Universidad de Valladolid

ESCUELA DE INGENIERÍA INFORMÁTICA

GRADO EN INGENIERÍA INFORMÁTICA
MENCIÓN EN TECNOLOGÍAS DE LA INFORMACIÓN

TRABAJO FIN DE GRADO

KITPRIV: TOOLKIT DE ANONIMIZACIÓN EN BASES DE DATOS RELACIONALES

Alumno:
Adrián de la Torre Villota

Tutora:
Mercedes Martínez González

Agradecimientos

Me gustaría dedicar unas palabras a todas aquellas personas que me han apoyado durante estos años de estudios universitarios. A mi tutora Mercedes, por su entrega y apoyo durante el desarrollo de este proyecto. A todos los profesores que me han transmitido sus conocimientos en mis cuatro años de formación académica. A mis compañeros de carrera, por ayudarme en aquellas cosas que no comprendía y hacerme entretenidos estos años. A mi familia, por su cariño y aguante en los momentos más complicados.

Resumen

La anonimización de datos, también conocido como ofuscación o enmascaramiento, es el proceso por el cual se consigue ocultar o enmascarar la identidad de los individuos a los cuales se refieren esos datos. Esto es particularmente interesante en el caso de datos sensibles o confidenciales, aquellos cuyo conocimiento podría afectar a los derechos de los individuos o a los intereses comerciales de las empresas u organizaciones.

En este trabajo se ha desarrollado un conjunto de procedimientos y funciones que implementan técnicas de anonimización y ofuscación sobre bases de datos relacionales. El fin de estas herramientas, es que mediante su utilización, otras personas puedan descubrir el funcionamiento de estas técnicas y aprender a utilizarlas. El desarrollo se ha inspirado en los procedimientos y funciones que proporcionan otras tecnologías de pago. Se ha prestado particular atención a los principios básicos de usabilidad, ya que se pretende utilizarlo con estudiantes.

Abstract

Data anonymization, also known as obfuscation or data masking, is the process by which it is possible to hide or mask the identity of individuals to whom these data refer. This is particularly interesting when working with sensitive or confidential data; those whose knowledge could affect / impact the rights of individuals or the commercial interests of companies and organizations.

In this project a series of procedures and functions have been developed to help with the implementation of anonymization and obfuscation techniques on relational data bases. The key objective is, that by using these techniques, other people can discover how the techniques work and learn how to best use them. The development has been inspired in the procedures and functions provided by other existing technologies that can be purchased in the market. Focus has been put mainly on the area of usability basic principles, as the intend is that other students can use it and benefit from it.

Índice general

Capítulo 1

Introducción

1.1. Contexto

En la actualidad, la gran mayoría de los sectores (médico, de información, bancario... etc) basa su actividad en datos de distinta índole. Su explotación es básica, ya sea para conocer la información de un paciente en un hospital, para realizar transacciones en un banco o algo tan común como realizar compras de forma física u online.

Para su uso, es necesario que sean generados, transmitidos a través de distintos canales y en última instancia, almacenados. Uno de los principales problemas que aparecen en este ciclo de vida de un dato es el grado de privacidad con que se deben tratar, ya que muchos de ellos son información personal que, de caer en malas manos, podría derivar en graves problemas.

Para proteger y evitar comprometer este tipo de información, existen distintas normativas dependiendo de los países y en concreto, en la Unión Europea se estableció el RGPD, Reglamento de Protección de Datos, en mayo de 2018 [1]. Este reglamento recoge todos los puntos a seguir para el correcto tratamiento de datos personales y su libre circulación.

Para su protección, como indica la AEPD [2], se pueden diferenciar tres enfoques generales para la anonimización de datos, cada uno de ellos está integrado a su vez por diversas técnicas:

- **Aleatorización:** "el tratamiento de los datos consiste en la eliminación de correlación con el individuo, mediante la adición de ruido, la permutación o la Privacidad

Diferencial."

- **Generalización:** "el tratamiento de los datos consiste en la alteración de escalas u órdenes de magnitud a través de técnicas basadas en agregación como K-Anonimato, Diversidad-L, o Proximidad-T."

- **Seudonimización:** "el tratamiento de los datos consiste en el reemplazo de valores por versiones cifradas o tokens, normalmente a través de algoritmos de HASH, que impiden la identificación directa del individuo, a menos que se combine con otros datos adicionales, que, como medida importante, deben estar custodiados de forma adecuada."

[En posteriores capítulos se hará una distinción más profunda entre seudonimización y anonimización. A pesar de estar este primero incluido como un enfoque de la anonimización y causar confusiones, en realidad son términos distintos.]

Hoy en día, uno de los lugares donde se pone en valor este tipo de seguridad es en las bases de datos (donde es habitual guardar la información). Estas, están pobladas por grandes cantidades de datos, muchos de ellos considerados sensibles y por tanto deben estar protegidos ya que cada vez más, se están produciendo violaciones, malos usos y robos de estos.

En este TFG se desarrollarán varios procedimientos y funciones de anonimización de datos, los cuales se basan en algunas de las técnicas especificadas en el apartado de "Técnicas de anonimización de datos", ya que se tratará de reemplazar datos por versiones cifradas, datos sintéticos, cambio de caracteres o adición de ruido a estos.

1.2. Motivación

Los avances tecnológicos y el conocimiento sobre estos (especialmente en delincuentes) está creciendo cada vez más y generando dificultades a la hora de tener protegida la información en las bases de datos. Es por eso, que hoy en día se está intentado descubrir nuevas formas de ofuscar datos, ya que la anonimización absoluta es compleja de garantizar y la reidentificación puede llegar a ser posible.

Las técnicas de anonimización y enmascaramiento de datos nacen con los objetivos de, no solo ocultar parte del dato para prevenir que sea visible a ojos de personas no autorizadas, si no que se debe evitar una posible asociación de los datos anonimizados con otros identificativos de la persona afectada, este hecho también conocido como inferencia.

Como se puede ver, la privacidad de la información es crucial y obligada en muchos casos por ley, en todos los ámbitos (profesional, educativo...) y algunas de estas técnicas son estudiadas en grados como el de Ingeniería Informática.

Debido a que no existen muchas opciones gratuitas para trabajar con la ofuscación y enmascaramiento de datos en bases de datos relacionales como por ejemplo MySQL, es complicado para los alumnos conocer de primera mano su funcionamiento. Es por esto, que se trataba de una buena oportunidad el construir (con los conocimientos adquiridos en mis estudios) una serie de procedimientos y funciones de anonimización de datos que simulasen a aquellos proporcionados por las versiones de pago y así poder ser utilizados en la universidad por futuros compañeros.

1.3. Objetivos

El objetivo principal de este proyecto es la creación de un toolkit (caja de herramientas) compuesta por procedimientos y funciones que permitan anonimizar datos sensibles a través de distintas técnicas como el intercambio de caracteres, sustitución por datos sintéticos, creación de valores hash...etc, y que los transforme en datos no identificables para evitar ser leídos/utilizados sin los permisos adecuados. Además, las herramientas deberán ser aptas para su uso en bases de datos relaciones (MySQL) cumpliendo con el principio de privacidad y garantizando la seguridad de los datos.

Otro de los objetivos del trabajo es que dichos procedimientos/funciones sean entendibles y fáciles de usar ya que están principalmente orientados para ser utilizados por alumnos, por lo que además del propio código, se incluirá una breve descripción del objetivo individual y su funcionamiento.

1.4. Organización de la memoria

Este documento se divide en nueve capítulos de la siguiente manera:

- **Capítulo 1. Introducción:** Introducción al tema y objetivos del proyecto.
- **Capítulo 2. Planificación del proyecto:** Se describe detalladamente en varios apartados la planificación de fechas y tareas que se deben llevar a cabo, riesgos que puedan surgir o el presupuesto del proyecto.

- **Capítulo 3. Fundamentos teóricos:** Breve introducción a los conceptos teóricos sobre el tema del proyecto.

- **Capítulo 4. Herramientas que aplican la anonimización:** Muestra de algunas de las opciones alternativas que existen en el mercado.

- **Capítulo 5. Análisis:** Se realiza un análisis del proyecto y se detallan los requisitos que se deben cumplir y casos de uso que existirán.

- **Capítulo 6. Diseño:** Se explica cuál es el diseño seguido para la realización del proyecto.

- **Capítulo 7. Implementación:** Especificación de las herramientas usadas y métodos llevados a cabo para la implementación del proyecto.

- **Capítulo 8. Plan de pruebas:** Se realiza una comprobación de que todas las herramientas se ejecutan correctamente.

- **Capítulo 9. Conclusiones:** Una pequeña reflexión a cerca del trabajo realizado y posibles mejoras de futuro.

- **Bibliografía**

- **Anexo. Manual de Usuario:** Manual de uso de las herramientas del toolkit.

- **Anexo. Resumen de Herramientas:** Esquema numerado del conjunto de herramientas del toolkit.

- **Anexo. Diccionario de Términos:** Explicación de algunos de los términos utilizados en el documento.

- **Anexo. Repositorio ficheros:** URL de acceso al repositorio que contiene los ficheros (código fuente y ejemplos) que se adjuntan con la memoria.

Capítulo 2

Planificación del proyecto

Previamente antes de comenzar a trabajar con el proyecto y poder llevarlo a cabo con el mayor éxito posible, se realizará una planificación con el fin de tener una idea general y marcar unas pautas. Primero se detallarán las distintas tareas y tiempo aproximado que llevará completarlas cada una. A continuación, se analizarán los distintos riegos que pueden surgir en el proceso del trabajo y que puedan modificar la planificación inicial y retrasar el proyecto. Finalmente, se estimará el presupuesto necesario que deberá invertirse para llevar a cabo el proyecto.

2.1. Planificación inicial

El proyecto se dividirá en distintas etapas y cada una de ellas se llevará a cabo de forma secuencial, por lo tanto, se seguirá una metodología en cascada. Cada una de las etapas o fases deberá ser completada para poder continuar con la siguiente y se desarrollará de forma paralela el trabajo indicado (en la etapa correspondiente) con la cumplimentación de la sección correspondiente de la memoria. A continuación, se especifican las distintas fases en las que se divide el proyecto:

1. **Inicio y estudio previo:** En esta primera fase, se estudiará a alto nivel cuál es el objetivo principal del proyecto y un pequeño estudio de mercado para conocer de la existencia o no de posibles alternativas. También se buscará información, se estudiará los conceptos teóricos necesarios y se realizará la planificación completa necesaria para llevar a cabo el proyecto.

2. **Análisis:** A continuación, se estudiarán los distintos tipos de requisitos que deba cumplir el proyecto y se detallarán los casos de uso que existan.

3. **Diseño:** En esta fase, se realizarán una serie de diagramas que reflejen el diseño del proyecto. En la memoria quedarán reflejados el modelo de dominio, un modelo conceptual y lógico para la base de datos.
   Al no tratarse de una aplicación como tal, se mostrarán algunos diagramas como el de paquetes o despliegue enfocados al sistema del proyecto, y se obviarán algunos como el tipo de patrón de diseño que pueda seguir una aplicación.

4. **Implementación:** Esta será la parte más extensa del proyecto pues cuenta con varias subpartes: primero se introducirán las tecnologías utilizadas, posteriormente se especificarán los pasos para su desarrollo e implementación, se detallarán las funciones que proporcionarán las herramientas del kit y finalmente se mostrará cómo deben ejecutarse algunas funciones características de las herramientas para su uso correcto.

5. **Pruebas:** Finalmente, se llevará a cabo un plan de pruebas donde se pueda comprobar el correcto funcionamiento de todas las herramientas propuestas y que los resultados obtenidos son los esperados. Estos resultados se validarán con ejemplos propuestos por la AEPD (Agencia Española de Protección de Datos) para asegurarse de que se cumple el objetivo con éxito. Para acabar, se realizará una comprobación general de todas las fases del proyecto.

A continuación, se detallarán las fechas de comienzo y fin del proyecto, así como las tareas propuestas para llevarse a cabo durante estas y que estén ajustadas a unos tiempos concretos. Si bien esta planificación inicial se tratará de cumplir con la máxima exactitud, no deja de ser una cierta estimación debido a posibles riesgos que puedan surgir. Como medida de prevención y dar un cierto margen, se estimará una semana a mayores a partir del plan ajustado para la realización de las tareas.

- **Fecha de inicio:** 1 de marzo
- **Fecha de finalización:** 7 de julio
- **Duración del proyecto:** 19 semanas

Para mostrar de una forma más visual y estructurada la planificación de las distintas tareas que se tienen que ejecutar, se ha construido un diagrama de Gantt. En él, se representan las fechas de las tareas con su correspondiente dependencia de finaliza a inicia con sus predecesoras donde una tarea no podrá comenzar si no se ha finalizado la anterior.

	Nombre de tarea	Duración	Comienzo	Fin
1	Búsqueda y prueba de herramientas similares	6 días	mié 01/03/23	mié 08/03/23
2	Estudio de conceptos teóricos	8 días	jue 09/03/23	dom 19/03/23
3	Planificación del proyecto	7 días	lun 20/03/23	mar 28/03/23
4	Fundamentos teóricos	7 días	mié 29/03/23	jue 06/04/23
5	Recolección de requisitos	5 días	vie 07/04/23	jue 13/04/23
6	Planificación y realización de casos de uso	6 días	vie 14/04/23	vie 21/04/23
7	Diseño e Implementación	39 días	lun 24/04/23	jue 15/06/23
8	Pruebas	5 días	vie 16/06/23	jue 22/06/23
9	Introducción, conclusiones, resumen y agradecimientos	5 días	vie 23/06/23	jue 29/06/23

Figura 2.1: Planificación del proyecto

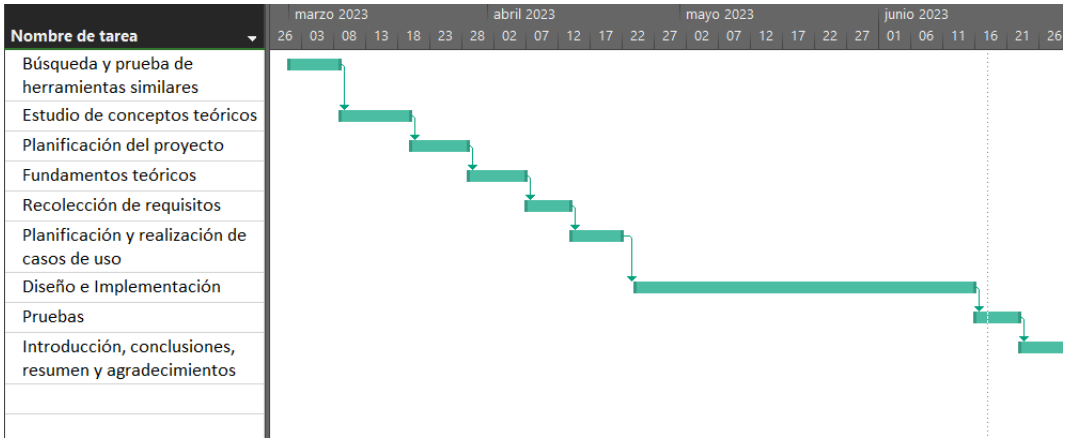


Figura 2.2: Diagrama de Gantt

Como se puede observar en el diagrama de Gantt (Figura 2.2), la tarea más larga y que más tiempo va a llevar es la de "diseño e implementación" que durará alrededor de 8 semanas.

2.2. Plan de riesgos

Tras realizar una planificación general del proyecto, se analizarán los posibles riegos que puedan surgir y que produzcan un efecto negativo en dicha planificación o afecte directamente en el trabajo. Primero se identificarán aquellos riegos que puedan surgir; a continuación,

se analizará su exposición mediante una matriz de riesgos en base a dos ejes: frecuencia e impacto; y finalmente se estudiarán posibles planes de contingencia con el fin de poder prevenir o mitigar aquellos riesgos que puedan tener un alto impacto en el proyecto.

Como se ha descrito, para realizar el análisis de riesgos y determinar la exposición que presentan, se creará una matriz de riesgos que ayudará de forma visual a comprender dichos parámetros. Como se muestra en la Figura 2.3 , se establece una escala del 1 al 5 para analizar la frecuencia de menor a mayor probabilidad de que ocurra, y manteniendo la misma escala se analizará el impacto dependiendo de la gravedad que generase en caso de llegar a producirse.

Fecuencia						
	5	Alto	Alto	Muy alto	Muy alto	Muy alto
	4	Medio	Medio	Alto	Muy alto	Muy alto
	3	Bajo	Medio	Alto	Alto	Muy alto
	2	Muy bajo	Bajo	Medio	Alto	Alto
	1	Muy bajo	Bajo	Medio	Alto	Alto
		1	2	3	4	5
		Impacto				

Figura 2.3: Matriz de Riesgos

En la Figura 2.4 y Figura 2.5 se muestran respectivamente los riegos identificados y su análisis, y algunos planes asociados a algunos de los riesgos que peor impacto generarían sobre el proyecto. Tras realizar el análisis de los riesgos identificados, algunos de ellos muestran una exposición baja lo cual nos permite obviarlos y simplemente aceptarlos. Por otro lado, aquellos que muestran un mayor impacto, nos plantea dos opciones: bien se puede realizar un plan de prevención donde se apliquen una serie de pasos para prevenir (o tratar de reducir) que el riesgo ocurra; o bien se puede plantear un plan de contingencia que se aplicaría una vez el riesgo haya aparecido y que tiene como fin tratar de mitigar aquellos efectos negativos que hayan podido producirse para intentar provocar los menores daños posibles.

Riesgo	Frecuencia	Impacto	Exposición
Planificación incorrecta en la duración de las tareas	5	5	Muy alto
Incorrecto funcionamiento del ordenador de trabajo	1	5	Alto
Innacesibilidad a la base de datos	2	4	Alto
Enfermedad del trabajador	2	4	Alto
Requisitos incorrectos y modificación necesaria	4	3	Alto
Fallos en la implementación	5	3	Muy alto
Aprendizaje de tecnologías o métodos de programación desconocidos	4	4	Muy alto
Aparición de otras ocupaciones prioritarias del trabajador	3	3	Alto

Figura 2.4: Análisis de riesgos

Riesgo	Plan
Planificación incorrecta en la duración de las tareas	Evitar el riesgo planificando los tiempos y fechas con un margen respecto al tiempo estimado
Incorrecto funcionamiento del ordenador de trabajo	Reducir el riesgo realizando copias de seguridad y tratar de disponer de otro ordenador
Innacesibilidad a la base de datos	Reducir el riesgo realizando copias de seguridad y tratar de disponer de otra base de datos
Enfermedad del trabajador	Reducir el riesgo aumentando el tiempo de planificación de las tareas
Requisitos incorrectos y modificación necesaria	Evitar el riesgo realizando un correcto análisis y recogida de requisitos desde el principio del proyecto. También realizar revisiones periódicas
Fallos en la implementación	Evitar el riesgo realizando una planificación previa de la implementación
Aprendizaje de tecnologías o métodos de programación desconocidos	Evitar el riesgo realizando un estudio más profundo sobre las tecnologías
Aparición de otras ocupaciones prioritarias del trabajador	Reducir el riesgo aumentando el tiempo de planificación de las tareas

Figura 2.5: Plan de riesgos

2.3. Presupuesto

En este apartado, se realizará un cálculo aproximado del presupuesto que es necesario para llevar a cabo el proyecto teniendo en cuenta aquellos costes que supongan el hardware y software, y los gastos en el personal involucrado.

El presupuesto necesario para los productos será de 0€, ya que el hardware que va a ser necesario se trata de un ordenador normal (que será el equipo personal del desarrollador) y una máquina virtual (proporcionada por la Universidad) para el alojamiento de la base de datos. En cuanto a los gastos de software se utilizarán servicios gratuitos o bien con licencia de Universidad. Si no se obtuvieran dichas licencias (tanto en hardware como en software), el coste de crear y mantener una base de datos MySQL en Azure sería de 225€/mes [3] y la compra de software como Visual Paradigm ascendería 320€ [4].

El gasto de personal en este caso también será de 0€ ya que se trata de un proyecto enfocado a estudios y por tanto el desarrollador no obtendrá remuneración por la realización del trabajo. Sin embargo, si se puede llegar a estimar cuanto presupuesto podría suponer en el apartado personal en base a datos registrados a nivel empresarial. En España, a fecha actual, un desarrollador junior puede llegar a tener un salario medio de 22.350€/año, lo que supone que, en una semana de 40 horas laborales, el sueldo se aproximaría a los 12€/hora. Dado que se estima alrededor de 300 horas de trabajo para completar este proyecto, podríamos calcular un gasto de personal de 3.600€ [5].

Capítulo 3

Fundamentos teóricos

Para entender mejor porqué es necesaria la anonimización y en qué consiste, se expondrá en este capítulo una breve introducción. También, se comentará cuál es el proceso que hay que seguir para conseguir anonimizar datos de forma correcta y cuáles son las técnicas actuales más usadas para llevar a cabo dicho proceso.

3.1. ¿Por qué es necesaria la anonimización de datos?

Recientemente en estos últimos años, el papel que juegan los datos en la sociedad se ha convertido en algo crucial, siendo estos, activos claves para casi cualquier proceso de nuestra vida cotidiana. La forma en que se recogen ha aumentado de una forma casi imperceptible para la ciudadanía y la capacidad para procesarlos y compartirlos se ha vuelto muy potente. Algunas de las principales tecnologías actuales donde juegan un papel crucial son el IoT, Blockchain, Inteligencia Artificial, Big Data o Linked Data; y su característica común, es que la gran mayoría de los datos usados por estas son almacenados en bases de datos. [6]

Tanto en su uso como su almacenamiento, el impacto de una brecha de datos supondría un gran desastre. Por un lado, la organización perdería la confianza de sus clientes, recibiría sanciones financieras y otras posibles consecuencias; por otro y posiblemente el más importante, es la exposición de privacidad a la que se enfrentaría las personas afectadas, ya bien sea en datos personales de identidad, salud, bancarios...etc.

A pesar de que hoy en día las principales amenazas siguen siendo externas, no se puede obviar que también existen las amenazas a los datos sensibles desde la perspectiva interna de

una organización. Hay un gran número de registros de robos, manipulación o uso indebido por parte de los empleados de información de clientes, identificación personal o datos de tarjetas de crédito. En gran medida, esto se debe a que personas con el rol de administrador de sistemas o bases de datos tienen acceso a estos o porque a menudo, para realizar pruebas o mejoras en los sistemas, las empresas realizan copias de los datos desde entornos de producción a entornos de desarrollo donde las medidas de seguridad son menores y su acceso está más extendido a un mayor número de empleados. [7]

Por todo esto y debido a las crecientes amenazas en los entornos informáticos, se han ido poco a poco implantando legislaciones de privacidad de datos como son el RGPD (Reglamento general de protección de datos) en la UE o la CCPA en los EE.UU., con el fin de asegurar la privacidad de los usuarios y la protección de sus datos personales, entendidos como derechos fundamentales. Las empresas y organizaciones están cada vez más invirtiendo en soluciones para cumplir con la seguridad de la información y para ello, una de las técnicas que usan es el enmascaramiento de datos. [6]

3.2. Introducción a la anonimización

Definición de "Anonimizar" según la RAE:

"Expresar un dato relativo a entidades o personas, eliminando la referencia a su entidad."[8]

Esta definición nos da una ligera idea de lo que significa anonimizar, pero para concretar más, la secretaría de estado de digitalización e inteligencia artificial nos define la anonimización de datos como: "Metodología y conjunto de buenas prácticas y técnicas que reducen el riesgo de identificación de personas, la irreversibilidad del proceso de anonimización y la auditoría de la explotación de los datos anonimizados, monitorizando quién, cuándo y para qué se usan" [9].

Anonimizar datos, también es conocido como enmascaramiento u ofuscación de datos, pero siempre y como hemos visto, tratando de cubrir tanto el objetivo de anonimización, como el de mitigación del riesgo de reidentificación, siendo este último un aspecto clave.

Para entenderlo de una forma más amigable, la anonimización de datos consiste en el reemplazo de datos sensibles mediante el uso de datos ficticios funcionales como caracteres, usando datos sintéticos o mediante otras técnicas; y su objetivo principal es proteger la información confidencial y privada en situaciones en las que se pueda ver comprometida. Es importante destacar, que el formato (la estructura) de estos datos sigue siendo el mismo y solo se cambia el valor para generar una incorreción frente a los datos originales.

La anonimización de datos debe regirse por el concepto de privacidad desde el diseño y por defecto (RGPD, art.25)[10], teniendo en cuenta 7 principios [11]:

- **Proactivo:** la identificación de los datos sensibles o de identificación indirecta, debe llevarse a cabo desde las primeras etapas de conceptualización. Además, se deberá establecer una escala de sensibilidad de los datos e informar a todos los implicados en el proceso de anonimización.

- **Privacidad por defecto:** se deberá realizar un estudio del grado de detalle o granularidad con la que serán anonimizados los datos para evitar problemas de confidencialidad. El uso de variables no esenciales puede llevar a la aparición de riesgos, aspecto (así como los beneficios) que también se deberán tener en cuenta.

- **Objetivo:** debido a que una anonimización completa es muy difícil de conseguir y existen riesgos de reidentificación, es crítico identificar el nivel de riesgo de reidentificación y establecer las medidas adecuadas, como por ejemplo planes de contingencia.

- **Funcional:** tras la anonimización de los datos, hay que tener en cuenta que estos serán explotados por otros usuarios y por ello, es importante mantener un grado de utilidad adecuado. Para conseguirlo, es imprescindible realizar un análisis de la finalidad del conjunto de datos e informar a los usuarios de las técnicas y procesos de anonimización utilizados para conseguir su distorsión.

- **Integral:** el proceso de anonimización no solo se trata de la distorsión de los datos, también se refleja durante el estudio de estos, mediante contratos de confidencialidad y su uso limitado. Durante todo el ciclo de vida del proceso de anonimización, se deben realizar auditorías que validen la confidencialidad de los datos.

- **Informativo:** es uno de los principios más críticos y claves. Todos los implicados en el ciclo de vida del proceso de anonimización deben ser informados y debidamente capacitados para asumir sus responsabilidades y riesgos asociados.

- **Atómico:** se recomienda que, en la medida de lo posible, las responsabilidades y funciones del proceso de anonimización se asignen a personas del equipo de forma independiente, no interfiriendo las unas con las otras.

En el proceso de anonimización, con el fin de facilitar la comprensión y mejorar la calidad de este, es importante definir un esquema basado en los tres niveles de identificación de personas: microdatos, identificadores indirectos y datos sensibles (principio de proactividad), donde se asigne un valor cuantitativo a cada uno de los niveles. Esta escala deber ser conocida por todo el personal implicado (principio de información) y es crítico para la Evaluación del Impacto en la Protección de los Datos Personales (EIPD).

3.3. Proceso de anonimización

Para llevar a cabo la anonimización de datos, se debe seguir una serie de pasos con el fin de garantizar que el proceso es seguro y cumple con los objetivos. La Guía de orientaciones y garantías en los procedimientos de anonimización de datos personales de la Agencia Española de Protección de Datos (AEPD) especifica una serie de fases [12] a seguir en el proceso:

- **Definición del equipo de trabajo:** Antes de comenzar con el proceso de anonimización, es recomendable definir cuáles serán las funciones y el detalle del alcance de cada uno de los actores, roles o perfiles que van a tratar con los datos a anonimizar. Es importante que cada uno de esos actores obre en el ámbito de competencia que les corresponde independientemente del resto de actores. El número de estos será por norma general proporcional al volumen de datos que se vayan a tratar.

- **Evaluación de riesgos de reidentificación:** El impacto sobre los recursos de una organización (económico, tecnológico, humano...) que genera el proceso de anonimización, debe ser estudiado con cuidado ya que se debe ajustar a los objetivos y requerimientos necesarios, y estos dependerán de las medidas de seguridad que se precisen implantar para garantizar la privacidad de los datos.

  Es importante recordar que ninguna técnica de anonimización puede llegar a garantizar completamente que esos datos puedan ser reidentificados, pues siempre existe una pequeña probabilidad de que así sea, pero se debe intentar atenuar ese problema mediante la correspondiente gestión de riesgos.

  Algunos de los riesgos que se deben tener en cuenta pueden ser originados por una mala implementación de procedimientos de anonimización, ineficaz formación o información del personal implicado en la anonimización, o inadecuada gestión de claves cuando se emplean métodos basados en algoritmos de cifrado. Pero el mayor riesgo de reidentificación es el hecho de que está implícito y se incrementa a medida que transcurre el tiempo desde la anonimización de los datos.

- **Definición de objetivos y finalidad de la información anonimizada:** Como en la gran mayoría de procesos, una de las partes más importantes es establecer el objetivo de los datos que se van a anonimizar. El responsable del tratamiento determinará los objetivos y finalidad que deberá cumplir la información anonimizada en función de los intereses del destinatario, y el proceso de diseño estará condicionado por esta finalidad dando lugar a datos de uso restringido (los cuales deberán ser reforzados en cuanto a seguridad se refiere) o abiertos.

- **Viabilidad del proceso:** En el proceso de anonimización de datos, es recomendable realizar un informe de viabilidad que refleje los motivos y condiciones específicas de

forma detallada. También se pueden incluir fundamentos o vinculaciones éticas de dicho proceso, y todo ello ser consultado por expertos de seguridad donde se dictamine la conformidad al umbral de riesgos aceptable que resulte de la EIPD o por el contrario no.

- **Preanonimización:** definición de variables de identificación: La preanonimicación es una de las primeras fases del proceso donde se determinan:

    - "Posibles variables de identificación: datos personales, identificadores directos o indirectos, datos especialmente protegidos y otros datos de carácter confidencial."

    - "Clasificación y sensibilidad de las variables por categorías: de identificación directa, identificación geográfica, numéricas, temporales, metadatos, etc."

    - "Variables de identificación que no puedan ser anonimizadas y que sea preciso eliminar del proceso de anonimización."

    - "Variables anonimizadas que sean imprescindibles para la finalidad a la que se van a destinar los datos anonimizados."

  Tras realizar la categorización de las variables, se deben establecer los criterios de protección que se requieran para garantizar la privacidad de las personas, intentando reducir al máximo la cantidad de información personal que vaya a ser utilizada durante el proceso de anonimización.

- **Eliminación/reducción de variables:** El objetivo de esta fase es reducir al mínimo necesario la cantidad de variables que permitan la identificación de las personas. Este hecho, además de optimizar el coste computacional, implica de forma directa una reducción del riesgo de reidentificación, pues a menor cantidad de datos personales, más complicado es la identificación y se evitan pérdidas de información, brechas de datos, robo de claves...

  Para llevar a cabo este paso, se pueden poner en práctica algunos aspectos como por ejemplo determinar la finalidad de los datos anonimizados, eliminar datos identificativos directos o indirectos, control segregado de usuarios con acceso a los datos personales... etc.

- **Selección de técnicas de anonimización:** Para llevar a cabo la anonimización de los datos, se debe estudiar qué técnica es la más apropiada y a la vez viable, con el fin de mantener la correcta privacidad de los datos y ajustarse a los requisitos establecidos. Entre ellas, algunas de las técnicas más comunes son: algoritmo de hash, algoritmo de cifrado, sello de tiempo, capas de anonimización, perturbación de datos o reducción de datos.

- **Segregación de la información:** Para garantizar la confidencialidad de la información anonimizada, es recomendable elaborar un mapa de sistemas de información que garantice entornos separados para cada tratamiento de datos personales. Esto implica también, la segregación del personal que accede a la información y datos personales.

- **Proyecto Piloto:** Previamente al desarrollo de la anonimización en producción, es recomendable mediante una pequeña muestra de datos de prueba, realizar un proyecto piloto con el fin de obtener unas conclusiones a alto nivel y obtener de forma objetiva conclusiones acerca de la viabilidad de las propuestas de los miembros de anonimización.

  Con todo ello, se pretende obtener resultados que permitan comprobar su fortaleza, cuantificar costes del proceso de anonimización, asegurar que se cumple con los objetivos y realizar una evaluación general.

- **Anonimización:** Tras un proceso de análisis, uno de los últimos pasos se trata de la propia anonimización de los datos. Es decir, realizar la disociación definitiva e irreversible de los datos personales para cumplir con el objetico marcado. Este proceso debe realizarse tantas veces como sea necesario y entre los pasos a seguir en esta fase se encuentra: determinación de la técnica de anonimización más apropiada, de los recursos y equipo técnico necesario, validación de la técnica y resultados obtenidos por expertos, reducción de datos no necesarios y que prevengan de la reidentificación, revisión periódica del proceso, auditorías... etc.

- **Formación e información al personal implicado:** Previo y durante el proceso de anonimización de datos se deberá tener formado e informado a todo el personal implicado el proceso acerca del cumplimiento de la normativa de protección de datos personales, en especial en lo relativo a las medidas de seguridad a las que se refiere el artículo 32 del RGPD.

- **Garantías jurídicas:** Debido que no se puede garantizar una absoluta anonimización y cabe una posibilidad de reidentificación de las personas, se deben tener en cuenta unas garantías jurídicas necesarias para preservar los derechos de los interesados.

  Algunos de los aspectos a tener en cuentas son: acuerdos de confidencialidad, compromiso del destinatario, realización de auditorías... y todas estas garantías deben quedar registradas en el contrato suscrito entre el responsable del tratamiento el destinatario de la información anonimizada.

- **Auditoría del proceso de anonimización:** Para garantizar el cumplimiento de la política de anonimización y de que se cumplen los objetivos, es necesaria la realización de auditorías (siempre con la calidad adecuada) proporcionando una opinión objetiva sobre el conjunto del proceso de anonimización. Estas auditorías pueden ser internas o

externas y tendrán carácter periódico.

Entre la información que debe quedar registrada por la auditoría se encuentra: el alcance y objetivo de la auditoría, definición del equipo auditor y recursos utilizados, fases y planificación de la auditoría, pruebas y verificaciones... etc.

- **Documentación:** Finalmente, destacar que todo el proceso de anonimización debe quedar documentado y ser accesible al personal implicado en el tratamiento de datos anonimizados.

3.4. Técnicas de anonimización de datos

En la fase de elección de técnicas de anonimización, existe una variedad amplia, pero todas tienen como objetivo garantizar el avance de la sociedad y su información mediante la compartición de datos entre entidades, su almacenamiento y publicación de datos abiertos, sin renunciar al derecho de las personas respecto a la protección de sus datos personales.

Hoy en día, podemos distinguir como indica la AEPD [11], principalmente tres enfoques generales, cada uno formado por varias técnicas:

- **Aleatorización:** "tratamiento de datos, eliminando la correlación con el individuo, mediante la adición de ruido, la permutación o la Privacidad Diferencial."

- **Generalización:** "alteración de escalas u órdenes de magnitud a través de técnicas basadas en agregación como K-Anonimato, Diversidad-L o Proximidad-T."

- **Seudonimización:** "reemplazo de valores por versiones cifradas o tokens, habitualmente a través de algoritmos de HASH, que impiden la identificación directa del individuo, a menos que se combine con otros datos adicionales, que deben estar custodiados de forma adecuada."

El hecho de que la seudonimización se enmarque como uno de los enfoques de la anonimización puede generar confusión ya que son términos distintos y es importante destacar esta diferencia.

El objetivo principal de la anonimización es la imposibilidad de reidentificación de los datos (o al menos es lo que se intenta), mediante la desvinculación completa de los datos

personales de los datos identificativos. La AEPD define la anonimización [12] como: *"la ruptura de la cadena de identificación de las personas."*

Por otro lado, la seudonimización sí que es susceptible de permitir la reidentificación ya que se tratan los datos personales sin desvincularlos de los datos identificativos del interesado. El RGPD define la seudonimización [1] como: *"aquella información que, sin incluir los datos denominativos de un sujeto, permiten identificarlo mediante información adicional, siempre que esta figure por separado y esté sujeta a medidas técnicas y organizativas destinadas a garantizar que los datos personales no se atribuyan a una persona física identificada o identificable".*
Para conseguir que la seudonimización sea una técnica de anonimización, se hace necesaria la adición de otras técnicas que cumplimenten a esta primera.

Es por todo esto que hay que tener en cuenta que la seudonimización y el cifrado son técnicas íntimamente relacionadas y es complicado establecer una línea de diferencia entre ellas en su aplicación a la seudoninimización o la anonimización.

Para profundizar en los tres enfoques descritos, a continuación, se mencionarán algunas de las diferentes técnicas más usadas pertenecientes a dichos enfoques:

- **ALEATORIZACIÓN:** Dentro de este enfoque, "las técnicas de basan en modificar o alterar la veracidad de los datos a nivel individual, respetando la distribución global de éstos, consiguiendo reducir así la vinculabilidad y la inferencia."
    - *Adición de ruido:* Se trata de alterar los datos de cada individuo a nivel local, respetando la distribución global de la muestra para que mantenga su utilidad.
    - *Permutación:* En este caso no se modifican los valores o su distribución, si no que se intercambian unos con otros dentro del conjunto de datos para reducir la vinculabilidad a los individuos.
    - *Privacidad Diferencial:* Es un caso especial de la aleatorización. El proceso de anonimización se aplica cada vez que un tercero realiza una consulta y es gestionado mediante vistas anonimizadas por el responsable del tratamiento de los datos. En este caso los valores permanecen invariables y custodiados por el responsable.
- **GENERALIZACIÓN:** "Este enfoque tiene por objetivo generalizar algunos atributos críticos de forma que se evite la singularización, reemplazando valores por categorías superiores en una jerarquía."
    - *K-Anonimato:* Esta técnica es útil cuando las relaciones entre varios conjuntos de atributos permiten generar identificadores a través de la vinculabilidad o la

inferencia a partir de identificadores indirectos. Una forma de aplicar esta técnica es generalizar atributos reemplazando valores por otros valores superiores en una escala o jerarquía.

- *Diversidad-L y Proximidad-T:* Ambas técnicas tratan de ser una evolución del Anonimato-K, donde se busca mejorar las garantías frente a ataques de inferencia directa (aunque siguen siendo vulnerables a otro tipo de ataques).

  "En el caso de Diversidad-L, el proceso asegura que existen al menos L valores diferentes para cada atributo, dentro de un mismo conjunto de al menos K individuos.

  En el caso de Proximidad-T, garantiza a mayores que la distribución de los atributos dentro de cada grupo muestra la misma distribución del conjunto original, generando mayor dificultad frente a los ataques por inferencia."

- **SEUDONIMIZACIÓN:** "La seudonimización no se considera un método de anonimización aunque combinado con otras técnicas sí puede llegar a ser considerado como tal. Además, resultan ser medidas útiles para mejorar la seguridad, reduciendo la vinculabilidad del conjunto de datos obtenido."

  - *Cifrado y funciones HASH:* Se trata de la aplicación de algoritmos de cifrado de diferentes características, mostrando como mayor problema aquellas que actúan con clave ya que existe el riesgo de que un atacante consiga la clave y pueda revertir el proceso. Sin embargo, el uso de algoritmos actualizados de cifrado hace que sea muy complejo que un atacante pueda descifrar los valores, pero, no hay que descuidarse de los avances tecnológicos que hacen que esas garantías disminuyan con el tiempo.

  - *Descomposición en tokens:* Esta técnica empleada de forma habitual en entornos financieros se basa en los principios del cifrado donde se suele aplicar un proceso unidireccional o irreversible.

  - *Cifrado homomórfico:* Es un concepto relacionado con el cifrado ordenable. El objetivo de esta técnica es permitir trabajar sobre datos cifrados resultando equivalente a otras operaciones aplicadas sobre el conjunto original.

    Con esta técnica es posible compartir datos cifrados, aplicar operaciones sin tener acceso a la información sensible y posteriormente descifrar el resultado final.

Además de los tres enfoques descritos junto a algunas de las técnicas asociadas a ellos, existen otras técnicas de anonimización de datos [13] que no pueden ser completamente englobadas en dichos enfoques.

Algunas de ellas son:

- *Supresión de registros:* Se trata de la eliminación de un registro completo en un conjunto de datos. Esta técnica afecta a múltiples atributos al mismo tiempo y se debe emplear para eliminar aquellos registros atípicos que son únicos o que no cumplen otros criterios, como la k-anonimidad.

- *Enmascaramiento de caracteres:* Esta técnica se refiere al intercambio de los caracteres de un valor de datos. Esto se puede hacer mediante el uso de un símbolo consistente (por ejemplo, "∗" o "X"). Se suele aplicar solo a algunos caracteres del atributo. Esta técnica es adecuada cuando el dato es una cadena de caracteres y ocultar una parte de ella es suficiente para proporcionar el grado de anonimato requerido.

- *Agregación de datos:* Se refiere a la conversión de un conjunto de datos de una lista de registros a valores resumidos y es aplicable cuando no se requieren registros individuales y los datos agregados son suficientes para cumplir con el objetivo.

3.5. Riesgos y retos asociados a la anonimización

En el proceso de anonimización de datos cobra gran importancia el análisis y evaluación de riesgos para evitar situaciones graves de reidentificación de individuos entre los conjuntos de datos anonimizados.

El hecho de que cada vez existe un mayor número de datos y la tecnología se desarrolla muy rápido, no podemos obviar de que el riesgo de reidentificación aumenta con el paso del tiempo, sin embargo, se debe tratar de minimizar al máximo esa posible situación.

Para el análisis de riesgos se deben plantear tres vectores principales asociados a la reidentificación [11]:

- **Singularización:** Este aspecto mide la posibilidad de extraer de un conjunto de datos algunos registros (o todos) que identifican a una persona, es decir, contempla el riesgo de extraer atributos que permitan identificar a uno o varios individuos.

- **Vinculabilidad:** Se mide la capacidad de vincular como mínimo dos registros de un único interesado o de un grupo de interesados, ya sea en el mismo conjunto de datos o en dos conjuntos de datos distintos. Si el atacante puede determinar que dos registros están asignados al mismo grupo de personas, pero no puede singularizar a las personas en este grupo, entonces la técnica es resistente a la singularización, pero no a la vinculabilidad.

- **Inferencia:** Mide la posibilidad de deducir con una probabilidad significativa el valor de un atributo a partir de los valores de un conjunto de otros atributos.

Además de los riesgos descritos asociados a la reidentificación, a la hora de anonimizar datos se plantean algunos retos como, por ejemplo: los usuarios, ¿cuántos usuarios tendrán acceso a esos datos?; los permisos, ¿a qué cantidad de datos pueden acceder los usuarios?; políticas, ¿existen políticas claras de gobernanza de datos?; o metadatos, ¿están los conjuntos de datos de la plataforma claramente etiquetados y descritos, de forma que los responsables del proceso de anonimización puedan comprender rápidamente su sensibilidad y las políticas de protección aplicables? [14]

Capítulo 4

Herramientas que aplican la anonimización

Este proyecto se basa en la creación de herramientas de anonimización, que si bien ya están disponibles en el mercado, la gran mayoría son de pago y esto hace que no sean tan accesibles a los alumnos en las universidades. En este capítulo se presentarán algunas de estas opciones y cúales son sus características.

4.1. MySQL Enterprise Masking and Deidentification

MySQL proporciona la posibilidad de usar bases de datos de forma gratuita, sin embargo, algunas funciones como es el caso del enmascaramiento de datos las restringe a versiones de pago. De hecho, este trabajo tiene como base tratar de emular algunas funciones de esta versión de MySQL, además de añadir alguna otras, para permitir a alumnos su uso de forma gratuita.

"MySQL Enterprise Masking and Deidentification" [15] de una forma sencilla e intuitiva, trata de dar una solución a las organizaciones para poder proteger sus datos sensibles de usos no autorizados mediante técnicas de ocultación de caracteres o reemplazo de valores reales por sustitutos. Como ventaja frente a otras tecnologías, es que esta versión se instala como parte de las bases de datos (es built-in), por lo que no hace falta usar herramientas externas y el enmascaramiento se lleva a cabo directamente en las bases de datos.

Algunas de las principales características que proporciona esta herramienta de MySQL Enterprise son:

- Permite cumplir con los requisitos regulatorios y leyes de privacidad de datos de la industria. Algunos son el General Data Protection Directive (GDPR), Payment Card Industry Data Security Standard (PCI DSS), Health Insurance Portability and Accountability Act (HIPAA), Data Protection Act...etc.

- Mejora la seguridad del trabajo en entornos de desarrollo, testeo o analytics, permitiendo trabajar con datos sensibles estando enmascarados o mediante datos sintéticos y así evitar problemas en caso de producirse brechas de seguridad. También se asegura un trabajo más fiable en el caso de compartición de estos datos sensibles con terceras partes.

- Proporciona funciones robustas de enmascaramiento de datos como, por ejemplo:

    - *Enmascaramiento selectivo:* oculta una porción particular de números o texto como números de teléfono o de tarjetas de crédito.

    - *Enmascaramiento estricto o relajado:* implementa una ocultación de la información de forma estricta o relajada.

    - *Sustitución por datos aleatorios:* Reemplaza valores reales por valores aleatorios mientras mantiene la consistencia en el formato.

    - *Desenfoque:* Añade un valor aleatorio dentro de un rango a valores existentes como por ejemplo un rango de salarios.

    - *Sustitución por diccionario:* Reemplaza valores de forma aleatoria por aquellos almacenados en un diccionario.

    - *Lista de bloques y sustitución:* Reemplaza específicamente los datos de la lista de bloques, pero dejando en su lugar los no incluidos en la lista de bloques.

4.2. Amnesia

"Amnesia" [16] se trata de una empresa privada que proporciona un software de anonimización de datos (de pago) que sirve como bien se indica para anonimizar datos personales y confidenciales.

Entre sus herramientas encontramos:

- *seudonimización de datos:* método de desidentificación que permite la eliminación o el reemplazo de los identificadores directos (como nombres, números de teléfono, etc)

- *enmascaramiento de datos:* permite ocultar parte de la información en el conjunto de datos utilizando caracteres alternativos.

- *K-nonimato y Km-anonimato:* agrupa y genera datos cuasi-identificadores en grupos con jerarquías superiores gracias a la generalización, reduciendo la especificidad de un atributo sustituyendo un valor específico por uno más general.

- *Estadísticas demográficas:* utiliza la generalización y la supresión, preservando la facilidad de uso de los datos.

Su uso es sencillo permitiendo que cualquier usuario pueda trabajar con esta herramienta. Tan solo se debe importar los datos originales en un fichero con delimitadores para separar los diferentes campos, a continuación, se escogerá la técnica que deseemos para el enmascaramiento de los datos o crearemos las jerarquías de generalización adecuadas y finalmente el programa procede a enmascarar los datos. El resultado, se puede exportar a otras plataformas, servidores proporcionados por la herramienta o a un fichero personal.

Como se puede observar, esta herramienta es potente, pero tiene el inconveniente de que no se trata de una herramienta built-in en ninguna de las bases de datos más conocidas.

4.3. Otras herramientas

Otras herramientas para la anonimización de datos son [17]:

- *Arcad DOT-Anonymizer:* Se trata de una herramienta que mantiene la confidencialidad de los datos de prueba ocultando la información personal. Funciona anonimizando los datos personales conservando su formato y tipo.

- *ARGUS:* Sus siglas significan “Anti Re-identification General Utility System”. Este software utiliza una amplia gama de diferentes métodos de anonimización estadística,

como la recodificación global (agrupación de categorías), supresión local, la aleatorización, adición de ruido, codificación superior e inferior. También permite la generación de datos sintéticos.

- *Eclipse:* Conjunto de herramientas de Privacy Analytics que permite la anonimización de los datos de salud.

- *ARX:* Se trata de un software de código abierto para anonimizar datos personales sensibles.

- *UTD Anonymisation Toolbox:* La empresa UT Dallas Data Security and Privacy Lab creó y compile varias técnicas de anonimización en una caja de herramientas para uso público.

También, al igual que ocurre con MySQL Enterprise, existen soluciones built-in para diferentes tipos de bases de datos como son las de Oracle o Microsoft SQL Server.

Capítulo 5

Análisis

En la fase de análisis se estudia y especifica de forma detallada aquellos requisitos que debe cumplir el proyecto para cumplir con el objetivo. La definición de requisito que proporciona la IEEE es:

"Una condición o capacidad que necesita el usuario para resolver un problema o conseguir un objetivo determinado" [18].

En este apartado también se analizarán los distintos casos de uso (es decir, la funcionalidad) que proporcionará las herramientas del toolkit que conforma el proyecto.

5.1. Requisitos funcionales

Los requisitos funcionales son aquellos servicios que el sistema debe ser capaz de prestar, en la forma que reaccionará a determinadas entradas. Estas no tienen que ser necesariamente entradas de usuarios, si no que pueden ser interacciones con otros sistemas, respuestas automáticas o procesos predefinidos.

- Las herramientas deben ser capaces de anonimizar correctamente los datos de tal forma que no sea posible la reidentificación de dicha información o sujeto.

- El conjunto de herramientas (toolkit) debe permitir al usuario escoger entre las distintas

herramientas en función de la técnica de anonimización que desee.

- Algunas de las herramientas deben permitir al usuario indicar los datos que quiere anonimizar.

- Algunas de las herramientas deben permitir al usuario indicar el nombre de las tablas y columnas con los datos que desee anonimizar.

- Algunas de las herramientas deben ser capaces de crear tablas externas donde almacenar los pares (datoOriginal – datoEnmascarado).

- Algunas de las herramientas deben permitir al usuario indicar el nombre de la tabla externa donde se almacenarán los pares (datoOriginal – datoEnmascarado).

- Algunas de las herramientas deben permitir al usuario indicar el carácter especial que se usará para la sustitución de otros y así realizar la anonimización.

- Algunas de las herramientas deben permitir al usuario indicar las posiciones iniciales y finales del dato donde se desea anonimizar el dato.

- Algunas de las herramientas deben permitir al usuario indicar la clave primaria de una tabla si esta existe.

- Algunas de las herramientas deben permitir al usuario indicar la posición desde la que se desea reanudar la anonimización de una tabla en caso de fallo.

- Algunas de las herramientas deben ser capaces de trabajar con tablas importadas a la base de datos por el usuario.

- Algunas de las herramientas deben permitir al usuario indicar el tipo de datos (int o char) que almacenará la tabla externa donde almacenar los pares (datoOriginal – datoEnmascarado).

- Algunas de las herramientas deben permitir al usuario indicar el rango de números para la generación aleatoria de estos.

- Algunas de las herramientas deben permitir al usuario recuperar los datos originales una vez anonimizados.

- Algunas de las herramientas deben ser capaces de crear una tabla auxiliar para permitir ejecutar varios procesos y así cumplir con el objetivo de la herramienta.

- Algunas de las herramientas deberán restaurar las tablas a su estado original en caso de ser posible.

5.2. Requisitos no funcionales

Los requisitos no funcionales son aquellos que, no se refieren directamente a las funciones específicas que debe proporcionar el sistema, si no a las propiedades y restricciones de los servicios que presta. En otras palabras, no hablan de "lo que" hace, sino de "cómo" lo hace el sistema.

- El toolkit deberá poder ser importado en bases de datos relaciones (MySQL).

- Las herramientas deberán soportar la codificación UTF-8.

- Las herramientas deberán ser capaces de funcionar bien mediante la llamada de estas e introducción manual de los parámetros de entrada directamente desde la base de datos, o mediante un fichero de configuración con los comandos adecuados.

5.3. Requisitos de información

Los requisitos de información indican el tipo de información que debe guardar el sistema.

- Las herramientas deberán borrar todos los datos con los que haya trabajado al finalizar la anonimización.

- Las herramientas almacenarán durante el proceso de anonimización información de los datos sin anonimizar.

- Las herramientas almacenarán durante el proceso de anonimización información de las tablas que se deseen anonimizar.

- Las herramientas almacenarán durante el proceso de anonimización información de las tablas importadas por el usuario en caso de ser necesarias para el proceso.

5.4. Requisitos de seguridad y privacidad

- El conjunto de herramientas no almacenará en ningún lado más datos de los necesarios.

- Las herramientas borrarán todos los datos utilizados cuando estas finalicen el proceso de anonimización.

- El tratamiento de datos deberá estar presente en el Registro de Actividades de Tratamiento de Datos Personales previsto en el artículo 30 del RGPD. [Este requisito no se cumplirá para este proyecto ya que no se utilizan datos reales.] [19]

5.5. Casos de uso

En este apartado se mostrará primero un diagrama de casos de uso de las herramientas propuestas. En este diagrama se muestran las acciones que puede realizar el usuario (en base a los servicios que proporcionan las herramientas) y que han sido indicadas en el apartado de requisitos funcionales.

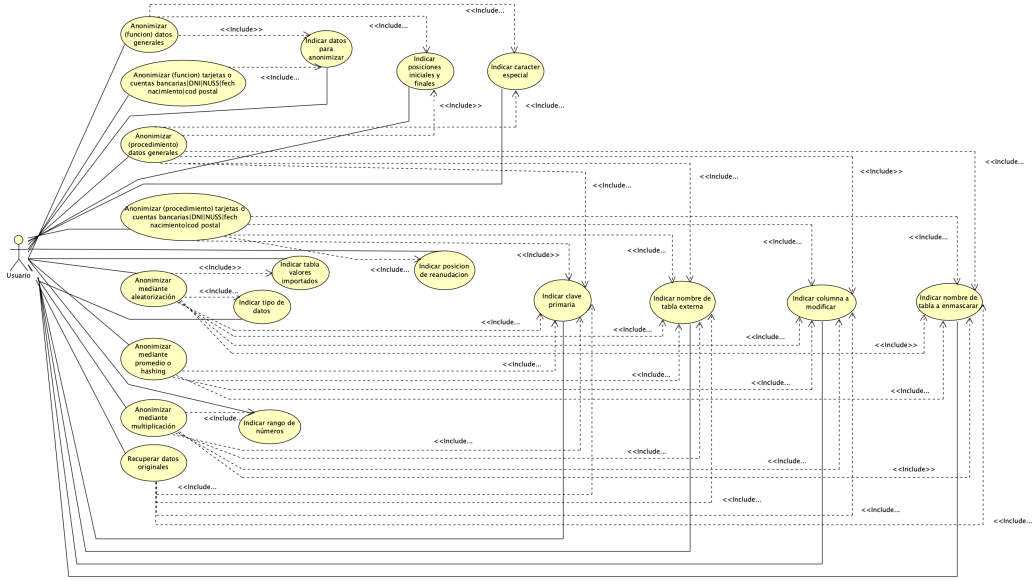


Figura 5.1: Diagrama de casos de uso

Para analizar más profundamente este diagrama, se presentan a continuación los distintos casos de uso detallados describiendo la secuencia de actividades que seguirá la base de datos con la intervención del usuario. [B]

Caso de Uso - 01	Anonimizar (funcion) datos generales
Descripción	El usuario selecciona una de las funciones que permite anonimizar datos de tipo general
Precondición	Las herramientas del toolkit (funciones) deben estar cargadas en la base de datos.
Secuencia normal	1. El usuario selecciona mediante el comando SQL adecuado la función que desee (apartado "anonimización de datos de propósito general") para llevar a cabo la anonimización. 2. El usuario introduce los parámetros de entrada requeridos. 3. El sistema muestra por pantalla el dato anonimizado.
Flujos alternativos	2.a El usuario introduce los parámetros de entrada de forma incorrecta, la bbdd informa del error y el caso de uso continua en el paso 1.
Postcondición	La función devuelve el dato anonimizado.

Figura 5.2: Caso de uso: Anonimizar (funcion) datos generales

Caso de Uso - 02	Anonimizar (funcion) tarjetas o cuentas bancarias|DNI|NUSS|fech nacimiento|cod postal
Descripción	El usuario selecciona una de las funciones que permite anonimizar tarjetas o cuentas bancarias, o DNI o NUSS o fecha de nacimiento o el código postal
Precondición	Las herramientas del toolkit (funciones) deben estar cargadas en la base de datos.
Secuencia normal	1. El usuario selecciona mediante el comando SQL adecuado la función que desee (apartado "anonimización de datos de propósito específico") para llevar a cabo la anonimización. 2. El usuario introduce los parámetros de entrada requeridos. 3. El sistema muestra por pantalla el dato anonimizado.
Flujos alternativos	2.a El usuario introduce los parámetros de entrada de forma incorrecta, la bbdd informa del error y el caso de uso continua en el paso 1.
Postcondición	La función devuelve el dato anonimizado.

Figura 5.3: Caso de uso: Anonimizar (funcion) datos específicos

Caso de Uso - 03	Anonimizar (procedimiento) datos generales
Descripción	El usuario selecciona uno de los procedimientos que permite anonimizar datos de tipo general
Precondición	Las herramientas del toolkit (procedimientos) deben estar cargadas en la base de datos. Las tablas de la base de datos que se quieran anonimizar deben estar pobladas con datos. Los datos que se deseen anonimizar deben ser de tipo VARCHAR.
Secuencia normal	1. El usuario selecciona mediante el comando SQL adecuado el procedimiento que desee (apartado "anonimización de datos de propósito general") para llevar a cabo la anonimización. 2. El usuario introduce los parámetros de entrada requeridos. 3. El sistema anonimiza los datos especificados en la tabla. 4. El sistema crea una tabla externa con el valor original.
Flujos alternativos	2.a El usuario introduce los parámetros de entrada de forma incorrecta, la bbdd informa del error y el caso de uso continua en el paso 1.
Postcondición	El procedimiento anonimiza los datos (de la columna de la tabla) seleccionados. El procedimiento crea una tabla externa con el valor original.

Figura 5.4: Caso de uso: Anonimizar (procedimiento) datos generales

Caso de Uso - 04	Anonimizar (procedimiento) tarjetas o cuentas bancarias|DNI|NUSS|fech nacimiento|cod postal
Descripción	El usuario selecciona uno de los procedimientos que permite anonimizar tarjetas o cuentas bancarias, o DNI o NUSS o fecha de nacimiento o el código postal
Precondición	Las herramientas del toolkit (procedimientos) deben estar cargadas en la base de datos. Las tablas de la base de datos que se quieran anonimizar deben estar pobladas con datos. Los datos que se deseen anonimizar deben seguir la estructura y tipo especificado en el manual de uso del toolkit.
Secuencia normal	1. El usuario selecciona mediante el comando SQL adecuado el procedimiento que desee (apartado "anonimización de datos de propósito específico") para llevar a cabo la anonimización. 2. El usuario introduce los parámetros de entrada requeridos. 3. El sistema anonimiza los datos especificados en la tabla. 4. El sistema crea una tabla externa con el valor original.
Flujos alternativos	2.a El usuario introduce los parámetros de entrada de forma incorrecta, la bbdd informa del error y el caso de uso continua en el paso 1.
	3.a El sistema detecta en la tabla que algún dato no se corresponde con la estructura o tipo adecuado. El caso de uso continua en el paso 1.
Postcondición	El procedimiento anonimiza los datos (de la columna de la tabla) seleccionados. El procedimiento crea una tabla externa con el valor original.

Figura 5.5: Caso de uso: Anonimizar (procedimiento) datos específicos

Caso de Uso - 05	Anonimizar mediante aleatorización
Descripción	El usuario selecciona el procedimiento que permite anonimizar datos mediante la técnica de aleatorización
Precondición	Las herramientas del toolkit (procedimientos) deben estar cargadas en la base de datos. Las tablas de la base de datos que se quieran anonimizar deben estar pobladas con datos. Los datos que se deseen anonimizar deben ser de tipo INT o VARCHAR.
Secuencia normal	1. El usuario importa en el servidor o la base de datos el diccionario de datos poblado, siguiendo las pautas del manual de uso del toolkit. 2. El usuario crea una tabla en la base de datos e importa los datos del diccionario 3. El usuario selecciona con el comando SQL adecuado el procedimiento que permite, mediante aleatorización, llevar a cabo la anonimización. 4. El usuario introduce los parámetros de entrada requeridos. 5. El sistema anonimiza los datos especificados en la tabla. 6. El sistema crea una tabla externa con pares (valorOriginal - valorAnonimizado).
Flujos alternativos	4.a El usuario introduce los parámetros de entrada de forma incorrecta, la bbdd informa del error y el caso de uso continua en el paso 3.
Postcondición	El procedimiento anonimiza los datos (de la columna de la tabla) seleccionados. El procedimiento crea una tabla externa con pares (valorOriginal - valorAnonimizado).

Figura 5.6: Caso de uso: Anonimizar mediante aleatorización

Caso de Uso - 06	Anonimizar mediante promedio o hashing
Descripción	El usuario selecciona uno de los procedimientos almacenados que permite la anonimización mediante el promedio o la generación de valores hash
Precondición	Las herramientas del toolkit (procedimientos) deben estar cargadas en la base de datos. Las tablas de la base de datos que se quieran anonimizar deben estar pobladas con datos. Los datos que se deseen anonimizar deben seguir tipo especificado en el manual de uso del toolkit.
Secuencia normal	1. El usuario selecciona mediante el comando SQL adecuado el procedimiento que desee (apartado "técnicas típicas en la anonimización") para llevar a cabo la anonimización. 2. El usuario introduce los parámetros de entrada requeridos. 3. El sistema anonimiza los datos especificados en la tabla. 4. El sistema crea una tabla externa con pares (valorOriginal - valorAnonimizado).
Flujos alternativos	2.a El usuario introduce los parámetros de entrada de forma incorrecta, la bbdd informa del error y el caso de uso continua en el paso 1.
Postcondición	El procedimiento anonimiza los datos (de la columna de la tabla) seleccionados. El procedimiento crea una tabla externa con pares (valorOriginal - valorAnonimizado).

Figura 5.7: Caso de uso: Anonimizar mediante promedio o hashing

Caso de Uso - 07	Anonimizar mediante multiplicación
Descripción	El usuario selecciona el procedimiento almacenado que permite la anonimización mediante multipicación de valores
Precondición	Las herramientas del toolkit (procedimientos) deben estar cargadas en la base de datos. Las tablas de la base de datos que se quieran anonimizar deben estar pobladas con datos. Los datos que se deseen anonimizar deben ser de tipo INT.
Secuencia normal	1. El usuario selecciona con el comando SQL adecuado el procedimiento que permite, mediante la multiplización, llevar a cabo la anonimización. 2. El usuario introduce los parámetros de entrada requeridos. 3. El sistema anonimiza los datos especificados en la tabla. 4. El sistema crea una tabla externa con valores numéricos aleatorios.
Flujos alternativos	2.a El usuario introduce los parámetros de entrada de forma incorrecta, la bbdd informa del error y el caso de uso continua en el paso 1.
Postcondición	El procedimiento anonimiza los datos (de la columna de la tabla) seleccionados. El procedimiento crea una tabla externa con valores numéricos aleatorios.

Figura 5.8: Caso de uso: Anonimizar mediante multiplicación

Caso de Uso - 08	Recuperar datos originales
Descripción	El usuario selecciona el procedimiento almacenado que permite recuperar datos originales tras haber sido anonimizados
Precondición	Las herramientas del toolkit (procedimientos) deben estar cargadas en la base de datos. Se debe haber ejecutado el procedimiento que usa la técnica de la multiplicación Los datos que se deseen recuperar deben ser de tipo INT. Debe existir una tabla externa creada por el procedimiento que usa la técnica de la multiplicación.
Secuencia normal	1. El usuario selecciona mediante el comando SQL adecuado el procedimiento que permite recuperar datos originales para llevar a cabo su función. 2. El usuario introduce los parámetros de entrada requeridos. 3. El sistema recupera los datos originales de la tabla.
Flujos alternativos	2.a El usuario introduce los parámetros de entrada de forma incorrecta, la bbdd informa del error y el caso de uso continua en el paso 1.
Postcondición	El procedimiento recupera los datos originales (de la columna de la tabla) seleccionados.

Figura 5.9: Caso de uso: Recuperar datos originales

Caso de Uso - 09	Indicar datos para anonimizar
Descripción	El usuario inserta los datos que desea anonimizar.
Secuencia normal	1. El usuario introduce como parámetros de la función los datos que desea anonimizar.
Postcondición	La función se ejecuta.

Figura 5.10: Caso de uso: Indicar datos para anonimizar

Caso de Uso - 10	Indicar posiciones iniciales y finales
Descripción	El usuario inserta las posiciones iniciales y finales para anonimizar.
Secuencia normal	1. El usuario introduce como parámetros de la función o procedimiento las posiciones iniciales y finales para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.11: Caso de uso: Indicar posiciones iniciales y finales

Caso de Uso - 11	Indicar carácter especial
Descripción	El usuario inserta el carácter especial para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro de la función o procedimiento el carácter especial para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.12: Caso de uso: Indicar carácter especial

Caso de Uso - 12	Indicar posición de reanudación
Descripción	El usuario inserta la posición de reanudación para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro del procedimiento la posición de reanudación en la tabla para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.13: Caso de uso: Indicar posición de reanudación

Caso de Uso - 13	Indicar tabla valores importados
Descripción	El usuario inserta el nombre de la tabla de valores importados para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro del procedimiento el nombre de la tabla de valores importados para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.14: Caso de uso: Indicar tabla valores importados

Caso de Uso - 14	Indicar tipo de datos
Descripción	El usuario inserta el tipo de datos para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro del procedimiento el tipo de datos para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.15: Caso de uso: Indicar tipo de datos

Caso de Uso - 15	Indicar rango de números
Descripción	El usuario inserta el rango de números para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro del procedimiento el rango de números para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.16: Caso de uso: Indicar rango de números

Caso de Uso - 16	Indicar clave primaria
Descripción	El usuario inserta el nombre de la clave primaria para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro del procedimiento el nombre de la clave primaria para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.17: Caso de uso: Indicar clave primaria

Caso de Uso - 17	Indicar nombre de tabla externa
Descripción	El usuario inserta el nombre de la tabla externa para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro del procedimiento el nombre de la tabla externa para anonimizar o recuperar datos originales.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.18: Caso de uso: Indicar nombre de tabla externa

Caso de Uso - 18	Indicar columna a modificar
Descripción	El usuario inserta el nombre de la columna para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro del procedimiento el nombre de la columna para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.19: Caso de uso: Indicar columna a modificar

Caso de Uso - 19	Indicar nombre de tabla a enmascarar
Descripción	El usuario inserta el nombre de la tabla para anonimizar.
Secuencia normal	1. El usuario introduce como parámetro del procedimiento el nombre de la tabla para anonimizar.
Postcondición	La función o procedimiento se ejecuta.

Figura 5.20: Caso de uso: Indicar nombre de tabla a enmascarar

Capítulo 6

Diseño

En este apartado se explicarán los distintos pasos seguidos para llevar a cabo el diseño del proyecto y se incluirán varios diagramas que ayudarán a explicarlo (varios de ellos se han adaptado al objetivo del proyecto ya que no se trata del diseño de una aplicación).

En primer lugar, se mostrará mediante un modelo de dominio una visión general de las diferentes entidades que juegan un papel en el proyecto. El modelo conceptual y lógico de la base de datos detallarán la estructura de las bases de datos que serán posibles de utilizar con el kit de herramientas. A continuación, se mostrará el diagrama de paquetes (muy sencillo ya que no se trata de una aplicación) y el diagrama de secuencia, que ayudará a comprender de una forma sencilla y visual el proceso de uso de las herramientas y la anonimización de los datos. Finalmente, el diagrama de despliegue mostrará los distintos componentes y su relación que forman parte del kit de herramientas.

6.1. Modelo de dominio

El modelo de dominio que se puede apreciar a continuación, representa la visión general de las entidades que forman parte del proyecto. La clase "Datos sin anonimizar (originales)" representa al conjunto de los datos en la base de datos y cada una de las instancias de esta clase se corresponde a cada una de las filas de una tabla con datos en la base de datos. A su vez, un dato está formado por unos "Atributos": con su nombre (el nombre de la columna), tipo y valor.

Para anonimizar los datos originales, existen varias técnicas de anonimización disponibles en el toolkit de herramientas y entre las cuáles, habrá una que se adecúe más al tipo de dato que se quiera enmascarar. Cada instancia de la clase "Datos sin anonimizar (originales)" se corresponde con una instancia de "Datos anonimizados". Esta última contiene las filas de datos de una tabla ya anonimizados y de nuevo están compuestos por unos atributos los cuáles (los valores) son los que han sufrido la anonimización tras aplicar la técnica.

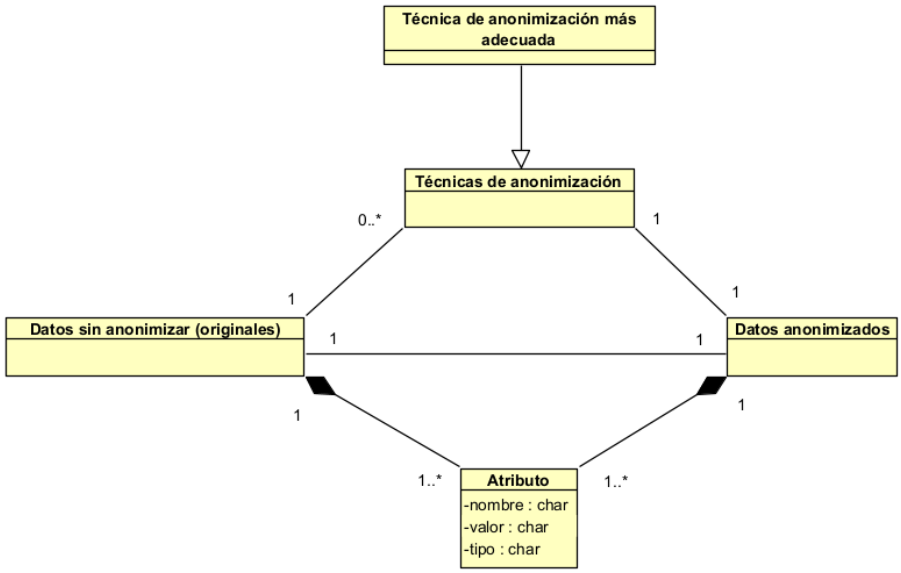


Figura 6.1: Modelo de dominio

6.2. Metamodelo de la base de datos

Para el análisis de diseño, suele ser común realizar una modelo de la base de datos cuando estas entran en juego en el proyecto (en este aspecto, para este proyecto serán necesarias cuando se utilicen las herramientas "procedimientos"). En este caso, dado que durante el proceso de anonimización se va a necesitar tratar con información del esquema de los datos además de los propios datos, realizaremos un metamodelo de la base de datos.

La base de datos va a estar formada (además de por las tablas originales del usuario), por otras tablas adicionales que serán creadas dinámicamente por las herramientas de anonimización según se ejecuten y que tendrán diferentes funciones. Algunas de ellas servirán de apoyo para el proceso de anonimización y serán borradas al finalizar la ejecución con el

fin de evitar almacenar cualquier tipo de información sensible. Por otro lado, algunas de esas tablas creadas no serán destruidas ya que tienen como objetivo servir de apoyo al usuario.

Por lo tanto, en esta sección se describirán las tablas que formarán parte de la base de datos entendiendo que, para cada una de las herramientas del toolkit siempre existirán dichas tablas.

6.2.1. Metamodelo conceptual

El metamodelo conceptual reflejará el análisis de las entidades de la base de datos, es decir, aquellos datos que habrá que usar/almacenar para llevar a cabo los procesos de anonimización.

- DATOS_ORIGINALES(atributo1, atributo2, atributo3, atributo4...): Estos son los datos originales contenidos en una entidad de la base de datos y que el usuario quiere anonimizar. Tendrá N atributos, tantos como columnas tenga dicha entidad.

- COMODIN(atributo1, atributo2, atributo3, atributo4... , index): Esta entidad es creada dinámicamente por las herramientas en el proceso de anonimización, y tiene como fin guardar (hacer una backup) los datos originales momentáneamente. Esto, permitirá borrar los datos de la entidad original, añadir ahí un atributo index (se parte de la idea de que la entidad original de datos no contiene un atributo de este tipo, siendo necesario) y traspasar de nuevo los datos a la entidad original.

  Se lleva a cabo este proceso porque no se puede añadir un nuevo atributo a una entidad previamente creada y que los nuevos valores tengan relación con los previos. Finalmente, para evitar comprometer datos sensibles, esta entidad es borrada.

- SEGURA(atributo1, atributo2, index): Las columnas que forman parte de esta entidad son dos (a parte del index): 'datosOriginales' que contiene los valores de los datos originales antes de ser anonimizados y formando relación uno a uno, la columna 'datosAnonimizados' siendo estos los valores generados tras el proceso de anonimización.

  Esta entidad será creada de la forma explicada para algunas herramientas y para otras solo mantendrá la columna 'datosOriginales'.

  Esta entidad tiene como fin servir de apoyo al usuario para la recuperación de los datos originales y que está enfocada al uso en entornos educacionales.

A continuación, se analizan las relaciones que existen entre las entidades descritas. Las relaciones son:

- Copia (Datos_Originales, Comodin): Los datos originales son copiados a la tabla Comodin.

- Guarda (Datos_Originales, Segura): Los datos originales se guardan en la tabla Segura.

Las cardinalidades que se generan entre las relaciones son:

- card_min (Datos_Originales, Copia) = 1

- card_max (Datos_Originales, Copia) = N

Para poder realizar la copia debe al menos existir como mínimo un dato original (una columna con valores). Si no, no habría nada que anonimizar. Sin embargo, puede haber 1 o más datos originales que se deseen anonimizar y por tanto copiar.

- card_min(Comodin, Copia) = 1

- card_max(Comodin, Copia) = 1

Como mínimo, para realizar la copia debe existir una entidad comodin por cada entidad de datos originales. En este caso, no habrá más de una entidad comodin (no es necesario) y por lo tanto la cardinalidad máxima es 1.

- card_min (Datos_Originales, Guarda) = 1

- card_max (Datos_Originales, Guarda) = 1

Para poder guardar en la entidad segura debe como mínimo haber un dato original (una columna con valores). En la entidad segura se guardará como máximo un dato original ya que las herramientas trabajan de uno en uno.

- card_min(Segura, Guarda) = 1

- card_max(Segura, Guarda) = 1

Se creará como mínimo una entidad segura por cada dato original (una columna con valores) que se quiera anonimizar. Como máximo se creará 1 entidad segura con el fin de exponer lo menos posible los datos originales y anonimizados.

El modelo conceptual se representa a continuación en la Figura 6.2:

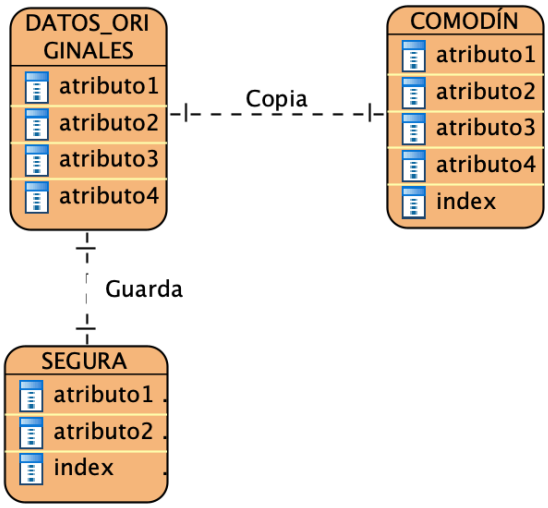


Figura 6.2: Modelo conceptual de la base de datos

En la entidad COMODIN, si bien se ha comentado antes que el atributo index se añade a la entidad DATOS_ORIGINALES y esta primera solo sirve de backup, para hacer más entendible el modelo, supondremos que COMODIN representa la versión de respaldo y la forma final de DATOS_ORIGINALES.

6.2.2. Metamodelo lógico

Para pasar del modelo conceptual al modelo lógico, transformamos las entidades descritas en el apartado anterior creando una tabla por cada entidad. Las tablas resultantes son:

- DATOS(atributo1, atributo2, atributo3, atributo4...)

- COMODIN(atributo1, atributo2, atributo3, atributo4. . . , index)

- SEGURA(atributo1, atributo2, atributo3, atributo4. . . , index)

Ahora estudiamos las relaciones existentes entre las tablas. Dado que ambas relaciones son uno a uno, las transformaríamos incluyendo todo en una misma tabla de la forma:

- SEGURA(atributo1, atributo2, index. . .)

Esta tabla no podría llegar a ser construida por razones de seguridad y eficiencia. Para que las herramientas sean capaces de realizar la anonimización de los datos y construir la tabla de apoyo para el usuario, mantendremos las tres tablas por separado.

Las tablas resultantes son las que se muestran a continuación en la Figura 6.3.

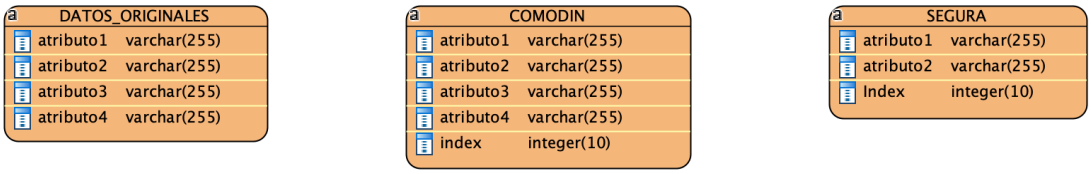


Figura 6.3: Modelo lógico de la base de datos

EJEMPLO: Para entender de una forma más simple este modelo y la forma en que se crean las tablas, se muestra a continuación un ejemplo con una posible instancia de una base de datos.

Tenemos una tabla 6.4a en la que se almacenan datos sobre elecciones, con tres atributos: ciudad, candidato y el código de la ciudad. El usuario desea anonimizar el atributo candidato porque lo considera un dato sensible y selecciona la herramienta que más se adecua para este caso. Una vez introducidos los parámetros, esta se ejecuta.

Tras varios pasos, se ejecuta el proceso en el que se crea una tabla “Comodin” de la forma que se muestra en la tabla 6.4b.

ciudad	candidato	codCidudad
Burgos	Pedro	1606
Alicante	Juan	6098
Galicia	Silvia	1721
Caceres	Maria	1566

(a) Tabla original

index	ciudad	candidato	codCidudad
1	Burgos	Pedro	1606
2	Alicante	Juan	6098
3	Galicia	Silvia	1721
4	Caceres	Maria	1566

(b) Tabla comodin

Figura 6.4: Tablas del proceso de anonimización

A continuación, la herramienta continuará su trabajo restaurando la tabla original con los datos originales más la columna index y sobre ella se realizará el proceso de anonimización. Finalmente, se creará una nueva tabla de apoyo al usuario que contendrá el par de datos originales y anonimizados como se muestra en la tabla 6.5.

datOriginal	datoAnonimizado
Pedro	P****
Juan	J***
Silvia	S*****
Maria	M****

Figura 6.5: Tabla segura

Se recuerda que no todas las herramientas crean una tabla segura con esta estructura, algunas simplemente guardan los valores originales.

6.3. Diagrama de paquetes

De forma muy simple ya que este proyecto no se enfoca en la creación de una aplicación, se mostrará mediante el diagrama de paquetes de la siguiente figura, las distintas clases e interfaces que forman parte del proyecto.

Como se puede ver, la vista (interfaz) a través de la cual interactuará el usuario con las herramientas del toolkit es el terminal de un ordenador. Con él, podrá acceder al servidor de la base de datos (que se presenta como un controlador ya que es quien maneja algunas de las principales órdenes) y posteriormente a la base de datos donde realizará la anonimización de los datos mediante las herramientas.

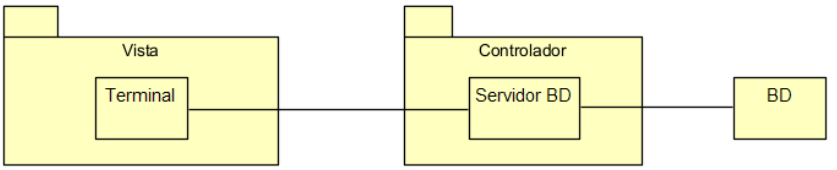


Figura 6.6: Diagrama de paquetes

6.4. Diagrama de secuencia

En este apartado se describirá mediante un diagrama de secuencia, los pasos que seguirá el usuario para llevar a cabo la anonimización de los datos que desee.

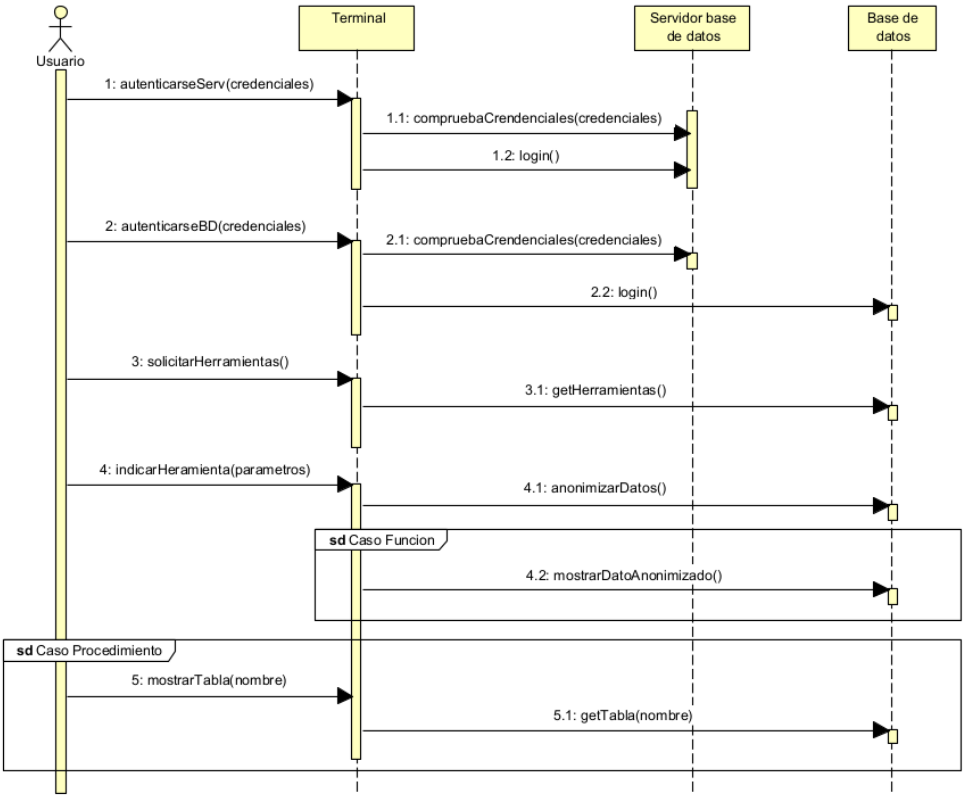


Figura 6.7: Diagrama de secuencia para anonimizar

Como se puede observar, el usuario siempre interactuará con el terminal ya que se trata de la única interfaz de la que se dispondrá. Esta interfaz puede variar, siendo posible que sea un terminal o similar.

En primer lugar, el usuario introducirá sus credenciales para acceder al servidor de baso de datos y posteriormente para acceder a la base de datos. Una vez se ha accedido, el usuario procederá, mediante los comandos SQL adecuados, a solicitar las herramientas que hay disponible y una vez decidido cuál se adecua más para su caso, indicará (escribirá) los parámetros necesarios para su ejecución.

Tras realizarse el proceso de anonimización, existen dos casos posibles: si se trata de una función, el termina mostrará por pantalla la solución de la anonimización directamente; en caso de tratarse de un procedimiento, entonces el usuario deberá solicitar mostrar la tabla con los datos que se han anonimizado.

6.5. Diagrama de despliegue

En el diagrama de despliegue que se muestra en la Figura 6.8, se muestra cómo aparecen las particiones físicas del sistema de información y la asignación dentro de estas de los distintos componentes software.

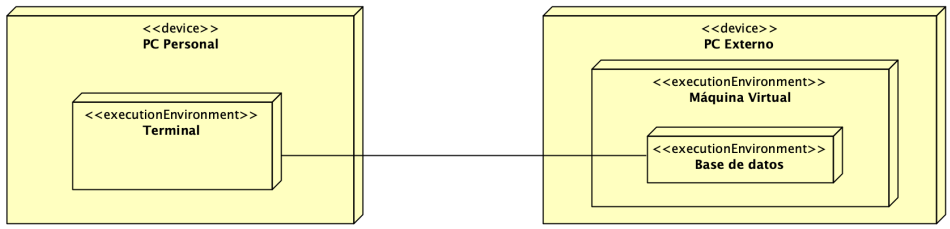


Figura 6.8: Diagrama de despliegue

Como se puede observar, contamos con dos dispositivos principales: un PC (ordenador personal) donde se encontrará alojada la aplicación terminal y un PC donde se ha creado una máquina virtual y dentro de ella se ha instalado un servidor de base de datos que alojará la base de datos. Ambos estarán conectados constantemente durante el proceso de anonimización permitiendo el intercambio de información y su almacenamiento en la base de datos.

Este modelo puede variar según donde tengamos alojada la base de datos. Es posible que

podamos tener el servidor instalado en nuestro propio PC, una máquina virtual o en Cloud.

Capítulo 7

Implementación

7.1. Tecnología utilizada

Este proyecto se basa en la implementación de un conjunto de herramientas (toolkit) que permitan la anonimización y que su manejo sea lo más sencillo posible, ya que principalmente están orientadas para ser distribuidas en entornos educacionales y usadas por alumnos (aunque también son compatibles para otro tipo de usuario). Debido a que, en estos entornos, aunque principalmente se usa el sistema operativo de Windows, hay alumnos que utilizan otros distintos (Linux, MacOS...) y por tanto la distribución de una aplicación podría suponer algunas barreras. En consecuencia, se ha decidido realizar la implementación en ficheros con la extensión adecuada que pudieran ser fácilmente puestos a disposición de los alumnos y ser usados por ellos.

La tecnología entorno a la que gira este proyecto son las bases de datos relacionales, en concreto MySQL (de Oracle) [20], ya que es la utilizada principalmente en la universidad. Los ficheros contienen la implementación del código escrito en lenguaje SQL (Structured Query Language) que es el utilizado en las de bases de datos. Para la escritura del código se ha utilizado el software Visual Studio Code (permite escribir el código de forma ordenada y clara, gracias a colores y otros aspectos) y MySQL Workbench, que permite de una forma más interactiva trabajar con bases de datos. El editor del que se ha hecho uso para escribir la memoria ha sido Overleaf [23], con el apoyo de Visual Paradigm [21], Microsoft Project [22] y Microsoft Excel para la implementación de algunos de los diagramas o tablas expuestas en esta memoria.

7.2. Importación del toolkit

El toolkit está formado por dos ficheros con extensión SQL. Esta extensión permite que sean cargados y usados en bases de datos MySQL.

Para su importación, supondremos que se está utilizando el modelo en el que disponemos de una máquina virtual y en ella hemos instalado el servidor de base de datos (es el caso que se suele dar en la universidad). El usuario debe, mediante el terminal, escribir el comando que se muestra a continuación:

```
scp -P 24252 /home/Funciones.sql
↪  usuario@virtual.lab.inf.uva.es:/var/lib/mysql-files
```

Con este comando, "scp", se especifica primero mediante "-P XXXX" el puerto de la máquina virtual, a continuación, la ruta local del fichero que queremos importar y finalmente la dirección del host junto con la ruta donde queremos guardar el fichero.

Cuando se trata de bases de datos, suele existir una ruta recomendable donde guardar los ficheros sensibles y esta es "/var/lib/mysql-files". Esto, se debe a que esa ruta normalmente (por defecto) está asociada a una variable de entorno de las bases de datos conocida como "secure_file_priv" [24], y que, de estar habilitada, deniega ciertas acciones y accesos a los ficheros contenidos en dicha ruta a aquellos usuarios que no tienen los permisos adecuados (suele ser accesible únicamente al administrador o root como medida de seguridad).

En este caso, bien se pueden guardar ahí o en cualquier otra ruta que el usuario considere preferible ya que se trata de código para uso en entornos educacionales. Pero hay que tener presente, que de ser otro entorno en el que se use, no proteger el código puede suponer su robo y conocer la forma en que las técnicas anonimizan los datos.

Una vez importados los ficheros, se escribirá el siguiente comando en el que se especifica la base de datos donde queremos importar las funciones/procedimientos:

```
sudo mysql -u root nombreBBDD < /ruta/Funciones.sql
```

[Para la importación de funciones, es posible que la base de datos informe de un problema. En este caso, la solución es activar la variable de entorno "log_bin_trust_function_creators" mediante el siguiente comando:]

```
SET GLOBAL log_bin_trust_function_creators = 1
```

7.3. Implementación de herramientas

En este apartado se mostrará algunos de los trozos de código más significativos que han sido utilizados para escribir las funciones y procedimientos que componen el toolkit.

- Comenzando con las funciones, para la herramienta más genérica (se detallará en el siguiente apartado), se proponen dos formas de implementación: una que consume más recursos computacionales y otra que menos.

  Esto, nuevamente está pensado para que el alumno pueda descubrir distintas formas de programar una función de anonimización.

  La primera de las formas, se trata de ir recorriendo mediante un bucle "while" el valor introducido por el usuario (ej, un string) carácter por carácter. Se coge el primer carácter, se analiza si debe ser anonimizado o no y se añade a una nueva variable. Se coge el segundo, se anonimiza o no, se concatena al anterior carácter y así sucesivamente hasta completar el valor introducido.

  ```
  while (i <= str_length) do
      if (i > long_inicial and i <= str_length-long_final) then
          set output_string = CONCAT(SUBSTR(output_string, 1, i-1),
          ↪  REPEAT(caracter, 1));
      else
          set output_string = CONCAT(SUBSTR(output_string, 1, i-1),
          ↪  SUBSTR(input_string, i, 1));
      end if;
      set i = i + 1;
  end while;
  ```

  La segunda forma y que consume menos recursos, se trata de usar la función CONCAT() junto con REPEAT() [25]. La primera, permite concatenar partes de un string especificando las posiciones inicial y final; la segunda, permite escribir repetidamente un carácter tantes veces como se especifique. Con la combinación de ambas, conseguimos el mismo resultado que la otra forma de programación, pero en apenas una línea de código.

  ```
  set output_string = CONCAT(SUBSTR(input_string, 1, long_inicial),
  ↪  REPEAT(caracter, total_symbol), SUBSTR(input_string,
  ↪  long_inicial + total_symbol + 1, str_length));
  ```

  Ambas formas serían aplicables al resto de funciones, aunque solo se ha programado de esta manera la herramienta más genérica con el objetivo de que el resto consumieran

pocos recursos. Todas las funciones siguen un patrón de programación similar, aunque con las diferencias necesarias para llevar a cabo las distintas técnicas de anonimización.

- La programación de los procedimientos tiene una implementación un poco distinta ya que entran en juego tablas de bases de datos.

  Como parte común a todos ellos, y ya comentado en el apartado de "Metamodelo Lógico", estos, primero crean una tabla "segura" donde guardar los pares "valorOriginal – valor Anonimizado" (algunas herramientas, otras solo guardan los valores originales) y una tabla "comodin o backup" que servirá para tener momentáneamente guardados los datos originales y realizar una copia de los valores.

  A continuación, se borran los valores de la tabla original y se obtienen los nombres de las columnas que la forman. Para llevar a cabo este proceso, se accede a la tabla Information Schema que alberga metadatos de la base de datos, entre los cuales se encuentran los nombres de las columnas de la tabla que se quiera. Además, dado que luego será necesario su uso, con la función "group_concat()" conseguimos que se devuelva un string con los nombres concatenados del grupo que lo forman.

  Después, se elimina la clave primaria de la tabla original en caso de que exista ya que esta pasará a pertenecer a la columna index auto incremental que añadiremos a continuación. Finalmente, traspasamos los valores originales de la tabla "comodin o backup" a la tabla original y borramos esta primera para evitar comprometer datos sensibles.

```
if (colum_primary_key != 'idP') then
    -- Hacemos una copia de los datos de la tabla original a una de
    ↪  backup
    drop table if exists backupTable;
    set @snt = CONCAT('create table backupTable like ',
    ↪  tab_a_modificar);
    prepare sent from @snt;
    execute sent;
    deallocate prepare sent;

    set @snt = CONCAT('insert backupTable select * from ',
    ↪  tab_a_modificar);
    prepare sent from @snt;
    execute sent;
    deallocate prepare sent;

    -- Borramos los datos de la tabla original
    set @snt = CONCAT('truncate table ', tab_a_modificar);
    prepare sent from @snt;
    execute sent;
    deallocate prepare sent;
```

```
        -- Obtenemos los nombres de las columnas de la tabla
        ↪  original
        set @snt = CONCAT('set @nomb_colum_tabla_salida = (select
        ↪  group_concat(COLUMNS.COLUMN_NAME order by
        ↪  COLUMNS.ordinal_position) from INFORMATION_SCHEMA.COLUMNS
        ↪  where TABLE_NAME = ',"'",tab_a_modificar,"'",')');
        prepare sent from @snt;
        execute sent;
        deallocate prepare sent;

        set nomb_columns_tabla = @nomb_colum_tabla_salida;

        -- Eliminamos la primary key de la tabla original en caso de
        ↪  que exista
        if (colum_primary_key != '') then
            set @snt = CONCAT('alter table ',tab_a_modificar,' drop
            ↪  primary key');
            prepare sent from @snt;
            execute sent;
            deallocate prepare sent;
        end if;

        -- Anadimos a la tabla original una columna auto increment como
        ↪  primary key
        set @snt = CONCAT('ALTER TABLE ',tab_a_modificar,' ADD idP int
        ↪  unsigned not null AUTO_INCREMENT, ADD PRIMARY KEY (idP)');
        prepare sent from @snt;
        execute sent;
        deallocate prepare sent;

        -- Copiamos de vuelta los datos a la tabla original
        set @snt = CONCAT('insert ', tab_a_modificar,'
        ↪  (',nomb_columns_tabla,') select * from backupTable');
        prepare sent from @snt;
        execute sent;
        deallocate prepare sent;

        -- Eliminamos la tabla de backup
        drop table backupTable;

    end if;
```

El resto de código hace lo propio para anonimizar los datos siguiendo un estilo de programación similar al de las funciones, de nuevo diferenciándose en los casos particulares para cada tipo de técnica.

Finalmente, como medida para mantener las tablas de la forma más original posible, se llevan a cabo una serie de pasos en función de cuál era la clave primaria original. En caso de pertenecer a un atributo que no es el que se quiere anonimizar o bien la tabla no tiene clave primaria, en este caso, tras la anonimización se restaura la clave primaria a su original y se elimina la columna index.

Si la clave primaria coincide con el atributo que se desea anonimizar, dado que tras la anonimización puede llegar a existir valores iguales, entonces en este caso no se elimina la columna index y se mantiene como clave primaria.

```
if (colum_primary_key != colum_a_modificar and colum_primary_key !=
↪  'idP') then

    -- Eliminacion del index en la tabla original
    set @snt = CONCAT('alter table ', tab_a_modificar,' drop column
    ↪  idP');
    prepare sent from @snt;
    execute sent;
    deallocate prepare sent;

    -- Restauramos la primary key de la tabla original en caso de
    ↪  que exista
    if (colum_primary_key != '') then
        set @snt = CONCAT('alter table ',tab_a_modificar,' add
        ↪  primary key (',colum_primary_key,')');
        prepare sent from @snt;
        execute sent;
        deallocate prepare sent;
    end if;
end if;
```

NOTA: Las tablas “seguras” que se crean para guardar los pares “valorOriginal- valorAnonimizado” están pensadas con el fin de que sirvan de apoyo a los alumnos para recuperar los datos originales en caso de necesitarlos y para que tengan una forma sencilla e intuitiva de comparar las dos formas de los valores.

En un entorno empresarial, estas tablas no deberían crearse ya que a nivel de seguridad pueden suponer un riesgo grande en caso de ser expuestas o robadas, la información anonimizada no serviría de nada.

Además, el nombre de tablas “seguras” está pensado para que los alumnos tomen conciencia de que esas tablas deberían asegurarlas mediante niveles de seguridad mayores (con los conocimientos adquiridos en la asignatura) como por ejemplo restringiendo su acceso a determinado tipo de usuarios.

7.4. Propuesta de técnicas y herramientas de anonimización

En este apartado se describirá el qué realizan las técnicas de anonimización utilizadas en la implementación de las herramientas que se incluyen en el toolkit, y el cómo lo hacen mediante las funciones y procedimientos creados para este proyecto.

Aunque se ha hecho incapié en el Apartado [3] de Fundamentos Teóricos, es importante la diferenciación entre anonimización y seudonimización. La anonimización trata de desvincular los datos personales de los datos identificativos, haciendo que estos no puedan ser reidentificados o llegar a ser inferidos. Por otro lado, la seudonimización no crea esa desvinculación, lo que genera la posibilidad de, mediante información adicional, volver a identificar un dato con su individuo.

Es importante esta diferencia, ya que para algunas de las técnicas (ej: hashing o sustitución de caracteres) utilizadas en la implementación, es especialmente difícil establecer una línea de diferencia en su aplicación a la seudonimización o la anonimización.
Para este proyecto, se establece la suposición de que este tipo de técnicas (junto con las herramientas del toolkit implementadas) aplicadas a un conjunto suficientemente grande de datos, permite la anonimización de estos no pudiendo ser inferidos o reidentificados.

7.4.1. Técnicas

Las técnicas de anonimización escogidas para su implementación son principalmente cinco:

- **Intercambio de caracteres:** Esta técnica anonimiza un valor, sustituyendo un conjunto de caracteres reconocibles (letras o números) del dato original, por caracteres especiales, como por ejemplo un $*$ o una X.

  Ej: *"Esto es un ejemplo"* $>$ *"Es********emplo"*

- **Aleatorización usando diccionario de datos:** Esta técnica anonimiza un conjunto de valores, sustituyendo estos por datos sintéticos (de forma aleatoria) los cuales son del mismo tipo y con igual significado (esto permite que, en caso de robo de datos, no exista la sospecha de datos anómalos). Para la obtención de los datos sintéticos, se deberá construir un diccionario de datos que contenga los valores.

Ej: Supongamos que existe una tabla en base de datos, en la que hay una columna con nombres de ciudades, y se desea anonimizar mediante la técnica descrita en este punto.

Para ello, se generará un diccionario de datos en el que se incluirá nombres de ciudades distintos a los de la tabla, siempre manteniendo una cierta concordancia a la situación. Es decir, si la columna de la tabla tiene nombres de ciudades españolas, en el diccionario de datos sería conveniente que se incluyeran nombres de ciudades españolas.

El proceso de anonimización lo que hará será, sustituir los nombres de ciudades de la tabla por nombres cogidos del diccionario de forma aleatoria.

Valores Originales		**Valores Anonimizados**
valladolid	>	*palencia*
burgos	>	*madrid*
tarragona	>	*sevilla*
valencia	>	*palencia*

El objetivo de esta técnica, es que en caso de que el intruso acceda a los datos, este seguirá viendo nombres de ciudades españolas y no pueda sospechar de que se ha producido una anonimización en los datos.

- **Promedio**: Esta técnica anonimiza un conjunto de valores numéricos, calculando el valor promedio a partir de los datos originales y sustituyendo cada uno de ellos por dicho valor promedio. Cabe destacar, que para aplicar esta técnica es recomendable que el conjunto de valores sea relativamente grande para evitar una posible reidentificación de los datos.

Valores Originales		**Valores Anonimizados**
12	>	*101*
28	>	*101*
119	>	*101*
245	>	*101*

- **Hashing**: Esta técnica anonimiza un valor de cualquier longitud, sustituyendo el dato original por un valor hash compacto y de longitud fija, calculado mediante una función criptográfica. En la implementación de esta técnica, se ha utilizado la función hash SHA-256 ya que es una de las que ofrece mayor nivel de seguridad a día de hoy.

Ej: *"Esto es un ejemplo"* >

"38e1cad08cd88efd203280451c0a415454a5d2c14c1a0d79bc5c77d295726cc5"

- **Funciones matemáticas:** Esta técnica anonimiza un valor, realizando una operación matemática sobre el dato original y lo sustituye por el resultado de dicha operación. Es recomendable que esta técnica sea usada en conjuntos de valores relativamente grandes.

  Además, el uso de esta técnica permite que el dato original sea recuperado tras la anonimización (sin haber sido guardado en ninguna otra parte) realizando una operación inversa a la usada para enmascarar el dato. Esta recuperación no significa que los valores sean susceptibles de ser reidentificados, sino que para recuperarlos se debe ejecutar una herramienta concreta.

  Para entenderlo de una forma más sencilla, se muestra a continuación un ejemplo:

  Suponemos que tenemos en una tabla, una columna con valores numéricos y se anonimizan mediante la operación de multiplicación x2:

Valores Originales		**Valores Anonimizados**
12	>	*24*
28	>	*56*
119	>	*238*
245	>	*490*

  Hasta este punto, es lo que se consideraría la técnica y proceso de anonimización. Pero como se ha comentado, esta técnica además permite la recuperación de los valores originales mediante la aplicación de la operación inversa, en este caso sería la división ÷2.

Valores Anonimizados		**Valores originales**
24	>	*12*
56	>	*28*
238	>	*119*
490	>	*245*

7.4.2. Herramientas

A continuación se explicará las diferentes herramientas que se han creado para poder implementar las técnicas descritas en el anterior apartado.

Para ello se han implementado dos tipos de herramientas a alto nivel: funciones y procedimientos.

- **Funciones**

  Las funciones, tienen como objetivo permitir al usuario anonimizar un solo dato a la vez de forma sencilla y rápida.

  Existe una variedad de este tipo de herramientas y cada una de ellas lleva a cabo la anonimización de una manera particular, pero todas ellas implementan la técnica de "Intercambio de caracteres".
  A su vez, las funciones se dividen en dos clases:

  → De propósito general: Las funciones de este apartado permitirán al usuario anonimizar datos de manera libre. Es decir, el dato puede ser de longitud variable y se puede escoger que partes del dato anonimizar y cuáles no.

  → De propósito específico: Las funciones de este apartado permitirán al usuario anonimizar datos de manera estricta. Es decir, el dato debe tener una longitud concreta y su anonimización estará prefijada.

  Aquí, se incluyen herramientas que permitirán anonimizar datos como: números de tarjetas o cuentas bancarias, el Número de la Seguridad Social (NUSS), el Documento de Identidad Nacional (DNI), la fecha de nacimiento o el código postal.

  La forma en que se anonimiza el DNI se adapta a las reglas propuestas por la AEPD [28]. El resto de datos siguen algunas pautas propuestas por MySQL y Oracle ya que la AEPD no las especifica.

- **Procedimientos**

  Los procedimientos, tienen como objetivo permitir al usuario anonimizar varios datos a la vez. De esta forma, se puede anonimizar una columna (con gran cantidad) de datos contenidos en una tabla de base de datos de forma rápida y sencilla.

  Existe una variedad de este tipo de herramientas y cada una de ellas lleva a cabo la anonimización de una manera particular. En este caso, los procedimientos implementan todas las técnicas explicadas en el apartado anterior, cada uno de ellos una técnica distinta como se indicará a continuación.
  A su vez, los procedimientos se dividen en varias clases:

- Procedimientos que utililizan la técnica de "Intercambio de caracteres": Esta colección de procedimientos, mediante esta técnica, anonimizarán los datos intercambiando caracteres legibles del dato original por caracteres especiales.

  → De propósito general: Los procedimientos de este apartado permitirán al usuario anonimizar datos de manera libre. Es decir, los datos pueden ser de longitud variable y se puede escoger que partes del dato anonimizar y cuáles no.

  → De propósito específico: Los procedimientos de este apartado permitirán al usuario anonimizar datos de manera estricta. Es decir, los datos deben tener una longitud concreta y su anonimización estará prefijada.

  Aquí, se incluyen herramientas que permitirán anonimizar datos como: números de tarjetas o cuentas bancarias, el Número de la Seguridad Social (NUSS), el Documento de Identidad Nacional (DNI), la fecha de nacimiento o el código postal.

  La forma en que se anonimiza el DNI se adapta a las reglas propuestas por la AEPD [28]. El resto de datos siguen algunas pautas propuestas por MySQL y Oracle ya que la AEPD no las especifica.

- Procedimiento que utiliza la técnica de "Aleatorización usando diccionario de datos": Este procedimiento, mediante esta técnica, permitirá al usuario anonimizar un conjunto de datos (texto o valores numéricos) sustituyendo los originales por datos sintéticos aleatorios obtenidos de un diccionario de datos (creado por el usuario).

- Procedimientos que utilizan la técnica de "Promedio" y "Hashing":
  → Promedio: Este procedimiento permitirá al usuario anonimizar un conjunto de datos numéricos, sustituyéndoles por el valor promedio calculado con los datos originales.

  → Hashing: Este procedimiento permitirá al usuario anonimizar un conjunto de datos alfanuméricos, sustituyéndo los originales por el valor hash calculado mediante la función criptográfica SHA-256.

- Procedimiento que utiliza la técnica de "Funciones matemáticas": Este procedimiento, permite al usuario anonimizar un conjunto de valores numéricos, sustituyéndo estos mediante la multiplicación de los valores originales por números aleatorios (dentro de un rango seleccionado por el usuario).

- Procedimiento que utiliza la técnica de "Funciones matemáticas" para recuperar datos originales: Este procedimento, mediante esta técnica, permite al usuario recuperar un conjunto de valores numéricos originales una vez estos han sido anonimizados. Esta herramienta implementa la división de los valores anonimizados

entre valores aleatorios (previamente utilizados para el proceso de anonimización).

Para el uso de este procedimiento, es necesario haber anonimizado previamente los datos numéricos con el procedimiento explicado en el punto anterior.

Esta es una explicación a grandes rasgos de las herramientas que se han implementado y qué técnicas utilizan cada una de ellas. Para una mayor comprensión de forma individual, de su descripción, su sintaxis y ejemplos de uso/resultado; se puede consultar el Manual de Usuario [A] o los ficheros (README.txt, Funciones_EjemplosUso.txt, Procedimientos_EjemplosUso.txt) [D] que se adjuntan con esta memoria.

Capítulo 8

Plan de pruebas

Para comprobar que todas las herramientas implementadas ejecutan sus funciones correctamente y el resultado es el esperado, se diseña un plan de pruebas a modo de validación.

8.1. Funciones

En esta sección, se validará el correcto funcionamiento de las funciones implementadas. Para cada una de ellas, se probrarán los dos escenarios posibles: el usuario inserta los parámetros de forma correcta (con variantes) o al contrario, inserta alguno de ellos incorrectamente y se genera un aviso.

8.1.1. Prueba 1.

Para este conjunto de pruebas, las funciones se ejecutarán de dos formas: la "estándar", se anonimizará con el caracter por defecto; y la "alternativa", se anonimizará con un caracter específico.

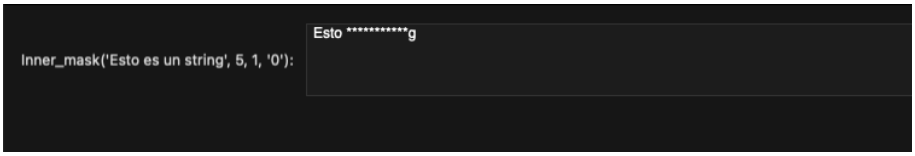


Figura 8.1: Ejecución estándar inner_mask()

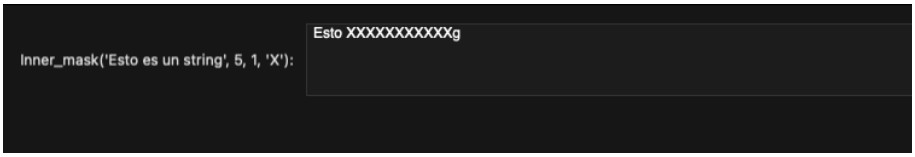


Figura 8.2: Ejecución alternativa inner_mask()

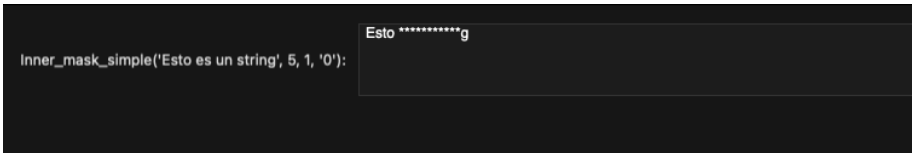


Figura 8.3: Ejecución estándar inner_mask_simple()

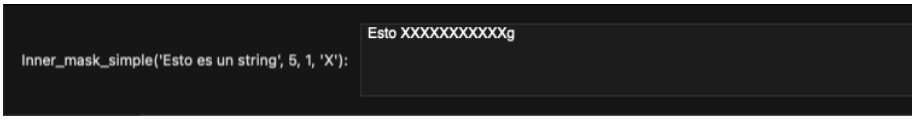


Figura 8.4: Ejecución alternativa inner_mask_simple()

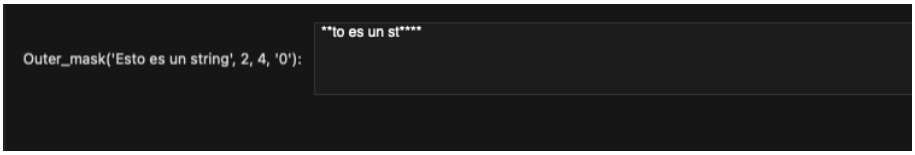


Figura 8.5: Ejecución estándar outer_mask()

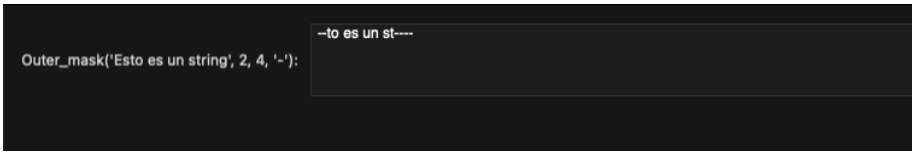


Figura 8.6: Ejecución alternativa outer_mask()

8.1.2. Prueba 2.

Para este conjunto de pruebas, las funciones se ejecutarán de dos formas: la "correcta", se introducen los parámetros de forma adecuada; y la "errónea", se introducen los parámetros de forma incorrecta.

Figura 8.7: Ejecución correcta numacc_mask()

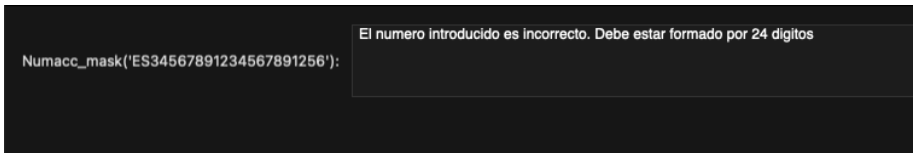


Figura 8.8: Ejecución errónea numacc_mask()

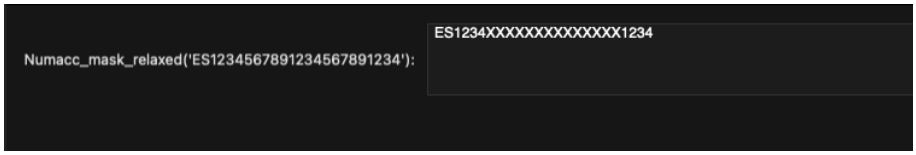


Figura 8.9: Ejecución correcta numacc_mask_relaxed()

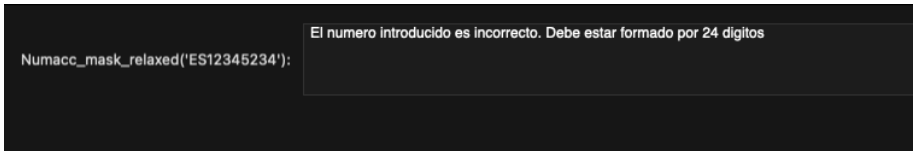


Figura 8.10: Ejecución errónea numacc_mask_relaxed()

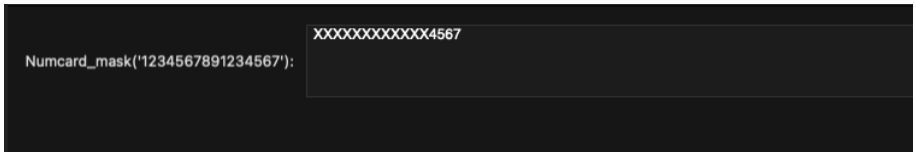


Figura 8.11: Ejecución correcta numcard_mask()

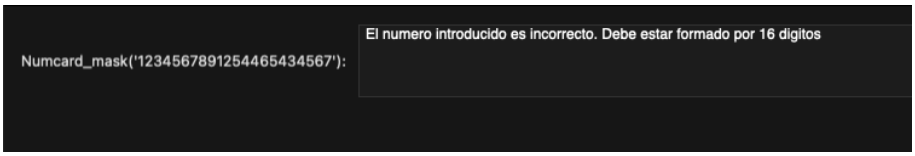


Figura 8.12: Ejecución errónea numcard_mask()

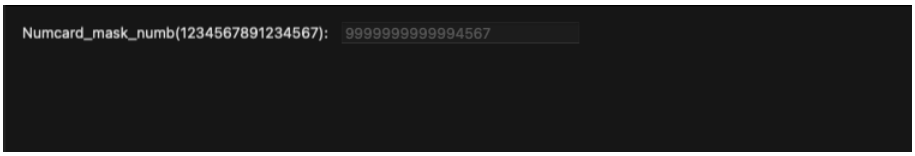


Figura 8.13: Ejecución correcta numcard_mask_numb()

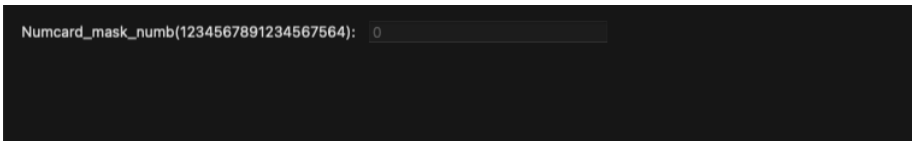


Figura 8.14: Ejecución errónea numcard_mask_numb()

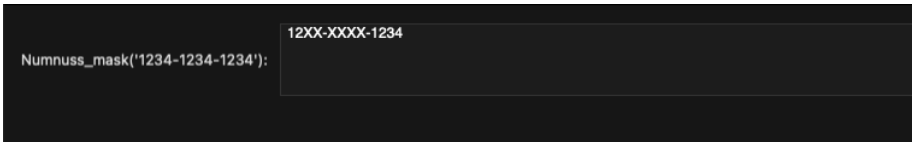


Figura 8.15: Ejecución correcta numnuss_mask()

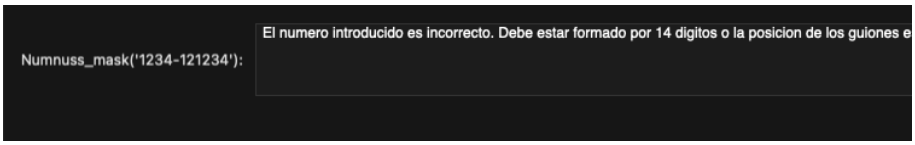


Figura 8.16: Ejecución errónea numnuss_mask()

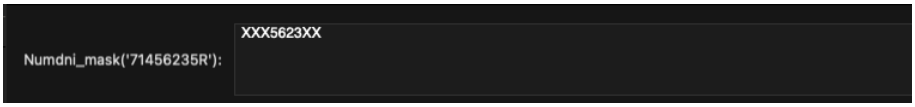


Figura 8.17: Ejecución correcta numdni_mask()

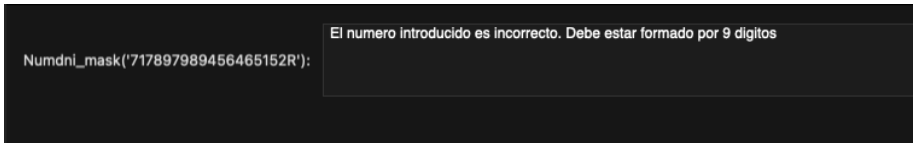


Figura 8.18: Ejecución errónea numdni_mask()

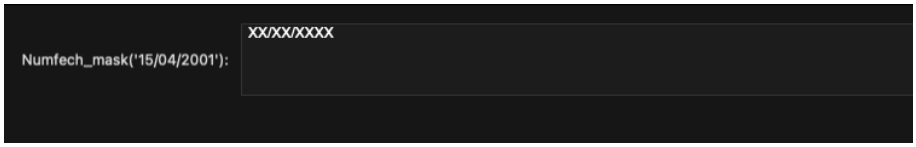


Figura 8.19: Ejecución correcta numfech_mask()

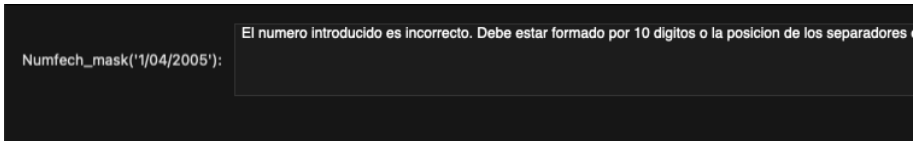


Figura 8.20: Ejecución errónea numfech_mask()

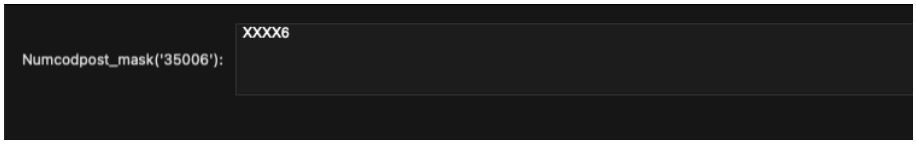


Figura 8.21: Ejecución correcta numcodpost_mask()

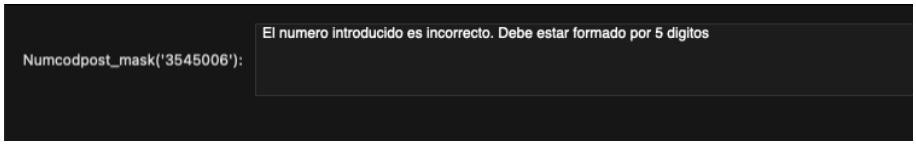


Figura 8.22: Ejecución errónea numcodpost_mask()

Para cada una de las funciones ejecutadas, aunque no se muestra, si el usuario introduce parámetros de más o de menos, los inserta con un tipo de dato inadecuado u algún otro error, la base de datos genera un aviso de dicho problema.

8.2. Procedimientos

En esta sección, se validará el correcto funcionamiento de los procedimientos implementados. Para cada uno de ellos, se probrarán los escenarios posibles: los datos de la tabla que se quieren anonimizar son correctos, variantes según el procedimiento, o alguno de los datos es incorrecto y se genera un aviso.

8.2.1. Prueba 3.

Para este conjunto de pruebas, los procedimientos se ejecutarán de dos formas: la "estándar", se anonimizará con el caracter por defecto; y la "alternativa", se anonimizará con un caracter específico.

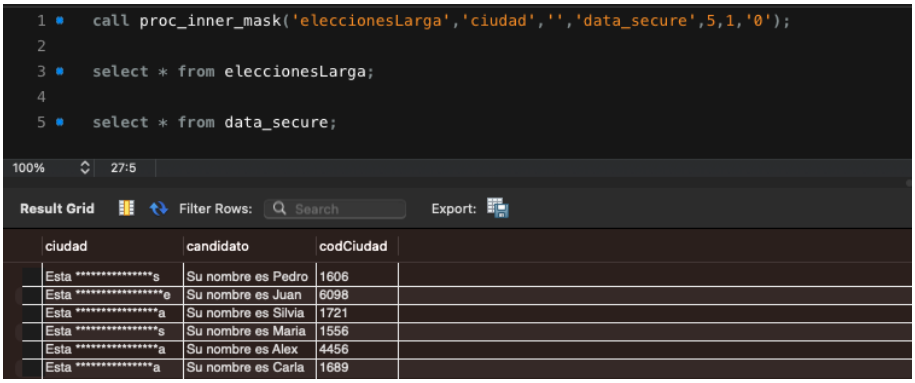


Figura 8.23: Ejecución estándar proc_inner_mask()

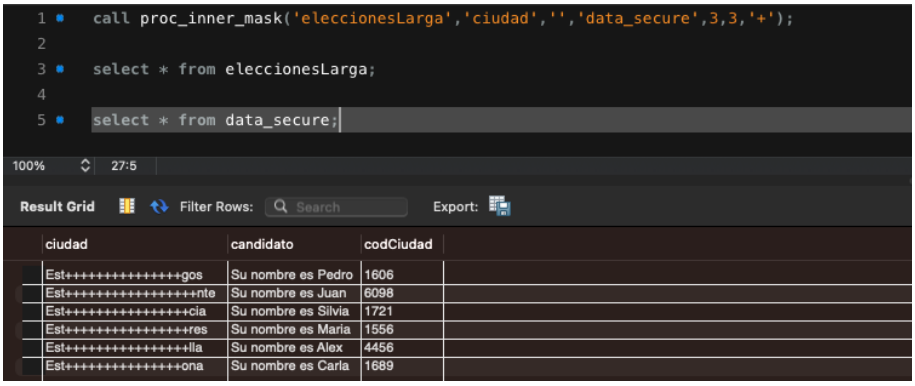


Figura 8.24: Ejecución alternativa proc_inner_mask()

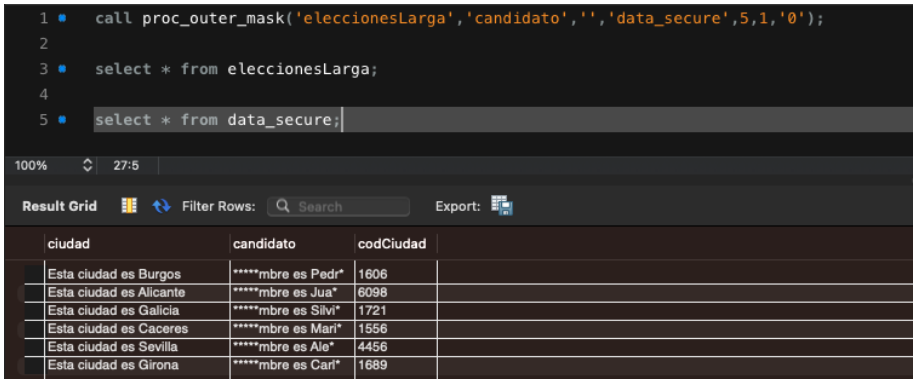


Figura 8.25: Ejecución estándar proc_outer_mask()

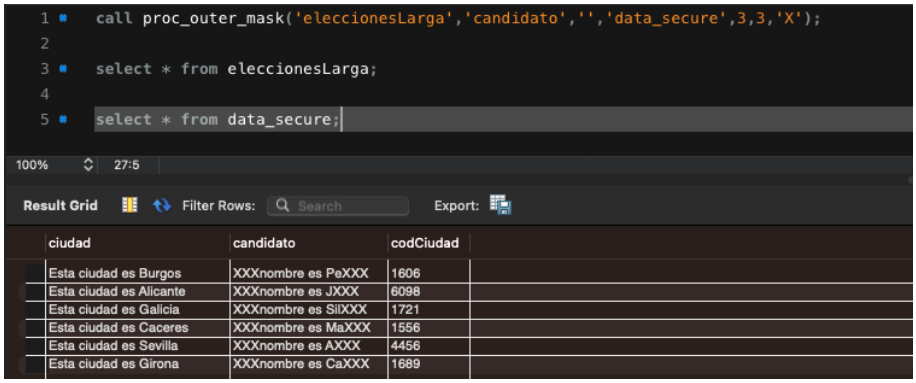


Figura 8.26: Ejecución alternativa proc_outer_mask()

8.2.2. Prueba 4.

Para este conjunto de pruebas, los procedimientos se ejecutarán de dos formas: la "correcta", los datos contenidos en la tabla que se quieren anonimizar son correctos; y la "errónea", alguno de los datos es incorrecto y se genera un aviso indicando la posición.

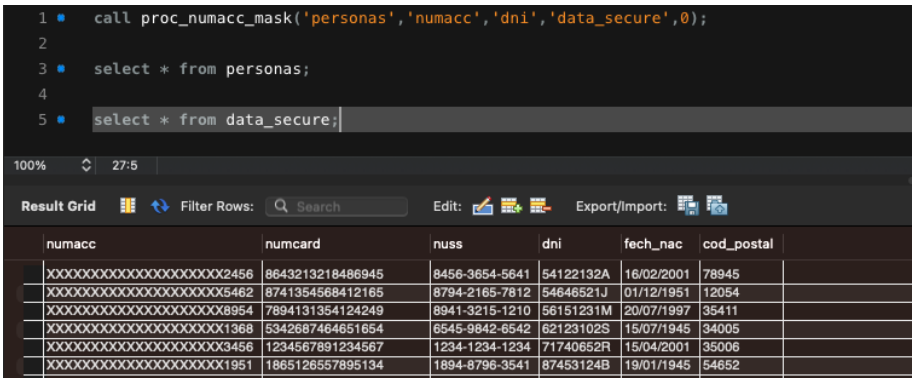


Figura 8.27: Ejecución correcta proc_numacc_mask()

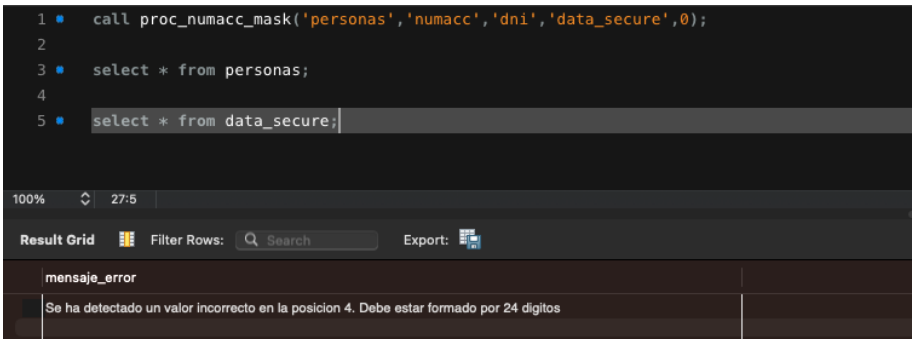


Figura 8.28: Ejecución errónea proc_numacc_mask()

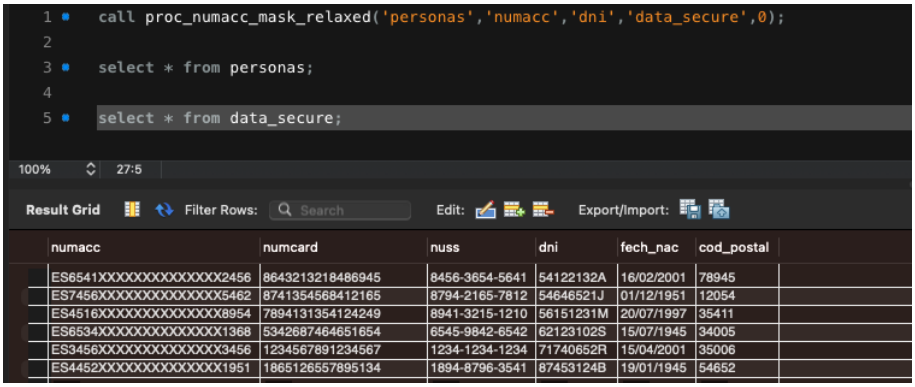


Figura 8.29: Ejecución correcta proc_numacc_mask_relaxed()

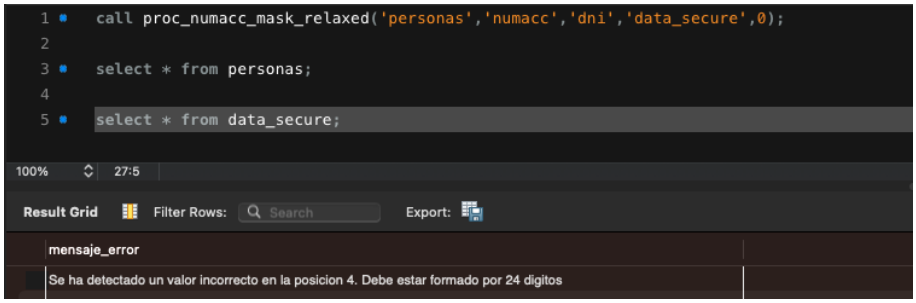


Figura 8.30: Ejecución errónea proc_numacc_mask_relaxed()

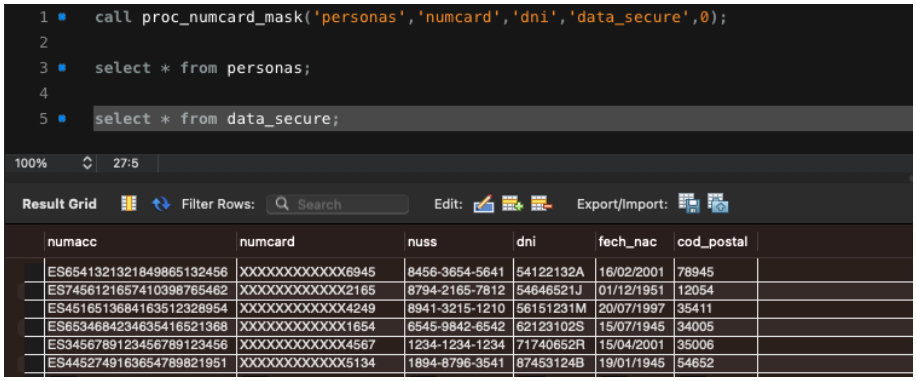
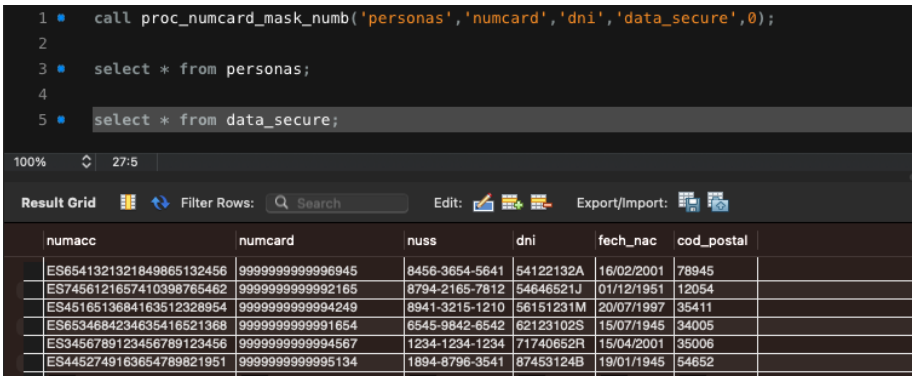


Figura 8.31: Ejecución correcta proc_numcard_mask()

Figura 8.32: Ejecución errónea proc_numcard_mask()



```
1  call proc_numcard_mask_numb('personas','numcard','dni','data_secure',0);
2
3  select * from personas;
4
5  select * from data_secure;
```

100% 27:5

Result Grid | Filter Rows: Search | Edit: | Export/Import:

numacc	numcard	nuss	dni	fech_nac	cod_postal
ES6541321321849865132456	9999999999996945	8456-3654-5641	54122132A	16/02/2001	78945
ES7456121657410398765462	9999999999992165	8794-2165-7812	54646521J	01/12/1951	12054
ES4516513684163512328954	9999999999994249	8941-3215-1210	56151231M	20/07/1997	35411
ES6534684234635416521368	9999999999991654	6545-9842-6542	62123102S	15/07/1945	34005
ES3456789123456789123456	9999999999994567	1234-1234-1234	71740652R	15/04/2001	35006
ES4452749163654789821951	9999999999995134	1894-8796-3541	87453124B	19/01/1945	54652

Figura 8.33: Ejecución correcta proc_numcard_mask_numb()

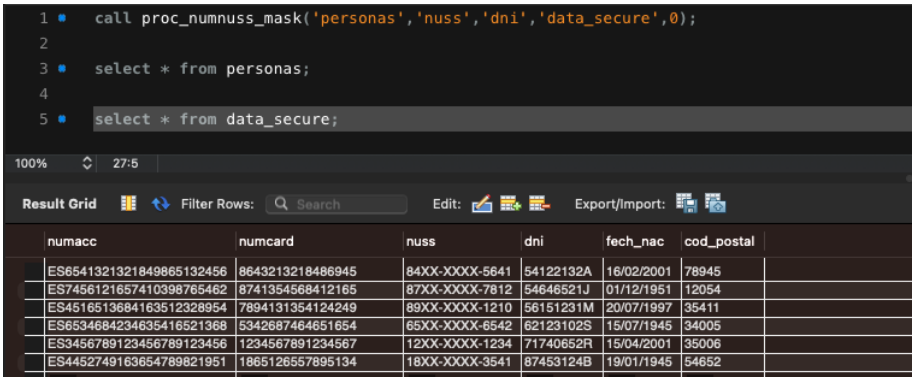
```
1  call proc_numcard_mask_numb('personas','numcard','dni','data_secure',0);
2
3  select * from personas;
4
5  select * from data_secure;
```

100% 27:5

Result Grid | Filter Rows: Search | Export:

mensaje_error
Se ha detectado un valor incorrecto en la posicion 5. Debe estar formado por 16 digitos

Figura 8.34: Ejecución errónea proc_numcard_mask_numb()



```
1  call proc_numnuss_mask('personas','nuss','dni','data_secure',0);
2
3  select * from personas;
4
5  select * from data_secure;
```

100% 27:5

Result Grid | Filter Rows: Search | Edit: | Export/Import:

numacc	numcard	nuss	dni	fech_nac	cod_postal
ES6541321321849865132456	8643213218486945	84XX-XXXX-5641	54122132A	16/02/2001	78945
ES7456121657410398765462	8741354568412165	87XX-XXXX-7812	54646521J	01/12/1951	12054
ES4516513684163512328954	7894131354124249	89XX-XXXX-1210	56151231M	20/07/1997	35411
ES6534684234635416521368	5342687464651654	65XX-XXXX-6542	62123102S	15/07/1945	34005
ES3456789123456789123456	1234567891234567	12XX-XXXX-1234	71740652R	15/04/2001	35006
ES4452749163654789821951	1865126557895134	18XX-XXXX-3541	87453124B	19/01/1945	54652

Figura 8.35: Ejecución correcta proc_numnuss_mask()

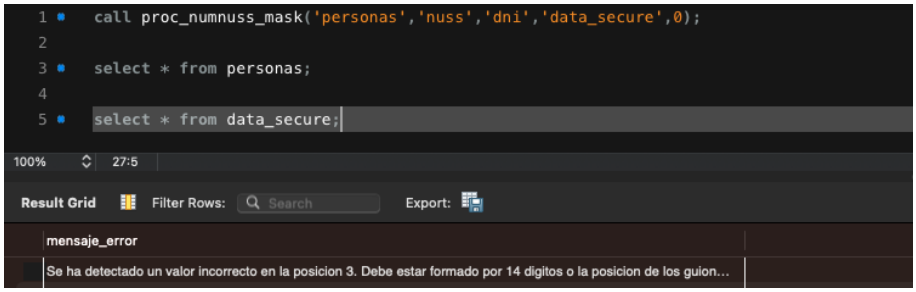
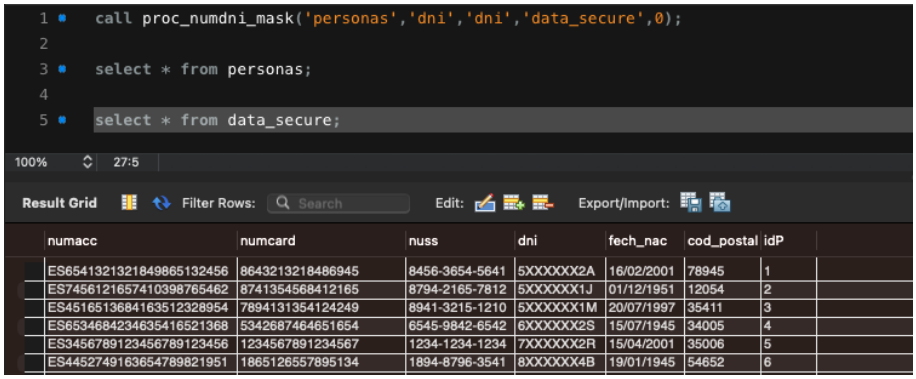


Figura 8.36: Ejecución errónea proc_numnuss_mask()



numacc	numcard	nuss	dni	fech_nac	cod_postal	idP
ES6541321321849865132456	8643213218486945	8456-3654-5641	5XXXXXX2A	16/02/2001	78945	1
ES7456121657410398765462	8741354568412165	8794-2165-7812	5XXXXXX1J	01/12/1951	12054	2
ES4516513684163512328954	7894131354124249	8941-3215-1210	5XXXXXX1M	20/07/1997	35411	3
ES6534684234635416521368	5342687464651654	6545-9842-6542	6XXXXXX2S	15/07/1945	34005	4
ES3456789123456789123456	1234567891234567	1234-1234-1234	7XXXXXX2R	15/04/2001	35006	5
ES4452749163654789821951	1865126557895134	1894-8796-3541	8XXXXXX4B	19/01/1945	54652	6

Figura 8.37: Ejecución correcta proc_numdni_mask()

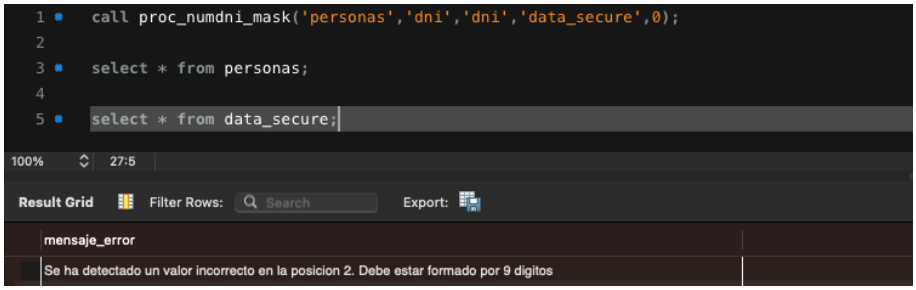


Figura 8.38: Ejecución errónea proc_numdni_mask()

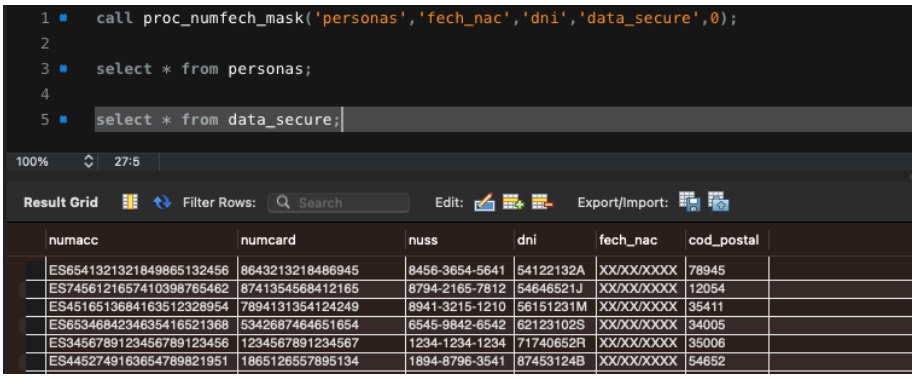


Figura 8.39: Ejecución correcta proc_numfech_mask()

Figura 8.40: Ejecución errónea proc_numfech_mask()

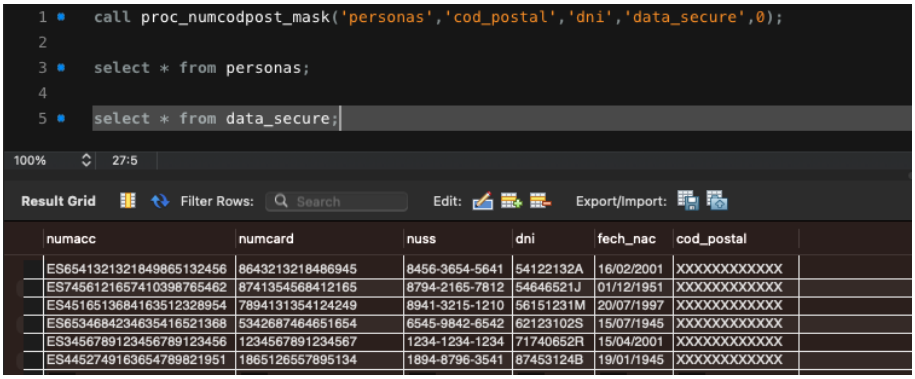


Figura 8.41: Ejecución correcta proc_numcodpost_mask()

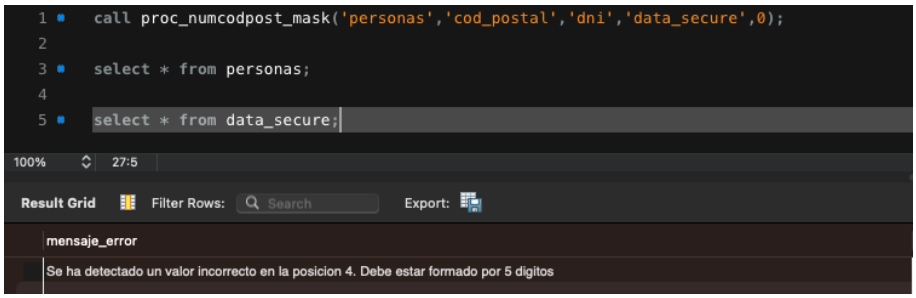


Figura 8.42: Ejecución errónea proc_numcodpost_mask()

Para este tipo de pruebas (aunque no se muestren todos los pasos) en los casos donde existe un dato erróneo, los pasos que se han seguido son:

1. Se ejecuta el procedimiento con los parámetros adecuados.

2. Nos avisa de que se ha producido un error. Existe algún dato en la tabla que no coincide con la estructura requerida y nos indica la posición.

3. Se lleva a cabo un UPDATE en la tabla para modificar el dato e insertarlo respetando la estructura.

4. Se vuelve a ejecutar el procedimiento, esta vez indicando en los parámetros la posición desde la que se quiere reanudar la ejecución del proceso de anonimización.

5. Finalmente, se anonimizan todos los valores de la columna seleccionada en la tabla.

EJEMPLO:

Suponemos como se observa en la Figura 8.43, que la tabla inicial tiene un dato de la columna “numacc” que no sigue la estructura, y el usuario no lo sabe. Se ejecuta el procedimiento con normalidad.

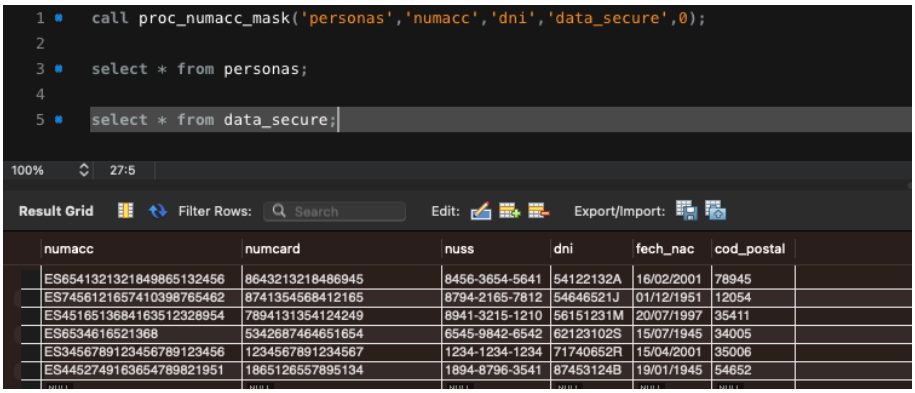


Figura 8.43: Ejecución inicial

Tras la ejecución, se nos avisa mediante un mensaje, que en la posición 4 de la tabla se encuentra un valor que no sigue la estructura adecuada.

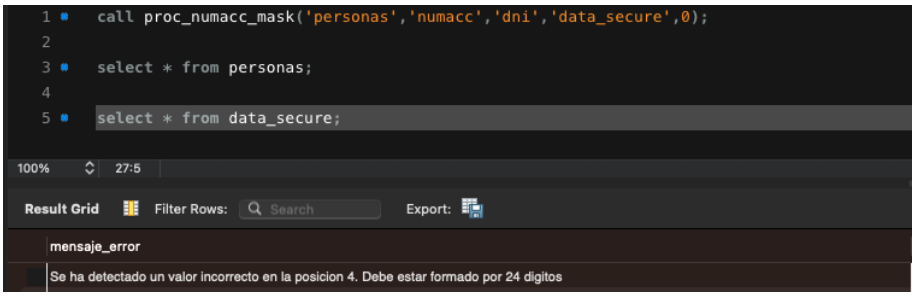


Figura 8.44: Aviso error

El usuario, mediante el UPDATE actualiza el dato en la tabla respetando la estructura adecuada.

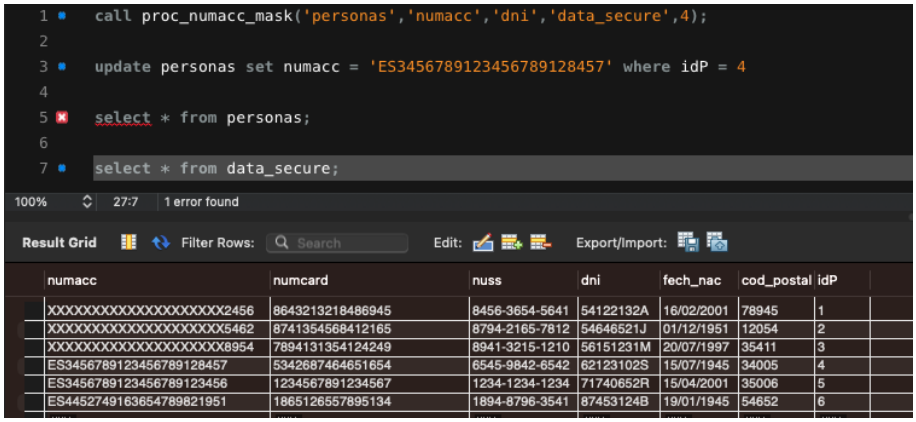


Figura 8.45: Actualización del dato

Finalmente, se vuelve a ejecutar el procedimiento. Esta vez hay que tener en cuenta actualizar el parámetro de posición a 4 para reanudar la anonimización desde esa fila de la tabla y cambiar el parámetro de clave primaria, pues tras la primera ejecución, esta ha cambiado a ser "idP".

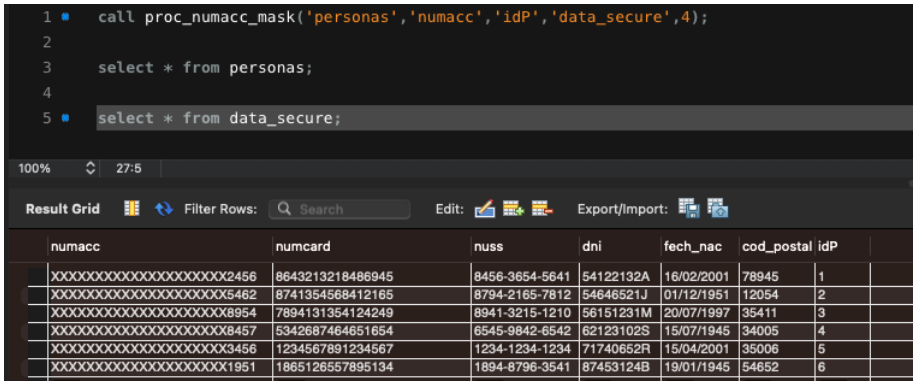


Figura 8.46: Reanudación de la anonimización

8.2.3. Prueba 5.

En esta validación, se comprueba que el procedimiento "randata_dictionary()" se ejecuta correctamente y el resultado es el esperado. Aunque no se muestra, previamente a la ejecución del procedimiento se ha cargado en una tabla los valores aleatorios contenidos en un fichero .txt

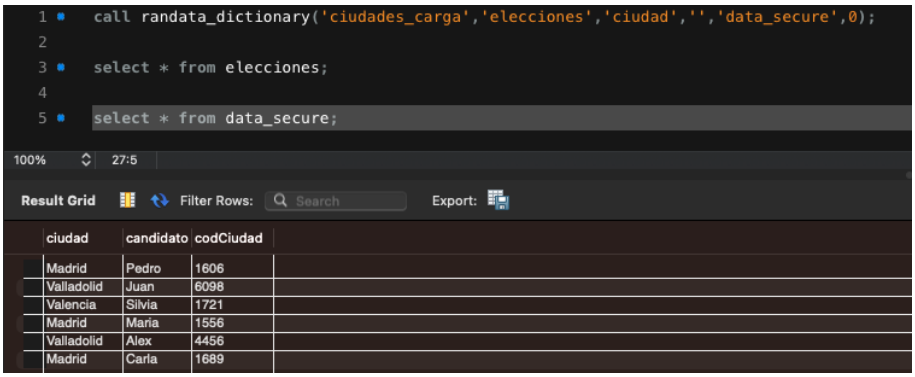


Figura 8.47: Ejecución del procedimiento y resultado

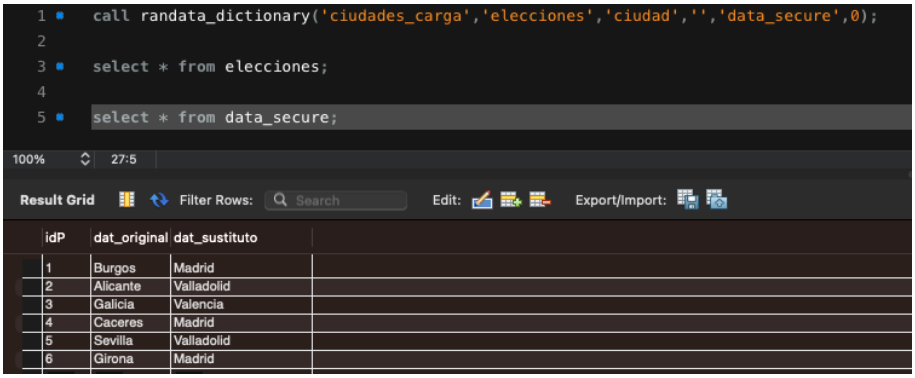


Figura 8.48: Tabla Segura

8.2.4. Prueba 6.

Para este conjunto de pruebas, los procedimientos se ejecutarán y se comprobará la obtención del resultado esperado y la correcta creación de las tablas seguras.

Para el primer procedimiento, se espera obtener una columna con valores generados a partir de la media de los valores originales.

Para el segundo procedimiento, se espera obtener una columna con valores HASH generados a partir de los valores originales.

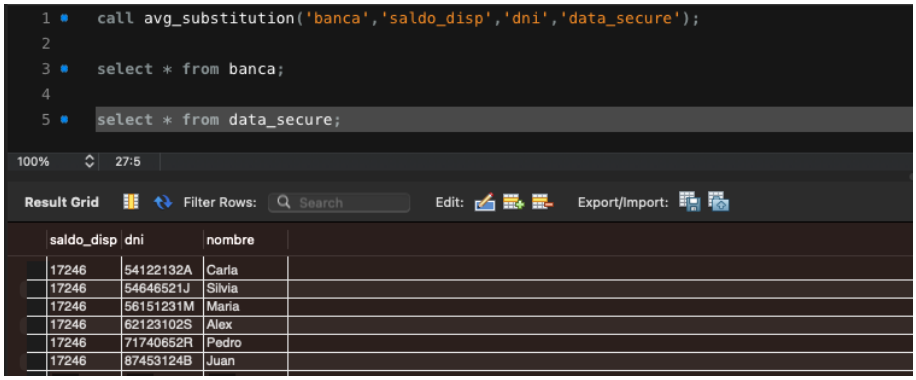


Figura 8.49: Ejecución del procedimiento y resultado

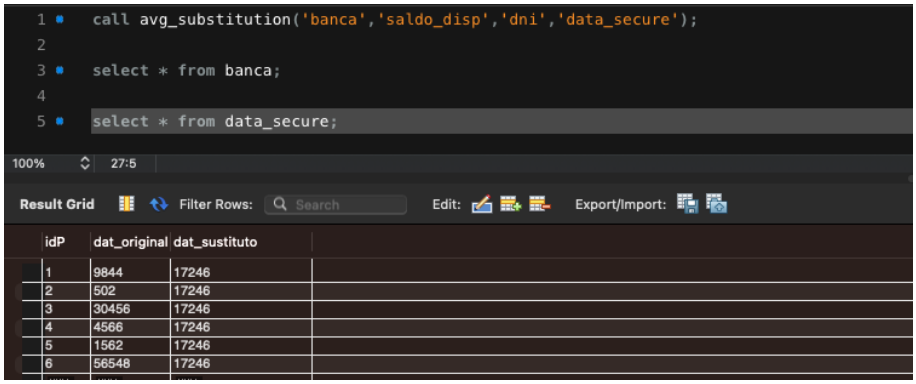


Figura 8.50: Tabla Segura

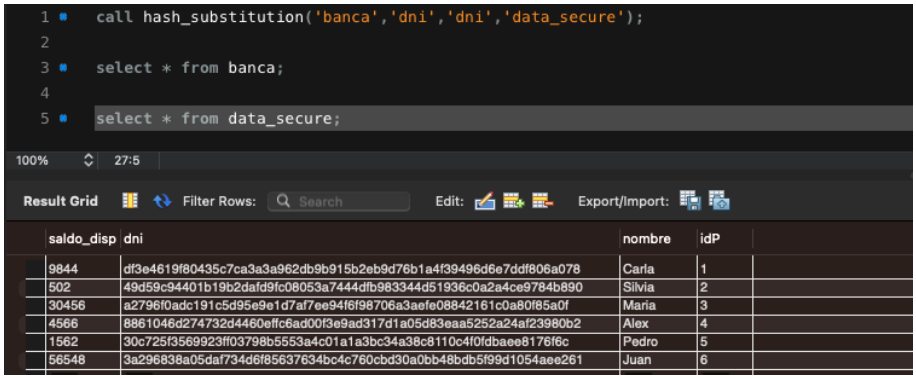


Figura 8.51: Ejecución del procedimiento y resultado

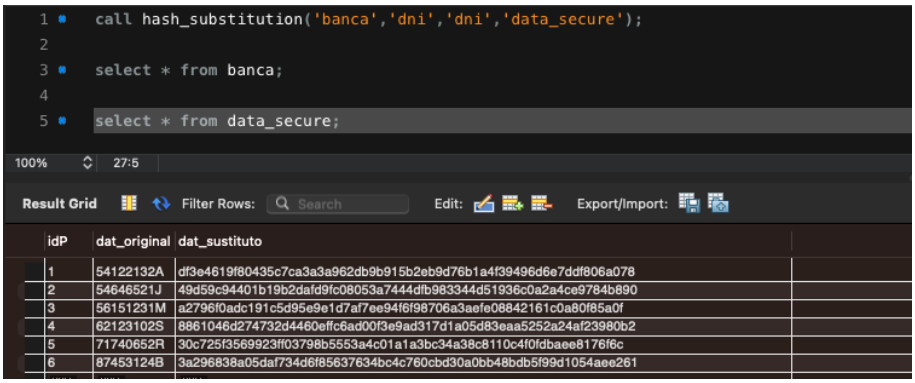


Figura 8.52: Tabla Segura

8.2.5. Prueba 7.

Para este conjunto de pruebas, los procedimientos se ejecutarán y se comprobará la obtención del resultado esperado y la recuperación de los datos originales.

Para el primer procedimiento, se espera obtener una columna con valores generados a partir de la multiplicación de los valores originales con otros aleatorios.

Para el segundo procedimiento, se espera obtener una columna con los valores originales previamente a ser anonimizados.

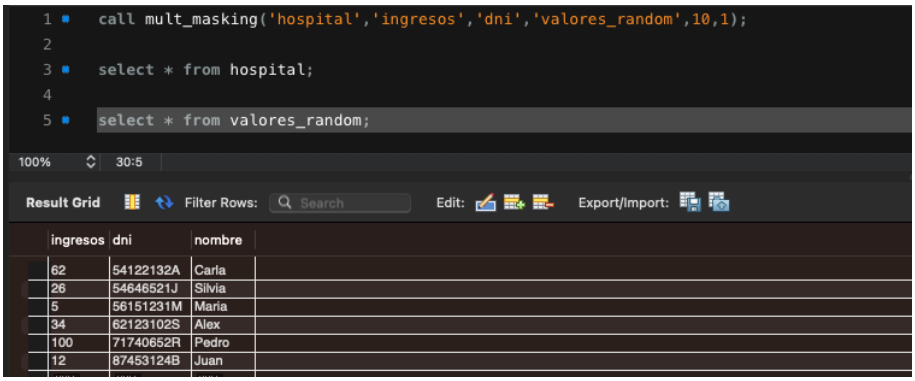


Figura 8.53: Estado de la tabla original

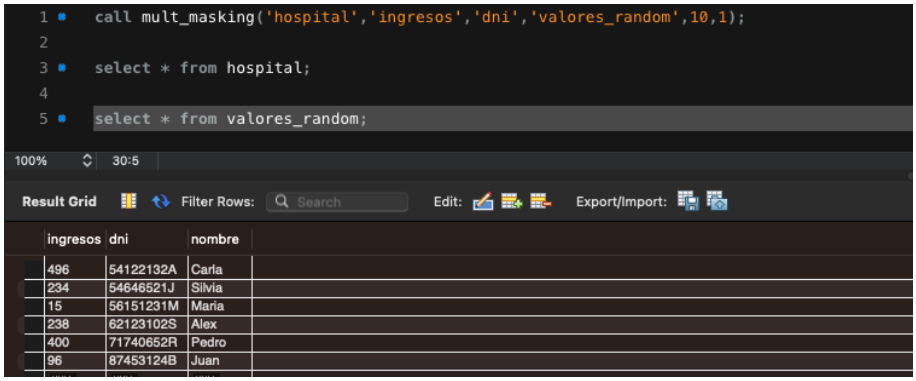


Figura 8.54: Ejecución del procedimiento y resultado

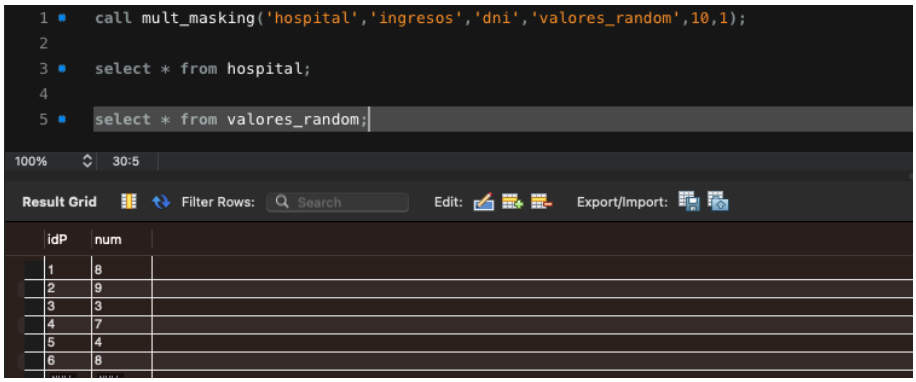


Figura 8.55: Tabla Valores Random

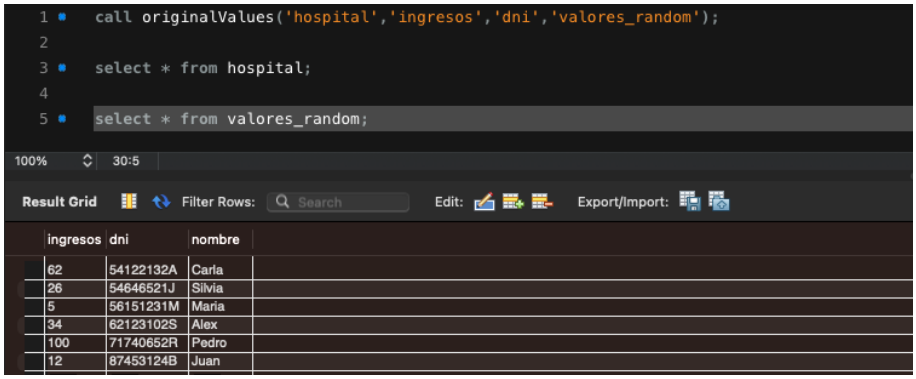


Figura 8.56: Recuperación de datos Originales

En todas las pruebas se ha realizado, aunque no se muestre, la correcta generación y población de las tablas seguras respectivamente para cada procedimiento.

Como resumen, se puede concluir que todas las herramientas funcionan según lo esperado y que los resultados obtenidos en cada una de las pruebas han sido los esperados.

Capítulo 9

Conclusiones

9.1. Conclusiones

Tras realizar este proyecto y llevar a cabo una breve recapitulación del mismo, se puede concluir que los objetivos principales se han conseguido con éxito. Se ha realizado un estudio conceptual sobre la anonimización, los procesos a seguir y se han comprendido cuáles son las técnicas más comunes de usar y como se emplean. Además, se ha realizado un análisis de las herramientas disponibles en el mercado que llevasen a cabo el proceso de anonimización de datos y emulando en concreto una de ellas, se ha conseguido implementar sus funciones para ponerlas a disposición de forma gratuita.

A pesar de haber cumplido con la gran mayoría de objetivos, uno que no se ha podido completar, es la recuperación de los datos originales sin ser guardados previamente en alguna otra parte. Sí se ha implementado una herramienta que lleva a cabo dicha función, pero de una forma un tanto restringida ya que solo es apta para números y no palabras (un posible mejora de futuro).

Dado que el Registro de Actividades de Tratamiento de Datos Personales (RAT) de la UVa [19] obliga a incluir una declaración del tratamiento de datos personales, en este proyecto no se han utilizado datos reales y todos los presentados son puramente ficticios. Sin embargo, las herramientas del toolkit diseñadas están diseñadas para poder ser usadas con datos reales.

La implementación de las herramientas, no solo ha ayudado a comprender aún más el funcionamiento de las técnicas de anonimización y sus objetivos, si no que además ha permitido aumentar los conocimientos sobre el lenguaje SQL. Durante los estudios universitarios

se habían adquirido las nociones más básicas, pero con este proyecto se han aprendido nuevas estructuras o uso de dicho lenguaje (ej, sentencias preparadas).

La planificación inicial se modificó un poco, ya que a pesar de que se ha intentado en todo momento ser lo más fiel posible a ella, han existido retrasos ajenos al proyecto como por ejemplo la realización de prácticas de empresa o asignaturas de la universidad que han generado un retraso en algunas de las tareas.

9.2. Trabajo Futuro

La idea principal y que se tuvo desde el primer momento sobre la creación de un conjunto de herramientas que permitieran la anonimización de datos, que fueran gratuitas y accesibles de una forma sencilla para los usuarios, se ha cumplido.

Sin embargo, existen posibilidades de mejora para este proyecto. Uno de los principales trabajos de futuro, sería la creación de una aplicación de escritorio que tuviera integradas las herramientas creadas en este proyecto y que permitiera conectarse a cualquier base de datos para aumentar sus funciones. Entre las principales consideraciones que habría que contemplar, además de su usabilidad o diseño, sería su portabilidad a distintos sistemas operativos y la garantía de privacidad y uso seguro de los datos.

Otra de las mejoras que se puede llevar a cabo es la ya comentada en el apartado anterior, con referencia a la posible recuperación de datos originales sea cual sea el tipo de dato, sin ser estos previamente guardados en algún lugar. Habría que implementar métodos más elaborados que permitieran recuperar por ejemplo palabras una vez sean estas anonimizadas.

Bibliografía

[1] Unión Europea. «Reglamento (UE) 2016/679 del Parlamento Europeo y del Consejo de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la Directiva 95/46/CE (Reglamento General de Protección de Datos)». En: (2016). URL: https://eurlex.europa.eu/legal-content/ES/TXT/?uri=CELEX%3A32016R0679

[2] Agencia Española de Protección de Datos (AEPD). «Introducción a la anonimización de datos: Técnicas y casos prácticos». En: *datos.gob.es* (2022). URL: https://datos.gob.es/es/documentacion/introduccion-la-anonimizacion-de-datos-tecnicas-y-casos-practicos

[3] *Microsoft Azure* «Calculadora de Precios». URL: https://azure.microsoft.com/es-mx/pricing/calculator/?cdn=disable

[4] *Visual Paradigm* «Visual Paradigm Pricing». URL: https://www.visual-paradigm.com/shop/vp.jsp?license=perpetual

[5] Glassdoor. «Sueldos para el puesto de Junior Software Developer en España». En: (2023). URL: https://www.glassdoor.es/Sueldos/junior-software-developer-sueldo-SRCH_KO0,25.htm#

[6] Agencia Española de Protección de Datos (AEPD). «La importancia de la anonimización y la privacidad de datos». En: *datos.gob.es* (2021). URL: https://datos.gob.es/es/blog/la-importancia-de-la-anonimizacion-y-la-privacidad-de-datos

[7] Ajay Kumar «Cómo las técnicas de ofuscación de datos pueden ayudar a proteger a las empresas». En: *ComputerWeekly.es* (2016). URL: https://www.computerweekly.com/es/consejo/Como-las-tecnicas-de-ofuscacion-de-datos-pueden-ayudar-a-proteger-a-las-empresas

[8] Real Acedemia Española «Definición "Anonimizar"». En: (2022). URL: https://dle.rae.es/anonimizar

[9] Secretaría de Estado de Digitalización e Inteligencia Artificial. «Introducción a la anonimización de datos: Técnicas y casos prácticos». En: *datos.gob.es* (2022). URL: https://datos.gob.es/es/documentacion/introduccion-la-anonimizacion-de-datos-tecnicas-y-casos-practicos

[10] RGPD UE. «Protección de datos desde el diseño y por defectos». En: *privacy-regulation.eu* (2016). URL: https://www.privacy-regulation.eu/es/25.htm

[11] Agencia Española de Protección de Datos (AEPD). «INTRODUCCIÓN A LA ANONIMIZACIÓN DE DATOS». En: *datos.gob.es* (2022). URL: https://datos.gob.es/sites/default/files/doc/file/informe-anonimizacion-es.pdf

[12] Agencia Española de Protección de Datos (AEPD). «Orientaciones y garantias en los procedimientos de anonimización de datos personale». En: *datos.gob.es* (2019). URL: https://www.aepd.es/sites/default/files/2019-09/guia-orientaciones-procedimientos-anonimizacion.pdf

[13] «Data Masking: 8 Techniques and How to Implement Them Successfully». En: *satoricyber.com* (2021). URL: https://satoricyber.com/data-masking/data-masking-8-techniques-and-how-to-implement-them-successfully/

[14] «Más que anonimizar (Explicando Palantir, #3)». En: *palantir.com* (2022). URL: https://blog.palantir.com/ms-que-anonimizar-explicando-palantir-3-1621c0fb98fd

[15] MySQL. «MySQL Enterprise Masking and De-identification». En: *mysql.com* (). URL: https://www.mysql.com/products/enterprise/masking.html

[16] Amnesia Anonymization Tool. URL: https://amnesia.openaire.eu/index.html

[17] Autoridad Nacional de Protección de Datos de Singapur. «Guía básica de anonimización». En: *aepd.es* (2022). URL: https://www.aepd.es/es/documento/guia-basica-anonimizacion.pdfs

[18] «IEEE Standard Glossary of Software Engineering Terminology». En: *IEEE Std 610.12-1990 (1990), págs. 1-84. doi:* URL: https://ieeexplore.ieee.org/document/159342

[19] David Sanz Esteban. Registro de actividades de tratamiento. Universidad de Valladolid. 2021. URL: https://secretariageneral.uva.es/competencias/proteccion-de-datos/registro-actividades-tratamiento/

[20] MySQL. URL: https://www.mysql.com

[21] Visual Paradigm. URL: https://www.visual-paradigm.com

[22] Microsoft Project. URL: https://www.microsoft.com/es-es/microsoft-365/project/project-management-software

[23] Overleaf. URL: https://www.overleaf.com

[24] MySQL. «Security-Related mysqld Options and Variables». En: *mysql.com* (). URL: https://dev.mysql.com/doc/refman/5.7/en/security-options.html

[25] MySQL. «String Functions and Operators». En: *mysql.com* (). URL: https://dev.mysql.com/doc/refman/8.0/en/string-functions.html

[26] Oracle. «Formatos de enmascaramiento predefinidos». En: *docs.oreacle.com* (). URL: https://docs.oracle.com/es-ww/iaas/data-safe/doc/predefined-masking-formats.html

[27] Oracle. «Unidad 12. Triggers, procedimientos y funciones en MySQL» En: *IES Celia Viñas (Almería)*. En: *josejuansanchez.org* (2022/2023). URL: https://josejuansanchez.org/bd/unidad-12-teoria/index.html

[28] Agencia Española de Protección de Datos (AEPD). «ORIENTACIÓN PARA LA APLICACIÓN PROVISIONAL DE LA DISPOSICIÓN ADICIONAL SÉPTIMA DE LA LOPDGDD». En: *aepd.es* (). URL: https://www.aepd.es/es/documento/orientaciones-da7_0.pdf

Apéndice A

Manual de usuario

Este manual de usuario tratará de guiarle mediante un ejemplo, los pasos a seguir para la importación y uso de las herramientas del toolkit creadas en este proyecto. A continuación, se detallarán todas las funciones y procedimientos disponibles en el toolkit para conocer cuál es su objetivo y se mostrará un ejemplo de su funcionamiento para una comprensión más sencilla.

A.0.1. Importación de las herramientas

En primer lugar, como se muestra en la Figura A.1, se importan los ficheros de Funciones.sql y Procedimientos.sql al servidor de base de datos. La ruta destino, como se ha comentado en el apartado 7.2 debe ser elegida con criterio por el usuario.

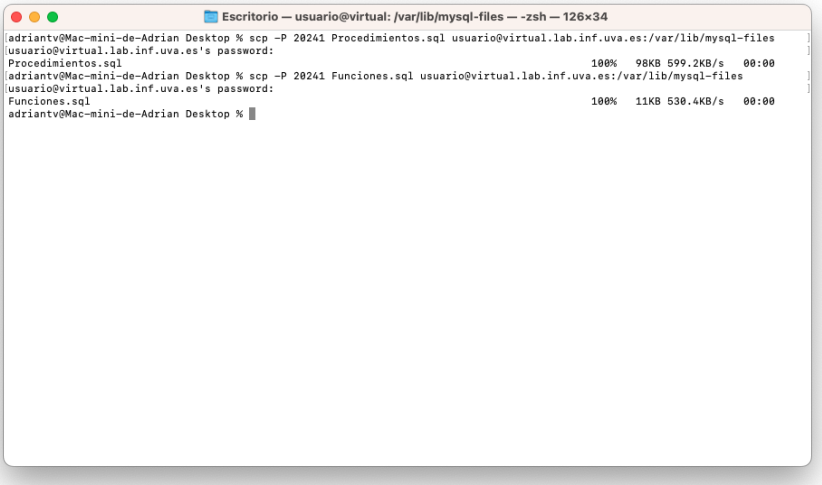


Figura A.1: Carga de ficheros .sql

Después, mediante SSH, se accede al servidor y posteriormente a la base de datos. Previamente a la importación de los ficheros a la base de datos, debe existir alguna ya creada, en este ejemplo, es "TFGadri".

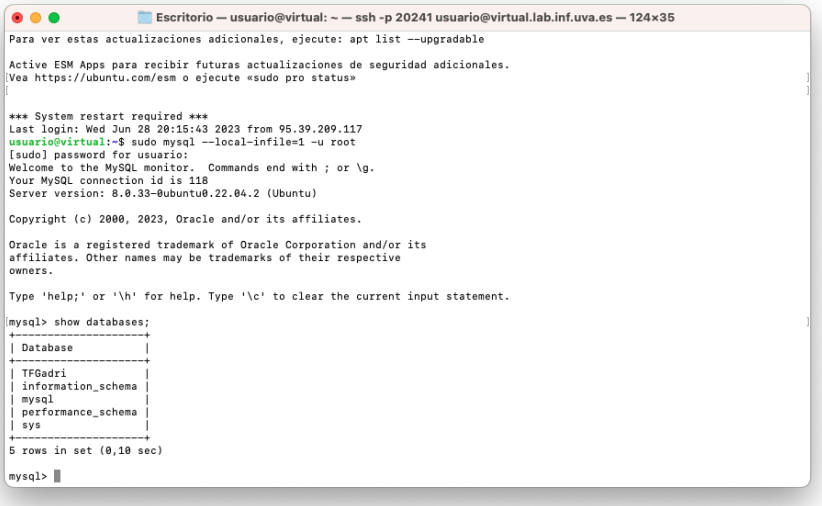


Figura A.2: Bases de Datos

Tras tener creada la base de datos, accedemos en el servidor a la carpeta donde se habían importado los ficheros de las herramientas y procedemos a cargarlas en la base de datos:

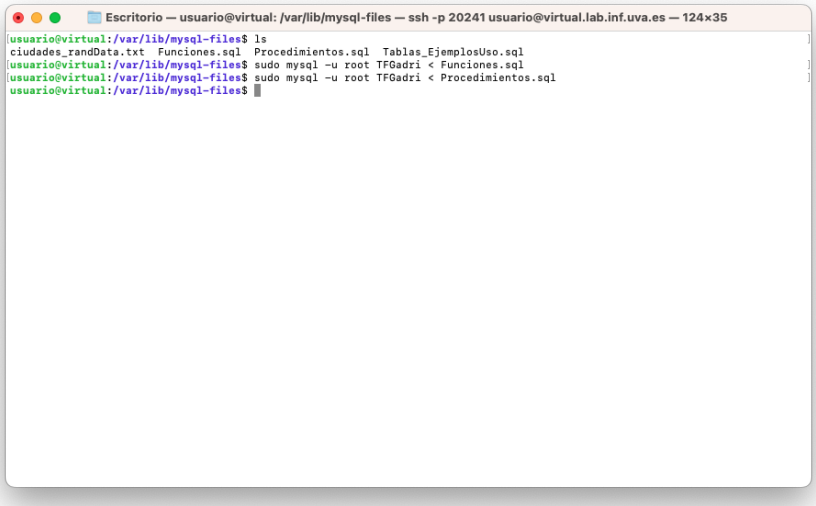


Figura A.3: Importación de herramientas

De nuevo, en la base de datos comprobamos que se han cargado todas las funciones y procedimientos que servirán como herramientas. Finalmente, ya podemos usarlas según la necesidad que se tenga.

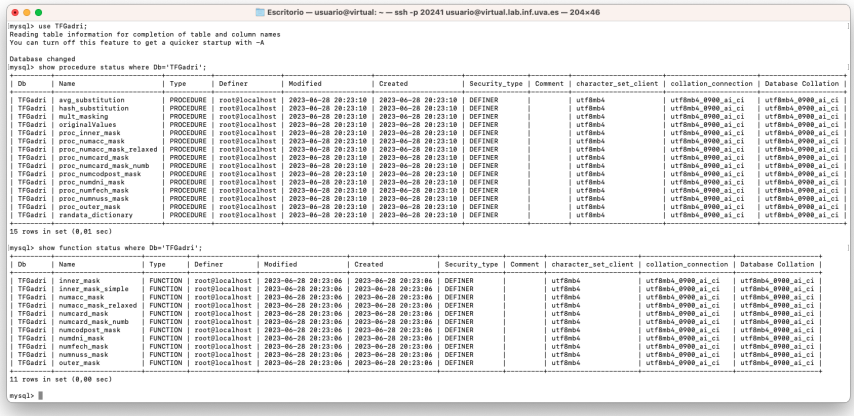


Figura A.4: Procedimientos y Funciones

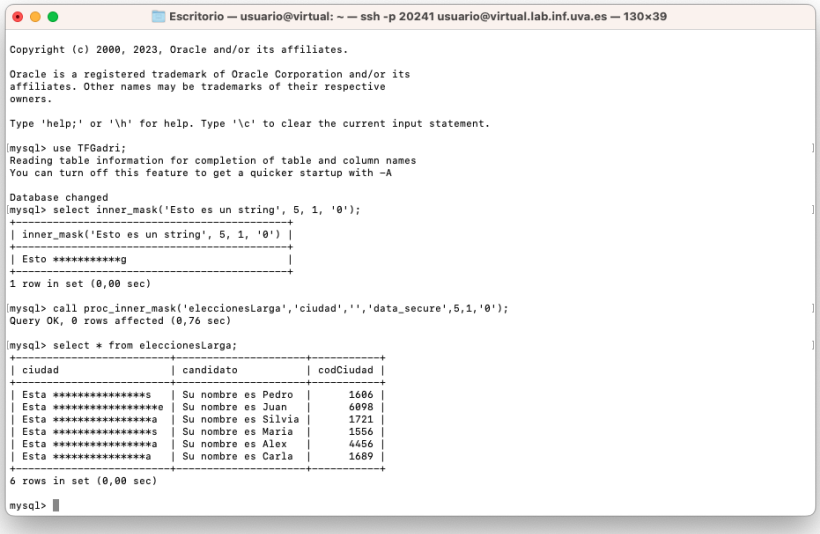


Figura A.5: Ejemplos de uso

[**CASOS ESPECIALES:**]

∗ Como se ha comentado en el apartado 7.2, en caso de obtener algún error al cargar las funciones a la base de datos, previo a este paso, se deberá activar en la base de datos el parámetro "log_bin_trust_function_creators".

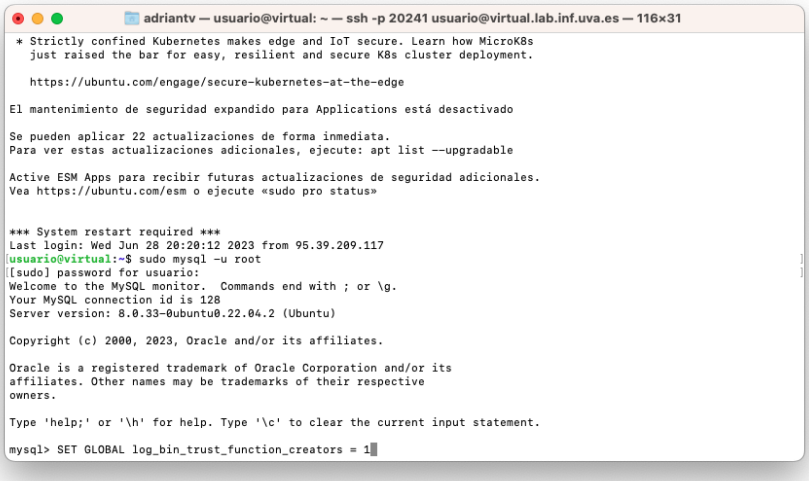


Figura A.6: Activación de variable "log_bin_trust_function_creators"

∗ Para la carga de datos desde un fichero .txt alojado en el servidor a una tabla en la base de datos (previamente creada) se debe usar el comando LOAD DATA LOCAL INFILE como se muestra en la figura A.7.
Para conseguir que este comando funcione, es necesario activar en el servidor la variable "–local-infile" mediante el comando "sudo mysql –local-infile=1 -u root" como se muestra en la figura A.2):

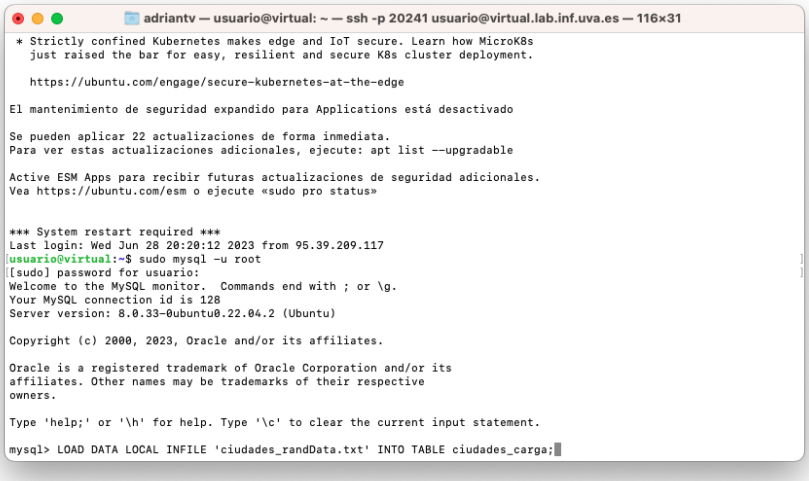


Figura A.7: Importación datos externos a base de datos

A.0.2. Herramientas disponibles en el toolkit

Las funciones y procedimientos se dividen en diferentes apartados abarcando cada uno de ellos unos objetivos diferentes.

- **FUNCIONES**

1. ENMASCARAMIENTO DE DATOS PARA ELIMINAR CARACTERISTICAS IDENTIFICATIVAS

   Mediante la técnica de sustitución de caracteres por un carácter especial, se consigue anonimizar ciertas partes de un dato sensible que lo hacen característico generando la imposibilidad de ser identificado.

   1.1 DE PROPOSITO GENERAL

   → inner_mask(): Enmascara el interior de un string pasado como argumento, dejando los extremos sin ocultar. Otros argumentos especifican la longitud de los extremos que se desean no enmascarar y el caracter que se desea

usar para ocultar la informacion.

> Un '0' como caracter supondra el uso por defecto.

SYNTAX:
inner_mask('expresion', long_inicial, long_final, 'caracter_ofusc')

```
mysql> SELECT inner_mask ('Esto es un string', 5, 1, '0');
+-----------------------------------------------------+
|  inner_mask('Esto es un string', 5, 1, '0')  |
+-----------------------------------------------------+
|  Esto **********g                                   |
+-----------------------------------------------------+
mysql> SELECT inner_mask ('Esto es un string', 1, 5, 'X');
+-----------------------------------------------------+
|  inner_mask('Esto es un string', 1, 5, 'X')  |
+-----------------------------------------------------+
|  EXXXXXXXXXXtring                                   |
+-----------------------------------------------------+
```

Figura A.8: Funcion inner_mask()

→ inner_mask_simple(): Se trata de la misma función descrita antes pero como se ha descrito en el apartado 7.3, implementada con un código que reduce el consumo de recursos.

→ outer_mask(): Ejecuta la funcion inversa, enmascara los extremos de un string pasado como argumento, dejando visible su interior. Otros argumentos especifican la longitud de los extremos que se desean enmascarar y el caracter que se desea usar para ocultar la informacion.

> Un '0' como caracter supondra el uso por defecto.

SYNTAX:
outer_mask('expresion', long_inicial, long_final, 'caracter_ofusc')

```
mysql> SELECT outer_mask ('Esto es un string', 5, 1, '0');
+-----------------------------------------------------+
|  outer_mask('Esto es un string', 5, 1, '0')  |
+-----------------------------------------------------+
|  *****es un strin*                                  |
+-----------------------------------------------------+
mysql> SELECT outer_mask ('Esto es un string', 1, 5, 'X');
+-----------------------------------------------------+
|  outer_mask('Esto es un string', 1, 5, 'X')  |
+-----------------------------------------------------+
|  Xsto es un sXXXXX                                  |
+-----------------------------------------------------+
```

Figura A.9: Funcion outer_mask()

1.2 DE PROPOSITO ESPECIFICO

1.2.1 Enmascaramiento de números de cuentas bancarias o tarjetas de crédito

→ numacc_mask(): Enmascara todos los digitos iniciales excepto los ultimos 4 del numero de cuenta bancaria introducida. El argumento tiene como restricciones ser un valor de tipo VARCHAR de 24 digitos.

SYNTAX:
numacc_mask('expresion')

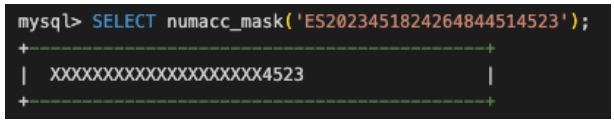


Figura A.10: Funcion numacc_mask()

→ numacc_mask_relaxed(): Similar pero no enmascara los 6 primeros digitos que indican: el IBAN (dos primeros digitos el pais de procedencia de la cuenta y dos de control) y seguido el codigo de la entidad financiera.

SYNTAX:
numacc_mask_relaxed('expresion')

Figura A.11: Funcion numaa_mask_relaxed()

→ numcard_mask(): Enmascara todos los digitos iniciales excepto los ultimos 4 del numero de tarjeta de credito introducido. El argumento tiene como restricciones ser un valorde tipo VARCHAR de 16 digitos.

SYNTAX:
numcard_mask('expresion')

Figura A.12: Funcion numcard_mask()

→ numcard_mask_numb(): En esta variante, la entrada y salida se trata de un numero entero y por lo tanto, se sustituye un carácter de enmascaramiento por un número, en este caso el 9.

SYNTAX:
numcard_mask_numb(numero)

```
mysql> SELECT numcard_mask_numb(1589478635214569);
+----------------------------------------+
|  9999999999994569                      |
+----------------------------------------+
```

Figura A.13: Funcion numcard_mask_numb()

1.2.2 Enmascaramiento del Número de Seguridad Social (NUSS)

→ numnuss_mask(): Enmascara todos excepto los dos primeros digitos (codigo de la provincia) y ultimos cuatro digitos del numero (dos pertenecientes al numero en si y dos que son de control). El argumento tiene como restricciones ser un valor de tipo VARCHAR de 12 digitos separados por guion de la forma: XXXX-XXXX-XXXX.

SYNTAX:
numnuss_mask('expresion')

```
mysql> SELECT numnuss_mask('8023-5978-1569');
+----------------------------------------+
|  80XX-XXXX-1569                        |
+----------------------------------------+
```

Figura A.14: Funcion numnuss_mask()

1.2.3 Enmascaramiento del Documento Nacional de Identidad (DNI)

→ numdni_mask(): Enmascara los tres primeros y dos ultimos digitos del DNI. El argumento tiene como restricciones ser un valor alfanumerico (tipo VARCHAR) de 9 digitos.

SYNTAX:
numdni_mask('expresion')

```
mysql> SELECT numdni_mask('71980657V');
+-----------------------------------------------------+
|  XXX8065XX                                          |
+-----------------------------------------------------+
```

Figura A.15: Funcion numdni_mask()

1.2.4 Enmascaramiento de datos personales como fecha de nacimiento o código postal

→ numfech_mask(): Enmascara los digitos de la fecha de nacimiento de una persona. El argumento tiene como restricciones ser un valor numerico de 8 digitos separados por una barra de la forma: XX/XX/XXXX.

SYNTAX:
numfech_mask('expresion')

```
mysql> SELECT numfech_mask('12/02/2001');
+-----------------------------------------------------+
|  XX/XX/XXXX                                         |
+-----------------------------------------------------+
```

Figura A.16: Funcion numfech_mask()

→ numcodpost_mask(): Enmascara los digitos (excepto el ultimo) del codigo postal de residencia de una persona. El argumento ¡ene como restricciones ser un valor numerico de 5 digitos.

SYNTAX:
numcodpost_mask('expresion')

```
mysql> SELECT numcodpost_mask(34006);
+-----------------------------------------------------+
|  XXXX6                                              |
+-----------------------------------------------------+
```

Figura A.17: Funcion numcodpost_mask()

- **PROCEDIMIENTOS**

1. ENMASCARAMIENTO DE DATOS PARA ELIMINAR CARACTERISTICAS

IDENTIFICATIVAS

Mediante la técnica de sustitución de caracteres por un carácter especial, se consigue anonimizar ciertas partes de un dato sensible que lo hacen característico generando la imposibilidad de ser identificado.

(Para este apartado no se mostrarán ejemplos ya que se trata de los mismos resultados que para las funciones, con la diferencia de que se aplica a todos los valores de una columna de una tabla de base de datos.)

1.1 DE PROPOSITO GENERAL

→ proc_inner_mask(): Enmascara el interior de un string pasado como argumento, dejando los extremos sin ocultar. Otros argumentos especifican la longitud de los extremos que se desean no enmascarar y el caracter que se desea usar para ocultar la informacion.

> Un '0' como caracter supondra el uso por defecto.

SYNTAX:
proc_inner_mask('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', long_inicial, long_final, 'caracter_ofusc')

→ proc_outer_mask(): Ejecuta la funcion inversa, enmascara los extremos de un string pasado como argumento, dejando visible su interior. Otros argumentos especifican la longitud de los extremos que se desean enmascarar y el caracter que se desea usar para ocultar la informacion.

> Un '0' como caracter supondra el uso por defecto.

SYNTAX:
proc_outer_mask('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', long_inicial, long_final, 'caracter_ofusc')

1.2 DE PROPOSITO ESPECIFICO

1.2.1 Enmascaramiento de números de cuentas bancarias o tarjetas de crédito

→ proc_numacc_mask(): Enmascara todos los digitos iniciales excepto los ultimos 4 del numero de cuenta bancaria introducida. El argumento tiene como restricciones ser un valor de tipo VARCHAR de 24 digitos.

ERROR:
En el caso de encontrase en la tabla un valor que no cumple las condi-

ciones, se da la posibilidad al administrador de actualizar la tabla con un valor correcto y continuar con el enmascaramiento desde la posicion donde se detuvo. Para ello se debe indicar en el parametro “index_inic” la posicion a continuar o 0 en caso inicial.

SYNTAX:
proc_numacc_mask('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', index_inic)

→ proc_numacc_mask_relaxed(): Similar pero no enmascara los 6 primeros digitos que indican: el IBAN (dos primeros digitos el pais de procedencia de la cuenta y dos de control) y seguido el codigo de la entidad financiera.

ERROR:
En el caso de encontrase en la tabla un valor que no cumple las condiciones, se da la posibilidad al administrador de actualizar la tabla con un valor correcto y continuar con el enmascaramiento desde la posicion donde se detuvo. Para ello se debe indicar en el parametro “index_inic” la posicion a continuar o 0 en caso inicial.

SYNTAX:
proc_numacc_mask_relaxed('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', index_inic)

→ proc_numcard_mask(): Enmascara todos los digitos iniciales excepto los ultimos 4 del numero de tarjeta de credito introducido. El argumento tiene como restricciones ser un valorde tipo VARCHAR de 16 digitos.

ERROR:
En el caso de encontrase en la tabla un valor que no cumple las condiciones, se da la posibilidad al administrador de actualizar la tabla con un valor correcto y continuar con el enmascaramiento desde la posicion donde se detuvo. Para ello se debe indicar en el parametro “index_inic” la posicion a continuar o 0 en caso inicial.

SYNTAX:
proc_numcard_mask('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', index_inic)

→ proc_numcard_mask_numb(): En esta variante, la entrada y salida se trata de un numero entero y por lo tanto, se sustituye un caracter de enmascaramiento por un numero, en este caso el 9.

ERROR:
En el caso de encontrase en la tabla un valor que no cumple las condiciones, se da la posibilidad al administrador de actualizar la tabla con

un valor correcto y continuar con el enmascaramiento desde la posicion donde se detuvo. Para ello se debe indicar en el parametro "index_inic" la posicion a continuar o 0 en caso inicial.

SYNTAX:
proc_numcard_mask_numb('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', index_inic)

1.2.2 Enmascaramiento del Número de Seguridad Social (NUSS)

→ proc_numnuss_mask(): Enmascara todos excepto los dos primeros digitos (codigo de la provincia) y ultimos cuatro digitos del numero (dos pertenecientes al numero en si y dos que son de control). El argumento tiene como restricciones ser un valor de tipo VARCHAR de 12 digitos separados por guion de la forma: XXXX-XXXX-XXXX.

ERROR:
En el caso de encontrase en la tabla un valor que no cumple las condiciones, se da la posibilidad al administrador de actualizar la tabla con un valor correcto y continuar con el enmascaramiento desde la posicion donde se detuvo. Para ello se debe indicar en el parametro "index_inic" la posicion a continuar o 0 en caso inicial.

SYNTAX:
proc_numnuss_mask('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', index_inic)

1.2.3 Enmascaramiento del Documento Nacional de Identidad (DNI)

→ proc_numdni_mask(): Enmascara los tres primeros y dos ultimos digitos del DNI. El argumento tiene como restricciones ser un valor alfanumerico (tipo VARCHAR) de 9 digitos.

ERROR:
En el caso de encontrase en la tabla un valor que no cumple las condiciones, se da la posibilidad al administrador de actualizar la tabla con un valor correcto y continuar con el enmascaramiento desde la posicion donde se detuvo. Para ello se debe indicar en el parametro "index_inic" la posicion a continuar o 0 en caso inicial.

SYNTAX:
proc_numdni_mask('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', index_inic)

1.2.4 Enmascaramiento de datos personales como fecha de nacimiento o código postal

→ proc_numfech_mask(): Enmascara los digitos de la fecha de nacimiento de una persona. El argumento tiene como restricciones ser un valor numerico de 8 digitos separados por una barra de la forma: XX/XX/XXXX.

ERROR:
En el caso de encontrase en la tabla un valor que no cumple las condiciones, se da la posibilidad al administrador de actualizar la tabla con un valor correcto y continuar con el enmascaramiento desde la posicion donde se detuvo. Para ello se debe indicar en el parametro "index_inic" la posicion a continuar o 0 en caso inicial.

SYNTAX:
proc_numfech_mask('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', index_inic)

→ proc_numcodpost_mask(): Enmascara los digitos (excepto el ultimo) del codigo postal de residencia de una persona. El argumento tiene como restricciones ser un valor numerico de 5 digitos.

ERROR:
En el caso de encontrase en la tabla un valor que no cumple las condiciones, se da la posibilidad al administrador de actualizar la tabla con un valor correcto y continuar con el enmascaramiento desde la posicion donde se detuvo. Para ello se debe indicar en el parametro "index_inic" la posicion a continuar o 0 en caso inicial.

SYNTAX:
proc_numcodpost_mask('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', index_inic)

2. GENERANDO DATOS ALEATORIOS USANDO DICCIONARIOS

Mediante la técnica de sustitución de datos originales por valores sintéticos aleatorios, en este caso valores localizados en un diccionario de datos, se consigue anonimizar un conjunto de estos haciendo imposible su identificación.

→ randata_dictionary(): Este procedimiento sustituira los valores de la columna de una tabla especificada como argumento, por valores ubicados en un diccionario previamente cargado en el servidor por el usuario.

Ademas, se debera pasar como argumento el nombre de una tabla externa donde se guardaran los pares valor original - valor sustituido para mantener su conocimiento (dicha tabla debe posteriormente ser securizada adecuadamente por el administrador) y el tipo, diferenciando si guardara valores de tipo INT o VARCHAR.

SYNTAX:
randata_dictionary('tabla_valores_importados', 'tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura', tipo_tabla_segura)

Para su uso seguir los siguientes pasos:

- PASO 1: Carga del fichero creado a la base de datos. Opciones:

  - OP1 (Uso del servidor): Cargar el fichero en el servidor y colocarlo en la ruta /var/lib/mysql-files (asumiendo que la variable de sistema secure_file_priv esta activada) u otra ruta considerada por el usuario.

  - OP2 (Uso MySQL Workbench): Cargarlo desde la base de datos directamente a una tabla mediante el comando:
  > LOAD DATA INFILE 'ruta_absoluta_al_fichero' INTO TABLE tabla_input;

  (Para OP2 debe estar creada la tabla_input como se muestra en el PASO 2)

- PASO 2: Creacion de la "tabla_input" para guardar los valores del fichero cargado.

  > create table ciudades_carga (ciudades varchar(50));

  A continuacion, cargar los valores a la tabla.

  **
  Al cargar con la opcion LOCAL, la base de datos puede dar un aviso de que dicha variable esta desactivada, seguir estos pasos:

  Acceder a la base de datos
  > SET GLOBAL local_infile=1;
  > quit;
  bye

  $ mysql –local-infile=1 -u root -p

  Acceder a la base de datos
  > use nombre_bbdd;
  **

  > LOAD DATA LOCAL INFILE 'nombre_fichero' INTO TABLE ciudades_carga;

- PASO 3: Llamar al procedimiento para su ejecucion

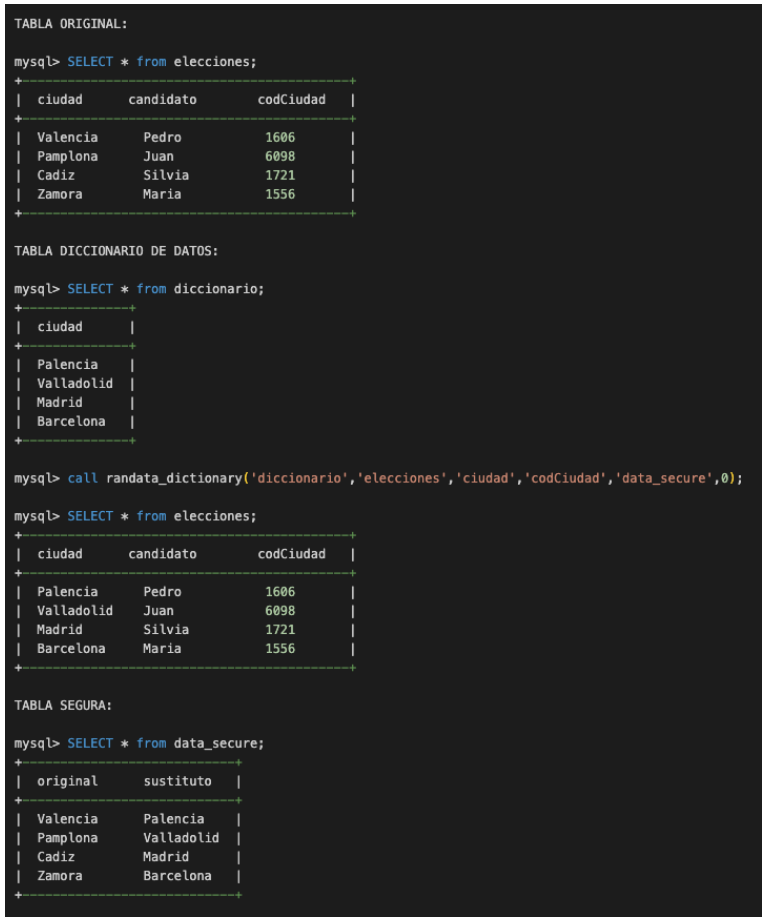


Figura A.18: Procedimiento randata_dictionary()

3. USO DE TECNICAS TIPICAS EN LA ANONIMIZACION

Mediante la técnica de sustitución de datos por valores calculados a través de operaciones matemáticas como la media o el uso de métodos criptográficos como el hashing, conseguimos la anonimización de datos y su imposibilidad de identificación.

3.1 Enmascaramiento mediante uso de valores promedio

→ avg_substitution(): Este procedimiento calculara el valor promedio de una columna de una tabla especificada como argumento. Tras esta operacion, sustituira los valores de dicha columna por el valor promedio (generando

a parte una nueva tabla en la que se guardaran los valores originales junto con el valor promedio). Este procedimiento tiene como restriccion que los datos de la columna especificada deben ser valores numericos enteros.

SYNTAX:
avg_substitution('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura')

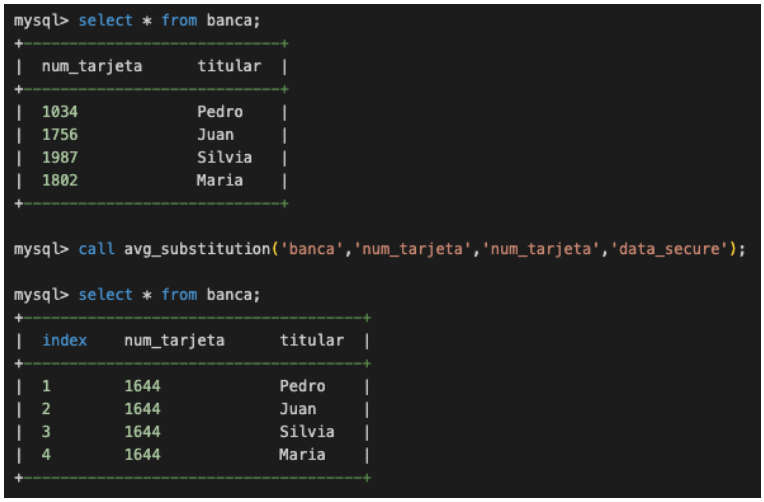


Figura A.19: Procedimiento avg_substitution()

3.2 Enmascaramiento mediante hashing

→ hash_substitution(): Este procedimiento sustituira cada valor de una columna de una tabla especificada en el argumento por un valor calculado mediante el algoritmo de hash SHA-256.

[Generara a parte una nueva tabla en la que se guardaran los valores originales junto con el valor hash].

SYNTAX:
hash_substitution('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_segura')

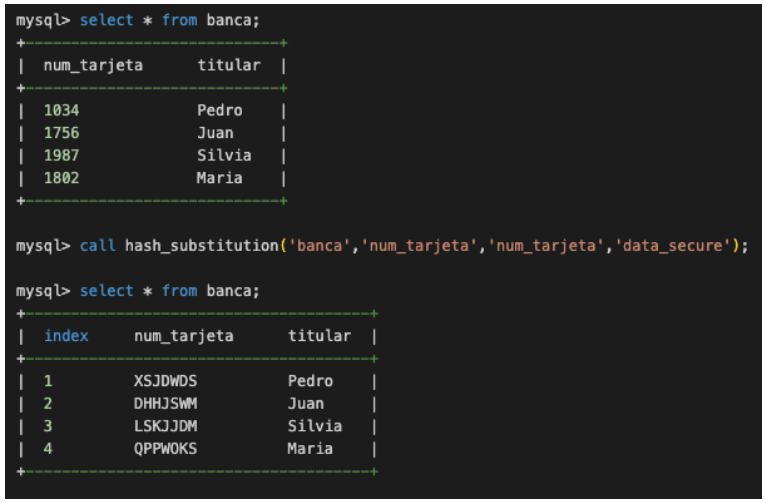


Figura A.20: Procedimiento hash_substitution()

4. ANONIMIZACION Y RECUPERACION DE DATOS

Mediante la técnica de sustitución de datos por valores calculados a través de operaciones matemáticas (por ejemplo, la multiplicación), conseguimos anonimizar datos y que estos no sean identificados. Una buena práctica, consiste en poder recuperar los datos originales (sensibles) sin tener que guardar estos en ninguna otra parte y así evitar ser expuestos. Esto lo conseguimos mediante operaciones matemáticas inversas a las aplicadas anteriormente (por ejemplo, la división).

4.1 Enmascaramiento mediante multiplicación con valores aleatorios

→ mult_masking(): Este procedimiento sustituye los valores de una columna de numeros por valores generados a partir de la multiplicacion del (valor original * valor aleatorio). Para la generacion del valor aleatorio, se puede definir el rango.

Ademas, se construye una tabla a parte donde se guardan los valores aleatorios en el orden en que se usaron. De esta forma, podremos mediante el procedimiento originalValues(), recuperar los valores originales.

Condicion: El numero minimo del rango para la generacion de valores aleatorios debe ser 1.

SYNTAX:
mult_masking('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_valores_random', num_max_rango, num_min_rango)

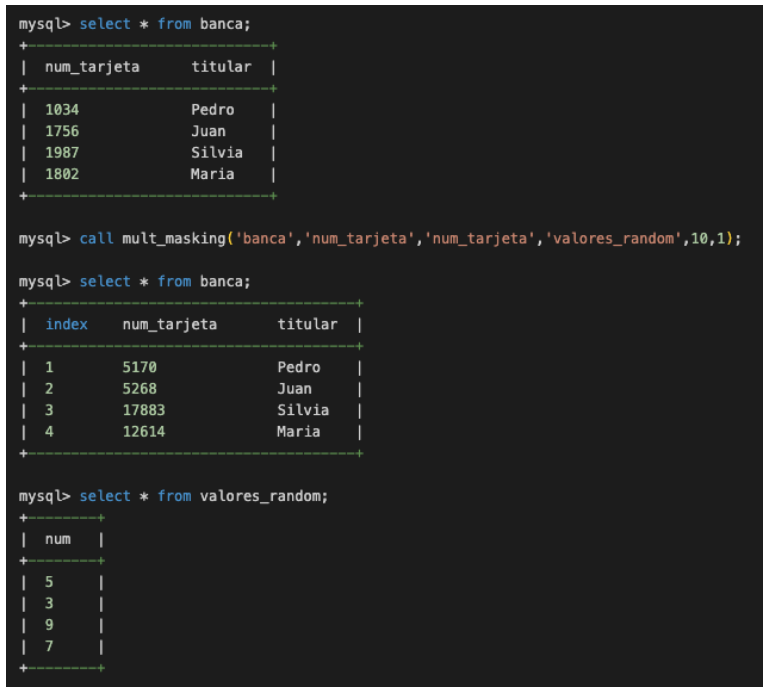


Figura A.21: Procedimiento mult_masking()

4.2 Recuperación de valores originales

→ originalValues(): Este procedimiento recupera los valores originales de una columna de numeros de una tabla. En concreto, es aplicable a la recuperacion de datos tras el enmascaramiento de estos mediante el procedimiento mult_masking().

Se basa en una funcion matematica, si el procedimiento mult_masking() enmascara valores multiplicando por un valor, para recuperar el original se debe dividir por dicho valor.

SYNTAX:
originalValues('tabla_a_modificar', 'colum_a_modificar', 'colum_primary_key', 'tabla_valores_random')

```
mysql> select * from banca;
+-------------------------------------------+
| index    num_tarjeta       titular  |
+-------------------------------------------+
| 1        5170              Pedro    |
| 2        5268              Juan     |
| 3        17883             Silvia   |
| 4        12614             Maria    |
+-------------------------------------------+

mysql> select * from valores_random;
+---------+
| num   |
+---------+
| 5     |
| 3     |
| 9     |
| 7     |
+---------+

mysql> call originalValues('banca','num_tarjeta','index','valores_random');

mysql> select * from banca;
+-----------------------------+
| num_tarjeta      titular  |
+-----------------------------+
| 1034             Pedro    |
| 1756             Juan     |
| 1987             Silvia   |
| 1802             Maria    |
+-----------------------------+
```

Figura A.22: Procedimiento originalValues()

Esta ha sido una presentación de las herramientas disponibles en el toolkit. Sin embargo, se recomienda encarecidamente que se lea los ficheros "README.txt, Funciones_EjemplosUso.txt y Procedimientos_EjemplosUso.txt" que se adjuntan con la memoria D.

A.0.3. Uso particular de herramienta

Una de las particularidades que tienen los procedimientos del apartado 1.2 es que, si durante el proceso de anonimización se encuentra un valor en la tabla que no cumple las condiciones, es posible corregir dicho error y retomar la anonimización desde ese punto.

Para entender mejor como se ejecutarían estos procedimientos en caso de error, se expone a continuación un ejemplo:

```
mysql> SELECT * from elecciones;
+-----------------------------------------------------------+
| ciudad      candidato      dni           codCiudad   |
+-----------------------------------------------------------+
| Valencia      Pedro        71987132V     1606        |
| Pamplona      Juan         78945212Z     6098        |
| Cadiz         Silvia       84154135T     1721        |
| Zamora        Maria        714552S       4598        |
| Bilbao        Sara         71235838J     2421        |
| Cuenca        Pablo        65468213M     1556        |
+-----------------------------------------------------------+

mysql> call proc_numdni_mask('elecciones','dni','dni','data_secure', 0);

mysql> 'Se ha detectado un valor incorrecto en la posicion 4. Debe estar formado por 9 digitos'

mysql> SELECT * from elecciones;
+-------------------------------------------------------------------+
| index    ciudad      candidato      dni           codCiudad   |
+-------------------------------------------------------------------+
| 1        Valencia      Pedro        7XXXXXX2V     1606        |
| 2        Pamplona      Juan         7XXXXXX2Z     6098        |
| 3        Cadiz         Silvia       8XXXXXX5T     1721        |
| 4        Zamora        Maria        714552S       4598        |
| 5        Bilbao        Sara         71235838J     2421        |
| 6        Cuenca        Pablo        65468213M     1556        |
+-------------------------------------------------------------------+

(Corregimos el valor de la posicion 4 del index de la tabla, supongamos que
lo reemplazamos por '71455247S')

mysql> call proc_numdni_mask('elecciones','dni','dni','data_secure', 4);

mysql> SELECT * from elecciones;
+-------------------------------------------------------------------+
| index    ciudad      candidato      dni           codCiudad   |
+-------------------------------------------------------------------+
| 1        Valencia      Pedro        7XXXXXX2V     1606        |
| 2        Pamplona      Juan         7XXXXXX2Z     6098        |
| 3        Cadiz         Silvia       8XXXXXX5T     1721        |
| 4        Zamora        Maria        7XXXXXX7S     4598        |
| 5        Bilbao        Sara         7XXXXXX8J     2421        |
| 6        Cuenca        Pablo        6XXXXXX3M     1556        |
+-------------------------------------------------------------------+
```

Figura A.23: Ejemplo reanudación proc_numdni_mask()

Apéndice B

Resumen de herramientas

- **FUNCIONES**

1. ENMASCARAMIENTO DE DATOS PARA ELIMINAR CARACTERISTICAS IDENTIFICATIVAS

1.1 DE PROPOSITO GENERAL

→ inner_mask()

→ inner_mask_simple()

→ outer_mask()

1.2 DE PROPOSITO ESPECIFICO

1.2.1 Enmascaramiento de números de cuentas bancarias o tarjetas de crédito

→ numacc_mask()

→ numacc_mask_relaxed()

→ numcard_mask()

→ numcard_mask_numb()

1.2.2 Enmascaramiento del Número de Seguridad Social (NUSS)

→ numnuss_mask()

1.2.3 Enmascaramiento del Documento Nacional de Identidad (DNI)

→ numdni_mask()

1.2.4 Enmascaramiento de datos personales como fecha de nacimiento o código postal

→ numfech_mask()

→ numcodpost_mask()

- **PROCEDIMIENTOS**

1. ENMASCARAMIENTO DE DATOS PARA ELIMINAR CARACTERISTICAS IDENTIFICATIVAS

1.1 DE PROPOSITO GENERAL

→ proc_inner_mask()

→ proc_outer_mask()

1.2 DE PROPOSITO ESPECIFICO

1.2.1 Enmascaramiento de números de cuentas bancarias o tarjetas de crédito

→ proc_numacc_mask()

→ proc_numacc_mask_relaxed()

→ proc_numcard_mask()

→ proc_numcard_mask_numb()

1.2.2 Enmascaramiento del Número de Seguridad Social (NUSS)

→ proc_numnuss_mask()

1.2.3 Enmascaramiento del Documento Nacional de Identidad (DNI)

→ proc_numdni_mask()

1.2.4 Enmascaramiento de datos personales como fecha de nacimiento o código postal

→ proc_numfech_mask()

→ proc_numcodpost_mask()

2. GENERANDO DATOS ALEATORIOS USANDO DICCIONARIOS

→ randata_dictionary()

3. USO DE TECNICAS TIPICAS EN LA ANONIMIZACION

3.1 Enmascaramiento mediante uso de valores promedio

→ avg_substitution()

3.2 Enmascaramiento mediante hashing

→ hash_substitution()

4. ANONIMIZACION Y RECUPERACION DE DATOS

4.1 Enmascaramiento mediante multiplicación con valores aleatorios

→ mult_masking()

4.2 Recuperación de valores originales

→ originalValues()

Apéndice C

Diccionario de términos

Este diccionario sirve de apoyo al lector para conocer las definiciones de algunos de los conceptos que se utilizan en este documento. Las definiciones que se muestran a continuación pueden encontrarse en la Guía de Anonimización de la AEPD [11] .

- **Datos personales:** toda información relacionada con una persona física identificada o identificable, "el interesado" (RGPD, art. 4.1).

- **Datos sensibles:** son todos aquellos datos personales referidos en el artículo 9 del RGPD (en especial datos financieros y médicos), los cuales suelen ser útiles para la realización de estudios, por lo que es importante que estén presentes. Pero a su vez es crítico que no se pueda identificar al propietario de estos, por tener importantes implicaciones para su privacidad al tratarse de información confidencial del individuo.

- **Tratamiento:** cualquier operación o conjunto de operaciones realizadas sobre datos personales o conjuntos de datos personales, ya sea por procedimientos automatizados o no, como la recogida, registro, organización, estruc- turación, conservación, adaptación o modificación, extracción, consulta, utilización, comunicación por transmisión, difusión o cualquier otra forma de habilitación de acceso, cotejo o interconexión, limitación, supresión o destruc- ción (RGPD, art. 4.2).

- **Ofuscación de datos:** tratamiento para cambiar o alterar datos sensibles o que identifican a una persona, con el objetivo de proteger la información confidencial.

- **Anonimización de datos:** define la metodología y el conjunto de buenas prácticas y técnicas que reducen el ries- go de identificación de personas, la irreversibilidad del

proceso de anonimización y la auditoría de la explotación de los datos anonimizados, monitorizando quién, cuándo y para qué se usan. Es decir, cubre tanto el objetivo de anonimización, como el de mitigación del riesgo de reidentificación, siendo este último un aspecto clave.

- **Reidentificación:** identificar a las personas específicas a las que pertenecen los datos a partir de ellos. Es uno de los riesgos clave a mitigar en un proceso de anonimización de datos.

- **Datos anónimos:** datos que en ningún momento han contenido información personal de un individuo, por lo que no son datos personales, ni están afectados por el RGPD.

- **Datos anonimizados:** son datos que permitían identificar a una persona física o jurídica en su forma original, pero que han pasado por un proceso de anonimización que imposibilita la reidentificación del propietario, por lo que ya no son datos personales, ni están afectados por el RGPD.

Apéndice D

Repositorio ficheros

En este anexo, se agrega la URL del repositorio que permite acceder a todos los ficheros que se adjuntan con esta memoria.

En este repositorio, se encuentran dos carpetas:

- **Código:** contiene los ficheros .sql de código fuente de las funciones y procedimientos implementados para este proyecto.
- **ManualUsuario_Ejemplos:** contiene varios ficheros entre los cuales se encuentra: el manual de usuario (README), ficheros con ejemplos de uso de funciones y procedimientos, tablas pobladas y un diccionario de datos ejemplo.

URL: `https://drive.google.com/drive/folders/1v9fuFmS43GzXMps43EPzpWfY2hDQkc9C?usp=share_link`