Escuela Politécnica Superior de Jaén
Máster Universitario en Seguridad Informática

Universidad de Jaén
Escuela Politécnica Superior de Jaén

TRABAJO FIN DE MÁSTER

BIOIDENTIDAD SEGURA CON BLOCKCHAIN

Alumno

Manuel Gallego Chinchilla

Tutores

Miguel Ángel García Cumbreras
(Departamento de Informática)

Julio Merelo Casado
(APD)

Octubre, 2022

(Página intencionalmente en blanco)

Universidad de Jaén

Departamento de Informática

Don Miguel Ángel García Cumbreras y Don Julio Merelo Casado, tutores del Trabajo Fin de Máster titulado: '**Bioidentidad segura con Blockchain**', que presenta Don Manuel Gallego Chinchilla, otorgan el visto bueno para su entrega y defensa en la Escuela Politécnica Superior de Jaén.

Jaén, Octubre de 2022

El alumno: Los tutores:

GALLEGO CHINCHILLA MANUEL - 26503035C

Firmado digitalmente por GALLEGO CHINCHILLA MANUEL - 26503035C
Fecha: 2022.10.20 20:35:36 +02'00'

GARCIA CUMBRERAS MIGUEL ANGEL - 52871272E

Firmado digitalmente por GARCIA CUMBRERAS MIGUEL ANGEL - 52871272E
Fecha: 2022.10.21 11:01:57 +02'00'

Julio Merelo Casado

Firmado digitalmente por Julio Merelo Casado
Fecha: 2022.10.23 22:56:00 +02'00'

Manuel Gallego Chinchilla

Miguel Ángel García Cumbreras

Julio Merelo Casado

(Página intencionalmente en blanco)

Agradecimientos

En primer lugar, me gustaría expresar mi agradecimiento a mis tutores, Miguel Ángel García Cumbreras y Julio Merelo Casado, por la ayuda que me han ofrecido a lo largo del trabajo y por darme la oportunidad de desarrollar la idea que tenía en mente.
Y por otra parte, me gustaría agradecer tanto a mis padres como a mi novia por su apoyo incondicional durante este tiempo.

FICHA DEL TRABAJO FIN DE TÍTULO	
Titulación	Máster en Seguridad Informática
Modalidad	Trabajo Teórico/Experimental
Especialidad (solo TFG)	Sin especialidad
Mención (solo TFG)	Sin mención
Idioma	Español
Tipo	Específico
TFT en equipo	No
Autor/a	Manuel Gallego Chinchilla
Fecha de asignación	25/04/2022
Descripción corta	Este proyecto nace de la idea de aunar la tecnología Blockchain con la identidad biométrica, dos de los sectores tecnológicos que más están creciendo en estos últimos años. Por una parte, la tecnología Blockchain está creciendo gracias a que muchos proyectos, empresas, y gobiernos la están adoptando para crear casos de uso descentralizados, trazables y fiables, mientras que la bioidentidad se está implementando en casi todas las acciones que requieren de un control de acceso, como móviles, acceso a edificios, dobles factores de autenticación, autorizaciones bancarias, etc. Al unir estas dos tecnologías se pueden crear casos de uso como el que se presenta en este TFM, el cual es una pasarela de autenticación que permite al usuario poder acceder a todas las aplicaciones que requieran identificación teniendo una sola identidad. Al realizar el registro se escanea el rasgo biométrico (cara o huella dactilar), se procesa y se genera un hash biométrico a partir de él. Este hash se almacena en la red Blockchain permitiendo así poder recuperarlo cuando el usuario inicie sesión. Una de las ventajas que ofrece esta implementación en Blockchain es que todas los servicios y aplicaciones que se encuentren dentro de la red usarán el mismo método de identificación por lo que registrándote una vez ya puedes acceder a todas ellas.

	A diferencia del método actual en el que para cada servicio o aplicación hay que darse de alta y posteriormente tener que recordar multitud de nombres de usuario y contraseñas. Para obtener un MVP (producto viable mínimo) se ha desarrollado una aplicación multiplataforma que permite probar el funcionamiento de esta pasarela y mediante el explorador de bloques Epirus de Telefónica poder visualizar las transacciones que se almacenan en la red Blockchain. Y con ello poder mostrar el funcionamiento básico que se va explicando a lo largo del trabajo.

NORMAS APLICADAS EN ESTE DOCUMENTO	
LOCALES	
TFT-UJA:2017	Normativa de Trabajos Fin de Grado, Fin de Máster y otros Trabajos Fin de Título de la Universidad de Jaén (Normativa marco UJA aprobada en Consejo de Gobierno)
TFT-EPSJ:2017	Normativa sobre Trabajos Fin de Grado y Fin de Máster en la Escuela Politécnica Superior de Jaén (Normativa EPSJ aprobada en Junta de Escuela)
TFT-EPSJ	Criterios de evaluación y normas de estilo para TFG y TFM de la Escuela Politécnica Superior de Jaén
NACIONALES E INTERNACIONALES	
ISO 2145:1978	Documentación - Numeración de divisiones y subdivisiones en documentos escritos
UNE 50132:1994	Traducción de la ISO 2145
IEEE	Estilo de referencias y citas de IEEE (Institute of Electrical and Electronics Engineers)

NORMAS UTILIZADAS COMO BASE O REFERENCIA	
NACIONALES	
UNE 157001:2014	Criterios generales para la elaboración formal de los documentos que constituyen un proyecto técnico
UNE 157801:2007	Criterios generales para la elaboración de proyectos de sistemas de información
*Estas normas se han utilizado **como base o referencia** para la inclusión de algunos contenidos y definiciones sobre elaboración de proyectos, entendiendo como **proyecto** la documentación consensuada entre una empresa y un cliente, que da lugar al perfeccionamiento de un contrato para la elaboración de una obra o la prestación de un servicio. Por consiguiente, no debe esperarse la aplicación de estas normas en cuanto a la completitud de los contenidos ni a la organización de los mismos.*	

Contenido

Índice de ilustraciones

Índice de tablas

1 ESPECIFICACIÓN DEL TRABAJO

En este capítulo se presenta la especificación del trabajo, con una estructura y contenidos **inspirados** en los criterios y recomendaciones que establece la norma UNE 157801:2007 - "*Criterios Generales para la elaboración de proyectos de Sistemas de Información*".

A lo largo del documento se utilizarán términos y acrónimos cuya descripción aparecen en el apartado 8 (Definiciones y abreviaturas).

1.1 Introducción

A lo largo de esta última década los sistemas informáticos han ido experimentando multitud de cambios en la identificación de las personas, especialmente con la mejora continua de los smartphones. Hasta llegar a día de hoy con la que a priori es la forma más segura y sencilla de identificación, mediante el uso de rasgos biométricos, estimándose un enorme crecimiento en los próximos años (ilustración 1).

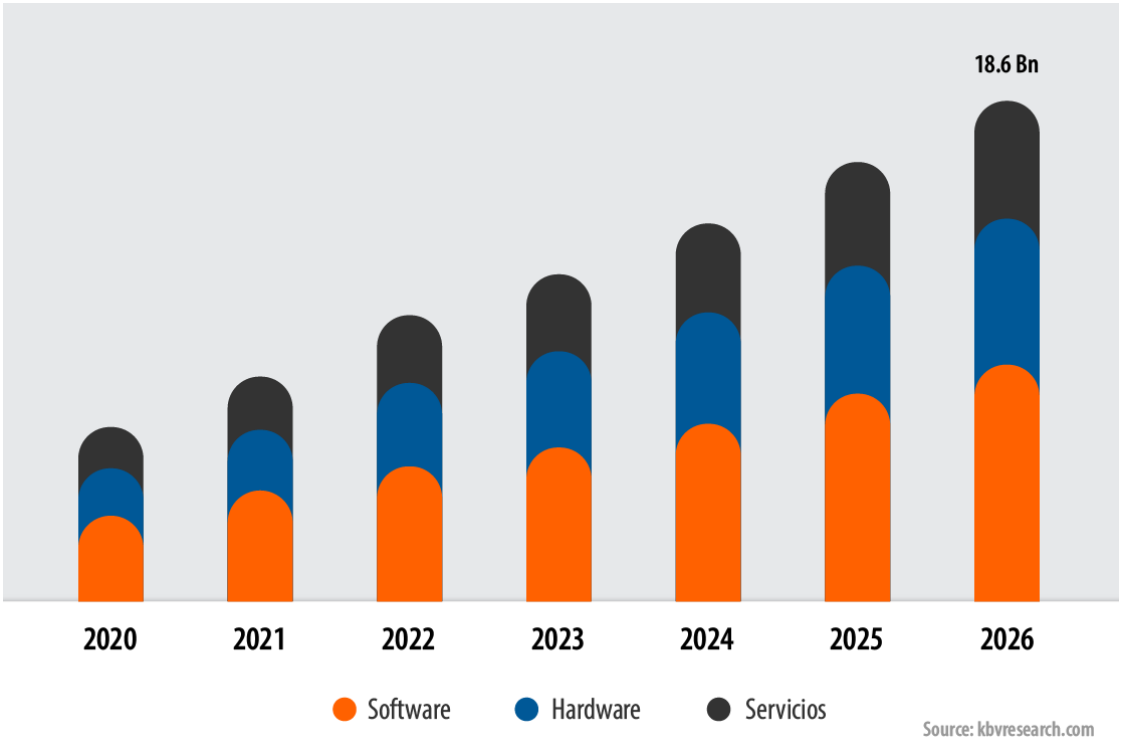


Ilustración 1. Tamaño del mercado global de tecnología biométrica [1]

Los rasgos biométricos proporcionan a los sistemas informáticos datos que permiten identificar unívocamente a un usuario, gracias al uso de aquellos rasgos que tienen características diferentes entre cada una de las personas (huella dactilar, iris, forma de la cara, geometría de la mano, voz, etc). En la actualidad podemos comprobar como la mayoría de identificaciones se realizan mediante este método, como por ejemplo, para desbloquear el móvil, acceder a aplicaciones bancarias, en el control presencial de las empresas, en aeropuertos para detectar a sospechosos, y en multitud de sitios y servicios. Esto ha sido posible en gran parte gracias a la obtención de sensores a un precio contenido y con una buena precisión en la lectura de los datos, lo que ofrece un nivel de seguridad muy alto a un precio reducido.

Por otra parte, toca hablar del otro pilar fundamental en este proyecto que es el uso de la tecnología **Blockchain**. Mucha gente asocia esta tecnología solo al mundo de las criptomonedas, pero cuenta con multitud de posibilidades, casos de uso y servicios que se pueden crear basándose en sus principales características: Distribuida, pública, funcionamiento mediante consenso, segura, inmutable, trazable y asegura la total privacidad. Al igual que con el uso de los sistemas biométricos esta tecnología no para de crecer, de ser cada vez más importante y de posibilitar ideas innovadoras, interesantes y útiles en el mundo descentralizado en el que nos estamos convirtiendo. Por ello, se estima que siga creciendo a pasos agigantados los próximos años, como vemos en la siguiente ilustración.

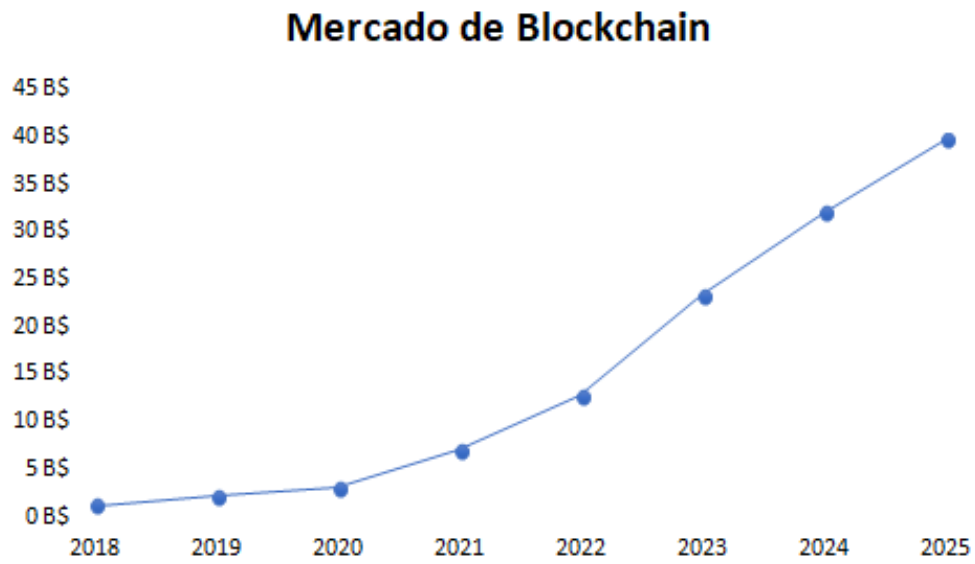


Ilustración 2. Mercado de la tecnología Blockchain [2]

Este proyecto nace tras ver la importancia presente y sobre todo futura de estos dos conceptos, integrándolos en una solución común para así conseguir un resultado final que cuente con las ventajas de la identificación biométrica y de la cadena de bloques. Por ello, se plantean una serie de retos en este trabajo:

- Desarrollo de un caso de uso centrado en ciberseguridad y basado en la tecnología Blockchain.
- Mejorar la confianza de los usuarios en el proceso de identificación.
- Ofrecer a las empresas una solución de identificación que cuente con un nivel de seguridad mayor que los métodos actuales.
- Realizar un importante avance en la integración de la bioidentificación en la cadena de bloques.

Todos estos retos proporcionan oportunidades para la mejora de la identificación en sistemas, servicios, e internet de las cosas (IoT), que puede dar lugar a una oportunidad comercial mediante el desarrollo de una solución tecnológica acorde a las necesidades que existen hoy día y enfocado plenamente en el proceso de identificación futuro.

1.2 Objetivos del trabajo

Los objetivos de este trabajo se van a enumerar a continuación, teniendo un claro objetivo principal y una serie de objetivos secundarios, que darían forma a un proyecto innovador que cuenta con las principales características de dos de los sectores tecnológicos más punteros actualmente.

El **objetivo principal** es aquel en el que se basa el proyecto, siendo la piedra angular sobre la que giran los objetivos secundarios, mejoras y nuevas ideas. Este objetivo principal se puede definir como **la creación de una herramienta de bioidentificación segura implementada y apoyada en la tecnología Blockchain**.

Para poder conseguir este objetivo se deben llevar a cabo una serie de acciones que nos facilitarán la obtención del producto final. Estas acciones pueden definirse como **objetivos específicos** y son las etapas a seguir en el proyecto:

- **Estudio acerca de la mejora que se produce con el uso de la bioidentificación con respecto a una identificación tradicional**, permitiendo ver los diferentes motivos por los que se usa este tipo de identificación.
- **Comparación entre los diferentes rasgos biométricos y elección final de los usados en el proyecto**, para así poder elegir aquel que se adapte mejor a nuestra idea, evaluando también la relación seguridad/coste.
- **Informe sobre las ventajas que ofrece el Blockchain en estos sistemas.**
- **Estudio y definición de la arquitectura general del sistema biométrico integrado en la cadena de bloques**, donde se definirá todo el proceso de identificación desde que el usuario intenta acceder hasta que la información se almacena en la cadena de bloques.
- **Estudio y elección sobre la plataforma ideal en la que desarrollar la herramienta**.
- **Conseguir interoperabilidad con multitud de sistemas y servicios**, lo que permitirá expandir las posibilidades de la herramienta creada.
- **Diseño de la interfaz y storyboard**, siguiendo en todo momento la idea de ofrecer al usuario una interfaz amigable, fácil de entender y con las tendencias actuales de diseño.
- **Desarrollo del sistema biométrico**, que será la etapa en la que se implementará todo lo estudiado y elegido en las etapas anteriores y proporcionará una primera versión ejecutable del proyecto.
- **Correcciones, pruebas finales y puesta en producción**.
- **Documentación de todo el proceso,** en esta memoria de TFM.

Una vez cumplidas todas las acciones definidas, se tendrá una herramienta de bioidentificación segura con Blockchain, y por tanto se habrá cumplido al 100% el objetivo principal de este proyecto.

1.3 Antecedentes y estado del arte

1.3.1 Reconocimiento de la identidad mediante rasgos biométricos

Antes de nada, toca recalcar la importancia que tiene la identificación en temas de ciberseguridad, ya que sin identificación no existirían la mayoría de los servicios/aplicaciones que tenemos y usamos a diario, las características principales que aporta a la ciberseguridad son:

- Confidencialidad: Al acceder a una aplicación solo puedes ver datos que pertenecen a tu perfil, y que nadie más puede ver.
- Identidad personal: Permite realizar acciones a tu nombre.
- Legalidad: La identificación es el proceso que permite que la mayoría de las acciones de las apps estén dentro del marco legal correspondiente.
- Automatización de procesos: Permite reducir personal y agilizar multitud de procesos manteniendo la seguridad casi al mismo nivel que la forma tradicional.

A día de hoy existen multitud de tipos de identificación, pero los más comunes son: Contraseñas, tarjetas de identidad, número de identificación personal, bioidentificación y firma electrónica. Por ello, viendo que la bioidentificación se encuentra ya bastante asentada, y pudiendo valorar su crecimiento y ventajas con respecto a los demás sistemas de identificación, se ha decidido que será el método utilizado en este proyecto.

Las primeras noticias de las que se tienen constancia acerca del reconocimiento de personas mediante rasgos biométricos data del año 1882, donde se definió el término antropometría. Este término fue definido por un policía francés llamado Alphonse Bertillon que presentó el primer sistema de identificación de personas basado en rasgos biométricos, con la finalidad de identificar a criminales.
En las primeras etapas de este sistema se clasificaba la forma de la nariz, de la cara o incluso la forma del cuerpo. En la siguiente ilustración podemos ver la clasificación de diferentes clases de narices definidas en la antropometría, publicada en la revista “Pearson´s Magazine” en 1901.

Ilustración 3. Pequeña clasificación de narices definidas en la antropometría

En un principio no se le dio valor ni se apoyó su propuesta, pero años más tarde se dieron cuenta de su enorme eficacia, identificándose a más de 200 infractores reincidentes en un solo año.

Tras unos años, Bertillon decidió mejorar el sistema y enfocarse en el uso de un mayor número de rasgos biométricos, más concretamente en 11 (altura de pie, altura sentado, largo y ancho de la cabeza, largo y ancho de la oreja derecha, largo del pie izquierdo, largo del dedo corazón y meñique de la mano izquierda y largo del antebrazo izquierdo). Para conocer la similitud entre dos personas se calculaba la distancia euclídea entre los vectores formados por los 11 componentes biométricos, se aplicaba una fórmula y si el resultado superaba un umbral se podía concluir que las medidas comparadas pertenecían a la misma persona.

El sistema de identificación de Bertillon no era para nada totalmente fiable, debido principalmente a que estas medidas biométricas pueden cambiar con el tiempo, y además no son únicas entre personas. Por esto, la ciencia criminal empezó a investigar las huellas dactilares ya que veían esta técnica con base más científica.

Tras demostrarse científicamente que no hay dos huellas dactilares iguales, las comisarías policiales empezaron a crear archivos de criminales con sus huellas, además de encontrar una muy buena aplicación en los estudios forenses, permitiendo grandes avances científicos en la ciencia de la biometría basada en la huella dactilar.

Debido al gran incremento de peticiones para comparar rasgos biométricos, y viendo como estas comparaciones son lentas, se ha tenido la necesidad de investigar en la

adquisición y comparación automática de estos rasgos mediante sistemas electrónicos.

A mediados del siglo XX se presentaron los primeros sistemas AFIS (sistema automático de identificación de huellas dactilares), y años más tarde métodos parecidos a AFIS pero para otros rasgos biométricos como iris o cara. Desde entonces, estos sistemas automáticos no han parado de mejorar en precisión, fiabilidad y tiempo.

Más recientemente, con la llegada de los smartphones, desarrollo de sensores más pequeños, más baratos y con un alto grado de precisión, se ha dado un salto cualitativo y cuantitativo a este sector, que junto con las inquietudes de la población en la seguridad de la información y el fraude de la identidad han creado la necesidad de desarrollar, invertir e innovar en métodos biométricos más allá del ámbito forense y centrándose en aplicaciones sociales [3].

Tanto es así, que hoy en día utilizamos sistemas biométricos para verificar la identidad en multitud de acciones diarias, como acceder a las oficinas de la empresa, desbloquear el teléfono, acceder a cuentas bancarias, autenticación segura de acceso al PC, etc. De hecho, según un estudio realizado por “Statista”, a partir de 2020 la totalidad de smartphones, tabletas y weareables fabricados incorporan tecnología biométrica (ilustración 4).

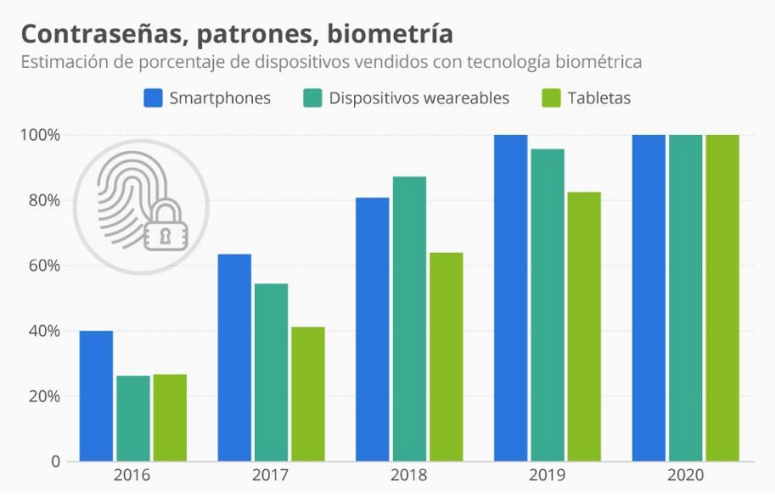


Ilustración 4. Porcentajes de dispositivos vendidos con tecnología biométrica [4]

1.3.2 Blockchain

La tecnología Blockchain es considerada una de las mayores innovaciones del siglo 21, y para poder explicar su historia es recomendable basarse en una línea temporal en la que ir explicando las diferentes fases por las que ha pasado.

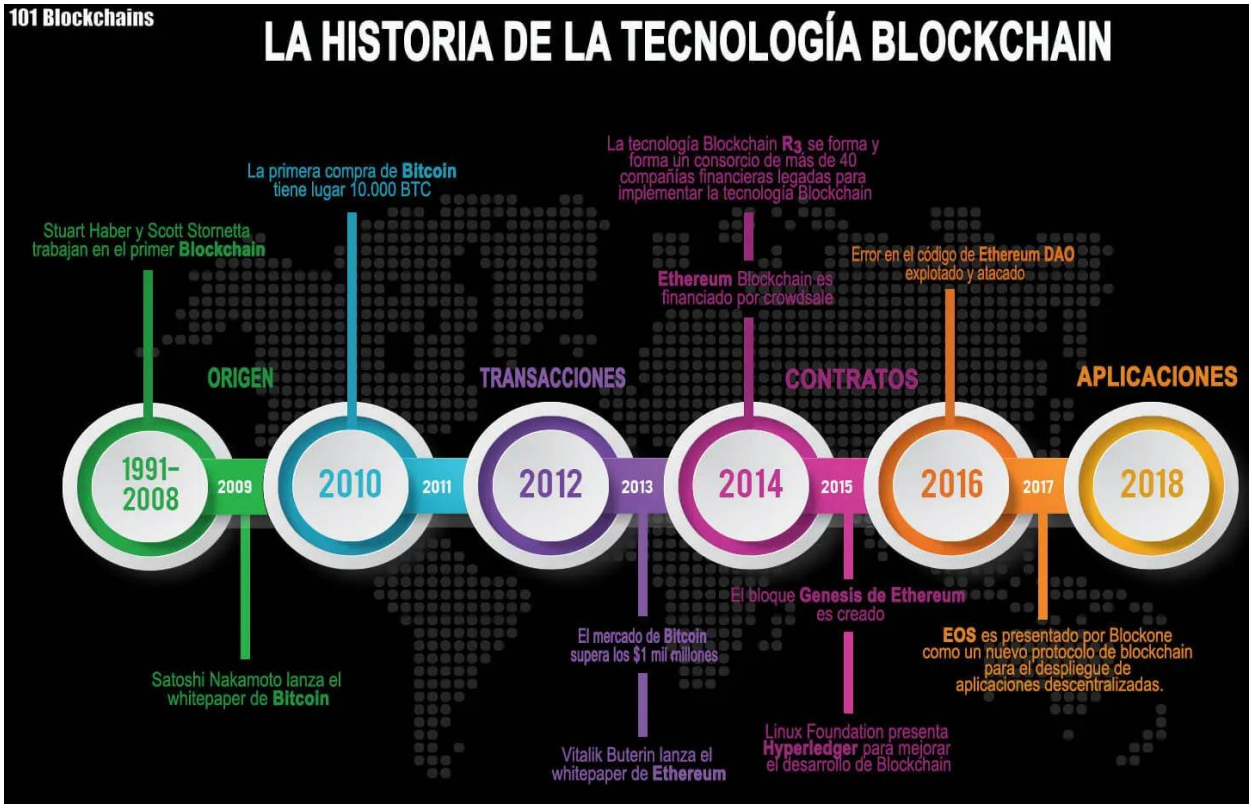


Ilustración 5. Línea del tiempo de la tecnología Blockchain [5]

Fase 0: 1991-2008

Es un concepto que surgió mucho antes de lo que todos nosotros nos creemos, ya que por el año 1991 Stuart Haber y W. Scott Stronetta propusieron un primer concepto de Blockchain, definiendo una idea que consistía principalmente en la creación de una cadena de bloques protegida criptográficamente para obtener marcas de tiempo inalterables dentro de documentos. Un año más tarde, presentaron una versión modificada de su primera idea que trataba de mejorar la eficiencia, apoyándose en los árboles de Merkle podían recopilar más documentos en un solo bloque.

Pero no fue hasta dos décadas después cuando empezó a coger relevancia aquella idea, más concretamente por el año 2008 con la creación de Bitcoin, que supuso el origen del dinero digital garantizando consigo la seguridad, trasparencia y privacidad

entre los usuarios y lo que es más importante: donde se contextualizó el primer Blockchain. En el primer informe que salió a la luz acerca de la tecnología Blockchain por parte del creador de Bitcoin (Satoshi Nakamoto) se proporcionaron detalles de cómo la tecnología estaba bien equipada para mejorar la confianza digital, en un momento donde dicha confianza era prácticamente nula por parte de los usuarios.

Fase 1: 2008-2013

Se puede definir como la fase en la que emerge el Bitcoin y es una primera fase centrada en realización de transacciones. Cabe destacar que, a raíz de lo que supuso Bitcoin para la tecnología Blockchain, mucha gente piensa que son lo mismo, pero no es así, ya que una es la tecnología que permite crear multitud de aplicaciones con características muy interesantes, y el otro concepto solo es una de las aplicaciones que se pueden desarrollar en la tecnología.

En esta etapa se creó el bloque Génesis a partir del cual empezaron a enlazarse los nuevos bloques que iban generándose, obteniendo como resultado un aumento significativo de la cadena gracias al transporte de información y transacciones.

Debido a que cada vez más gente conocía el proyecto Bitcoin también empezaba a sonar la tecnología en la que se basaba, y eso dio lugar a la tercera fase la cual se centra en el desarrollo de nuevas aplicaciones y mejoras de la tecnología.

Fase 2: 2013-2015

Fase crucial en la evolución del Blockchain, ya que el proyecto Ethereum salió a la luz y por consiguiente se hizo público el concepto de Smart contract (contrato inteligente).

Uno de los primeros contribuyentes al código de Bitcoin, Vitalik Buterin, centrándose en las limitaciones de Bitcoin comenzó a desarrollar una red que pudiera realizar varias funciones además de ser P2P, y por el año 2013 nació Ethereum, publicitándose como una nueva Blockchain pública que ofrecía multitud de funcionalidades adicionales a las que ofrecía Bitcoin, como la posibilidad de registrar contratos, lo que daba lugar a una red que aparte de contar con su criptomoneda también permitía crear aplicaciones descentralizadas (Dapps).

Hay que destacar que la Blockchain de Ethereum vio la luz oficialmente 2 años después, en el año 2015 y que desde entonces no ha parado de crecer, siendo actualmente una de las mayores aplicaciones basadas en Blockchain que existe.

Fase 3: 2015-2018

En el 2015 se presentó el proyecto Hyperledger, que busca avanzar en la colaboración de todo tipo de sectores para el desarrollo Blockchain, y que se centra en fomentar el uso de la tecnología para mejorar rendimiento y confiabilidad de los sistemas actuales.

Esta fase está centrada sobre todo en la creación de multitud de aplicaciones como NEO (primera plataforma Blockchain de código abierto), IOTA (centrada en el ecosistema IoT) o Monero (criptomoneda que intenta solucionar diversos problemas de seguridad y privacidad). Pero todo lo comentado hasta ahora se centra en redes Blockchain públicas, y cabe destacar que en esta fase las empresas empezaron a adoptar la tecnología internamente, con la idea de mejorar diversos procesos apoyados en las ventajas que ofrece Blockchain.

Fase 4: 2018-Actualidad

En estos últimos años la tecnología Blockchain no ha parado de crecer, cada vez más y más empresas se suman a proyectos descentralizados y cada vez se invierte más dinero por parte de empresas y organismos públicos.
Actualmente se está trabajando en diversos temas de seguridad, pero principalmente se están desarrollando proyectos de identidad digital y se está comenzando a regular todo por parte de la Unión Europea. Esto hace presagiar el brillante futuro de la tecnología y el asentamiento definitivo en la sociedad a corto-medio plazo [5].

1.3.2.1 Aplicaciones generales

Votación digital: Con el uso del Blockchain se pueden crear sistemas de votación digital que ofrezcan las mismas garantías legales que los sistemas de voto tradicional. Tanto es así que multitud de Ciudades y pueblos en todo el mundo ya lo usan para sus particulares votaciones. Una arquitectura básica de una aplicación que ofrezca este servicio es:

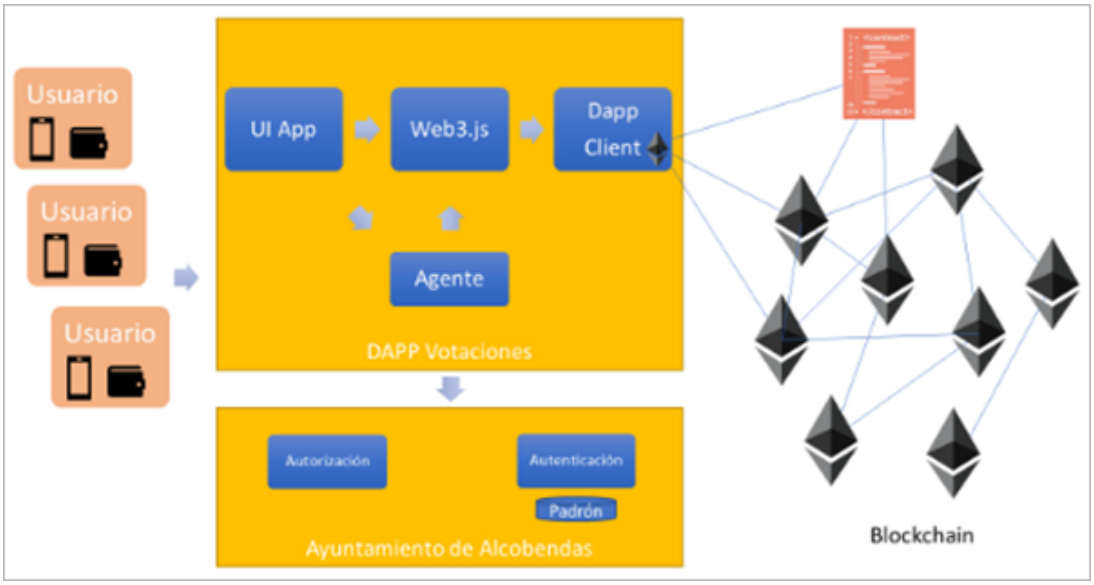


Ilustración 6. Arquitectura básica de la aplicación de votación [6]

Cadenas de producción: Centrado sobre todo en sistemas logísticos, empresas de supervisión de alimentos y trazabilidad de cara al usuario final, permitiendo al comprador ver por todas las etapas que ha pasado el producto hasta llegar a su casa.

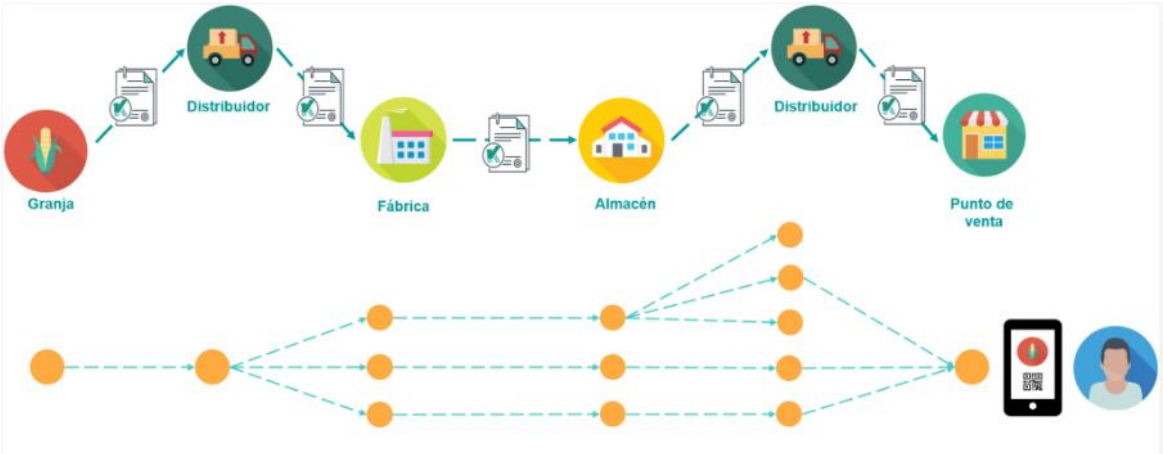


Ilustración 7. Trazabilidad de productos [7]

Contratos digitales: La cadena de bloques ha permitido crear acuerdos comerciales que se renuevan automáticamente mediante un programa informático. Esto aporta una serie de características con respecto a los contratos tradicionales, como la reducción de tiempo y costes. Un ejemplo es Comforce.

Ilustración 8. Sistema de gestión de contratos [8]

1.3.2.2 Aplicaciones orientadas a ciberseguridad

Viendo la evolución y características del Blockchain, y observando cómo evoluciona la sociedad hacia un mundo descentralizado, podemos imaginarnos la multitud de sectores que pueden utilizar la cadena de bloques para crear aplicaciones. Algunas de ellas son:

Almacenamiento en la nube: Aporta seguridad a las empresas, contando con procedimientos que permiten guardar la información en diferentes nodos, situados en distintas ubicaciones. Un ejemplo de esto es Nasuni.

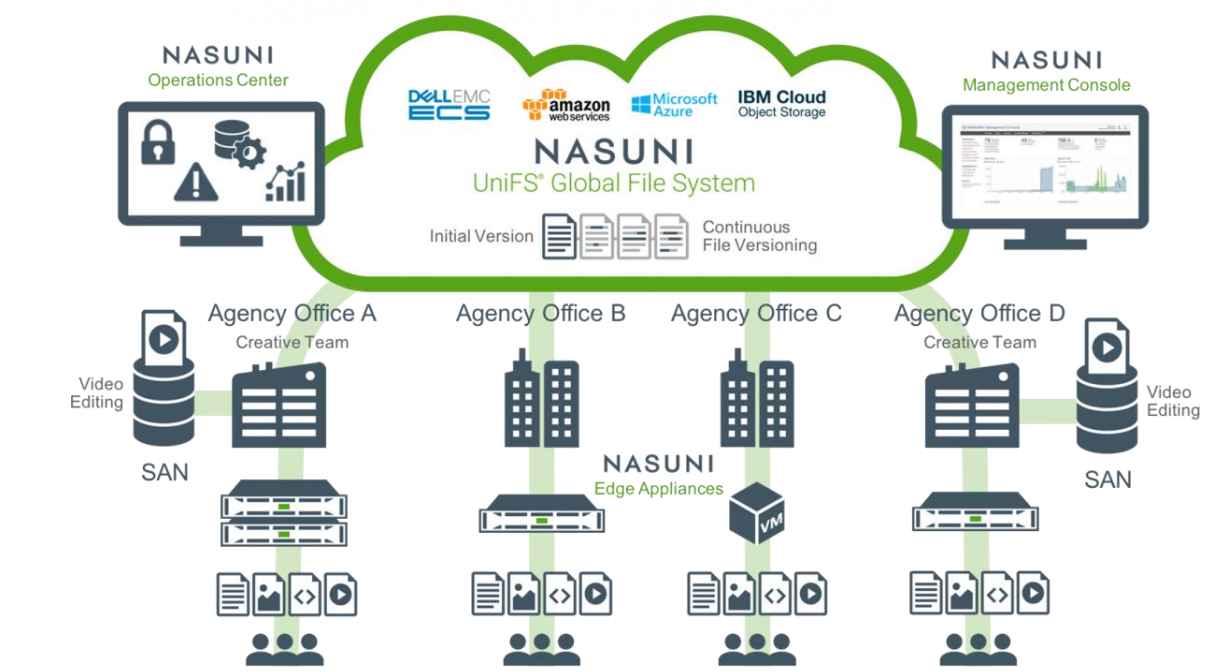


Ilustración 9. Blockchain en el almacenamiento en la nube [9]

Identidades digitales: Permite evitar fraudes y ya encontramos aplicaciones relacionadas con la identidad en firmas electrónicas y automatización. Por otra parte, también está avanzando mucho en temas de registro, identificación y verificación de datos e identidades. Un ejemplo de esto es el proyecto AlastriaID.

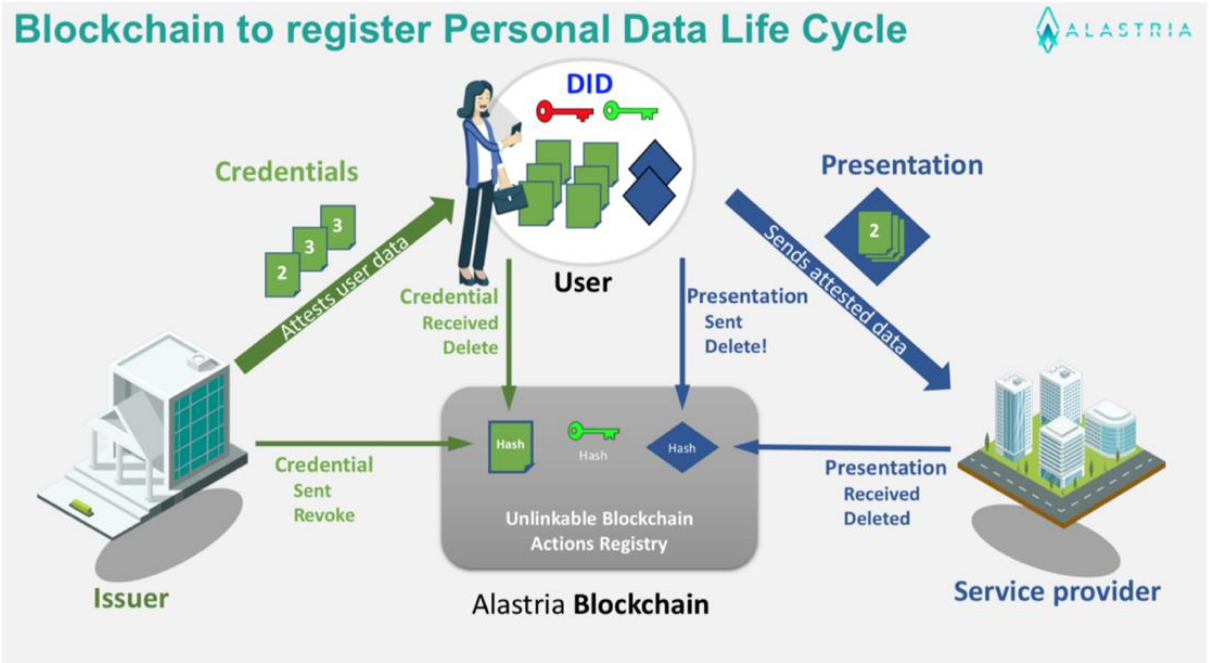


Ilustración 10. Alastria ID [10]

1.4 Descripción de la situación de partida

Como se ha comentado en el apartado de Antecedentes y estado del arte, la bioidentificación no es algo nuevo, pero sí que su uso ha crecido exponencialmente en estos últimos años, empujado principalmente por la fragilidad de la identificación "user/password" y por la facilidad de uso que aporta al usuario este tipo de identificación, ya que no tiene que estar pendiente de recordar ninguna contraseña y le genera mayor seguridad identificarse con un rasgo que solo él posee.

En cuanto a Blockchain cabe destacar el incremento que se produce año tras año en inversiones a esta tecnología, lo que hace esperar que dentro de unos pocos años esté consolidada como una de las tecnologías más importantes. En estos momentos no existen muchos proyectos que intenten aunar estos dos aspectos, por lo que puede ser interesante e innovador el desarrollo del mismo, pudiendo marcar un antes y un después en el proceso de identificación biométrica.

El ayuntamiento de las Rozas en Madrid ha lanzado el reto "Bio-identidad segura con Blockchain", con el cuál intentan crear un sistema de identificación biométrica que les ayude a identificar a sus ciudadanos de una manera rápida y automatizada, y han calculado el impacto que supondría la implementación de ese sistema en su ciudad.

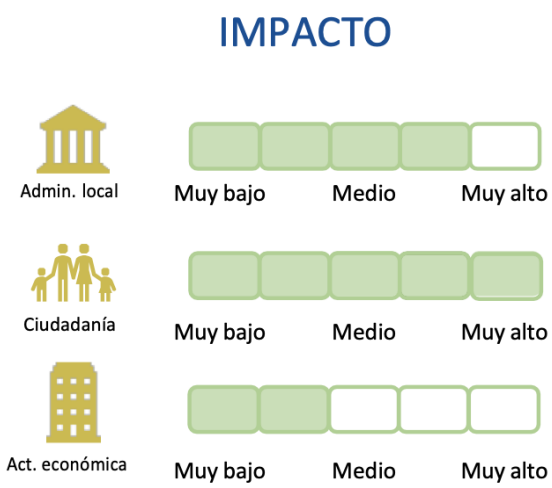


Ilustración 11. Impacto del proyecto en Las Rozas [11]

Como podemos ver en la imagen, supondría un impacto alto en la administración local, un impacto muy alto en la ciudadanía y un impacto medio-bajo en la economía.

1.4.1 Resumen de las deficiencias y carencias identificadas

Una vez definida cuál es la situación de partida, es necesario identificar las deficiencias y carencias que pretende resolver este trabajo. Como ya se ha comentado anteriormente, la bioidentificación está en auge, y cada vez más empresas, sistemas, y servicios implementa este tipo de identificación.

Por ello, este trabajo pretende detectar cuáles son las deficiencias y carencias existentes y poder solventarlas con el uso de herramientas tecnológicas.

Teniendo identificadas las deficiencias, se puede realizar una evaluación y análisis de cuáles son las tecnologías que mejor solventan ese problema, y cuál es la manera más idónea de usar dicha tecnología para sacarle el máximo rendimiento (Tabla 1).

Deficiencia identificada	Solución propuesta
El usuario no sabe cómo actuar para que el sensor escanee correctamente el rasgo biométrico	Proporcionar información de cómo realizar dicha acción e incluso mostrarle un video explicativo
Rasgos biométricos que requieren de mucha colaboración del usuario	Elección de rasgos que tengan una gran facilidad de uso
Problema del RGPD a la hora de almacenar datos biométricos	Utilizar el secreto compartido de Shamir junto con un modelo de confianza cero
Lentitud de la red Blockchain a la hora de validar bloques	Almacenar los bloques no validados hasta que se pueda completar la validación. (no poner timeout)
Usuario que sufre lesiones en el rasgo registrado	Método de verificación alternativo en estos casos
Pérdida de la información biométrica almacenada	Uso de la tecnología Blockchain para distribuir copias de estos datos

Tabla 1. Identificación de deficiencias y posibles soluciones

1.4.2 Experiencia previa

Como se ha comentado en los puntos anteriores, este proyecto se centra en 2 bloques bien diferenciados, como son el sistema móvil de identificación biométrica y una infraestructura en Blockchain para almacenar los datos necesarios.

Por una parte, en cuanto al desarrollo de aplicaciones móviles, se tiene cierta experiencia en Android e IOS, gracias a diversas asignaturas cursadas tanto en la carrera como en el máster de ingeniería informática, pero no en el desarrollo multiplataforma, que es donde se pretende realizar.

Por otra parte, con respecto a la tecnología Blockchain cabe destacar que no tenía experiencia previa, pero gracias a la realización de las prácticas de empresa del máster en seguridad he podido iniciarme y adentrarme en esta tecnología.

Como apunte, destacar que para el desarrollo de este TFM he estado en contacto directo con técnicos de Alastria, realizando reuniones que han permitido resolver ciertas dudas y problemas típicos de un desarrollo software. Y además, he tenido la suerte de estar en contacto con especialistas en Blockchain (Banco Santander, DigitelTS, Telefónica, Inetum, Wealize, etc), que gracias a su invitación me ha permitido asistir a las reuniones semanales del proyecto Alastria ID, del cual he podido sacar y adaptar varios conceptos a este trabajo, como por ejemplo el uso de la clave compartida de Shamir para el almacenamiento de la clave hash biométrica.


NOTICIAS ➤ SEGURIDAD

Telefónica Tech e Inetum desplegarán Alastria ID

Se trata de un modelo de identidad autogestionada (Self-Sovereign Identity), definido por la Comisión de Identidad de este consorcio en su red blockchain productiva.

PRIVACIDAD

La app que promete acabar con el papeleo

Un grupo de grandes corporaciones y el consorcio de 'blockchain' Alastria trabajan para devolver a las personas el control de sus datos (y obtener de ellos información de mayor calidad). Aspiran a cumplir el sueño de la identidad digital soberana.

Ilustración 12. Noticias de Computing.es y Elpaís.com sobre Alastria ID

1.5 Alcance

El alcance del proyecto nos permite definir hasta dónde llega el trabajo, detallando cuáles van a ser los entregables que se harán una vez finalizado el mismo. No sólo se especifican los resultados vinculados al producto final, sino que también se detallan los entregables que hacen referencia a la gestión y control de la ejecución del proyecto.

1. Informe resultante del estudio entre los diferentes rasgos biométricos, donde se realizará una comparación entre los principales rasgos usados para la identificación, comparando parámetros como la falsa aceptación, falso rechazo, fiabilidad, coste, tamaño y facilidad de uso.

2. Código fuente del sistema biométrico desarrollado. Dependiendo de la plataforma para la que se desarrolle contará con un lenguaje de programación u otro, y englobará todo el código necesario para la ejecución del sistema de identificación biométrico.

3. Arquitectura final del sistema. Se presenta un esquema donde se puede ver de forma visual cómo funciona a grandes rasgos, y que cuenta con una explicación detallada del mismo.

4. Pequeña guía que explique la integración del sistema biométrico en la tecnología Blockchain, ya que al ser dos modalidades muy distintas resulta interesante incluir un documento en el que se detalle todo el proceso de integración.

5. Ejecutable de la aplicación que permita la correcta distribución e instalación del software en la plataforma elegida.

6. Documentación del proyecto, que será este mismo documento, donde se describen todos los procedimientos seguidos, se justifican las decisiones

tomadas y se detalla toda la información necesaria seguida en el desarrollo del mismo.

1.6 Hipótesis y restricciones

El TFM se define como una asignatura de 12 créditos, lo que supone que la duración total del proyecto será de 300 horas incluyendo todas las etapas del ciclo de vida, con la excepción del mantenimiento. Por consiguiente, la principal restricción aplicable es la limitación de la duración del trabajo.

Por otra parte, dentro del desarrollo del mismo, se encuentran una serie de restricciones que son enumeradas a continuación:

- No apto para toda la población: Hace referencia al momento de identificación en el sistema, ya que como hay que escoger un rasgo específico puede que el usuario no cuente con él por diversos motivos tanto temporales (ej: manos escayoladas por algún accidente) como permanentes (ej: amputación).
- Necesario el uso de la aplicación en un dispositivo que cuente con el sensor biométrico de identificación adecuado: Este sensor será el encargado de realizar el proceso de identificación por lo que es obligatorio tenerlo para que el sistema funcione, un ejemplo de esto puede ser un móvil sin sensor de reconocimiento de huella dactilar.
- Capacidad de la red Blockchain: Dependiendo de las características que tenga la red Blockchain con la que se trabaje, el sistema desarrollado contará con unos límites de capacidad y de velocidad.

Finalmente, se debe definir la hipótesis principal que guía este trabajo, que trata de cumplir con el objetivo principal y los objetivos específicos descritos en el apartado 1.2, por lo que se puede definir como el desarrollo de un sistema de identificación biométrico apoyado en la tecnología Blockchain, pudiendo darle una funcionalidad adecuada y una interfaz que siga los estilos y estándares pertinentes.

1.7 Estudio de alternativas y viabilidad

En este apartado se enumerarán las alternativas que se han tenido en cuenta, se justificará la elegida y las razones por las que se han descartado las demás. Hay que tener presente que este trabajo cuenta con dos partes muy diferenciadas, como son: El sistema de identificación biométrico, y la configuración y puesta a punto de la red Blockchain, que servirá para almacenar la información biométrica de los usuarios aportando multitud de ventajas con respecto al almacenamiento actual.

1.7.1 Sistema de identificación biométrico

Existe un gran abanico de posibilidades con respecto a la elección del rasgo biométrico a usar en nuestro sistema. Cabe destacar, que lo idóneo en estos tipos de sistemas es el uso de más de un rasgo biométrico, ya que la combinación de varios de ellos puede dar lugar a la obtención de un sistema altamente seguro, ya que como sabemos ningún sistema de identificación es seguro por completo.

En este sistema, vamos a priorizar dos características muy importantes, como son el coste de adquisición y la fiabilidad. Por lo que pretendemos desarrollar un sistema que se pueda implementar a un bajo coste y que su seguridad sea la más alta posible, optimizando al máximo la relación coste/seguridad. Para lograr este objetivo, se necesita realizar un estudio de los diferentes rasgos biométricos para posteriormente decidir cuáles de ellos serán usados en el sistema.

Definimos una escala que permita cuantizar cuánto de bueno o malo es un rasgo según ciertas características, para así posteriormente cuantizar y calcular la relación coste/seguridad. A cada uno de los elementos le asignamos una puntuación:

Escala	Puntuación
Muy alto	9
Alto	7
Medio	5
Bajo	3
Muy bajo	1

Tabla 2. Escala de puntuación

Una vez definida la escala que se va a usar, pasamos a comparar los diferentes rasgos biométricos con la idea de poder seleccionar aquellos que mejor se adapten a nuestro objetivo.

Para poner en contexto a cada uno de los rasgos biométricos y poder realizar una comparación más completa, se van a medir parámetros como la fiabilidad, facilidad de uso, prevención de ataques, aceptación y el coste total que supondría la adquisición de un sistema que autentifique por ese rasgo biométrico.

Rasgo biométrico	Fiabilidad	Facilidad de uso	Prevención de ataques	Aceptación	Coste	Relación seguridad/coste
Huella dactilar	Muy alta	Alta	Alta	Alta	Bajo	30/3 = **10**
Voz	Alta	Alta	Media	Bajo	Bajo	22/3 = **7,3**
Cara 3D	Alta	Alta	Alta	Media	Bajo	26/3 = **8,6**
Iris/retina	Muy alta	Media	Muy alta	Media	Alto	28/7 = **4**
Geometría dedos/mano	Alta	Alta	Alta	Bajo	Medio	24/5 = **4,8**
Vascular	Muy alta	Muy alta	Alta	Bajo	Medio	28/5 = **5,6**

Tabla 3. Comparativa entre los diferentes rasgos biométricos [12] [13]

Tras la comparación, se ha obtenido una lista ordenada de mayor a menor puntuación de los rasgos biométricos en función a su relación entre la seguridad y el coste.

Al ser un proyecto flexible, se pueden plantear diferentes situaciones, como el uso de un rasgo biométrico para acceder a aplicaciones estándar, o el uso de la combinación de varios rasgos biométricos para sistemas de máxima seguridad (bancos, salud, datos personales, etc). En el primer supuesto (apps con seguridad media) se puede usar el reconocimiento facial, ya que todos los móviles disponen de cámara frontal y sería muy cómodo, mientras que para apps que requieran seguridad máxima se podría usar el método de huella dactilar y posteriormente el método de reconocimiento facial 3D, consiguiendo un sistema robusto y fiable en la etapa más delicada de un sistema informático, la identificación del usuario.

1.7.1.1 Hardware y software necesario

Todo sistema biométrico cuenta con dos componentes fundamentales, el hardware y el software, necesarios para el proceso de reconocimiento. Dentro del componente hardware se incluyen los sensores y todos aquellos dispositivos que permiten extraer la característica deseada, y una vez obtenida dicha información, se realizan tareas de acondicionamiento de los datos para que sean utilizados por el sistema biométrico (software).

Como tenemos dos métodos diferentes de identificación por rasgos biométricos, contamos con hardware y software diferente a la hora de obtener y procesar dicha información. En el caso del reconocimiento mediante **huella dactilar**, requerimos:

Hardware

Existen diferentes tipos de sensores para el escaneo de huellas, ordenándose de más antiguos a más modernos, tenemos ópticos, capacitivos y ultrasónicos.

*** Ópticos:** Este escáner implica el uso de luz óptica para capturar y escanear huellas digitales. Funciona capturando una fotografía digital de la huella dactilar, para posteriormente mediante algoritmos encontrar patrones únicos de líneas y crestas. Este tipo de escáneres están desfasados, y se usan en sistemas y móviles de bajo presupuesto, debido a que tienen un diseño voluminoso y pueden ser fácilmente engañados con prótesis o imágenes de alta resolución.

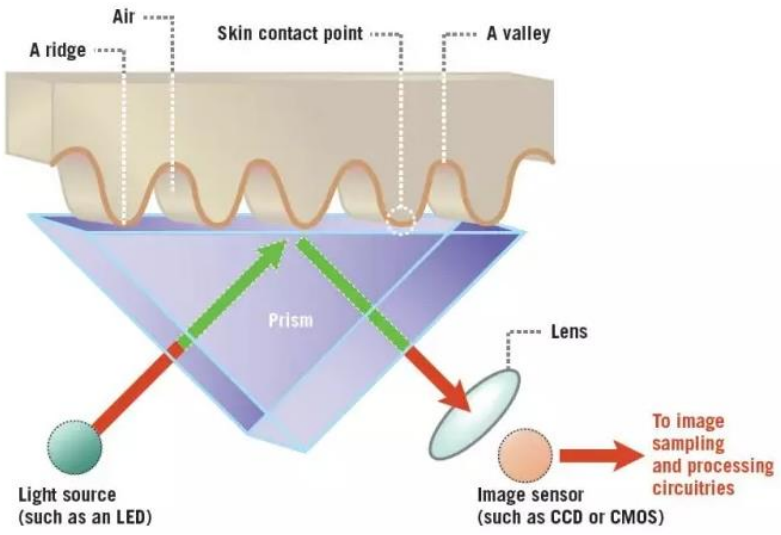


Ilustración 13. Sensor óptico [14]

*** Capacitivos:** A diferencia de los ópticos (capturan imagen 2D de la huella), los escáneres capacitivos capturan diferentes detalles de la huella mediante el uso de señales eléctricas, gracias a la utilización de condensadores que almacenan datos de la misma. En este proceso al no haber captura de imágenes 2D, los datos de huellas son mucho más seguros y no pueden ser fácilmente engañados con prótesis o fotografías, dando lugar a su utilización masiva hoy en día.

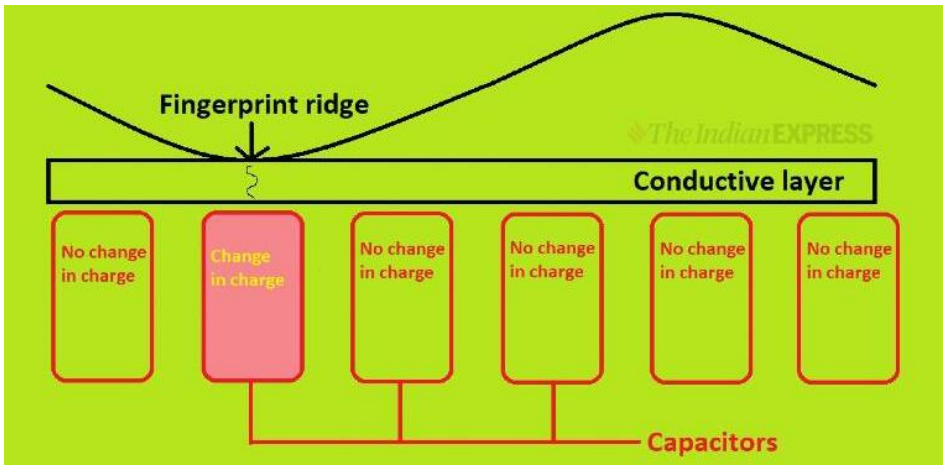


Ilustración 14. Sensor capacitivo [15]

*** Ultrasónicos:** Utiliza un sonido ultrasónico de una frecuencia muy alta, necesitando la combinación de transmisor y receptor ultrasónico. Se produce un pulso ultrasónico, que se envía a través del transmisor, y tras golpear en la huella, se recoge la parte reflejada de la señal por el receptor, capturando una representación 3D de la misma. De todos modos, hay que tener en cuenta que cuanto más tiempo permanezca el dedo en el escáner se obtendrá una representación más detallada.

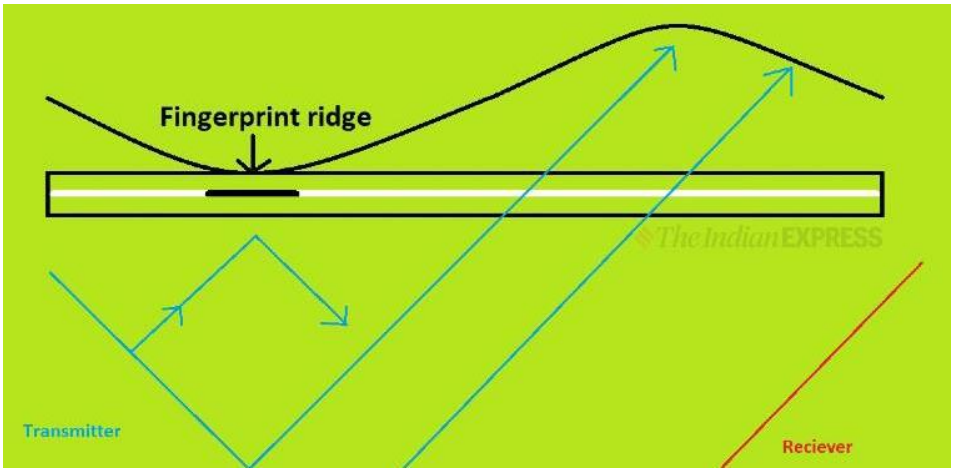


Ilustración 15. Sensor ultrasónico [15]

Para nuestro sistema nos decantamos por usar un sensor capacitivo, ya que es el predominante actualmente, es bastante fiable y fácil de instalar [16]. Por otra parte, no debemos olvidarnos que una vez obtenida y procesada la imagen de la huella, debemos almacenar las características que hemos extraído mediante el software, para así comparar estos datos con los almacenados en posteriores identificaciones. Por ello, otro elemento hardware necesario para la creación de un sistema de identificación por huella dactilar, es un disco duro, que permite almacenar gran cantidad de datos y características de cada uno de los usuarios del sistema.

Software

Tras la recogida de información por parte del hardware, toca procesarla, almacenarla y contar con un mecanismo para comparar la huella actual con las de la base de datos. Para ello, contamos con multitud de programas, tanto de pago como gratuitos, algunos de ellos son:

*** Biometric Evaluation Framework NIST:** Es un conjunto de clases de C++, códigos de error y patrones de diseño utilizados para crear un entorno común que proporcione registro, gestión de datos, manejo de errores y otras funcionalidades que se necesitan para muchas aplicaciones utilizadas en las pruebas de software biométrico.

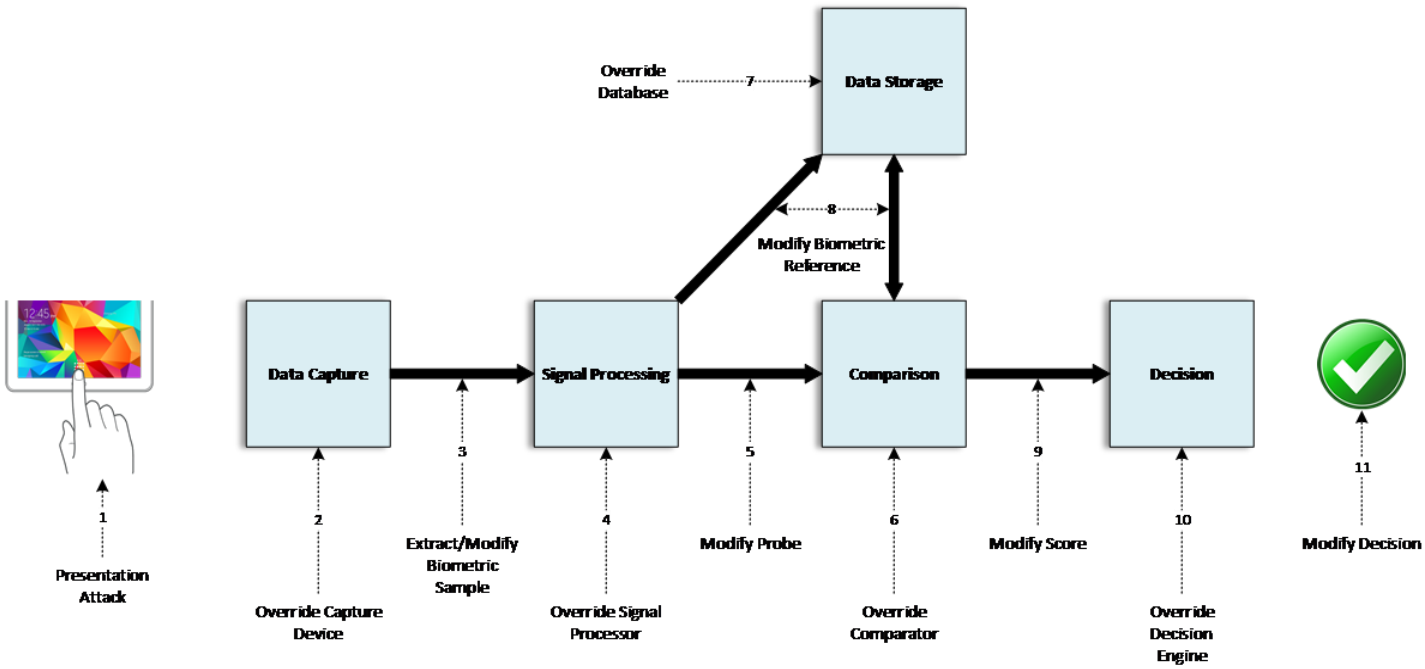


Ilustración 16. Biometric Evaluation Framework [17]

*** NIST Biometric Image Software (NBIS):** Implementación de referencia del estándar AN2K. Un sistema de clasificación de patrones de huellas dactilares basado en la red neuronal PCASYS, una gran colección de utilidades de imágenes de uso general (IMGTOOLS) para apoyar el procesamiento de imágenes de huellas dactilares y un algoritmo de coincidencia de huellas dactilares.

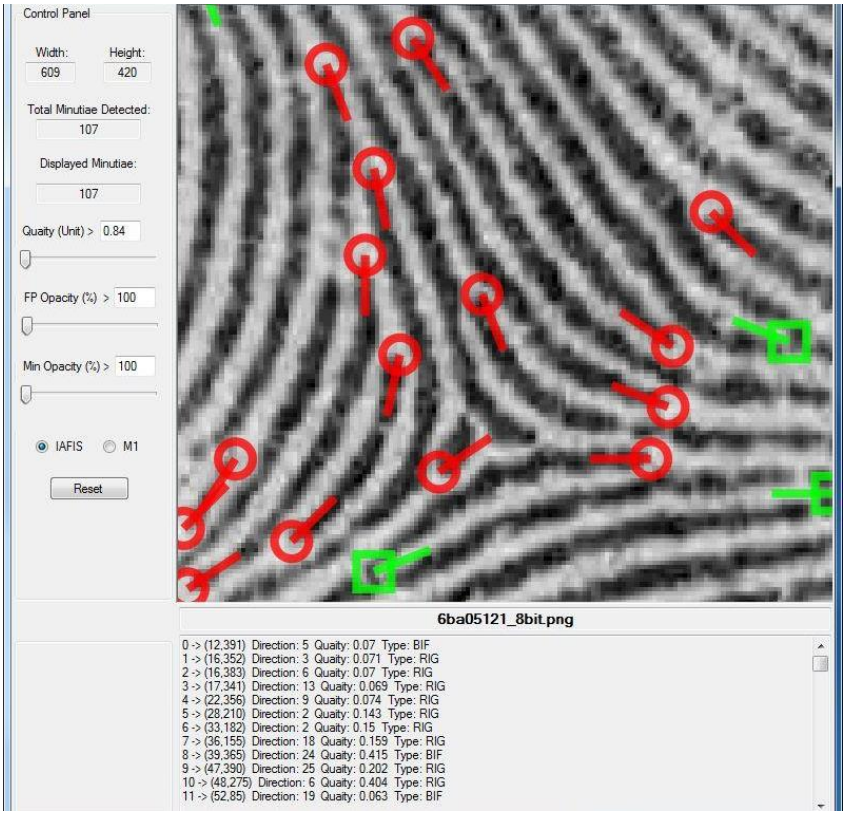


Ilustración 17. NBIS [18]

*** Detect-faces:** API de Google que permite detectar varios rostros en una imagen y extraer de cada uno de ellos sus atributos faciales clave. Esta funcionalidad se encuentra dentro del servicio “Cloud Vision” y cabe destacar que es de pago.

Ilustración 18. Detect-faces Google [19]

*** Face recognition:** API desarrollada por Luxand que detecta y compara rostros humanos, siendo capaz de predecir edad, género y emociones en el reconocimiento. Cuenta con un periodo de prueba gratuito y posteriormente se pasa al plan de pago.

Ilustración 19. Face-recognition Luxand [20]

Cada una de estas soluciones cuenta con características diferentes, como diferentes funcionalidades, APIs, y licencias, por lo que antes de elegir un framework deberíamos realizar un estudio sobre cuál es el que más convendría usar dependiendo del caso de uso.

Y para el **reconocimiento facial en 3D**, necesitamos:

Hardware

A diferencia de la cantidad de sensores que se encuentran disponibles para el reconocimiento de huella dactilar, para el reconocimiento facial únicamente nos hace falta un elemento hardware, una cámara, que permita generar y obtener las imágenes y datos necesarios para crear y registrar el patrón biométrico facial de la persona que se pretende identificar. Este patrón biométrico, al igual que sucede con el reconocimiento de huella dactilar, debe ser almacenado, por lo que se requiere otro elemento hardware a parte de la cámara, un disco duro, en el que poder almacenar los patrones y poder consultarlos y compararlos con los rasgos de la persona que desea identificarse en un momento dado.

Software

Una vez la cámara obtiene la imagen se pasa a la capa software, en la cual se procesa y extraen las características necesarias para la identificación. Se realiza una comparación entre los datos almacenados junto con la imagen de entrada, analizando matemáticamente las dos para determinar si coinciden.

Este tipo de programas se apoyan en técnicas de inteligencia artificial y aprendizaje automático, llegando a obtener sistemas de reconocimiento facial que cuenten con los estándares más altos de seguridad y fiabilidad. Algunos de estos programas que permiten completar el proceso de reconocimiento facial junto con el hardware, son:

- SmileID: Software de Electronic Identification que permite implementar una autenticación facial, y que cuenta con algoritmos de inteligencia artificial y aprendizaje automático para ofrecer una fiabilidad total [21].
- Nexa Face: SDK de algoritmo de comparación facial, desarrollado por Aware [22].
- FindFace: Solución software desarrollada por Ntech Lab, para negocios (minorista, casinos, seguridad corporativa) y para gobiernos (seguridad pública, transporte, instituciones educativas) [23].

1.7.2 Red Blockchain

En cuanto a la tecnología Blockchain, se pueden definir 3 alternativas de implementación, ya sea una red pública, permisionada o privada.

Red pública: Aquella red en que cualquiera puede unirse y participar, pero que cuenta con diversas desventajas como que requiere una gran potencia computacional, poca privacidad para las transacciones y seguridad más baja que el resto de redes Blockchain.

Red permisionada: La responsabilidad de mantener la red recae en un grupo de organizaciones, pudiendo determinar quién tiene permiso para acceder a los datos o enviar transacciones.

Red privada: Administrada por una sola organización la cual controla quien tiene permiso, cuando ejecutar un protocolo de consenso y es la responsable del mantenimiento del libro [24].

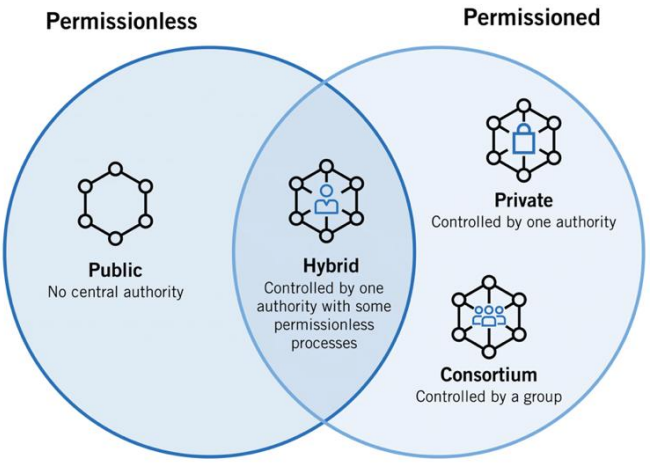


Ilustración 20. Tipos de redes Blockchain [25]

1.8 Descripción de la solución propuesta

Para poder realizar una descripción breve de la solución propuesta se enumerarán una serie de características significativas de la solución con el fin de conocer los principales detalles del proyecto.

Como ya se ha comentado al inicio de este documento, el objetivo de la realización del proyecto se centra en obtener un sistema que permita identificar personas mediante alguno de sus rasgos biométricos y poder guardar dichos datos y los diferentes accesos en una red Blockchain. Los puntos que mayor complejidad tienen a la hora de desarrollar esta propuesta son la obtención de un sistema biométrico fiable, y la integración del mismo en una red Blockchain.

Tal y como se ha visto en el apartado anterior, se cuenta con multitud de alternativas para el desarrollo de este proyecto, pero finalmente toca decidir qué combinación de ellas escoger para obtener el mejor sistema posible. En este caso se han decidido escoger las siguientes:

- Sistema de identificación biométrico: Huella dactilar mediante un sensor capacitivo y el software NIST Biometric Image (NBIS) y reconocimiento facial con el software SmileID.

- Red Blockchain: Se ha optado por una red de tipo permisionada, ya que es una red más controlada, administrada por un grupo de organizaciones y cuenta con mayor privacidad que la red Blockchain pública.
- Interfaz: Como bien sabemos predominan 3 formas de desarrollo móvil hoy día como son IOS, Android y multiplataforma, y tras evaluar las 3 se ha decidido que la interfaz que permita al usuario interactuar con el sistema sea multiplataforma, y con ello poder llegar a más de un 90% de la población que cuente con Smartphone.

1.9 Tecnologías utilizadas

Para obtener la solución final, el proyecto hace uso de una serie de tecnologías necesarias para poder desarrollar todas las funcionalidades esperadas. Estas tecnologías se justifican a continuación.

- Software usado para la documentación del proyecto: **Microsoft Word**, **Google Docs**, **Google Slides**. Software que permite explicar a través de texto e imágenes conceptos derivados del desarrollo del proyecto, y que facilita la comprensión del mismo.
- Planificación temporal: **GanttProject**. Herramienta que permite distribuir las tareas del proyecto a lo largo del tiempo, sacando como resultado un diagrama de Gantt.
- Máquina dedicada a mantener un nodo regular levantado en la Blockchain: **Amazon Web Services (AWS).** Servicio de Amazon que permite alquilar una instancia con las características hardware que necesites, permitiendo con esto poder operar con la Blockchain desde una máquina dedicada.
- Conexión remota con la instancia de AWS: **SSH**. Protocolo de conexión remota segura que ayuda a acceder a la instancia de AWS contratada.
- Descarga de repositorios necesarios: **GitHub**. Software que permite descargar los repositorios necesarios para poder instalar un nodo en la instancia AWS.

- Desarrollo y despliegue de Smart Contracts: **Remix**. IDE de programación que permite desarrollar y desplegar contratos bajo la red Ethereum.

1.10 Metodología de desarrollo de software

Para que un proyecto de ingeniería consiga lograr el resultado esperado, debe seguir una metodología de desarrollo software, ya que es el elemento fundamental para conseguir un desarrollo ordenado, óptimo y verificado.

A la hora de seleccionar la metodología que se va a seguir en este proyecto, hay que evaluar las características, restricciones y finalidad, por lo que siendo un proyecto experimental que cuenta con una fuerte restricción temporal y que solo se puede desarrollar por una persona, parece destacar una metodología por encima de las demás, y es la basada en incrementos. Por tanto, este Trabajo Fin de Máster seguirá la llamada metodología incremental, cuyo objetivo es el de obtener un crecimiento progresivo de la funcionalidad. Con esto el producto va evolucionando con las diversas entregas que se realizan hasta conseguir el resultado final.
Cada uno de los incrementos cuenta con 5 fases: Comunicación, planeación, modelado, construcción y despliegue.

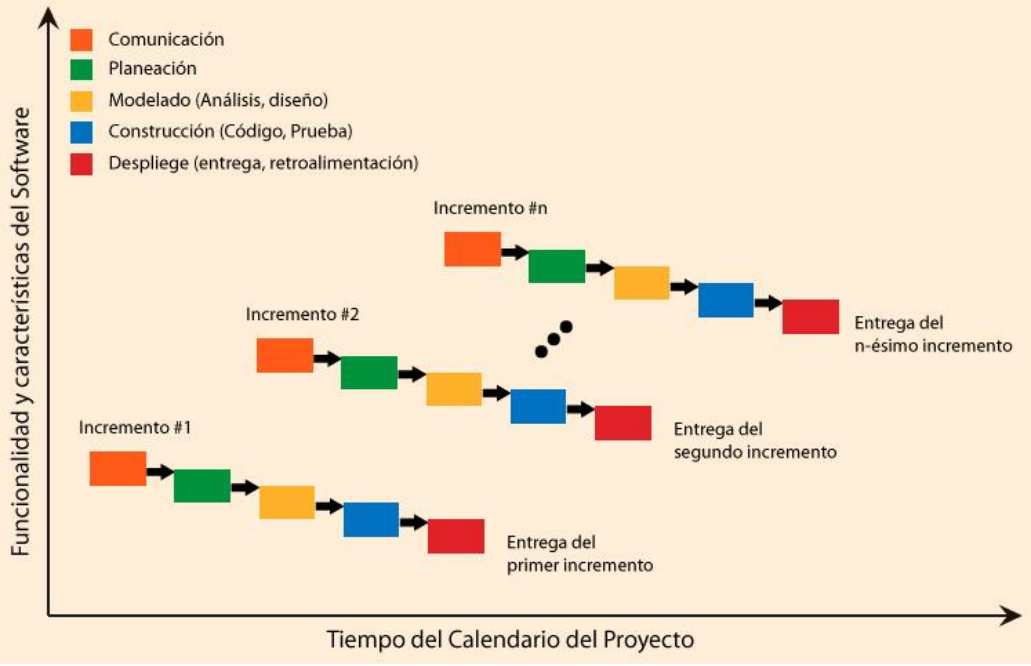


Ilustración 21. Modelo del proceso incremental [26]

Y cada una de las fases se definen en la siguiente imagen:

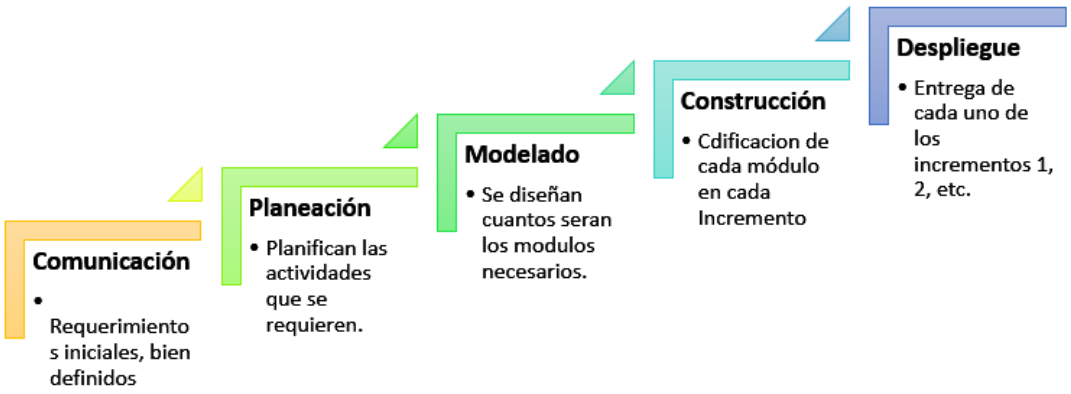


Ilustración 22. Explicación de cada una de las fases [27]

Para poder hacer un buen uso de este tipo de desarrollo, deben quedar claras las diferentes ventajas que supone trabajar con él, y los procedimientos que deberán seguirse para hacer un uso adecuado y que permita obtener un producto final de calidad de una forma óptima. Las principales características de esta metodología son:

- Se adapta a las necesidades que surja.
- Es un modelo propicio a cambios.
- Incrementos pequeños.
- Fácil administración de tareas.
- Facilidad a la hora de gestionar riesgos.
- Es más sencillo probar y depurar en iteraciones pequeñas.
- Modelo más flexible, reduciendo así el coste si se produce un cambio de alcance o requisitos.

1.11 Análisis de riesgos

Uno de los puntos más importantes a la hora de plantear y desarrollar un proyecto es la identificación de riesgos, que se centra en aquellos aspectos relevantes que pueden condicionar el correcto desarrollo de la planificación temporal, económica, o incluso social que pueden afectar a la óptima resolución del proyecto. Para poder obtener una visión clara de la probabilidad y del impacto de cada uno de los riesgos, se va a utilizar una matriz de probabilidad-impacto (ilustración 22).

		Riesgo = Probabilidad x Impacto				
Probabilidad	0,9	0,05	0,09	0,18	0,36	0,72
	0,7	0,04	0,07	0,14	0,28	0,56
	0,5	0,03	0,05	0,10	0,20	0,40
	0,3	0,02	0,03	0,06	0,012	0,24
	0,1	0,01	0,01	0,02	0,04	0,08
		0,05	0,10	0,20	0,40	0,80
		Impacto				

Ilustración 23. Matriz de probabilidad-impacto

A partir de la matriz anterior, se puede valorar cada uno de los riesgos identificados en el proyecto, con la idea de conocer cuáles son los más peligrosos y por tantos más prioritarios.

Para realizar una valoración de los riesgos de una forma estructurada y que pueda comprenderse fácilmente, se agruparán en la siguiente tabla (Tabla 5) estos conceptos:

- Riesgo identificado: Descripción breve del riesgo.
- Impacto: Valorará el nivel de peso que supondría ese riesgo en el desarrollo del trabajo. Las categorías serán: Muy bajo (0.05), bajo (0.1), medio (0.2), alto (0.4) y muy alto (0.8).
- Probabilidad: Valorará la posibilidad de que ocurra dicho riesgo. Las categorías serán: Muy bajo (0.1), bajo (0.3), medio (0.5), alto (0.7) y muy alto (0.9).

- Nivel: Agrupará los riesgos en tres niveles (verde, amarillo y rojo), siguiendo los valores de impacto y probabilidad. Teniendo como base la matriz de riesgo expuesta anteriormente.

Valoración de riesgos			
Riesgo identificado	**Impacto**	**Probabilidad**	**Nivel**
Planificación temporal no ajustada a la realidad	Alto	Bajo	0.12
Falta de habilidad o conocimiento en alguna de las partes del proyecto	Alto	Medio	0.2
Fallos en la arquitectura del sistema tras la fase de desarrollo	Muy alto	Bajo	0.24
Fallo en el sensor de identificación	Muy alto	Muy bajo	0.08
Usuarios que no saben realizar correctamente la acción de identificación	Medio	Bajo	0.06
Pérdida total o parcial del proyecto desarrollado	Muy alto	Muy bajo	0.08
Rendimiento del sistema inferior al esperado	Medio	Bajo	0.06
Calidad de la interfaz de usuario inferior a la deseada	Medio	Bajo	0.06

Tabla 4. Valoración de los riesgos encontrados

Una vez identificados los riesgos y clasificados dentro de un nivel de peligrosidad para el desarrollo del proyecto, hay que proponer diferentes acciones para efectuar en el caso de que sucediera alguno de ellos. Las acciones posibles a realizar son las de mitigar, aceptar, evitar o trasferir, teniendo en cuenta que la opción de aceptar no requerirá de ninguna acción y la de evitar eliminará por completo dicho riesgo. Estas acciones se recogen en el plan de contingencia (Tabla 3).

Plan de contingencia		
Riesgo	**Respuesta**	**Plan de contingencia**
Planificación temporal no ajustada a la realidad	Mitigar	Realizar jornadas de seguimiento que permita reordenar las tareas para así ajustarse a los límites temporales.
Falta de habilidad o conocimiento en alguna de las partes del proyecto	Mitigar	Obtener el conocimiento necesario a través de lecturas o cursos, permitiendo superar esas habilidades en el menor tiempo posible.
Fallos en la arquitectura del sistema tras la fase de desarrollo	Mitigar	Validar que toda la arquitectura del sistema cumple con el objetivo final y comprobar que no existen vulnerabilidades relacionadas con la arquitectura planteada
Fallo en el sensor de identificación	Mitigar	Instalando sensores de calidad y con garantía por parte del vendedor
Usuarios que no saben realizar correctamente la acción de identificación	Evitar	Tutoriales y documentación para que los usuarios puedan aprender a usar de forma correcta el sensor y con ello poder identificarse en el sistema sin problemas.
Pérdida total o parcial del proyecto desarrollado	Mitigar	Realizando copias de seguridad cuando se acabe de desarrollar un entregable.
Rendimiento del sistema inferior al esperado	Mitigar	Evaluando y comprobando el rendimiento de cada entregable, permitiendo modificar aquellas implementaciones que ofrezcan resultados por debajo de un umbral.
Calidad de la interfaz de usuario inferior a la deseada	Evitar	Realizando un estudio exhaustivo acerca de los requisitos que debe cumplir y haciendo un prototipo muy realista del diseño final de la propia interfaz.

Tabla 5. Plan de contingencia de los riesgos asociados al proyecto

1.12 Estimación del tamaño y esfuerzo

La estimación del tamaño y esfuerzo sirve como base para la planificación temporal y el presupuesto, que serán conceptos que se detallarán en apartados posteriores.
Tiene como objetivo detallar y estimar la extensión y personal necesarios para la ejecución del trabajo. Teniendo presente que se enmarca dentro de un Trabajo Fin de Máster, y por consiguiente cuenta con una restricción temporal (en torno a 300 horas) y una restricción de personal (sólo se dispone del autor del trabajo), por lo que, de forma independiente a lo expuesto en este apartado, deberán ser consideradas las dos limitaciones comentadas.

Para especificar las métricas que se aplican al trabajo se debe elegir alguna de las técnicas de estimación, como pueden ser Planning Poker, Bucket System, Story Points, Dot Voting, etc. En este caso se ha optado por usar Story Points, ya que es una técnica sencilla, que se adapta muy bien al problema, y que hace uso de conceptos que ya se han desarrollado en apartados anteriores, y describe otros nuevos que servirán para facilitar el desarrollo del apartado siguiente, el de planificación temporal.

Los Story Points se pueden definir como una medida que permite expresar el esfuerzo estimado de cada una de las subtareas del proyecto. Destacando que el valor asignado individualmente a cada subtarea es irrelevante, ya que lo importante es el valor relativo que guarda esa tarea con respecto a las otras, que será lo que determinará el esfuerzo necesario. Un ejemplo de esto podría ser una tarea que tiene esfuerzo 2 y otra que tiene esfuerzo 4, se determinaría que la segunda tarea requiere el doble de esfuerzo que la primera.

De cada tarea se necesita determinar la cantidad de trabajo a realizar, la complejidad de dicho trabajo, la experiencia previa y el riesgo o incertidumbre [29], detallándose todo esto en la siguiente tabla.

Tarea	ID	Horas	Complejidad	Experiencia previa	Riesgo	Esfuerzo estimado
Planificación y estudio de la necesidad	1	10	1	Si	Bajo	**1**
Estudio para elegir el rasgo biométrico más idóneo en el sistema	2	10	2	Si	Bajo	**1.5**
Informe sobre la necesidad del uso del Blockchain en el proyecto	3	10	2	No	Bajo	**2**
Definición de la arquitectura general del sistema	4	40	4	No	Alto	**3.5**
Desarrollo del sistema biométrico	5	40	3	Si	Alto	**3**
Configuración y puesta a punto de nodos en la cadena de bloques	6	30	3	No	Medio	**2.5**
Desarrollo y despliegue de Smart contract	7	40	4	No	Medio	**3**
Integración del sistema biométrico en la Blockchain	8	40	5	No	Alto	**4.5**
Pruebas de uso	9	10	2	Si	Bajo	**1.5**
Diseño y desarrollo de una aplicación en la que integrar el sistema	10	30	3	Si	Medio	**2.5**
Documentación	11	40	2	Si	Bajo	**1.75**

Tabla 6. Estimación de esfuerzo y tamaño mediante Story Points

Como ya se ha explicado anteriormente, el valor del esfuerzo estimado no tiene importancia si se observa de forma independiente, por lo que tras rellenar la tabla anterior se pueden sacar diferentes conclusiones, como que las tareas que más esfuerzo requieren son las que tienen indentificador 4, 5, 7 y 8, mientras que las tareas que menos esfuerzo requieren son las 1, 2 y 9.
Con estos datos se puede realizar una planificación temporal bastante precisa y que permita ajustar al máximo el coste (tanto en tamaño como en esfuerzo) de cada una de las tareas.

1.13 Planificación temporal

La planificación temporal permite representar como ha ido avanzando el proyecto a lo largo del tiempo, teniendo que adaptar esta planificación a las diferentes metodologías elegidas para el desarrollo del mismo. En este caso seguiremos una metodología incremental, por lo que nos tendremos que centrar en conceptos como incrementos y resultados. Para representar las diferentes etapas a lo largo del tiempo, se hará uso del diagrama de Gantt (Ilustración 12), donde se podrá ver la extensión temporal de dicha etapa junto con un color que determinará la dificultad de la misma, que será un vector de colores (Tabla 3), donde se representarán 5 niveles de dificultad, siendo el nivel 1 el más bajo (color amarillo) y el nivel 5 el más alto (color rojo).

Nivel de esfuerzo	Color
1	
2	
3	
4	
5	

Tabla 7. Niveles de esfuerzo representados con colores

La mayoría de las tareas son independientes entre sí, destacando que la tarea de “desarrollo del sistema” requiere que todas las tareas anteriores de elección de plataforma, arquitectura general, comparaciones y estudios estén realizadas para poder comenzar el desarrollo de la tarea principal.

Nombre	Fecha de ini...	Fecha de ...
Estudio y planificación de la necesidad	6/6/22	8/6/22
Elección del rasgo biométrico	9/6/22	13/6/22
Ventajas que aporta el Blockchain al proyecto	14/6/22	17/6/22
Definición de la arquitectura general del sistema	20/6/22	29/6/22
Desarrollo del sistema biométrico	30/6/22	11/7/22
Puesta a punto de nodos en la Blockchain	12/7/22	19/7/22
Desarrollo y despliegue de Smart Contracts	20/7/22	29/7/22
Integración del sistema biométrico en la Blockhain	1/8/22	10/8/22
Pruebas de uso	11/8/22	15/8/22
Desarrollo de una aplicación para integrar el sistema	16/8/22	23/8/22
Documentación	6/6/22	23/8/22

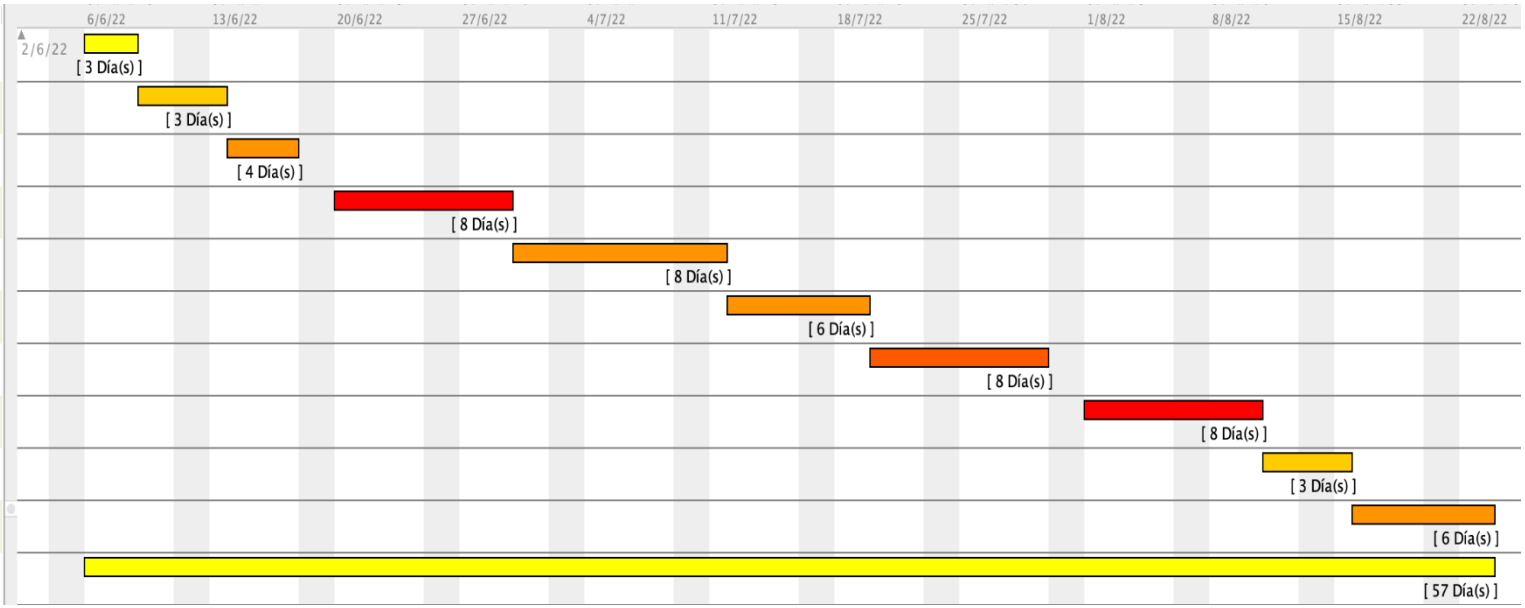


Ilustración 24. Diagrama de Gantt

Como podemos ver en la ilustración anterior se identifican 11 tareas, contando con una que está presente en todas ellas, que es la que hace referencia a la documentación del trabajo, desarrollándose de forma paralela desde el comienzo del proyecto.

Una vez vista la planificación temporal de una forma visual mediante el diagrama de Gantt, se detallará cada tarea, explicando los procesos que la componen, la duración, el nivel de esfuerzo medio y el número de horas de dedicación.
Hay 57 días entre el inicio del proyecto hasta el final, por lo que siendo 300 horas las que hay que dedicar al desarrollo del TFM, la media de horas por día es de -> 300/57 = 5,25 horas al día.

- **Planificación y estudio de la necesidad.** Es el primer paso que se debe dar, realizándose una introducción para que el proyecto tenga un objetivo claro y evitar problemas a posteriori. También se debe hacer una evaluación acerca de las necesidades que pretende superar, analizando diferentes situaciones y proyectos que se están llevando a cabo.
    - Duración: 3 días.
    - Número de horas: 15,75 horas.
    - Nivel de esfuerzo: 1
- **Elección del rasgo biométrico.** Estudio y comparación de diferentes rasgos biométricos, con el fin de elegir el que mejor se adapte a la necesidad del proyecto en cuanto a precisión, seguridad, coste y facilidad de uso.
    - Duración: 3 días.
    - Número de horas: 15,75 horas.
    - Nivel de esfuerzo: 2
- **Informe sobre la necesidad del Blockchain en el proyecto.** Evaluación de las ventajas que aporta el Blockchain en el proceso de identificación y cómo podría cambiar por completo estos sistemas.
    - Duración: 4 días.
    - Número de horas: 21 horas.

  - Nivel de esfuerzo: 3

- **Definición de la arquitectura general del sistema.** Una de las tareas más importantes del proyecto, ya que define por completo el flujo de trabajo que tendrá el sistema final.
  - Duración: 8 días.
  - Número de horas: 42 horas.
  - Nivel del esfuerzo: 5
- **Desarrollo del sistema biométrico.** Etapa que se centra en la programación y obtención de un sistema de identificación biométrica que cuente con las características definidas en la documentación.
  - Duración: 8 días.
  - Número de horas: 42 horas.
  - Nivel de esfuerzo: 3
- **Puesta a punto de nodos en la Blockchain.** Instalación, configuración y puesta a punto de diferentes nodos dentro de la Blockchain de Alastria.
  - Duración: 6 días.
  - Número de horas: 31,5 horas.
  - Nivel de esfuerzo: 3
- **Desarrollo y despliegue de Smart Contracts.** Programación de la lógica y funcionalidades del sistema dentro de la Blockchain.
  - Duración: 8 días.
  - Número de horas: 42 horas.
  - Nivel de esfuerzo: 4
- **Integración del sistema biométrico en la Blockchain.** Etapa crucial en la que se une el sistema biométrico desarrollado con el ecosistema Blockchain configurado.
  - Duración: 8 días.
  - Número de horas: 42 horas.
  - Nivel de esfuerzo: 5

- **Pruebas de uso.** Realización de diferentes pruebas para comprobar que la integración realizada en la etapa anterior no causa problemas.
    - Duración: 3 días.
    - Número de horas: 15,75 horas.
    - Nivel de esfuerzo: 2
- **Desarrollo de una aplicación para integrar el sistema.** Prototipo de aplicación que permita usar el sistema en diversas acciones de identificación diarias de una forma sencilla, rápida y fiable.
    - Duración: 6 días.
    - Número de horas: 31,5 horas.
    - Nivel de esfuerzo: 3
- **Documentación.** Es una tarea que empieza en etapas tempranas del proyecto y que terminará cuando se considere que cuenta con una calidad razonable.
    - Duración: Aproximadamente desde el inicio al fin del proyecto.
    - Nivel de esfuerzo: 1

1.14 Presupuesto

El objetivo de este apartado es el de poder determinar y justificar el coste económico que tiene la elaboración de este trabajo, teniendo siempre presente, que aún siendo un trabajo experimental, tiene un coste asociado, ya que toda actividad tiene un precio. Como ocurre en la mayoría de los proyectos tecnológicos, los recursos asociados a este proyecto son hardware, software, y humanos. Y para poder desglosar el presupuesto en cada uno de ellos de forma detallada, se va a pasar a explicarlos uno a uno.

El coste de cada recurso se va a calcular sin tener en cuenta la estimación por horas, ya que en el apartado “1.14.4 Amortización” será donde se realice esta estimación.

1.14.1 Recursos hardware

El coste hardware de este proyecto procede de dos actividades bien diferenciadas, como son los sensores biométricos utilizados para la identificación y todo lo relacionado con la red Blockchain.

Para la red Blockchain se requieren dos elementos hardware, un ordenador común donde se desarrollarán los Smart contract y una máquina específica donde se encontrará levantado el nodo. Para lo primero se hace uso del ordenador personal. Los principales elementos hardware asociados a esta máquina son los que aparecen en la siguiente tabla:

Coste hardware	
Componente	Coste estimado en €
Intel Core i7 de cuatro núcleos a 2.9 GHz	310
SSD de 512 GB	60
16 GB de memoria LPDDR3	55
Intel HD Graphics 630 1536 MB	300
TOTAL	725

Tabla 8. Coste hardware para el desarrollo de Smart contracts

Hablando de las características necesarias para levantar un nodo en la cadena de bloques, tenemos que por norma común necesitamos cumplir los siguientes requisitos:

Hardware	Mínimo	Deseado
CPUs	2	4
Memoria	4 GB	8 GB
Disco duro	128 GB	256 GB

Tabla 9. Requisitos para levantar un nodo en una red Blockchain [30]

Viendo esta tabla se ha concluido que la mejor opción es contratar una instancia de un proveedor de servicio, como por ejemplo de Amazon Web Services (AWS) y se han valorado diferentes opciones de su catálogo de instancias. En la siguiente tabla

se pueden ver los precios por hora del uso del servicio, destacando que el disco duro que ofrece Amazon es EBS (Elastic Block Store) por lo que el almacenamiento es dinámico y se va reservando el que se usa.

Nº GPUs	Memoria	Disco duro	Precio
2	4GB	EBS	0,101 €
2	8GB	EBS	0,132€
4	8GB	EBS	0,202€

Tabla 10. Precios AWS

La instancia elegida finalmente es la que cuenta con 2 CPUs, 8 GB de RAM, y un disco duro de 128 GB, por lo que se podría concluir que el precio estimado por hora sería de 0,132€. Si la instancia dedicada al nodo va a estar levantada desde la tarea llamada “puesta a punto de nodos en la Blockchain” que comienza el 12 de Julio y finalizará al acabar el proyecto el 23 de Agosto, contamos con un total de 43 días, lo que equivale a un aproximado de 1032 horas, por lo que el coste estimado por el uso de la instancia sería de 1032*0,132= 136,22€.

Una vez definido el coste hardware referente a la red Blockchain, se va a pasar a definir el coste de los sensores. Cabe destacar que dependiendo de la funcionalidad final del sistema hay diversas opciones, ya que si el sistema sirve de identificación para entrar a un edificio se comprarán los dos sensores por separado, mientras que si la identificación se realiza por aplicación móvil, éste ya contará con los dos sensores por lo que este coste sería el valor del móvil.

Hardware	Coste individual	Coste total
Huella dactilar (DEFROBOT SEN0359) Reconocimiento facial 3D (Anviz Facepass)	41€ 294€	335€
Smartphone de gama media con los dos sensores necesarios	200€	200€

Tabla 11. Coste hardware biométrico

Para el presupuesto hardware de este proyecto se va a considerar la opción móvil, por lo que el coste hardware biométrico será de 200€.

Una vez definidos todos los puntos relacionados con el coste hardware se puede hacer un pequeño resumen en el que se detalle el coste de todas las partes y obtener así un presupuesto final para el proyecto (Tabla 12).

Coste Hardware	
Gasto	Coste en €
Desarrollo de Smart contracts	725
Instancia AWS	136,22
Hardware biométrico	200
TOTAL	1061,22

Tabla 12. Coste hardware total

1.14.2 Recursos software

Aquí se incluyen todas las herramientas empleadas en el desarrollo de este trabajo Fin de Máster, tanto las usadas en el desarrollo de la red neuronal, las usadas en el desarrollo de la aplicación móvil, el editor de textos para la documentación, etc.
Se debe señalar, que no todo el software usado cuenta con el mismo tipo de licencia, pudiendo tener licencias con un único pago, pagos anuales, mensuales, o licencias gratuitas.

Coste software		
Producto/Concepto	Tipo de pago	Coste estimado en €
Gantt Project Planificación temporal	No aplicable	0
Microsoft 365 Family Documentación y presentación	Anual	99
macOS Big Sur Sistema Operativo	No aplicable	0
Remix IDE	No aplicable	0

Entorno de desarrollo		
Framework desarrollo multiplataforma Entorno de desarrollo	No aplicable	0
Zotero Herramienta para la bibliografía	No aplicable	0
Socio de Alastria Red Blockchain	-	-
TOTAL		99

Ilustración 25. Coste software

1.14.3 Recursos humanos

Debido a la limitación en personal con la que cuenta este trabajo Fin de Máster, el coste en recursos humanos solo va a incluir el gasto de una única persona. Para ello se estima un salario mensual, que vaya acorde con las competencias desarrolladas y puesto de trabajo que ocupa durante el desarrollo del mismo.

Como bien se define en la planificación temporal, este trabajo se divide en dos partes muy bien diferenciadas, el desarrollo de un sistema de identificación biométrico en Android, y el desarrollo de una solución basada en Blockchain que permita integrar la identificación, por lo que se pueden asignar distintos salarios, dependiendo de qué trabajo esté desempeñando en cada momento el trabajador.

- Desarrollo de la arquitectura Blockchain: El rol seleccionado es el de Blockchain Developer, que cuenta con un salario anual de 29.000€ según el portal Glassdoor (https://www.glassdoor.es/Sueldos/blockchain-developer-sueldo-SRCH_KO0,20.htm).
- Desarrollo del sistema de identificación Android: Cuenta con un salario de 32.000€ al año según el portal Glassdoor (https://www.glassdoor.es/Sueldos/desarrollador-android-sueldo-SRCH_KO0,21.htm).

Por otra parte, se debe añadir el gasto en seguridad social, que supone un 33% del coste total. Todo esto se puede ver resumido en la siguiente tabla:

Recursos humanos			
Concepto	Salario mensual (€)	Meses	Coste estimado (€)
Blockchain Developer	29000/12 = 2416,66	2	2.416,66* 2 = 4.833,32
Android Developer	32000/12 = 2666,66	2	2.666,66 * 2 = 5.333,32
Seguridad social	-	-	(4.833,32 + 5.333,32)* 0,33 = 3.354,99
TOTAL			13.521,63

Ilustración 26. Coste en recursos humanos

1.14.4 Amortización

En este apartado calcularemos la amortización del hardware y del software, tal y como lo define la Agencia tributaria dentro del grupo de bienes “Equipos electrónicos e informáticos. Sistemas y programas”. Aunque se tratará de manera diferente la parte hardware del software, y esto es debido a que el hardware, con el transcurso de los años, pierde valor, e incluso puede llegar a quedar obsoleto, mientras que el software no cuenta con estos inconvenientes, ya que muchas de las licencias te incluyen las nuevas actualizaciones de forma gratuita.

En el coste hardware no se incluye el presupuesto definido para contratar la instancia de AWS que se usa para mantener un nodo levantado en la red Blockchain, ya que es un servicio que se subcontrata y por consiguiente ese hardware lo amortizará la empresa propietaria.

Para poder calcular la amortización, hay que definir los siguientes conceptos:

- Valor inicial: Precio de los bienes a la hora de comprarlos.

- Valor residual: Precio de los bienes una vez pasada su vida útil. Como este valor se desconoce, se considerará que el software valdrá un 15% del valor inicial y el hardware un 30%.
- Amortización anual: Es la pérdida de valor de un activo a lo largo del tiempo.
- Vida útil: Es el máximo número de años de amortización de un bien. En este caso viene definido por la Agencia Tributaria, y es de 6 años para el hardware y de 10 años para el software.

Cabe destacar que este proyecto no tiene una duración de un año, sino que, siguiendo la planificación temporal, tiene una duración de 4 meses, por lo que la amortización debe adaptarse a ese periodo, tal y como vemos en la siguiente tabla.

Amortización					
Concepto	Valor inicial (Vi)	Valor residual (Vr)	Vida útil (Vu)	Amortización anual(Aa) (Vi-Vr)/Vu	Amortización 4 meses (Aa/12)*4
Hardware	925€	277,5€	6 años	107,92€	35,97€
Software	99€	14,85€	10 años	8,42€	2,81€
TOTAL				116,34€	38,78€

Ilustración 27. Amortización de hardware y software

1.14.5 Coste total

Tras desglosar el coste de cada uno de los recursos identificados en el trabajo, y obtener la amortización de hardware y software, se puede calcular el coste total, permitiendo aportar una cifra final aproximada del coste del trabajo. Hay que señalar, que aparte de los recursos hardware, software y humanos desglosados anteriormente, existen unos costes indirectos que también deben estar presentes en esta sumatoria final, y corresponden a gasto de agua, luz, uso de instalaciones, internet, etc. Como es una cifra que depende de muchos parámetros, se estima incluirlo como un gasto equivalente al 5% del total (hardware + software + recursos humanos).

Por último, como sucede en la mayoría de los presupuestos de este tipo de proyectos, se debe estimar un porcentaje de beneficios, que será un dinero reservado para

imprevistos, adquisición, mejora o reparación de equipos y que permita el crecimiento de la empresa (mejorando instalaciones, contratando más personal, etc), estimando este beneficio en un 15% del total acumulado.

Hay que señalar que el coste final de hardware será la sumatoria entre el coste de la subcontratación del servicio AWS para la instancia donde se encuentra nodo de la red Blockchain, y la amortización hardware calculada en el apartado anterior, al igual que para el software, cuyo coste final se centrará en dicha amortización.

Coste total	
Concepto	Coste estimado (€)
Coste hardware	136,22+35,97 = 172,19
Coste software	2,81
Coste de recursos humanos	13.521,63
Costes indirectos 5%	13.696,63*0,05 = 684,83
Beneficios 15%	14.381,46*0,15 = 2.157,21
TOTAL	16.538,67

Tabla 13. Coste total

El coste final estimado es de 16.538,67€, aunque podría disminuirse optimizando el gasto de recursos humanos si no estuvieran las limitaciones temporales y de personal.

1.15 Normas y referencias

A continuación, se presentan las normas, reglamentos y referencias de cualquier tipo que han sido de aplicación en la elaboración del trabajo y/o en la puesta en práctica del mismo.

1.15.1 Disposiciones legales y normas aplicadas

Esta aplicación tiene la posibilidad de obtener información sobre diferentes rasgos biométricos de los usuarios, y al considerarse datos de carácter personal, debe cumplir con la Ley Orgánica de Protección de Datos de Carácter Personal (LOPD).

Para cumplir con la protección de datos en aplicaciones móviles que hagan uso de rasgos biométricos, la LOPD exige que se cumplan los siguientes puntos [31]:

- Se debe conocer la persona que accede a dichos datos.
- Se debe conocer los datos a los que se ha accedido.
- Informar sobre los derechos ARCO, con los cuáles el usuario tiene derecho de acceso, rectificación, cancelación y oposición de la información almacenada sobre su persona.
- Fecha y hora exactas en la que se hizo el acceso mediante la bioidentificación.

1.15.2 Mecanismos de control de calidad

El aseguramiento de la calidad en la redacción del trabajo implica la verificación de la completitud (falta de omisiones), la integridad de la documentación, así como una redacción clara, concisa y entendible por todos los participantes e interesados en dicho trabajo.

Al estar el trabajo desarrollado por una sola persona y verificado por un/a tutor/a, no es necesario llevar un documento de control de edición y revisión de la documentación. De esta forma, se han utilizado mecanismos básicos para la verificación de la integridad y completitud, que incluyen un control de versiones a nivel de documento y un control sencillo de la trazabilidad de especificaciones y requisitos.

2 DISEÑO INICIAL

Esta sección tiene como principal misión especificar aspectos como la arquitectura del sistema y los modelos de diseño referentes a la funcionalidad, datos e interfaz. Siendo el apartado que permite asentar las ideas y construir una base sobre la que posteriormente se desarrollará el software, ayudando a obtener un desarrollo eficiente mediante el uso de la metodología incremental, la cual acelerará en gran medida el propio desarrollo gracias a los aspectos que se definan en este apartado.
Se realizará un análisis y diseño general, es decir, englobando todas las iteraciones especificadas en la planificación temporal, mientras que en el apartado 3 se realizará una explicación exhaustiva del desarrollo.

2.1 Requisitos iniciales

Este apartado se centra en explicar qué debe hacer el sistema, indicando los requisitos principales a alto nivel, sin entrar a detallar cómo lograr dichos requisitos, que será explicado en el apartado 2.2 (Especificaciones del sistema). Existen dos tipos de requisitos:

- **Requisitos funcionales:** Son aquellos requisitos que definen una funcionalidad del sistema.
- **Requisitos no funcionales:** Son los requisitos que imponen distintas restricciones en el diseño, ya que describen propiedades o cualidades que debe tener el producto.

2.1.1 Requisitos funcionales

Cabe destacar que a la hora de abordar el listado de requisitos funcionales, se contará con dos expresiones claves, como son “debe” (Para identificar un requisito principal) y “debería” (para formular una sugerencia). Este listado es:

- El usuario debe poder identificarse en cualquier momento, es decir, contar con una disponibilidad del servicio 24/7.
- El usuario debe poder registrarse en el sistema si no posee cuenta previa.

- Dependiendo del tipo de aplicación el usuario debe superar la identificación biométrica propuesta (reconocimiento facial – Huella dactilar) para poder acceder al servicio en cuestión.
- El sistema debe registrar en la cadena de bloques qué usuario se ha registrado y en qué servicio.
- El sistema debe guardar los datos biométricos encriptados y distribuidos.
- El sistema debe ser compatible con los principales sistemas operativos móviles (Android e IOS).
- El usuario debería poder actualizar sus datos biométricos en caso de cualquier tipo de accidente.

2.1.2 Requisitos no funcionales

Se espera que el resultado de este proyecto cuente con características propias de un sistema de identificación, ofreciendo garantías tanto a organizaciones que incorporen este servicio como a usuarios que lo utilicen. Para ello se definen una serie de requisitos no funcionales que serán los encargados de conducir y modelar el diseño y el desarrollo del sistema, acotando las propiedades y cualidades que debe tener.

- El hardware biométrico debe superar unos controles de calidad que midan tanto la falsa aceptación (FAR) como el falso rechazo (FRR), consiguiendo con ello un sistema biométrico eficaz y seguro.
- Interfaz típica de un proceso de identificación, fácil de interpretar y que permita al usuario un dominio completo al usarlo por primera vez.
- El Blockchain debe contar con nodos validadores conocidos y de confianza, que puedan verificar transacciones y evitar diferentes tipos de fraudes.
- El servicio puede adaptarse a cualquier tipo de aplicación que requiera identificación, sobre todo aquellos que requieran de una seguridad mayor.
- El servicio debe proporcionar tiempos de respuesta aceptables.

2.2 Casos de uso

En este apartado se explicarán los diferentes casos de uso del proyecto, que no es más que la descripción detallada de cada una de las acciones del usuario.

Hay que tener en cuenta que cada uno de los casos de uso utiliza los rasgos biométricos seleccionados, por lo que se deben señalar las diferentes relaciones de inclusión (<<include>>). Por otra parte, hay que destacar que en cada uno de estos casos de uso se utiliza la tecnología Blockchain, ya sea para el registro (guardar los datos biométricos), identificación (consultar los datos), o modificación (guardar datos nuevos y revocar antiguos).

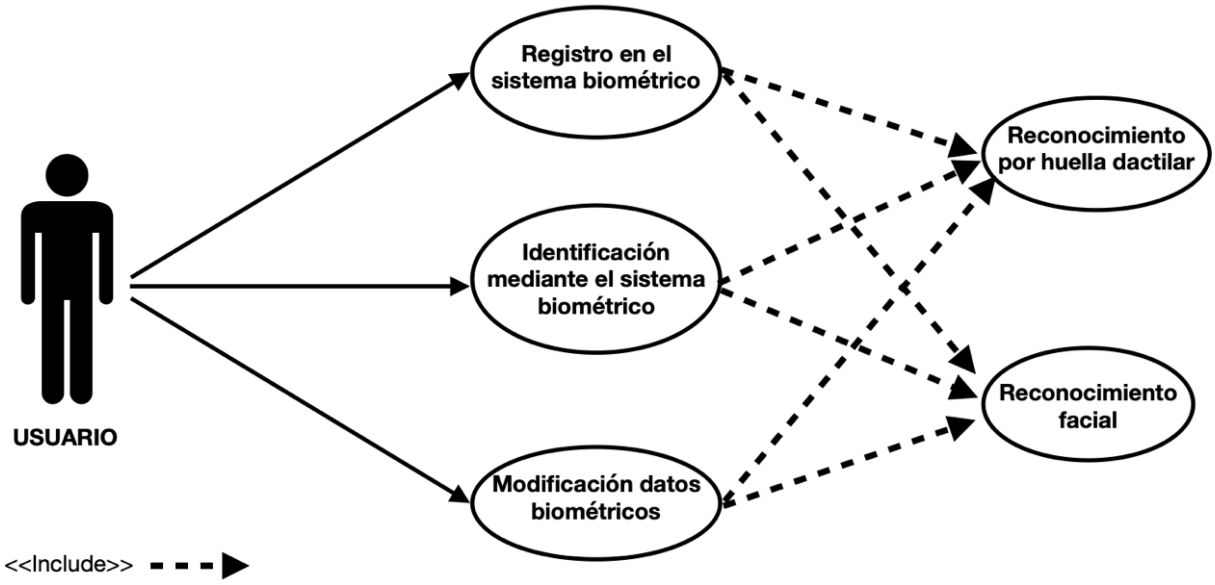


Ilustración 28. Casos de uso

Una vez mostrado el esquema general de los casos de uso, toca detallarlos individualmente. Para ello se seguirá el Marco de Desarrollo de la Junta de Andalucía (MADEJA) [32], el cual cuenta con diversos campos a rellenar que permiten desglosar de forma muy detallada cada uno de estos casos de uso.

CU-01	**Registro en el sistema biométrico**	
Versión	1.0 (01/07/2022)	
Dependencias	Dispositivo móvil con lector de huella dactilar y/o cámara frontal	
Precondición	El usuario no está registrado en el sistema de identificación	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el usuario desee usar la funcionalidad de registro biométrico	
Secuencia normal	**Paso**	**Acción**
	1	El usuario abre la aplicación y aparece la pantalla de identificación. Como el usuario no se encuentra identificado biométricamente en el sistema, pulsa sobre el botón de registro.
	2	El sistema le muestra una nueva pantalla, en la que debe rellenar el formulario pertinente con sus datos personales.
	3	Empieza el proceso de registro biométrico, en el que dependiendo de la app debe realizar el reconocimiento facial mediante la cámara frontal del móvil y/o registro dactilar con el lector de huellas.
	4	El sistema informa al usuario que el registro se ha realizado con éxito.
Postcondición	Ninguna	
Excepciones	**Paso**	**Acción**
	3	Si el usuario no sabe realizar correctamente el registro biométrico
		E.1 El sistema informará mediante un mensaje de texto cómo realizar el proceso.
		E.2 Si el usuario sigue fallando en la acción, se pasará a ofrecerle un vídeo explicativo de cómo debe realizarse correctamente el registro.
		E.3 Tras E.1 y E.2 se regresará a la pantalla principal.

Ilustración 29. Caso de uso: Registro en el sistema biométrico

CU-02	Identificación mediante el sistema biométrico	
Versión	1.0 (01/07/2022)	
Dependencias	Dispositivo móvil con lector de huella dactilar y/o cámara frontal	
Precondición	El usuario debe estar registrado en el sistema de identificación	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el usuario desee usar la funcionalidad de identificación mediante el sistema biométrico.	
Secuencia normal	**Paso**	**Acción**
	1	El usuario abre la aplicación y aparece la pantalla de identificación. Como se encuentra identificado biométricamente en el sistema, pulsa sobre el botón de inicio de sesión.
	2	El sistema inicia el proceso de identificación, comenzando con el proceso de reconocimiento facial, y dependiendo de la aplicación sigue con el escaneo de la huella dactilar.
	3	Tras identificar correctamente al usuario termina el servicio y se accede de forma directa a la aplicación correspondiente.
Postcondición	Ninguna	
Excepciones	**Paso**	**Acción**
	2	Si el usuario no realiza correctamente el escaneo biométrico
		E.1 El sistema informará mediante un mensaje de texto cómo realizar el proceso.
		E.2 Si el usuario sigue fallando en la acción, se pasará a ofrecerle un vídeo explicativo de cómo debe realizarse correctamente el registro.
		E.3 Tras E.1 y E.2 se regresará a la pantalla principal.

Ilustración 30. Caso de uso: Identificación mediante el sistema biométrico

CU-03	Modificación datos biométricos	
Versión	1.0 (01/07/2022)	
Dependencias	Dispositivo móvil con lector de huella dactilar y/o cámara frontal	
Precondición	El usuario debe estar registrado en el sistema de identificación y debe haber tenido un cambio significativo en alguno de los rasgos de identificación.	
Descripción	El sistema deberá comportarse como se describe en el siguiente caso de uso cuando el usuario desee usar la funcionalidad de modificación de datos biométricos.	
Secuencia normal	**Paso**	**Acción**
	1	El usuario abre la aplicación y aparece la pantalla de identificación. Se encuentra identificado, pero el sistema no le reconoce por haber experimentado un cambio en el rasgo. Pulsa sobre el botón de modificar datos biométricos.
	2	El sistema inicia el proceso de modificación, le hace identificarse con el rasgo inalterado (en apps con doble reconocimiento biométrico) o enviando un mensaje de verificación al correo personal.
	3	En aplicaciones de seguridad máxima también se envía un mensaje de verificación al correo electrónico del usuario.
	4	Tras pasar las tareas 2 y 3 con éxito, el sistema pasa a escanear nuevamente el rasgo afectado.
Postcondición	Los datos biométricos del rasgo afectado que se encontraban guardados antes de la modificación pasan a ser inválidos.	
Excepciones	**Paso**	**Acción**
	2 y 4	El usuario no realiza correctamente el escaneo biométrico
		E.1 El sistema informará mediante un mensaje de texto cómo realizar el proceso.
		E.2 Si el usuario sigue fallando en la acción, se pasará a ofrecerle un vídeo explicativo de cómo debe realizarse correctamente el registro.
		E.3 Tras E.1 y E.2 se regresará a la pantalla principal.

Ilustración 31. Caso de uso: Modificación datos biométricos

2.3 Análisis y diseño del sistema

2.3.1 Análisis

Una vez definidos los casos de uso, se pueden sacar diferentes conclusiones e interpretaciones acerca de cómo debe ser el sistema.
Ahora toca especificar las características que tiene, siguiendo los requisitos iniciales y casos de uso explicados en apartados anteriores.

- Este sistema de identificación que sirve como pasarela a distintas aplicaciones, estará disponible para Android (>v5.0) e IOS (>IOS10).
- Es necesaria la integración del sistema de identificación biométrica con una red Blockchain, que permita almacenar de forma distribuida los datos de cada usuario.
- El almacenamiento biométrico en la red Blockchain debe cumplir con el RGPD, por lo que se debe definir un plan de encriptación y compartición de estos datos.
- Para el correcto funcionamiento del sistema, el dispositivo móvil debe tener conexión a internet en el momento de la identificación.

Una vez realizado el análisis acerca de los detalles que supone cumplir los diferentes casos de uso, sería bueno destacar cuáles pueden ser las mejoras que se le pueden aplicar a las diferentes funcionalidades explicadas anteriormente. En cuanto la variedad de sistemas operativos y dispositivos, la mejora pasaría por optimizar el sistema para que también sirviera como pasarela de identificación en aplicaciones web y de escritorio. Para ello, antes debe producirse un movimiento por parte de los fabricantes de ordenadores (que ya se está produciendo), en el que se incorpore hardware específico, como lector de huellas, o webcam de calidad con la que poder realizar el reconocimiento facial.

Esta mejora sería bastante interesante, ya que desde el ordenador se producen multitud de accesos a aplicaciones por segundo, y podría ayudar a simplificar y centralizar este proceso, garantizando la total seguridad a aquellas empresas que lo implementen para los accesos a sus aplicaciones.

2.3.2 Diseño y arquitectura del sistema

Es uno de los apartados más importantes del proyecto, ya que se centra en explicar detalladamente cómo será la arquitectura del sistema, cómo funcionarán todos los actores/partes del proyecto y cómo se distribuirán los datos biométricos por la red Blockchain. Tras esto, se mostrará un prototipo de la interfaz del sistema.

2.3.2.1 Arquitectura del sistema y funcionamiento

La arquitectura general del sistema cuenta con 3 actores bien definidos. Por una parte, se encuentra el usuario, por otra el dispositivo móvil (que tiene los sensores biométricos necesarios), y por último la red Blockchain. Las diferentes comunicaciones se pueden ver en la siguiente imagen.

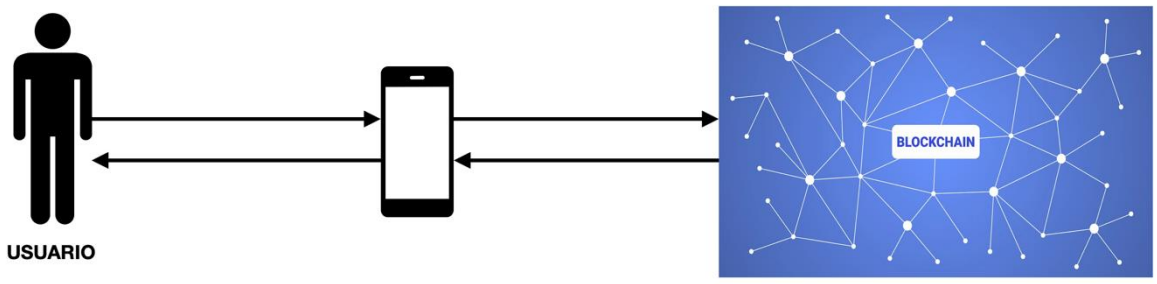


Ilustración 32. Arquitectura general del sistema

Se va a desarrollar un supuesto que ayudará a explicar el funcionamiento del sistema: El usuario quiere acceder a una aplicación, la cual utiliza el proceso de identificación desarrollado en este proyecto.

- Paso 1: El usuario accede a la aplicación y se encuentra con la pantalla de inicio de sesión. Pulsa sobre “Iniciar sesión”.
- Paso 2: Comienza el proceso de identificación biométrica, el dispositivo móvil solicita al usuario que escanee en el sensor el rasgo biométrico.
- Paso 3: El sistema tiene que comparar los datos biométricos introducidos con los almacenado en la red Blockchain, para ello recupera los datos almacenados y compara las dos muestras.

- Paso 4: Si el sistema determina que el usuario es quién dice ser, le da acceso a la aplicación, en caso contrario le vuelve a pedir que repita el proceso de identificación.

Todo este proceso se detallará y profundizará más en el apartado 3 del documento, en el momento en el que se pase a desarrollar toda esta funcionalidad. Aparte de esta, el proyecto cuenta con dos más, como son la de registro de rasgo biométrico (necesaria para almacenar por primera vez los datos y que posteriormente pueda realizar el usuario el proceso de inicio de sesión), y la funcionalidad de actualizar rasgos biométricos (necesaria para solventar cualquier problema que haya tenido el usuario en el rasgo biométrico específico para la identificación).

2.3.2.2 Distribución de datos biométricos

En los apartados anteriores se ha hablado de que la información biométrica de cada usuario se almacena en la red Blockchain, y para ello hay que definir el proceso de almacenamiento teniendo en cuenta que hay que cumplir la RGPD.

En este caso, la información va a estar almacenada de forma distribuida, es decir, que la totalidad de los datos no los va a almacenar una sola máquina, sino que una máquina solo contará con parte ellos. Con esto nos aseguramos que un atacante nunca pueda disponer de los datos biométricos de un usuario, ya que como mucho podría obtener alguna de las partes, pero al estar encriptados y ser independientes de las demás, no le aportaría ningún tipo de información. Por ello, se debe buscar un método que cumpla con lo siguiente:

- Cumplimiento de la RGPD.
- Datos encriptados.
- Datos distribuidos.
- Fácil recuperación para consultas.
- Imposible recuperación por parte de un atacante.
- Minimizar el problema de pérdida/borrado de datos por ataques externos.

Para realizar este proceso se va a seguir la teoría del secreto compartido de Shamir, el cuál cumple perfectamente con todos los requisitos expuestos.

Secreto compartido

Se puede definir como la distribución segura de información privada de forma segmentada dentro de una red distribuida, lo que hace que sea un método idóneo para salvaguardar información sensible como claves privadas o datos biométricos [33].

Secreto compartido de Shamir – Esquema

Es una implementación del concepto de secreto compartido, propuesta en 1979 por el criptógrafo Adi Shamir, permitiendo dividir la información en muchos fragmentos y necesitando solo parte de ellos para reconstruir el mensaje original. El número mínimo de fragmentos necesarios para reconstruir el secreto se conoce como “Umbral” [33].

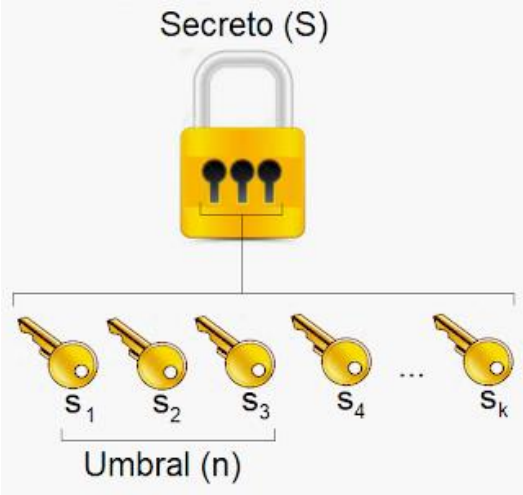


Ilustración 33. Secreto compartido de Shamir

Aplicación del secreto compartido de Shamir para mantener la privacidad de los datos biométricos

Con este algoritmo se puede distribuir información de forma segura, eficiente y privada. La idea principal de usarlo en este sistema es la de cifrar los datos biométricos de un usuario, dividir en fragmentos y distribuir aleatoriamente por diferentes nodos de la red.

Al no ser un proceso centralizado, los hackers no pueden actuar, por lo que obtenemos un sistema de muy alta seguridad, ya que, si un atacante quisiera obtener la información biométrica de un usuario, no bastaría con robar un fragmento, sino que debería robar tantos fragmentos como marque el umbral, por lo que sería prácticamente inviable. Aparte de todo esto, si por un casual obtuviera ese número mínimo de fragmentos, los tendría cifrados, por lo que de todas formas sería imposible que el atacante obtuviera esa información en claro, ya que no dispone de la clave privada.

Para recuperar los datos biométricos de un usuario (por ejemplo, para compararlos con una solicitud de inicio de sesión), el sistema recuperaría el número de fragmentos que marque el umbral, y tras recuperarlos, recompondría los datos biométricos en claro y los compararía con los que se acaban de introducir en el sistema para ver si el usuario que quiere acceder es quién dice ser.

El desarrollo de este proceso se detallará en la tarea correspondiente del apartado 3, que será donde realmente se implemente en el sistema.

2.3.2.3 Diseño de la interfaz y storyboard

Como el resultado del proyecto es la realización de una pasarela de identificación que permita integrarse en cualquier servicio/aplicación, la idea es sacar un módulo en el que se encuentren las conexiones con la red Blockchain y toda la lógica necesaria para el proceso de identificación biométrica. Por ello, lo ideal es sacar esta pasarela con la característica de que pueda integrarse con los estilos de la interfaz general del sistema en el que se aplica.

Teniendo esto en cuenta, el diseño de la interfaz que se va a mostrar a continuación no tiene por qué ser el que finalmente se vea en producción, pero ayudará en gran medida a mostrar los diferentes botones que se van a encontrar en la pasarela y los diferentes pasos que hay que seguir tanto para el registro de un nuevo usuario como para la identificación de uno ya registrado. Eso sí, lo más característico de la interfaz de la pasarela será el logo propio de este proyecto, llamado Blockchain Biometric Identity (BBI).

Ilustración 34. Logo de Blockchain Biometric Identity

Pantalla principal

Es la pantalla inicial de la pasarela, que aparece cuando se abre la aplicación que tiene integrado este sistema. Aquí se puede introducir el nombre de usuario de un usuario ya registrado, o poder registrarse en este sistema de identificación.

Ilustración 35. Pantalla principal

Acceso de un usuario ya registrado

Cuando en la pantalla principal se pone un nombre de usuario que se encuentra registrado, se pasa a la siguiente pantalla, en la cual se solicita el escaneo del rasgo biométrico por parte del usuario, a través del cual se accede al servicio en cuestión si la identificación se realiza de forma correcta, y vuelve a pedir el escaneo en caso contrario.

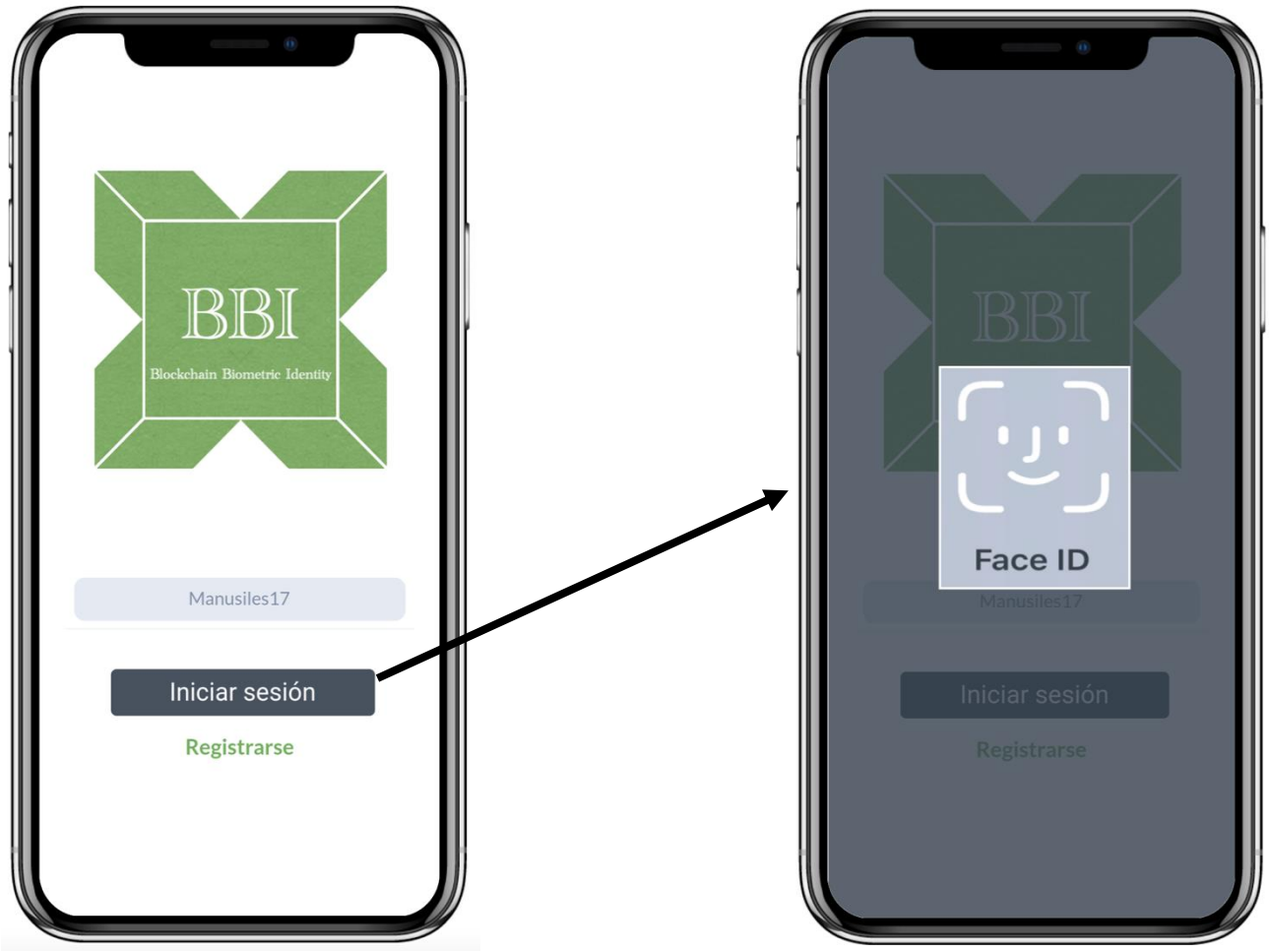


Ilustración 36. Acceso de un usuario ya registrado

Registro de un nuevo usuario

A esta pantalla se accede cuando se pulsa sobre el botón de “Registro de un nuevo usuario” dentro de la pantalla principal. Al acceder hay que rellenar diferentes campos relacionados con los datos personales del usuario, como nombre de usuario, nombre y apellidos, correo y teléfono. Tras ello, se pasa a una nueva pantalla en la cual el

sistema pide el escaneo del rasgo biométrico para poder almacenarlo en la red Blockchain para posteriores identificaciones.

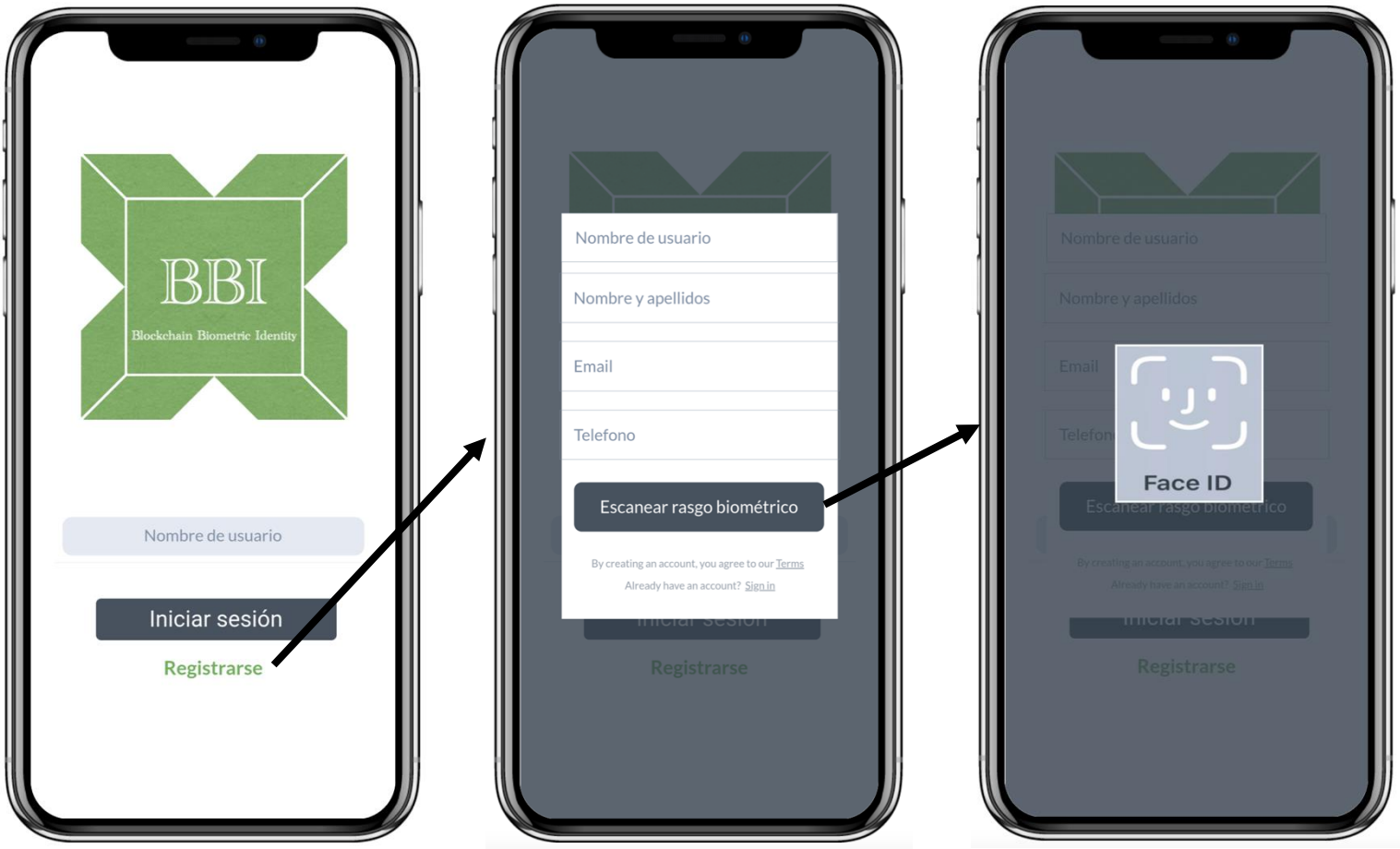


Ilustración 37. Registro de un nuevo usuario

Estas pantallas se han realizado en MarvelApp (marvelapp.com), que es una aplicación que permite definir fácilmente storyboard e interfaces, teniendo en el siguiente enlace la interfaz que se acaba de presentar: https://marvelapp.com/prototype/958d218

3 DESARROLLO

Este apartado se divide en iteraciones, siguiendo la metodología incremental que se ha descrito anteriormente (sección 1.10). Cada una de estas tareas viene definida en la planificación temporal del proyecto, detallada en el primer apartado. Dentro de cada una de las iteraciones se explica cómo se ha conseguido lograr el objetivo que perseguía y los entregables finales resultantes, mostrándose los resultados que se han ido obteniendo en el desarrollo del proyecto y así poder evaluar finalmente la progresión hacia el resultado final.

Para tener una visión global de cada tarea, ver dónde está explicada y con ello demostrar que se ha realizado con éxito, se va a especificar en la siguiente tabla la relación de cada tarea con el apartado correspondiente del documento.

Tarea	Apartado
Estudio y planificación de la necesidad	1.1, 1.2 y 1.4
Elección del rasgo biométrico	1.7
Ventajas que aporta Blockchain al proyecto	1.1, 1.2, 1.4 y 1.8
Definición de la arquitectura general del sistema	2.1, 2.2 y 2.3
Desarrollo del sistema biométrico	**3.1**
Puesta a punto de nodos en la Blockchain	**3.2**
Desarrollo y despliegue de Smart Contracts	**3.3**
Integración del sistema biométrico en la Blockchain	**3.4**
Pruebas de uso	**3.5**

Tabla 14. Distribución de las tareas en el documento

Con esta tabla-resumen se puede observar que en este apartado de desarrollo se encuentra el grueso del proyecto, tratándose de aquellas tareas que requieren de una implementación a nivel de código y que por consiguiente son más prácticas que las desarrolladas en apartados anteriores.

3.1 Primera Iteración: Desarrollo del sistema biométrico

3.1.1 Tratamiento de los datos biométricos

Antes de pasar al desarrollo del sistema toca definir cómo se van a tratar los datos biométricos, desde el momento en el que se escanean hasta que son guardados en la red Blockchain. Por ello, este apartado se centrará en definir este tratamiento siguiendo el orden natural de los acontecimientos del sistema.

Registro de un usuario

El sistema le pide al usuario que escanee el rasgo biométrico necesario para poder registrarse, ya que ese rasgo biométrico será el que posteriormente se use en el proceso de identificación.

Tras el escaneo, se obtiene un conjunto de datos en el que se encuentran las características principales del rasgo escaneado y que permiten identificarlo con respecto a otros usuarios. La primera solución que se viene a la mente es la de guardar directamente en claro esas características biométricas del usuario, pero ocuparía mucho espacio, sería ineficiente, lento, y además incumpliría el RGPD, por ello, se ha optado por simplificar el proceso.

Una vez obtenido el resultado del escaneo, donde tendríamos en crudo las características biométricas del usuario, se procedería a usar esas características para generar un hash (hash biométrico), con el que conseguiríamos seguir identificando unívocamente a un usuario, pero optimizando el proceso y teniendo solamente que almacenar ese hash en la red Blockchain.

Hay que destacar que existen multitud de funciones hash, eligiendo en este caso la función SHA-256, ya que es muy usada y cuenta con un alto nivel de seguridad, por lo que nos permite proteger y codificar de forma segura la información.

Para darle aún más seguridad al hash biométrico, se ha decidido que ese valor no será el que se almacene en la red, sino un hash resultante de aplicarle SHA-256 al hash biométrico, consiguiendo con eso que los atacantes que intenten obtener un

hash biométrico no lo puedan conseguir, ya que como mucho conseguirían un hash derivado del biométrico, y apoyándonos en las características de las funciones hash nunca podría recuperar el hash biométrico.

Todo el proceso que se ha explicado viene resumido en la siguiente ilustración:

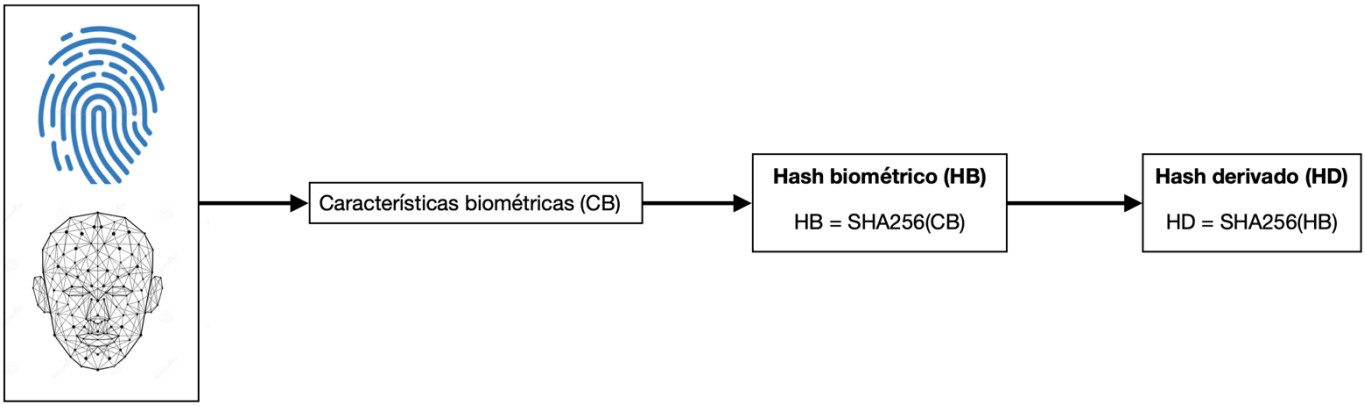


Ilustración 38. Tratamiento datos biométricos - Registro

Identificación de un usuario registrado

Sabiendo cómo se ha almacenado la información en el registro, en la identificación sólo hay que seguir el mismo proceso para obtener hashes equivalentes.

En este proceso el sistema le pide al usuario que introduzca su nombre de usuario (username), y comprueba si realmente se encuentra registrado, en caso afirmativo lanza la pantalla de escaneo biométrico, y en caso de no estar registrado se lo comunica para que inicie el proceso de registro.

En caso de que el usuario esté registrado, escanea el rasgo biométrico con el sensor, y directamente el sistema calcula su hash biométrico, que será el que se pase a la comparación junto con el nombre de usuario.

Una vez estos dos datos entran en la función de identificación, el hash biométrico se transforma en el hash derivado pasándolo por el SHA-256 y se busca el hash del usuario que ya estaba almacenado en el sistema. Si estos dos hashes coinciden la

identificación es correcta y se le permite al usuario acceder al servicio/aplicación, en caso contrario se vuelve a pedir que escanee el rasgo.

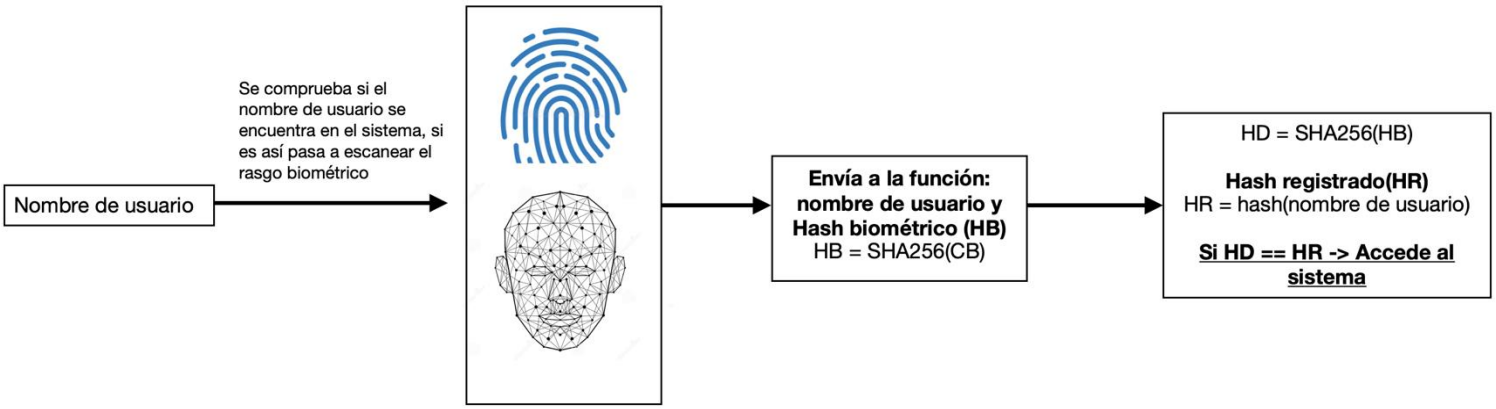


Ilustración 39. Tratamiento datos biométricos - Identificación

3.1.2 Revisión de frameworks multiplataforma

Esta comparación es un paso necesario antes de empezar con el desarrollo del sistema biométrico, ya que determinará cuál será la herramienta usada para el desarrollo multiplataforma. Hay que señalar que existen en el mercado multitud de ellas, pero para esta comparación se han escogido las 3 más usadas: React native, Xamarin y Flutter.

La idea es realizar una comparación amplia, que abarque diferentes aspectos importantes, como son datos técnicos, características generales, y tendencias.
Se va a empezar comparando las tendencias de cada uno en los últimos años.

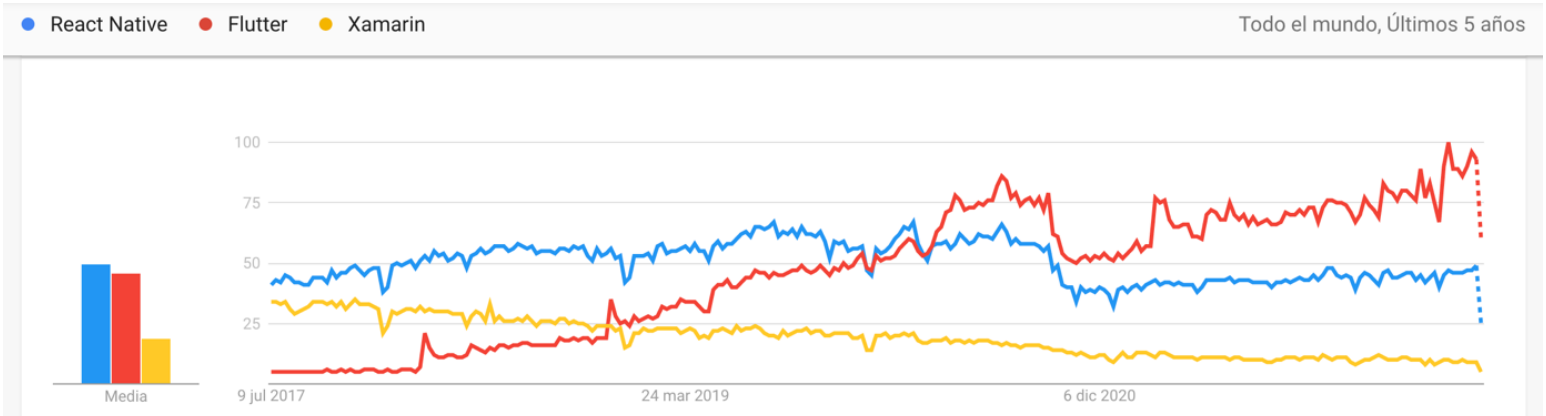


Ilustración 40. Tendencia en tecnologías de desarrollo móvil de los últimos años

Por el 2017, React y Xamarin eran los más populares, contando con unos números muy parecidos. Desde ahí la tendencia de Xamarin ha sido decreciente, ya que siempre ha ido bajando en popularidad y uso. Por otra parte, React Native ha mantenido sus números en el tiempo, viendo como comenzó con un breve crecimiento, que en tiempos de pandemia se convirtió en una acusada recesión, aunque hay que señalar que desde entonces se ha estabilizado su popularidad. Y para acabar hay que destacar el increíble crecimiento de Flutter en estos últimos 5 años, empezando como un framework casi desconocido a ser actualmente el más popular de los 3.

Tras esta comparación se va a pasar a mostrar una tabla-resumen con las características principales de cada uno de los frameworks.

	React Native	Xamarin.forms	Flutter
Código abierto	Si	Si	Si
Licencia	MIT	MIT	BSD
Código compartido	JavaScript	C#	Dart
Paradigma	Declarativo	Imperativo	Declarativo
OEM Widgets	Si	Si	No
Recarga en vivo	Si	No	Si
Gestor de paquetes	NPM, Yarn	NuGet	Pub
Portabilidad	Si	No	No
Acceso a cámara	Si	Plugin	Plugin

Tabla 15. Comparación características generales [34]

Y para finalizar se van a comparar algunos aspectos técnicos que repercuten de manera directa en el rendimiento del dispositivo donde se instale la aplicación. Como:

- Tiempo de inicio.
- SLOC multiplataforma: Métrica que mide la eficiencia de un producto software.
- Almacenamiento.

	React Native	Xamarin	Flutter
Tiempo de inicio	283 ms	296 ms	2060 ms
SLOC multiplataforma	711	877	951
Almacenamiento	45.16 MB	77.98 MB	50.41 MB

Tabla 16. Comparación datos técnicos [34]

Se puede ver como React Native arrasa en estas métricas tan importantes, ya que es el que menor tiempo de arranque tiene, menor número de líneas de código fuente y encima es el que menos almacenamiento requiere de los 3 frameworks.

Viendo todas las comparaciones y teniendo siempre presente la finalidad de la aplicación a desarrollar, se ha decidido que el framework usado para este proyecto sea **React Native**, ya que aparte de ser el mejor en los datos técnicos comparados, también hace uso de JavaScript (lenguaje familiar) y cuenta con muchas facilidades para el acceso a hardware (lector biométrico).

3.1.3 Arquitectura y funcionamiento de React Native

En cuanto a la arquitectura, sigue el modelo Flux, que es similar al modelo vista controlador (MVC), pero cuenta con un flujo de datos unidireccional. La idea es que los datos viajen desde la vista por medio de acciones y lleguen al dispositivo de almacenamiento (Store) donde se actualizará de nuevo la vista.

Hay que destacar que la Store solo guarda los datos y estado de la aplicación, no contando con métodos propios para modificar lo almacenado, que se realiza a través de acciones y dispatchers [35].

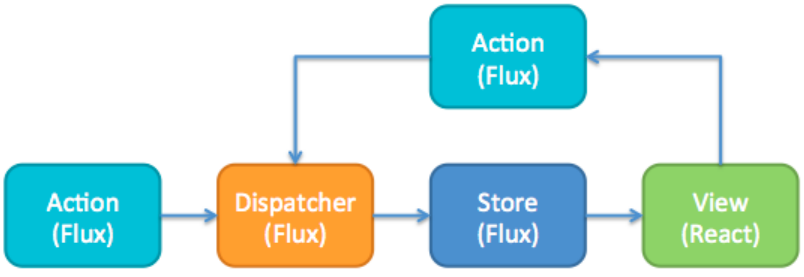


Ilustración 41. Arquitectura Flux [35]

En cuanto al funcionamiento, todo gira entorno a los componentes, que no son más que pequeños elementos autónomos que se pueden reutilizar en diversas pantallas, con la idea de separar el código en piezas lógicas.

Lo que sucede básicamente en React Native es que el compilador convierte los componentes (en formato JSX), en elementos nativos de la interfaz para Android e IOS. Con ello, las aplicaciones desarrolladas tienen un aspecto muy similar a las nativas, un rendimiento y una experiencia de navegación y usuario casi idéntico [36].

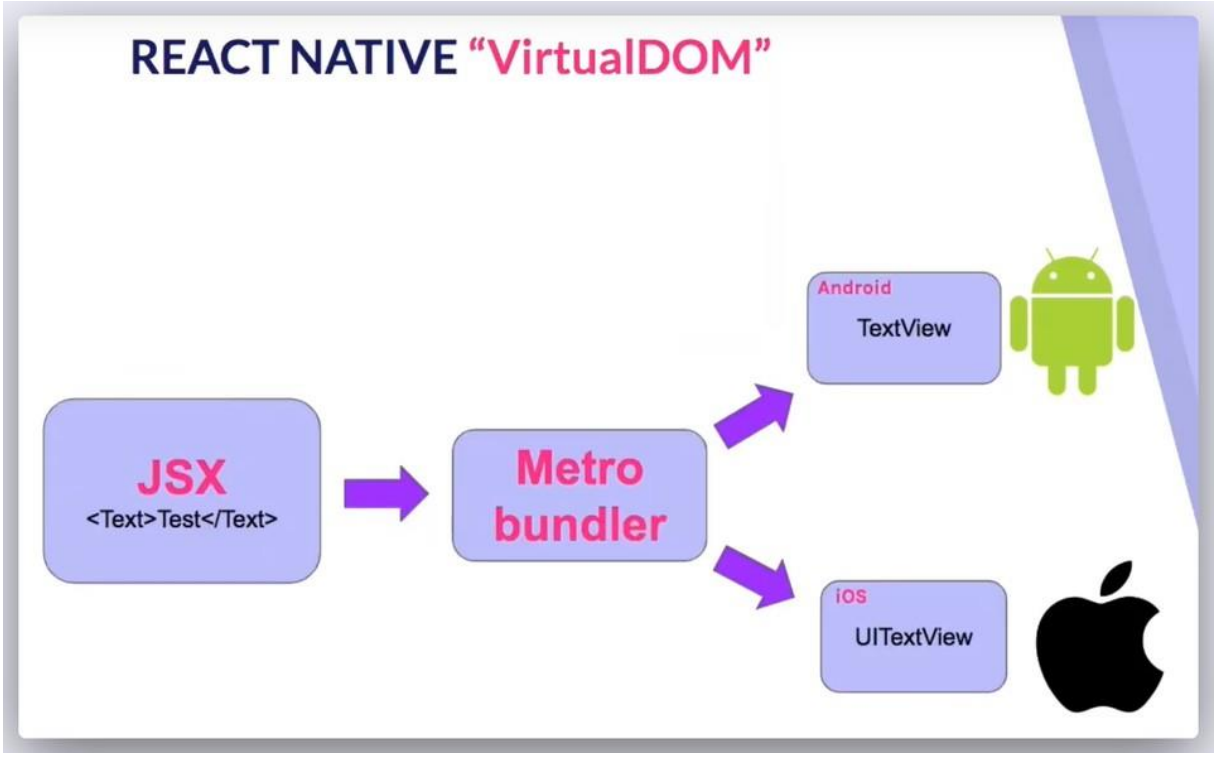


Ilustración 42. Funcionamiento de React Native [36]

Por otra parte, toca destacar que cuando se ejecuta la aplicación, React Native crea dos hilos en los que tenemos lo siguiente:

- Hilo dedicado al código nativo de Android e IOS (interfaz nativa y librerías nativas integradas).
- Hilo que ejecuta JavaScript mediante una máquina virtual, destacando que dicha ejecución se produce de forma nativa, sin compilación ni transpilación.

Por ello, nace el puente (Bridge), para permitir la comunicación entre ambos hilos y así permitir que JavaScript esté conectado con la parte nativa [36].

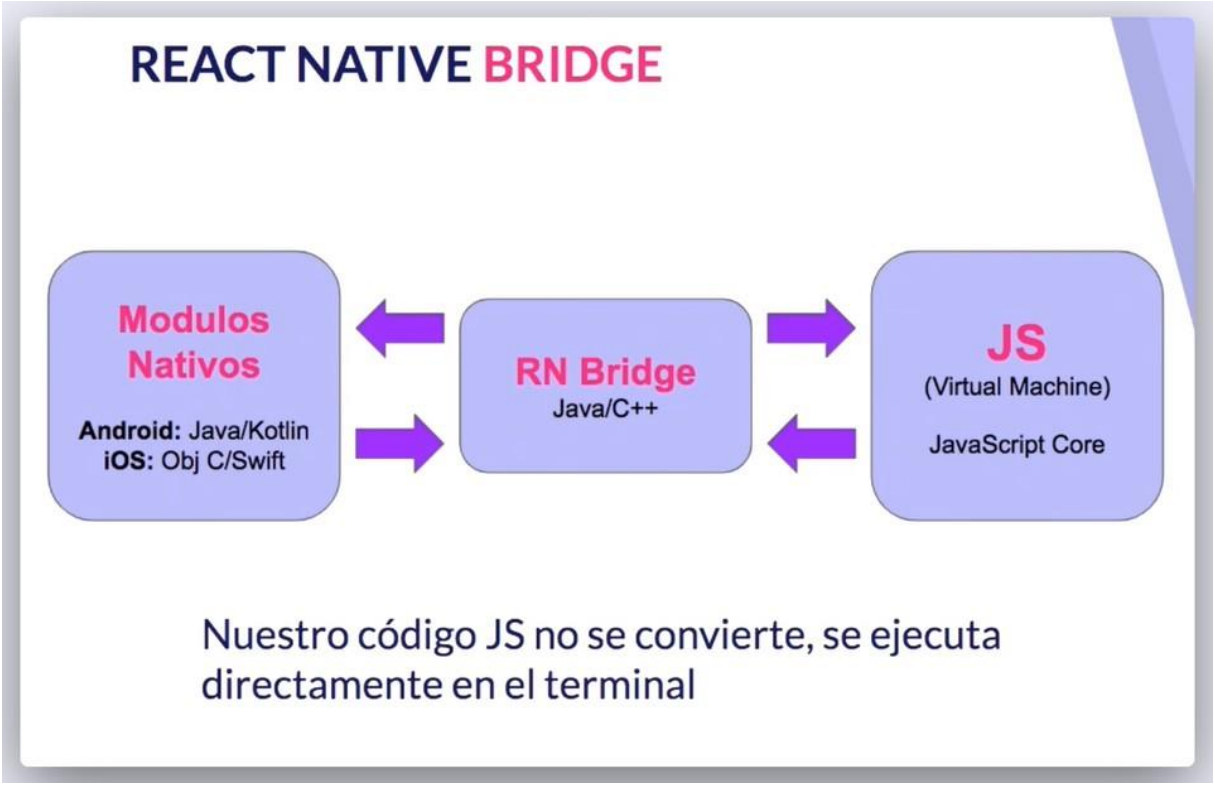


Ilustración 43. Puente entre hilos [36]

3.1.4 Dificultades encontradas y soluciones

Tras definir cuál va a ser el tratamiento de los datos biométricos, se ha pasado a la fase de buscar APIs y funciones con las cuáles poder obtener un resultado biométrico, y poder así tratar esa información según los esquemas explicados en el apartado anterior.

Hay que decir que la búsqueda se ha hecho tanto en documentación nativa de Android e IOS, como en React Native, obteniendo lo siguiente:

- Tanto el sistema nativo de Android como el de IOS cuentan con módulos seguros donde almacenan los valores biométricos. En Android, por ejemplo, esta información se encuentra en el sistema operativo Trusty, que proporciona un entorno de ejecución de confianza (TEE -Trusted Execution Environment-), que es un SO seguro que corre a la par del SO normal, y que por ninguno de los medios es accesible. Este cuenta con un pequeño kernel, un controlador del kernel de

Linux (trasfiere datos entre el entorno seguro y Android) y una biblioteca del espacio de usuario para comunicarse con las tareas/servicios seguros [37].

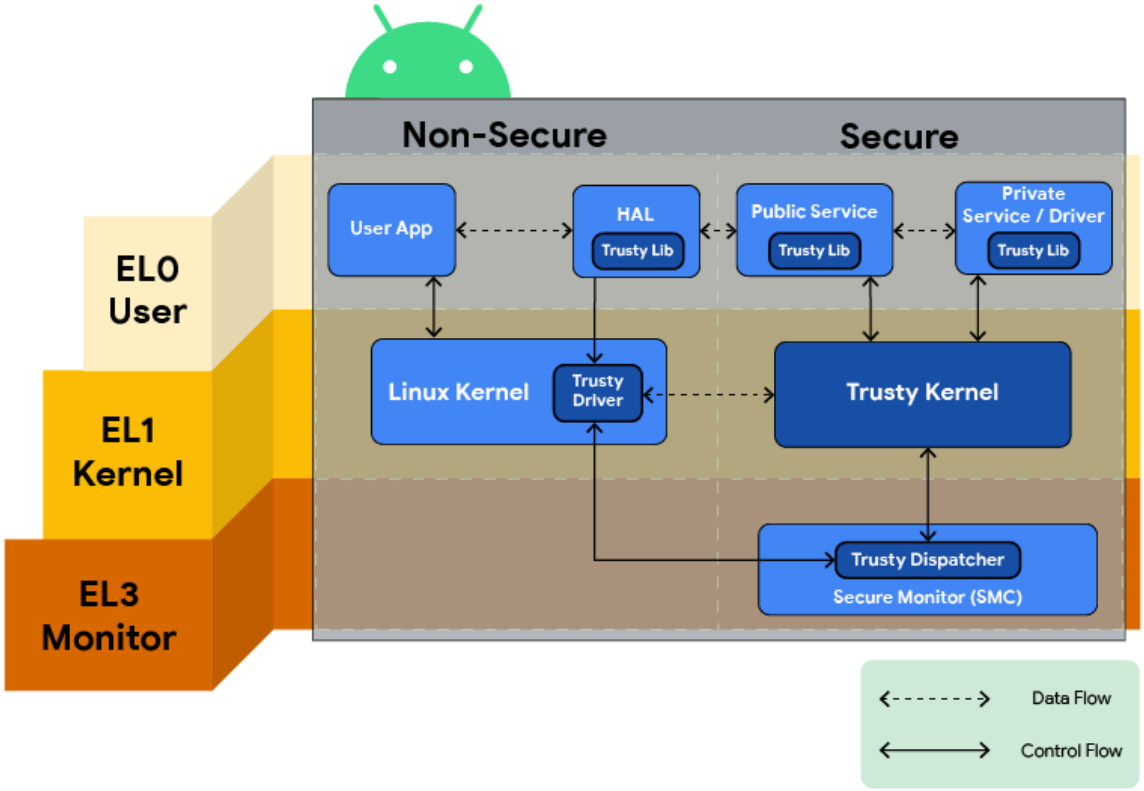


Ilustración 44. Diagrame de descripción general confiable [37]

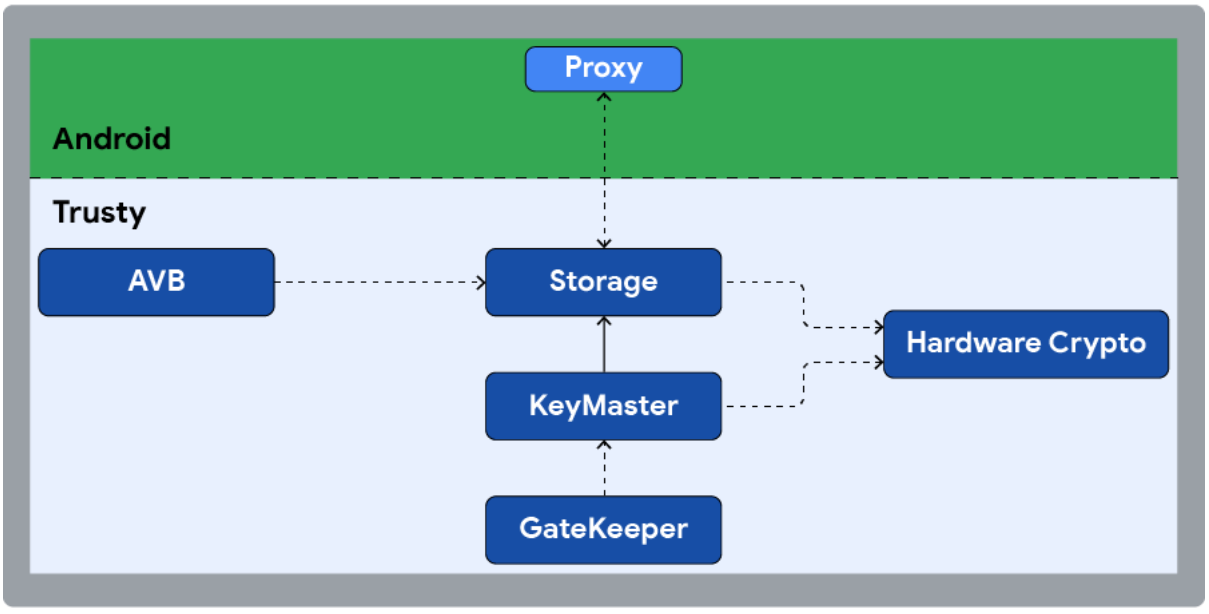


Ilustración 45. Descripción general de Trusty [37]

- Por otra parte, indagando en APIs y funcionalidades compatibles con React Native, hay que decir que existen multitud de posibilidades y desarrollos a la hora de usar los datos biométricos, pero todas ellas ofrecen lo mismo, que es la posibilidad de saber si el hardware soporta el escaneo biométrico, y otra funcionalidad que devuelve verdadero o falso tras el escaneo para saber si el usuario es quién dice ser. Algunas de las APIs que ofrecen esto son “React-native-fingerprint-scanner” [38] y “React-native-bytte-bio-lib” [39].

Viendo lo comentado, se puede concluir que a día de hoy los sistemas móviles no ofrecen la posibilidad de devolver los datos biométricos escaneados, que es precisamente lo que se necesita en este trabajo. Por ello, se va a explicar la solución que se va a llevar a cabo, comentando sobre todo el por qué y las ventajas que puede tener para el futuro.

La solución

Desarrollar una aplicación móvil con React Native que simule el comportamiento detallado en apartados anteriores, con la diferencia de que una vez el usuario se identifique biométricamente, se le asigne un hash a través de su nombre de usuario, simulando con ello que ese es el hash biométrico obtenido del escaneo.

¿Por qué desarrollarla?

- Una aplicación práctica permite ver más fácil el comportamiento del sistema.
- A día de hoy puede implementarse sin problema en sistemas biométricos que cuenten con sensores externos (entradas a edificios, ficha de trabajadores, accesos restringidos, etc), ya que estos sensores sí ofrecen la información biométrica necesaria.
- A medio/largo plazo se puede esperar que los SO/Arquitecturas de los smartphones permitan tratar los datos biométricos, siguiendo la idea de que el usuario pueda acceder a todos los servicios a través de sus rasgos. Por ello, el desarrollo de este sistema podría ser de provecho en años venideros.

3.1.5 Desarrollo del sistema

Tras detallar todo lo necesario, comienza el proceso de desarrollo del sistema. Siguiendo el storyboard (definido en el apartado 2), se puede dividir el desarrollo en dos pantallas principales, como son la pantalla principal y la pantalla de registro.

Para obtener una explicación detallada, primero se va a comentar cómo se ha estructurado el proyecto dentro de React Native, después se pasará a explicar las funciones más importantes del código y por último se mostrará el resultado final del sistema.

Estructura del proyecto

En la siguiente ilustración se puede observar la estructura general del proyecto, teniendo marcado en rojo aquellos directorios y archivos que se han creado o modificado de la estructura principal de un proyecto React Native vacío.

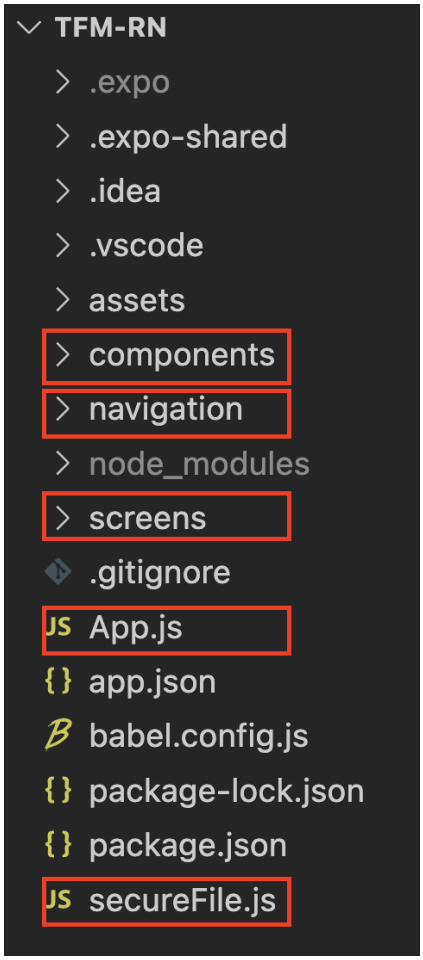


Ilustración 46. Estructura general del proyecto

- **Components:** Directorio en el que se encuentran los diferentes componentes usados y reutilizados en el código, como botones o entradas de texto. Esta característica que nos ofrece React Native hace que se obtenga un código más limpio y fácil de entender, ya que la definición de estilos y configuraciones de componentes se hacen fuera del código principal.

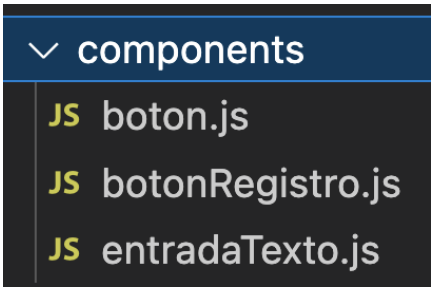


Ilustración 47. Directorio Components

- **Navigation:** Siguiendo las buenas prácticas de desarrollo en React Native, hay que crear un directorio de navegación, donde se encuentren las diferentes pilas de navegación de la app. En este caso, al ser una aplicación sencilla solo basta con definir la pila de navegación principal.

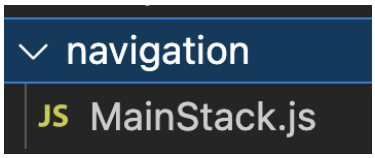


Ilustración 48. Directorio Navigation

- **Screens:** Directorio donde se encuentran las diferentes pantallas de la pasarela. En este caso son 3, la pantalla principal, la de registro, y la que simula el acceso a la aplicación tras pasar la identificación correctamente.

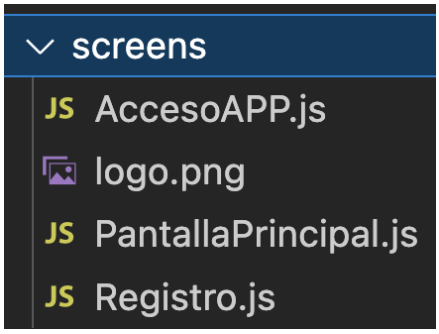


Ilustración 49. Directorio Screens

- **App.js:** Es el archivo JavaScript predeterminado que se genera al crear el proyecto, y el cuál se inicia al ejecutarlo.
- **secureFile.js:** Archivo JavaScript que genera todo el proceso de identificación, tanto el diálogo de escaneo como la comparación con los datos biométricos ya almacenados en el dispositivo.

Funciones más importantes del código

- Función que permite pasar de unas pantallas a otras, definiendo la estructura de la pila.

```
const MainStack = () => {
   return(
      <NavigationContainer>
         <Stack.Navigator>
            <Stack.Screen
               name = "PantallaPrincipal"
               component = { PantallaPrincipal }
            />

            <Stack.Screen
               name = "Registro"
               component = { Registro }
            />

            <Stack.Screen
               name = "AccesoAPP"
               component = { AccesoAPP }
            />
         </Stack.Navigator>
      </NavigationContainer>
   )
}
```

- Función que detecta si el dispositivo es compatible con la identificación biométrica.

```
useEffect(() => {
        (async () => {
        const compatible = await
LocalAuthentication.hasHardwareAsync();
        setBiometrics(compatible);
        })();
    }, []);
```

- Funcionalidad que realiza todo el proceso de identificación biométrica, desde el escaneo hasta la comparación con la huella almacenada.

```
const [grantAccess, setGrantAccess] = useState(false);

  useEffect(() => {
    (async () => {
      const auth = await
LocalAuthentication.authenticateAsync();
      if (auth.success) setGrantAccess(true);
      else setGrantAccess(false);
    })();
  }, []);
```

Por otra parte, en cuanto a la aparición de notas informativas debidas a problemas en el correcto funcionamiento de la pasarela, tenemos:

- Mensaje informativo si el dispositivo no soporta el escaneo biométrico → Se deshabilitan los botones de “Iniciar sesión” y “Registro”.

```
<Boton
    onPress = { () => setRenderContent(true)}
    disabled={!biometrics}
    text={"Iniciar sesión"}>
</Boton>

{renderContent && <SecureFile user={user}
navigation={navigation} access={true} />}

<BotonRegistro
    onPress = { () =>
    {navigation.navigate("Registro")}}
    disabled={!biometrics}
/>

<Text style={styles.TextError}>
    {!biometrics
        ? "Este dispositivo no es compatible con el escaneo
        biométrico"
    : ""}
</Text>
```

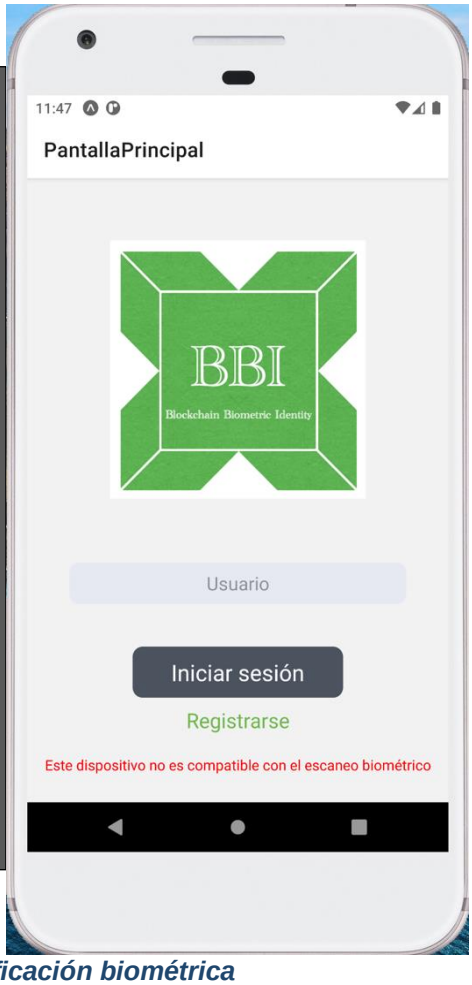


Ilustración 50. El dispositivo no soporta identificación biométrica

- Obligación en el registro de rellenar todos los campos que aparecen antes de pasar al escaneo biométrico.

```
<Boton
        onPress = { () => setRenderContent(true)}
        text={"Escaneo"}>
</Boton>

{renderContent &&
        user != null &&
        name != null &&
        email != null &&
        phone != null &&
        <SecureFile
                navigation={navigation} access={false} />
}

{renderContent &&
        (user == null ||
        name == null ||
        email === null ||
        phone === null) && (
        <Text style={styles.TextError}>
                Debes completar todos los campos
        </Text>
)}
```

Ilustración 51. Mensaje de error cuando no se completan todos los campos

3.2 Segunda iteración: Puesta a punto de nodos en la Blockchain

Para almacenar los datos biométricos en una red Blockchain, lo primero de todo es encontrar una red real que se adapte a nuestras necesidades y que cuente con las características deseadas. Como se ha comentado en el apartado 1 de este documento, se ha optado por el uso de una red Blockchain permisionada, por lo que para poder acceder a ella se necesita superar un proceso de validación, para que den permiso al nodo que se va a levantar.

En este caso se va a trabajar con Alastria (consorcio multisectorial), que cuenta con dos redes permisionadas operativas, la red T y la red B, de las cuáles se debe elegir una de ellas en la que se levantarán los nodos de este proyecto. Para elegir entre una u otra se va a realizar una pequeña comparación:

Red Quorum (T)	Red Besu (B)
Mejora la privacidad de los contratos y transacciones	Implementa la especificación EAA
Mejor desempeño	Trabaja con diferentes protocolos de consenso
Código abierto	Monitorización sencilla de los nodos
Impulsado por la comunidad	Menor número de transacciones
Madura	Menor número de nodos
Mayor número de nodos	Ocupa menos espacio en disco
Mayor número de transacciones	Menos fallos de red

Tabla 17. Diferencias entre la red T y red B de Alastria

Como se puede apreciar en la tabla, se han combinado características propias de las tecnologías Quorum y Besu con características propias de la red T y B de Alastria. Viendo las dos opciones se ha decidido elegir la red B, basada en Hyperledger Besu, ya que, al ser más nueva, tener menos nodos, ocupar menos espacio en disco y contar con menor cantidad de fallos, vemos que se adapta muy bien al problema que plantea este proyecto.

El siguiente paso es acceder a la documentación que proporciona Alastria de la red B en GitHub (https://github.com/alastria/alastria-node-besu) y seguir los pasos para levantar un nodo. Esta instalación consta de 2 pasos principales, que son:

- **Proceso de instalación del nodo**
  - Descargamos el repositorio de GitHub:

    sudo git clone https://github.com/alastria/alastria-node-besu.git
  - Dentro del directorio "Docker-compose" editamos el fichero de entorno ".env", modificando la variable "NODE_NAME" por el nombre que le vamos a poner a nuestro nodo. En este caso, el nodo se llamará "REG_ProteccionDigital_B_2_8_00", ya que la empresa que se hace cargo del nodo es ProteciónDigital, se va a levantar en la red B, cuenta con dos procesadores y con 8 GB de RAM.
  - Se levanta el contenedor ejecutando: docker-compose up -d
  - Para mantener el control de las claves se crea el directorio "keys" en la ruta /data/Alastria-node-besu/. Dentro del directorio creado se almacena la clave privada, la clave pública y el nodeAddress.
  - Apagamos el contenedor y lo volvemos a levantar para que se apliquen los cambios realizados (dlcoker-compose down y Docker-compose up -d).

- **Permisionado del nuevo nodo**
  - Hay que rellenar un formulario en el que detallar los parámetros importantes del nodo (IP, enode, detalles del sistema, etc).
  - La espera máxima que hay en la red para la aceptación de un nuevo nodo es de 1 hora, tras ello, el nuevo nodo empezará a tener conexiones con otros nodos (peers) y empezará a sincronizar la cadena.

Una vez cumplimentados estos 2 pasos, toca abrir una serie de puertos que permitan poder recibir y enviar información y ver métricas del nodo. Estos puertos son:

Puerto	Tipo	Desde	Definición
30303	TCP	0.0.0.0	Transferencia de datos del cliente Ethereum
30303	UDP	0.0.0.0	Listener y descubridor de puertos del cliente Ethereum
9545	TCP	185.180.8.152	Métricas desde el servicio externo de Prometheus

Tabla 18. Puertos abiertos

Tras completar los 2 pasos previos, ya se tendría el nodo levantado en la red, y para ver si es así accedemos a la interfaz que tiene Alastria (https://alastria-netstats2.planisys.net:8443/login) en la que se pueden ver las métricas de sus nodos.

Time	System	Chain Height	Target Chain Height	Blocks Behind	Total Difficulty	Block Time (5m avg	Time Since Last Blo	Peer Count	% Peer Limit Used
2022-07-01 10:52:28....	REG_CDM_4_4_00:95...	10.347.049	10.347.049	0	1.03e+7	1.00 s	1 second, 500 millise...	5	20.00%
2022-07-01 10:52:28....	REG_Grant_Thornton:...	10.347.047	10.347.047	0	1.03e+7	1.00 s	3 seconds, 500 millis...	4	16.00%
2022-07-01 10:52:28....	REG_Clavei:9545	10.347.047	10.347.047	0	1.03e+7	1.00 s	3 seconds, 500 millis...	4	16.00%
2022-07-01 10:52:28....	REG_SM-SERVICES_...	10.347.046	10.347.046	0	1.03e+7	1.00 s	4 seconds, 500 millis...	4	16.00%
2022-07-01 10:52:28....	REG_Telefonica:9545	10.347.045	10.347.045	0	1.03e+7	1.00 s	5 seconds, 500 millis...	4	16.00%
2022-07-01 10:52:28....	REG_FIDESOL:9545	10.347.045	10.347.045	0	1.03e+7	1.00 s	5 seconds, 500 millis...	4	16.00%
2022-07-01 10:52:28....	REG_TruBlo_B_2_4_0...	10.347.043	10.347.043	0	1.03e+7	1.00 s	7 seconds, 500 millis...	5	20.00%
2022-07-01 10:52:28....	REG_ProteccionDigit...	10.347.042	10.347.042	0	1.03e+7	1.00 s	8 seconds, 500 millis...	4	16.00%
2022-07-01 10:52:28....	REG_Asvin_004_016_...	10.347.042	10.347.042	0	1.03e+7	1.00 s	8 seconds, 500 millis...	5	20.00%
2022-07-01 10:52:28....	REG_Asvin_005_016_...	10.347.041	10.347.041	0	1.03e+7	1.00 s	9 seconds, 500 millis...	4	16.00%
2022-07-01 10:52:28....	REG_eprocessmed_B...	10.347.040	10.347.040	0	1.03e+7	1.00 s	10 seconds, 500 milli...	4	16.00%
2022-07-01 10:52:28....	REG_AlastriaID:9545	10.347.039	10.347.039	0	1.03e+7	1.00 s	11 seconds, 500 milli...	5	20.00%
2022-07-01 10:52:28....	REG_Asvin_001_011_...	10.347.037	10.347.037	0	1.03e+7	1.00 s	13 seconds, 500 milli...	5	20.00%
2022-07-01 10:52:28....	REG_Metrovacesa_0...	10.347.036	10.347.036	0	1.03e+7	1.00 s	14 seconds, 500 milli...	4	16.00%

Podemos ver como se encuentra sincronizado por completo, ya que el último bloque sincronizado se corresponde con el bloque más alto de la cadena. Otro de los aspectos a tener en cuenta en estas métricas son los Peers (puntos de conexión con otros nodos), y como el nodo que se ha levantado es “regular”, solo debería tener conexión con los nodos “validadores” de la red.

Una vez realizado todo el proceso de instalación y de verificación del nodo, se puede dar por finalizada esta iteración y pasar a la siguiente, donde se desarrollarán los Smart Contracts necesarios y se hará uso de este nuevo nodo para desplegarlos por la red Blockchain.

3.3 Tercera iteración: Desarrollo y despliegue del smart contract

Una vez disponible el nodo dentro de la red Blockchain, toca desarrollar las diferentes funcionalidades que permitan facilitar la tarea de identificación. Todas ellas se han desarrollado dentro del Smart contract llamado “biometricID” y estarán disponibles cuando se suba a la Blockchain.

Antes de nada, hay que definir las variables necesarias para controlar correctamente todas las funcionalidades, estas son:

- Usuario: Es la estructura encargada de almacenar todos los datos personales.

- listaRegistrados: Estructura encargada del control de usuarios registrados, en la que por cada hash de usuario devuelve “true” si está registrado y “false” si no lo está.

- Usuarios: Estructura que relaciona un hash biométrico con un usuario.

```
struct usuario{
        string username;
        string nombre;
        string correo;
        string telefono;
        bytes32 biometricHASH;
}

mapping(bytes32 => bool) listaRegistrados;
mapping(bytes32 => usuario) usuarios;
```

Una vez definidas las estructuras de datos que se van a usar, se puede pasar a detallar las diferentes funcionalidades desarrolladas. Estas son:

- addUsuario: Función que se llama cuando un usuario se quiere registrar en el servicio, para ello se le pasa el nombre de usuario, nombre y apellidos, email, teléfono y el hash biométrico resultante del escaneo. Una vez almacenados estos datos, se incorpora a la lista de registrados y se le asignan todos sus datos personales a un hash derivado del hash biométrico.

- getUsuario: Función que permite comprobar de una manera rápida y óptima si un usuario que intenta acceder al sistema se encuentra registrado o no.

- comparacionBiometrica: Función que compara una huella registrada con un intento de acceso, para comprobar si el usuario es quién dice ser. Si el resultado de la comparación es correcto se permite acceder, en caso contrario se pedirá la repetición del escaneo.

```
function addUsuario(string calldata _username, string calldata _nombre, string calldata
_correo, string calldata _telefono, string calldata _biometricHASH) public{

        bytes32 hashUser = sha256(bytes(_username));

        usuario memory newUser = usuario({
                username: _username,
                nombre: _nombre,
                correo: _correo,
                telefono: _telefono,
                biometricHASH: sha256(bytes(_biometricHASH))
            });

        listaRegistrados[hashUser] = true;
        usuarios[hashUser] = newUser;
}



function getUsuario(string calldata _username) public view returns (bool){
        bytes32 hashUser = sha256(bytes(_username));
        return listaRegistrados[hashUser];
}



function comparacionBiometrica(string calldata _username, string calldata _biometricHASH)
public view returns (bool){
        bytes32 hashUser = sha256(bytes(_username));
        bytes32 hashBiometric = sha256(bytes(_biometricHASH));

        return (hashBiometric == usuarios[hashUser].biometricHASH);
    }
}
```

3.3.1 Despliegue del Smart contract

Para desplegar un Smart contract dentro de la red de Alastria existen multitud de métodos, los principales son: Hardhat (hardhat.org), Truffle (trufflesuite.com), Remix (remix.ethereum.org), que son herramientas e IDEs para el desarrollo en Ethereum. Cada uno de ellos ofrece diversas ventajas y normalmente el desarrollador se queda con aquel que tiene más manejo.

En este caso se ha optado por la opción de Remix, ayudándonos para ello en Metamask (Aplicación tipo billetera que permite interactuar con una cuenta Ethereum). Los pasos que se han seguido son los siguientes:

Paso 1: Añadir la red B a Metamask.

Ilustración 52. Añadir la red B a Metamask

Paso 2: Desplegar el contrato en Remix. Para ello se debe seleccionar “Injected Provider - Metamask”, y a continuación confirmar la conexión a través de una notificación que aparece automáticamente. Tras esto, se pulsa sobre el botón de “Deploy” y se confirma la transacción.

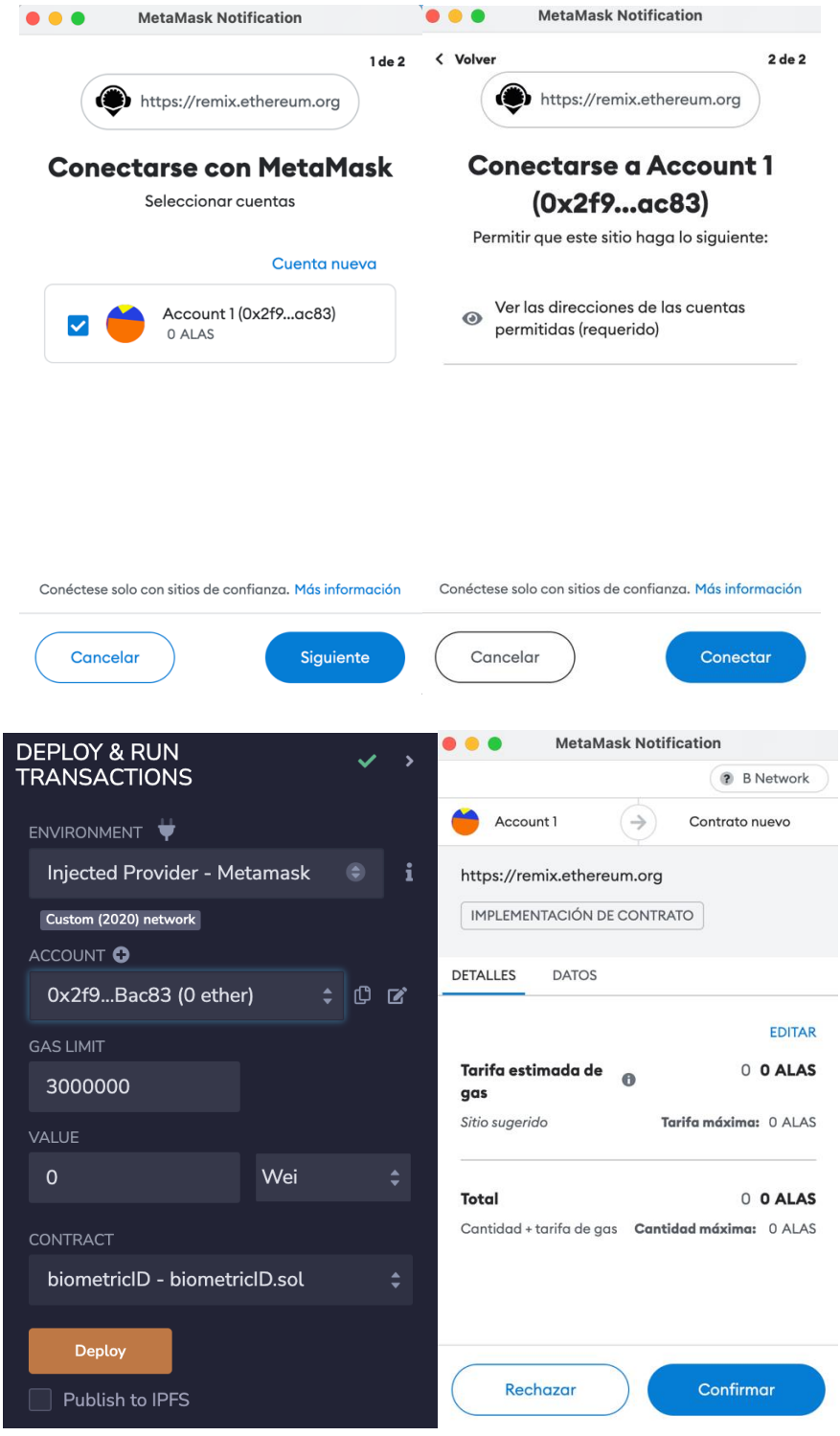


Ilustración 53. Despliegue del smart contract

Una vez hechos estos dos pasos ya se ha desplegado el contrato en la Blockchain, y se puede ver información acerca de la transacción en el terminal de Remix.

Ilustración 54. Información acerca de la transacción

Otra de las ventajas que ofrece el despliegue en Remix, es la facilidad para probar el contrato, ya que te permite hacerlo de una forma intuitiva a través del propio IDE.

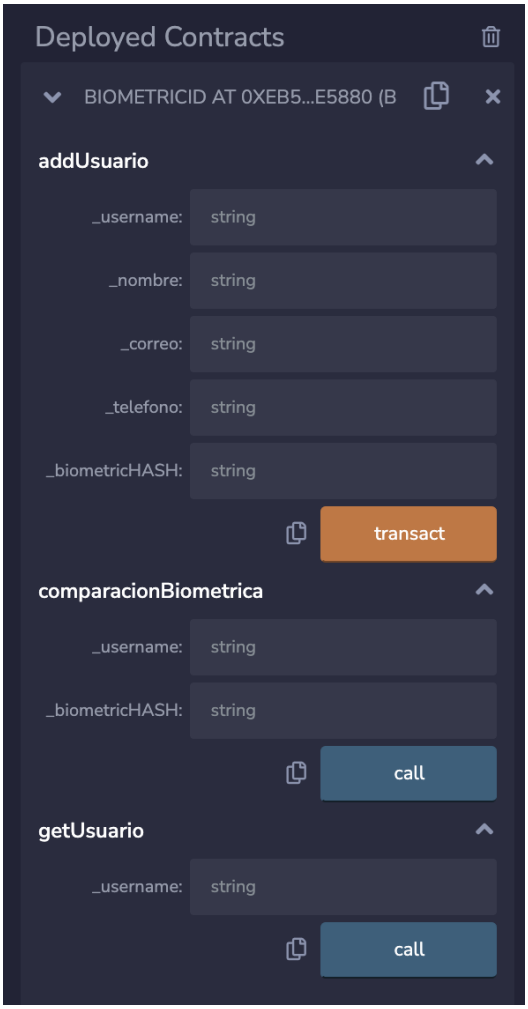


Ilustración 55. Apartado de pruebas del contrato

3.3.2 Explorador de bloques

Es una de las herramientas más útiles que pueden existir dentro de una red Blockchain, y es debido a la gran cantidad de información que nos proporciona acerca de la red. No sólo datos generales, sino datos específicos de cada bloque, de cada transacción e incluso de los contratos que se despliegan.

Lo ideal es levantar un explorador de bloques local, que sólo monitorice las acciones del proyecto, como por ejemplo BlockScout, el cuál puede implementarse a través de Docker dentro del propio nodo (https://github.com/alastria/blockscout). Pero obviamente, para implementar esto se requiere de una cierta cantidad de tiempo, el cual no se tiene, ya que se debe recordar que el desarrollo de este proyecto no debe superar las 300 horas.

Por consiguiente, se ha optado por usar el explorador de bloques que ya está implementado en la red B de Alastria, llamado “Epirus Explorer” y mantenido por Telefónica (https://redb.trustos.telefonica.com/dashboard). Aquí se pueden ver datos generales de la red, como el número total de transacciones, de tokens, de peers y de contratos creados.

Ilustración 56. Epirus Explorer – Dashboard

En el apartado de contratos aparece una lista con todos los contratos desplegados en la red, y accediendo a cada uno de ellos se puede ver información más detallada. Por

ejemplo, en la siguiente imagen se puede ver información acerca del contrato que hemos desplegado anteriormente.

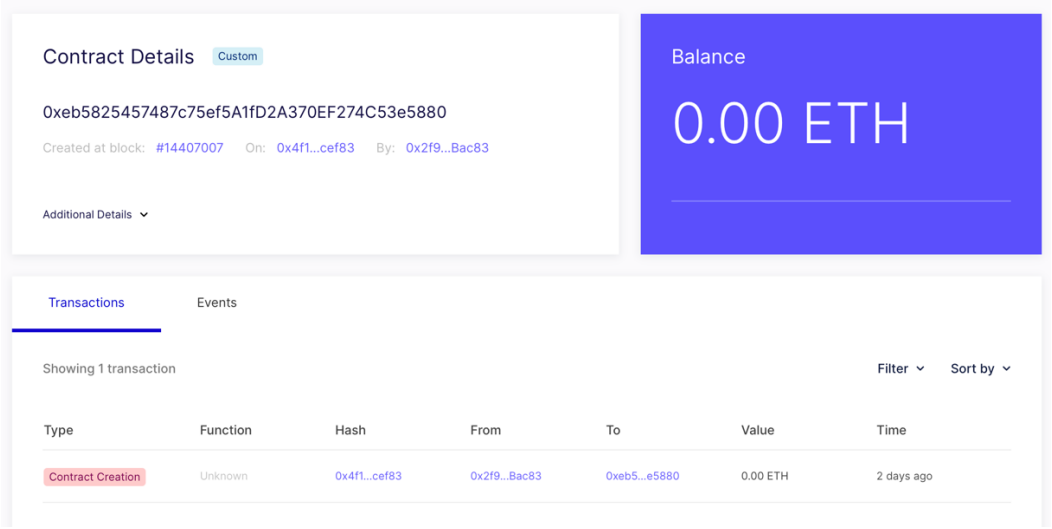


Ilustración 57. Epirus Explorer – Contracts

Los apartados de “transacciones” y “bloques” cuentan con el mismo funcionamiento que el de contratos, pero para que la búsqueda de estos elementos no sea tan tediosa, Epirus Explorer ofrece también un buscador, en el cuál puedes buscar por dirección, hash de la transacción o número de bloque. Por ejemplo, si se consulta la ilustración 52 se puede ver el hash de la transacción del despliegue del contrato (0x4f15cbf1…..), por lo que bastaría solo con escribirlo en el buscador para obtener información detallada.

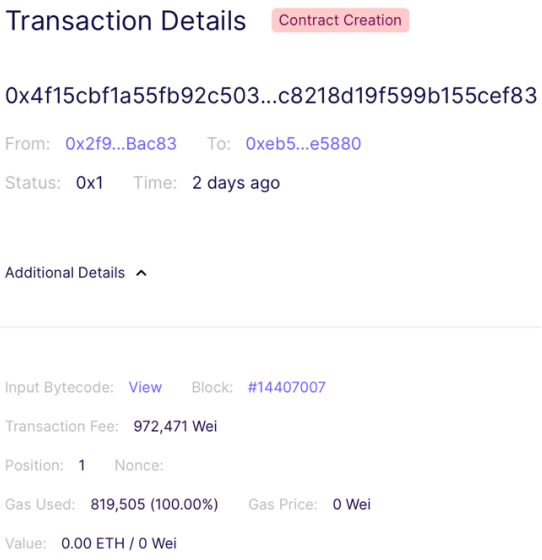


Ilustración 58. Epirus Explorer - Detalles de la transacción

3.4 Cuarta iteración: Integración del sistema biométrico en la Blockchain

Tras tener terminado el diseño de la aplicación móvil, y desplegado el contrato dentro de la Blockchain, toca integrar las funcionalidades desplegadas dentro de la aplicación, para así obtener el producto final.

Aquí existen diversas alternativas:

- Que cada cliente firme las transacciones, por lo que debe tener una wallet activada (ej. Metamask) en el dispositivo. Se implementaría el proceso en el front-end.
- Que el servicio se encargue de firmar las transacciones, implementándose consigo un servidor que sería el encargado de la firma.

Para cumplir con el Esquema Nacional de Seguridad (https://ens.ccn.cni.es/es/esquema-nacional-de-seguridad-ens), y pensando en la protección de la clave privada para la firma, se ha decidido implementar la opción 2, con la cual se tendrá un servidor que hará de middleware entre la aplicación y el contrato, y tendrá las siguientes funcionalidades:

- Recibir registro de un nuevo usuario, validar los campos y enviar transacción firmada con los datos de registro para que se almacenen en la Blockchain.
- Recibir solicitud de inicio de sesión, comprobar si el usuario está registrado, y en caso afirmativo comprobar su hash biométrico con el que tiene almacenado en la Blockchain, tras ello devolver respuesta a la aplicación.

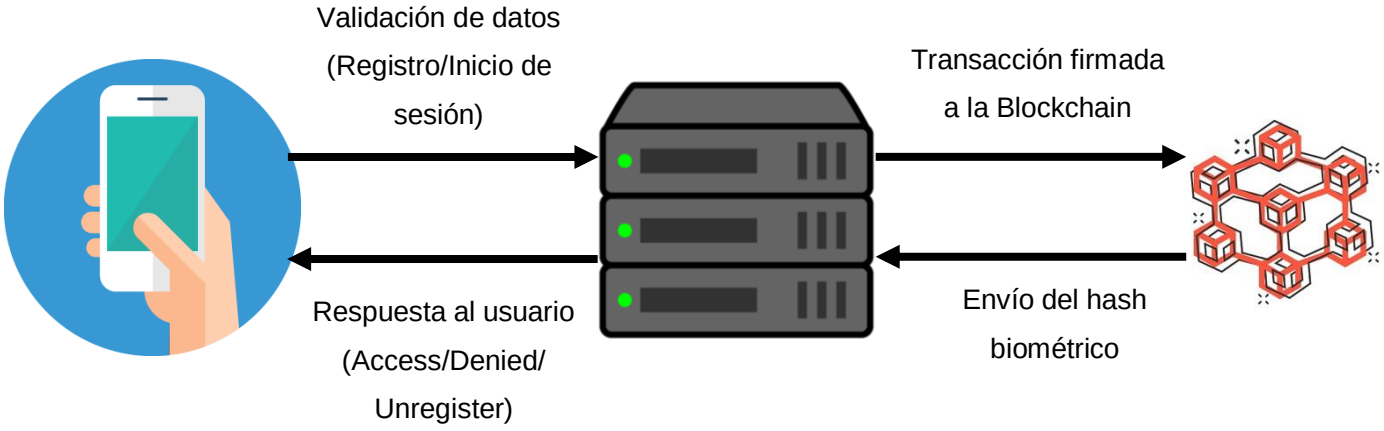


Ilustración 59. Esquema explicativo del funcionamiento del middleware

3.4.1 Implementación del middleware

Como se puede observar en la ilustración anterior, el middleware se encuentra en medio de la conexión entre la aplicación móvil y la red Blockchain. Por ello, en este apartado se va a explicar cómo se ha implementado cada una de estas comunicaciones.

3.4.1.1 Conexión con la aplicación móvil

Para obtener un entendimiento entre el servidor y la aplicación, se han definido dos tipos nuevos: NEW_USER (para cuando la aplicación quiera registrar un nuevo usuario) y LOG_IN (para cuando un usuario quiera autenticarse). Tras ello, ya se evalúan las peticiones dependiendo del tipo de dato que provenga de la aplicación.

Además, el servidor devuelve mensajes a la aplicación para mantener al usuario informado en todo momento del estado de sus peticiones. Devuelve “USEROK” cuando ha logrado registrar un nuevo usuario en la Blockchain, “ACCESS” cuando ha validado los datos de autenticación y son correctos, “DENIED” cuando tras evaluar los datos biométricos no coinciden con los almacenados y “UNREGISTRER” cuando el usuario que quiere acceder al sistema no se encuentra registrado.

```
const typesDef = {
  NEW_USER: "newuser",
  LOG_IN: "login"}

wsServer.on('request', function(request) {
  const connection = request.accept(null, request.origin);

  connection.on('message', function(message) {
    const dataFromClient = JSON.parse(message.utf8Data);

    if (dataFromClient.type === typesDef.NEW_USER) {
      connection.sendUTF("USEROK");
      console.log("Nuevo usuario registrado");

    }else if (dataFromClient.type === typesDef.LOG_IN) {
      connection.sendUTF("ACCESS");
      console.log("El usuario "+dataFromClient.content+" accedió");
    }});

  connection.on('close', function(connection) {
    console.log("El cliente se desconectó");
  });});
```

3.4.1.2 Firma y envío de transacciones a la Blockchain

La conexión con la red Blockchain se centra en tres pasos, que se explican a continuación:

- Conectarse al nodo (instanciar el objeto web3): Es necesaria la IP pública del nodo y tener el puerto 8545 abierto. Por otra parte, el objeto wallet debe recibir la clave privada del nodo, necesaria para firmar transacciones en la Blockchain.

```
//Conexión con el nodo (instancia del objeto web3)
const { ethers } = require("ethers");
const provider = new ethers.providers.JsonRpcProvider("http://54.171.101.63:8545");
const wallet = new ethers.Wallet("PRIVATE-KEY", provider);
```

- Tener el ABI (Interfaz Binaria de Aplicación) disponible e instanciarlo: El ABI es el modo estándar de interactuar con Smart contracts, permitiendo así poder entender de una forma sencilla el contrato desplegado. Este código se puede ver en el archivo de metadatos que se genera tras el despliegue.
- Inicializar el contrato: Se le debe pasar su dirección, el ABI y el objeto wallet.

```
//Inicializar el contrato
const contract = new ethers.Contract("0xeb5825457487c75ef5A1fD2A370EF274C53e5880", abi,
wallet)
```

Una vez hecho todo esto, ya se puede interactuar con el contrato desde el servidor, destacando que automáticamente el objeto wallet sabrá aquellas funciones que tiene que firmar (ayudado por el ABI) y aquellas que son sólo de consulta.

Las 3 funciones desplegadas en el contrato se usan en el servidor de la siguiente manera

```
//Llamar a la funcion de addUser para crear un nuevo usuario
const registro = await contract.addUsuario(user, name, email, phone, bioHASH);

//Comprobar si el usuario está registrado
const getUser = await contract.getUsuario(user);

//Comprobar si el hash biométrico que ha pasado coincide con el almacenado
const acceso = await contract.comparacionBiometrica(user, bioHASH);
```

3.5 Pruebas finales

Para finalizar el desarrollo y pasar a la siguiente etapa, se van a realizar diferentes pruebas para verificar y validar la pasarela de autenticación. Centrándose en aspectos como la seguridad, las interrupciones y la interfaz.

Empezando por las pruebas de seguridad, se pueden definir las siguientes:

Prueba	Respuesta	¿Cómo se obtiene la seguridad?
La aplicación se conecta a una base de datos	Sí	Implementando un middleware que será el encargado de acceder a la red Blockchain, y pueda firmar y enviar transacciones gracias al uso de la clave privada. De esta manera el usuario final no está en disposición de realizar esas acciones a través del front-end.
La aplicación utiliza permisos GPS, cámara, micrófono, etc	No	--
La aplicación usa los datos biométricos del usuario	Sí	Para no tratar, manejar y enviar directamente los datos biométricos en crudo, se calcula un hash a partir de dichos datos (hash biométrico), con el cuál se mantiene la privacidad del usuario, ya que si un agente malicioso obtiene ese hash no va a poder recuperar los datos biométricos por ninguno de los medios.
La aplicación tiene entradas de texto	Sí	Se realiza una validación evitando la entrada de diferentes códigos maliciosos, entradas en blanco de campos obligatorios y tipos de entrada que no se adaptan a lo que realmente se pide. Estas validaciones se centran en las entradas de texto que corresponden con el registro de un nuevo usuario.

Tabla 19. Pruebas de seguridad

Tras realizar las diferentes pruebas referentes a la seguridad toca realizar las que hacen referencia a la interrupción del programa. Toda aplicación puede interrumpirse por distintos eventos externos a la misma, por lo que es necesario hacer pruebas con estas interrupciones para ver si la aplicación funciona de manera correcta cuando trata con ellas. Las interrupciones producidas con tal de probar el funcionamiento del sistema han sido:

- Llamada entrante.
- Batería baja.
- Apagado del móvil por quedarse sin batería.
- Notificaciones de redes sociales.

Obteniendo un resultado muy satisfactorio, ya que la aplicación volvía a funcionar de forma correcta tras una de estas interrupciones, por lo que se puede concluir que esta validación se ha realizado con éxito.

Por último, toca realizar las pruebas referentes a la interfaz, en las que se busca una interacción humano-máquina aceptable (Tabla 20).

Prueba	Explicación
Consistencia de la interfaz en toda la aplicación	Ejecución estable y coherente de la aplicación
Mensajes de error	Mensajes temporales que aparecen en la aplicación cuando sucede un evento inesperado, como por ejemplo intentar registrar un usuario sin completar todos los campos obligatorios
Claridad y alineación	Impedir las pantallas muy cargadas de elementos y que los que haya estén alineados entre sí
Posición y tamaño de los caracteres	Los cuadros de texto se tienen que encontrar en posiciones visibles y el tamaño de los caracteres tiene que ser el mismo en todas las pantallas de la aplicación

Tabla 20. Pruebas realizadas a la interfaz

En cuanto a las pruebas de validación del sistema, se le ha pasado una versión beta a diferentes expertos de Alastria para que dieran su visto bueno, además se le han pasado los archivos de código fuente para que pudieran valorar la optimización del mismo.

Para incidir aún más en la importancia de superar estas pruebas, hay que destacar quiénes han sido los expertos consultados:

- Julio Merelo, CTO de APD y CIO de Alastria.
- Alexander Herranz, Profesor de Ethereum Blockchain en la UPV.
- Iker Ruiz, Blockchain Developer de Alastria.

Los resultados obtenidos son más que satisfactorios, ya que los 3 expertos consultados han dado un feedback excelente acerca de todos los aspectos que han valorado, pudiendo concluir que las pruebas finales de validación de la pasarela han sido pasadas con éxito.

Una vez superadas todas las pruebas, se puede concluir que el desarrollo del sistema ha acabado, y se puede pasar a la etapa de preproducción, en la que se empezará definiendo un modelo de negocio adecuado para el sistema desarrollado.

4 EXPERIMENTACIÓN, RESULTADOS Y DISCUSIÓN

Una vez acabado el apartado 3, ya se tiene disponible el producto final desarrollado en este TFM, por lo que es hora de mostrar los resultados obtenidos y de explicar cómo se comporta la aplicación en diversos casos. Estos casos concretamente van a ser: A la hora de registrarse un usuario, a la hora de que un usuario registrado acceda a la aplicación y cuando un usuario no registrado intente acceder.

Hay que recordar que en el apartado anterior se ha enseñado y explicado cómo se comporta la aplicación en aquellos casos en los que el hardware no soporta identificación biométrica (ilustración 50) y cuando un usuario al registrarse no completa todos los campos obligatorios (ilustración 51), por lo que entre esas explicaciones y las que se van a desarrollar a continuación, se podría dar por cerradas todas las posibilidades que se puede encontrar un usuario al manejar la pasarela de autenticación.

4.1 Registro de un usuario

Se accede a la pantalla de registro, a través del botón “Registrarse” de la pantalla principal. Tras ello, se completan todos los campos con los datos del nuevo usuario y se pincha sobre “Escaneo”, permitiendo así que el dispositivo escanee la huella dactilar del nuevo usuario, para calcular su hash biométrico y poder almacenarla en la red Blockchain.

Una vez hecho todo esto el usuario ya estaría registrado correctamente.

Ilustración 60. Registro de un usuario

4.2 Acceso de usuarios registrados y no registrados

Se va a empezar explicando el acceso de un usuario registrado, siguiendo los datos del apartado anterior de registro, se puede observar cómo se ha dado de alta en la pasarela un usuario con nombre “mgc00104”.

Ahora, este usuario va a autenticarse en la pasarela para poder acceder a la aplicación que desea. Lo primero, es escribir su nombre de usuario y darle al botón de “Iniciar sesión”, tras ello se abre el diálogo de escaneo, en el cual el usuario escanea la huella dactilar para que la aplicación le calcule su hash biométrico y pueda comparar ese hash con el que ya tiene almacenado del usuario en la Blockchain. Si coinciden le da acceso a la aplicación final, en caso contrario se lo notifica.

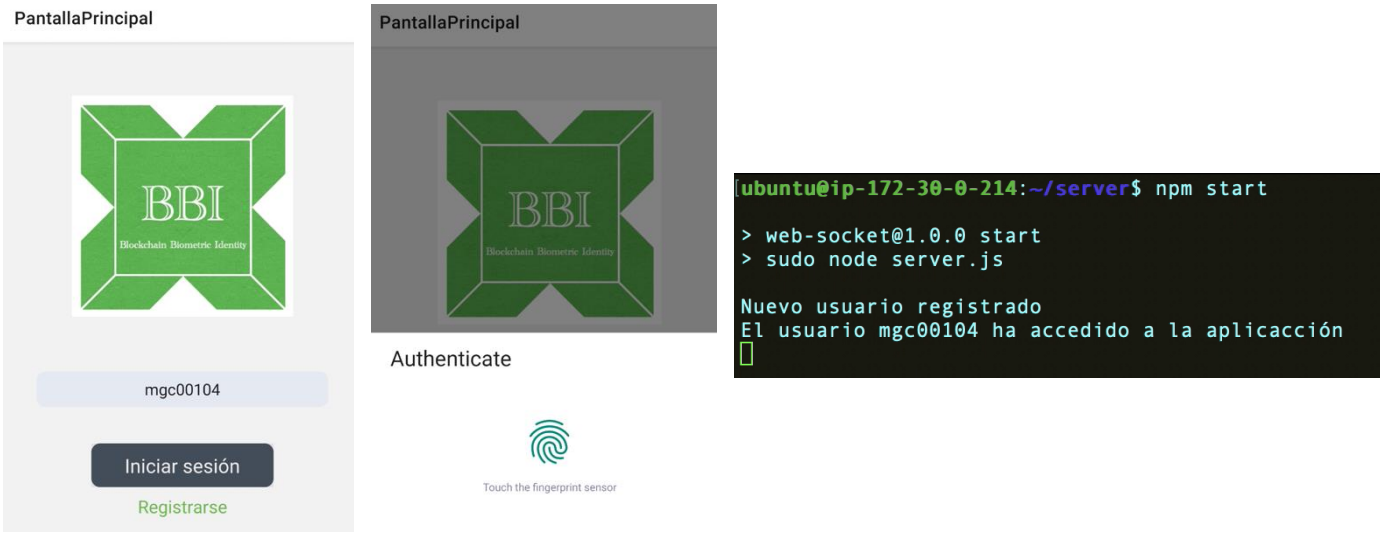


Ilustración 61. Inicio de sesión de un usuario registrado

Por otra parte, cuando un usuario no registrado intenta acceder a la aplicación por medio de esta pasarela, seguiría el mismo proceso que se ha explicado anteriormente, a diferencia de que cuando escanee su huella dactilar, y se consulte si el usuario introducido está registrado y devuelva falso, se le denegará el acceso a la aplicación final y se le notificará. En la siguiente ilustración se puede ver como aparece en el registro LOG del middleware:

Ilustración 62. LOG de acceso de un usuario no registrado

4.3 Resultados y discusión

Lo primero de todo es decir que aparte de las explicaciones dadas en este punto, se ha realizado un vídeo explicativo de la pasarela de autenticación en el que se puede ver cómo se interactúa con la propia interfaz y con el explorador de bloques de la red B de Alastria (Epirus Explorer). El vídeo se encuentra en el siguiente enlace: https://drive.google.com/file/d/1iRYpeNhbbkcFjEK0XcAAC9MkjmvqBxRV/view?usp=sharing

Pasando a comentar los resultados obtenidos en esta fase de desarrollo, hay que destacar diferentes puntos que creo que han facilitado el trabajo realizado:

- La gran cantidad de información/documentación de la cuál dispone Alastria en su repositorio de GitHub (https://github.com/alastria), tanto para levantar un nodo en la red, como para realizar diferentes configuraciones y personalizaciones al mismo.
- La librería ethers.js (https://docs.ethers.io/v5/) que ha permitido enlazar el servidor con el nodo de la red Blockchain de una forma sencilla.
- El framework React Native, en el que mediante el lenguaje de programación JavaScript se ha podido desarrollar la pasarela de autenticación con compatibilidad multiplataforma.

Gracias a lo comentado en estos puntos, y todo lo explicado a lo largo del apartado 3, se ha podido obtener una versión final de la pasarela de autenticación biométrica haciendo uso de la tecnología Blockchain, tal y como se pretendía al inicio del proyecto, y como se definió al principio de este documento en el apartado “1.2 Objetivos del trabajo”.

5 MODELO DE NEGOCIO

Una vez finalizado por completo el desarrollo del proyecto, se puede pasar a la siguiente fase, que es la de comercialización del resultado obtenido.

Para poder abordar mejor esta sección y conseguir detallarla con claridad, se ha dividido la explicación en diferentes apartados, donde se incluyen aspectos como el establecimiento de objetivos desde el punto de vista comercial, análisis del entorno económico y plan de mercado.

5.1 Objetivos

Para todo modelo de negocio es necesario establecer una serie de objetivos cualitativos y cuantitativos desde un punto de vista comercial, que obviamente serán los que marcarán las pautas y pasos a seguir en la comercialización del producto. Esto se puede ver detallado en la siguiente tabla.

Tipo de objetivos	Objetivos
Cualitativos	• Tener una buena reputación • Ofrecer mayor seguridad a los usuarios • Facilitar el proceso de identificación • Comodidad para las aplicaciones y/o servicios a la hora de la identificación • Facilitar el cumplimiento del RGPD
Cuantitativos	• Obtener beneficios • Aumentar los clientes • Minimizar costes de los clientes • Aumentar la popularidad del producto • Maximizar la confianza de empresas y usuarios en la identificación

Tabla 21. Objetivos comerciales

5.2 Análisis del entorno

Para realizar un análisis completo del entorno económico se necesita tener en cuenta factores políticos, tecnológicos, socio-culturales y económicos. Por suerte, existe un diagrama que es perfecto para la realización de este tipo de análisis y que es muy usado en proyectos de diversos sectores, el conocido como análisis DAFO.

Ilustración 63. Estructura del análisis DAFO [40]

Esta herramienta permite estudiar la situación de partida del producto analizando sus características internas a través de las debilidades y fortalezas, y analizando sus características externas a través de amenazas y oportunidades.

Ayuda a explorar nuevas soluciones, a identificar las barreras que limitarán los objetivos propuestos y a decidir sobre la dirección que tomará el producto comercialmente hablando.

	Aspectos negativos	Aspectos positivos
Análisis interno	**DEBILIDADES** • Servicio poco conocido • Fondos limitados para promocionar el producto • Gran número de empresas con procesos de identificación robustos • Posible reticencia de las empresas a cambiar a un proceso descentralizado en el que no controlan los datos de acceso • Poca experiencia en el desarrollo de proyectos	**FORTALEZAS** • Novedoso • Fácil de utilizar • Los clientes no guardan datos de identificación • Fácil distribución • Capacidad de atraer nuevas inversiones • Datos biométricos distribuidos y encriptados • Minimiza costes a los clientes (sin preocupaciones por el RGPD en el proceso y sin mantenimiento de datos de identificación)
Análisis externo	**AMENAZAS** • Servicios similares que puedan hacer la competencia • Servicio más completo que el nuestro • Cambio de tendencias tecnológicas • Cambios en la ley que afecte a los datos personales	**OPORTUNIDADES** • Escasa competencia • Auge de la identificación biométrica • Auge de los servicios descentralizados • Auge de las nuevas tecnologías • Mayor confianza en la identificación biométrica • El número de potenciales clientes va en aumento

Tabla 22. Análisis DAFO

5.3 Plan de mercado

Conseguir un plan de mercado completo pasa por definir cuál va a ser la política llevada a cabo en diferentes aspectos [41]. En este proyecto se han considerado los siguientes:

- Producto: Debe definir qué es lo que vende y las características que tiene.
- Precio: Por cuánto lo va a vender al cliente y de dónde sale el total del dinero generado.
- Distribución: Cómo se va a vender el producto.
- Comunicación: Como será la forma en la que se promocione el producto.

Teniendo en cuenta las dimensiones más relevantes en cuanto al plan de mercado, se proponen una serie de políticas preliminares que sirvan como primeros pasos para la puesta en marcha del prototipo como producto comercial.

5.3.1 Política de producto

	Explicación
¿Qué se vende?	Servicio de identificación biométrico en Blockchain, ofreciendo una pasarela de identificación para el acceso a servicios y aplicaciones digitales.
Características	• Integración de la tecnología biométrica con la tecnología Blockchain • Reducción de costes en el proceso de identificación • Servicio disponible para todos los socios de la red Blockchain • Almacenamiento seguro de datos biométricos • Disponible para IOS y Android • Facilita la identificación al usuario final

Tabla 23. Política de producto

5.3.2 Política de precio

A día de hoy existen diversas políticas de precio para servicios software, intentando en todo momento adaptarse a las necesidades del cliente y con ello optimizarle el gasto en el propio producto. Como se suele decir, hacer pagar por lo que realmente consumes. Adaptándolo al producto que se ofrece en este proyecto, se podrían tener las siguientes políticas:

- **Pago por identificaciones**, es decir, tener un precio fijo por cada una de las identificaciones que el software del cliente haga mediante nuestra pasarela.
- **Tarifa mensual fija**, pensado para empresas muy grandes en las que cada día pasan por la pasarela multitud de usuarios. A estas empresas le sale más rentable adquirir el servicio como un gasto fijo, independientemente del número de identificaciones.
- **Pago mensual variable**, es una mezcla de las dos políticas anteriores, adaptada a incrementos de identificaciones predecibles, pudiendo dar la facilidad por ejemplo de dividir una mensualidad en 2 tarifas diferentes, por ejemplo, los primeros 15 días tener la política de tarifa mensual fija (porque esos 15 días esperas un gran número de identificaciones) y para los otros 15 días seguir una política de pago por iteraciones (porque prevés que vas a tener pocos accesos).

Viendo las diferentes políticas, se pueden definir una serie de características con las cuáles se puede entender mejor a qué tarifa se adapta un software.

Tipo de pago	Usuarios	Incrementos de identificaciones	Ejemplo
Pago por identificaciones	Bajo/medio número de identificaciones diarias	Predecible	Aplicación de citas para peluquería (Booksy)
Pago mensual fijo	Alto número de identificaciones diarias	Impredecible	Aplicación de música (Spotify)
Pago mensual variable	Alto número de identificaciones diarias	Predecible	Aplicación de banco (BBVA)

Tabla 24. Resumen de las diferentes tarifas

Tras explicar las facilidades que se pueden dar mediante las diferentes tarifas, se va a pasar a comparar el precio de cada una de ellas (sólo se van a comparar las dos primeras tarifas, ya que la tercera es un mix de las otras), teniendo en cuenta obviamente que esto es un primer estudio y cuando el producto llegue a fase de producción estos precios pueden variar. Toda la información se muestra en la siguiente tabla:

Tipo de tarifa	Identificaciones mensuales	Precio por identificación	Precio total
Pago por identificaciones	<10.000	0.03€	-
Pago por identificaciones	10.000-100.000	0.02€	-
Pago por identificaciones	+100.000	0.015€	-
Tarifa mensual	No depende de número de identificaciones	-	13.500€

Tabla 25. Precios de cada tarifa

Lo ideal viendo estas tarifas es calcular cuándo deja de ser rentable contratar la tarifa de pago por iteraciones y pasar a la tarifa de pago fijo mensual. Esto se puede calcular fácilmente con la siguiente fórmula: 13.500/0.015 = 900.000, por lo que a partir de 900.000 identificaciones mensuales sale más barato contratar la tarifa mensual.

Como se puede observar, y como se comentó anteriormente, esta tarifa sólo es rentable para aquellas aplicaciones/servicios muy grandes que cuenten con miles y miles de accesos diarios.

Por último, hablando de la tarifa mensual variable, la idea es pagar el 50% de la tarifa mensual fija para aquellos 15 días de mayor multitud de accesos, y pasar a la de pago por iteraciones para los 15 días de menos accesos.

5.3.3 Estrategia de comunicación y distribución

Considerando lo importante que es la obtención de nuevos clientes y lo que supone una buena estrategia publicitaria, hay que diseñar una campaña eficiente y eficaz, intentando lograr el máximo número posible de clientes.

Para ello, los diferentes métodos de publicidad para este producto serán:

- Publicidad en páginas especializadas en identificación digital.
- Patrocinar eventos de seguridad informática.
- Realizar charlas en congresos.
- Posicionar el producto para búsquedas en Google de soluciones de identificación para servicios/apps.
- Realizar material audiovisual, en la que los posibles clientes puedan ver el funcionamiento externo del sistema.
- Realizar presentaciones técnicas en los que demostrar la seguridad del proceso de identificación.

Por otra parte, y para finalizar hay que hablar sobre la forma de distribución del producto, que supone una de las decisiones más importantes a la hora de lanzar y hacer visible el servicio, y que repercutirá en el futuro del mismo. Existen diferentes estrategias para la distribución, como son:

- Módulo programado en el que se encuentre toda la lógica del producto junto con las conexiones necesarias a la Blockchain y que permita adaptarse a la interfaz del cliente, para que así no difiera del diseño general del servicio/aplicación.
- Solución individualizada a cada cliente, optimizando los recursos hardware y software donde se implementará.

En este caso se ha optado por la primera opción, ya que permite ofrecer un precio más competitivo a los clientes, permitiendo además que el responsable de la aplicación pueda modular el servicio convenientemente a sus necesidades.

6 CONCLUSIONES Y TRABAJOS FUTUROS

Una vez terminado el proyecto se pueden presentar una serie de conclusiones obtenidas del mismo:

- El desarrollo y comercialización de este producto supone un gran avance en la identificación biométrica.
- El uso de la tecnología Blockchain permite obtener un sistema de identificación biométrica descentralizado y totalmente seguro.
- Ambas tecnologías cuentan con un gran potencial, y presumiblemente junto a la inteligencia artificial sean los 3 sectores tecnológicos que liderarán el avance científico-técnico en los próximos años.
- El uso generalizado del smartphone y su mejora continua han permitido que la bioidentificación sea reconocida como un método muy útil de identificación por toda la sociedad.
- Cada vez más empresas y particulares hacen uso de los distintos métodos de identificación biométrica, por lo que es un sector al alza.
- El Blockchain se está consolidando en la sociedad, permitiendo descentralizar procesos para acabar con todos los inconvenientes de los sistemas centralizados tradicionales.
- El resultado final de este proyecto es obtener un sistema piloto de identificación, que pueda tener una implicación exponencial en la sociedad, ya que abre el camino a la idea de que una persona pueda identificarse en todos sitios a través de sus rasgos biométricos, y con esto olvidarnos de la idea de cuentas de usuario. Un ejemplo básico sería que un usuario pudiera acceder a las instalaciones de su empresa a través de su huella, y que bajo ese mismo sistema global y descentralizado pudiera acceder mediante la biometría a su cuenta de correo a través del móvil.

Además de las conclusiones presentadas, hay que comentar las principales fortalezas que tiene este trabajo con respecto a otros sistemas similares, poniendo en valor el avance logrado con el desarrollo de este proyecto. Las más interesantes son:

- Sistema de bioidentificación integrado en Blockchain, ofreciendo multitud de ventajas respecto a los sistemas centralizados.
- Fácil y rápida implementación en todo tipo de sistemas de identificación biométricos, ya que no está solo desarrollado para un caso de uso particular.
- Registrándose en este sistema el usuario puede acceder a cualquier servicio que esté adherido a la red Blockchain, no teniendo que crear y recordar multitud de cuentas distintas.

Por último, pueden definirse varias líneas de trabajos futuros, todos ellos con el objetivo común de intentar mejorar y adaptarse a los paradigmas tecnológicos y sociales actuales y futuros, y con la finalidad de facilitar y conseguir una identificación de usuario totalmente segura, óptima, y que permita reducir el coste a las empresas.

Las líneas por las que pueden seguir los trabajos futuros pueden ser:

- Estudio y desarrollo de un estándar para que los diferentes sistemas biométricos traten y almacenen los datos de la misma forma.
- Desarrollo de un protocolo que permita a los Smartphone poder compartir los datos biométricos almacenados, asegurando el anonimato del usuario y cumpliendo el RGPD. Una idea podría ser hashear los datos almacenados, y aplicarle hashes derivados.
- Seguir apostando por el uso de una pasarela de identificación única, basándose en las ventajas que ofrece tanto a empresas como a usuarios.

7 APÉNDICES

7.1 Guía original del Trabajo Fin de Título

PROPUESTA DE TRABAJO DE FIN DE MASTER

MASTER EN:
Master en Seguridad Informática

TÍTULO DEL TRABAJO:
Tecnologías blockchain aplicadas a ciberseguridad

Idioma:
Castellano

DESCRIPCION CORTA DEL TFM:
En este proyecto se realizará un estudio de blockchain y tecnologías asociadas, aplicado al ámbito de la ciberseguridad. Tras este análisis y estudio se pondrá en marcha un caso de uso.

DATOS DEL TRABAJO FIN DE GRADO:

Modalidad del TFM
Trabajo Teórico/Experimental

TUTOR DEL TRABAJO FIN DE GRADO:

Tutor/a

Nombre:
MIGUEL ÁNGEL GARCÍA CUMBRERAS
Departamento:
Informática
A. Conocimiento:
LENGUAJES Y SISTEMAS INFORMÁTICOS

☑ **Este TFM tiene un segundo tutor**
○ **El tutor es de la Escuela**
◉ **El tutor es ajeno al Centro**

Segundo Tutor

Nombre:
Julio J. Merelo Casado
DNI:
77331475R
Empresa/Entidad:
APD
Puesto:
Director tecnológico
Nivel Académico:
Ingeniero en Informática

DATOS DEL ALUMNO ASIGNADO

Alumno/a asignado/a

Nombre:
Manuel Gallego Chinchilla
DNI:
26503035C
Dirección:
CL CARRASCAL 26, SILES
Teléfono:
676723607
Correo-E:
mgc00104@red.ujaen.es

CONOCIMIENTOS/REQUISITOS PREVIOS RECOMENDADOS

No se requieren.

OBJETIVOS DEL TFM:

1. Realizar un estudio de blockchain y tecnologías relacionadas, aplicadas a la ciberseguridad.
2. Analizar distintos casos de estudio.
3. Seleccionar un caso de estudio y ponerlo en práctica.
4. Redactar una memoria que recoja todo el trabajo llevado a cabo.

METODOLOGÍA A DESARROLLAR:

Se utilizará una metodología tradicional desde el punto de vista de investigación teórica y el diseño y puesta en marcha de un caso de uso práctico.

DOCUMENTOS Y FORMATOS DE ENTREGA:

La documentación se entregará en formato digital y estará compuesta por:

- Memoria de TFM, en formato PDF
- Anexos para la presentación del TFM, en PDF:
  - Propuesta de TFM, e histórico de modificaciones si ha tenido.
  - Informe favorable del tutor.
  - Autorización de publicación

DOCUMENTOS ENTREGADOS:

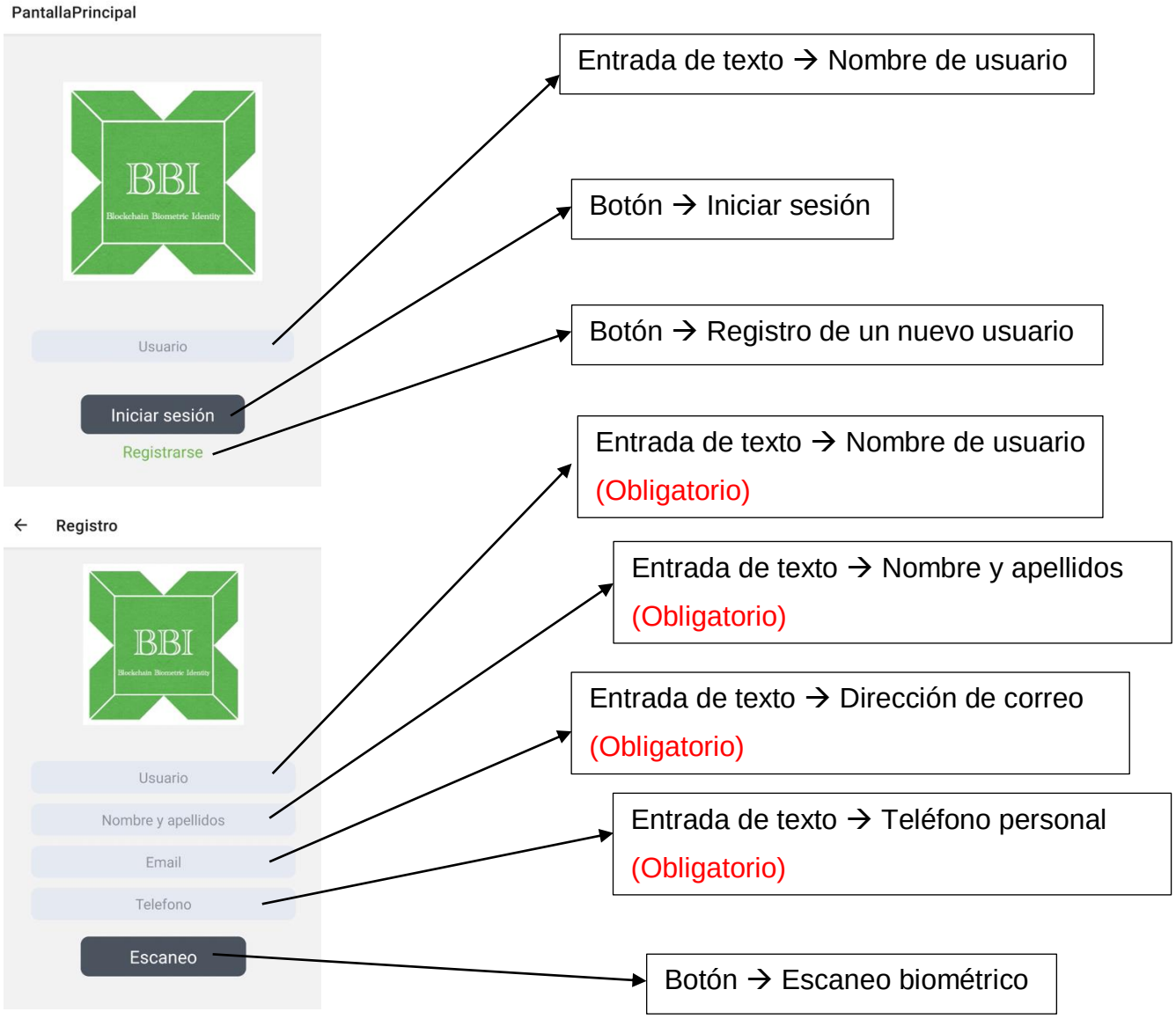
Documentos entregados

- [x] **Solicitud de propuesta de TFM cumplimenada y firmada.**
- [x] **Acreditación del nivel académico del tutor/a.**
  Solo en caso de no vinculación a la Universidad.

7.2 Manual de usuario

Debido al desarrollo de un prototipo software como resultado de este proyecto, se presenta a continuación un breve manual dirigido a facilitar su uso tanto por usuarios experimentados como nóveles. A lo largo de esta guía de usuario, la explicación de cada funcionalidad se puede revisar atendiendo a la explicación textual y al conjunto de imágenes, que representan la interfaz de la aplicación móvil.

Cabe destacar que el objetivo final de este proyecto no ha sido centrarse al 100% en la interfaz, ya que lo principal era el tratamiento seguro de los datos biométricos en la Blockchain, por lo que la interfaz que se presenta es simple y muy intuitiva, desarrollada para probar de una forma sencilla el correcto funcionamiento de lo planteado.



8 DEFINICIONES Y ABREVIATURAS

8.1 Definiciones

En la siguiente tabla se presentan algunos términos usados a lo largo del trabajo que requieren ser explicados para obtener así una mejor comprensión del mismo.

Término	Definición
Bioidentidad	Identificación de un usuario a partir de sus rasgos biométricos
Blockchain	Tecnología que permite almacenar datos de una manera que es casi imposible de falsificar
Diagrama de Gantt	Herramienta gráfica que ayuda a ilustrar el tiempo dedicado a cada tarea a lo largo del tiempo
Peers	Nodos de la red con los que estás conectado
Alastria	Consorcio de servicios corporativos con Blockchain
Hyperledger Besu	Cliente de Ethereum que se optimiza cuando trabaja en entornos de consorcio
React native	Framework para desarrollo multiplataforma
Smart contract	Es un programa informático que asegura, facilita, hace cumplir y ejecuta acuerdos registrados entre diversas partes

Tabla 26. Definiciones

8.2 Abreviaturas

En la siguiente tabla se presentan algunas abreviaturas usadas a lo largo del trabajo que requieren ser explicadas para obtener así una mejor comprensión del mismo.

Acrónimo	Significado
LOPD	Ley orgánica de protección de datos
BOE	Boletín oficial del estado
Dapps	Aplicaciones descentralizadas
P2P	Punto a punto
ARCO	Acceso, rectificación, cancelación y oposición
RGPD	Reglamento general de protección de datos
DAFO	Debilidades, amenazas, fortalezas y oportunidades
MADEJA	Marco de desarrollo de la Junta de Andalucía
FAR	Falsa aceptación
FRR	Falso rechazo
SLOC	Líneas de código fuente
SO	Sistema operativo

Tabla 27. Abreviaturas

9 BIBLIOGRAFÍA

[1] «La biometría inteligente va a desplazar a los passwords por siempre?», *Panamax, Inc.*, 23 de septiembre de 2021. https://www.panamaxil.com/es/blog/porque-biometria-inteligente-se-esta-apoderando-de-las-contrasenas (accedido 27 de mayo de 2022).

[2] «Blockchain: la tecnología del pasado que está transformando el futuro » Ideas». https://www.economiadigital.es/ideas/blockchain-la-tecnologia-del-pasado-que-esta-transformando-el-futuro.html (accedido 27 de mayo de 2022).

[3] F. Serratosa, «La biometría para la identificación de las personas», p. 50.

[4] «Infografía: En dos años, todos los smartphones estarán equipados con tecnología biométrica», *Statista Infografías*. https://es.statista.com/grafico/11129/en-dos-anos-todos-los-smartphones-estaran-equipados-con-tecnologia-biometrica/ (accedido 27 de mayo de 2022).

[5] «Historia de la tecnología Blockchain: Guía definitiva», *101 Blockchains*, 3 de diciembre de 2018. https://101blockchains.com/es/historia-de-la-blockchain/ (accedido 13 de junio de 2022).

[6] «Sistema de votación de participación ciudadana basado en tecnología blockchain», *ESMARTCITY*. https://www.esmartcity.es/comunicaciones/comunicacion-sistema-votacion-participacion-ciudadana-basado-tecnologia-blockchain (accedido 4 de junio de 2022).

[7] «6 aplicaciones de blockchain en la industria alimentaria y agrícola», *Bitcoin Mexico - El mejor portal de noticias Bitcoin*, 10 de agosto de 2020. https://www.bitcoin.com.mx/6-aplicaciones-de-blockchain-en-la-industria-alimentaria-y-agricola/ (accedido 4 de junio de 2022).

[8] «4-objetivos-sistema-gestion-de-contratos | Software de Gestion de Contratos - Administracion y Control», 4 de abril de 2016. https://www.comforce.co/4-objetivos-de-un-sistema-de-gestion-de-contratos/4-objetivos-sistema-gestion-de-contratos/ (accedido 5 de junio de 2022).

[9] «Nasuni. A blockchain-like system to store information», *Grayhats*, 17 de enero de 2021. https://www.grayhats.com/blog/nasuni-a-blockchain-like-system-to-

store-information/ (accedido 4 de junio de 2022).

[10] A. B. Ecosystem, «La identidad digital de Alastria presenta su primer MVP», *Medium*, 8 de abril de 2020. https://alastria-es.medium.com/la-identidad-digital-de-alastria-presenta-su-primer-mvp-696750d687ac (accedido 4 de junio de 2022).

[11] «FICHA RETO Bio-identidad segura con Blockchain.pdf». Accedido: 17 de junio de 2022. [En línea]. Disponible en: https://www.lasrozas.es/sites/default/files/inline-files/FICHA%20RETO%20Bio-identidad%20segura%20con%20Blockchain.pdf

[12] «Biométrica: sistemas con mas potencial de aceptación Europa 2018», *Statista*. https://es.statista.com/estadisticas/1035767/sistemas-biometricos-con-mas-potencial-de-aceptacion-en-europa/ (accedido 14 de junio de 2022).

[13] C. T. Borja y Á. G. Bueno, «Sistemas Biométricos», p. 39.

[14] «Sensores de huellas dactilares, como funcionan?» https://www.biometricos.net/2018/04/sensores-de-huellas-dactilares-como.html (accedido 14 de julio de 2022).

[15] «Tech InDepth: comprensión de los sensores de huellas dactilares capacitivos, ópticos y ultrasónicos», *Noticias del Mundo en español*, 20 de abril de 2022. https://noticiasdelmundo.news/tech-indepth-comprension-de-los-sensores-de-huellas-dactilares-capacitivos-opticos-y-ultrasonicos/ (accedido 14 de julio de 2022).

[16] «¿Qué es un sensor de huellas dactilares? ¿Se le puede engañar?» https://ayudaleyprotecciondatos.es/2020/05/18/sensor-huellas-dactilares/ (accedido 14 de junio de 2022).

[17] «DRAFT Strength of Function for Authenticators - Biometrics». https://pages.nist.gov/SOFA.html (accedido 14 de julio de 2022).

[18] «Fingerprint Minutiae Viewer (FpMV)», *NIST*, mar. 2014, Accedido: 14 de julio de 2022. [En línea]. Disponible en: https://www.nist.gov/services-resources/software/fingerprint-minutiae-viewer-fpmv

[19] «detect-faces | API de Cloud Vision», *Google Cloud*. https://cloud.google.com/vision/docs/detecting-faces?hl=es-419 (accedido 14 de julio de 2022).

[20] «Luxand.cloud - Facial Landmark Detection API». https://luxand.cloud/luxand.cloud (accedido 14 de julio de 2022).

[21] «Cómo funciona el reconocimiento facial y su seguridad».

https://www.electronicid.eu/es/blog/post/como-funciona-reconocimiento-facial/es (accedido 14 de junio de 2022).

[22] A. Biometrics, «Reconocimiento facial», *Aware*. https://www.aware.com/es/reconocimiento-facial/ (accedido 14 de junio de 2022).

[23] «Inicio - NtechLab», *NtechLab*. https://ntechlab.com/es/ (accedido 14 de junio de 2022).

[24] «¿Qué es la tecnología de blockchain? - IBM Blockchain | IBM». https://www.ibm.com/es-es/topics/what-is-blockchain (accedido 14 de junio de 2022).

[25] J. S. Hurtado, «Qué es Blockchain y cómo funciona la tecnología Blockchain», *Think. Innov.*, mar. 2022, Accedido: 14 de junio de 2022. [En línea]. Disponible en: https://www.iebschool.com/blog/blockchain-cadena-bloques-revoluciona-sector-financiero-finanzas/

[26] «Modelo Incremental». https://www.instintoprogramador.com.mx/2018/04/modelo-incremental.html (accedido 14 de junio de 2022).

[27] «Modelos de Desarrollo». https://josmer1202.wixsite.com/ingenieriasoftware/single-post/2015/04/20/modelos-de-desarrollo (accedido 14 de junio de 2022).

[28] J. Barato, «Los Hábitos de un Director de Proyectos Eficaz: Gestión Eficaz de los Riesgos del Proyecto», *Los Hábitos de un Director de Proyectos Eficaz*, 6 de abril de 2014. http://jose-barato.blogspot.com/2014/04/gestion-eficaz-de-los-riesgos-del.html (accedido 30 de mayo de 2022).

[29] «Guía para estimar usando story points». https://es.linkedin.com/pulse/gu%C3%ADa-para-estimar-usando-story-points-roberto-a-r-moran (accedido 30 de mayo de 2022).

[30] «Alastria Node for Alastria-T Network». Alastria Blockchain Ecosystem, 28 de mayo de 2022. Accedido: 15 de junio de 2022. [En línea]. Disponible en: https://github.com/alastria/alastria-node-quorum

[31] «Los datos biométricos son datos personales según el RGPD y la LOPD», *ClickDatos*, 21 de enero de 2018. https://clickdatos.es/regulacion-datos-biometricos-lopd-y-rgpd/ (accedido 30 de mayo de 2022).

[32] «Guía para la redacción de casos de uso | Marco de Desarrollo de la Junta de Andalucía». https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/416 (accedido

21 de junio de 2022).

[33] K. Technologies, «A beginner's guide to Shamir's Secret Sharing», *Medium*, 7 de marzo de 2020. https://medium.com/@keylesstech/a-beginners-guide-to-shamir-s-secret-sharing-e864efbf3648 (accedido 23 de junio de 2022).

[34] A. J. A. Zamora, «Estudio comparativo de frameworks multiplataforma para desarrollo de aplicaciones móviles», p. 111.

[35] Bruno, «Comparación React contra Angular», *Medium*, 11 de abril de 2018. https://codearmy.co/comparaci%C3%B3n-react-contra-angular-2402f761b14e (accedido 3 de agosto de 2022).

[36] «React Native: ¿Qué es y para que sirve?», *OpenWebinars.net*, 18 de junio de 2019. https://openwebinars.net/blog/react-native-que-es-para-que-sirve/ (accedido 3 de agosto de 2022).

[37] «TEE de confianza», *Android Open Source Project*. https://source.android.com/security/trusty?hl=es-419 (accedido 2 de agosto de 2022).

[38] S. Butt, «Implement Touch ID and Face ID Authentication in Your React Native App», *Medium*, 19 de mayo de 2020. https://betterprogramming.pub/touch-id-and-face-id-authentication-in-your-react-native-app-1e455c1fe080 (accedido 2 de agosto de 2022).

[39] «react-native-bytte-bio-lib», *npm*. https://www.npmjs.com/package/react-native-bytte-bio-lib (accedido 2 de agosto de 2022).

[40] «Qué es un análisis DAFO y cómo se utiliza en la empresa». https://www.mastermarketing-valencia.com/ventas-y-gestion-comercial/blog/que-es-dafo/ (accedido 4 de julio de 2022).

[41] «¿Qué aspectos debe contener una política de producto?», 19 de septiembre de 2017. https://www.das.es/blog/politica-de-producto/ (accedido 4 de julio de 2022).