



Escola d'Enginyeria de Telecomunicació i
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

IDENTIDAD DIGITAL SOBERANA

TÍTOL DEL TFG: Identidad digital soberana

TITULACIÓ: Grau en Enginyeria Telemàtica

AUTOR: Laia Muñoz Moruno

DIRECTOR: Olga León Abarca

Juan Hernández Serrano

DATA: 7 de setembre del 2020

Título: Identidad digital soberana

Autor: Laia Muñoz Moruno

Director: Olga León Abarca

Juan Hernández Serrano

Data: 7 de setembre del 2020

Resumen

El reciente desarrollo de las TIC (Tecnologías de la Información y la Comunicación) ha cambiado por completo el concepto de identidad, ahora congrega conjuntamente los diferentes planos sociales del individuo: familia, trabajo y amigos. La identidad digital se construye ya no solo a partir de lo que somos, sino también de lo que hacemos en la red y de cómo nos relacionamos, por ello ha tomado tanta relevancia. A su vez, con más de tres mil millones de usuarios en Internet, cada uno con múltiples identidades digitales, la gestión de estas identidades es muy importante. Las personas a menudo tenemos la sensación de que no disponemos del control de nuestra información, debido a la gran cantidad de empresas privadas que a diario comercializan con nuestros datos sin nuestro consentimiento explícito.

Es así como surge el concepto de identidad digital soberana, una expresión de identidad digital en la que el usuario tiene pleno control de sus datos, manejando así quiénes pueden acceder a ellos y en qué términos.

En este proyecto, el principal objetivo es dar a conocer uPort, una potente tecnología que próximamente formará parte de nuestra vida diaria. Para lograr esa meta, se ha llevado a cabo un análisis en profundidad de este nuevo paradigma de identidad, presentando las principales diferencias y ventajas con respecto a un sistema de identidad centralizado. Por otro lado, se ha realizado una búsqueda de posibles casos de uso y se ha implementado una aplicación llamada Cibertickets que demuestra cómo, haciendo uso de uPort como un sistema de gestión de la identidad digital soberana, se solucionan múltiples problemas del proceso de compra de entradas para eventos, cómo por ejemplo, evitar el uso de bots para comprar tickets masivamente y poder revenderlos, la frecuente dependencia de terceras empresas intermediarias como Facebook o Google para realizar el registro o incluso, la dificultad de ciertas compañías para cumplir con la GDPR (General Data Protection Regulation).

Título: Identidad digital soberana

Autor: Laia Muñoz Moruno

Director: Olga León Abarca

Juan Hernández Serrano

Data: 7 de setembre del 2020

Overview

The recent development of ICT (Information and Communications Technologies) has radically altered the concept of identity, which today embodies the different social levels of an individual, such as: family, work and friends. Digital identity is built not only upon what we are, but also from what we do online and how we interact to others, that is why it has become so relevant. Moreover, with more than three billion users on the Internet, each of them with multiple digital identities, managing these identities is very important. People often have the feeling that we do not have control of our information, due to the large number of private companies that daily commercialize with our data without our explicit consent.

This is how the concept of sovereign digital identity arises, it is an expression of digital identity in which the user has full control of their data, apart from being able to manage who can access them and on which terms.

In this project, the main objective is to promote uPort, a powerful technology that will be soon part of our daily lives. To achieve this goal, a depth analysis of this new identity paradigm has been carried out, outlining the main differences and advantages with respect to a centralized identity system. On the other hand, a search for possible use cases has been executed and an application called Cibertickets has been implemented. It shows how using uPort as a management system of sovereign digital identity, multiple problems in the process of buying tickets for events are solved. For example, prevent the use of bots to buy tickets massively and be able to resell them, the frequent dependence on third-party intermediary companies such as Facebook or Google to register or even the difficulty of certain companies to accomplish the GDPR (General Data Protection Regulation).

AGRADECIMIENTOS

Después de una intensa etapa de 8 meses, hoy es el gran día: escribo este apartado de agradecimiento para dar por finalizado mi trabajo de final de grado. Ha sido una etapa de mucho aprendizaje, tanto a nivel personal como profesional. Realizar este trabajo ha tenido un impacto muy positivo en mí y es por ello que me gustaría agradecer a todas aquellas personas que me han ayudado y apoyado en este camino.

En primer lugar, me gustaría agradecer a mis tutores, Olga León y Juan Hernández, por haber tutorizado y enfocado el proyecto como es debido. Me han sabido guiar en todo momento y enseñarme nuevas referencias que me han sido de gran ayuda para el proyecto, ya que desde nuestra primera reunión mostraron una gran confianza en que lo podíamos lograr y supieron transmitirme esa ilusión.

Gracias a la Universidad Politécnica de Catalunya, por haberme facilitado a lo largo de todos estos años los conocimientos suficientes y las herramientas necesarias para la realización de este Trabajo Final de Grado, ya que, de no ser así, no me habría sido posible desarrollar esta investigación.

Por último, pero no por ello menos importante, gracias a mi familia y amigos que han supuesto en todo momento un apoyo moral esencial, ayudándome a superar los obstáculos a nivel psicológico que un proyecto tan grande lleva consigo, así como por sus sabios consejos y comprensión continua.

¡Muchas gracias a todos!

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1. Identidad digital	2
1.1 Fundamentos de la identidad	2
1.1.1 Datos personales y sensibles	3
1.2 Diferenciación entre identidad e identidad digital	4
1.3 Identidad digital soberana en profundidad	5
1.4 Casos de uso de la identidad digital soberana	6
1.4.1 Identificación en servicios web	6
1.4.2 Verificación de credenciales educativas y titulaciones	6
1.4.3 Identificación segura en sistemas de votación electrónicos	6
1.5 Tres modelos de identidad digital	7
1.5.1 Identidad centralizada o tradicional	7
1.5.2 Identidad federada	9
1.5.3 Identidad descentralizada o centrada en el usuario	10
1.6 Problemas de los modelos: Tradicional e IDP	11
1.6.1 Pérdida del control de nuestros datos	13
1.7 Necesidad de descentralización en la gestión de la identidad	14
1.8 Retos del sistema de gestión de identidad digital	15
CAPÍTULO 2. uPort	17
2.1 ¿Qué es uPort?	17
2.2 Beneficios de uPort	18
2.2.1 Requerimientos de infraestructura bajos	18
2.2.2 Disminución del riesgo de seguridad	18
2.2.3 Facilita el cumplimiento del GDPR	18
2.2.4 Costes reducidos	19
2.2.5 Escalable	19
2.3 Creación de nuestra identidad	19
2.3.1 DIDs	20
2.3.2 Vincular datos extra con un DID determinado	20
2.4 Almacenamiento e intercambio de datos off-chain	21
2.4.1 JWT	22
2.5 Interacción con otras aplicaciones	25

2.6	Implementación real de uPort en Zug	27
2.6.1	Contexto social	27
2.6.2	Funcionamiento	28
2.6.3	Experiencia del usuario	28
CAPÍTULO 3.	Problemas existentes en los sistemas actuales	31
3.1	Problemática para los usuarios en la compra de entradas.....	31
3.1.1	¿Qué son los bots y son todos malos?	31
3.1.2	¿Cómo funcionan?	32
3.1.3	Legalidad	35
3.2	Problemas de almacenamiento de datos para las empresas.....	36
3.2.1	Historia y conceptos básicos de la GDPR	36
3.2.2	Principios de protección de datos	37
3.2.3	Derechos de las personas	38
3.2.4	Multas y sanciones	42
CAPÍTULO 4.	Cibertickets, nuestra solución basada en uPort	44
4.1	Tecnologías, herramientas y lenguajes utilizadas	45
4.1.1	MongoDB.....	45
4.1.2	Angular	45
4.1.3	NodeJS.....	45
4.1.4	ExpressJS	46
4.1.5	Git.....	46
4.1.6	Mockplus	46
4.2	Patrón MVC.....	47
4.3	Librerías utilizadas	48
4.3.1	uPort Credentials	48
4.3.2	uPort Transports	51
4.3.3	did-JWT y did-RESOLVER	57
4.4	Algoritmos criptográficos utilizados	58
4.4.1	Diffie-Hellman	58
4.4.2	X25529 / CURVE 25519.....	58
4.4.3	ES256K-R.....	58
4.5	Mockups e interfaz de usuario	59
4.5.1	Pantalla de inicio.....	59

4.5.2	Información detallada de un evento.....	60
4.5.3	Confirmación de compra.....	61
4.5.4	Autenticación y pago del propietario.....	61
4.5.5	Ver tus entradas compradas.....	62
4.5.6	Intercambio de una entrada.....	62
4.5.7	Autenticación y pago del nuevo comprador.....	63
4.5.8	Verificación de la validez de una entrada.....	63
4.6	Implementación.....	64
4.6.1	Compra de entradas.....	64
4.6.2	Intercambio de entrada.....	68
4.6.3	Verificación de la entrada para el acceso al concierto.....	69
CONCLUSIONES Y LÍNEAS FUTURAS.....		73
BIBLIOGRAFÍA.....		74

LISTA DE FIGURAS

Fig. 1.1 Diagrama de identidad centralizada	7
Fig. 1.2 Diagrama de identidad federada	9
Fig. 1.3 Diagrama de identidad descentralizada	10
Fig. 2.1 Formato de un DID	20
Fig. 2.2 Formato del encabezamiento de un JWT	23
Fig. 2.3 Formato del payload de un JWT	24
Fig. 2.4 Formato de la firma de un JWT	25
Fig. 2.5 Formato de un JWT completo y codificado	25
Fig. 2.6 Interfaz de usuario para el flujo de divulgación selectiva.....	26
Fig. 2.7 Diagrama de flujo del proceso de divulgación selectiva [24]	27
Fig. 2.8 Esquema del funcionamiento de la identidad digital en Zug.....	29
Fig. 3.1 Funcionamiento de los bots en la compra de entradas	33
Fig. 3.2 Threads de un bot comprando entradas paralelamente	34
Fig. 3.3 Principios de protección de datos.....	38
Fig. 3.4 Derechos de las personas según la GDPR	39
Fig. 4.1 Diagrama de flujo del modelo Vista-Controlador	47
Fig. 4.2 Esquema de creación de solicitudes mediante QR	55
Fig. 4.3 Flujo del envío de notificaciones push a un usuario	56
Fig. 4.4 Pantalla principal de todos los conciertos disponibles.....	59
Fig. 4.5 Pantalla de los detalles de un concierto	60
Fig. 4.6 Pantalla de agregación al carrito	60
Fig. 4.7 Pantalla de confirmación de compra	61
Fig. 4.8 Pantalla para loggearse y pagar mediante el QR	61
Fig. 4.9 Pantalla de los conciertos comprados por un usuario	62
Fig. 4.10 Pantalla de intercambio de entrada con otro usuario	62
Fig. 4.11 Pantalla para confirmar el intercambio mediante el QR	63
Fig. 4.12 Pantalla para comprobar la validez mediante el QR.....	63
Fig. 4.13 Esquema de la compra de una entrada para un concierto	65
Fig. 4.14 Parámetros Selective Disclosure Request	66
Fig. 4.15 Parámetros Selective Disclosure Response.....	67
Fig. 4.16 Parámetros de la verificación generada	67
Fig. 4.17 Esquema del intercambio de una entrada para un concierto	68
Fig. 4.18 Esquema de la verificación de una entrada para un concierto	70
Fig. 4.19 Parámetros de la solicitud de la entrada	71
Fig. 4.20 Parámetros de la respuesta a la solicitud de la entrada	72

LISTA DE TABLAS

Tabla 1.1 Clasificación de datos sensibles y no sensibles	3
Tabla 4.1 Parámetros de la función new Credentials	49
Tabla 4.2 Parámetros de la función createVerification	49
Tabla 4.3 Parámetros de la función createDisclosureRequest.....	50
Tabla 4.4 Parámetros de la función authenticateDisclosureResponse	51
Tabla 4.5 Parámetros de la función getImageDataURI	52
Tabla 4.6 Parámetros de la función paramsToQueryString.....	53
Tabla 4.7 Parámetros de la función messageToURI	53
Tabla 4.8 Parámetros de la función push.send	54

LISTA DE ACRÓNIMOS

TIC	Tecnologías de la Información y la Comunicación
ICT	Information and Communications Technologies
GDPR	General Data Protection Regulation
DNI	Documento Nacional de Identidad
IP	Internet Protocol
SSI	Self-Sovereign Identity
MIT	Massachusetts Institute of Technology
PIN	Personal Identification Number
IDP	Identity Provider
HIPAA	Health Insurance Portability and Accountability Act
COPPA	Children's Online Privacy Protection Act
SMS	Short Message Service
CCPA	California Consumer Privacy Act
ERC	Ethereum Requests for Comments
DID	Decentralized Identifiers
URL	Uniform Resource Locator
JSON	JavaScript Object Notation
HTTP	HyperText Transfer Protocol
ETHR	Ethereum
BLOB	Binary Large Objects
JWT	JSON Web Token
RFC	Requests for Comments
HMAC	Hash-based Message Authentication Code
RSA	Rivest, Shamir y Adleman
ECDSA	Elliptic Curve Digital Signature Algorithm
SHA	Secure Hash Algorithm
IANA	Internet Assigned Numbers Authority
HTML	HyperText Markup Language

XML eXtensible Markup Language
SAML Security Assertion Markup Language
IFZ Institute of Financial Services of Zug
TI&M Technology, Innovation & Management
QR Quick Response
ID Identification
UE Unión Europea
NoSQL Not Only Structured Query Language
UI User Interface
NPM Node Package Manager
REST Representational State Transfer
API Application Programming Interface
MVC Modelo-Vista-Controlador
URI Uniform Resource Identifier
ECDH Elliptic-Curve Diffie–Hellman
ECC Elliptic Curve Cryptography
DH Diffie–Hellman

INTRODUCCIÓN

Desde el inicio de la era de Internet y la consecuente globalización, el usuario ha perdido cada vez más el control sobre sus datos personales, hasta el punto de no asombrarse cuando las grandes compañías tecnológicas conocen con detalle sus gustos, preferencias y su ubicación continua.

A lo largo de estas últimas décadas, la sociedad ha vivido muy centrada en desarrollar nuevas tecnologías sobre Internet, olvidando así un aspecto tan importante como la privacidad o el anonimato. Sin embargo, en los últimos años, a raíz de ver cómo se comercializa con nuestros datos y cómo ciertos fraudes pueden afectar a nuestra vida diaria, todos los usuarios han empezado a tomar un poco más de consciencia de las implicaciones que conlleva registrarse en una nueva red social o buscar información sobre nuestros gustos en cualquier buscador. A pesar de estos problemas, no debemos resignarnos a convivir con ello ya que existen otro tipo de soluciones, aunque actualmente no sean tan populares, para devolverle a los usuarios el control sobre sus datos mientras seguimos proporcionándole los numerosos beneficios y comodidades de Internet, de una manera segura.

El objetivo de este Trabajo de Final de Grado es analizar una de las principales soluciones basada en la identidad digital soberana sobre la blockchain e implementar todas las ventajas de esta tecnología en un caso de uso real, solucionar los múltiples problemas actuales del proceso de compra de entradas para eventos. Veremos cómo esta tecnología posee múltiples beneficios, por un lado, principalmente está pensado para ofrecer privacidad a los usuarios, pero colateralmente, como el usuario gestiona su información personal, ésta no depende de empresas privadas y las mismas, no se ven obligadas a cumplir con las estrictas regulaciones de protección de datos.

Para ofrecer una idea general al lector de los contenidos desarrollados a lo largo de este informe, hacemos ahora una pequeña explicación de la estructura del mismo, este proyecto se estructura en cuatro capítulos. En el capítulo 1, encontramos una introducción al concepto de identidad digital soberana y por qué es tan necesaria. En el capítulo 2, presentamos uPort, una herramienta para la gestión de la identidad digital que usaremos más adelante. En el capítulo 3, se exponen los problemas de los sistemas de venta de entradas actuales, tanto para usuarios como para las empresas. Finalmente, en el capítulo 4, se explica en detalle la implementación de Cibertickets, una web que juntamente con uPort permitiría acabar con gran parte de los inconvenientes actuales, como los bots o el cumplimiento de la GDPR.

CAPÍTULO 1. Identidad digital

Cuando pensamos en identidad una de las primeras ideas que se nos viene a la mente es el DNI (Documento Nacional de Identidad), un sistema centralizado entorno al gobierno de cada país. Sin embargo, nuestra identidad como individuos va más allá de eso. En este primer capítulo, definiremos el concepto genérico de identidad tal y como la conocemos hoy en día, y posteriormente introduciremos conceptos más novedosos como identidad digital e identidad digital soberana, analizaremos los retos de este nuevo paradigma, sus distintos modelos, ventajas e implementaciones posibles en nuestra vida diaria.

1.1 Fundamentos de la identidad

La identidad es un componente fundamental en nuestra vida. En nuestra sociedad actual, prácticamente todas las actividades que hacemos a diario giran en torno a la identidad, desde pagar con una tarjeta de crédito hasta reservar un libro en la biblioteca, pasando por realizar un examen en la universidad. Actualmente, nuestra identidad está alejándose cada vez más del viejo sistema basado en papel y ahora las cosas han cambiado y se dirigen hacia la identidad digital. A pesar de ello, existen muchos problemas relacionados con la identidad, a diario se dan casos de fraudes de suplantación o robo de identidad y el simple gesto de digitalizarlo no implica resolver los problemas intrínsecos, sino todo lo contrario, en la mayoría de casos se pueden incluso agravar. Por ello, es ahora especialmente importante, trabajar en un sistema de identidad digital seguro y al resguardo de las actividades maliciosas ejecutadas por hackers.

La identidad es, sin lugar a dudas, uno de los derechos fundamentales de cualquier persona. De hecho, todo ser humano necesita una identidad para poder vivir y trabajar en nuestra sociedad, es por ello que nadie pone en duda su importancia. La estructura fundamental consiste en los siguientes parámetros:

- Nombre y apellidos
- Fecha de nacimiento
- Nacionalidad
- Número de pasaporte / DNI
- Número de la seguridad social

Sin alguno de estos campos resulta imposible realizar acciones tan básicas como ser propietario de un inmueble, votar, disponer de una cuenta bancaria o acceder a un puesto de trabajo. Es por ello, dada su importancia, que se realiza una clasificación más detallada de los datos de carácter personal dividiéndolos en sensibles y personales según su importancia para la privacidad y por ello deben ser tratados y almacenados con diferentes requisitos.

1.1.1 Datos personales y sensibles

Los datos personales son aquellos que contienen información que permita la identificación directa de un individuo. Cuando hablamos de datos personales consideramos que la información no suele obtenerse de manera aislada, sino que, típicamente, lo que ocurre es que las empresas recopilan y almacenan múltiples fragmentos de información sobre sus clientes, y es cuando unimos toda esta información cuando podemos considerarla como datos personales. Expresado con otras palabras, podríamos decir que una información se convierte en datos sensibles, solo si al unirla con más datos se puede utilizar para identificar a un posible sujeto.

Por otro lado, los datos sensibles (o también llamados confidenciales) son un conjunto específico de “categorías especiales” que deben tratarse con mayor seguridad, ya que se relacionan directamente con los derechos fundamentales de las personas.

En la siguiente tabla podéis ver la clasificación de cierta información en los dos tipos de datos, más adelante veremos cómo estos dos subconjuntos son tratados de manera diferenciada por la ley y cómo su procesado, almacenamiento y distribución también debe hacerse siguiendo criterios específicos [4]:

Sensible	Personal (no sensible)
Opiniones políticas	Cookies
Salud psicológica o física	Dirección del hogar
Origen étnico	Dirección IP (Internet Protocol)
Creencias religiosas	Nombre y apellidos
Afiliación sindical	Dirección de correo electrónico
Datos genéticos o biométricos	Identificadores online

Tabla 1.1 Clasificación de datos sensibles y no sensibles

1.2 Diferenciación entre identidad e identidad digital

Partimos de la base de que todos sabemos quiénes somos y somos capaces de identificarnos a nosotros mismos, en cambio cuando tratamos con terceros éstos requerirán algún tipo de parámetro para reconocernos (desde nuestro nombre, nuestro DNI o cualquier tipo de información única), eso es lo que se conoce actualmente como identidad.

Por otro lado, podríamos decir que una identidad digital es un mecanismo mediante el cual la identidad de un usuario se reproduce por medios digitales. Este nuevo medio nos permite abarcar información de múltiples fuentes, combinando diversos atributos y de esta manera identificar de manera inequívoca a un individuo. Cuando se habla de identidad digital, cualquier característica intrínseca a un individuo se conoce como, "atributo de identidad". Como existen una gran cantidad de variables que pueden definirnos, se considera que no hay límites y pueden ser indicadores tan sofisticados como, por ejemplo:

- Atributos biométricos (huellas dactilares, facciones del rostro o la presión ejercida al usar el dispositivo móvil).
- Atributos gubernamentales.

Finalmente, una pequeña modificación de la identidad digital es lo que se conoce como identidad digital soberana, explicada con más detalle en el siguiente apartado, significa que un usuario u organización tiene la propiedad exclusiva de su identidad, tanto de su expresión digital como analógica y sus principales requisitos [5] son los siguientes:

- **Privada:** Solo el propio usuario debe ser capaz de gestionar y controlar su identidad, es decir, poder decidir qué información comparte y con quién, cuando desea actualizarla o incluso borrarla.
- **Portable:** Mantenerla accesible independientemente del lugar donde el usuario se encuentre.
- **Transparente:** Los sistemas y algoritmos utilizados para administrar y operar con las identidades digitales deben ser abiertos y transparentes.
- **Persistente:** Acompañe a un individuo desde el nacimiento hasta la vejez, pero que, a la misma vez, se pueda modificar o eliminar a lo largo de la vida si es necesario.
- **Personal:** Única de cada individuo, inconfundible.
- **Consentimiento:** El intercambio de datos del usuario con cualquier entidad solo debe realizarse bajo el consentimiento expreso del sujeto implicado.
- **Minimización:** La divulgación de información a terceros se debe limitar a la mínima necesaria para llevar a cabo la acción deseada.

1.3 Identidad digital soberana en profundidad

Gracias a la reciente sofisticación de los dispositivos móviles, los avances en criptografía y la aparición de la tecnología blockchain, disponemos de todas las herramientas necesarias para crear un nuevo sistema de gestión de identidades, basado en el concepto de la identidad soberana, SSI (Self-Sovereign Identity).

La identidad digital soberana [6], es un concepto nuevo y muy innovador, que ha empezado a tener especial importancia en el marco de la tecnología de blockchain. Esta novedosa manera de concebir la identidad personal cambiará completamente la manera de manejar y administrar nuestros datos en un entorno globalizado más conectado día a día.

Una condición previa relevante para la identidad digital soberana es que las identidades digitales no están almacenadas en una plataforma determinada, ni controladas por un operador determinado, sino que permanecen portátiles e interoperables en múltiples plataformas, de modo que las personas son libres de elegir el operador de identidad en el que más confían, y cambiarse de un operador a otro, si así lo desea. De esta manera, un usuario con identidad auto soberana, tiene el poder de controlar quién accede a su información y bajo qué circunstancias, almacenar su identidad y utilizarla para demostrar su identidad de manera online. Últimamente este término tan revolucionario ha atraído la atención de empresas, gobiernos y personas individuales interesadas en la privacidad, intimidad y el acceso a los datos en el ambiente interconectado en el que vivimos.

En una estructura de identidad digital soberana, el individuo que posee la identidad, es siempre dueño total y soberano de su identidad. Los datos de su identidad se almacenan cifrados, normalmente mediante criptografía asimétrica. Así es como el usuario puede distribuir sus datos con terceros de manera segura y evitando una filtración de datos no deseada.

Además, el individuo controla personalmente cada intercambio de información, es decir, cada transacción de datos se ejecuta bajo las reglas que el usuario haya establecido. El dueño de los datos decide:

- Qué información se comparte
- Cuánta información se comparte
- Con quién se comparte

Este control tan detallado es la diferencia primordial entre un sistema de identidad digital soberana frente a un sistema de identidad digital federal o centralizado, que son los más utilizados actualmente.

Finalmente, la distribución de la información, desde el usuario hacia esas terceras partes interesadas, se lleva a cabo usando un sistema descentralizado totalmente. En este tipo de sistemas, cada participante puede aprobar o no mediante consenso si los datos de identidad proporcionados son verdaderos o falsos. De este modo, evitamos la existencia de una autoridad central que en algunos casos puede imponer reglas o prohibir ciertas acciones. Así es como se asegura que los datos otorgados mantienen la integridad y no han sido modificados de ninguna manera durante el proceso.

1.4 Casos de uso de la identidad digital soberana

1.4.1 Identificación en servicios web

La integración de la tecnología de identidad digital soberana con la identificación en servicios webs nos permitiría crearnos cuentas o acceder a ellas apuntando a la identidad que ya tenemos creada. De esta manera los servicios web nos darían acceso prácticamente automático evitando que el usuario deba rellenar continuamente formularios con su identidad para cada servicio al que quiere acceder de manera individual.

1.4.2 Verificación de credenciales educativas y titulaciones

La vinculación de la identidad de un usuario con un certificado educativo seguro criptográficamente que evite falsificaciones. Este sistema (Proyecto de Certificados Digitales [7]) ya se utiliza en el MIT (Massachussets Institute of Technology) y usa la blockchain de Bitcoin [8]. Su funcionamiento se basa en guardar la información de un estudiante y un certificado educativo firmado por la entidad emisora, de esta manera es irrefutable que el estudiante haya obtenido ese certificado y, además, permite comprobar, mediante la firma digital, que realmente ha sido emitido por una autoridad competente y que no se trata de una copia/falsificación.

1.4.3 Identificación segura en sistemas de votación electrónicos

Uno de los problemas actuales que más interés está generando en Gobiernos de todo el mundo es la mejora de los sistemas de votación actuales, dependientes de sistemas centralizados de identificación (como el uso del DNI para identificarte) y la gran utilización de papel. Estas dos importantes dependencias permiten que se lleven a cabo actividades fraudulentas como la manipulación de los votos de los electores, y es por ello que mediante el uso de un sistema de identidad soberana se evitan las duplicaciones y las identidades falsas, lo que nos aporta una mayor transparencia en el proceso de las elecciones.

Adicionalmente, gracias a la criptografía segura, por detrás de estos sistemas descentralizados, nos aseguramos de que no sea posible relacionar al elector con su voto, evitando coacciones o persecuciones por motivos políticos como ocurre en otros países de nuestro entorno o incluso en regiones más pequeñas de nuestro país. Otras aplicaciones más genéricas serían en bancos, eCommerce, videojuegos, gobierno, salud, seguros, préstamos, pagos o incluso viajes.

Es muy curioso ver como los problemas planteados anteriormente, tienen una gran similitud y presentan unos principios de control y seguridad parecidos a los problemas que las blockchains, como por ejemplo Bitcoin, intentan resolver en el plano económico. Este hecho se debe a que el mundo de la identidad digital se ha gestionado bajo la misma visión que el dinero, al fin y al cabo, datos confidenciales e identidades valen dinero. Son muchas las empresas que se dedican a manejarlos y para ellas, nuestros datos son su mercancía, nuestra herramienta para evitarlo es la creación de sistemas descentralizados.

1.5 Tres modelos de identidad digital

Los modelos de identidad digital [9] han ido avanzando a través de tres etapas principalmente desde la llegada de Internet: la primera fase es la de la identidad centralizada o tradicional, la segunda se basa en la identidad federada y finalmente la tercera habla de identidad centrada en el usuario, auto soberana o descentralizada.

1.5.1 Identidad centralizada o tradicional

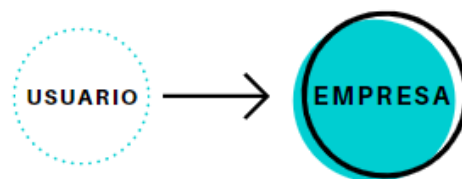


Fig. 1.1 Diagrama de identidad centralizada

En los inicios de la era de Internet, se implementó el sistema de identidad centralizada o tradicional, el más antiguo, simple y utilizado de todos. Consiste en una organización que emite identidades digitales para un usuario con la finalidad de que esta persona pueda utilizar y acceder a un servicio determinado.

La confianza entre el usuario y la organización se establece mediante el uso de secretos compartidos, generalmente en forma de un nombre de usuario y

contraseña, aunque a veces se extiende a otros "secretos" como la fecha de cumpleaños, apellido materno, PIN (Personal Identification Number), etc. Actualmente, los secretos compartidos se incrementan con factores adicionales como tokens físicos o biométricos.

Así es como los datos personales de los usuarios, ya sean compartidos por ellos mismos u obtenidos de otras fuentes, se almacenan en una base de datos de la organización, un proceso que repite cada organización, aplicación o sitio web en el que se desea iniciar sesión. Como resultado, este modelo requiere la creación y la administración de credenciales independientes para cada servicio y el cliente se encuentra en clara situación de dependencia hacia la empresa, por ello podemos verlo representado de una manera más pequeña en la figura anterior.

Ventajas

- Es un modelo ampliamente extendido, fácil de entender y de implementar.
- Ayuda a la organización a gestionar el cumplimiento, la responsabilidad y otros riesgos al mantener los datos internos y al controlar directamente a todos los actores, lo que reduce el riesgo en comparación con depender de un proveedor de identidad externo (Modelo federado descrito en el segundo apartado).
- Al utilizar credenciales independientes para cada servicio se mejora la seguridad y la privacidad, siempre que no se reutilicen los pares de nombres de usuario/contraseña.

Desventajas

- Es un sistema poco escalable o sostenible a medida que el uso de servicios digitales se va extendiendo más y más a cada una de nuestras facetas de la vida cotidiana.
- Una brecha de seguridad en una de estas organizaciones puede ser desastroso, exponiendo los datos personales de miles de usuarios.
- La experiencia de usuario es negativa debido a la necesidad de mantener cientos de credenciales para cada aplicación o servicio, provocando así el olvido o la reutilización de contraseñas.
- La autenticación se produce solo en un sentido, el usuario se autentica frente a la empresa, pero no viceversa, por ello no evitamos estafas de phishing.

1.5.2 Identidad federada

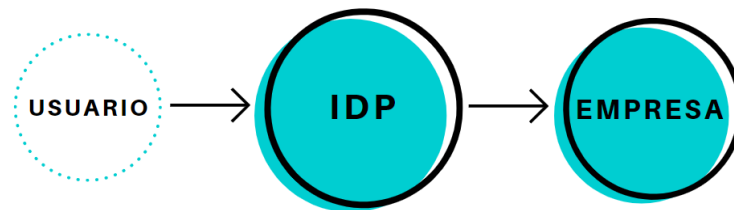


Fig. 1.2 Diagrama de identidad federada

En este modelo, parecido al anterior, se añade una tercera parte de confianza que actúa como un “proveedor de identidad”, un IDP (Identity Provider) entre el usuario y la empresa que ofrece el servicio al cual se desea acceder. El IDP emite las credenciales digitales, proporcionando una experiencia de inicio de sesión único con el IDP que luego puede usarse sin problemas en otros servicios, reduciendo así la cantidad de credenciales separadas que el usuario necesita mantener.

El proceso sería el siguiente: el usuario inicia sesión en el IDP, posteriormente el IDP asocia su inicio de sesión con el inicio de sesión del servicio al que intenta acceder mediante protocolos como OAuth [10] u OpenID Connect [11]. Sin embargo, la relación de confianza entre el usuario y el IDP se mantiene de la misma manera que en la identidad centralizada, generalmente a través de secretos compartidos, y puede fortalecerse con factores adicionales para proporcionar un mayor nivel de seguridad a la organización. La principal diferencia es que los datos de identidad se centralizan ahora en el IDP, además, en este caso tanto el cliente como la empresa también se encuentran en clara situación de dependencia del IDP, por ello podemos verlos representados de una manera más pequeña en la figura anterior.

Un ejemplo común del modelo IDP es el "inicio de sesión con redes sociales" usando su Facebook, Google, Twitter u otra identificación social para acceder a un servicio de terceros. En este caso, estos gigantes tecnológicos actúan como su IDP, esta opción es aceptable en entornos de baja confianza, como el comercio electrónico, pero no en entornos de alta confianza, como los bancos.

Ventajas

- Permite a los usuarios acceder a múltiples servicios con una única credencial, simplificando así la autenticación y reduciendo el número de nombres de usuario/contraseña que debemos memorizar.

Desventajas

- Necesidad de confianza plena en la tercera parte de confianza ya que se encuentra en medio de cualquier interacción entre usuario y compañía.
- Fuerza al usuario a crear dos relaciones, por una parte, con el IDP y por la otra con la empresa que ofrece el servicio al cual intentan acceder.
- La autenticación se produce solo en un sentido, el usuario se autentica frente a la empresa, pero no viceversa, por ello no evitamos estafas de phishing.

1.5.3 Identidad descentralizada o centrada en el usuario

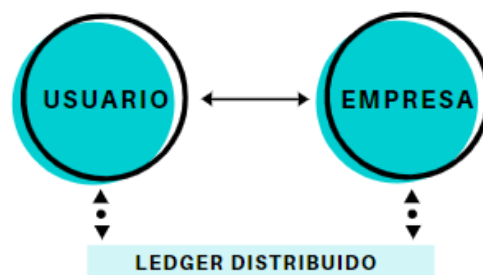


Fig. 1.3 Diagrama de identidad descentralizada

Finalmente, el tercer modelo es un cambio de paradigma por completo, donde ya no existe una relación de dependencia entre usuario, empresa e IDP, sino que se establece un modelo con dos partes, el usuario y la empresa, ahora considerada una entidad igualitaria, por ese motivo se simbolizan ambas partes del mismo tamaño en la figura anterior.

El funcionamiento de este modelo se basa en la existencia de un “monedero digital” que contiene credenciales digitales verificables y firmadas que pueden demostrar criptográficamente cuatro aspectos:

1. Quién emitió esa credencial
2. A quién se emitió esa credencial
3. Si ha sido alterada desde que se emitió
4. Si ha sido revocada por el emisor inicial

La principal diferencia en este patrón es que las credenciales pueden ser emitidas y firmadas digitalmente por cualquier persona u organización y así, utilizarse en cualquier lugar en el que se confíe. El sistema de identidad descentralizada es lo suficientemente robusto, incluso para entornos de alta confianza, como las finanzas, gobierno o salud. Las organizaciones pueden optar por confiar solo en las credenciales que ellas han emitido, credenciales

emitidas por otros, o alguna combinación, de acuerdo con sus necesidades de seguridad y cumplimiento según el servicio que ofrezcan.

Ventajas

- Los secretos compartidos de los sistemas anteriores pueden ser remplazados por seguridad criptográfica.
- La autenticación es mutua, cuando una entidad nos emite una credencial, podemos saber exactamente de quién se trata y evitar estafas de phishing.
- Simplifica el cumplimiento de regulaciones como la GDPR, HIPAA (Health Insurance Portability and Accountability Act), COPPA (Children's Online Privacy Protection Act) [12] u otras normativas eliminando a intermediarios. Mediante estas leyes, los gobiernos de todo el mundo buscan preservar la privacidad de los ciudadanos. Con la ley de HIPAA se garantiza el anonimato de la información médica, con la ley COPPA se protege la privacidad de los niños y con la GDPR se otorga a los usuarios más control sobre sus datos.

Desventajas

- Al tratarse de un sistema novedoso, se requieren modificaciones en los sistemas para poder emitir credenciales y verificarlos, actualizar las interfaces de usuario para eliminar las peticiones de nombres de usuario/contraseñas.
- Las empresas dejarían de disponer del correo electrónico de sus usuarios para poder comunicarse con ellos a modo promocional, lo que a su vez sería una ventaja para el usuario que vería reducida la cantidad de spam que recibe.

1.6 Problemas de los modelos: Tradicional e IDP

Los sistemas centralizados o federados de gestión de identidad actuales presentan serios problemas [13] que amenazan continuamente la seguridad, conocerlos y tener una breve noción de ellos ayudará a evitarlos en los sistemas descentralizados del futuro próximo:

- **Uso de contraseñas poco seguras:** Se trata de uno de los conflictos más comunes, ya que la mayoría de aplicaciones basan su sistema de autenticación en una contraseña para poder acceder al sistema. Si no aplicamos políticas activas de contraseña, obligando a nuestros usuarios a cambiarla periódicamente, quedarán obsoletas. Por parte del usuario, sus responsabilidades son utilizar contraseñas complejas, que impidan

que los hackers las descubran fácilmente y evitar utilizar la misma combinación para múltiples aplicaciones, aumentando así el riesgo.

- **Falta de inclusividad:** Actualmente la identidad es expedida por organismos centralizados como los diferentes estados del mundo que pueden poner ciertas normas (éticas o no éticas) antes de emitir una identidad a un ciudadano. Debido a esta situación, existen 1100 millones de personas en el mundo que carecen de identidad, quedando excluidas de manera automática de oportunidades políticas, económicas o sociales debido a los sistemas de identificación actuales, considerados arcaicos e inaccesibles para el 14,5% de la población mundial.
- **Proceso de registro repetitivo:** Existen múltiples plataformas separadas e independientes una de las otras, esto provoca la necesidad de registrarse muchas veces para cada servicio al que queremos acceder. Por ejemplo, al abrir una cuenta bancaria, contratar un seguro del hogar, viajar al extranjero, pedir un préstamo... No solo causa una gran pérdida de tiempo para el usuario, sino que también, este sistema abre un sinfín de puntos de acceso a nuestra información controlado por empresas y entidades distintas que pueden tener numerosas brechas de seguridad.
- **Mal uso de la información personal:** Actualmente, todos los servicios y aplicaciones online requieren un sistema de registro, pero se desconoce casi por completo el mal uso y la manipulación que posteriormente reciben esos datos. Existen numerosos casos donde se ocultó una recogida masiva de información personal con el pretexto de realizar una encuesta, puramente informativa y académica, es decir, los datos identificativos están fuera del control del usuario. Por ese motivo son necesarios estrictos protocolos y medidas de seguridad para limitar el uso que las compañías hacen de nuestra identidad.
- **Protocolos de autenticación débiles:** La autenticación basada en contraseñas fue solo el primer paso, ya que no puede garantizar por sí misma los niveles de seguridad necesarios. Debido a las carencias, se empezó a implementar la autenticación basada en dos factores, añadiendo así una capa extra de defensa mediante el uso de confirmación por email o SMS (Short Message Service). Sin embargo, los hackers han empezado a innovar en técnicas que consigan evadir esta doble autenticación, por ejemplo, falsificando estos SMS o emails para conseguir que el usuario desvele sus credenciales. Es aquí donde entra en juego la autenticación multifactor, basada en parámetros biométricos, tokens, certificados digitales... A pesar de ser lo más seguro hoy en día, puede ser un proceso algo tedioso y que empeore la experiencia del usuario.

1.6.1 Pérdida del control de nuestros datos

Para entender cómo hemos llegado hasta aquí, es importante mirar hacia atrás. En la década de 1990, emerge la World Wide Web y surge la posibilidad de almacenar nuestros libros y nuestras investigaciones de manera online para poderlos recuperar fácilmente, así fue como, poco a poco, se crearon: enciclopedias y bibliotecas enteras. Poco a poco, los navegadores de Internet mejoraron, y evolucionamos desde la "Web de sólo lectura" a la "Web de lectura y escritura", también conocida como Web 2.0.

Publicar, tuitear, dar me gusta y comentar son ahora algunas de las interacciones básicas de Internet. Todos los días, los usuarios se convierten en creadores de billones de datos, simplemente diciéndole a una empresa, como Facebook, lo que "les gusta" un contenido en específico. Y en silencio, casi imperceptiblemente, a medida que la Web 2.0 crecía, los usuarios perdieron el control de su identidad digital [14], que ahora se encuentra dispersa por toda Internet, en todas las aplicaciones y servicios con los que interactuamos, ya sea en casa o en el trabajo. Esta situación ha llegado aún más lejos, ahora mismo los gigantes tecnológicos están rastreando el comportamiento de los usuarios online y offline, sin su consentimiento o sin conocimiento activo.

La mayoría de nosotros hemos perdido la pista de la incontable cantidad de formularios de registro que hemos rellenado, donde hemos proporcionado nuestra información de identidad personal con una ligera suposición de que se puede confiar en el proveedor y de que este protegerá nuestros datos. Pero estas proliferaciones incontroladas de nuestros datos unido a la incapacidad de los proveedores de servicios para almacenar adecuadamente los elementos críticos de nuestra identidad han demostrado una y otra vez que son la causa principal de las frecuentes violaciones y brechas de seguridad. Los proveedores de identidades sociales como Facebook, Twitter y LinkedIn se apresuraron a abordar este problema centralizando la identidad y presentándola al servicio solicitante con el consentimiento del usuario, el conocido "Registro con Facebook, Google...". Algunos de estos proveedores de identidades centralizados se han convertido en objetivos fáciles para que los piratas informáticos obtengan información de los usuarios de una fuente en lugar de tener que buscar a través de múltiples proveedores. Estas infracciones no solo afectan a nuestra vida social, profesional y financiera, sino que también nos han llevado a aceptar la nueva realidad donde la privacidad ya no es una norma social o una elección personal.

Por otro lado, los proveedores también están sujetos a normas de responsabilidad de las diferentes jurisdicciones geográficas al almacenar los datos de los usuarios y están obligados a cumplir con varios requisitos de cumplimiento de la privacidad, como HIPAA, GDPR, CCPA (California Consumer Privacy Act), etc. Esto hace que el problema de identidad sea un arma de doble

filo, difícil de resolver, tanto para el usuario como para el proveedor. Es en este punto, donde surge la necesidad de descentralizar la identidad, eliminar estos proveedores de servicios centralizados si queremos volver a obtener el control de nuestros datos, así como liberar a ciertas empresas de la gran responsabilidad que supone el almacenamiento de datos de usuario.

1.7 Necesidad de descentralización en la gestión de la identidad

Los principales motivos [15] para implementar un sistema de identidad soberana descentralizado frente a los actuales sistemas centralizados son:

- Seguridad de la información

En primer lugar, los sistemas de gestión de identidad digital actuales son principalmente centralizados, donde existen bases de datos centralizadas que contienen enormes cantidades de registros de identidad (miles de millones). Por la importancia de su información y por su gran tamaño, estas bases de datos son objetivos directos de mucho valor para los agentes de amenazas. Además, la retribución por un ataque con éxito crece exponencialmente con el número de registros que almacene esa base de datos, podríamos decir que cuantas más identidades tenga, más valiosa y más inconsistente es.

En segundo lugar, los registros de identidad que almacenan son sencillos de robar y utilizar posteriormente para suplantar la identidad. Asimismo, una única base de datos centralizada implica que, si esa base de datos o los servidores relacionados son atacados, un gran número de personas se verían perjudicadas. Un estudio reciente, llevado a cabo por ForgeRock [16], muestra que la información personal es el principal objetivo de las violaciones cometidas por los hackers, suponiendo el 97% de las infracciones en 2018. A pesar de los esfuerzos de los gobiernos y las empresas para aumentar la seguridad, aproximadamente 2.800 millones de datos de usuarios se filtraron en 2018, provocando unas pérdidas de unos 654.000 millones de dólares.

En último lugar, normalmente estas bases de datos son propiedad de una organización, que consideramos una tercera parte de confianza y confiamos en ella para que se encargue de guardar nuestros datos, pero también puede tener brechas de seguridad interna que acaben implicando filtraciones de datos. Esta situación ha ocurrido en varias ocasiones, por ejemplo, con Facebook que gestiona identidades y datos privados de los usuarios bajo una estructura centralizada y por ello ha sido víctima de fallos de seguridad y hackeos que han provocado el robo y la filtración de información.

- Evadir el control por parte de oligopolios

Con la estructura actual, existen unas pocas empresas que almacenan en sus bases de datos las identidades de prácticamente todas las personas del planeta, formando así un oligopolio. Estas empresas, en un futuro, podrían cobrar grandes cantidades de dinero para poder acceder a los datos y esto crearía grandes barreras de entrada a nuevas empresas que quieran participar en este mercado. En este escenario, existiría una ausencia de presión competitiva, precios elevados y una reducción de la innovación, por ello, los sistemas de identidad soberana descentralizados son la mejor lucha contra esta situación, ya que nos permitiría:

1. Existencia de infraestructuras descentralizadas, seguras y sin censura para el uso de identidades
2. Control absoluto de los datos por parte del propio usuario
3. Mayor privacidad para los usuarios
4. Aumentar la innovación de los servicios de identidad

- Restricciones de acceso

En estos sistemas centralizados, los datos iniciales son propiedad de una persona y la representan. En cambio, esta persona se suele enfrentar a limitaciones a la hora de acceder o manipular estos datos digitales, ya que existe un tercero que se encarga de limitar lo que la persona puede hacer con su información.

Esta situación sucede cuando disponemos de una identidad digital almacenada en una empresa y queremos eliminarla de la base de datos. La organización puede darnos constancia de que ha borrado la información correspondiente al registro de identidad, pero, por otro lado, ha guardado una copia de seguridad de esa información por motivos de seguridad empresarial y monitoreo. El resultado final es que no ha sido suprimido de manera definitiva y, por tanto, es la prueba irrefutable de que no tenemos acceso total a nuestros propios datos ni a la gestión de ellos.

1.8 Retos del sistema de gestión de identidad digital

Cada nueva tecnología que queremos implementar conlleva una serie de desafíos a los que debemos enfrentarnos si queremos que finalmente esa nueva técnica sea aplicable en la vida cotidiana de todos los individuos.

En este caso, un sistema adecuado de gestión de identidad digital supone los siguientes retos:

1. Debe ser lo suficientemente rápido para garantizar que un usuario averigüe su consulta a tiempo.
2. Necesita disponer un gran nivel de seguridad, a pesar de la existencia de ataques y aplicaciones con grandes brechas de seguridad

Sin embargo, hasta el momento han existido numerosas dificultades para superar ambos retos de manera simultánea. Además, debido a la falta de recursos para solucionar problemas de seguridad, el panorama actual no ha evolucionado tan rápido como lo han hecho otras tecnologías.

CAPÍTULO 2. uPort

Una vez hemos adquirido los conceptos teóricos sobre la identidad digital soberana, nos preguntaremos, ¿cómo podemos implementarlo en nuestros proyectos reales de programación? La respuesta a esta pregunta la resolveremos en este segundo capítulo donde hablaremos de uPort, una herramienta que permite gestionar identidades descentralizadas y puede ser integrada en cualquier entorno, tanto en la parte del cliente como del servidor. Además, analizaremos sus beneficios, el mecanismo de funcionamiento y la prueba piloto que se lleva a cabo en Zug, Suiza.

2.1 ¿Qué es uPort?

uPort [17] fue desarrollado por ConsenSys y es un sistema seguro, fácil de utilizar y descentralizado de identidad digital soberana basado en Ethereum [18]. Esta blockchain es una plataforma pública y distribuida de software, de código abierto y basada en Blockchain (un sistema descentralizado y autónomo) que permite a los desarrolladores crear e implementar aplicaciones descentralizadas.

Todo el sistema Ethereum está soportado por un sistema global de "nodos". Los nodos son voluntarios que descargan toda la cadena de bloques de Ethereum a sus escritorios y hacen cumplir plenamente todas las reglas de consenso del sistema, manteniendo la red honesta y recibiendo recompensas a cambio. Esas reglas de consenso, así como muchos otros aspectos de la red, son dictadas por "Smart Contracts". Están diseñados para realizar automáticamente transacciones y otras acciones específicas dentro de la red con partes en las que no se confía necesariamente. Las condiciones que ambas partes han de cumplir están programadas en el contrato, una vez finalizado se desencadena una transacción u otra acción específica.

El modelo de identidad presentado por uPort funciona sobre una infraestructura de criptografía de clave pública, también conocida como asimétrica. Es el método criptográfico que usa un par de claves para el envío de mensajes cifrados. Cada persona o entidad posee un par de claves. Una clave es pública y se puede entregar a cualquier persona, la otra clave es privada y el propietario debe guardarla de modo que nadie tenga acceso a ella. Además, los métodos criptográficos garantizan que esa pareja de claves solo se puede generar una vez, de modo que se puede asumir que no es posible que dos personas hayan obtenido casualmente la misma pareja de claves. Mediante estos mecanismos criptográficos, uPort devuelve la propiedad de la identidad al individuo al permitir que los usuarios registren su propia identidad en Ethereum, envíen y soliciten credenciales, firmen transacciones y administren de manera segura claves [19]

y datos en su sistema de identidad abierto, completamente independiente de terceras partes centralizadas.

Una de los principales objetivos de uPort, es facilitar la usabilidad e interacción con la blockchain Ethereum para hacer que la identificación digital sea accesible para todos, independientemente de los conocimientos técnicos de los usuarios.

Para conseguir este objetivo, la tecnología de uPort contiene principalmente dos componentes:

- **Librerías para desarrolladores:** Mediante el uso de las librerías de desarrolladores es cómo los desarrolladores de aplicaciones de terceros pueden integrar en sus aplicaciones el soporte para uPort.
- **Aplicación móvil:** La aplicación móvil de uPort contiene las claves del usuario, que se puede recuperar en caso de pérdida del dispositivo.

2.2 Beneficios de uPort

2.2.1 Requerimientos de infraestructura bajos

Puesto que estamos basándonos en una instancia pública de Ethereum, no es necesaria la implementación de servidores o nodos propios ni la creación y mantenimiento de una base de datos compleja que almacene las credenciales de los usuarios. Además, gracias a las facilidades que nos proporciona uPort [20][15], no es necesario que el usuario disponga de conocimientos para interactuar con la blockchain de Ethereum o sobre cómo comprar su propio gas, sino que todo esto es gestionado por el servidor de gas de uPort.

2.2.2 Disminución del riesgo de seguridad

Debido a la inexistencia de estos servidores centralizados y a la distribución de su propia identidad a cada ciudadano, se reduce la susceptibilidad ciberataques o a robos de datos, en caso de que haya una vulnerabilidad en un sistema concreto.

2.2.3 Facilita el cumplimiento del GDPR

Cualquier aplicación que utilice este sistema de autenticación cumple completamente con la legislación de la GDPR, ya que asegura que la empresa verifica y obtiene la mínima cantidad de información necesaria para un uso específico, reduciendo así la responsabilidad de los proveedores de servicios, que almacenan simplemente los datos que utilizan.

2.2.4 Costes reducidos

En comparación con otros sistemas de identidad, la implementación de uPort es actualmente gratuita ya que se basa en una red de prueba (testnet). Es cierto que una vez se desee migrar la aplicación a una red pública el gas tendría un coste de entre 1\$ y 10\$.

2.2.5 Escalable

Con esta tecnología, las identidades digitales se emiten de manera off-chain, esto implica que no existen costes de transacciones on-chain, por tanto, es mucho más escalable cuando queremos tratar con grandes volúmenes de población, ya que de otra manera el coste económico sería inasumible para la mayor parte de implementaciones.

2.3 Creación de nuestra identidad

En sus inicios, uPort requería interacción continua con la blockchain Ethereum lo que restaba escalabilidad al proyecto, por ello a mediados de 2018 se llevó a cabo una gran reestructuración de su arquitectura, que continúa teniendo su base en Ethereum pero solo un conjunto reducido de transacciones requieren interacción con la blockchain. En nuestro caso de uso, que podéis ver explicado más adelante, ninguna operación requiere interacciones con la blockchain y los principales beneficios de ello son:

- Creación de identidad de manera instantánea
- La verificación del número de teléfono ya no es necesaria
- Al instalarse la app, se crea automáticamente un par de claves que se traduce en una identidad anónima para el usuario.

Esta nueva arquitectura está basada en el Ethereum Standard ERC1056 (Ethereum Requests for Comments). Por ello, a la hora de crear un nuevo usuario, en lugar de registrar uno o varios Smart Contracts en la cadena de bloques, todo lo que debe hacer ahora es crear un par de claves de manera local. Este proceso no requiere ningún tipo de transacción y, por tanto, podemos decir que es un método independiente de la blockchain.

El proceso de creación del par de claves es tan rápido y fluido que se podrían crear millones de identidades en un solo día. Esto significa que finalmente se pueden admitir aplicaciones a gran escala, como lo son los proyectos de identidad nacional.

2.3.1 DIDs

Hoy en día, en la mayoría de las soluciones de identidad digital, los DID (Decentralized Identifiers) [21] se basan fundamentalmente en la criptografía de clave pública. Un DID es generado a partir de una clave pública, mientras que la clave privada correspondiente es guardada en secreto por el titular del DID. Esto permite que el propietario del DID tenga un control completo sobre su identidad.

Para comprender mejor que son los DIDs, una buena analogía sería pensar que son como una URL (Uniform Resource Locator), aplicado al campo de la identidad digital. Del mismo modo que escribir "https://www.google.com/" en el navegador te lleva a la página de inicio, escribir el DID de un usuario determinado permite extraer un documento, habitualmente en formato JSON (JavaScript Object Notation), que contiene una serie de "reclamos" sobre la identidad de ese usuario.

Antes de continuar hablando de los DIDs es importante conocer su estructura y todos sus campos, a continuación, podemos observar un ejemplo:

```
did:ethr:0xb50df7f85f0c812e99c3f95f209fb6a2d8e934b5
```

Esquema Red Identificador único de la red

Fig. 2.1 Formato de un DID

Si desglosamos un DID en sus principales partes, el primer componente, el esquema, denota que se trata de una identidad digital, al igual que HTTP (HyperText Transfer Protocol) denota el protocolo utilizado en la URL de un sitio web. El siguiente término, en este ejemplo ETHR (Ethereum) se conoce como la red, que se refiere a la red en la que se encuentra, por ejemplo, Ethereum. Por último, tenemos un identificador único, que es exclusivo para un usuario en esa red determinada.

2.3.2 Vincular datos extra con un DID determinado

Ahora que ya tenemos claro el concepto de identidad y como la gestiona uPort mediante las claves y los DIDs. El siguiente paso es analizar cómo relacionar una identidad con unos datos específicos, que en nuestro caso de uso se tratará de entradas de conciertos, pero todo a su debido tiempo.

Una identidad se puede vincular criptográficamente a un conjunto de datos almacenados de manera off-chain, es decir independiente de la blockchain.

Cada identidad es capaz de almacenar el hash de un conjunto de datos BLOB¹ (Binary Large Objects), donde se almacena esta relación de manera segura lo veremos en el apartado siguiente con más detalle. Las identidades, por sí mismas, son capaces de actualizar este conjunto de datos, como, por ejemplo:

- Agregando una foto de perfil o a un amigo
- Otorgando a otras entidades/aplicaciones permiso temporal para leer o escribir datos específicos asociados a su identidad.

Una vez ya conocemos como vincular el DID de un usuario con un conjunto de datos relacionado con él, la pregunta es, ¿dónde almacena los datos del usuario si no se encuentran ni en un servidor ni en una blockchain?

2.4 Almacenamiento e intercambio de datos off-chain

A lo largo de todo este capítulo hemos mencionado el concepto 'off-chain' y de la independencia de blockchain a la hora de implementar uPort, a continuación, explicaremos cómo es posible almacenar, compartir y recuperar datos en un sistema off-chain [22].

1. Almacenamiento de los datos de los usuarios

Los datos off-chain se almacenan en una entidad administrada por el usuario, que se puede alojar localmente en un teléfono inteligente, en un hub de identidad privado o en ambos para mayor redundancia. Un hub de identidad privado es un agente alojado en la nube de manera segura y autónoma que puede almacenar los datos personales del usuario, mediante estos sistemas solo las personas o aplicaciones con explícito permiso por parte del usuario podrán acceder a esos datos personales. Sin embargo, en la versión actual de uPort, los datos del usuario se almacenan localmente dentro de la aplicación móvil uPort, que el usuario puede usar para autenticarse en servicios de terceros.

2. Compartición los datos de los usuarios entre diferentes aplicaciones

Debido a que los datos privados se almacenan localmente en la aplicación uPort del usuario, las aplicaciones no pueden simplemente leer la blockchain pública para descubrir información sobre una identidad. En su lugar, deben solicitar la información privada al usuario directamente.

¹ Los BLOB son elementos utilizados en las bases de datos para almacenar datos de gran tamaño que cambian de forma dinámica.

La aplicación de uPort proporciona una interfaz de consentimiento simple para que aplicaciones de terceros soliciten los datos privados de los usuarios y los usuarios aprueben o rechacen esta solicitud. Esta interfaz es conocida como 'Solicitud de divulgación selectiva' (Selective Disclosure Request), y ofrece a los usuarios un control completo sobre sus datos de identidad.

Estos datos adquieren la forma de JWT (JSON Web Tokens) firmados y tienen campos específicos diseñados para su uso con clientes de uPort que vienen descritos en las especificaciones de uPort.

3. Copia de respaldo y recuperación de los datos

uPort ofrece un hub privado de respaldo de datos de usuario, llamado Caleuche. Caleuche ofrece a los usuarios de uPort la capacidad de almacenar copias cifradas simétricamente de sus datos privados en un servidor. Su seguridad reside en que ningún usuario, ni siquiera el dueño, podrá leer nunca los datos almacenados en este servidor y no se almacenaran copias adicionales. La aplicación móvil será la encargada de comunicarse con este hub para recuperar la información perdida.

En cualquier caso, los usuarios tienen la opción de darse de baja de este servicio de respaldo. Optar por la copia de seguridad de datos privados pone en cierto peligro los datos de identidad del usuario, y a su misma vez, perder su teléfono inteligente también significa perder sus datos de identidad, ya que no hay copia de seguridad. Pero esa es, en última instancia, una decisión del usuario.

2.4.1 JWT

2.4.1.1 ¿Qué es?

JWT [23] es un estándar abierto, pertenece al RFC (Requests for Comments) 7519 que define una forma compacta y auto contenida de transmitir información de forma segura entre varias entidades a partir de un objeto JSON. Esta información puede verificarse y ser fiable ya que está firmada digitalmente. Los JWTs se pueden firmar usando un secreto pre compartido, usando el algoritmo HMAC (Hash-based Message Authentication Code) o mediante un par de claves pública/privada usando RSA (Rivest, Shamir y Adleman) o ECDSA (Elliptic Curve Digital Signature Algorithm).

Aunque los JWTs se pueden simplemente cifrar para proporcionar también confidencialidad entre las partes, nos centraremos en los tokens firmados que es el mecanismo que utilizaremos en el proyecto. Los tokens firmados pueden verificar la confidencialidad e integridad de los datos que contiene, mientras que los tokens cifrados solo ocultan esos datos de otras entidades en caso de ser interceptados. Cuando los tokens se firman utilizando pares de claves públicas / privadas, la firma también certifica que solo la parte que posee la clave privada

es la que la firmó, de esta manera se comprueba que los datos que se proporcionan son verídicos.

2.4.1.2 Utilidades y posibles implementaciones

Existen diferentes escenarios y casos de uso donde los JWTs son muy útiles y ampliamente utilizados:

- **Autorización:** Este es el escenario más común donde se usan los JWTs. Una vez que el usuario ha iniciado sesión se le proporciona un JWT y cada solicitud posterior lo incluirá, esto le permitirá acceder a rutas, servicios y recursos que están permitidos con ese token y que están prohibidos para el resto de usuarios. Algunas de sus principales ventajas son la pequeña sobrecarga a la hora de trabajar con ellos y su capacidad para usarse fácilmente en diferentes entornos.
- **Intercambio de información:** Por otro lado, son una buena manera de transmitir información de forma segura entre las partes. Debido a que los JWTs pueden firmarse, por ejemplo, utilizando pares de claves públicas/privadas, se puede estar seguro de que los remitentes son quienes dicen ser. Además, como la firma se calcula utilizando el encabezado y la carga útil, también puede verificar que el contenido no haya sido alterado durante el envío, asegurado la integridad.

2.4.1.3 Estructura

En su forma compacta, un JWT consta de tres partes separadas por puntos (.):

- **Encabezamiento:** El encabezado generalmente consta, a su vez, de dos partes: el tipo de token, que es JWT, y el algoritmo de firma que se utiliza, como por ejemplo HMAC SHA256 (Secure Hash Algorithm) o RSA. Este JSON con dos campos se codifica en Base64 para formar la primera parte del JWT.

En la siguiente imagen, podemos ver el formato decodificado de uno de los JWT utilizados en la implementación de Cibertickets, en este caso el algoritmo de firma utilizado es ES256K-R, que explicaremos más adelante.

```
{  
  "typ": "JWT",  
  "alg": "ES256K-R"  
}
```

Fig. 2.2 Formato del encabezamiento de un JWT

- **Carga útil (payload):** La segunda parte del JWT es la carga útil, que contiene los reclamos, también conocidos en inglés como claims. Los reclamos contienen una serie de parámetros con datos de la entidad e información adicional del propio reclamo. Existen tres tipos de: reclamos registrados, públicos y privados.
 - **Registrados:** Se trata de un conjunto de parámetros predefinidos que no son obligatorios, pero sí recomendables, para poder obtener un conjunto de reclamos útiles e interoperables. Algunos de estos campos los podemos ver en la figura inferior y son: iss (emisor) o exp (tiempo de vencimiento).
 - **Públicos:** Estos parámetros pueden ser definidos a voluntad por aquellos que usan los JWTs. Pero para evitar colisiones, deben definirse siguiendo el Registro de JWT de IANA (Internet Assigned Numbers Authority).
 - **Privados:** Estos son parámetros personalizados creados para compartir información entre dos o más entidades que previamente lo hayan acordado y, por tanto, no se consideran ni registrados ni públicos. Este es el caso de los campos 'requested, permissions, callback o type' que han sido acordados por uPort para intercambiar JWT entre la aplicación móvil y un servidor, en este caso el de Cibertickets.

El JSON con todos estos campos se codifica en Base64 para formar la segunda parte del JWT.

```
{
  "iat": 1597051341,
  "exp": 1597051941,
  "requested": [
    "name"
  ],
  "permissions": [
    "notifications"
  ],
  "callback": "https://5aada3dae644.ngrok.io/callback",
  "type": "shareReq",
  "iss":
  "did:ethr:0x623fe050f3873b5c540ed97a8a6006ebc34db1d1"
}
```

Fig. 2.3 Formato del payload de un JWT

- **Firma:** Para crear la parte de la firma, se debe coger el encabezado codificado, la carga útil codificada, un secreto, el algoritmo especificado

esta interfaz, conocida como 'Flujo de divulgación selectiva' (Selective Disclosure Flow) el cliente acepta (clickando el botón de login) o bien cancela el acceso.

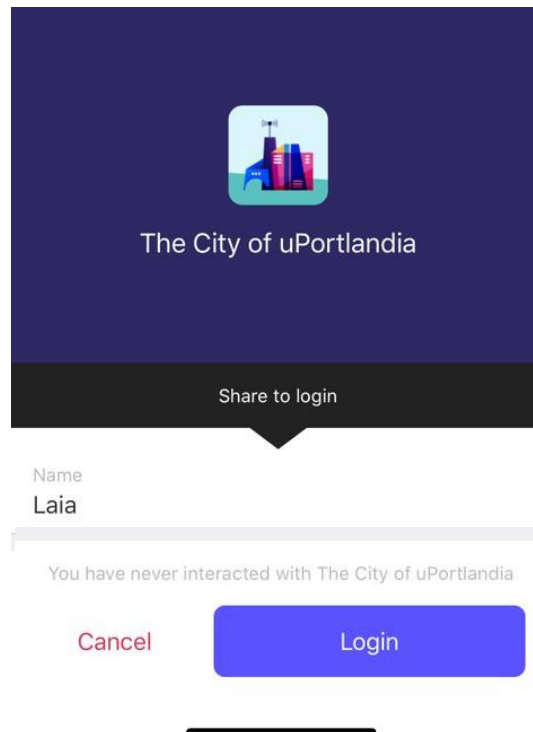


Fig. 2.6 Interfaz de usuario para el flujo de divulgación selectiva

Además, podemos observar en la figura superior, en la parte inferior de la pantalla, cuántas veces hemos interactuado con esa aplicación, de esta manera nos permite conocer si en otras ocasiones hemos confiado en ella.

En el diagrama de flujo siguiente, distinguimos dos partes:

- Solicitud de los datos necesarios (Selective Disclosure Request)
- Respuesta de la solicitud (Selective Disclosure Response)

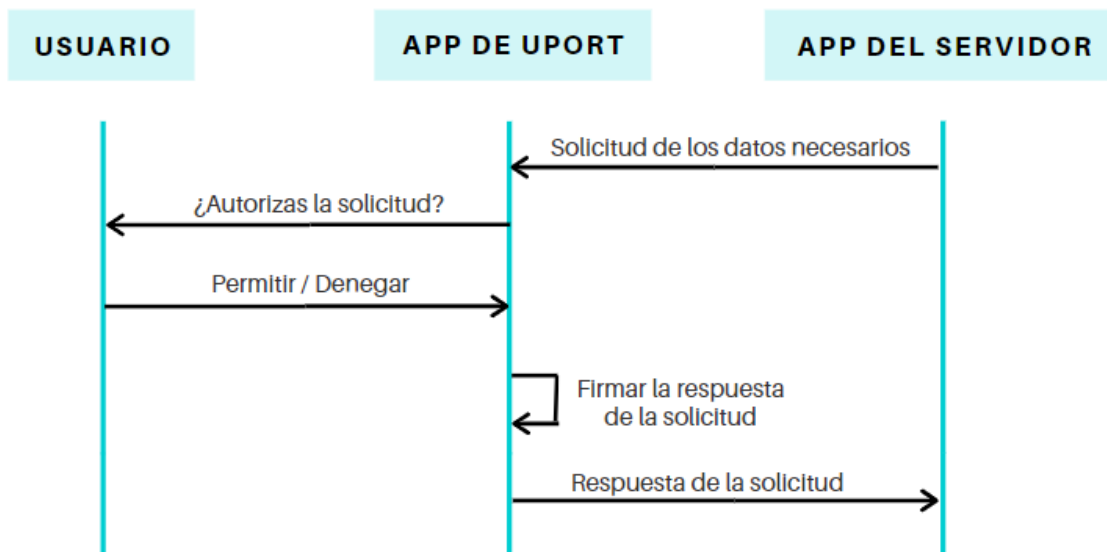


Fig. 2.7 Diagrama de flujo del proceso de divulgación selectiva [24]

2.6 Implementación real de uPort en Zug

2.6.1 Contexto social

En la mayoría de las democracias actuales, los ciudadanos son llamados a votar para unas pocas elecciones: presidenciales, parlamentarias o locales. El representante que eligen toma decisiones políticas en su nombre. Esto es lo que llamamos democracia indirecta.

En cambio, existe otro modelo democrático, nombrado democracia directa, que es el sistema implantado en Suiza, que los lleva a las urnas alrededor de cuatro veces al año para votar sobre una gran cantidad de asuntos relevantes para su cantón en particular, por ejemplo, fumar en restaurantes, financiar museos y extender rutas de autobuses locales. Aunque esta es la forma de gobierno más democrática, también crea un proceso engorroso, costoso y que consume mucho tiempo. Es por ello, que fueron uno de los primeros países en empezar a explorar el funcionamiento de las identidades digitales basadas en blockchain para poder mejorar así la eficiencia, la seguridad, la fiabilidad y la accesibilidad a un conjunto de servicios ofrecidos por su gobierno. De esta manera, usando las identidades digitales se puede prescindir de agentes que se encarguen de contar los votos después de cada proceso electoral.

2.6.2 Funcionamiento

En este contexto, fue la ciudad de Zug quién apostó por uPort para crear la primera implementación real de un proyecto de identidad soberana en el mundo que funciona sobre la blockchain de Ethereum e impulsado por el gobierno autónomo. El gobierno [20] trabajó en conjunto con la ciudad de Zug, el IFZ (Institute of Financial Services of Zug) de la Universidad de Lucerna, el integrador TI&M (Technology, Innovation & Management) para la plataforma y Luxoft para implementar la votación.

En el verano de 2017, lanzaron un programa piloto para generar identificaciones de residentes en la blockchain pública Ethereum. Después del programa piloto, Zug lanzó oficialmente el programa en noviembre de 2017.

Mediante este sistema, la identidad digital no solo permite una mayor confianza entre los ciudadanos y el gobierno, sino que también abre nuevas y significativas oportunidades para mejorar las interacciones digitales entre ellos. En el caso de Zug, las primeras elecciones donde se introdujeron las votaciones electrónicas mediante identidades digitales fue en primavera de 2018.

2.6.3 Experiencia del usuario

A continuación, se detallan los pasos necesarios [25] para obtener una identidad digital soberana en Zug:

- I. Los residentes de Zug descargan la aplicación de uPort (disponible tanto en la App Store como en la Play Store) y se crean una cuenta.
- II. La aplicación de uPort genera una clave privada única que representa la identidad del usuario en su dispositivo móvil y se encarga de desplegar dos Smart Contracts en la red de Ethereum, que actúa como agente de identidad del usuario (*inicialmente las identidades fueron desplegadas en la red pública de prueba Rinkeby, pero posteriormente se incluyó el soporte en la Main-net*).
- III. El residente tiene la oportunidad de hacer una copia de seguridad de su clave privada, permitiéndole recuperar el acceso a su identidad en caso de que pierda el acceso a su teléfono, por pérdida o robo. Con esta configuración, el residente posee el control completo de su identidad y de todos sus datos asociados, además es imposible que pierda el acceso en caso de que pierda su clave privada.
- IV. Posteriormente, el residente visita el sitio web de Zug para registrarse escaneando un código QR (Quick Response) que le permite interactuar con la plataforma de gobierno electrónico de Zug por primera vez. Cabe

destacar que la ciudad de Zug tiene su propia identidad en la red pública de Ethereum lo que le permite firmar y verificar datos.

- V. El residente ingresa su fecha de nacimiento y número de pasaporte en el sitio web de Zug. La solicitud se firma criptográficamente y se envía a la ciudad como una nueva petición de solicitud de Zug ID (Identificación).

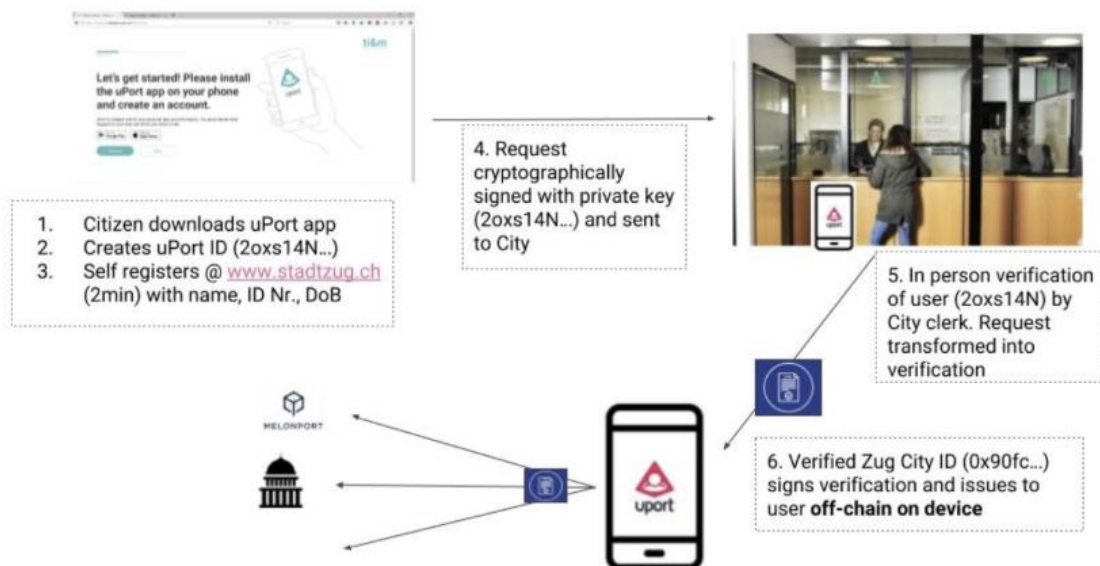


Fig. 2.8 Esquema del funcionamiento de la identidad digital en Zug

- VI. El residente debe personarse en la oficina de registro de ciudadanos de la ciudad para una verificación presencial de sus datos en un máximo de 14 días, de esta manera se consigue verificar que los datos del documento físico del ciudadano corresponden con los datos introducidos en el sistema digital.
- VII. Una vez confirmado, el secretario de la ciudad (que utiliza su identidad personal de uPort con unos permisos de administrador específicos) le otorga unas credenciales verificables que contienen su Zug ID firmada por la identidad de la ciudad. A continuación, otras organizaciones, públicas y privadas, pueden ofrecer servicios que requiera el uso del Zug ID, que se encontrará disponible en la aplicación uPort del usuario.
- VIII. De esta manera, los ciudadanos obtienen acceso a varios servicios mostrando simplemente su Zug ID en la aplicación uPort del usuario, en este caso podría ser, votando en las próximas elecciones.

- IX. El Zug ID es un objeto JSON que contiene toda la información verificada de un usuario. Estos datos no se almacenan en la blockchain sino en el dispositivo de la persona, en un entorno off-chain. En vez de tenerlos accesibles públicamente o almacenados en un proveedor de servicios centralizado, para cada transacción, Alice podrá seleccionar cuidadosamente que información debe compartir. Adicionalmente, su número de pasaporte, su fecha de nacimiento u otros datos sensibles nunca serán revelados a otras personas que escaneen la blockchain.
- X. De este modo, los usuarios pueden omitir el engorroso proceso actual de registro e inicio de sesión, y simplemente iniciando sesión con su cuenta de uPort, pueden votar y cerrar la sesión sin la necesidad de acudir a un centro de votación. De cara al gobierno las ventajas son la posibilidad de verificar la identidad de la persona que está votando sin depender de intermediarios o de infraestructura de conteo de votos. Esta prueba piloto demuestra que las identidades controladas por los usuarios respaldan la modernización de las iniciativas de votación electrónica, lo que podría ahorrarle a la ciudad millones de personas y costes de productividad.
- XI. De cara al futuro, la idea es utilizar estos IDs digitales para poder acceder a otros servicios de la ciudad, como por podría ser el servicio de alquiler de bicicletas, actualmente implementado bajo el nombre comercial AirBie, reservar un libro de la biblioteca, acceso a autobuses autónomos o para las aplicaciones de compartir coche y todo esto sin la necesidad de registrarte e iniciar sesión en cada uno de ellos de manera independiente, sino todo unificado con el Zug ID en la aplicación de uPort.

CAPÍTULO 3. Problemas existentes en los sistemas actuales

Actualmente existen un gran número de personas descontentas con los sistemas de adquisición de entradas para acudir a eventos, por ejemplo, conciertos. Este conjunto de personas está formado tanto por usuarios particulares que desean comprar un ticket como por empresarios, que se ven obligados a cumplir una estricta normativa en la cual han de invertir mucho tiempo y dinero. A lo largo de este próximo capítulo analizaremos ambas problemáticas y todos sus detalles, desde el funcionamiento hasta las legislaciones vigentes.

3.1 Problemática para los usuarios en la compra de entradas

3.1.1 ¿Qué son los bots y son todos malos?

Un bot (abreviatura de "robot") es un programa automatizado [27] que se ejecuta en Internet para realizar una tarea específica o un conjunto de tareas. Los bots de tickets, por lo tanto, son un tipo de bot que realiza tareas relacionadas con la emisión de tickets, como por ejemplo analizar los diferentes precios, verificar el inventario de asientos lanzados recientemente o incluso comprar entradas, donde está el principal negocio.

Los bots malignos provocan que Internet sea un lugar profundamente injusto, esto queda evidenciado fácilmente con la venta de tickets, para asistir a conciertos, festivales, teatro... Por este motivo, tanto los usuarios como las organizaciones de venta de entradas online se han posicionado en primera línea de la batalla contra los bots nocivos.

Para poder observar la magnitud del problema, debemos tener en cuenta que, según el fiscal general de Nueva York [26], solo uno de estos bots consiguió hacerse con 1.012 entradas para un concierto en apenas 1 minuto. A causa de esto, los fans frustrados se ven obligados a acudir a portales de reventa donde los precios pueden exceder el 1000% del valor nominal.

Las partes interesadas, desde los políticos hasta los músicos pasando por los fans, reclaman justicia en el proceso de emisión de tickets online. Algunos artistas han hecho todo lo posible para eliminar los bots maliciosos de las ventas de sus conciertos, incluida la emisión de tickets de manera, totalmente, offline. No cabe duda que, con la combinación correcta de tecnología y legislación, es posible mantener la venta de entradas online en el siglo XXI mientras se asegura que las entradas lleguen a manos de verdaderos fanáticos a un su precio nominal.

A pesar de ello, no todos los bots son necesariamente malignos. Existen bots que están constantemente trabajando de manera silenciosa para conseguir que nuestras vidas digitales funcionen sin problemas. Por ejemplo, se encargan de llenar e ir actualizando nuestros perfiles con nuevas noticias, nos informan del clima, proporcionan información sobre cotizaciones de acciones en tiempo real y nos ayudan a comparar precios de diferentes servicios. Estos bots son tan importantes que, según los datos de Imperva [28], representaron el 37,2% de todo el tráfico web en 2019.

De hecho, muchos bots son beneficiosos e incluso necesarios para un sitio web que funcione correctamente. Los bots araña (también llamados bots de Crawler) se encargan de indexar webs para Google y otros motores de búsqueda, determinando así las clasificaciones de búsqueda. Los bots de Fetcher crean vistas previas del contenido de un sitio web para dispositivos móviles y los bots de monitoreo alertan a los administradores cuando un sitio web no se ejecuta como debería, lo que podría derivar en la detección temprana de un problema de seguridad. Tampoco todos los bots de tickets son malos, por ejemplo, un bróker de tickets autorizado podría utilizar un bot para obtener información actualizada de los precios y un inventario de las entradas disponibles para ofrecerle estadísticas al vendedor de tickets principal.

Desafortunadamente, por cada bot "bueno", hay uno maligno listo para hacer daño. Los hackers usan estos bots online para interrumpir o manipular la sesión y para robar o suplantar identidades. El mismo informe de Imperva, asegura que casi 1 de cada 4 solicitudes web (24.1%) fueron hechas por un bot nocivo en 2019. Por otro lado, estos bots dañinos son especialmente frecuentes en la emisión de entradas, hasta el punto de llegar a representar el 39.9% del tráfico de todos los portales de emisión de tickets en 2019. Es tal su importancia, que cuando las personas hablan sobre los bots de tickets, normalmente se refieren a estos bots de tickets maliciosos.

3.1.2 ¿Cómo funcionan?

A continuación, veremos cómo los estafadores y especuladores utilizan bots para obtener ventajas injustas y llevan a cabo actividades fraudulentas en cada paso del proceso de venta de entradas.

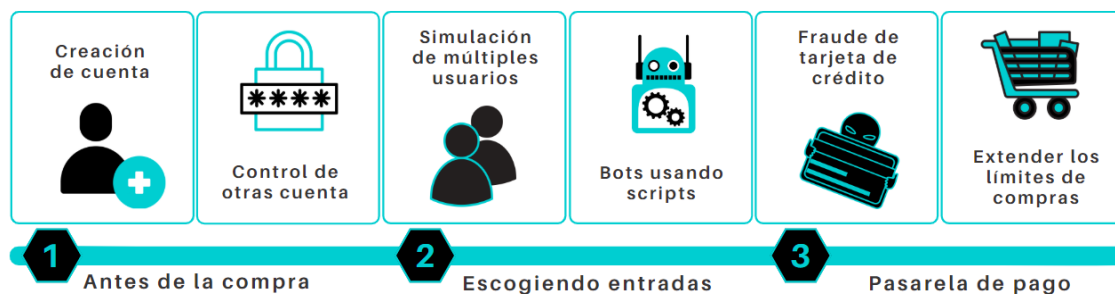


Fig. 3.1 Funcionamiento de los bots en la compra de entradas

3.1.2.1 Antes de la compra

Antes de que las entradas online se pongan a la venta, los bots malignos empiezan su trabajo y se utilizan para crear cuentas falsas o para obtener el control de cuentas legítimas ya existentes.

Por ejemplo, en el año 2017 un bróker de tickets aparentemente usó 9.047 cuentas independientes de Ticketmaster para comprar 315.528 entradas para la obra musical "Hamilton" [29] y otros eventos populares durante un período aproximado de 20 meses.

Creación de cuentas

Los estafadores abusan del proceso de registro para crear una cuenta y mediante el uso de bots consiguen crear cuentas en masa. Posteriormente, estas cuentas se usan incorrectamente con la finalidad de evitar los límites de compra de entradas, ya que la mayoría de las compañías tienen un límite de 4 o 6 entradas por cliente.

Control de otras cuentas

La segunda opción que hemos comentado anteriormente para evitar tener que crear cuentas de nuevos usuarios en masa, es la de obtener el control de cuentas legítimas ya existentes. En este caso, el importante papel de los bots es adivinar los nombres de usuarios y contraseñas comunes (cracking de credenciales) o bien mediante fuerza bruta, realizar múltiples intentos de inicio de sesión para conseguir un par de nombre de usuario/contraseña robados (stuffing de credenciales).

3.1.2.2 Escogiendo entradas

En el momento en el que las entradas online se ponen a la venta, los especuladores usan la ventaja de velocidad y volumen de los bots de tickets para

ganarles la partida a las personas reales y así obtener el mayor inventario posible. Los operadores de bots utilizan esta gran velocidad de computación en varios navegadores para eludir los límites de las entradas por cliente.

Al combinar la velocidad de una máquina con un gran volumen de cuentas, los operadores de bots reservan sin ningún tipo de esfuerzo cientos de entradas tan pronto como comienza la venta.

Tickets Management Threads							
Section	Row	Seats	Qty	Price	Account	Time Remaini...	Status
24L	P	15-16	2	US \$129.95		08:25	Ticket(s) Ready
							Awaiting CAPTCHA
C	5	13-14	2	US \$216.75		08:31	Ticket(s) Ready
C	4	11-12	2	US \$216.75		08:06	Ticket(s) Ready
C	5	15-16	2	US \$216.75	tmspinner2@mailin...	08:36	Purchasing

Fig. 3.2 Threads de un bot comprando entradas paralelamente

Existen tres partes en este proceso de reserva y compra de múltiples entradas:

Rastreo/Análisis

Con este análisis previo, los especuladores, o mejor dicho sus bots, se configuran para ser capaces de monitorear unos tipos de entradas específicas que posteriormente saben que tendrán más salida en el mercado de reventa.

Expedición

Es poco después, en la fase de expedición cuando utilizan la velocidad de computación y las secuencias de comandos ya programadas para reservar y comprar los tickets rápidamente mucho antes que cualquier humano. Para podernos hacer una idea, un bot configurado para seleccionar y comprar entradas puede llegar fácilmente a la página de pago en el mismo tiempo que le tomaría a un fan real en escribir su dirección de correo electrónico. Si tenemos en cuenta que un solo bot puede abrir 100 ventanas y, simultáneamente, pasar a la página de pago en todas ellas a la vez, podemos observar claramente la desigualdad.

Denegación de inventario

Recientemente, ha aparecido una nueva generación de bots (llamados bots de denegación de inventario) que se encargan de que las entradas aparezcan en páginas de reventa como Viagogo o StubHub incluso antes de que salgan a la venta para el público general en la página oficial.

La denegación de inventario funciona de la siguiente manera, se utilizan los bots para agregar entradas al carrito de compra, de esta manera se consigue que dejen de estar disponibles para que los fans las compren. Los especuladores saben que algunos fans al ver el mensaje "no hay entradas disponibles", querrán ir al evento de todos modos y pagarán lo que sea para obtener un ticket. Entonces una vez ya tienen el carrito lleno, los especuladores colocan las entradas (que realmente aún no han adquirido) en los sitios de reventa. Cuando los fans compran las entradas a los precios de reventa escandalosamente inflados, es cuando los bots de los especuladores compran y pagan realmente los tickets, de esta manera consiguen obtener una mayor ganancia en el proceso, ya que evitan comprar entradas que posteriormente quizás podrían no tener la oportunidad de revender.

3.1.2.3 Pasarela de pago

Los especuladores casi siempre utilizan bots para exceder el límite de entradas permitidas, lo que rompe los términos y condiciones de uso de las compañías de venta. Sin embargo, esa no es la única infracción que se comete, mientras que algunos estafadores pagan estas entradas con tarjetas de crédito legítimas, existen otro grupo que realizan todo esto con información de tarjetas robadas o pirateadas, lo que aumenta sus ganancias, ya que ni siquiera gastan el dinero del valor nominal de la entrada.

Compra

En este momento de la compra, se realiza el fraude con las tarjetas de crédito usando información de tarjetas robadas para comprar los tickets. De antemano o durante la compra, los estafadores utilizan bots para verificar la validez de las tarjetas robadas y para identificar fechas de vencimiento o códigos de seguridad necesarios para el uso de las tarjetas robadas.

3.1.3 Legalidad

Los bots de compra de entradas online han existido durante al menos 20 años, sin embargo, ha sido en los últimos 5 años cuando los gobiernos han comenzado a atacar a los bots con legislación. A pesar de ello, no existe una regulación global al respecto y dependiendo del país, los bots de tickets online pueden ser legales o ilegales, al menos técnicamente hablando, aunque a la práctica sigan existiendo y llevando a cabo sus actividades maliciosas por todo el mundo.

En lo que respecta a la UE (Unión Europea), donde pensamos desplegar inicialmente nuestra solución, en abril de 2019 el Parlamento de la UE votó a favor de prohibir el uso de bots de tickets para comprar entradas masivamente para posteriormente revenderlas con la finalidad de garantizar la accesibilidad a las entradas a todos los individuos.

Esta legislación supone la primera regulación de la UE sobre el tema, y también deja la puerta abierta para que los estados miembros aprueben leyes adicionales, más específicas, con respecto a la reventa de entradas. El Consejo de la UE adoptó la legislación en noviembre de 2019, por lo que los estados miembros ahora disponen de dos años para transformar las regulaciones en leyes nacionales.

3.2 Problemas de almacenamiento de datos para las empresas

3.2.1 Historia y conceptos básicos de la GDPR

El derecho a la privacidad forma parte de la Convención Europea de Derechos Humanos desde 1950 y establece que "toda persona tiene derecho al respeto de su vida privada, familiar y de su hogar". Sobre esta premisa, la UE ha tratado de garantizar la protección de este derecho mediante múltiples legislaciones en distintos países.

A medida que la tecnología ha ido avanzando y Internet fue inventado, la UE reconoció la necesidad de protecciones más modernas y en línea con el contexto tecnológico actual [30]. En 1995, se aprobó la Directiva Europea de Protección de Datos, estableciendo estándares mínimos de privacidad y seguridad de datos, a partir de los cuales cada estado miembro implementó su propia ley. Sin embargo, Internet siguió evolucionando y pronto las regulaciones de la época se fueron quedando obsoletas. En 2000, la mayoría de las instituciones financieras empezaban a ofrecer servicios bancarios online. En 2006, Facebook se abrió al público y en 2011, un usuario de Google demandó a la empresa por escanear sus correos electrónicos personales. Es por ello, que tan sólo dos meses después de eso, la autoridad de protección de datos de Europa declaró que la UE necesitaba "un enfoque integral sobre la protección de datos personales" y se comenzó a actualizar la directiva de 1995.

A partir de ese momento, empezó la redacción de diversas normativas que culminó con la actualmente conocida como GDPR, que entró en vigor en 2016 después de ser aprobada por el Parlamento Europeo y que desde el 25 de mayo de 2018 todas las organizaciones deben cumplir.

El Reglamento General de Protección de Datos es la ley de privacidad y seguridad más estricta del mundo. A pesar de que fue redactado y aprobado por la Unión Europea, impone obligaciones a todas las organizaciones independientemente del lugar de su sede principal, siempre y cuando recopilen o almacenen datos relacionados con personas de la UE.

Con la GDPR, Europa impone su postura firme sobre la privacidad y la seguridad de los datos en un momento en que cada vez más personas confían sus datos personales a servicios en la nube y las infracciones, especialmente de las grandes empresas, son algo cotidiano. La regulación en sí es ambiciosa, de gran alcance y bastante liviana en detalles, lo que hace que el cumplimiento de GDPR sea una tarea desalentadora pero obligatoria.

Esta regulación define un conjunto de términos legales [31], a continuación, describiré los más importantes con la finalidad de facilitar la comprensión del resto de puntos de este capítulo:

- **Datos personales:** Cualquier información que se relacione con un individuo que pueda identificarlo directa o indirectamente. En la GDPR se distinguen dos tipos de datos, personales y sensibles (para más información, consultar el apartado 1.1.1 de este mismo documento). Los datos seudónimos también pueden incluirse en la definición si es relativamente fácil identificar a alguien a partir de ellos.
- **Procesado de datos:** Cualquier acción realizada sobre los datos, ya sea automatizada o manual. Por ejemplo, la recopilación, grabación, organización, estructuración, almacenamiento, utilización, borrado...
- **Sujeto de datos:** La persona propietaria de los datos que se procesan, dependiendo de la entidad estamos hablando de clientes o visitantes de un determinado sitio web.
- **Controlador de datos:** La persona de la empresa que decide por qué y cómo se procesarán los datos personales. Este cargo es extensible también a cualquier trabajador de la compañía que maneje datos de usuarios durante la ejecución de sus tareas.
- **Procesador de datos:** Un tercero que procesa datos personales en nombre de un controlador de datos. La GDPR tiene reglas especiales para estos individuos y organizaciones, que pueden incluir servidores en la nube o proveedores de servicios de correo electrónico.

3.2.2 Principios de protección de datos

El artículo 5 del GDPR [32] establece siete principios básicos que son clave para la comprensión del régimen general de protección de datos.

- **Legalidad, equidad y transparencia:** El procesamiento debe ser legal, justo y transparente para el cliente.
- **Limitación de propósito:** Se deben procesar los datos para los fines legítimos y especificados explícitamente al interesado cuando fueron recopilados.

- **Minimización de los datos recopilados:** Se debe recopilar y procesar solo la cantidad de datos que sea absolutamente necesaria para los fines especificados.
- **Precisión:** Se deben mantener los datos personales correctos y actualizados.
- **Límite de almacenamiento:** Solo se permite almacenar datos de identificación personal durante el tiempo que sea necesario para el propósito especificado.
- **Integridad y confidencialidad:** El procesamiento debe realizarse de tal manera que se garantice la seguridad, integridad y confidencialidad adecuadas (por ejemplo, mediante técnicas de encriptación).
- **Responsabilidad:** El controlador de datos es responsable de poder demostrar que el procesamiento de los datos cumple con todos estos principios establecidos en la GDPR.



Fig. 3.3 Principios de protección de datos

3.2.3 Derechos de las personas

Como ya hemos mencionado anteriormente, cualquier persona que utilice Internet, es considerado un sujeto de datos. Para proteger a este colectivo, la GDPR les reconoce un conjunto de derechos de privacidad, cuyo objetivo es proporcionar a las personas un mayor control sobre los datos que prestan a las organizaciones. Es por ello que, como organización, es importante comprender estos derechos para garantizar el cumplimiento de la GDPR.



Fig. 3.4 Derechos de las personas según la GDPR

3.2.3.1 Derecho a ser informado

El derecho a ser informado [33], recogido en los artículos 13 y 14 de la GDPR, es uno de los requisitos clave de transparencia de la GDPR. Se trata de proporcionar a los usuarios información clara, concisa, fácilmente accesible sobre lo que haces con sus datos personales, es decir, comunicarles tus objetivos a la hora de recopilar esos datos, durante cuánto tiempo los almacenarás o si serán compartidos con alguna otra entidad, esto es a lo que llamamos "información de privacidad", además debe proporcionarse mediante un lenguaje sencillo para facilitar el entendimiento de cualquier ciudadano y antes de un mes desde que se produjo la recogida.

La información de privacidad puede ofrecerse usando un gran número de técnicas para hacerlas sencillas y visuales, como, por ejemplo, mediante iconos, ventanas emergentes o incluso alertas de voz.

El cumplimiento de este punto puede ayudar posteriormente a cumplir con otros aspectos de la GDPR, además de fomentar la confianza de las personas hacia la empresa y obtener más información útil sobre ellas.

3.2.3.2 Derecho al acceso

El derecho al acceso [34] otorga a las personas el derecho de obtener una copia de sus datos personales, así como otra información complementaria. Ayuda a los usuarios a comprender cómo y por qué se están utilizando sus datos y, en caso deseado, a verificar si la compañía lo está haciendo legalmente.

Cabe destacar que cada individuo solo podrá solicitar su propia información y no la de cualquier otra persona, independientemente de la relación o parentesco entre ellas (a excepción de que la información también les implique a ellos),

queda en manos de la empresa, verificar que el usuario es realmente quién dice ser.

Sin embargo, la GDPR no especifica ningún formato estándar a la hora de realizar una petición de sus datos a una empresa, es por ello que diseñar un formulario específico podría ayudar a ambas partes a simplificar el proceso. De nuevo, la información proporcionada debe ser clara, concisa, fácilmente accesible y mediante un lenguaje sencillo para facilitar el entendimiento de cualquier ciudadano.

3.2.3.3 *Derecho a la rectificación*

Según el artículo 16 del GDPR, las personas tienen derecho a que se rectifiquen [35] sus datos personales en caso de que sean erróneos, incompletos o inexactos. Este derecho tiene una estrecha relación con el principio de precisión de la GDPR (*Artículo 5 (1) (d)*). Sin embargo, es posible que a pesar de haberse tomado medidas para asegurar que los datos personales fueran precisos cuando se obtuvieron, este derecho a la rectificación impone una obligación específica, por parte de la empresa, de reconsiderar la precisión de los mismos cuando se solicite por parte del cliente.

Esta solicitud de rectificación puede realizarse verbalmente o por escrito y la compañía tiene como máximo un mes para responder. De acuerdo con el Artículo 18, un individuo tiene derecho a solicitar la restricción del procesamiento de sus datos personales mientras la empresa esté verificando la exactitud y corrección de sus datos, veremos este derecho de una manera más detallada en el apartado 6.3.5.

3.2.3.4 *Derecho a la eliminación (derecho al olvido)*

Según el artículo 17 de la GDPR, uno de los puntos más controvertidos de esta regulación, las personas tienen derecho a que se borren [36] sus datos personales, tanto de servidores principales como de respaldo. Esto también se conoce como el "derecho a ser olvidado". El derecho no es absoluto y solo se aplica en ciertas circunstancias como, por ejemplo:

- Los datos personales ya no son necesarios para el objetivo inicial para el que fueron recopilados o procesados.
- El usuario retira explícitamente su consentimiento.

En caso de que los datos personales de ese usuario hayan sido revelados a terceros, se debe contactar con cada una de estas entidades para informarles de la petición, con la finalidad de que puedan eliminar también el contenido que tengan almacenado.

En este caso, la GDPR tampoco especifica ningún formato estándar a la hora de realizar una petición de borrado de los datos a una empresa, es por ello que diseñar un formulario específico podría ayudar a ambas partes a agilizar el proceso.

3.2.3.5 *Derecho a restringir el procesamiento*

De acuerdo con el artículo 18 de la GDPR las personas poseen el derecho de restringir el procesamiento de sus datos personales [37] en ciertas circunstancias. Esto significa que un individuo puede limitar la forma en la que una organización utiliza sus datos. Cuando el procesamiento está restringido, se permite el almacenamiento, pero no la utilización de los datos, esta suele ser una alternativa previa a solicitar el borrado permanente de los mismos.

Las personas tienen derecho a restringir el procesamiento de sus datos personales cuando tengan una razón concreta para desear la restricción, por ejemplo, estar en desacuerdo con el contenido de la información que la compañía tiene o cómo ha procesado sus datos anteriormente. En la mayoría de los casos, no se le solicitará que restrinja los datos personales de un individuo de forma indefinida (ya que, en ese caso, tendría la misma utilidad que el borrado definitivo), pero si deberá tener la restricción en vigor durante un cierto período de tiempo.

3.2.3.6 *Derecho a la portabilidad de datos*

El derecho a la portabilidad de los datos [38] otorga a las personas el derecho a recibir los datos personales que hayan proporcionado a una entidad, en un formato estructurado, portable, de uso común y legible por una máquina. Además, también les da derecho a solicitar que una entidad transmita estos datos directamente a otra entidad de forma segura. De esta manera, permitimos que los usuarios puedan mover, copiar y transferir información personal para poderla utilizar en diferentes servicios o aplicaciones.

En este caso, los datos personales proporcionados a una entidad se extienden más allá del nombre de usuario, edad o correo electrónico y pueden incluir el historial de uso de la página web, datos de tráfico o ubicación o búsquedas realizadas por el usuario.

La GDPR contempla dos situaciones posibles para hacerle llegar al usuario sus datos personales:

- Transmisión directa de los datos solicitados por el individuo.
- Proporcionar acceso a una herramienta automatizada que permita al individuo extraer los datos solicitados por sí mismo, en cuyo caso se deberá asegurar que el mecanismo empleado es seguro y sencillo.

3.2.3.7 *Derecho a objetar el procesamiento*

El artículo 21 de la GDPR otorga a las personas el derecho de oponerse al procesamiento de sus datos personales [39] en cualquier momento. Esto, efectivamente, permite a las personas detener o evitar que se procesen sus datos. Esta objeción puede estar relacionada con todos los datos personales que una compañía tenga sobre un individuo o solo con cierta información. A su misma vez, también puede estar relacionado con un propósito específico para el que se estén procesando los datos.

Sin embargo, existen excepciones cuando el procesado de datos se está llevando a cabo con fines de investigación científica, histórica o estadística, donde el derecho a la objeción está más limitado.

3.2.3.8 *Derechos en relación con la toma de decisiones automatizada y la elaboración de perfiles*

Finalmente, los usuarios tienen derecho a no ser objeto de una decisión basada exclusivamente en un tratamiento automatizado [40]. Sin embargo, existen algunas excepciones a esta regla, como en los casos donde se haya dado un consentimiento explícito a la toma de decisiones automatizadas. Cuando una compañía lleve a cabo este tipo de procedimientos, se debe informar al usuario de los siguientes puntos:

- Informar al individuo sobre que decisiones son automatizadas.
- Darle la opción de que una persona humana revise la decisión tomada de manera automática, ya que es uno de los derechos contemplados por la GDPR.
- Una vez conocidos los puntos anteriores, dar al usuario la posibilidad de impugnar la decisión automatizada.

3.2.4 **Multas y sanciones**

Infringir alguna de las prácticas anteriores puede exponer a la compañía a costosas multas y provocar daños a su reputación. La GDPR se basa en múltiples criterios a la hora de sancionar, como la gravedad de la infracción, la intencionalidad, el grado de responsabilidad, infracciones previas o incluso la cooperación con las autoridades de control, a partir de todos estos valores se diferencian dos tipos de infracciones [41] según su nivel de gravedad:

- **Infracciones leves:** Se considera leve las situaciones donde la empresa no pueda demostrar que aplica un nivel de seguridad adecuado con los datos recopilados, no haya establecido un acuerdo sobre el nivel de procesamiento de datos o no coopere con las autoridades de control. En este caso se sanciona con hasta 10 millones de euros o el 2% del volumen de

facturación anual de la empresa, será escogida la cifra más elevada de las dos.

- **Infracciones graves:** Esta multa se aplica, por ejemplo, si se han infringido los derechos individuales, como en el caso de haber procesado datos sin el explícito consentimiento del propietario de los mismos. Se sanciona con hasta 20 millones de euros o el 4% del volumen de facturación anual de la empresa, será escogida la cifra más elevada de las dos.

Como conclusión, las fuertes multas de la GDPR tienen como objetivo garantizar que poner en peligro la seguridad de los datos de los usuarios y violar los estándares de privacidad y seguridad sea demasiado costoso como para que ninguna compañía se pueda permitir el no adoptarlas.

CAPÍTULO 4. Cibertickets, nuestra solución basada en uPort

A raíz de detectar todos los problemas vistos en el capítulo anterior y observar cómo afectaban a un gran número de personas, surgió la oportunidad de aprovechar los conocimientos tecnológicos y las nuevas tecnologías presentes en el mercado para solucionar una problemática tan extendida, de ahí surgió la idea de Cibertickets. En este cuarto capítulo veremos cómo se ha implementado Cibertickets, que tecnologías, algoritmos y librerías utiliza y cómo trabaja conjuntamente con uPort, ofreciendo un portal donde comprar entradas de conciertos, asignadas a una identidad digital soberana para así evitar los bots y reventa. Además, la propia base de datos de Cibertickets no almacena ningún dato personal de los usuarios, evitando así las regulaciones vigentes. Para más información sobre el código fuente del proyecto, se puede encontrar en el repositorio del proyecto en GitHub².

Por tanto, con esta implementación se consigue mejorar la experiencia para todas sus partes integrantes:

- Para los fans, se obtiene una experiencia justa a la hora de comprar sus entradas, es decir, se tiene la misma oportunidad que cualquier otro fan de conseguir los tickets disponibles a su valor nominal.
- Para un artista, el beneficio es poder poner las entradas en manos de personas reales para que así sus espectáculos estén completamente llenos. En el caso de reventa, al subir tanto el precio real de las entradas algunas localidades acaban vacías por falta de público que pueda pagar esos precios desorbitados.
- Para una compañía de venta de entradas, el beneficio es poder proporcionar acceso a humanos reales para comprar los tickets disponibles y eliminando así cualquier automatización que abuse del sistema y arruine la experiencia de compra de entradas para los fanáticos reales y a continuación provoque una opinión negativa de la empresa. Además, con la proliferación del uso de los datos personales y la gran cantidad de datos gestionados, la posibilidad de incumplir alguno de los artículos requeridos para la protección de los usuarios europeos y, por tanto, la multa de la Agencia correspondiente, solo es cuestión de tiempo y con este mecanismo podría evitarse.

² <https://github.com/Lails21/TFG>

4.1 Tecnologías, herramientas y lenguajes utilizadas

4.1.1 MongoDB



MongoDB [42] es un sistema de base de datos NoSQL (Not Only Structured Query Language), de código abierto y orientado a documentos. En vez de almacenar los datos en tablas, lo más común en las bases de datos relacionales, MongoDB guarda estructuras de datos tipo JSON que facilita notablemente la lectura. Algunas de las principales ventajas que ofrece:

- Fácilmente escalable gracias a su carácter descentralizado.
- Integración de datos rápida, sencilla y flexible.
- Optimización a la hora de realizar consultas que impliquen grandes cantidades de datos almacenados.

4.1.2 Angular



Angular [43] es un framework para el desarrollo de aplicaciones web de manera eficiente y dinámica, es de código abierto, escrito en TypeScript y sustentado por Google. Algunas de las principales ventajas que ofrece:

- Arquitectura jerárquica basada en componentes que permite una gran modulación del código, reutilizando así en diferentes secciones componentes de una naturaleza similar.
- La existencia de Angular Material, una librería de estilos que contiene una gran variedad de componentes para la UI (User Interface) que facilitan la creación de estilos y animaciones dentro de nuestra aplicación.

4.1.3 NodeJS



NodeJS [44] es un entorno de ejecución para JavaScript creado por Ryan Dahl, es decir, una plataforma de servidor para seguir a aplicaciones en tiempo real. Además, utiliza el mismo motor de JavaScript que Chrome y está orientado a eventos asíncronos, lo cual lo convierte en una opción eficiente y liviana. Algunas de las principales ventajas que ofrece:

- Posee un ecosistema de paquetes, llamado NPM (Node Package Manager), considerado el ecosistema de librerías de código abierto más extenso del mercado.
- Es una tecnología ligera, con una curva de aprendizaje poco pronunciada que permite acortar el tiempo de desarrollo de una aplicación.
- Permite un soporte directo de los documentos JSON, por lo tanto, es especialmente útil para construir REST (Representational State Transfer) APIs (Application Programming Interface) e integrarlo con bases de datos NoSQL como MongoDB.

4.1.4 ExpressJS

The logo for ExpressJS, featuring the word "express" in a lowercase, sans-serif font.

Express [45] es una librería que proporciona utilidades básicas para el desarrollo web, se podría definir como un framework que recibe, maneja y responde peticiones web, por ejemplo, HTTP (GET, POST, PUT, DELETE...). Además, permite definir ciertos ajustes como el puerto de escucha del servidor o la estructura de las rutas que se utilizan para generar y responder a peticiones. Actualmente, se trata de uno de los frameworks más utilizados de Node. Algunas de las principales ventajas que ofrece:

- Facilidad de configuración y customización, mediante el uso de middlewares que permiten realizar tareas adicionales como, por ejemplo, autenticación.
- Utiliza JavaScript al igual que NodeJS, por tanto, no añade complejidad o conocimientos extra para los desarrolladores.

4.1.5 Git



Git [46] es una herramienta de control de versiones que mejora la eficiencia, controla y registra los cambios en los diferentes archivos del proyecto. Además, permite una mejor coordinación y una ayuda adicional a la hora de compartir código cuando trabajan varias personas en el mismo proyecto. Algunas de las principales ventajas que ofrece:

- Permite comparar el código con las versiones anteriores y en caso de necesitarlo, revertir los cambios al estado previo.
- Posibilidad de trabajar en paralelo sobre el mismo proyecto y fusionar posteriormente todas las mejoras mediante el uso de ramas.
- Conocer quién ha modificado recientemente el código pudiendo haber causado algún problema o incompatibilidad.

4.1.6 Mockplus

Previamente a la creación de la interfaz gráfica de usuario es necesaria la realización de unos prototipos o mockups para definir cómo queremos que luzca nuestra página web en este caso. De esta manera conseguimos tener una idea aproximada del objetivo que queremos obtener, además nos permite visionar fácilmente el producto final, detectar posibles fallos antes de empezar, nos ahorra tiempo más adelante y en caso de trabajar para un cliente, discutir los pequeños detalles estéticos del resultado.

Para llevar a cabo este procedimiento, en este proyecto he sido utilizado el software Mockplus [47] que facilita el proceso de prototipado y agiliza la creación de bocetos. Asimismo, posee un gran abanico de elementos completamente

personalizables, de manera que consigue con una mayor sencillez, un diseño muy detallado de los mockups.

4.2 Patrón MVC

El patrón de MVC (Modelo-Vista-Controlador) [48] se basa en un patrón de arquitectura de software que desliga los datos de una aplicación, y la lógica de control de la aplicación y la interfaz del usuario en tres componentes distintos tal y como podemos ver en el siguiente diagrama:

- **Modelo:** Representa la información y los datos que maneja el sistema, los mecanismos de persistencia y la lógica de negocio. Además, lleva un registro de los controladores y las vistas del sistema.
- **Vista:** También conocido como interfaz de usuario, está formado por la información que se le envía al usuario y las interacciones con este mediante el uso de un formato adecuado, legible e intuitivo.
- **Controlador:** Se encarga de responder a eventos, normalmente acciones instanciadas por el usuario, y realiza peticiones al modelo cuando se requiere la solicitud de cierta información. En otras palabras, podríamos decir que el controlador actúa como intermediario entre la vista y el modelo.

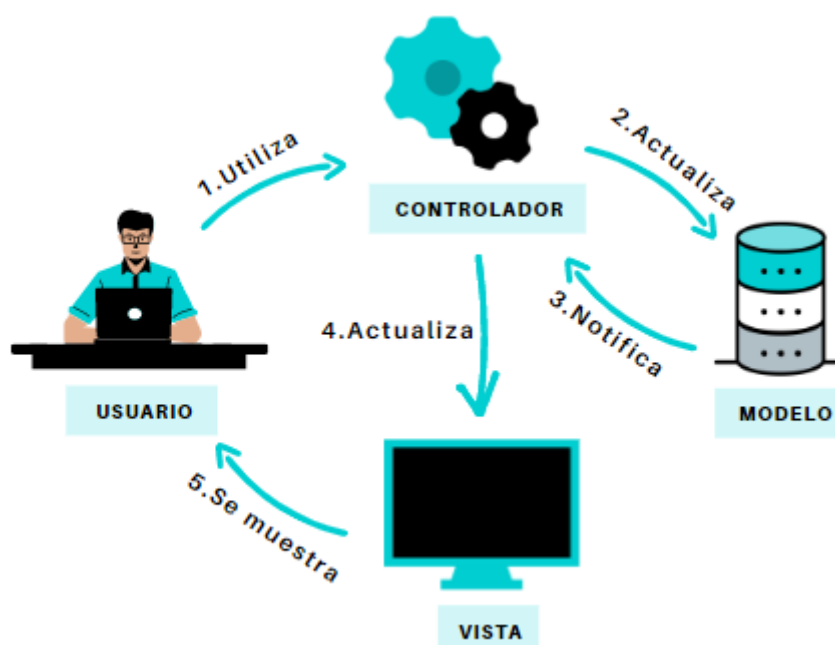


Fig. 4.1 Diagrama de flujo del modelo Vista-Controlador

4.3 Librerías utilizadas

4.3.1 uPort Credentials

uPort Credentials [49] proporciona una solución simple para que los usuarios de un determinado servicio inicien sesión en su aplicación/servidor, creen, compartan, intercambien y administren credenciales privadas, como información de identidad y detalles de contacto, por ejemplo, email o número de teléfono, basado en las especificaciones de los DIDs. Por otro lado, aparte de pedirle a un usuario datos verificados sobre sí mismo, el servidor también puede ayudar al usuario a construir su identidad verificando sus datos mediante firmas, por ejemplo, un título universitario verificado por la universidad o una entrada de concierto verificada por la empresa emisora.

Para poder implementar todas estas funcionalidades se utiliza la librería ‘uport-credentials’ que permite principalmente:

- Crear y verificar solicitudes de autenticación
- Solicitar datos verificados
- Firmar y verificar datos de los usuarios firmados por otras identidades para facilitar la comunicación segura entre dos partes

Se trata de una librería específica para el lado del servidor, donde las claves se almacenan de forma segura. Además, simplifica el proceso de creación de identidades dentro de las aplicaciones JavaScript. Para conseguir una solución de extremo a extremo, se utiliza esta librería en combinación con ‘uport-transport’ para permitir la comunicación entre un navegador web/servidor y la aplicación uPort del usuario.

Como ha sido comentado anteriormente, uPort Credentials cuenta con una gran cantidad de funciones [50] y a continuación se explican las más importantes y utilizadas en el proyecto:

- **new Credentials:** Esta función instancia un nuevo objeto de uPort Credentials que en nuestro caso está formado por los siguientes parámetros, a pesar de que podría contener algunos más.

Parámetro	Tipo	Descripción
appName	String	Nombre de nuestra aplicación a la cual queremos crearle una identidad

did	DID	Identificador único para nuestra aplicación
privateKey	String	Clave privada de 32 bytes codificada en hexadecimal
resolver	Resolver	Objeto de resolución para resolver documentos DID

Tabla 4.1 Parámetros de la función new Credentials

- **createVerification:** Esta función se encarga de crear una credencial, es decir un JWT ya firmado, que en nuestro caso de uso será la correspondiente entrada del concierto.

Parámetro	Tipo	Descripción
sub	DID	DID del dueño de la credencial, es decir, el usuario que ha comprado la entrada
claim	JSON Object	Objeto con todos los valores importantes de la credencial, este campo es completamente personalizable, en nuestro caso la información del concierto
exp	Number	Hora a la que vence la credencial y deja de ser válida (en segundos desde epoch)

Tabla 4.2 Parámetros de la función createVerification

- **createDisclosureRequest:** Esta función sirve para crear un JWT que contenga un Selective Disclosure Request y devuelve una promesa con el JWT si ha funcionado correctamente o con un error en caso contrario.

Parámetro	Tipo	Descripción
type	String	Debe tener el valor 'shareReq'
iss	DID	DID de la identidad firmante, en este caso el DID de Cibertickets
iat	Number	Hora de emisión de la solicitud
exp	Number	Hora de expiración del JWT
callbackURL	String	URL del callback donde se devolverá la respuesta a la solicitud
requested	Array	Vector de los datos autofirmados solicitados al usuario, actualmente pueden ser: ["name", "email", "image", "country", "phone"]
permissions	Array	Vector de los permisos necesarios, actualmente solo es soportado el método de notificaciones

Tabla 4.3 Parámetros de la función createDisclosureRequest

- authenticateDisclosureResponse:** Esta función es la encargada de autenticar el JWT que contiene la respuesta de divulgación selectiva (Selective Disclosure Response) del cliente de uPort, es decir, obtiene el token, lo verifica y lo analiza. Devuelve una promesa con el JWT autenticado si ha funcionado correctamente o con un error en caso contrario.

Parámetro	Tipo	Descripción
type	String	Debe tener el valor 'shareResp'

iss	DID	DID de la identidad firmante, en este caso el DID del usuario/cliente que comparte sus datos
aud	DID	DID del creador de la solicitud, en este caso el DID de Cibertickets
iat	Number	Hora de emisión de la respuesta a la solicitud
exp	Number	Hora de expiración del JWT
req	String	El JWT original codificado en la Selective Disclosure Request
own	JSON Object	Objeto de los datos autofirmados solicitados al usuario, por ejemplo pueden ser: <pre>{"name": "Laia", "email": "laia@example.com"}</pre>
capabilities	Array	Vector con un JWT que le otorga a Cibertickets los permisos solicitados, actualmente consiste en un token para enviar notificaciones push al usuario
boxPub	String	Clave pública de curva elíptica (Curve25519) codificada en base64 con 32 bytes de longitud utilizada para enviar mensajes cifrados al usuario

Tabla 4.4 Parámetros de la función `authenticateDisclosureResponse`

4.3.2 uPort Transports

La librería ‘uport-transport’ [51] se compone de una colección de funciones, llamada *transports*, que se utilizan para configurar diversos canales de comunicación entre una aplicación/servidor y el dispositivo móvil del cliente, esta librería permite:

- Enviar mensajes a los usuarios usando un código QR

- Enviar solicitudes y recibir respuestas a través de URL
- Enviar notificaciones push cifradas

En definitiva, incluye simplemente funciones que consumen peticiones de envío de mensajes junto a algunos parámetros de transporte y posteriormente envían esas cadenas de caracteres al dispositivo de un cliente uPort. Existen varios métodos a la hora de enviarle mensajes al usuario y alguno de ellos, también permiten recibir una respuesta a una solicitud determinada.

Muchas de estas funciones disponibles se pueden combinar para crear métodos de transporte específicos para un determinado caso de uso y entorno. Además, como ya ha sido mencionado anteriormente, se puede usar 'uport-transport' en combinación con otras librerías, como la que hemos visto anteriormente, 'uport-credentials' que se encarga de la creación de mensajes, los que después queremos enviar.

Como ya ha sido comentado anteriormente, uPort Transport cuenta con una gran cantidad de funciones [52] y a continuación se explican las más importantes y utilizadas en el proyecto:

- **getImageDataURI:** Esta función, a partir de una cadena de datos dada, devuelve una imagen en forma de URI (Uniform Resource Identifier) que, en este caso, corresponde a un código QR. Posteriormente, esta imagen se puede mostrar en el frontend mediante la etiqueta 'img' de HTML configurando el atributo 'src' con la imagen deseada.

Parámetro	Tipo	Descripción
data	String	Cadena de datos, típicamente una URI de uPort

Tabla 4.5 Parámetros de la función getImageDataURI

- **paramsToQueryString:** Esta función agrega parámetros como parámetros de consulta a la URL y posteriormente la convierte a formato String.

Parámetro	Tipo	Descripción
url	String	URL previamente creada sobre la cual queremos añadirle nuevos parámetros
params	JSON Object	Objeto de parámetros válidos que queremos agregar al fragmento de URL dado

Tabla 4.6 Parámetros de la función `paramsToQueryString`

- **messageToURI:** Esta función se encarga de convertir un JWT en una URI según el esquema deseado, para posteriormente poder convertirla en un código QR mediante el método `getImageDataURI`.

Parámetro	Tipo	Descripción
message	String	El mensaje que queremos codificar como si fuera una URI
type	String	Esquema deseado de la URI, 'universal' o 'deeplink', por defecto está seleccionado el primero de ellos

Tabla 4.7 Parámetros de la función `messageToURI`

- **push.send:** Esta función permite enviarle una notificación push a la aplicación móvil de uPort de un determinado cliente, del que posee un token válido. El resultado de la función es el envío de la notificación en caso de éxito y un error en caso de fallo.

Parámetro	Tipo	Descripción
pushToken	String	Token para enviar las notificaciones push, para obtenerlo es necesario solicitar permisos en la petición inicial
pubEncKey	String	Clave pública de curva elíptica (Curve25519) codificada en base64 con 32 bytes de longitud utilizada para enviar mensajes cifrados al usuario

Tabla 4.8 Parámetros de la función push.send

4.3.2.1 Peticiones

Las solicitudes [53] siempre consisten en URLs que se encapsulan de maneras distintas para que puedan ser manejadas por la aplicación móvil de uPort. Actualmente, existen tres métodos de transporte principales para enviar la URL y para gestionar solicitudes:

1. **Códigos QR:** Los mensajes necesarios se envían mediante el escaneo de un código QR a la aplicación móvil de uPort del cliente. Se puede implementar el modal y flujo predeterminados por uPort o personalizar los códigos QR. Además, el uso de un servidor de mensajería propietario, llamado Chasqui, permite recibir respuestas tanto en el servidor de mensajería como en el propio servidor de la entidad que ofrece el servicio web.
2. **Notificaciones push:** Los mensajes se cifran y posteriormente, se envían a la aplicación móvil de uPort del cliente a través de una notificación push, utilizando un servicio de notificación push proporcionado por uPort, que explicaremos más adelante.
3. **Mediante URL:** Este método solo puede utilizarse cuando un cliente uPort y la aplicación web de la empresa están en el mismo dispositivo móvil, en este caso, las solicitudes y respuestas se transfieren a través de una URL. Los mensajes se envían mediante una URL y las solicitudes se devuelven en una URL que se parsea y se devuelve.

A continuación, nos centraremos en el funcionamiento de los dos primeros métodos [54] ya que serán los que se implementarán en nuestro proyecto.

Códigos QR

Uno de los principales beneficios de este formato, es que al codificar la solicitud en un código QR, es muy fácil para los usuarios, con la aplicación uPort instalada, escanearlo. A partir de iOS 11, también se puede escanearlos con la cámara del sistema y automáticamente se abrirá la aplicación.

Si bien este mecanismo se usa fundamentalmente para interactuar con una aplicación web corriendo en un navegador, existen otras maneras, ya que los códigos QR se pueden imprimir o mostrar en conferencias, carteles u otros casos de uso del mundo real.

Sin embargo, una desventaja es que si las solicitudes que deseamos enviar son demasiado grandes se generarán códigos QR grandes que pueden ser difíciles de escanear. Para evitar este inconveniente, uPort ha creado formatos de URL compactas para reducir al máximo el tamaño de las solicitudes.

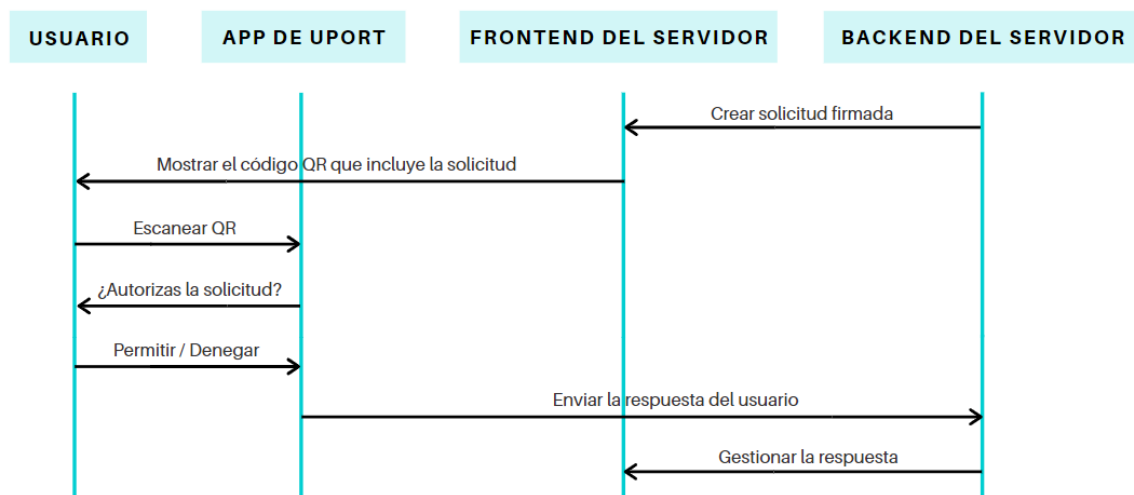


Fig. 4.2 Esquema de creación de solicitudes mediante QR

1. La URL que incluye la solicitud se muestra como un código QR en el navegador web.
2. El usuario, usando la aplicación móvil de uPort, escanea el código QR y acepta o declina la solicitud
3. La respuesta se envía directamente al backend del servidor, que se comunica con el frontend para mostrar la información necesaria en el navegador web.

Notificaciones Push

Como parte del Selective Disclosure Flow, se pueden solicitar permisos a un usuario para enviarle las solicitudes directamente a su aplicación uPort mediante notificaciones push.

Una de las principales ventajas de las notificaciones push es que permiten que la interacción fluya mucho más rápido y de manera más simple para los usuarios, algo realmente útil si tienen que interactuar con múltiples solicitudes en su teléfono.

Por otra parte, para asegurarse de que el servicio de notificaciones push no almacene ninguna información sobre lo que contiene la solicitud en sí, todas las solicitudes deben estar encriptadas. Por tanto, esto significa que para enviar una solicitud como notificación push se realizan dos pasos: cifrado y una llamada REST.

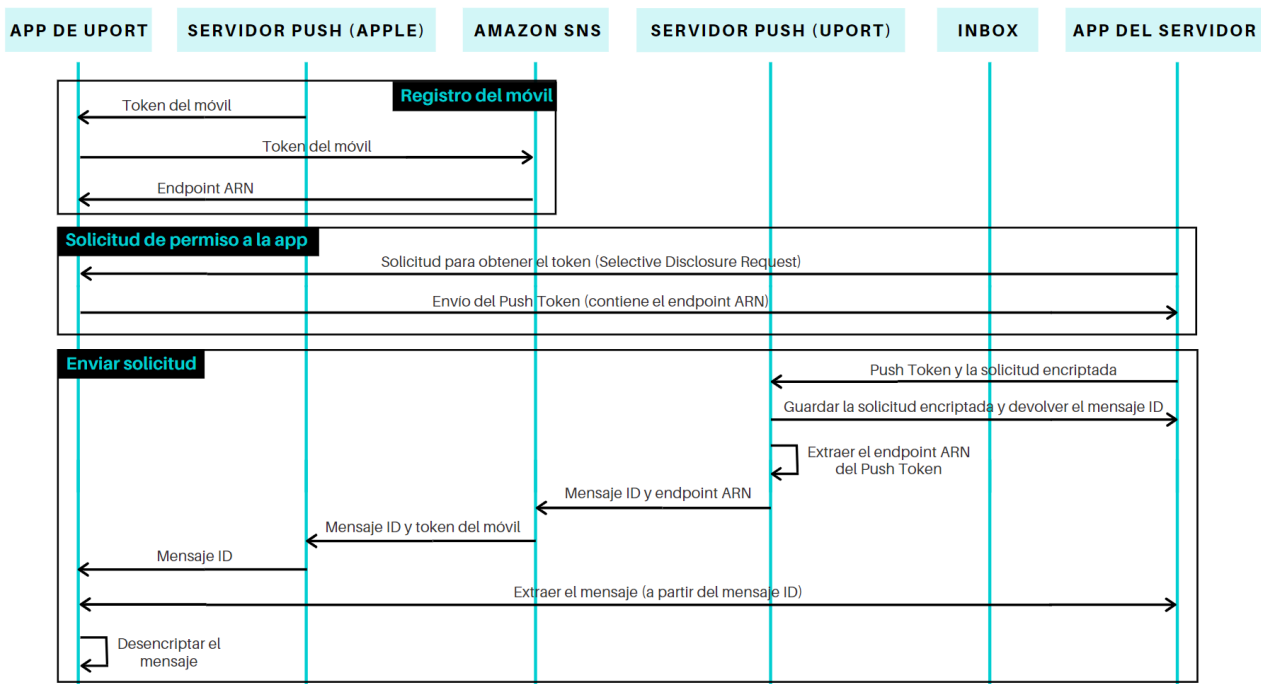


Fig. 4.3 Flujo del envío de notificaciones push a un usuario

1. El servidor de la compañía interesada realiza un Selective Disclosure Flow contra el usuario solicitando permisos de envío de notificaciones utilizando el transporte que desee (normalmente se realiza mediante un código QR).
2. El usuario autoriza el envío de un "PushToken" a la aplicación web de la empresa, así podrá ser utilizado para enviar las notificaciones.

3. La aplicación web de la empresa recibe una respuesta que contiene PushToken.
4. De ahora en adelante, todas las solicitudes futuras se envían primero a un servidor Push mantenido por uPort autenticado utilizando el PushToken como token de portador, esta implementación está basada en el RFC 6750.

4.3.2.2 Respuestas

Hay dos métodos de transporte principales para manejar las respuestas [53] a las solicitudes:

1. **Mediante URL:** La respuesta se pasa a través de un URL y en el destino se parsea. Esta librería proporciona funciones de ayuda para el parseo de estos datos, así como diferentes *listeners* para recibir la respuesta. WARNING: Como ha sido mencionado anteriormente, este método solo puede utilizarse cuando un cliente uPort y la aplicación web de la empresa están en el mismo dispositivo móvil.
2. **Servidor de mensajes:** La respuesta se envía y extrae a través de un servidor de mensajes. Se puede implementar un servidor propio de mensajes, o se puede usar Chasqui por defecto, un servicio de servidor de mensajes proporcionado por uPort.

Alternativamente, sin necesidad de implementar ninguno de los métodos anteriores se pueden recibir respuestas en el propio servidor de la entidad que ofrece el servicio web mediante un callback. Con la finalidad de simplificar la 'Prueba de Concepto' realizada para esta investigación, este último ha sido el mecanismo escogido.

4.3.3 did-JWT y did-RESOLVER

Como consecuencia del uso de las librerías mencionadas anteriormente y su utilización de JWT y DIDs, son necesarias dos más para poder gestionar correctamente estos dos objetos de datos.

En primer lugar, la librería did-JWT [55] permite firmar, decodificar y verificar JWTs utilizando los algoritmos ES256K, ES256K-R y Ed25519. Las claves públicas se obtienen a partir del DID de la identidad que firma el reclamo, que se pasa como un parámetro, llamado iss (issuer), en el JWT codificado.

En segundo lugar, la librería did-RESOLVER [56][49] está pensada como una interfaz simple para permitir que las aplicaciones de Javascript resuelvan documentos DID a partir de identificadores descentralizados.

4.4 Algoritmos criptográficos utilizados

Estas librerías mencionadas anteriormente utilizan diversos algoritmos criptográficos que a continuación, explicaremos brevemente:

4.4.1 Diffie-Hellman

El protocolo de establecimiento de claves Diffie-Hellman [57] fue uno de los primeros protocolos de clave pública concebido por Ralph Merkle y nombrado en honor a Whitfield Diffie y Martin Hellman. Es un método de intercambio seguro de claves criptográficas a través de un canal público e inseguro, realizado de manera anónima (sin previa autenticación) y entre partes que no han tenido contacto previo. La seguridad de este protocolo se debe a la gran dificultad a la hora de calcular logaritmos discretos, por ello se considera un problema sin solución en un espacio de tiempo razonable.

Actualmente, su principal uso es el acuerdo de claves simétricas que posteriormente serán utilizadas para cifrar todos los datos de una sesión, por ello también se las conoce como claves de sesión.

4.4.2 X25529 / CURVE 25519

Curve25519 [58] es una curva elíptica que ofrece 128 bits de seguridad y está diseñada para usarla en conjunto con el protocolo de establecimiento de claves de curva elíptica de Diffie-Hellman, ECDH (Elliptic-Curve Diffie-Hellman). Es una de las curvas ECC (Elliptic Curve Cryptography) más rápidas, se usa para cifrar y no está cubierta por ninguna patente.

En sus inicios, Curve25519 fue definido como una función DH (Diffie-Hellman). Desde entonces, Daniel J. Bernstein propuso que se usara el nombre Curve25519 para la curva subyacente y el nombre X25519 para la función DH, actualmente muchos son los artículos que combinan ambas nomenclaturas.

4.4.3 ES256K-R

uPort utiliza un algoritmo de firma digital de curva elíptica basado en criptografía de curva elíptica. En este caso, la curva elíptica en particular se conoce como ES256K-R [59] (por ahora en fase experimental) y al igual que la anterior, también ofrece 128 bits de seguridad y fue diseñada principalmente para firmar los JWTs. El algoritmo ES256K-R es básicamente igual al ES256K pero con el bit de recuperación agregado, utilizado para extraer la clave pública de la firma.

4.5 Mockups e interfaz de usuario

A la hora de diseñar la interfaz de usuario para la página web se ha priorizado la simplicidad, con la finalidad de que cualquier usuario independientemente de su nivel informático pueda familiarizarse con la aplicación en un breve período de tiempo, ya que la oferta de conciertos y eventos va dirigida a personas con rangos de edades muy diversos. La aplicación se diferencia principalmente en tres funcionalidades:

- Compra de entradas para distintos tipos de eventos
- Intercambio de entradas entre dos usuarios
- Verificación de la validez de una entrada para acceder al concierto

A continuación, se muestran los mockups con el diseño de la interfaz de usuario para cada una de estas funcionalidades. Se puede observar un elemento común en todas las pantallas, la barra de herramientas situada en la parte superior. En el margen izquierdo se encuentra el nombre de la plataforma de venta de entradas y en el margen derecho distinguimos tres pequeños iconos: el primero para acceder a la página de verificación para validar una entrada, el segundo para acceder al carrito de compra y el tercero para acceder a la sección de intercambio de entradas.

4.5.1 Pantalla de inicio

En la pantalla de inicio, se puede encontrar en forma de cuadrícula toda la oferta de eventos existentes, en este caso, se trata de conciertos y, por tanto, lo hemos acompañado con el nombre del artista y el precio del ticket.

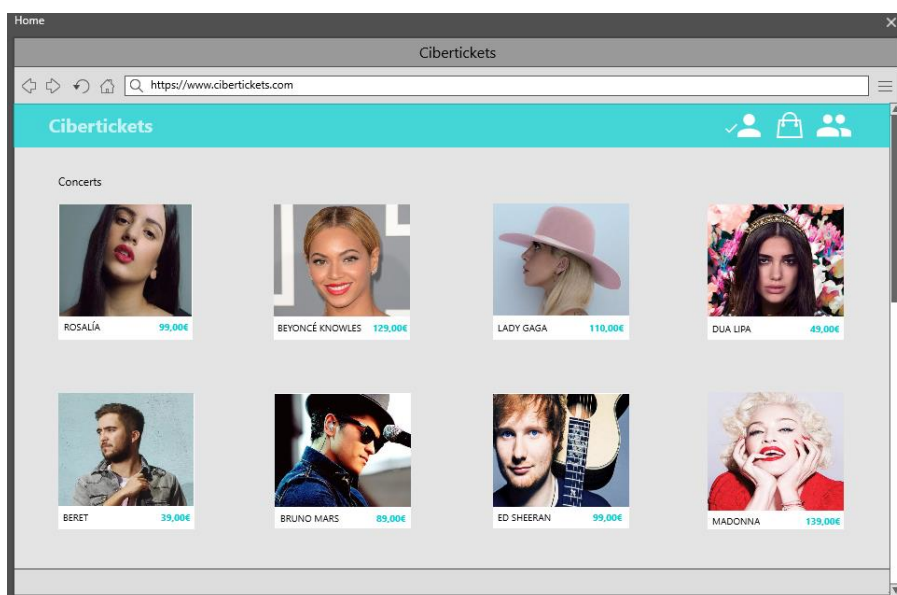


Fig. 4.4 Pantalla principal de todos los conciertos disponibles

4.5.2 Información detallada de un evento

Al seleccionar en el menú anterior un concierto, automáticamente se abre una nueva sección con más detalles del mismo, por ejemplo, sumado al nombre del artista y su precio, aparecen el lugar donde se llevará a cabo, la fecha y la hora.

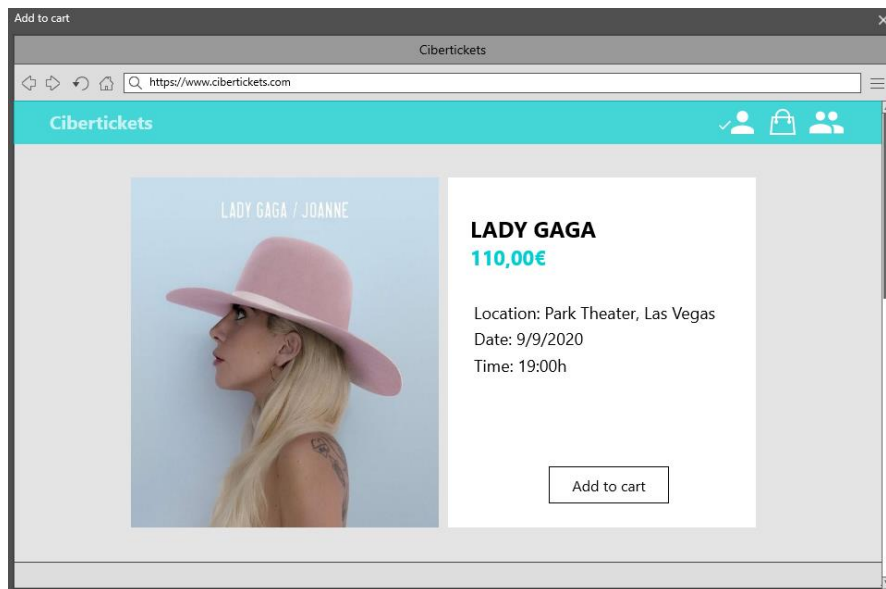


Fig. 4.5 Pantalla de los detalles de un concierto

En segundo lugar, una vez se haga click sobre el botón “Add to cart” (añadir al carrito), se despliega un pop up ocupando toda la pantalla que le permite al usuario dirigirse hacia el carrito para llevar a cabo la confirmación y el pago.

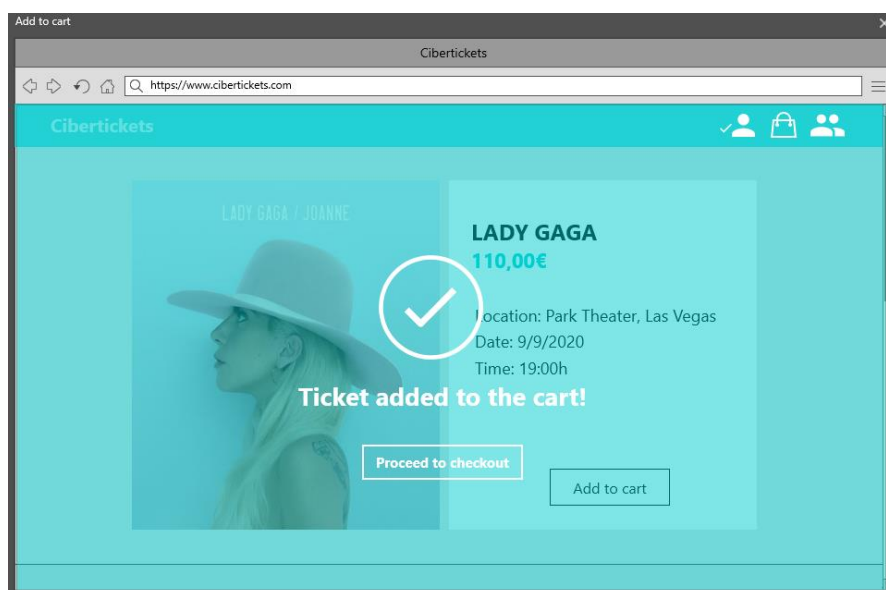


Fig. 4.6 Pantalla de agregación al carrito

4.5.3 Confirmación de compra

Una vez el usuario llega al carrito, se muestran los detalles de la compra que está a punto de realizarse y el precio total de la transacción. En la parte inferior de la pantalla aparece un botón para aceptar, autenticarse y pagar.

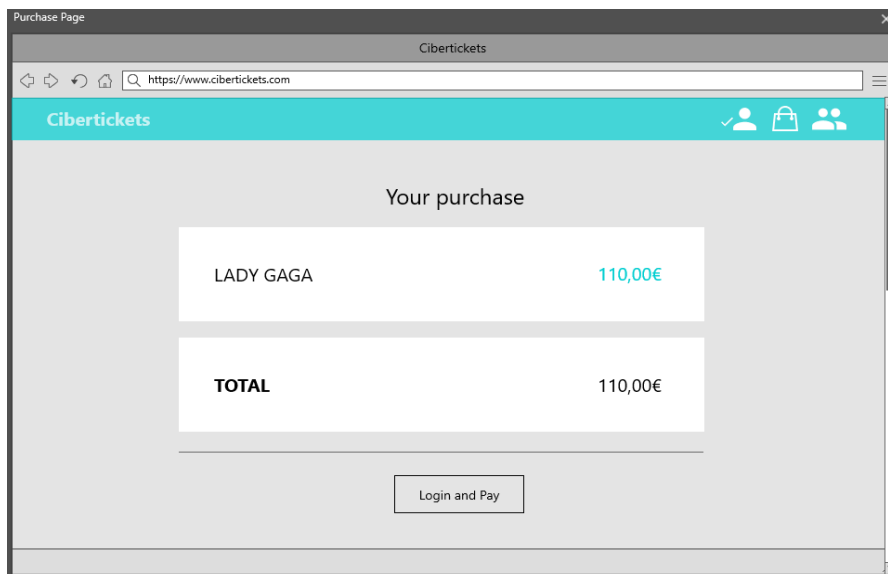


Fig. 4.7 Pantalla de confirmación de compra

4.5.4 Autenticación y pago del propietario

En este último paso, aparece un código QR que se escanea con la app de uPort, se autentica la identidad del usuario y se descarga la entrada en formato digital.

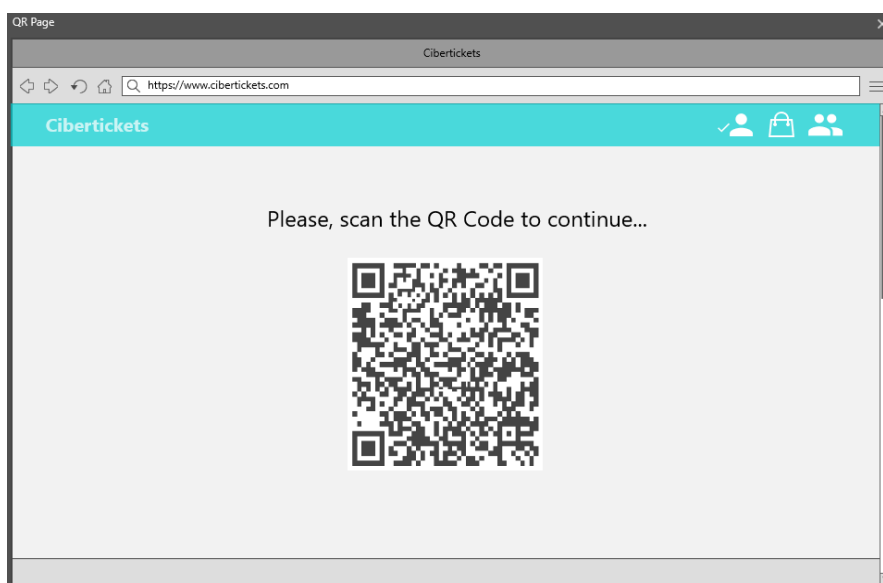


Fig. 4.8 Pantalla para loggarse y pagar mediante el QR

4.5.5 Ver tus entradas compradas

Mediante el tercer icono de la toolbar, el usuario puede introducir su DID en el campo 'Owner's DID' y le aparecerá un listado con todas las entradas que posee.

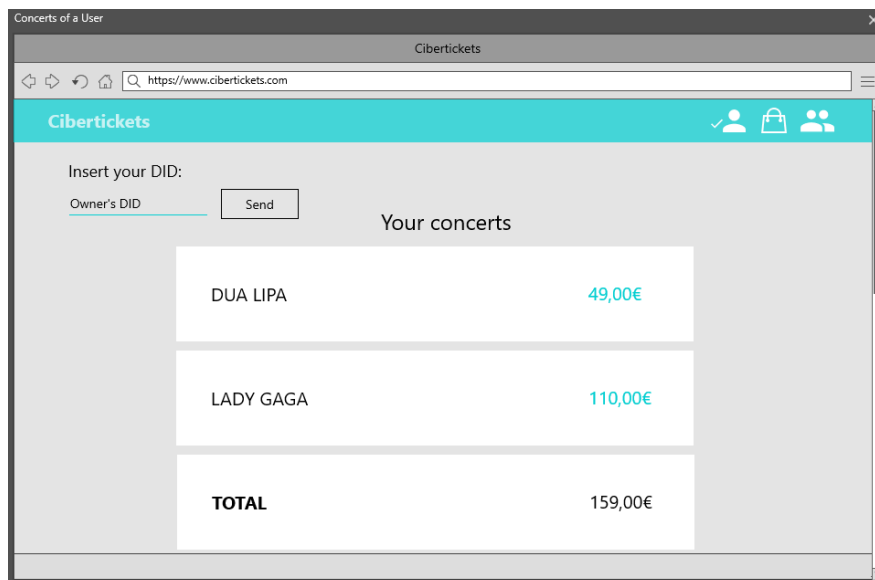


Fig. 4.9 Pantalla de los conciertos comprados por un usuario

4.5.6 Intercambio de una entrada

Al seleccionar una de las entradas de la lista anterior, aparecen los detalles del concierto junto a un campo de texto ('Friend's DID') para introducir el DID del amigo al cual queremos cederle la entrada.

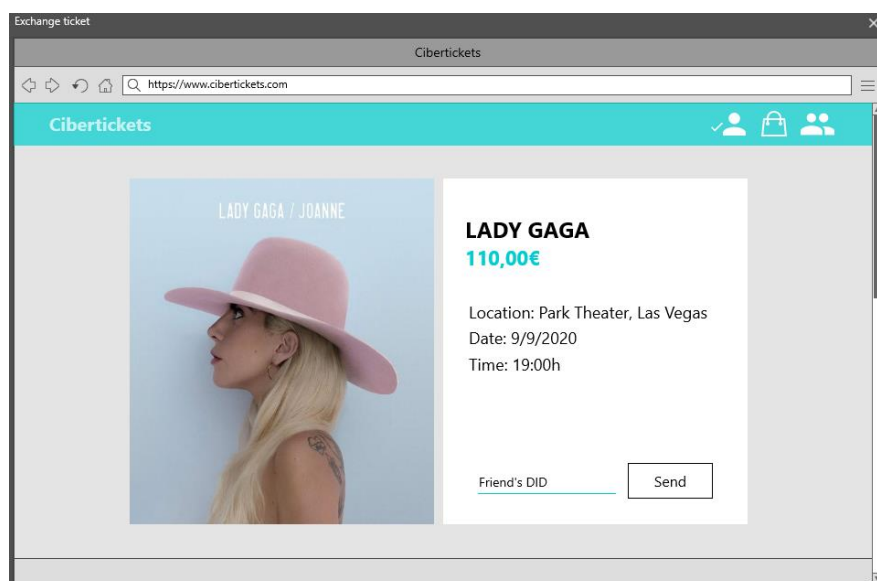


Fig. 4.10 Pantalla de intercambio de entrada con otro usuario

4.5.7 Autenticación y pago del nuevo comprador

En esta última pantalla del flujo de intercambio, aparece un QR para que el nuevo dueño de la entrada lo escanee y de esta manera se autentica la identidad del nuevo usuario, se descarga la entrada en formato digital y se invalida la entrada del propietario inicial, para evitar que pueda acceder al evento el día indicado.

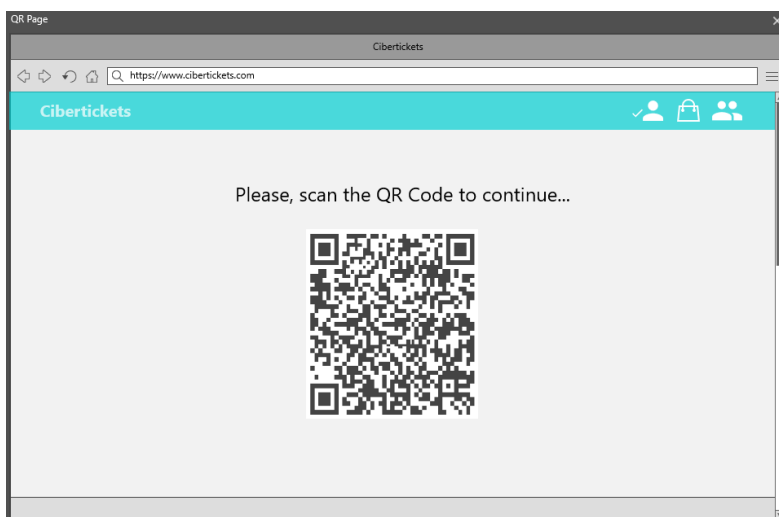


Fig. 4.11 Pantalla para confirmar el intercambio mediante el QR

4.5.8 Verificación de la validez de una entrada

Finalmente, para acceder al último flujo debemos clicar en el primer icono de la barra superior, se generará un código QR para que el usuario lo escanee, de esa manera se verifica la validez de la entrada y aparece junto al QR una notificación permitiéndole o negándole el acceso al evento.

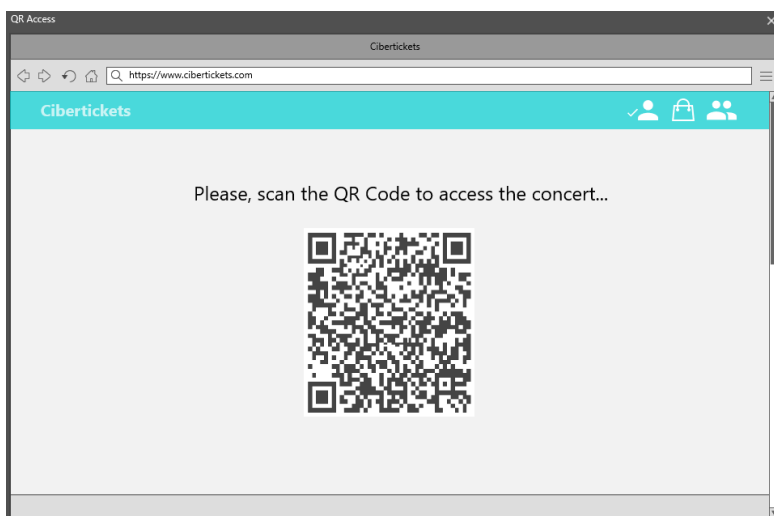


Fig. 4.12 Pantalla para comprobar la validez mediante el QR

4.6 Implementación

El primer paso para crear cualquier solución del lado del servidor con uPort-credentials es obtener una identidad para la aplicación de Cibertickets, para poder firmar peticiones y verificar datos. Además, cada usuario de uPort está identificado con un DID único asociado a la aplicación móvil.

4.6.1 Compra de entradas

El primer flujo de Cibertickets consiste en la compra de una entrada para un concierto y sigue los pasos descritos a continuación (todos realizados por parte del usuario):

- Escoger el concierto deseado de la lista del menú principal.
- Acceder al detalle del concierto y añadirlo al carrito (en esta Prueba de Concepto, solo se permite comprar una única entrada).
- Entrar en el carrito, donde veremos un resumen de nuestra compra y posteriormente, mediante el botón 'Login and Pay' dirigirnos al siguiente paso.
- Cuando aparezca el QR, escanearlo con la app de uPort de nuestro dispositivo móvil para permitir la compartición de datos con Cibertickets y proceder al inicio de sesión.
- Una vez que Cibertickets obtiene del usuario los datos necesarios, procede al envío de la entrada en formato digital, que el usuario puede aceptar o declinar.

Mediante el siguiente diagrama de flujo [60], vamos a observar en detalle la tecnología utilizada y como se relaciona el frontend de Cibertickets con la app de uPort del dispositivo móvil del usuario. La finalidad principal es solicitar datos verificados y el cliente de uPort, que representa una identidad digital soberana, debe aprobar la solicitud de divulgación selectiva (Selective Disclosure Request), este mecanismo es el medio principal para validar las credenciales de un usuario. Una vez satisfecha la lógica del lado del servidor con los datos obtenidos, se puede considerar que un usuario está autenticado gracias a las credenciales verificadas que divulgó y por ello, se le pueden expedir las entradas.

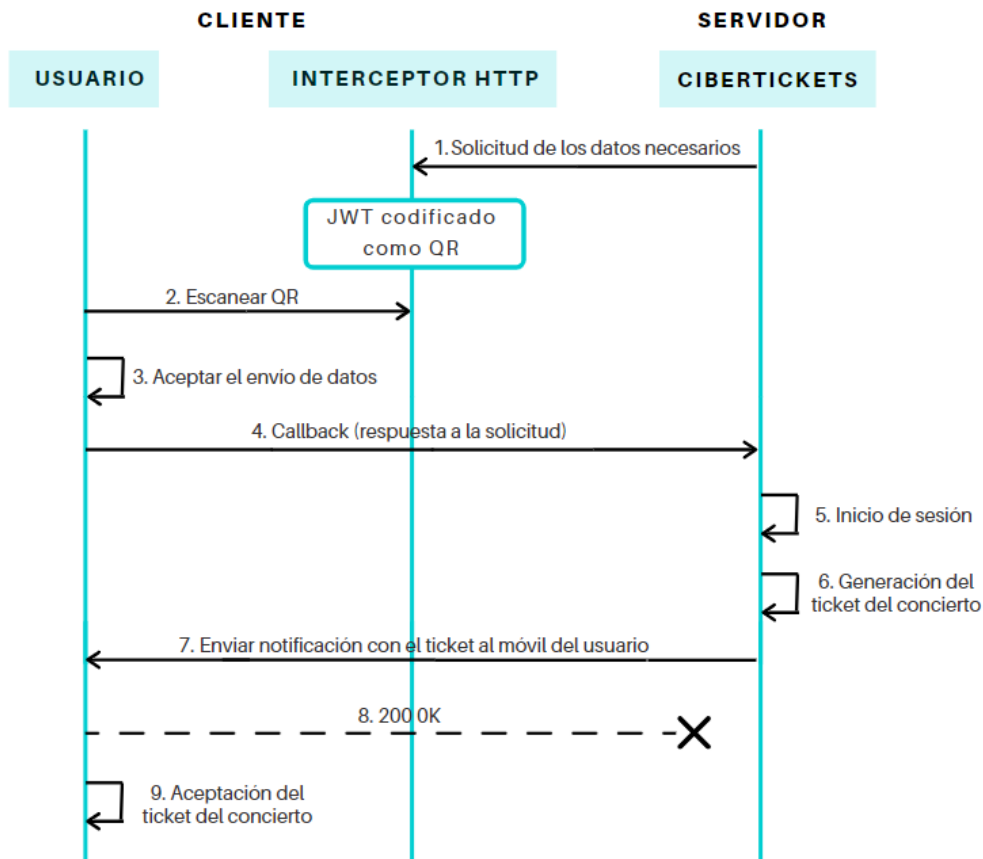


Fig. 4.13 Esquema de la compra de una entrada para un concierto

A grandes rasgos, podríamos decir que generar estas entradas o cualquier tipo de verificación, ayuda al usuario a construir su propia identidad digital y añade valor real a tu empresa asegurando que tus usuarios no son bots sino que son realmente personas físicas. Para realizar este proceso, el servidor de Cibertickets debe firmar criptográficamente los datos de usuario con el nombre de su aplicación y enviar verificaciones como un JWT a sus usuarios a través de un QR o mediante una notificación push. Como resultado, Cibertickets da veracidad de que los datos de ese usuario son correctos y, por tanto, cualquiera que tenga acceso al DID de Cibertickets puede verificar que la entrada, en este caso, proviene realmente de la empresa en cuestión y no corresponde a una falsificación.

Servicio de inicio de sesión, solicitud, envío de datos y creación del ticket

En primer lugar, el servicio de inicio de sesión con solicitud de datos para posteriormente generar el ticket incluye dos partes, que corresponden con dos endpoints diferenciados:

4.6.2 Intercambio de entrada

El segundo flujo de Cibertickets consiste en el intercambio de una entrada entre dos usuarios y sigue los pasos descritos a continuación (este apartado se realiza exclusivamente usando la aplicación web, no requiere interacción con la aplicación de uPort en el móvil, por tanto, se trata de un intercambio de datos entre el servidor y el navegador web):

- En este caso, el usuario que desea intercambiar la entrada con un amigo debe acceder al apartado de intercambio de entrada de la web de Cibertickets, introducir su DID para obtener una lista de las entradas que posee.
- Posteriormente, selecciona la entrada que quiera intercambiar, le aparecen más detalles del concierto y un cuadro de texto para introducir el DID del amigo al que le cede su entrada.
- Una vez introducido, se muestra un QR que el amigo debe escanear con la app de uPort, en este punto la entrada del usuario inicial deja de ser válida y pasa a ser legítima para su amigo, que pasará a tener la entrada digital autorizada en su dispositivo móvil (este último paso es igual al proceso de compra de entradas del apartado anterior)

Mediante el siguiente diagrama de flujo, podemos observar en detalle la tecnología utilizada y como se relaciona el frontend y el backend de Cibertickets.

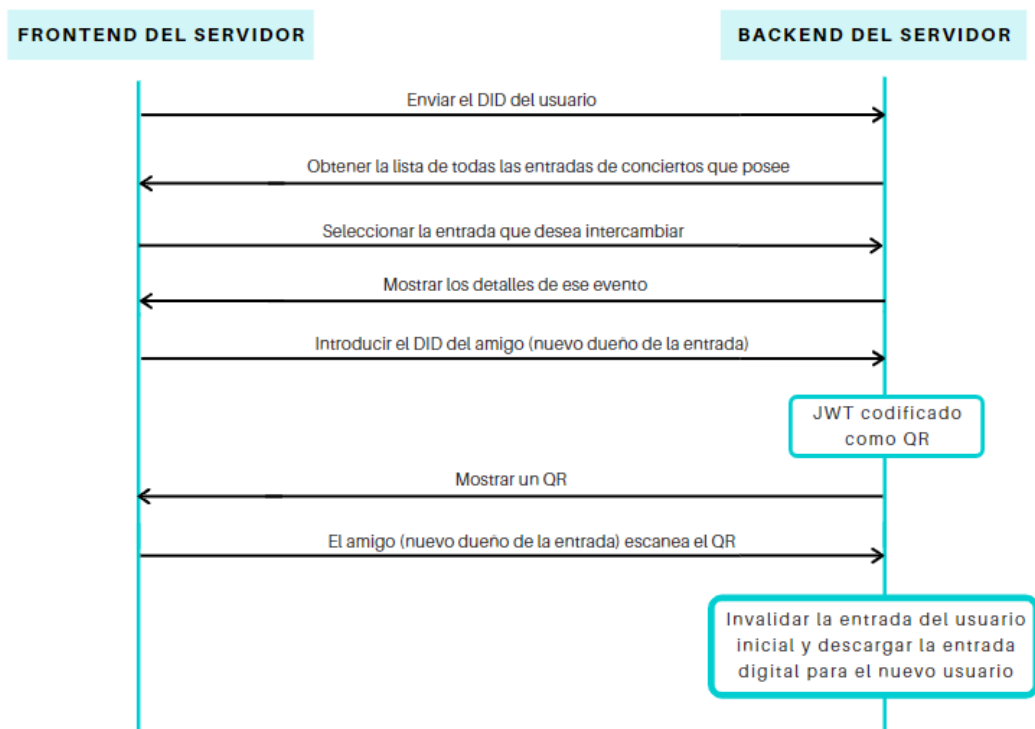


Fig. 4.17 Esquema del intercambio de una entrada para un concierto

4.6.3 Verificación de la entrada para el acceso al concierto

El tercer flujo de Cibertickets consiste en la verificación de una entrada para acceder al concierto y sigue los pasos descritos a continuación (este apartado requiere la interacción de un usuario y un miembro de seguridad para controlar el acceso):

- En este caso, el personal de seguridad encargado de controlar al acceso al recinto del evento, debe acceder al apartado de verificación de la web de Cibertickets, en ella aparecerá un QR.
- El usuario debe escanear el QR con la app de uPort, que automáticamente detecta la entrada que se desea verificar.
- El usuario introduce el código del dispositivo o bien se autentica mediante FaceID o huella dactilar.
- Se comprueba que todos los campos del ticket sean válidos, especialmente la validez y se permite o se deniega el acceso de ese cliente al concierto.

Mediante el siguiente diagrama de flujo [61], observaremos en detalle la tecnología utilizada y como se relaciona el frontend de Cibertickets con la app de uPort del dispositivo móvil del usuario. Podemos ver como el proceso de solicitar una entrada para poderla verificar es bastante similar al de crear la entrada (apartado 4.6.1), ya que, en ambos casos, solicitas información al usuario y compruebas su veracidad en el servidor de Cibertickets.

La principal novedad de esta parte es que la solicitud inicial ya no será una solicitud de inicio de sesión genérica, sino una solicitud de un conjunto específico de datos verificados, en este caso la entrada.

Para realizar este proceso, al igual que anteriormente, el servidor de Cibertickets debe firmar criptográficamente con el nombre de su aplicación una solicitud para obtener los datos verificados de un usuario (mediante esta firma el usuario puede confiar en que realmente la aplicación legítima de Cibertickets está realizando esa petición) y, posteriormente, enviar estas solicitudes como un JWT a sus usuarios a través de un QR o mediante una notificación push.

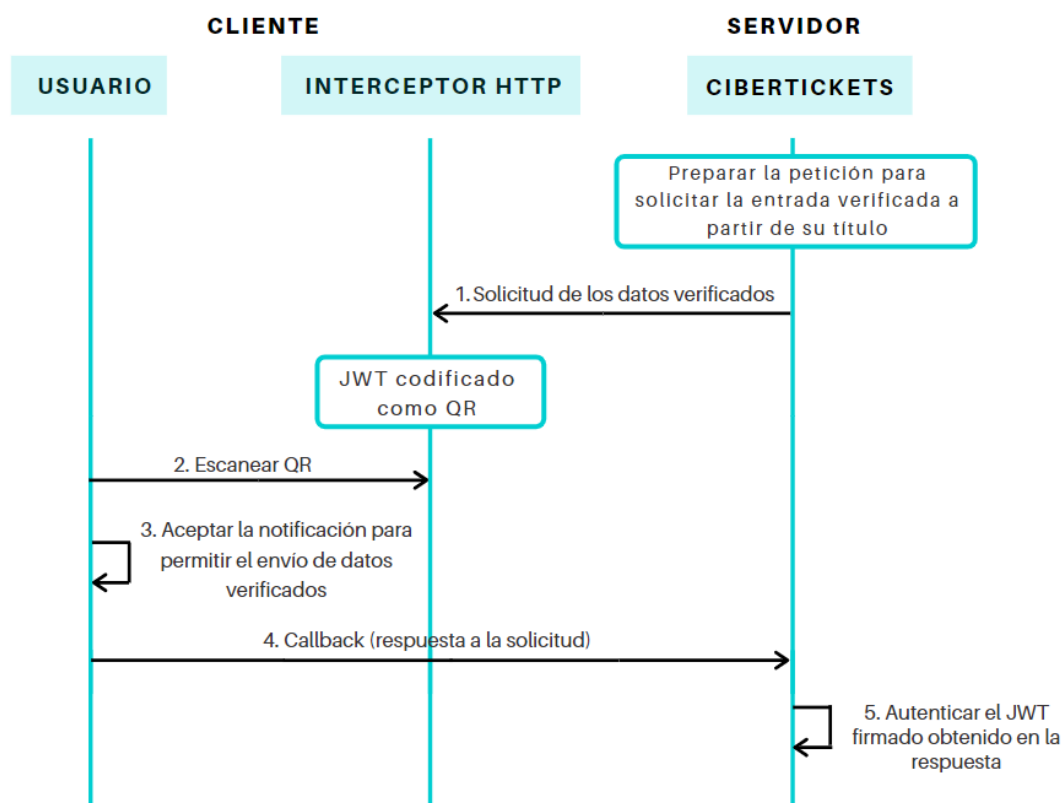


Fig. 4.18 Esquema de la verificación de una entrada para un concierto

Servicio de solicitud de entradas verificadas

El servicio de solicitud de entradas verificadas para comprobar su validez y permitir o denegar al acceso de un usuario al evento, está formado por dos endpoints diferenciados:

- El primero, escucha las peticiones entrantes y se encarga de crear un mensaje para solicitar al usuario las credenciales verificadas, a partir de un identificador único que se corresponde con el título. Lo encapsula como un JWT firmado por Cibertickets (*createDisclosureRequest*) (1). Para hacer llegar esta solicitud al usuario se hace uso de la librería *uport-transport*, que convierte el JWT en una URI (*paramsToQueryString*) y después en un QR (*getImageDataURI*) para que la aplicación móvil de uPort pueda consumirlo mediante su escaneo (2). Al escanearlo, al usuario le aparecerá una alerta informándole de que está a punto de compartir información verificada, en este caso la entrada, este puede aceptar o declinar. En el caso de nuestra aplicación, este es el formato del JWT, que incluye el parámetro 'verified' (vector que contiene el título identificativo de la credencial solicitada, según este valor se solicitarán al usuario unos datos u otros), en nuestro ejemplo vemos como es

denominado 'Concert', ya que es el nombre escogido durante el proceso de creación de la entrada (Fig. 4.16). Además, también se especifica mediante un atributo que URL actuará como callback, es decir donde se enviará la respuesta que dé el usuario (3), tanto si acepta como si declina la solicitud. En el caso de aceptar, se enviarán la entrada requerida por Cibertickets.

```
{
  header: { typ: 'JWT', alg: 'ES256K-R' },
  payload: {
    iat: 1598789506,
    exp: 1598790106,
    verified: [ 'Concert' ],
    callback: 'https://0cd2988cd87b.ngrok.io/verify',
    type: 'shareReq',
    iss: 'did:ethr:0x623fe050f3873b5c540ed97a8a6006ebc34db1d1'
  },
  signature: 'E2H3_C55tX9Ba2o1DrQJycVCuR81i7nwqNTomu2bCrAumMKkm2AV8Jpeaiw9FjVKTe6IU2E171fQNaCNnyEqEwE',
  data: 'eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiIsInR5cGU6IjY9LmVjYXQiOiJlOTg3ODk1MDYsImV4cCI6MTU5ODc5MDUwNTBmMzgzM2I1YzU0MGVkb0tdh0GE2MDA2ZWJjMzRkYjFkMSJ9'
}
```

Fig. 4.19 Parámetros de la solicitud de la entrada

- El segundo, representa el callback (URL definida en el paso anterior) para recibir la respuesta (4) a la solicitud previa de la entrada, es decir, en este punto Cibertickets conoce si el usuario ha accedido a compartir su ticket verificado o no. Una vez recibido el JWT de la respuesta se autenticará (*authenticateDisclosureResponse*) mediante la verificación de la firma del payload del JWT, una vez verificado se comprueba la validez (5) de la entrada y si todo ha ido correcto se permite el acceso al concierto o, por el contrario, se deniega. En el caso de nuestra aplicación, este es el formato del JWT donde se obtiene la entrada del concierto verificada, en este caso identificada como 'Concert', como ya se mencionó anteriormente. Además de la información del propio concierto, también se incluyen campos como 'exp', que corresponde a la fecha de expiración o el DID del usuario que pretende acceder al evento.

CONCLUSIONES Y LÍNEAS FUTURAS

La seguridad en la red es actualmente uno de los temas más complejos de la ingeniería de redes y existen muchos profesionales dedicados a mejorar los sistemas actuales. A lo largo de este Trabajo de Final de Grado hemos podido observar cómo se empieza a desarrollar y a implementar los fundamentos de la identidad digital soberana, demostrando así que un cambio de paradigma es posible.

El principal objetivo era analizar el funcionamiento y las ventajas de la identidad digital soberana sobre la blockchain, así como poner en práctica todos estos conocimientos para programar un sistema seguro de compra de entradas para conciertos y así comprobar de manera empírica todos los beneficios teóricos. Ambos objetivos han sido cumplidos con éxito y podemos decir que, gracias a nuestra solución, Cibertickets, hemos erradicado gran parte de los conflictos actuales.

En primer lugar, al permitir la compra únicamente de una entrada por usuario, desincentivamos la reventa de entradas, ya que deja de existir el gran beneficio económico que se conseguía con múltiples entradas. Por otro lado, al tratarse de un proceso manual que requiere un dispositivo móvil para escanear un código QR, evitamos el uso de bots y nos aseguramos de que son personas físicas quienes acceden a nuestra plataforma. Finalmente, cada usuario gestiona todos sus datos personales en todo el proceso y frente a Cibertickets simplemente se identifican mediante un DID anónimo de Ethereum, por tanto, la empresa no debe guardar datos sensibles de los usuarios y evita así, cumplir con la estricta normativa de la GDPR.

Me siento muy afortunada de haber podido trabajar en este proyecto ya que, gracias a él, he aprendido una nueva forma de enfocar la tecnología capaz de revolucionar las relaciones, tanto comerciales como sociales, teniendo múltiples aplicaciones para evitar distintos delitos cibernéticos.

Debido al gran interés que me ha causado este tema, de cara al futuro, me gustaría seguir con detalle su evolución y poder implementar en la plataforma de Cibertickets dos nuevas funcionalidades:

- Pasarela de pago en el proceso de intercambio de entrada, estableciendo un precio máximo de venta (igual al valor nominal de la entrada), para así fomentar aún menos la reventa de entradas.
- Habiendo conseguido el punto anterior, ampliar la cantidad de entradas que puede adquirir un usuario, fijando un límite máximo por DID, ya que de todas maneras no podrá revenderlas a un precio mayor

BIBLIOGRAFÍA

- [1] Vescent, H., Young, K., Duffy, H. K., Sabadello, M., Zagidulin, D., & Caballero, J., *A Comprehensive Guide to Self Sovereign Identity*, The Purple Tornado, California (2019).
- [2] Birch, D. & Conway, E., *Identity is the New Money*, London Publishing Partnership, Londres (2014).
- [3] Mohanty, D., *Blockchain for Self Sovereign Digital Identity*, Goodreads, California (2020).
- [4] *¿Qué datos personales se consideran sensibles?*, Comisión Europea (2018) https://ec.europa.eu/info/law/law-topic/data-protection/reform/rules-business-and-organisations/legal-grounds-processing-data/sensitive-data/what-personal-data-considered-sensitive_es
- [5] Mühle, A., Grüner, A., Meinel, C. Gayvoronskaya, T., *A Survey on Essential Components of a Self-Sovereign Identity*, Hasso Plattner Institute (2018) <https://arxiv.org/pdf/1807.06346.pdf>
- [6] *What is self-sovereign Identity?*, Sovrin (2018) <https://sovrin.org/faq/what-is-self-sovereign-identity/>
- [7] *Digital Certificates Project*, Massachusetts Institute of Technology – MIT (2016) <https://certificates.media.mit.edu/>
- [8] *¿Cómo funciona Bitcoin?*, Bitcoin (2018) <https://bitcoin.org/es/como-funciona>
- [9] Ruff, T., *The Three Models of Digital Identity Relationships*, Medium (2018) <https://medium.com/evernym/the-three-models-of-digital-identity-relationships-ca0727cb5186>
- [10] *What is OAuth and why does it matter?*, OAuth Community Site (2018) <https://oauth.net/>
- [11] *Welcome to OpenID Connect*, OpenID Connect (2020) <https://openid.net/connect/>
- [12] P., *HIPAA vs COPPA vs GDPR – Data Compliances For Mobile Apps*, BlueWhaleApps (2020) <https://bluewhaleapps.com/blog/hipaa-vs-coppa-vs-gdpr>
- [13] *Digital Identity*, Digital Identity Systems, Inc (2020) <https://id2020.org/digital-identity>
- [14] Pro, T., *Reclaiming our digital identity*, TechRadar (2019) <https://www.techradar.com/news/reclaiming-our-digital-identity>

- [15] *Qué es la Identidad Soberana*, Bit2Me Academy (2020) <https://academy.bit2me.com/que-es-la-identidad-soberana/>
- [16] *ForgeRock U.S Consumer Data Report*, ForgeRock Press (2018) <https://www.forgerock.com/about-us/press-releases/forgerock-us-consumer-data-breach-report-data-breaches-cost-654-billion>
- [17] *About uPort*, uPort (2018) <https://developer.uport.me/overview/index>
- [18] Rosic, A., *What is Blockchain Technology? A Step-by-Step Guide For Beginners*, Blockgeeks (2020) <https://blockgeeks.com/guides/what-is-blockchain-technology/>
- [19] *A Deep Dive on End-to-End Encryption: How Do Public Key Encryption, Surveillance Self-Defense* (2019) <https://ssd.eff.org/en/module/deep-dive-end-end-encryption-how-do-public-key-encryption-systems-work>
- [20] Kohlhaas, P., *Zug ID: Exploring the First Publicly Verified Blockchain Identity*, Medium (2018) <https://medium.com/uport/zug-id-exploring-the-first-publicly-verified-blockchain-identity-38bd0ee3702>
- [21] Rutter, C., *The Case for Self-Sovereign Identity*, R3 (2020) <https://www.r3.com/blog/the-case-for-self-sovereign-identity/>
- [22] Sena, M., *Privacy Preserving Identity System for Ethereum dApps*, Medium (2018) <https://medium.com/uport/privacy-preserving-identity-system-for-ethereum-dapps-a3352d1a93e8>
- [23] *JSON Web Tokens Introduction*, JWT (2017) <https://jwt.io/introduction/>
- [24] *Selective Disclosure Flow*, uPort (2019) <https://developer.uport.me/flows/selectivedisclosure>
- [25] *Government Issued Blockchain Identity: Zug Case Study*, ConsenSys (2019) <https://consensys.net/blockchain-use-cases/government-and-the-public-sector/zug/>
- [26] *Five of the Companies Regularly Used Illegal Bots To Procure Tickets For Sale On The Secondary Market*, New York State Office (2017) <https://ag.ny.gov/press-release/2017/ag-schneiderman-announces-419-million-settlements-six-companies-illegally>
- [27] *Ticket Bots: Everything You Need to Know*, Queue-it (2020) [https://queue-it.com/blog/ticket-bots/#:%7E:text=A%20bot%20\(short%20for%20%E2%80%9Crobot,released%20seats%2C%20or%20purchasing%20tickets](https://queue-it.com/blog/ticket-bots/#:%7E:text=A%20bot%20(short%20for%20%E2%80%9Crobot,released%20seats%2C%20or%20purchasing%20tickets)

- [28] Roberts, E., *Bad Bot Report 2020: Bad Bots Strike Back*, Imperva (2020) <https://www.imperva.com/blog/bad-bot-report-2020-bad-bots-strike-back/>
- [29] *Ticketmaster says Bot Army bought 30.000 'Hamilton' tickets*, Variety (Gene Maddaus) (2017) <https://variety.com/2017/digital/news/ticketmaster-hamilton-prestige-entertainment-renaissance-ventures-1202578292/>
- [30] *The History of the General Data Protection Regulation*, European Data Protection Supervisor (2017) https://edps.europa.eu/data-protection/data-protection/legislation/history-general-data-protection-regulation_en#:~:text=In%202016%2C%20the%20EU%20adopted,as%20law%20across%20the%20EU
- [31] *GDPR - Glossary of terms and definitions*, Legal Services (2018) <https://www.ucl.ac.uk/legal-services/gdpr-glossary-terms-and-definitions#:~:text=The%20GDPR%20replaces%20the%20previous,introducing%20new%20concepts%20and%20terminology.&text=Data%20subject%3A%20a%20natural%20person,a%20data%20controller%20or%20processor>
- [32] *The principles*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/principles/>
- [33] *Right to be informed*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-to-be-informed/>
- [34] *Right of access*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-of-access/>
- [35] *Right to rectification*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-to-rectification/>
- [36] *Right to erasure*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-to-erasure/>
- [37] *Right to restrict processing*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-to-restrict-processing/>
- [38] *Right to data portability*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-to-data-portability/>

- [39] *Right to object*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-to-object/>
- [40] *Right related to automated decision making including profiling*, ICO (2018) <https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/rights-related-to-automated-decision-making-including-profiling/>
- [41] *Fines / Penalties*, General Data Protection Regulation (GDPR) (2020) <https://gdpr-info.eu/issues/fines-penalties/>
- [42] *Advantages of Mongodb*, MongoDB Company (2017). <https://www.mongodb.com/advantages-of-mongodb>
- [43] *The Good and the Bad of Angular Development*, AltexSoft (2020) <https://www.altexsoft.com/blog/engineering/the-good-and-the-bad-of-angular-development/>
- [44] *NodeJS*, Desarrollo Web (2016) <https://desarrolloweb.com/home/nodejs>
- [45] Volodymyr, T., *Express.js Mobile App Development: pros and cons of Node.js framework*, Apiko (2020) <https://apiko.com/blog/express-mobile-app-development/>
- [46] *What is Git: become a pro at Git with this guide*, Atlassian Git (2018) <https://www.atlassian.com/git/tutorials/what-is-git>
- [47] *Design, Prototype & Collaborate better and faster*, Mockplus (2019) <https://www.mockplus.com/>
- [48] Aguilar, J. M., *¿Qué es el patrón MVC en programación y por qué es útil?*, CampusMVP (2019) <https://www.campusmvp.es/recursos/post/que-es-el-patron-mvc-en-programacion-y-por-que-es-util.aspx>
- [49] *uPort Credentials*, uPort Developer (2019) <https://developer.uport.me/uport-credentials/index>
- [50] *uPort Credentials Library*, uPort Developer (2019) <https://developer.uport.me/uport-credentials/reference/index>
- [51] *uPort Transports*, uPort Developer (2019) <https://developer.uport.me/uport-transports/index>
- [52] *uPort Transports Library*, uPort Developer (2019) <https://developer.uport.me/uport-transports/reference/index>

- [53] *uPort Transports Requests and Response*, uPort Developer (2019) <https://developer.uport.me/uport-transport/guides/modules>
- [54] *Request/Response Transports*, uPort Developer (2019) <https://developer.uport.me/transport/index>
- [55] *did-JWT Library*, GitHub (2020) <https://github.com/decentralized-identity/did-jwt#:~:text=The%20did%2DJWT%20library%20allows,attribute%20of%20the%20encoded%20JWT>
- [56] *did-RESOLVER Library*, GitHub (2020) [https://github.com/decentralized-identity/did-resolver#:~:text=Javascript%20DID%20Resolver,from%20Decentralized%20Identifiers%20\(DIDs\).&text=The%20library%20does%20not%20implement,packages%20that%20applications%20can%20add](https://github.com/decentralized-identity/did-resolver#:~:text=Javascript%20DID%20Resolver,from%20Decentralized%20Identifiers%20(DIDs).&text=The%20library%20does%20not%20implement,packages%20that%20applications%20can%20add)
- [57] Lake, J., *What is the Diffie–Hellman key exchange and how does it work?*, Comparitech (2019) <https://www.comparitech.com/blog/information-security/diffie-hellman-key-exchange/>
- [58] Chain, C., *A Deep Dive into X25519*, CoinEx Chain - Medium (2019) <https://medium.com/@CoinExChain/a-deep-dive-into-x25519-7a926e8a91c7>
- [59] *EcdsaSecp256k1RecoverySignature2020*, GitHub (2020) <https://github.com/decentralized-identity/EcdsaSecp256k1RecoverySignature2020>
- [60] *uPort Credentials – Create Verification Example*, uPort Developer (2019) <https://developer.uport.me/credentials/createverification>
- [61] *uPort Credentials – Request Verification Example*, uPort Developer (2019) <https://developer.uport.me/credentials/requestverification>