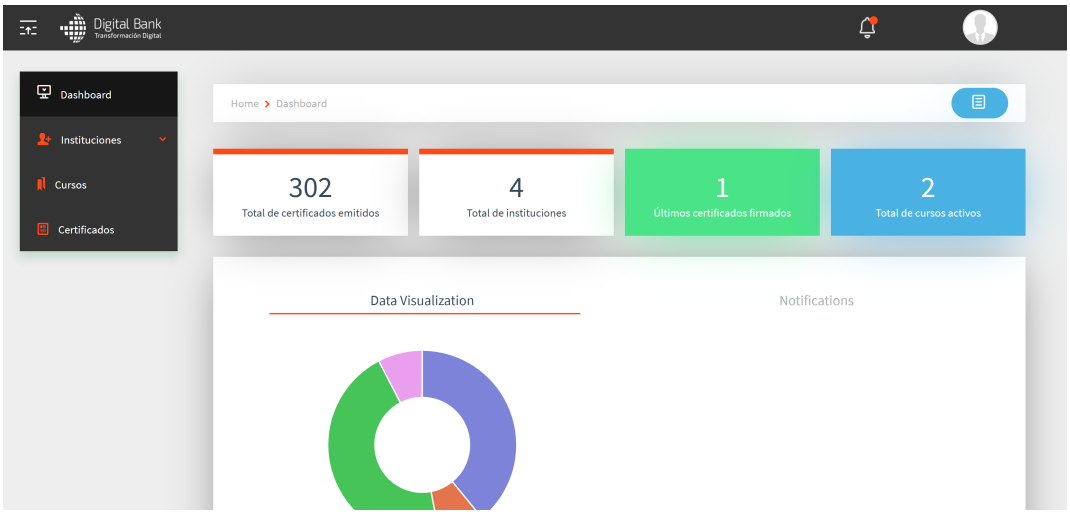


Figura 7.2: Vista del tablero principal Instructor Trustify

7.3.2. Panel principal Instructor

Este panel permite permite al Instructor conocer de forma resumida la información crítica de los documentos emitidos, instrucciones o instituciones habilitados a gestionar cursos, cantidad de documentos firmados en la última fecha y el total de cursos que aún están activos y en el futuro requerirán la aprobación del coordinador.

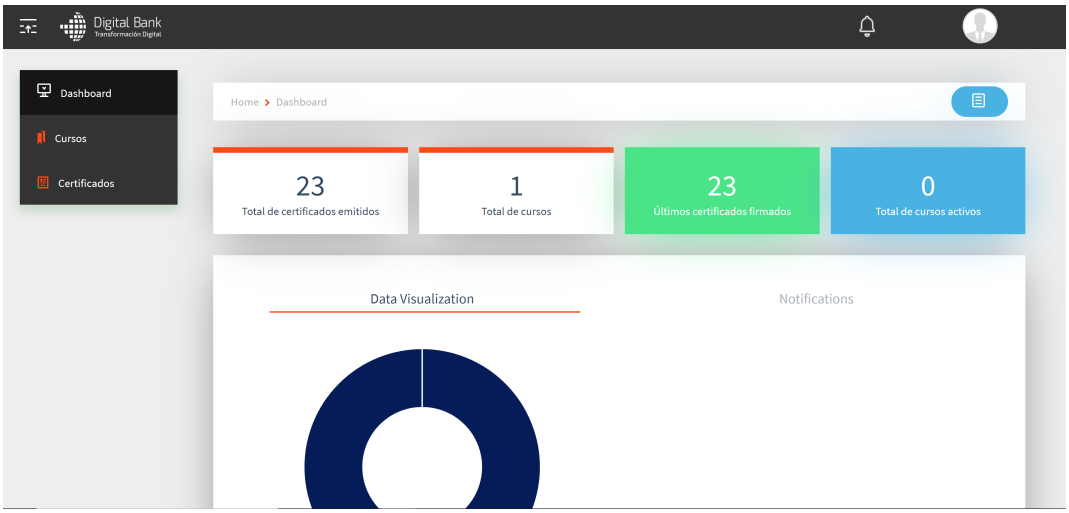


Figura 7.3: Vista del tablero principal Coordinador Trustify

7.3.3. Cursos

Las Figuras 7.4, 7.5, 7.6 y 7.7 muestran las vistas implementadas para la administración de un curso, en las cuales se pueden crear, listar o modificar su información.

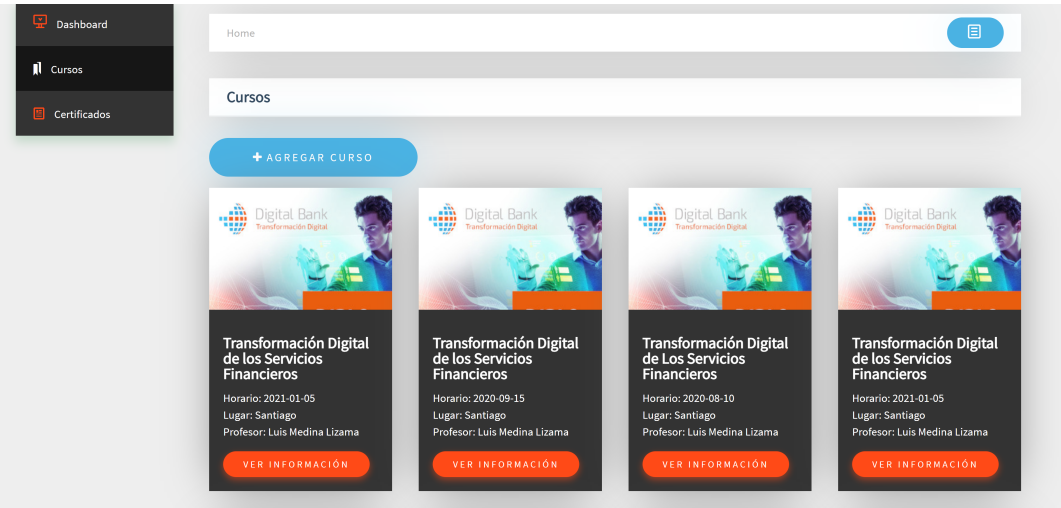


Figura 7.4: Vista de Cursos

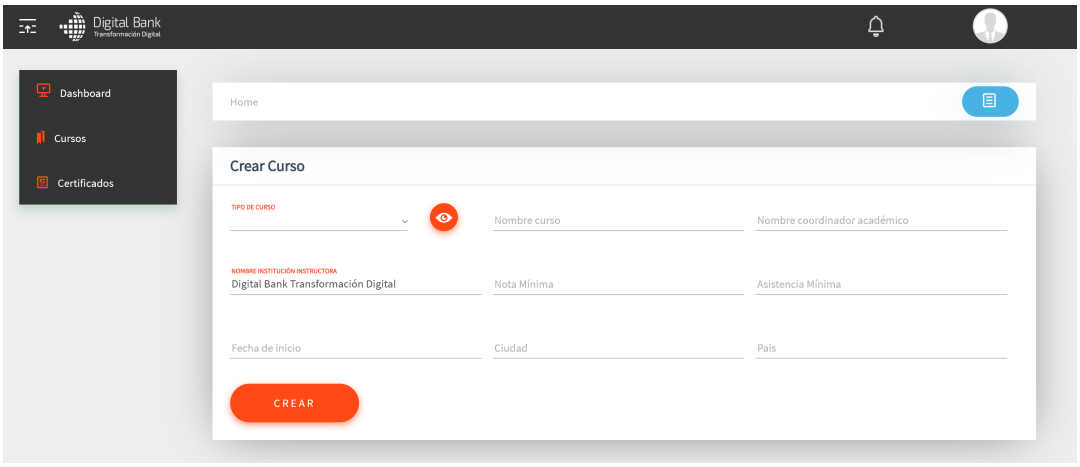


Figura 7.5: Vista Creación de Cursos

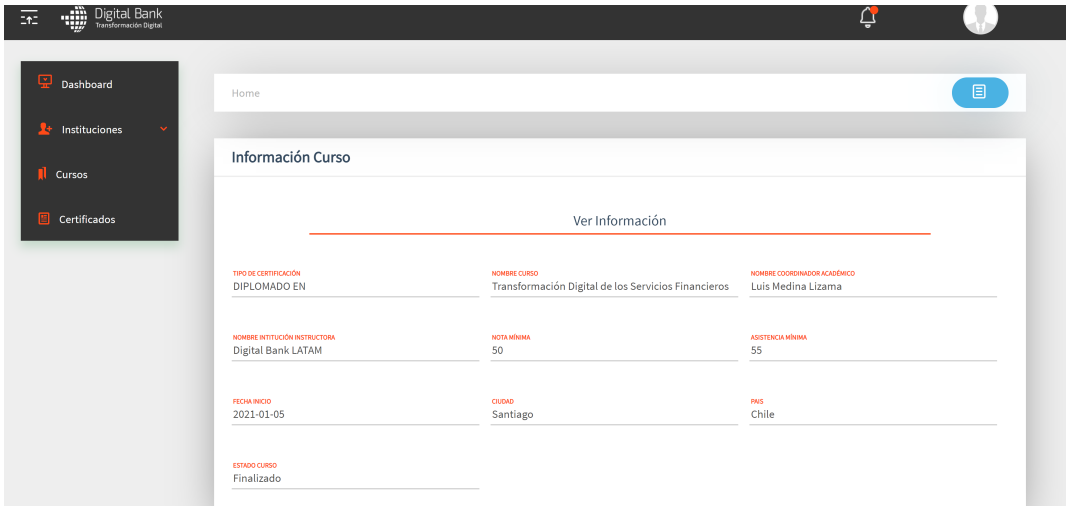


Figura 7.6: Vista Información de Curso

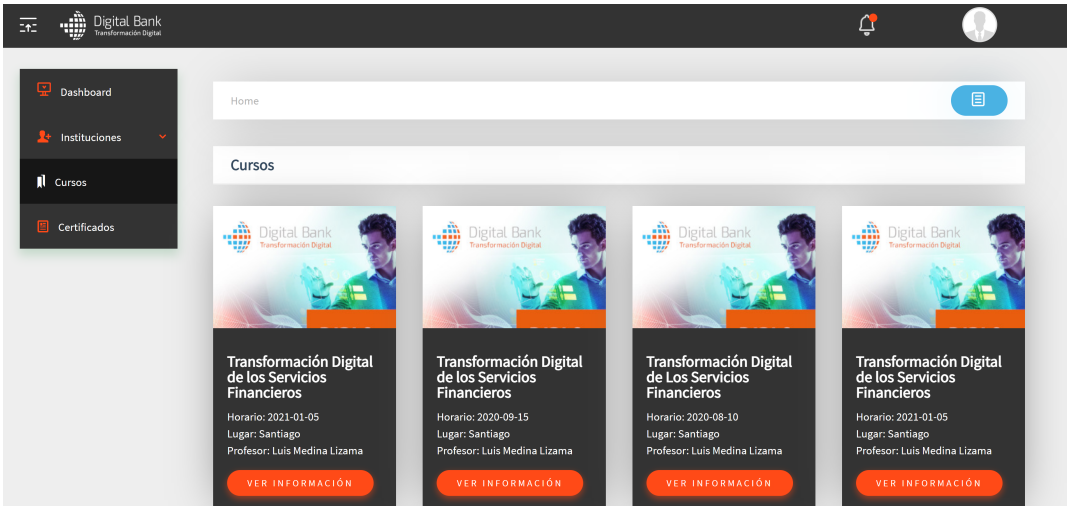


Figura 7.7: Vista de Cursos Coordinador

7.3.4. Proceso de Firma Instructor

Las figuras 7.8, 7.9 y 7.10, muestran el desarrollo de las vistas para el proceso de firma por parte del Instructor. El instructor debe subir la información de las notas del curso seleccionado en un formato establecido en Excel, esta información es tratada en el sistema y mostrada al Instructor para una revisión final y posterior firma utilizando Metamask como conexión a Blockchain para generar su firma única y personal.

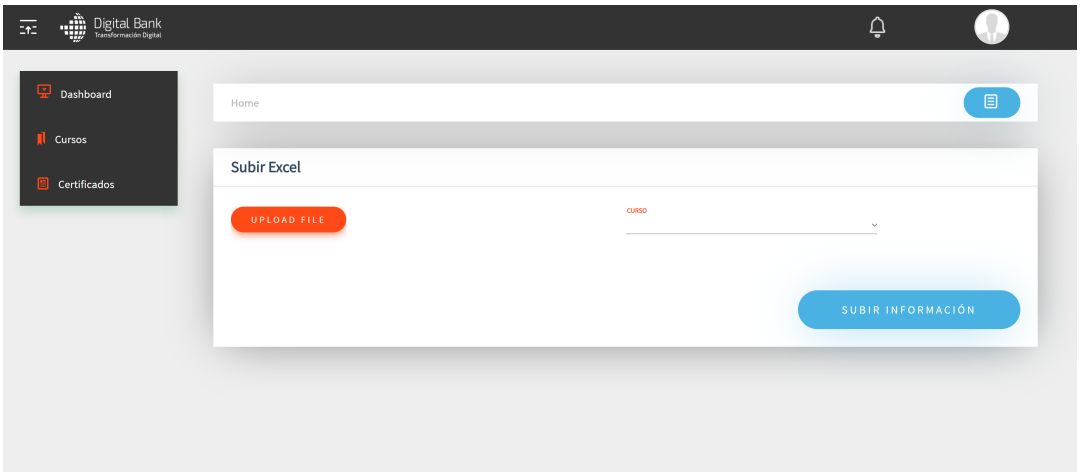


Figura 7.8: Vista Subir información de alumnos al Curso

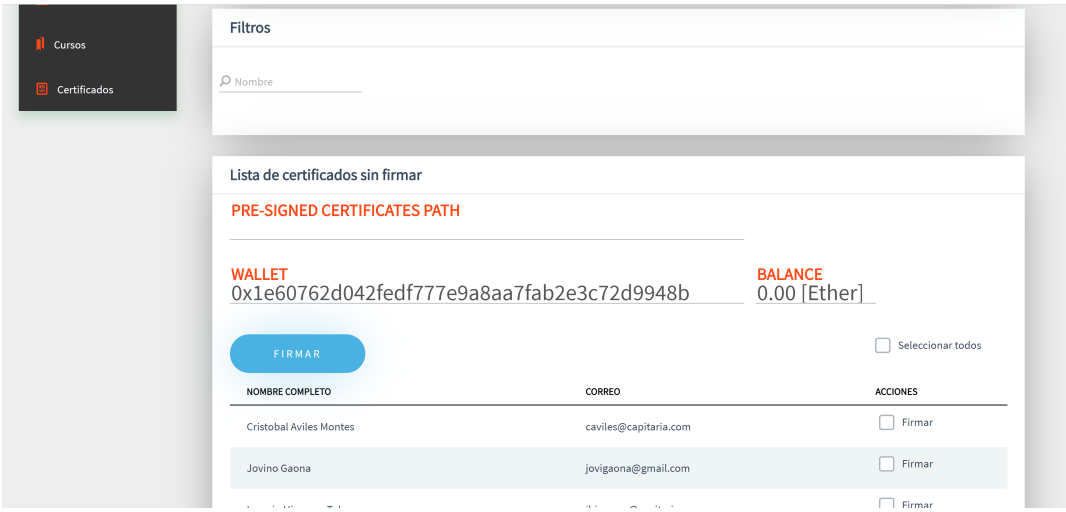


Figura 7.9: Vista Información Subida de alumnos al Curso

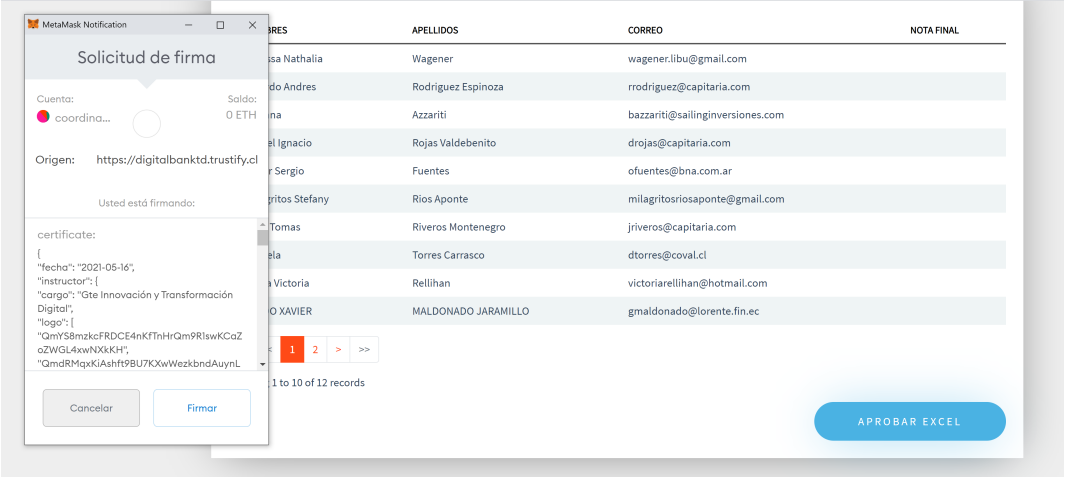


Figura 7.10: Vista de firma de información

7.3.5. Proceso firma Coordinador

Las figuras 7.11 y 7.12, muestran el desarrollo de las vistas para el proceso de firma por parte del Coordinador. La vista para seleccionar el curso a revisar es equivalente a la vista de la figura 7.7, por lo que se omitió en esta sección. La firma por parte del coordinador también es apoyada por Metamask como herramienta de conexión a blockchain.

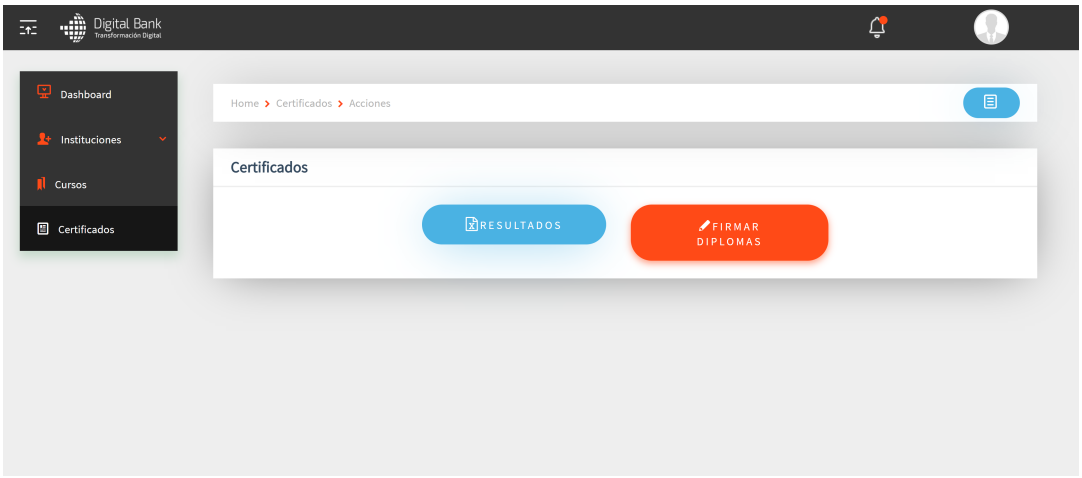


Figura 7.11: Vista de opciones Coordinador

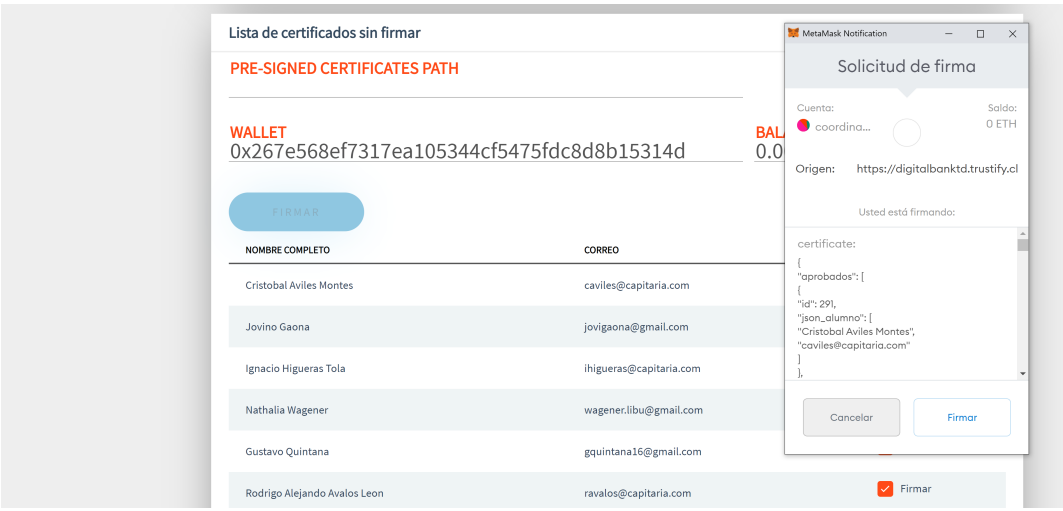


Figura 7.12: Vista de siguientes eventos

7.4. Registro de Instructor

La Figura 7.13, 7.14, 7.15 y 7.16 muestra la secuencia a seguir para crear y validar un Instructor. Este mismo proceso de aplica para la creación y validación de un Coordinador. Cada nuevo usuario debe crear su contraseña una vez creado su perfil e informar la wallet que utilizará para realizar las firmas como prueba de identidad.

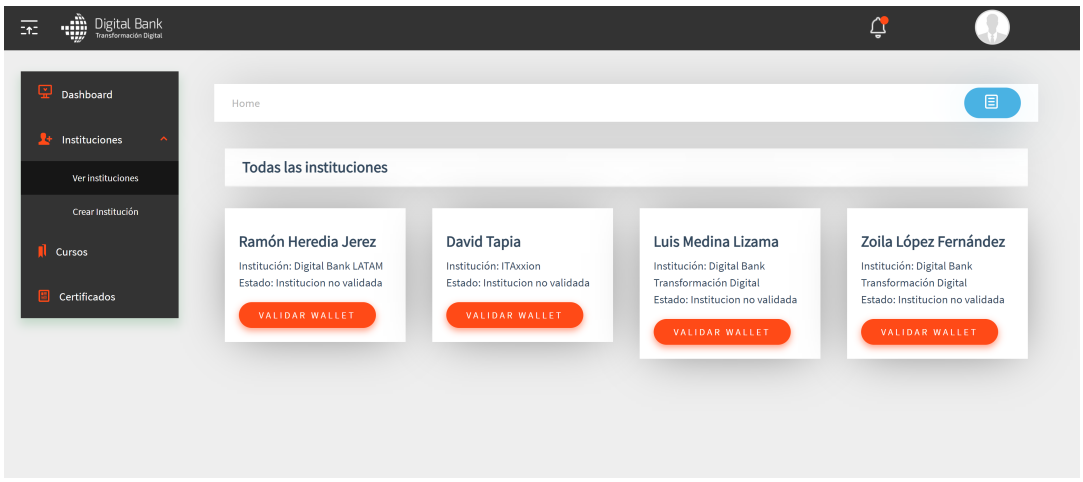


Figura 7.13: Vista de Instructores

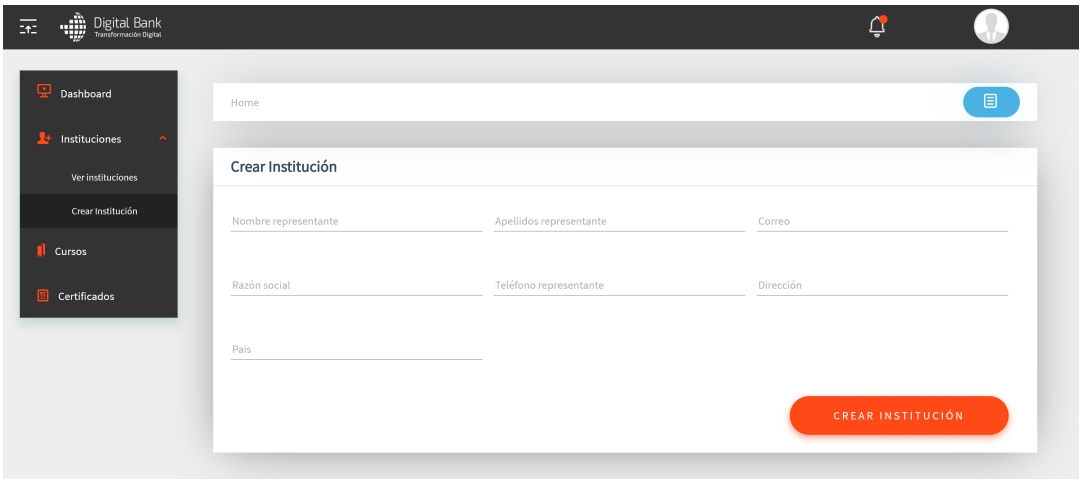


Figura 7.14: Vista crear un Instructor

Figura 7.15: Vista crear contraseña

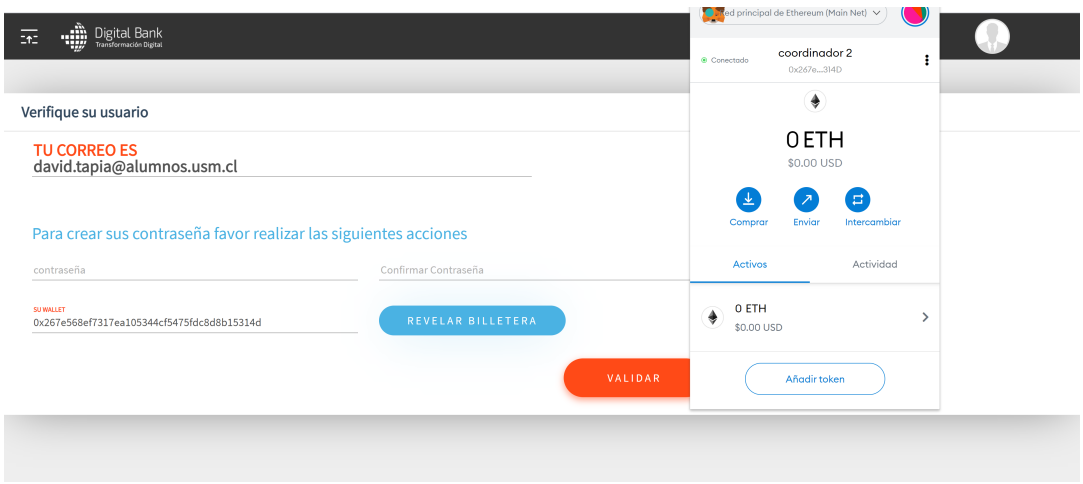


Figura 7.16: Vista crear contraseña con wallet

7.5. Ingreso a la plataforma

La Figura 7.17 es la vista de control de ingresos a la plataforma.

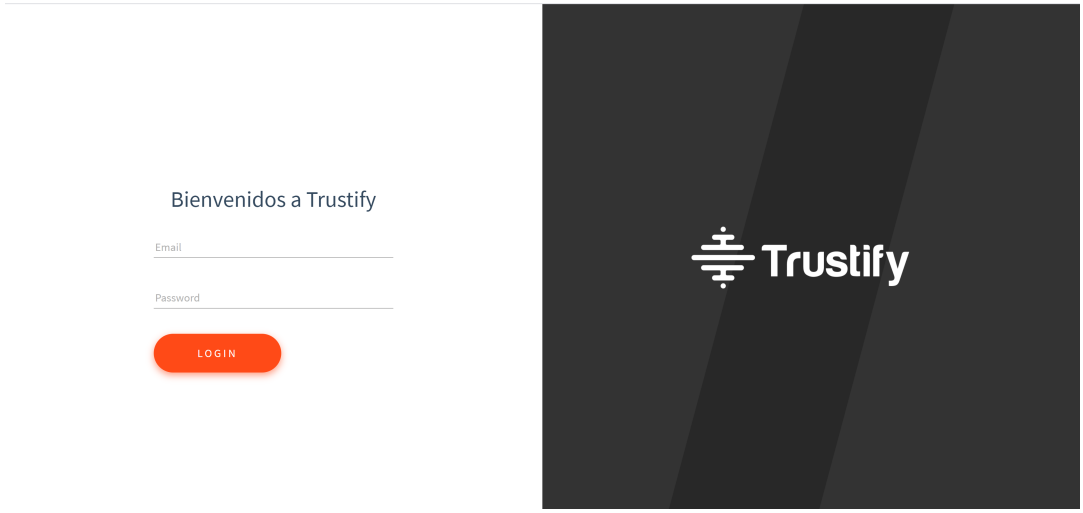


Figura 7.17: Vista de interfaz de ingreso

7.6. Documentos

Las Figuras 7.18 y 7.19 muestran el resultado de la emisión de documentos tanto como para los usuarios enrolados en la plataforma como para los usuarios externos.

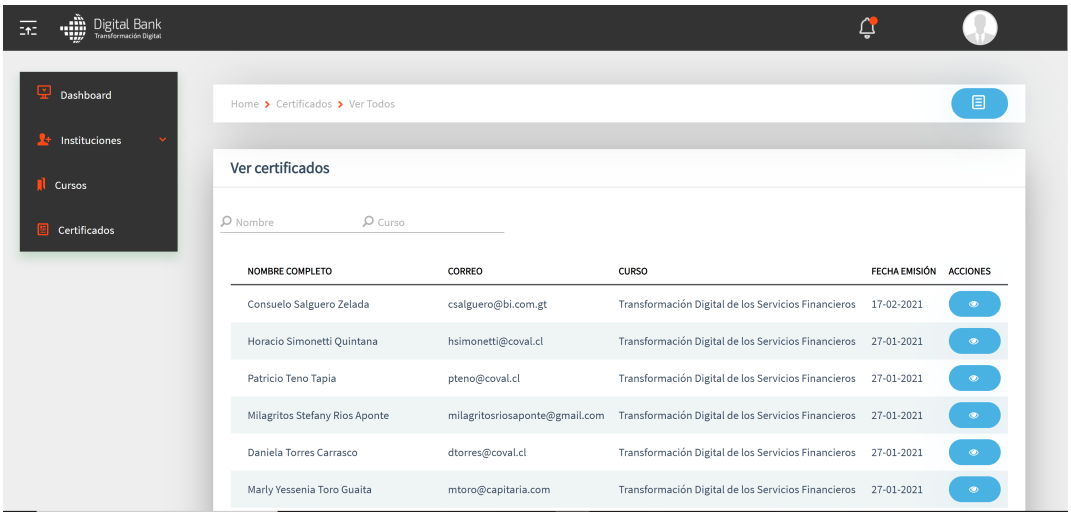


Figura 7.18: Vista de interfaz de ingreso

Figura 7.19: Vista de interfaz de ingreso

7.7. Diagramas de secuencias

En esta sección se presentan los diagramas de secuencias y comunicaciones lógicas entre los procesos y componentes involucrados en los 3 principales flujos que realiza la plataforma. Se deja presente que el proceso de creación de un coordinador es parte del despliegue de la plataforma, por lo cual para estos efectos se omite dicho proceso.

7.7.1. Creación de Instructor

El instructor, corresponde al perfil que cuenta con los privilegios de cargar información que luego es validada por un coordinador. En la figura 7.20. se muestra el paso a paso que debe seguir un usuario coordinador de una institución para crear y validar instructores:

- El coordinador ingresa a la plataforma.
- El coordinador crea un usuario instructor.
- El coordinador valida al usuario instructor como firmante en el blockchain.

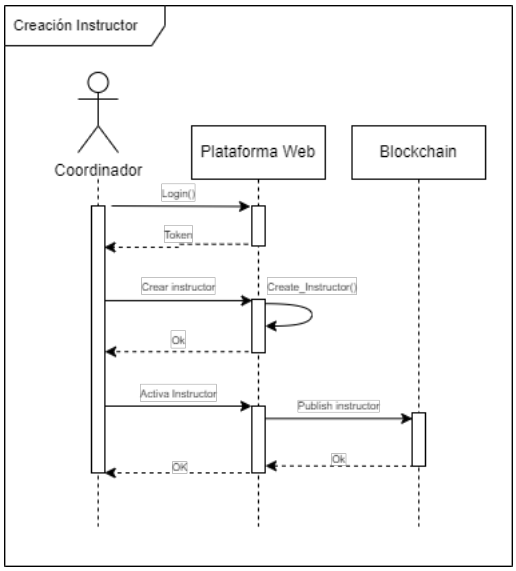


Figura 7.20: diagrama de secuencia Creación de un instructor

7.7.2. Flujo de firma de documentos

Una vez que los usuarios están validados se procede a realizar el flujo de creación de curso y firma para su posterior emisión de documentos.

- El instructor crea el curso.
- El instructor sube el archivo con las notas del curso.
- El instructor revisa y finaliza firmando la información del archivo.
- El coordinador obtiene el curso pendiente de firma.
- El coordinador revisa y firma la información del curso.

- El sistema publica en el blockchain cada documento creado.

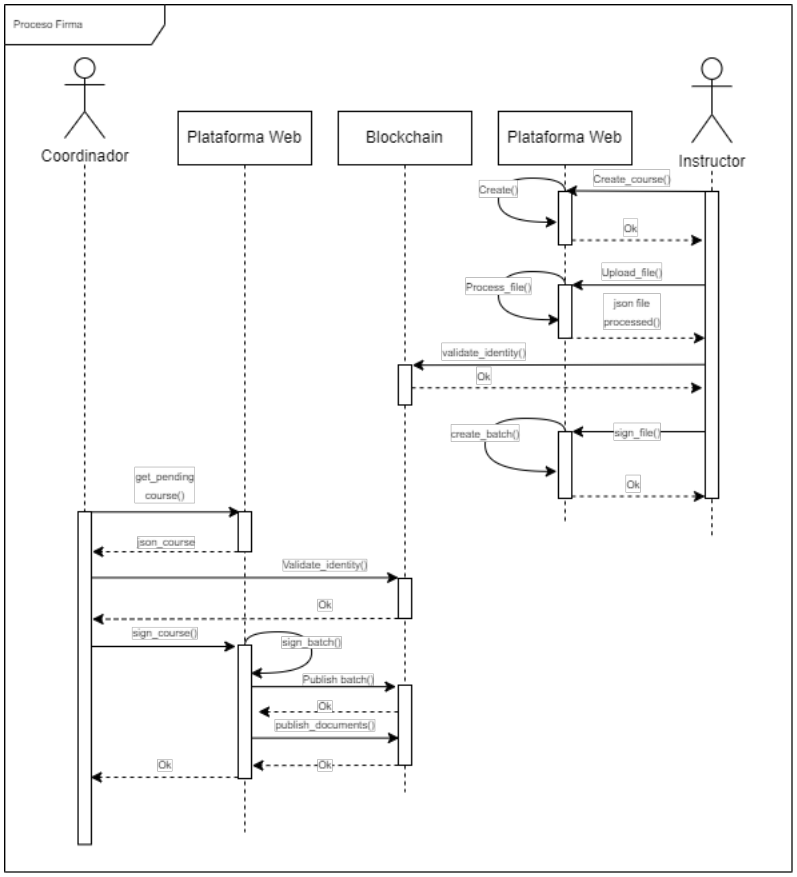


Figura 7.21: diagrama de secuencia Proceso de firma

7.7.3. Verificación de documento

Cuando los documentos están emitidos se pueden realizar el proceso de verificación única de los documentos con el código único que poseen. Esta acción la puede realizar el alumno beneficiario o cualquier tercero que posea el código.

- El usuario ingresa el código a consultar.
- El sistema realiza búsqueda en el blockchain.
- El sistema reconstruye la información y el documento.
- Si el usuario lo requiere puede descargar el documento en formato PDF.

La interacción mencionada se puede ver en el siguiente diagrama:

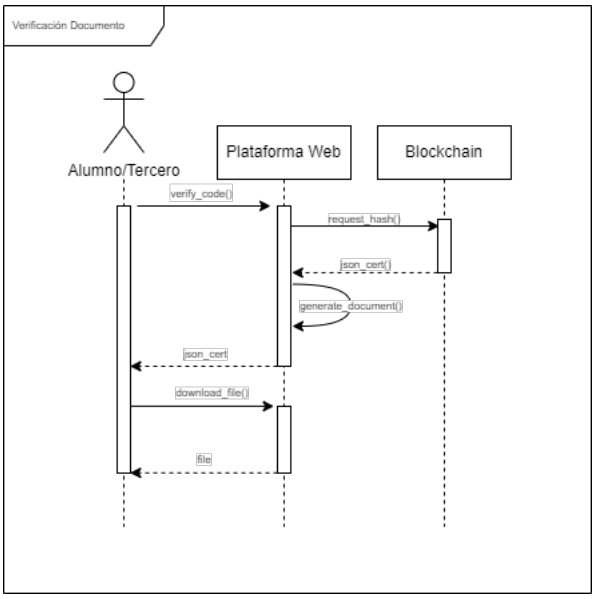


Figura 7.22: diagrama de secuencia Verificación de documento

Capítulo 8

Conclusiones

En el presente capítulo se abordan nuevamente los objetivos definidos, con la finalidad de garantizar su cumplimiento, y a su vez verificar si efectivamente las hipótesis han podido ser demostradas en el plano fáctico.

8.1. Conclusiones generales

- La necesidad que dio origen a esta memoria se logró manejar en buena forma. Al cierre de este trabajo, la consultora *Digital Bank Transformación Digital* sigue haciendo uso activo del sistema, mientras que en paralelo, se realizan gestiones comerciales y exploración de nuevas arquitecturas y funcionalidades.

- La plataforma de administración posee una interfaz que permite; cargar, gestionar, firmar, emitir y validar las certificaciones. Generando un registro persistente e independiente de la plataforma "Trustify" para cada certificado satisfactoriamente emitido.

- Durante el transcurso del Programa de Memorias Multidisciplinarias 2019, la plataforma quedó disponible en una fase de validación y pruebas en *https://demo.trustify.cl/*, la cual estuvo en constante retroalimentación con la contraparte IT Axxion y luego, con Digital Bank Transformación Digital. Esto permitió agilizar el desarrollo del sistema frente a ajustes en los requerimientos y necesidades vistas anteriormente.

8.2. Conclusiones específicas

- El modelo de datos presentado en el diseño de la solución permite asociar a un estudiante con sus certificaciones personales, notas obtenidas en el curso, firmantes, certificación obtenida, inicio y fin de curso, entre otros. Esto permite crear una trazabilidad completa de la certificación, la que al ser digital, es fácilmente compartible y, a su vez, estar referenciada en un sistema distribuido como lo es el blockchain y a la vez alojado en IPFS contar con el registro cuando se necesite de manera persistente.
- La integración de la tecnología utilizada para la emisión y almacenamiento persistente se hizo posible gracias al esfuerzo conjunto del equipo trabajo. Partiendo por el acucioso estudio de la tecnología blockchain, el aprendizaje del lenguaje de programación Solidity y la integración de front con Metamask. El uso de códigos QR asociados de forma dinámica a cada diploma permite verificar y recuperar una copia digital del certificado obtenido.
- Finalmente pero no menos importante, una plataforma web que esté disponible desde internet y que permite el acceso a coordinadores, instructores y terceros acceder desde un computador o smartphone para comprobar de manera sencilla una credencial obtenida. Esto actúa como habilitador de sinergias y optimiza procesos de contratación basados en competencias.

8.3. Trabajo futuro

Se espera en un futuro, que la solución implementada sea adoptada por otras instituciones educativas del país, logrando generar nuevos procesos de certificación e incluso robustecer la plataforma, considerando la integración y uso de firma electrónica avanzada, configurando “Trustify” como una plataforma de certificación exitosa, tanto a nivel comercial como técnico.

Figura 8.1: Logo del servicio Trustify

Dentro de los principales hitos a desarrollar en el futuro, podemos mencionar:

- Fortalecer la escalabilidad a nivel técnico, pues al estar en la red ethereum la cantidad máxima de transacciones por segundo es cercana a 17. Lo que permitiría evitar posibles congestiones de red.

- Implementar una nueva interfaz de diseño de procesos de negocio, generando una plataforma tipo WorkFlow en la que se certifiquen procesos de negocios que requieran contar con una capacidad de alta persistencia y no repudio.
- Integrar firma electrónica avanzada, para permitir que toda certificación emitida por la plataforma cuente con respaldo legal equivalente a la firma hológrafa.
- Integrar el uso de redes blockchain tipo consorcio basadas en el protocolo ethereum. Lo que permitiría contar con una permisiología más dura, resguardando del acceso público información sensible que haya sido sometida a una certificación hecha en Trustify.

Anexos

En este apartado, se revisa los diseños preliminares de las principales vistas implementadas, el flujo de información y acciones y la disposición de la información para el usuario que debe utilizarlas. Estas vistas comprenden el ingreso a la plataforma, creación de usuario, panel principal de la plataforma, creación y edición de cursos, proceso de informa para instructor y coordinador, y verificación de documento en el sistema.

A.1. Mockups de las interfaces

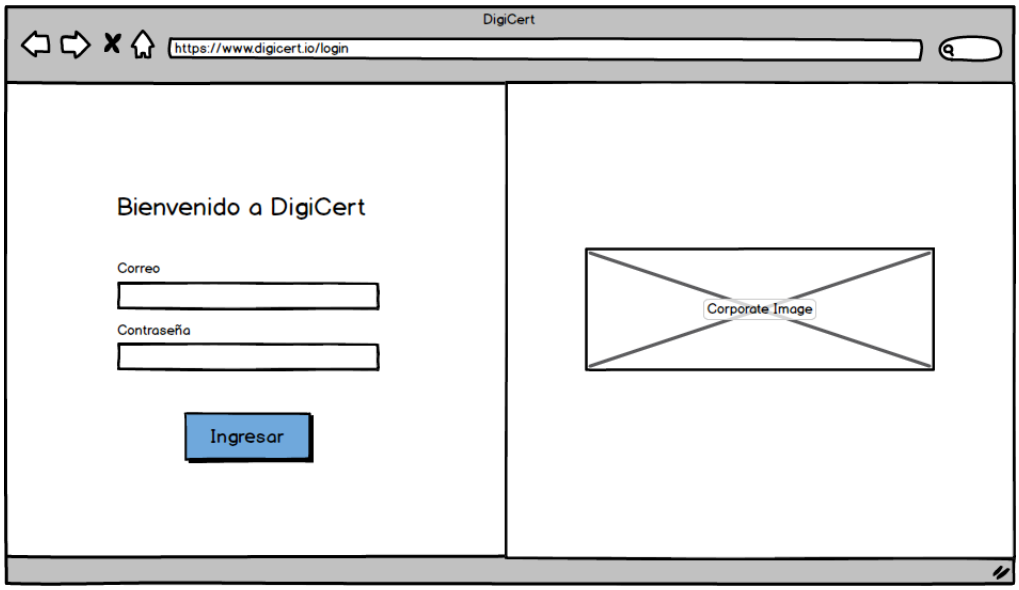


Figura A.2: Ingreso

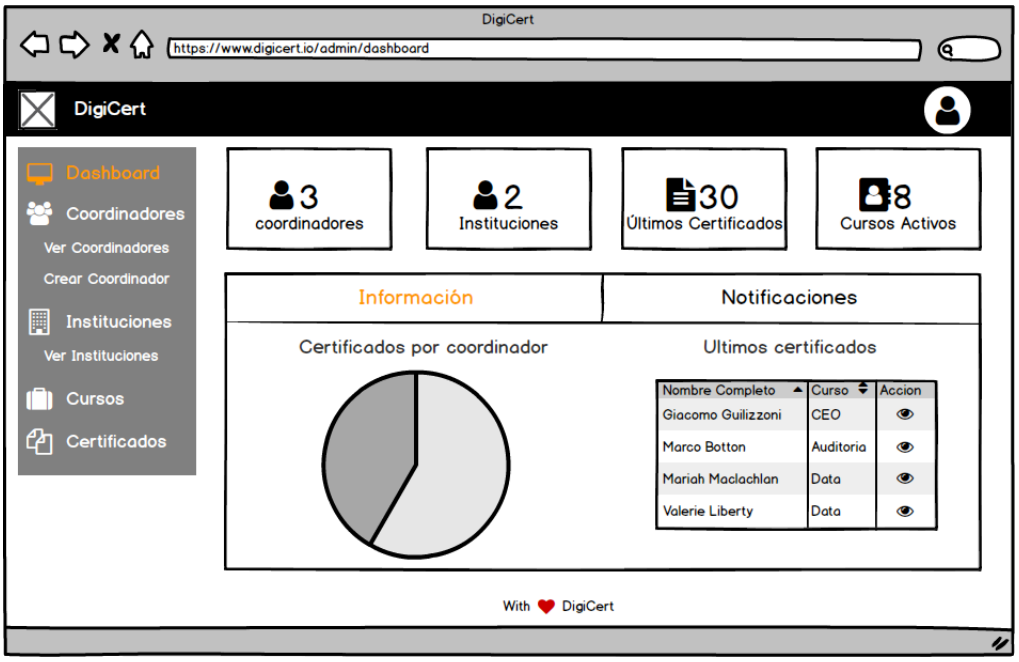


Figura A.3: Panel de Administración

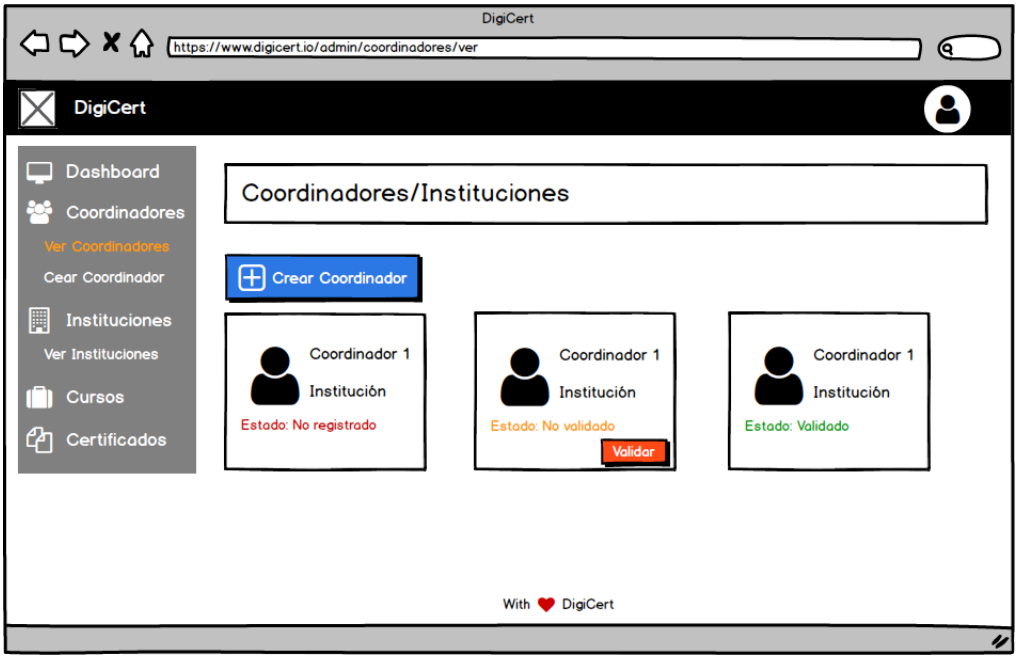


Figura A.4: Ver Coordinadores/Instructores

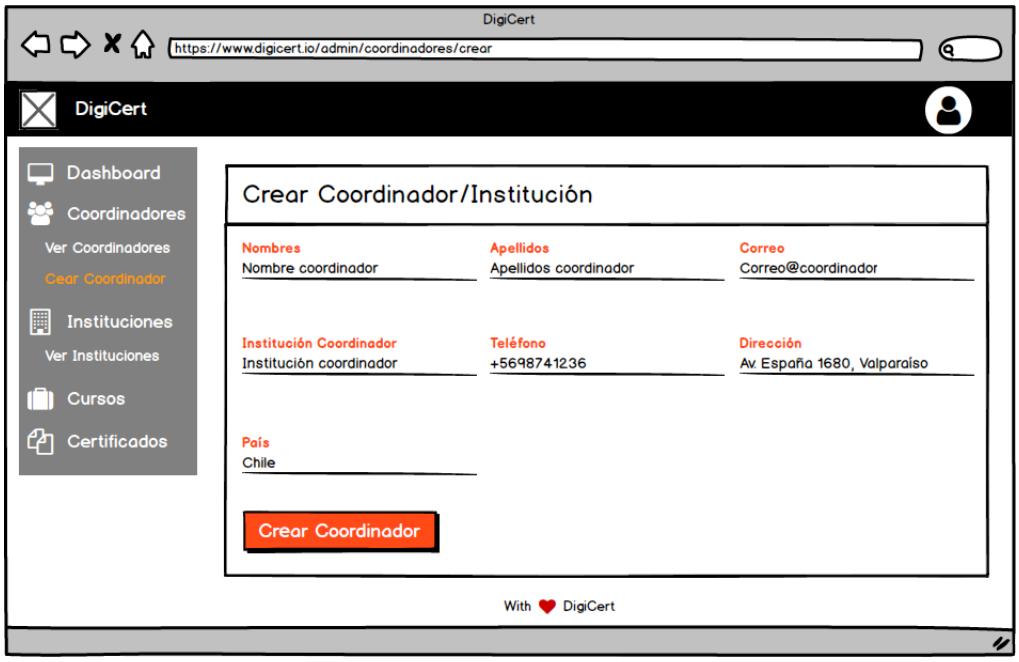


Figura A.5: Crear Coordinador/Instructor

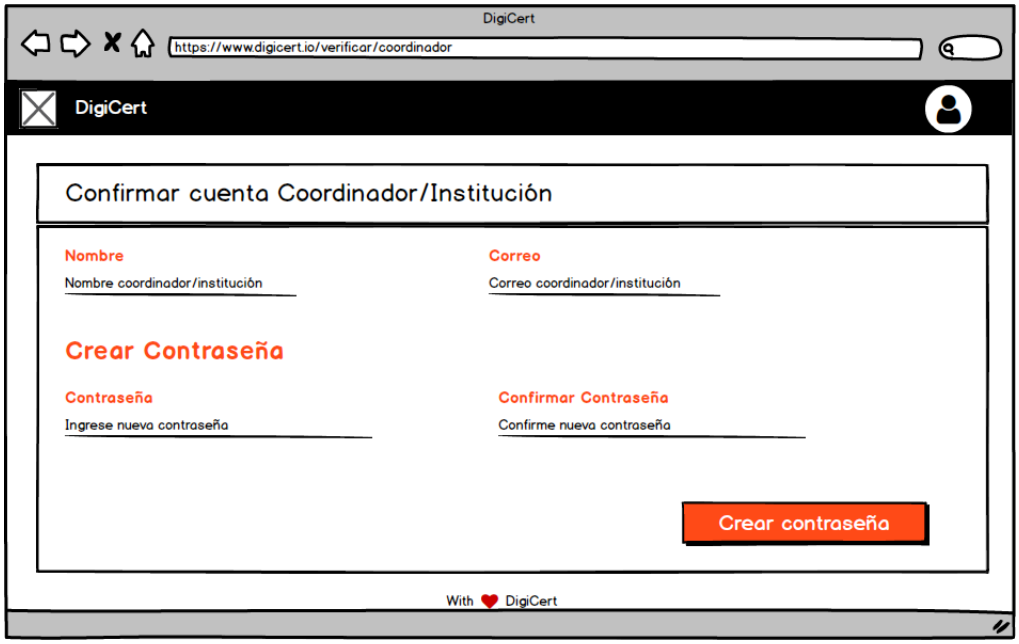


Figura A.6: Crear clave de acceso

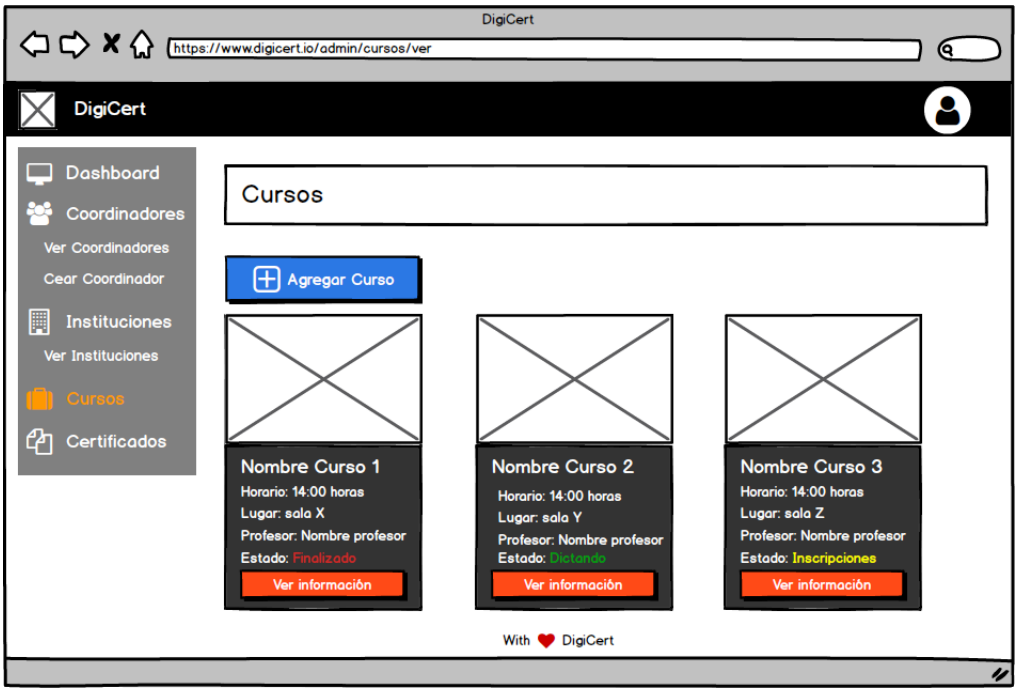


Figura A.7: Ver cursos

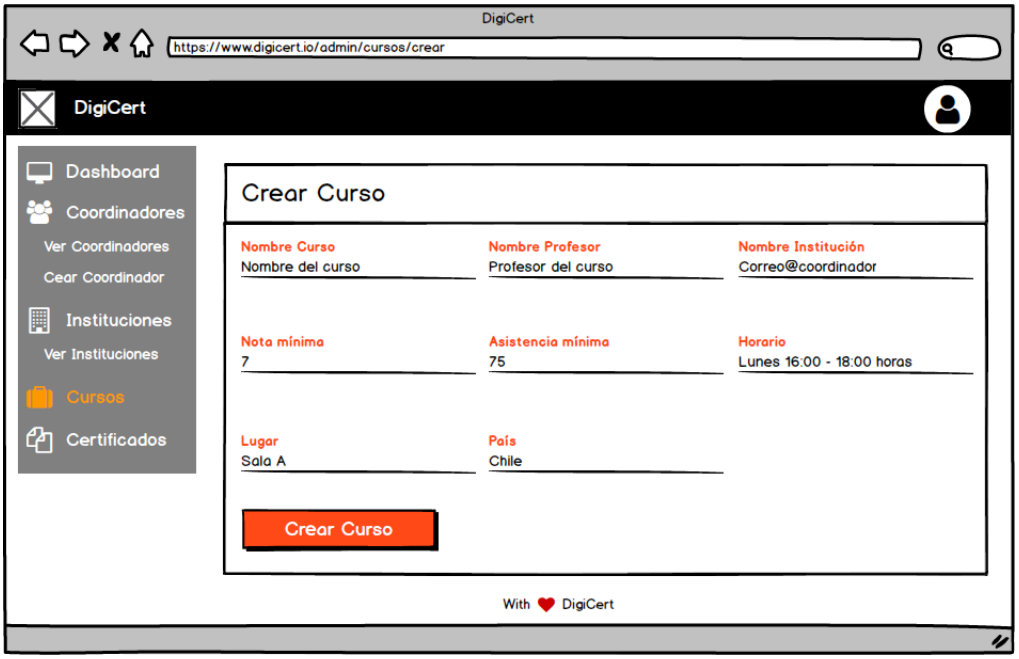


Figura A.8: Crear un curso

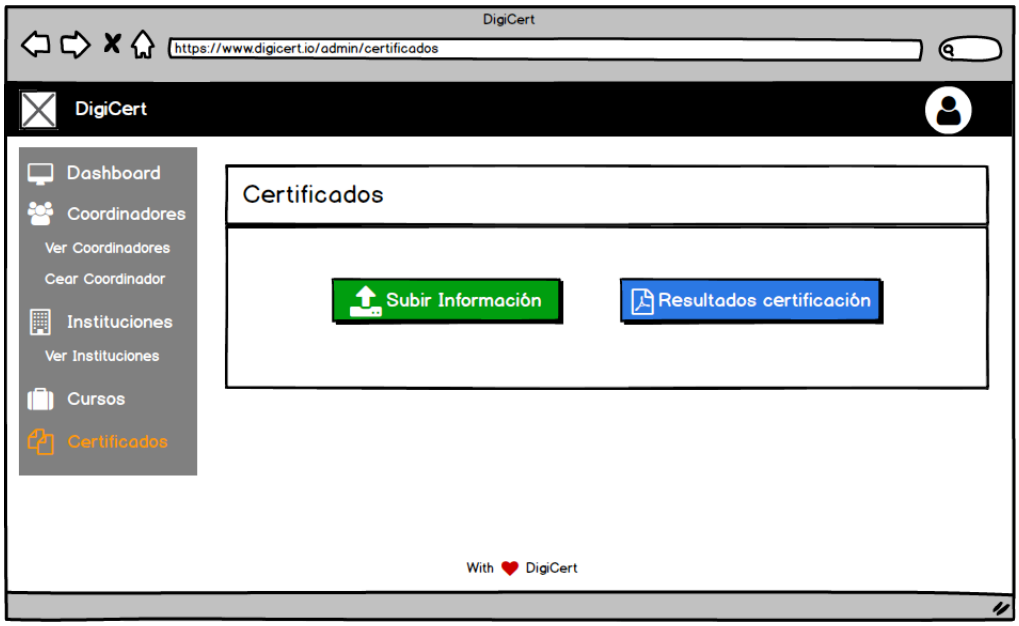


Figura A.9: Opciones de documentos Instructor

Figura A.10: Subir información del curso

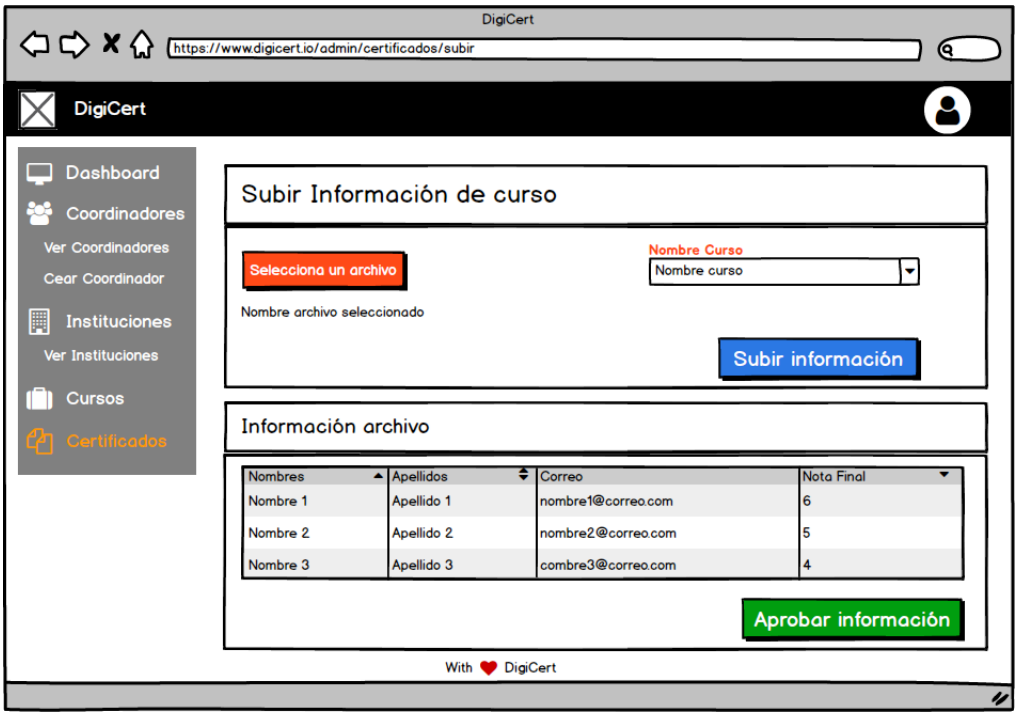


Figura A.11: Firmar información del curso por Instructor

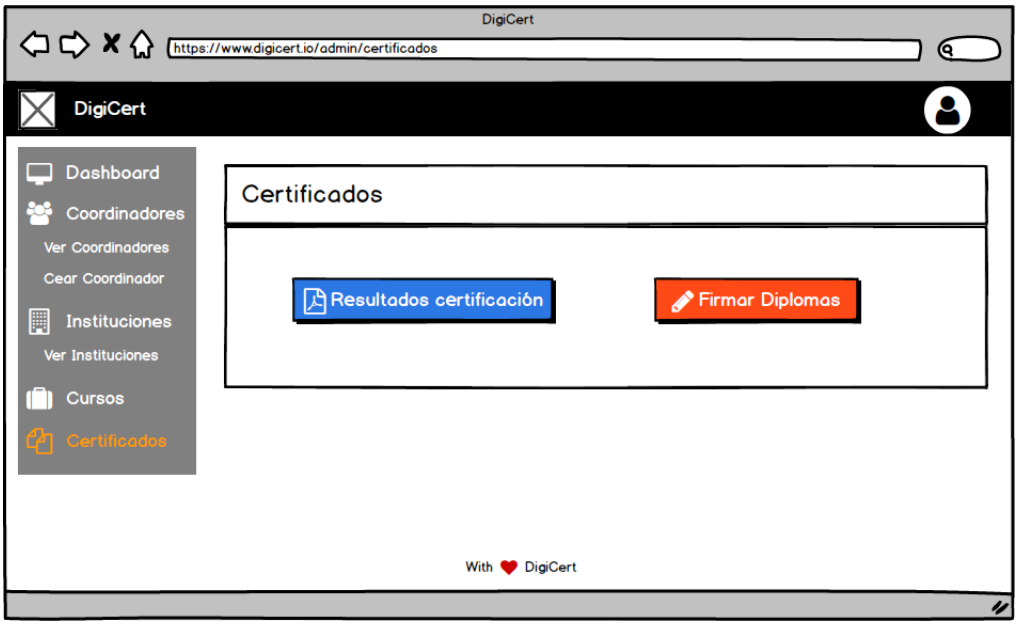


Figura A.12: Opciones de documentos Coordinador

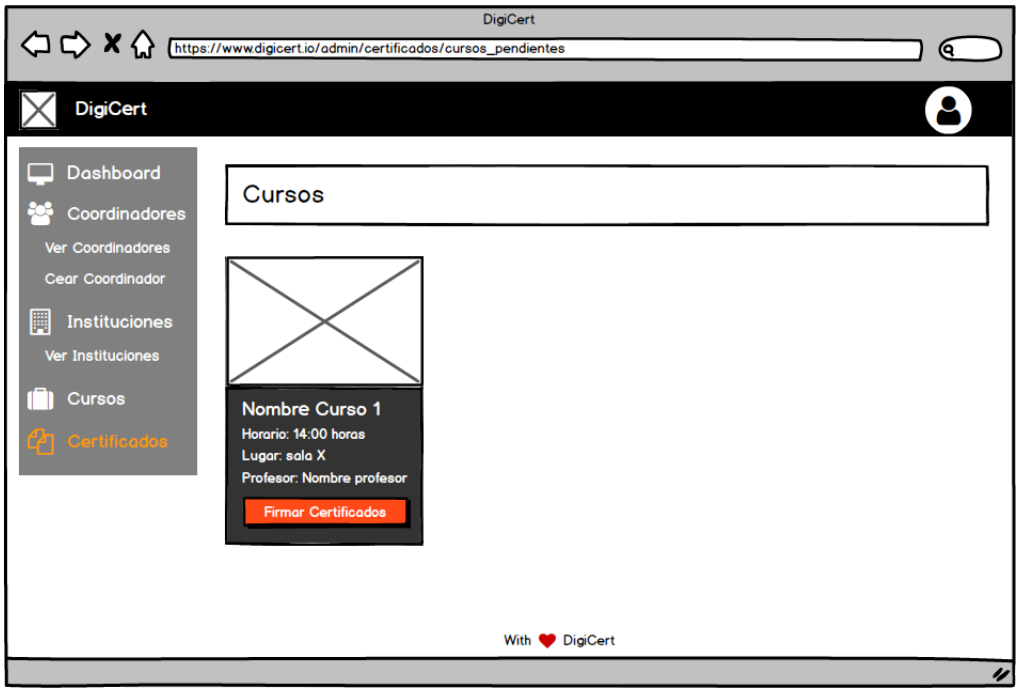


Figura A.13: Cursos pendientes de firmar por Coordinador

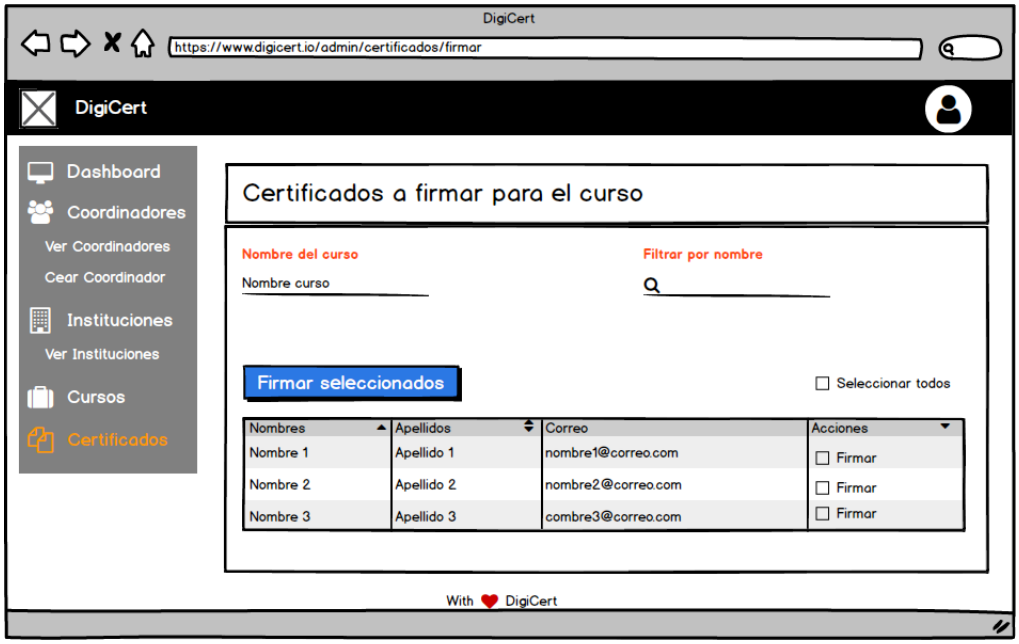


Figura A.14: Firmar cursos por Coordinador

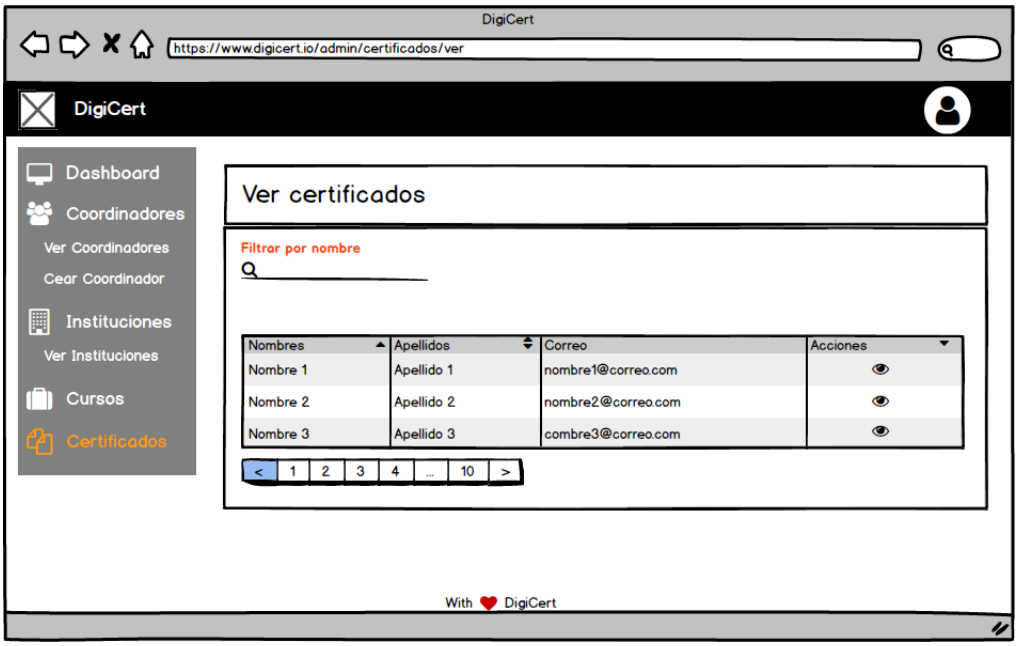


Figura A.15: Lista de documentos emitidos

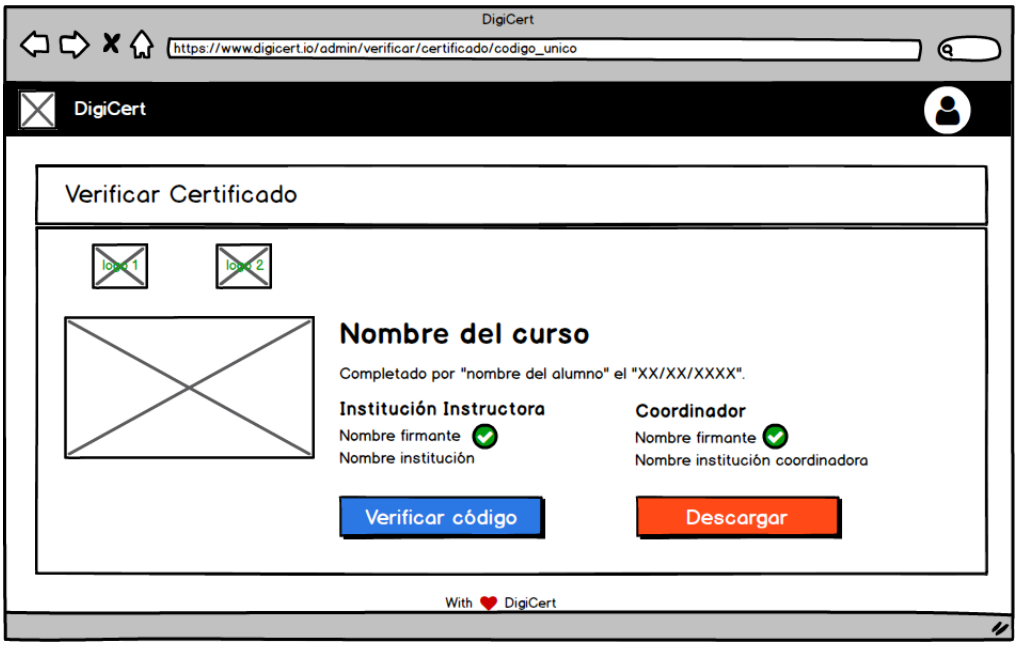


Figura A.16: Verificar documento

Referencias

[1] Programa de Memorias Multidisciplinarias Universidad Técnica Federico Santa María
Recuperado de https://www.desafiostecnologicos.usm.cl/empresas-han-participado/2019
[Consultado: 15/07/2021]

[2] What is Blockchain Technology?
Recuperado de https://consensys.net/knowledge-base/about-blockchain-technology/
[Consultado: 10/07/2021]

[3] What is IPFS?
Recuperado de https://docs.ipfs.io/concepts/what-is-ipfs/
[Consultado: 20/07/2021]

[4] What are smart contracts on blockchain?
Recuperado de https://www.ibm.com/topics/smart-contracts
[Consultado: 20/07/2021]

[5] What is Ethereum? The Foundation for our digital future
Recuperado de https://ethereum.org/en/what-is-ethereum/
[Consultado: 20/07/2021]

[6] Ley N° 19799 Sobre documentos electrónicos, firma electrónica y servicios de certificación de dicha firma
Recuperado de https://www.bcn.cl/leychile/navegar?idNorma=196640
[Consultado: 25/07/2021]

[7] Ley N° 20720 Sustituye el régimen concursal vigente por una ley de reorganización y liquidación de empresas y personas, y perfecciona el rol de la superintendencia del ramo
Recuperado de https://www.bcn.cl/leychile/navegar?idNorma=1058072
[Consultado: 28/007/2021]

[8] Los documentos electrónicos están definidos por la LFE como «toda representación de un hecho, imagen o idea que sea creada, enviada, comunicada o recibida por medios electrónicos y almacenada de un modo idóneo para permitir su uso posterior».
Recuperado de https://www.carey.cl/firma-electronica-cuando-puede-utilizarse-y-cuando-no/
[Consultado: 28/06/2021]

[9] Digital Diploma debuts at MIT
Recuperado de https://news.mit.edu/2017/mit-debuts-secure-digital-diploma-using-bitcoin-blockchain-technology-1017
Consultado: 25/07/2021

[10] OpenCerts
Recuperado de https://www.ssg.gov.sg/opencerts.html
[Consultado: 25/07/2021]

[11] Let's Encrypt is a free, automated, and open certificate authority brought to you by the nonprofit Internet Security Research Group (ISRG).
Recuperado de https://letsencrypt.org/
[Consultado: 25/07/2021]

[12] Google CEO preaches 'mobile first'
Recuperado de https://www.computerworld.com/article/2520954/google-ceo-preaches–mobile-first-.html
[Consultado: 25/07/2021]

[13] Ethereum Mainnet Statistics
Recuperado de https://ethernodes.org/
[Consultado: 25/07/2021]

[14] MVC
Recuperado de https://developer.mozilla.org/es/docs/Glossary/MVC
[Consultado: 20/07/2021]

[15] Richards, M. (2015). Software Architecture Patterns
Recuperado de http://www.oreilly.com/programming/free/software-architecture-patterns.csp
[Consultado: 01/03/2021]

[16] Microsoft Corp. (2012). The MVVM Pattern
Recuperado de https://msdn.microsoft.com/en-us/library/hh848246.aspx
[Consultado: 02/03/2021]

[17] Fielding, R. T. (2000). Architectural Styles and the Design of Network-based Software Architectures, 76-106 (Tesis doctoral). University of California, Irvine.
Recuperado de https://www.ics.uci.edu/˜fielding/pubs/dissertation/fielding_dissertation.pdf
[Consultado: 02/03/2021]

[18] Precio de Ether en Pesos Chilenos
Recuperado de https://www.coinbase.com/converter/eth/clp
[Consultado: 02/03/2022]

[19] What is MetaMask? Really... What is it?
Recuperado de https://medium.com/@seanschoi/what-is-metamask-really-what-is-it-7bc1bf48c75
[Consultado 22/06/2021]

[20] Decentralized Applications - dApps.
Recuperado de https://www.investopedia.com/terms/d/decentralized-applications-dapps.asp
[Consultado 22/06/2021]

[21] web3.js - Ethereum JavaScript API
Recuperado de https://web3js.readthedocs.io/en/v1.4.0/
[Consultado 22/06/2021]

[22] The JavaScript Object Notation (JSON) Data Interchange Format
Recuperado de https://datatracker.ietf.org/doc/html/rfc8259
[Consultado 22/06/2021]

[23] OMG Unified Modeling Language
Recuperado de https://www.omg.org/spec/UML/2.5.1/PDF
[Consultado 22/06/2021]

[24] Solidity
Recuperado de https://docs.soliditylang.org/en/v0.8.6/
[Consultado 22/06/2021]

[25] Google, Inc. (2021). Insights para mejorar la experiencia del usuario móvil.
Recuperado de https://developers.google.com/speed/pagespeed/insights/?hl=es
[Consultado: 28/07/2021]

[26] Google, Inc. (2019). Auditar apps web con Lighthouse.
Recuperado de https://developers.google.com/web/tools/lighthouse/
[Consultado: 28/07/2021]

[27] Aruba Cloud.
Recuperado de https://www.arubacloud.com/company/about-aruba.aspx
[Consultado: 10/12/2021]

[28] Manifesto for Agile Software Development
Recuperado de https://agilemanifesto.org/
[Consultado: 10/12/2021]