



UNIVERSIDAD DE VALLADOLID



**ESCUELA TÉCNICA SUPERIOR
DE INGENIEROS DE TELECOMUNICACIÓN**

TRABAJO FIN DE GRADO
**GRADO EN INGENIERÍA DE TECNOLOGÍAS ESPECÍFICAS DE
TELECOMUNICACIÓN, MENCIÓN EN TELEMÁTICA**
**“PROSPECTOR DE NOTIFICACIONES
ADMINISTRATIVAS DE TRÁFICO”**

AUTOR: REBECA GARCÍA COCA

TUTOR: MÍRIAM ANTÓN RODRÍGUEZ

Junio 2018

TÍTULO: “PROSPECTOR DE NOTIFICACIONES ADMINISTRATIVAS DE TRÁFICO”

AUTOR: REBECA GARCÍA COCA

TUTOR: MÍRIAM ANTÓN RODRÍGUEZ

DEPARTAMENTO: TEORÍA DE LA SEÑAL, COMUNICACIONES E INGENIERÍA TELEMÁTICA

Miembros del Tribunal

PRESIDENTE: MÍRIAM ANTÓN RODRÍGUEZ

SECRETARIO: DAVID GONZÁLEZ ORTEGA

VOCAL: MARIO MARTÍNEZ ZARZUELA

SUPLENTE 1: FRANCISCO JAVIER DÍAZ PERNAS

SUPLENTE 2: M^a ÁNGELES PÉREZ JUÁREZ

FECHA DE LECTURA: Junio 2018

CALIFICACIÓN:

RESUMEN DEL PROYECTO

El presente Trabajo Fin de Grado se centra en el desarrollo de un nuevo sistema de procesamiento de datos no estructurados que se publican diariamente en internet, para su tratamiento, consolidación y posterior análisis. En concreto, el desarrollo de este sistema surge por la necesidad de acceder, procesar y analizar los datos sobre sanciones o multas de tráfico que los organismos públicos (ayuntamientos y otras instituciones) están obligados a publicar en el Tablón Edictal Único del Boletín Oficial del Estado (BOE).

En esta memoria se expone el diseño y los pasos seguidos en la implementación del sistema continuando con las pruebas de rendimiento y validación. Finalmente, se concluye con las conclusiones y futuras líneas.

PALABRAS CLAVE

Tablón Edictal Único, sanciones de tráfico, prospección, información desestructurada, procesamiento de datos, Cuadro de Mando.

ABSTRACT

This end-of-degree project focuses on the development of a new system for processing unstructured data that is published daily on the Internet, for its treatment, consolidation and subsequent analysis. Specifically, the development of this system arises from the need to access, process and analyze data on penalties or traffic punishment that public organisms (council and other institutions) are obliged to publish on the Tablón Edictal Único of the Boletín Oficial del Estado (BOE).

In this report the design and the steps followed in the implementation of the system are exposed, continuing with the performance and validation tests. Finally, it concludes with the conclusions and future lines.

KEYWORDS

Tablón Edictal Único, traffic punishment, research, unstructured information, data processing, Dashboard.

A mi familia y amigos, por haber estado en las buenas, pero sobre todo en las malas. A esos que empezaron siendo compañeros de clase y se convirtieron en grandes amigos, por su ayuda y por todos esos momentos inolvidables juntos. A AlamoConsulting, por darme esta oportunidad. Y a mi tutora, por ayudarme en todo lo posible con este trabajo.

Índice de Contenidos

Índice de Contenidos	9
Índice de Figuras	11
Índice de Tablas.....	12
1. INTRODUCCIÓN Y OBJETIVOS.....	13
2. ANÁLISIS DE TECNOLOGÍAS	16
2.1. Introducción.....	16
2.2. Arquitectura MVC.....	16
2.3. Tecnologías para plataformas web	17
2.4. Gestores de base de datos	18
2.5. Tecnologías utilizadas	19
2.5.1. Tecnologías del lado del servidor.....	19
2.5.2. Tecnologías del lado del cliente	20
2.5.3. Base de datos.....	21
3. TRABAJO REALIZADO	22
3.1. Introducción.....	22
3.2. Diseño del Sistema.....	22
3.2.1. Introducción	22
3.2.2. Fuente de datos externa	22
3.2.3. Contexto Global del Sistema.....	24
3.2.4. Funcionamiento.....	25
3.2.5. Arquitectura del Sistema	27
3.2.6. Diagrama de Componentes del Sistema.....	29
3.2.7. Interacción de los componentes	29
3.2.7.1. Carga de Datos	32
3.2.7.2. Procesado de Datos	32
3.2.7.3. Visualización de Datos	33
3.2.8. Diseño de la Base de Datos	34
3.2.8.1. Modelo Conceptual de Base de Datos.....	34
3.2.9. Requisitos Hardware y Software del Sistema.	35
3.2.9.1. Requisitos Software.....	35
3.2.9.2. Requisitos Hardware	36
3.3. Implementación del Sistema.....	38

3.3.1.	Introducción	38
3.3.2.	Base de Datos	38
3.3.2.1.	Modelo físico de datos.....	39
3.3.2.2.	Procedimientos almacenados.....	47
3.3.3.	Servicios Windows.....	48
3.3.4.	FrontEnd.....	49
3.3.4.1.	Interfaz administrador	49
3.3.4.2.	Interfaz analista	50
3.4.	Pruebas y Resultados	52
3.4.1.	Introducción	52
3.4.2.	Resultados	52
3.4.2.1.	Pruebas de rendimiento	53
3.4.2.2.	Pruebas de certificación.....	59
3.5.	Planificación y presupuesto	60
3.5.1.	Introducción	60
3.5.2.	Planificación.....	60
3.5.3.	Presupuesto	62
4.	CONCLUSIONES Y FUTURAS LÍNEAS DE TRABAJO.....	64
4.1.	Conclusiones	64
4.2.	Futuras líneas de trabajo.....	65
4.2.1.	Otras funcionalidades	65
4.2.2.	Arquitectura del sistema.....	66
ANEXO		70
Marco de Regulación Legal		70
BIBLIOGRAFÍA		72

Índice de Figuras

Figura 2-1. Componentes arquitectura MVC	17
Figura 3-1. Fragmento inicial de uno de los documentos.....	23
Figura 3-2. Segundo fragmento de uno de los documentos	24
Figura 1-3. Fragmento de un documento en una lengua distinta al castellano.....	24
Figura 3-4. Contexto global del sistema.....	25
Figura 3-5. Esquema general de los módulos de funcionamiento del sistema	26
Figura 3-6. Esquema general del sistema, arquitectura lógica	28
Figura 3-7. Diagrama de componentes del sistema.....	29
Figura 3-8. Flujo global de funcionamiento del sistema	30
Figura 3-9. Diagrama de secuencia del Módulo de Carga de Datos	32
Figura 3-10. Diagrama de secuencia inicial del Módulo de Procesamiento	32
Figura 3-11. Diagrama de secuencia con un ejemplo de tratamiento de datos	33
Figura 3-12. Diagrama de secuencia del Módulo de Visualización de Datos.....	34
Figura 3-13. Modelo Conceptual de Base de Datos	35
Figura 3-14. Arquitectura física del sistema.....	38
Figura 3-15. Esquema Físico de la Base de Datos del Sistema.....	41
Figura 3-16. Configuración de los motores en la interfaz de administrador	50
Figura 3-17. Interfaz para la generación de ficheros	50
Figura 3-18. Interfaz de los usuarios analistas	51
Figura 3-19. Planificación del proyecto	62
Figura 4-1. Infraestructura AWS del sistema	68

Índice de Tablas

Tabla 3-1. Requisitos Software del sistema.....	36
Tabla 3-2. Requisitos Hardware del sistema	37
Tabla 3-3. Resultados prueba número de documentos procesados	54
Tabla 3-4. Resultados prueba tamaño de documentos procesados.....	55
Tabla 3-5. Resultados prueba número de sanciones procesadas	57
Tabla 3-6. Resultados prueba número de personas procesadas.....	58
Tabla 3-7. Resumen de tareas y tiempo dedicado al proyecto	63
Tabla 3-8. Costes del personal del proyecto.....	63

Capítulo 1

1. INTRODUCCIÓN Y OBJETIVOS

1.1. Contexto y motivación

En la actualidad, se ha producido un aumento considerable de información pública que aparece diariamente en internet y otras fuentes abiertas, que se considera de interés. El problema de estos datos públicos radica habitualmente en que están desestructurados, es decir, que no siguen un formato estándar o no se encuentran almacenados en estructuras de datos relacionales predefinidas, y su tratamiento y gestión con las herramientas y técnicas de inteligencia de negocio que hasta ahora se venían usando para el procesamiento de datos, ya no es factible.

En los últimos años el avance experimentado por las nuevas tecnologías y los sistemas de información más avanzados están permitiendo la construcción de herramientas-sistemas de gestión que permiten el acceso, tratamiento y procesamiento de datos de cualquier tipología y almacenados en cualquier formato, en el menor tiempo posible.

Este proyecto ha permitido desarrollar un sistema de prospección, tratamiento y representación de información sobre notificaciones administrativas de tráfico. Este aplicativo recibe el nombre de Prospector de Notificaciones Administrativas de Tráfico, que a lo largo de esta memoria se referenciará abreviadamente como ProNAT. El origen de los datos que procesa son los documentos PDF publicados diariamente en el Boletín Oficial del Estado para comunicar las sanciones de tráfico que no se han podido notificar a los ciudadanos.

1.2. Objetivos

Este proyecto consiste en el desarrollo de un sistema de prospección o búsqueda masiva de información pública en internet de notificaciones administrativas de tráfico para su procesamiento, tratamiento y visualización de resultados. Los objetivos en los cuales se basan las fases de desarrollo son los siguientes:

1. Estudio del contexto y la motivación que llevan a la realización del proyecto.
2. Presentación del contexto de funcionamiento global del sistema.
3. Diseño del sistema de prospección que se adapta a las necesidades.
4. Implementación del sistema en base al diseño del mismo.
5. Visualización de los resultados del procesamiento de los datos.
6. Realización de pruebas del sistema para la obtención de resultados.
7. Extracción de conclusiones del desarrollo del proyecto.
8. Planteamiento de líneas futuras ofreciendo continuidad al proyecto.

1.3. Estructura del documento

Para poder facilitar la estructura de la memoria se describe brevemente a continuación una estructura de la misma, es decir, cada uno de los apartados que se incluyen en el documento:

Capítulo 1. Introducción y Objetivos.

En este primer capítulo se exponen las motivaciones del proyecto y se establecen los objetivos del mismo recogiendo la idea general del trabajo que se va a realizar. Además, se enuncia la estructura de la memoria.

Capítulo 2. Análisis de tecnologías.

En esta sección se analizan las posibles tecnologías con las que implementar el aplicativo expuesto en este trabajo, determinando las elegidas para el desarrollo y los motivos de esta elección.

Capítulo 3. Trabajo Realizado.

En los distintos apartados que integran este capítulo de la memoria, se describe en detalle el trabajo realizado para el análisis, diseño, implementación y pruebas del sistema objeto del proyecto. A continuación se resumen dichos apartados:

- Diseño del sistema.

Este capítulo describe el diseño del sistema Prospector de Notificaciones Administrativas de Tráfico, ProNAT, así como los componentes que lo forman y la interacción entre ellos.

- Implementación.

El apartado Implementación, recoge la descripción de los pasos que se han seguido en el desarrollo de la implementación del sistema diseñado, desde la carga inicial de datos hasta la visualización de los resultados obtenidos en el procesado

- Pruebas y Resultados.

En este capítulo se explican las pruebas que se han realizado para conocer los límites del aplicativo, el volumen de datos que maneja y los tiempos de ejecución alcanzados. Además, en este punto, se presentan un conjunto de estadísticas o indicadores globales obtenidos tras el análisis de los resultados de las pruebas del sistema.

Capítulo 4. Conclusiones y Líneas de Trabajo Futuras.

En el capítulo 4, se muestran el análisis de la consecución de los objetivos y las posibles líneas futuras de trabajo a realizar.

Anexo. Marco de Regulación Legal

Para finalizar, en el anexo se detalla el marco de la legislación del Estado Español en la que se basa el aplicativo implementado en el trabajo.

Capítulo 2

2. ANÁLISIS DE TECNOLOGÍAS

2.1. Introducción

A lo largo de este apartado se pretende explicar las ventajas que ofrecen tanto la arquitectura como las tecnologías utilizadas en la implementación del sistema ProNAT. Para ello se ha realizado un análisis de las posibilidades que existen para desarrollar un aplicativo web, centrándose en las tecnologías más usadas hoy en día.

2.2. Arquitectura MVC

El modelo de la arquitectura Modelo-Vista-Controlador, MVC, se compone de (Microsoft, 2012):

- Modelos, son las partes de la aplicación que implementan la lógica de datos. Son el nexo de unión entre la base de datos y el resto del sistema, almacenando la información que se procesa y representa.
- Vistas, son los componentes que se muestran en la interfaz de usuario de la aplicación. Normalmente se basan en los modelos obtenidos con el tratamiento de la información existente en la base de datos.
- Controladores, son los encargados de capturar las peticiones de los usuarios, controlando el tratamiento de los datos de los modelos y representando los resultados a través de las vistas.

La interacción de los componentes que se acaban de exponer con los usuarios del sistema se muestra en la figura 2-1.

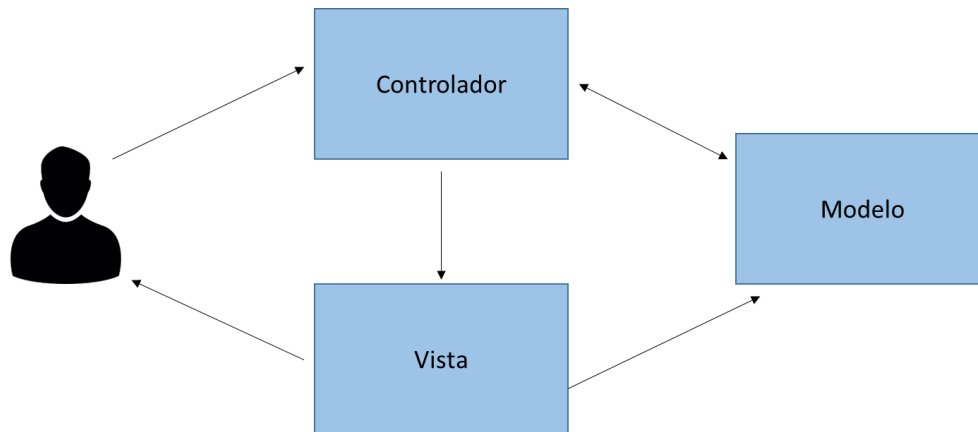


Figura 2-1. Componentes arquitectura MVC

En comparación con una arquitectura basada en Formularios Web este tipo de modelo ofrece la separación de los diferentes aspectos de la aplicación, los datos, la lógica de negocio y la representación al usuario, a la vez que proporciona un mínimo acoplamiento entre estos. Gracias a esta independencia en los elementos que intervienen en el sistema se facilita su implementación y, además, permite el desarrollo en paralelo de este (Microsoft, 2012).

2.3. Tecnologías para plataformas web

En la actualidad las tecnologías de desarrollo web han evolucionado mucho, existiendo multitud de frameworks y librerías con las que poder implementar un sistema como el que se presenta en este trabajo. Además, todas estas herramientas están en una evolución constante con las mejoras ofrecidas en cada nueva versión (IngenioVirtual, 2015). Dentro de este gran número de posibilidades las tecnologías de desarrollo web más utilizadas en la actualidad son (Gallego, 2014):

- **JSP:** esta tecnología ha sido desarrollada por una coalición de empresas entre las que se encuentran Oracle, IBM y Red Hat. Se basa en el lenguaje Java que, junto con HTML, permiten generar páginas web de forma dinámica y sencilla. Todas las aplicaciones desarrolladas con JSP deben ejecutarse en un servidor de aplicaciones Java, aunque este pueda estar integrado en otro.

- PHP: desarrollada por PHP Group y con licencia libre, esta tecnología de lado de servidor se ha utilizado para programar la mayoría de los servidores en Internet. Normalmente se integra con Apache y MySQL en entornos Linux.
- ASP.NET: es la evolución del ASP clásico para plataformas Windows. A pesar de no tener una licencia libre, lo que limita la comunidad de desarrolladores, si posee la estabilidad y la calidad que ofrece su desarrollador, Microsoft.

2.4. Gestores de base de datos

Un sistema gestor base de datos, SGBD, permite almacenar, modificar y extraer información de una base de datos, ofreciendo la posibilidad de interactuar con los datos a través de un lenguaje de programación (Gitmedio, 2016).

Las bases de datos se pueden clasificar en dos tipos, relacionales y no relacionales, esta misma clasificación se hace extensible a los SGBD. A diferencia de los sistemas no relacionales, los relacionales permiten organizar los datos desestructurados en tablas y establecer conexiones, es decir ‘relaciones’, entre ellas (UPCI, 2014). Este apartado se va a centrar en el análisis de sistemas relacionales ya que es el tipo utilizado para llevar a cabo el aplicativo expuesto en este trabajo. A continuación se exponen los más utilizados en la actualidad:

- MySQL: es un sistema de gestión de base de datos relacional, multihilo y multiusuario distribuido por Oracle bajo licencia GPL y comercial (Gitmedio, 2016).
- Oracle SQL: al igual que MySQL, es un sistema de gestión de base de datos desarrollado por la compañía Oracle. Es una de los más completos ya que ofrece soporte de transacciones, estabilidad y escalabilidad, además de ser multiplataforma. Su principal desventaja es el elevado coste de la licencia, aunque existe una versión express que es gratuita, sólo no se permite su uso comercial (Gitmedio, 2016).
- Microsoft SQL Server: es un software propietario de gestión de base de datos creado por la compañía Microsoft. Este sistema, además de poseer las mismas ventajas expuestas para Oracle SQL, soporta procedimientos almacenados y permite trabajar en modo cliente-servidor. Su desventaja fundamental es su

precio, aunque posee una versión gratuita para pequeños entornos (Iruela, 2016) (Gitmedio, 2016).

2.5. Tecnologías utilizadas

2.5.1. Tecnologías del lado del servidor

Para desarrollar el aplicativo web que forma parte del sistema se ha elegido el entorno de desarrollo ASP.NET. Este framework ofrece la posibilidad de utilizar cualquier lenguaje compatible con CRL, Common language Runtime. De entre estos posibles lenguajes se ha utilizado C# por el conocimiento previo de este. La combinación de ASP.NET y C# ofrece las siguientes características que facilitan el desarrollo de todos los requisitos del sistema ProNAT (Microsoft, 2007):

- El código se ejecuta en el lado del servidor, generando dinámicamente el contenido de las páginas web.
- Es totalmente compatible con distintos dispositivos y exploradores.
- Compila el código optimizando el rendimiento, esto es debido a que ASP.NET realiza una segunda compilación en código nativo.
- Proporciona una infraestructura de seguridad para gestionar el acceso de los usuarios y realizar otras tareas de seguridad. En esta aplicación web se realiza la autenticación de usuarios mediante formularios ASP.NET, que permiten el control de acceso a través de una cuenta de Windows.
- Su administración de estados permite almacenar y gestionar información de toda la aplicación. Lo que ofrece la posibilidad de acceder desde varios equipos manteniendo los estados del usuario.
- Permite desarrollar el aplicativo utilizando la arquitectura MVC. Este es modelo elegido para la implementación del sistema por las ventajas expuestas en apartados anteriores de este capítulo.

Para la implementación del código diseñado se ha utilizado la herramienta Visual Studio, este entorno de desarrollo integrado ofrece una total compatibilidad con el lenguaje y el entorno de desarrollo web elegidos.

Con el fin de aprovechar al máximo las posibilidades que ofrece ASP.NET se han utilizado los siguientes complementos a la hora de desarrollar el código:

- ASP.NET Razor, es una sintaxis de marcado de plantilla de ASP.NET, basado en el lenguaje de programación C#. Permite al programador construir páginas web con código HTML. De esta forma, con ASP .NET, se pueden implementar páginas estáticas a las que se agrega contenido de forma dinámica según las peticiones del usuario (Microsoft, 2017).
- PDF Focus .NET, librería que permite trabajar fácilmente con documentos PDF. Este complemento se utiliza en este proyecto como ayuda para parsear los documentos publicados en el BOE (Sautinsoft, 2017).

2.5.2. Tecnologías del lado del cliente

Además del framework ASP.NET es necesario utilizar otras tecnologías para poder implementar todos los requisitos del aplicativo. A continuación se enumeran los lenguajes y librerías utilizados en el desarrollo del lado del cliente:

- HTML es un lenguaje de programación web que se utiliza para estructurar y representar el contenido para las páginas webs. En este trabajo se ha utilizado la versión HTML5 ya que ofrece nuevas características, reduce el uso de los recursos y, además, permite una mayor compatibilidad con los nuevos dispositivos (W3, 2017). Como se ha expuesto en el apartado anterior, gracias a la sintaxis Razor este lenguaje permite integrar la interfaz del aplicativo dentro del resto del código diseñado para el sistema.
- CSS es un mecanismo simple que ofrece a los desarrolladores el control total sobre estilo y formato en el que se representarán los documentos (W3, 2017).
- JavaScript es un lenguaje de script orientado a objetos (Mozilla Corporation, 2017). En este proyecto se ha utilizado en el lado del cliente para permitir la interacción con el usuario, principalmente para realizar validaciones de formularios y peticiones Ajax al servidor. Este tipo de peticiones permiten actualizar el estado de la página web sin tener que volver a cargar el documento completo, lo que mejora la respuesta a las acciones del usuario (Mozilla Corporation, 2012).

- Para el diseño de la aplicación web se han utilizado las librerías Bootstrap y JQuery. Junto con las anteriores tecnologías, permiten la creación una interfaz responsive que se adapte a cualquier tipo de dispositivo y que facilite la interacción del usuario. Estos frameworks, además, son totalmente compatible con la mayoría de los navegadores web (jQuery Foundation, 2017) (Otto, Mark; Thornton, Jacob; Bootstrap contributors , 2017). En cuanto a los gráficos que presenta la interfaz de usuario han sido implementados con la librería de JavaScript Highcharts (Highcharts, 2017).

2.5.3. *Base de datos*

Por otro lado, el gestor de base de datos utilizado es Microsoft SQL Server. Esta herramienta al igual que ASP.NET está diseñado por Microsoft, por lo que ofrecen una total compatibilidad, facilitando su integración y configuración. SQL Server ofrece una seguridad transparente al desarrollador que protege los datos tanto en reposo como en movimiento. Además de las anteriores características, permite un alto rendimiento con grandes volúmenes de datos. Por estos motivos este gestor es el elegido para la implementación de la base de datos (Microsoft, 2017).

Además, para extraer la información necesaria de la base de datos, se ha decidido utilizar procedimientos almacenados de SQL Server. Se ha seleccionado esta tecnología frente a otras por dos razones: en primer lugar, se ejecutan directamente en el gestor de base de datos, y por tanto la velocidad de procesamiento de la información es mayor, y en segundo lugar, ofrece una separación entre la lógica de la aplicación y la extracción y procesamiento de los datos, facilitando el mantenimiento de la aplicación. En el apartado que expone la implementación del sistema se detalla el uso de esta herramienta en el desarrollo tanto del modelo físico como de los procedimientos almacenados.

Capítulo 3

3. TRABAJO REALIZADO

3.1. Introducción

Una vez realizado el análisis de contexto del sistema, este apartado se centra en el diseño del mismo. En primer lugar se presentará una visión general del sistema describiendo los principales módulos y flujo de funcionamiento. Posteriormente se detallará la arquitectura técnica y arquitectura de sistemas que dará soporte a la solución y los componentes que lo integran. Por último, se ilustrarán las distintas interacciones entre los componentes para entender el funcionamiento del sistema.

3.2. Diseño del Sistema

3.2.1. *Introducción*

En este apartado se pretende describir la arquitectura del sistema y todos aquellos componentes que interactúan entre sí dentro de la solución, indicando los algoritmos de flujo de información que se aplican dentro de la solución para la obtención, procesado y visualización final de la información, y la explotación posterior de dicha información por parte de los usuarios que utilizan ProNAT. Se establecerán, además, todos aquellos requisitos mínimos tanto Software como Hardware necesarios para el correcto funcionamiento del sistema.

3.2.2. *Fuente de datos externa*

Desde el 1 de junio de 2015, el BOE, Boletín Oficial del Estado, se ha convertido en el tablón de anuncios obligatorio de todas las Administraciones Públicas de España. El Tablón Edictal Único es el nuevo medio de notificación electrónica que utilizan todas las Administraciones para hacer llegar a los ciudadanos los avisos que no han podido ser entregados de forma personal (Ministero de la Presidencia y para las administraciones territoriales, 2015). Un modelo parecido al que siguen los actuales Boletines Provinciales, pero que tendrá una única ubicación y recopilará todos los avisos de los diferentes

organismos públicos y sus materias. Aunque puedan seguir existiendo otros boletines o sistemas de notificación mediante anuncios, desde esta fecha, es el mencionado medio de notificación el que centraliza todos los anuncios que realicen las Administraciones Públicas de España (Ministerio de la Presidencia y para las administraciones territoriales, 2015).

Puesto que este nuevo medio electrónico de notificaciones se aplica a todos y cada uno de los anuncios de notificación que deben publicar las Administraciones Públicas, se incluyen también los avisos relacionados con sanciones o multas de tráfico. El sistema ProNAT se limita a realizar la prospección de este tipo de notificaciones.

Cada administración es la encargada de realizar sus propias notificaciones, por lo que existe gran variedad en los documentos que se publican en la sede electrónica del TEU. Aunque todos ellos se publican en un formato común, PDF, no todos están estructurados de igual manera. El sistema ProNAT es capaz de procesar las distintas cabeceras de estos documentos para su clasificación. Realizando además una búsqueda en el resto del contenido para encontrar los datos de las sanciones, que normalmente se pueden localizar en tablas al final de los documentos. En las Figuras 3-1 y 3-2 se puede ver, a modo de ejemplo, fragmentos obtenidos de un documento publicado en el TEU (Ministerio de la Presidencia y para las administraciones territoriales, 2017).



Suplemento de Notificaciones

ADMINISTRACIÓN LOCAL

SALAMANCA

AYUNTAMIENTO DE SALAMANCA

Anuncio de notificación de 14 de noviembre de 2017 en procedimiento de Notificaciones de denuncia.

ID: N1700744441

De conformidad con lo dispuesto en los artículos 91 y 92 del Real Decreto Legislativo 6/2015, de 30 de octubre, por el que se aprueba el texto refundido de la Ley sobre Tráfico, Circulación de Vehículos a Motor y Seguridad Vial (LTSV), se hace pública notificación de los expedientes sancionadores que a continuación se relacionan, ya que habiéndose intentado la notificación en el último domicilio conocido, ésta no se ha podido practicar.

Asimismo, se requiere al titular del vehículo, en el supuesto de que no fuera el responsable de la infracción en el momento en que los hechos fueron denunciados, para que en el plazo de veinte días naturales contados desde la recepción de la presente

Figura 3-1. Fragmento inicial de uno de los documentos

LTSV = Ley sobre Tráfico, Circulación de Vehículos a Motor y Seguridad Vial, RGC = Reglamento General de Circulación, OTSV = Ordenanza de Tráfico, Aparcamiento, Circulación y Seguridad Vial

EXPEDIENTE	DENUNCIADO/A	IDENTIF	LOCALIDAD	FECHA	MATRÍCULA	CUANTÍA EUROS	PRECEPTO	ARTº	PTOS	REO
10158083	MARQUE MADRINE JESUS ANTONIO	11889452	VILLARES DE LA REINA	07/09/2017	8138-BLS	200,00	RGC	94	0	
10158086	PEREZ ADRIAN JAVIER	20180216	LA POBLA DE VALLBONA	09/09/2017	2802-FIN	90,00	RGC	94	0	
10158124	COUSO CASTRO VICTOR	45887168	HERVAS	09/09/2017	8166-LJP	90,00	RGC	94	0	
10158137	OLLEDO ROSSETI ALVARO MANUEL	70160669	SEVILLA	09/09/2017	8867-FGU	90,00	RGC	94	0	
10158204	TORRES LOPEZ ROSAIDA	70801060	PARADA DE ARRIBA	10/09/2017	1801-0668	90,00	RGC	94	0	
10158314	PEREZ SANCHEZ DE HERRICOLAIDA C	07908761	SALAMANCA	11/09/2017	4101-0001	90,00	RGC	94	0	
10158417	MARTIN OLIVERAS ALBERTO	70884882	CARRASCAL BARREGAS	11/09/2017	4838-DJN	90,00	RGC	170	0	
10158421	OTONDALEZ RODRIGUEZ JORGE	70880110	SALAMANCA	11/09/2017	0688-DJN	200,00	RGC	91	0	
10158448	SOTO REAL MARIA DEL MAR	21706229	SALAMANCA	11/09/2017	3071-DJL	200,00	RGC	94	0	
10158490	DE LA VEGA SANCHEZ MARIA ROSARIO	18871076	CASTELDEFELS	01/08/2017	4638-IRJ	90,00	RGC	94	0	
10158491	MARTIN HERNANDEZ MIGUEL	07100001	VILLAMAYOR	17/06/2017	2271-APP	90,00	RGC	94	0	
10158516	OTONDALEZ ORADO ALEJANDRO JAVIER	71222938	SALAMANCA	13/09/2017	281-0000-01	60,00	RGC	94	0	
10158540	VILLALBA CARLOS ALBERTO	188776170	SALAMANCA	13/09/2017	1370-89C	90,00	RGC	94	0	
10158513	OTONDALEZ ORADO ALEJANDRO JAVIER	71222938	SALAMANCA	14/09/2017	281-0000-01	60,00	RGC	94	0	
10158659	SANCHEZ MATEOS ELICER	07908778	SANTA MARTA TORMES	15/09/2017	0768-LJP	90,00	RGC	94	0	

Figura 3-2. Segundo fragmento de uno de los documentos

Las administraciones públicas de los territorios en los que existe una según lengua oficial pueden realizar publicaciones en esta lengua, por lo que es necesario dotar al sistema de una base con la que poder procesar estos documentos. En este trabajo se han analizado distintos documentos de estas administraciones, teniendo en cuenta los resultados de este análisis a la hora de implementar el sistema. La Figura 3-3 muestra un ejemplo de los datos de uno de estos documentos (Ministerio de la Presidencia y para las administraciones territoriales, 2017).

EXPEDIENTEA	IZENA	IDENTIF	HERRIA	DATA	MATRIKULA	EURO	AGINDU	ART	PUNTU	ERRE
48/1591843	INSTALACIONES PEREZ AUSIN SC.,	J09375353	BARAKALDO	11/07/2017	2274JTK	200,00	RD 1428/03	167.1	0	
48/6236847	VENCORTE S L.,	B48739080	BILBAO	13/05/2017	4119GSN	100,00	RD 1428/03	52.1	0	

Figura 1-3. Fragmento de un documento en una lengua distinta al castellano

3.2.3. Contexto Global del Sistema

El siguiente diagrama muestra el contexto global del sistema. De un lado están las Comunidades Autónomas y Organismos Públicos (ayuntamientos) que publican las sanciones o multas de tráfico en el Tablón Edictal Único del Boletín Oficial del Estado. ProNAT se conecta a esta fuente pública de datos des-estructurados para extraer, procesar y consolidar con periodicidad diaria, la información de interés, esto es, sanciones o multas de tráfico publicadas en este medio. La interacción entre el sistema y el repositorio de datos externo Tablón Edictal Único se realiza a través de un servicio web. Del otro lado

del sistema están los usuarios finales del sistema quienes a través de un Cuadro de Mando Analítico visualizan la información que procesa, consolida y explota por distintos criterios y que es de utilidad para el análisis de sanciones de tráfico.



Figura 3-4. Contexto global del sistema

Los siguientes apartados abordan el diseño de la arquitectura interna de los módulos y servicios ProNAT y su funcionamiento.

3.2.4. Funcionamiento

El sistema ProNAT cuenta con dos roles de usuario, el usuario administrador o encargado de la parametrización y administración del funcionamiento del aplicativo y el usuario final, que interviene en la consulta o visualización de la información. La descomposición de funcionalidades que caracterizan el sistema da lugar a los cuatro núcleos o módulos principales de funcionamiento del sistema.

La siguiente figura representa el flujo global de funcionamiento del sistema y el esquema general de módulos o componentes que lo integran:

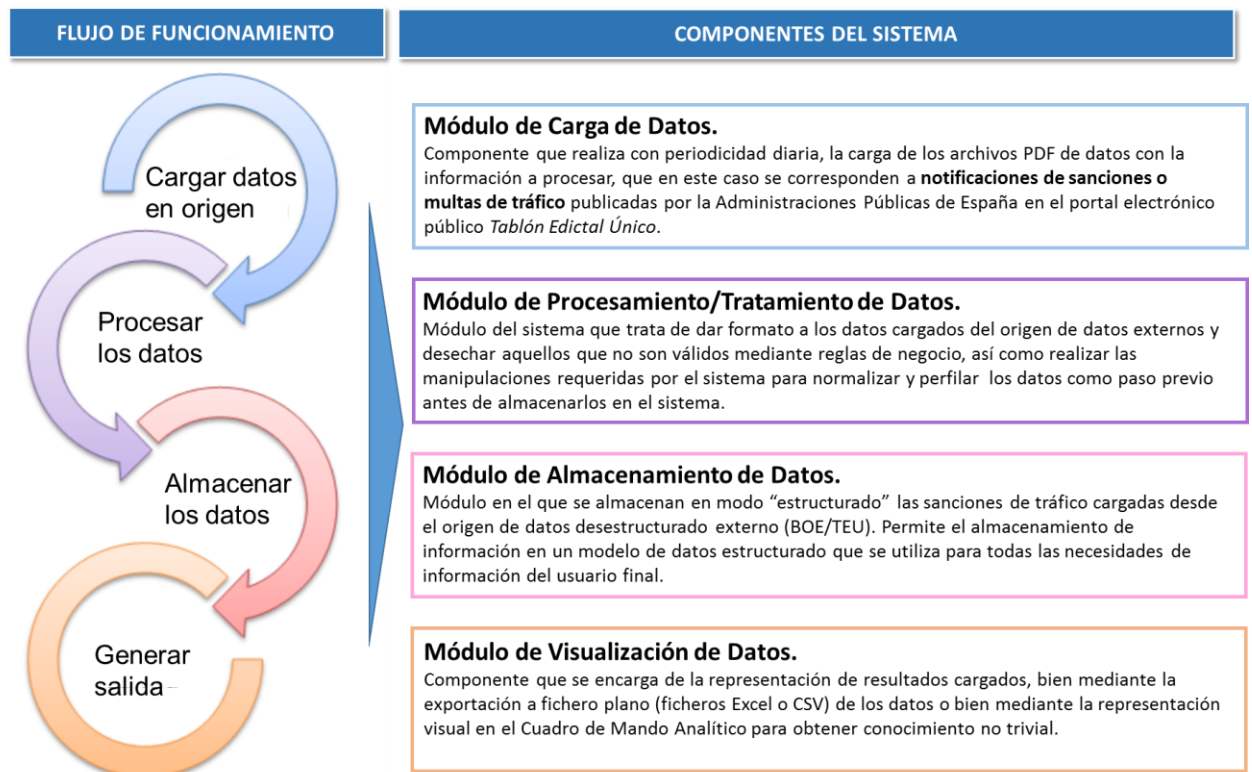


Figura 3-5. Esquema general de los módulos de funcionamiento del sistema

A continuación se describe el objetivo de cada uno de los módulos que integran el sistema, ProNAT, Prospección de Notificaciones Administrativas de Tráfico:

- a) Módulo de Carga de datos: El sistema es capaz de cargar los archivos PDF de datos con la información a procesar, que en este caso se corresponden a notificaciones de sanciones o multas de tráfico publicadas por la Administraciones Públicas de España en el portal electrónico público Tablón Edictal Único. Los datos que se leen de esta fuente, no son homogéneos ni normalizados en cuanto a formato y estructura. Es por esto, que el módulo de carga de datos deberá utilizar un algoritmo de búsqueda de información previamente diseñado para procesar archivos de información desestructurados y que almacenan gran cantidad de datos. El subsistema de carga de datos se inicia mediante la ejecución de un servicio programado previamente definido por el usuario con permisos de administración.
- b) Módulo de Procesado de datos: Los datos origen cargados, al ser datos desestructurados, deben sufrir un proceso de transformación, normalización y limpieza. En este módulo se trata de dar formato a los datos cargados desde el

origen o fuente de datos y desechar aquellos que no son válidos mediante reglas de negocio y manipulaciones implementadas por el sistema. En definitiva, los registros provenientes de los archivos PDFs publicados en el Tablón Edictal Único del BOE requieren de un proceso de normalización, limpieza y perfilado, eliminando aquellos registros e información que no son considerados válidos para el objetivo de negocio de nuestro sistema.

- c) Módulo de Almacenamiento de datos: Los datos de sanciones de tráfico cargados desde los orígenes externos y transformados debidamente por el componente anterior se insertan en el modelo de información del sistema que es la base de datos que se utiliza para todas las necesidades de información del usuario final.
- d) Módulo de Visualización de datos: La representación de los datos posterior a su tratamiento y almacenamiento puede seguir dos vías de interpretación. Se puede tener una representación de resultados cargados mediante la exportación a fichero plano, ficheros Excel o CSV, es decir, se puede exportar el resultado antes de la representación visual en el Cuadro de Mando Analítico o se puede proceder a la explotación del dato para obtener conocimiento no trivial. Uno de los objetivos del proyecto es la visualización de los datos tras su procesado, en concreto, se desean visualizar indicadores de interés para el usuario final, sobre las sanciones de tráfico, filtradas por determinadas dimensiones. A continuación algunos ejemplos:
 - El número de documentos PDF procesados en un determinado período de tiempo.
 - Por provincia y/o comunidad autónoma el número de sanciones de tráfico publicadas.
 - El porcentaje de sanciones correspondientes a particulares y empresas.
 - El ranking de las provincias con más sanciones de tráfico al mes, etc.

3.2.5. *Arquitectura del Sistema*

El diseño del aplicativo ProNAT sigue el patrón de arquitectura de aplicaciones distribuidas, caracterizado porque cada componente ofrece los servicios de presentación,

lógica de negocio, integración y de persistencia o acceso a los datos. Los componentes que realizan los mismos tipos de funciones se pueden agrupar en capas que están organizadas en una jerarquía o pila. Así, los componentes que están en una misma capa interactúan entre ellos directamente y con los componentes de las capas inferiores - pero nunca invocan a los componentes de las capas superiores.

Basándonos en los principios generales de diseño de arquitecturas distribuidas de software, se han establecido las pautas para construir la arquitectura de los módulos del sistema ProNAT separando en capas funcionales cada uno de los componentes o servicios que darán la funcionalidad requerida de cada módulo. Este patrón de arquitectura permite múltiples beneficios en términos de reutilización, escalabilidad, mantenimiento y la optimización del funcionamiento.

Cada una de las capas de la arquitectura del sistema son agrupaciones lógicas de componentes de software y cada una de estas capas ayudan a diferenciar entre los tipos de tareas que realizan los componentes.

ProNAT: PROSPECTOR DE NOTIFICACIONES ADMINISTRATIVAS DE TRÁFICO.

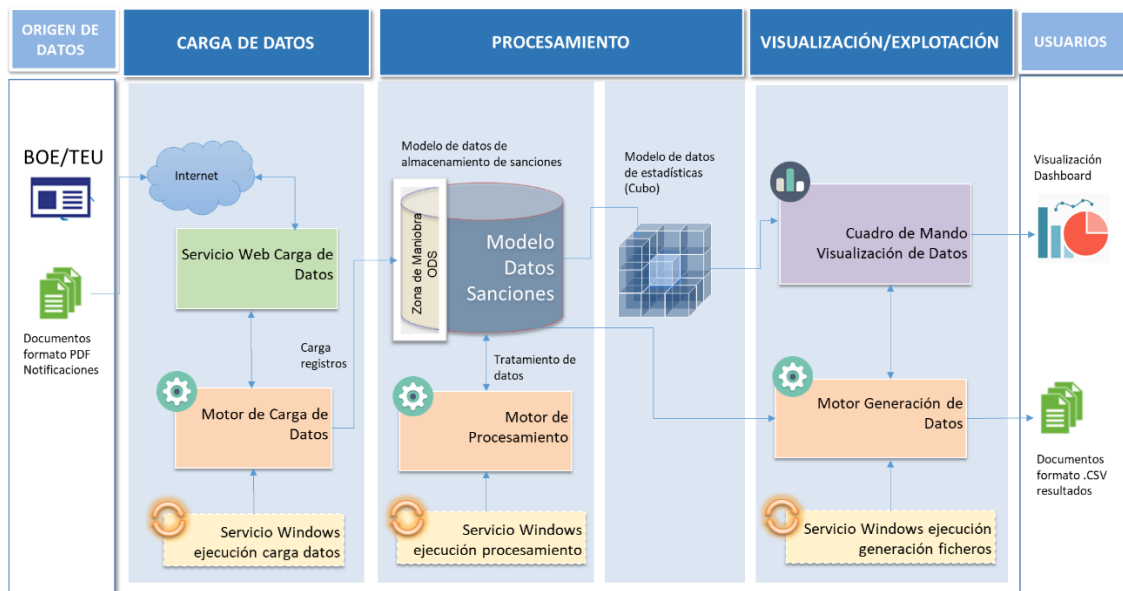


Figura 3-6. Esquema general del sistema, arquitectura lógica

3.2.6. Diagrama de Componentes del Sistema

En la siguiente ilustración, Figura 3-7, se muestra en detalle cómo están separados y organizados los principales componentes que conforman el Sistema, y en que niveles se encuentran cada uno (presentación, lógica, integración o datos), de acuerdo a las funcionalidades que cada uno de ellos tenga asignadas realizar.

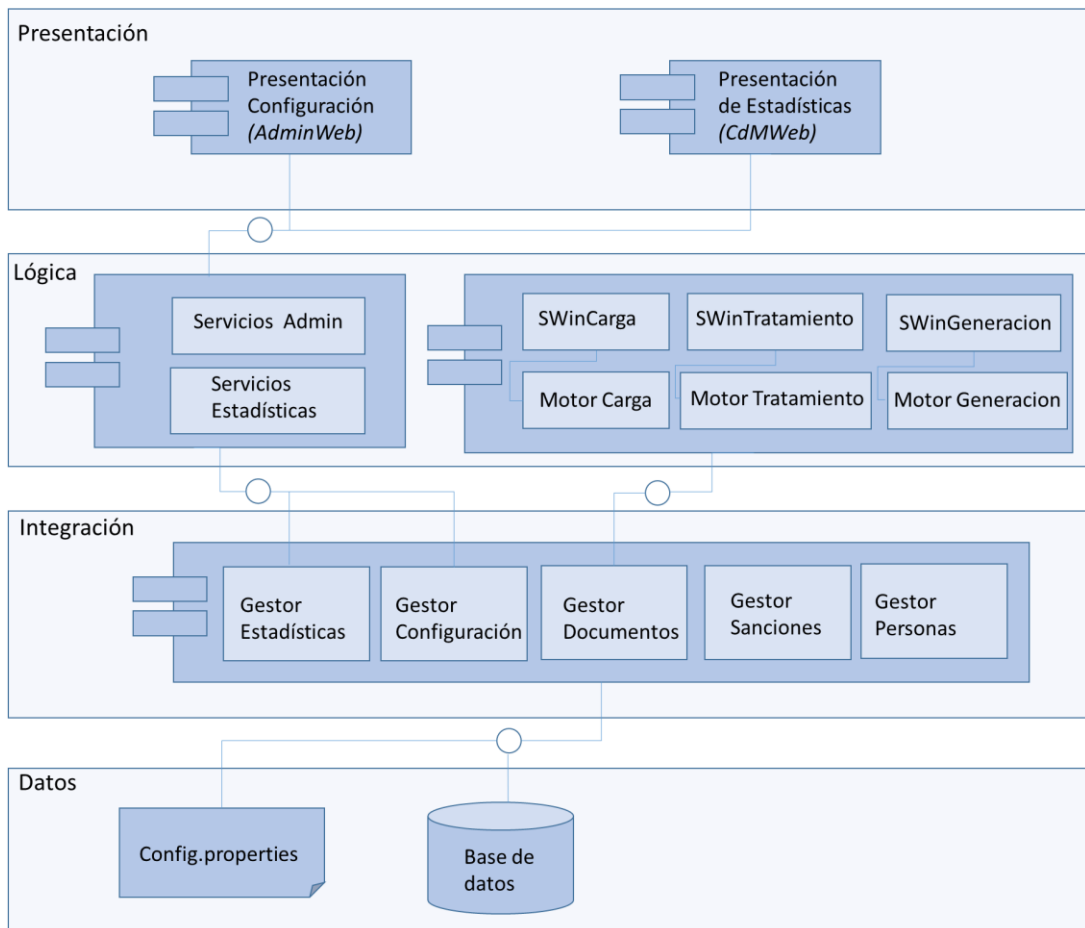


Figura 3-7. Diagrama de componentes del sistema

3.2.7. Interacción de los componentes

Una vez introducidos los componentes fundamentales para la implantación del sistema ProNAT, se describe en la Figura 3-8 cómo colaboran entre ellos para conseguir el correcto funcionamiento del sistema:

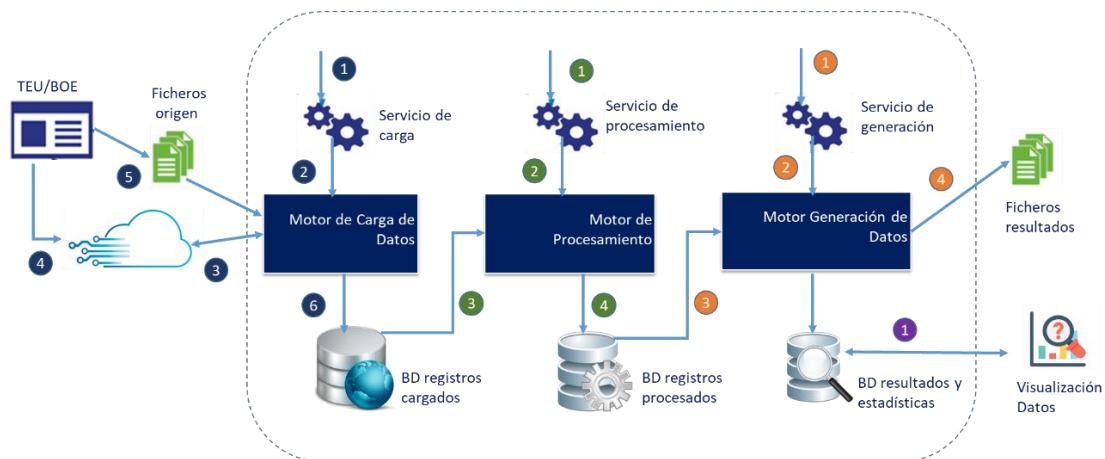


Figura 3-8. Flujo global de funcionamiento del sistema

A continuación se detalla el flujo de funcionamiento del sistema, ProNAT, Prospector de Notificaciones Administrativas de Tráfico:

e) Flujo de funcionamiento Motor de Carga:

1. Se dispara el Servicio Windows de Carga de Datos. Este evento programado inicia la ejecución del subsistema de carga de datos externos y por tanto la carga de registros que se encuentren publicados en el BOE/TEU en ese día relativos a sanciones de tráfico.
2. Se solicita la ejecución del componente Motor de Carga.
3. A través de un servicio web el Motor de Carga accede a internet.
4. Se accede a través del servicio web, a la dirección del portal Tablón Edictal Único del BOE dónde se publican diariamente las notificaciones de sanciones de tráfico en documentos PDF.
5. Se recuperan los documentos PDF publicados en el Tablón Edictal Único del BOE. En concreto se cargan los documentos con sanciones en materias de Tráfico, circulación y seguridad vial.
6. Se almacenan en el modelo de información propio del sistema los datos de identificación del documento PDF.

f) Flujo de funcionamiento del Motor de Procesamiento/Tratamiento:

1. Se dispara el Servicio Windows de Procesamiento. Este evento programado inicia la ejecución del subsistema de procesamiento.
2. Se solicita la ejecución del componente Motor de Carga.
3. Se realiza el procesamiento de la información de los documentos PDF, extrayendo del listado de sanciones las personas (tanto físicas como jurídicas) que han sido sancionadas, así como la sanción o sanciones asociadas con todos los datos que aquí aparezcan. Además, se realizará un tratamiento de estos datos previo a su almacenamiento.
4. Cada persona sancionada y sanción procesada se almacenará en la base de datos propia de la herramienta de prospección.

g) Flujo de funcionamiento del Motor de Generación:

1. Se dispara el Servicio Windows de Generación de Resultados que inicia la ejecución del motor de generación de estadísticas con los datos recuperados de la prospección.
2. Se ejecuta el Motor de Generación de Resultados.
3. Se recuperan los datos resultado de la prospección previa (almacenados en el BD interna del sistema).
4. Se genera el informe o fichero de datos resultado de la prospección. Este fichero contendrá todos los datos de identificación del documento dónde se publicó la sanción, los datos descriptivos de la sanción y adicionalmente, los datos de contacto resultado de la prospección.

h) Flujo de funcionamiento del Módulo de Visualización de Resultados

1. Bajo petición del usuario, se realizará una representación de las estadísticas obtenidas con los datos procesados en el Cuadro de Mando Analítico del sistema.

3.2.7.1. Carga de Datos

En la Figura 3-9, se ilustra la interacción de componentes del primer núcleo de funcionamiento. En primer lugar, se realiza la extracción de los datos originales del sistema de origen y después se introducen los datos en el almacén de datos.

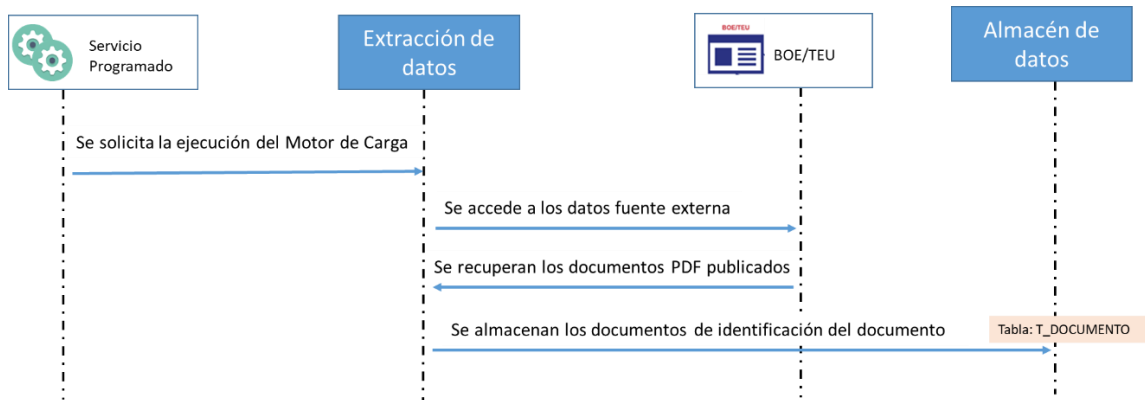


Figura 3-9. Diagrama de secuencia del Módulo de Carga de Datos

3.2.7.2. Procesado de Datos

El primer paso del procesado de datos se muestra en el diagrama de secuencia, Figura 3-10. El Motor de Procesamiento del Sistema se encarga de extraer las personas y las sanciones que se encuentran en los documentos PDF obtenidos. Todo este proceso se desencadena tras la recuperación de los datos originales desde la fuente de datos origen.

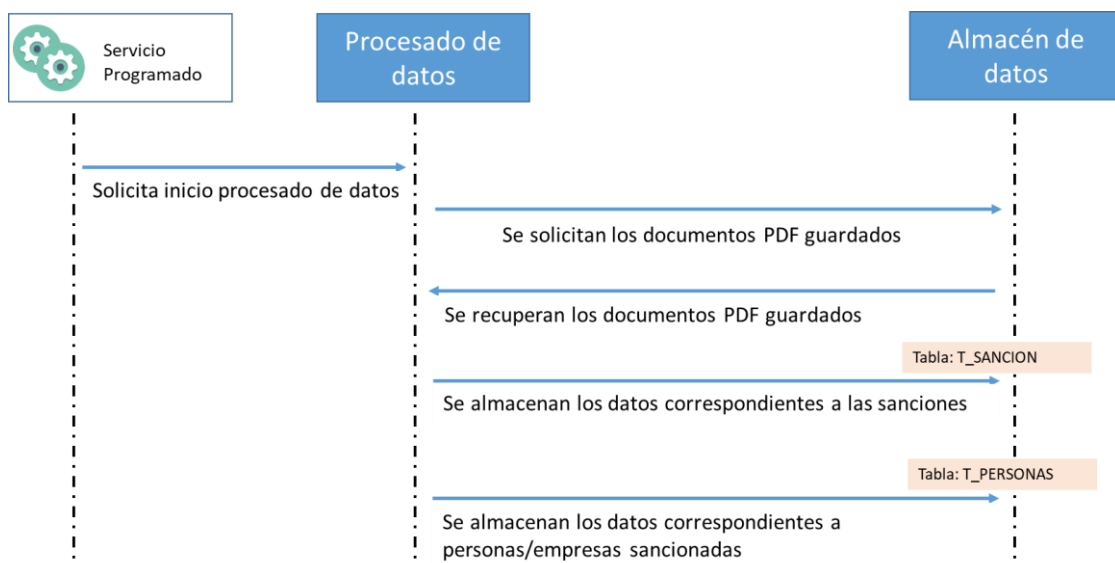


Figura 3-10. Diagrama de secuencia inicial del Módulo de Procesamiento

El siguiente paso consiste en ejecutar la lógica de negocio requerida en cada caso para el tratamiento, limpieza y normalización de la información des-estructura que ha sido cargada desde el origen de datos público, Tablón Edictal Único. Un ejemplo de este proceso es el necesario para identificar la provincia en la que se interpone la sanción. En el siguiente diagrama de secuencia (Figura 3-11) se muestra este ejemplo de tratamiento de datos.

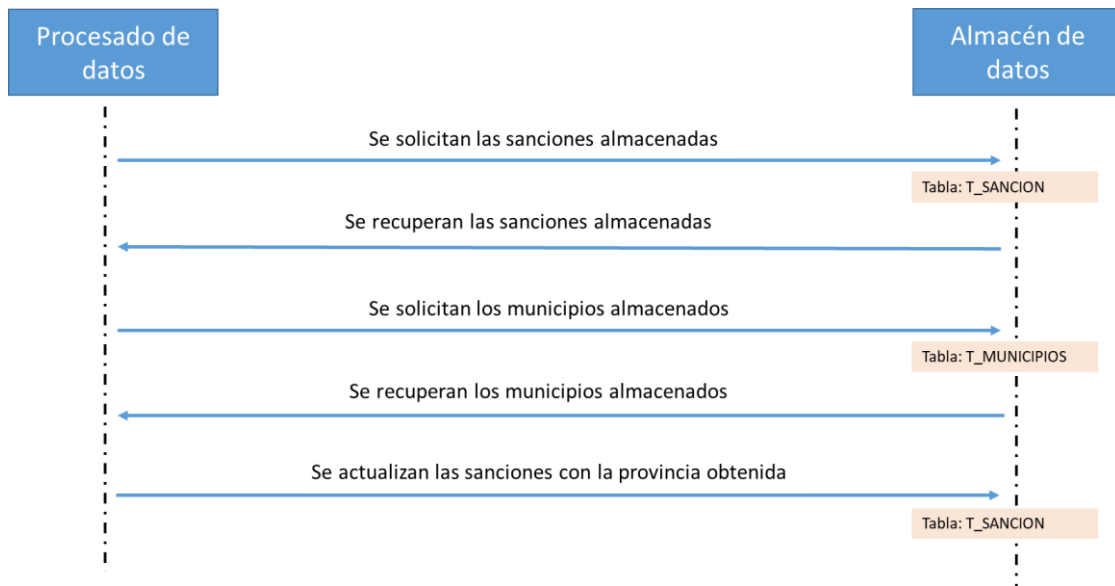


Figura 3-11. Diagrama de secuencia con un ejemplo de tratamiento de datos

Otro ejemplo de tratamiento sobre los datos que realiza el sistema es la normalización de los nombres de las personas sancionadas. Para ello se utiliza un diccionario de nombres almacenado en la base de datos.

3.2.7.3. Visualización de Datos

Bajo petición del usuario, el módulo de visualización de datos es el encargado de solicitar los datos al almacén de información, base de datos del sistema. Para posteriormente realizar la representación oportuna que mostrar al interesado.

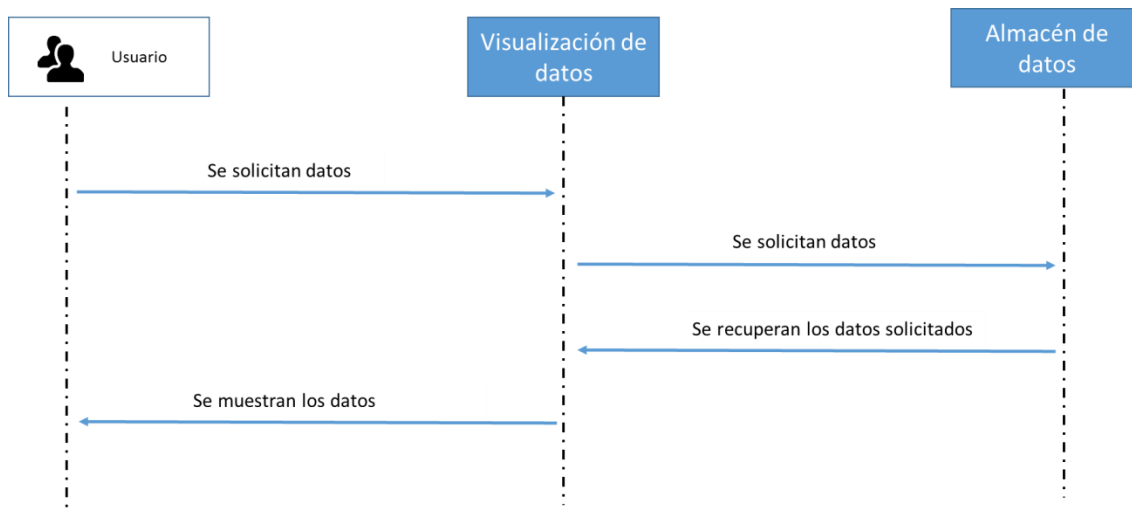


Figura 3-12. Diagrama de secuencia del Módulo de Visualización de Datos

3.2.8. Diseño de la Base de Datos

El diseño de la base de datos que determina como la información que maneja el sistema están lógicamente y físicamente organizados. Para ello, el diseño de la base de datos se establece en dos niveles de detalle. El primer nivel muestra el modelo conceptual de la base de datos, representado en este apartado a través de un diagrama entidad-relación, el cual es independiente del entorno tecnológico utilizado. El segundo nivel presenta el modelo a implementar de la base de datos, descrito mediante un diagrama físico en el apartado correspondiente a la implementación ya que depende del gestor de base de datos seleccionado para su implementación.

3.2.8.1. Modelo Conceptual de Base de Datos

En este apartado se presenta el esquema o Modelo Conceptual de Datos del Sistema. Los elementos contenidos en el esquema conceptual son derivados directamente de los requisitos funcionales y de negocio del sistema. El modelo conceptual de base de datos del Sistema ProNAT se presenta en la siguiente Figura 3-13.

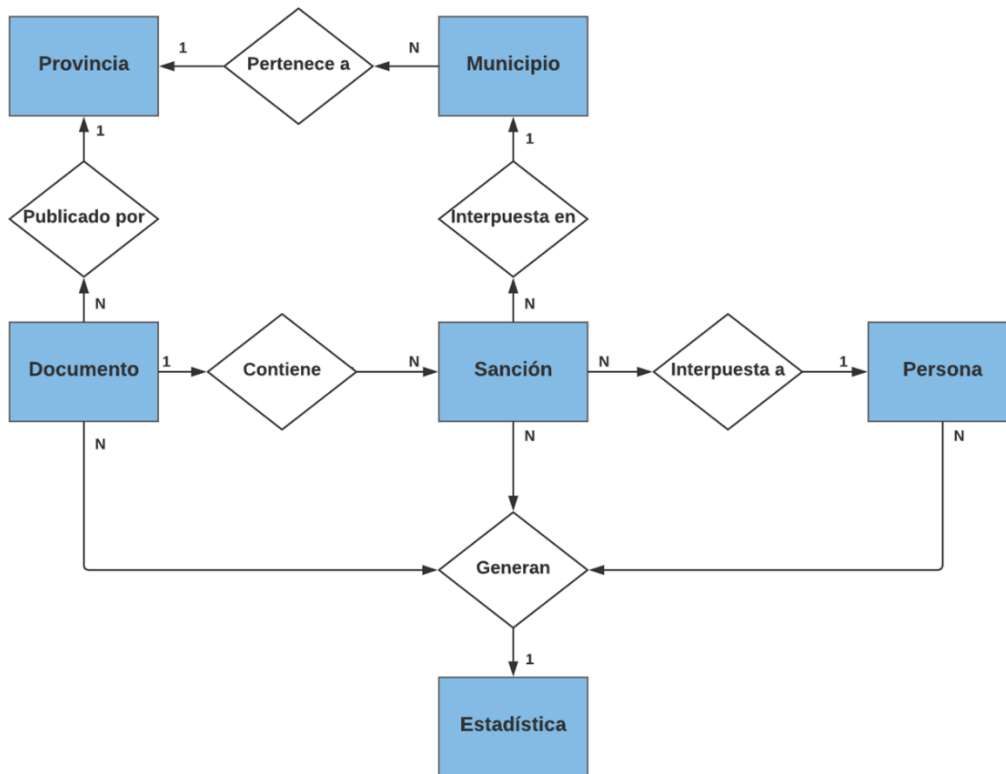


Figura 3-13. Modelo Conceptual de Base de Datos

3.2.9. Requisitos Hardware y Software del Sistema.

Los requisitos del sistema ProNAT se distribuirán de dos formas: *Requisitos Hardware* estableciendo todos los componentes físicos (Servidores, Memoria Ram, CPU, Discos, Procesadores, etc.) que son requeridos por la herramienta para su funcionamiento y *Requisitos Software*, es decir, todos aquellos componentes software requeridos por la herramienta (Frameworks de programación, librerías, conectores, gestor de base de datos, etc.).

3.2.9.1. Requisitos Software

Sistema	Requisitos Software
<p>Servidor App. NET</p> <p>Alojamiento de las Aplicaciones .NET Motor de carga, Motor de procesamiento y Motor de Generación de Ficheros.</p>	<ul style="list-style-type: none"> • Servicio Windows .exe (continuo) • Conectores: • BBDD: .NET Client • Web: interno .NET masivo. • Framework: .NET 4.5.2 o superior
<p>Servidor Web</p> <p>Alojamiento de la aplicación web .NET para la administración de la solución y de la aplicación web de visualización de datos.</p>	<ul style="list-style-type: none"> • App Web .NET • IIS 7.0 ó superior • Acceso a BBDD. • Integración básica (usuario y contraseña).
<p>Base de Datos</p> <p>Almacenamiento de los datos cargados y tratados por el sistema</p>	<ul style="list-style-type: none"> • SQL Server 2014 • Usuario BBDD con permisos de ejecución de procesos almacenados.

Tabla 3-1. Requisitos Software del sistema

3.2.9.2. *Requisitos Hardware*

Sistema	Requisitos Hardware
<p>Servidor Interno</p> <p>Alojamiento de las Aplicaciones .NET (Motores de carga, tratamiento y generación)</p>	<ul style="list-style-type: none"> • Procesador: 2 x Intel Xeon 2.8 GHz • Memoria RAM: 7 GB • Disco: 30 GB + 300 MB

Sistema	Requisitos Hardware
<p>Servidor Outer (DMZ) Alojamiento del Web Service que permite la salida a internet del motor de carga de datos.</p>	<ul style="list-style-type: none"> • Procesador: 2 x Intel Xeon 2.8 GHz • Memoria RAM: 3 GB • Disco: 30 GB + 100 MB
<p>Base de Datos Almacenamiento de los datos requeridos por el sistema.</p>	<ul style="list-style-type: none"> • Procesador: 1 x Intel Xeon 2.8 GHz • Memoria RAM: 3 GB • Disco: 30 GB + 50 GB
<p>Servidor Interno Alojamiento del front-end: aplicativo web de administración y visualización de datos</p>	<ul style="list-style-type: none"> • Procesador: 2 x Intel Xeon 2.8 GHz • Memoria RAM: 7 GB • Disco: 30 GB + 300 MB

Tabla 3-2. Requisitos Hardware del sistema

La siguiente figura representa la arquitectura física del sistema ProNAT, dónde además se puede observar la interacción de los principales elementos software del diseño elaborado:

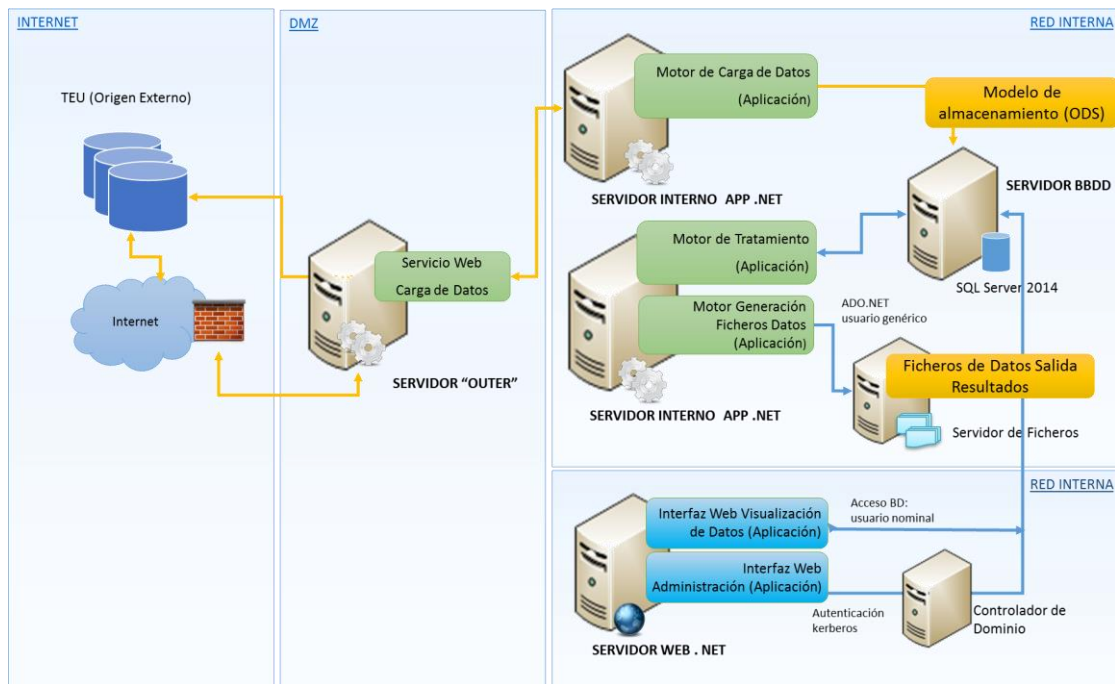


Figura 3-14. Arquitectura física del sistema

3.3. Implementación del Sistema

3.3.1. Introducción

En este apartado se detallan los aspectos más importantes para la implementación del Sistema ProNAT, estableciendo una correspondencia entre los componentes del diseño y los elementos concretos implementados en la tecnología seleccionada.

En primer lugar, se expondrá como se almacenan los datos dentro del entorno y los procedimientos desarrollados para el tratamiento de documentos que contienen las notificaciones o avisos de sanciones de tráfico. Después se detallarán los servicios Windows necesarios para poder lanzar los distintos motores del sistema. Y por último, se describirá tanto la interfaz a la que accederán los administradores, cómo la utilizada por los usuarios analistas en su interacción con el sistema.

3.3.2. Base de Datos

Como se expone anteriormente en este trabajo, la base de datos del sistema ProNAT se ha implementado sobre el gestor de bases de datos Microsoft SQL Server versión 2014. A continuación se describen los elementos creados con esta tecnología divididos según la naturaleza de los mismos.

3.3.2.1. *Modelo físico de datos*

En esta sección se presenta el resultado de la conversión del esquema conceptual de la base de datos diseñado anteriormente a un esquema o Modelo Físico de Datos. Un modelo relacional donde se incluyen los detalles de implementación física de acuerdo al gestor de base de datos seleccionado para el sistema y que describe gráficamente la estructura de la base de datos y sus características físicas.

El Modelo de Almacenamiento de Datos del Sistema ProNAT se puede dividir en dos partes claramente diferenciadas:

- El Modelo de Datos que permite almacenar la información obtenida del origen de datos externo (Tablón Edictal Único) una vez debidamente normalizada. El esquema de entidad-relación del apartado del Modelo Conceptual de la Base de Datos es la base para el diseño de las tablas. Está compuesto por las tablas:
 - T_PERSONA
 - T_SANCIÓN
 - T_DOCUMENTO
 - T_ESTADÍSTICAS
 - T_PROVINCIA
 - T_MUNICIPIO
- También se han identificado algunas tablas auxiliares que permite almacenar los parámetros de configuración y otra información tipificada que requiere el sistema para la búsqueda de documentos en el Tablón Edictal Único. A esta parte corresponden las tablas:
 - T_DICCIONARIO_NOMBRES
 - T_CABECERA_DOCUMENTO
 - T_CABECERA_DOCUMENTO_SINÓNIMOS
 - T_CONFIG
 - T_USUARIO

El esquema físico de la base de datos del Sistema ProNAT, que será creado en nuestro caso sobre el gestor de base de datos Microsoft SQL Server, es mostrado en la siguiente Figura 3-14.

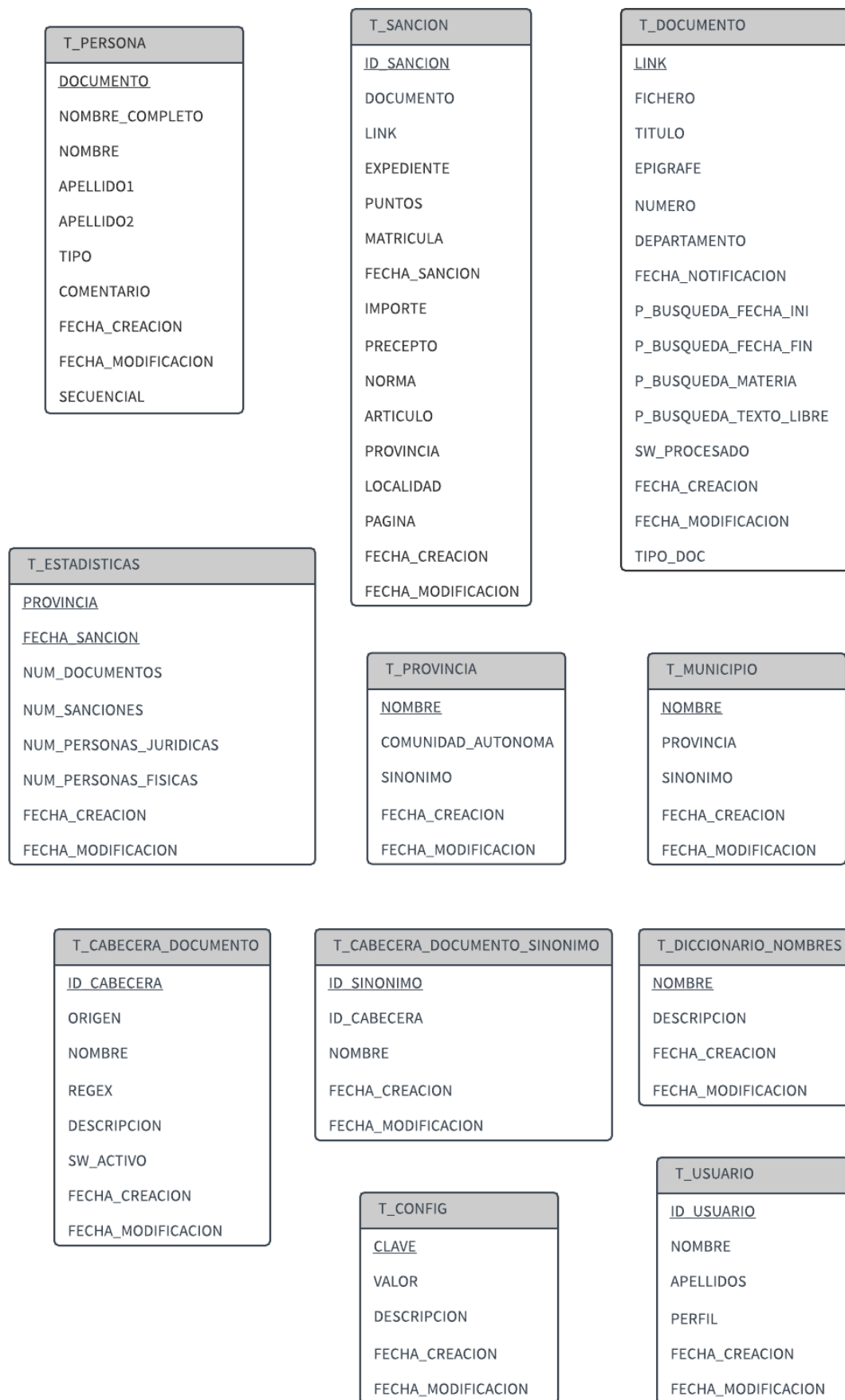


Figura 3-15. Esquema Físico de la Base de Datos del Sistema

A continuación, se describen cada una de las tablas que integran el Modelo Físico que permite el almacenamiento de información requerida por el sistema:

- T_PERSONA: Esta tabla contiene las personas sancionadas. Cada registro de esta tabla contiene los siguientes campos:
 - *Documento*: Documento identificativo de la persona o, en el caso de no obtener este dato en la prospección, un identificador generado automáticamente.
 - *Nombre_completo*: Nombre completo o razón social del sancionado.
 - *Nombre*: Nombre del sancionado.
 - *Apellido1*: Primer apellido del sancionado.
 - *Apellido2*: Segundo apellido del sancionado, en el caso de que exista.
 - *Tipo*: Tipo de persona, Física o Jurídica.
 - *Comentario*: Incluye cualquier comentario o aclaración sobre el registro de la persona sancionada.
 - *Fecha_creación*: Fecha en la que se crea el registro.
 - *Fecha_modificación*: Fecha en la que se actualiza el registro.
 - *Secuencial*: Valor secuencial utilizado para construir el valor del campo documento cuando se obtiene un dato incompleto en la prospección.
- T_SANCION: En esta tabla se almacenarán las sanciones encontradas en los documentos. Cada registro de esta tabla contiene los siguientes campos:
 - *Id_sanción*: Valor auto numérico que identifica la sanción.
 - *Documento*: Documento identificativo de la persona asociada a la sanción.
 - *Link*: URL que identifica el documento donde se ha encontrado la sanción.
 - *Expediente*: Código del expediente de la sanción.
 - *Puntos*: Número de puntos de la sanción, en caso de conllevar de puntos.
 - *Matrícula*: Matrícula del vehículo.
 - *Fecha_sanción*: Fecha en la que se produce la sanción.
 - *Importe*: Importe de la sanción, en caso de conllevar sanción económica.

- *Precepto*: Precepto de la sanción.
- *Norma*: Normativa de la sanción.
- *Artículo*: Código del artículo de la ley que se aplica a la sanción.
- *Provincia*: Provincia en la que se produce la sanción.
- *Localidad*: Localidad en la que se produce la sanción.
- *Página*: Página del documento en la que se encuentra la sanción.
- *Fecha_creación*: Fecha de creación del registro.
- *Fecha_modificación*: Fecha de modificación del registro.
- **T_DOCUMENTO**: Esta tabla contiene la información de los documentos encontrados tras la prospección realizada sobre Tablón Edictal Único. Cada registro de esta tabla contiene los siguientes campos:
 - *Link*: URL que identifica el documento.
 - *Fichero*: Documento PDF descargado del TEU.
 - *Título*: Título del documento.
 - *Epígrafe*: Descripción corta que identifica el contenido del documento.
 - *Número*: Número del documento y su fecha de publicación.
 - *Departamento*: Departamento que publica el documento, puede ser el estado, la administración local, etc.
 - *Fecha_notificación*: Fecha de la notificación del documento.
 - *Provincia*: Provincia a la que pertenece la Administración Pública que publica el documento.
 - *P_búsqueda_fecha_ini*: Fecha de inicio del proceso de la búsqueda del documento.
 - *P_búsqueda_fecha_fin*: Fecha de fin del proceso de la búsqueda del documento.
 - *P_búsqueda_materia*: Código que identifica la materia que se utiliza en la búsqueda del documento.

- *P_búsqueda_texto_libre*: Texto que se utiliza en la búsqueda del documento-
- *Sw_procesado*: Indicador de si los datos del documento ya han sido procesados o no.
- *Fecha_creación*: Fecha de creación del registro.
- *Fecha_modificación*: Fecha de modificación del registro.
- *Tipo_doc*: Texto descriptivo del tipo de documento.
- **T_ESTADISTICAS**: Contiene las estadísticas de interés que se calculan a partir de los datos recopilados por el sistema. Cada registro de esta tabla contiene los siguientes campos:
 - *Provincia*: Nombre de la provincia en la que se produjeron las sanciones
 - *Fecha_sanción*: Fecha en la que se produjeron las sanciones
 - *Num_documentos*: Número de documentos procesados
 - *Num_sanciones*: Número de sanciones
 - *Num_personas_jurídicas*: Número de personas jurídicas a las que se les interpuso una sanción.
 - *Num_personas_físicas*: Número de personas físicas a las que se les interpuso una sanción.
 - *Fecha_creación*: Fecha de creación del registro.
 - *Fecha_modificación*: Fecha de modificación del registro.
- **T_PROVINCIA**: Contiene las provincias de España y la información adicional sobre estas que necesita el sistema para procesar los datos extraídos.
 - *Nombre*: Nombre de la provincia
 - *Comunidad_autónoma*: Comunidad autónoma a la que pertenece la provincia.
 - *Sinónimo*: nombre con el que se identifica la provincia en la segunda lengua oficial para aquellos territorios en los que se publiquen sanciones en varios idiomas.

- *Fecha_creación*: Fecha de creación del registro.
- *Fecha_modificación*: Fecha de modificación del registro.
- T_MUNICIPIO: Contiene los municipios de España y la información adicional sobre estas que necesita el sistema para procesar los datos extraídos.
 - *Nombre*: Nombre del municipio
 - *Provincia*: Provincia a la que pertenece el municipio.
 - *Sinónimo*: nombre con el que se identifica el municipio en la segunda lengua oficial para aquellos territorios en los que se publiquen sanciones en varios idiomas.
 - *Fecha_creación*: Fecha de creación del registro.
 - *Fecha_modificación*: Fecha de modificación del registro.
- T_DICCIONARIO_NOMBRES: Contiene todos los nombres con frecuencia igual o mayor a 20 personas en el territorio español. Esta lista ha sido extraída de la sede electrónica del Instituto Nacional de Estadísticas (INE, 2017). Se utiliza para el tratamiento y normalización de los nombres y apellidos de las personas sancionadas. Cada registro de esta tabla contiene los siguientes campos:
 - *Nombre*: Nombre incluido dentro de la lista.
 - *Descripción*: Incluye cualquier comentario o aclaración sobre el registro.
 - *Fecha_creación*: Fecha de creación del registro.
 - *Fecha_modificación*: Fecha de modificación del registro.
- T_CABECERA_DOCUMENTO: Define los nombres columnas de las tablas origen. Estas tablas aparecen en los documentos PDF publicados en el Tablón Edictal Único y contienen los datos de las sanciones. Cada registro de esta tabla contiene los siguientes campos:
 - *Id_cabecera*: Valor auto numérico que identifica la cabecera.
 - *Orden*: Número de columna de las tablas del documento.
 - *Nombre*: Nombre de la columna de las tablas del documento.
 - *Regex*: Patrón que define los posibles valores.

- *Descripción*: Texto explicativo de la columna de las tablas del documento.
- *Sw_activo*: Indica si esta columna aparece en los documentos.
- *Fecha_creación*: Fecha de creación del registro.
- *Fecha_modificación*: Fecha de modificación del registro.
- T_CABECERA_DOCUMENTO_SINONIMOS: Define los posibles nombres de las columnas de las tablas origen no recogidos en la tabla anterior. Estos valores pueden ser sinónimo, abreviaturas o traducciones a los distintos idiomas oficiales del territorio español. Cada registro de esta tabla contiene los siguientes campos:
 - *Id_sinónimo*: Valor autonumérico que identifica el sinónimo.
 - *Id_cabecera*: Valor que identifica la cabecera a la que pertenece el sinónimo.
 - *Nombre*: Texto con el valor del sinónimo
 - *Fecha_creación*: Fecha de creación del registro.
 - *Fecha_modificación*: Fecha de modificación del registro.
- T_CONFIG: Contiene los valores de los parámetros configurables. Cada registro de esta tabla contiene los siguientes campos:
 - *Clave*: Nombre del parámetro.
 - *Valor*: Valor del parámetro.
 - *Descripción*: Texto descriptivo del parámetro.
 - *Fecha_creación*: Fecha de creación del registro.
 - *Fecha_modificación*: Fecha de modificación del registro.
- T_USUARIO: Contiene la información de los usuarios que podrán acceder al sistema.
 - *Id_usuario*: Identificador del usuario
 - *Nombre*: Nombre del usuario
 - *Apellidos*: Apellidos del usuario

- *Perfil*: Rol del usuario.
- *Fecha_creación*: Fecha de creación del registro.
- *Fecha_modificación*: Fecha de modificación del registro.

3.3.2.2. *Procedimientos almacenados*

Basándose en el diseño realizado, se ha desarrollado la siguiente lista de procedimientos almacenados que se encargarán de la interacción entre el sistema y la información almacenada en la base de datos:

- *P_A_GET_CABECERAS_DOCUMENTOS*: obtiene las cabeceras (nombres de las columnas) y los sinónimos de estas, para poder procesar las tablas que aparecen en los documentos.
- *P_A_GET_CCAA*: obtiene la lista de comunidades autónomas del territorio español.
- *P_A_GET_CONFIG*: obtiene el valor asignado en ese momento a la variable de configuración indicada.
- *P_A_GET_DICCIONARIO_NOMBRES*: obtiene el listado de nombres con los que se compararán los nombres de las personas obtenidas en la carga.
- *P_A_GET_DOCUMENTOS*: obtiene la información almacenada sobre los documentos extraídos de la prospección.
- *P_A_GET_EVOLUTIVO*: obtiene, para las provincias indicadas, el número de sanciones interpuestas para cada día durante el periodo de tiempo elegido.
- *P_A_GET_FICHERO*: obtiene los datos almacenados en la tabla indicada durante el periodo de tiempo elegido.
- *P_A_GET_KPI_GLOBALES*: obtiene el número de documentos y sanciones procesadas para las provincias indicadas durante el periodo de tiempo elegido.
- *P_A_GET_PROVINCIAS*: obtiene la lista de provincias que pertenecen a la comunidad autónoma indicada.

- P_A_GET_TIPO_PERSONA: obtiene el porcentaje de personas físicas y jurídicas sancionadas, para las provincias indicadas durante el periodo de tiempo elegido.
- P_A_GET_TOP_PROVINCIAS: obtiene las provincias con el mayor número de sanciones durante el periodo de tiempo elegido.
- P_A_INSERT_DOCUMENTO: añade los datos de los documentos obtenidos en la prospección.
- P_A_INSERT_ESTADISTICAS: añade las estadísticas calculadas para una provincia y una fecha concreta.
- P_A_INSERT_PERSONA: añade los datos de las personas obtenidas de los documentos cargados.
- P_A_INSERT_SANCION: añade los datos de las sanciones obtenidas de los documentos que han sido prospeccionados.
- P_A_UPDATE_CONFIG: actualiza el valor de la variable de configuración elegida.
- P_A_UPDATE_DOCUMENTO_PROCESADO: actualiza el estado del documento después de realizar el procesado de la información que contiene.
- P_A_UPDATE_PERSONA: actualiza los datos de la persona indicada.
- P_A_VALIDA_USUARIO: verifica si el usuario tiene permiso para visualizar la interfaz de usuario a la que accede.

3.3.3. Servicios Windows

Como se describe en los apartados anteriores es necesaria la implementación de diversos servicios Windows que realice una serie de tareas predefinidas y con el fin de extraer, procesar y generar los datos que explotará la aplicación. A continuación, se exponen los servicios implementados:

- Servicio Windows de Carga de Datos: el objetivo de este servicio es el encargado de lanzar el motor de carga de datos.

- Servicio Windows de Procesamiento: se encarga de invocar el motor de procesamiento de datos.
- Servicio Windows de Generación de Resultados: su finalidad es lanzar la ejecución programada del motor de generación de resultados.

Se podrá configurar el rango de horas en las que se ejecutan estos servicios a través de la interfaz de usuario a la que tendrán acceso los administradores del sistema. Esto permitirá optimizar adecuadamente la ejecución y el procesamiento de los componentes del sistema en los servidores.

3.3.4. *FrontEnd*

Según el perfil definido cada usuario tendrá acceso a unas características de la aplicación web que actúa de FrontEnd del sistema. A continuación, se exponen las interfaces para cada uno de los perfiles de usuario fijados en este sistema.

3.3.4.1. *Interfaz administrador*

Sólo los usuarios identificados como administradores o con perfil de administrador los que accederán a esta interfaz que ofrece las siguientes posibilidades:

- Programar la ejecución de los tres motores que intervienen en el sistema: Motor de Carga de datos, Motor de Procesamiento y Motor de Generación de resultados. Para ello se pueden establecer tres modos de ejecución distintos:
 - Modo Rango: Define el rango horario en el que se ejecutará el motor.
 - Modo Flag: Una vez establecido, obliga al motor a ejecutarse.
 - Modo Horas: El motor se ejecutará una vez a la hora especificada.

Para seleccionar la opción deseada se dispone de la interfaz que se muestra en la Figura 3-16.

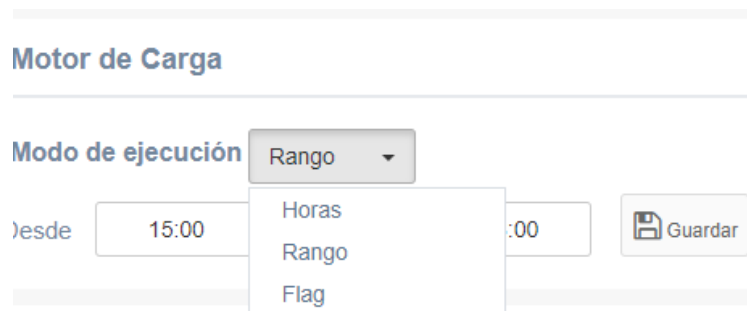


Figura 3-16. Configuración de los motores en la interfaz de administrador

- Definir otras variables de configuración. Esta página web incluye un sencillo formulario con el que establecer los valores de configuración del sistema que se deseen.
- Generar ficheros con la información almacenada en la base de datos. En la Figura 3-17 podemos ver la interfaz desarrollada para esta funcionalidad.



Figura 3-17. Interfaz para la generación de ficheros

3.3.4.2. *Interfaz analista*

Los usuarios con perfil de análisis de información disponen de una aplicación web a modo de Cuadro de Mando Analítico. Este permite la visualización de la información, que el sistema previamente ha procesado y consolidado, para poder realizar el análisis de los datos extraídos en la prospección de sanciones administrativas de tráfico.

En la Figura 3-18 se muestra la interfaz de usuario del sistema para la visualización de resultados de la prospección.

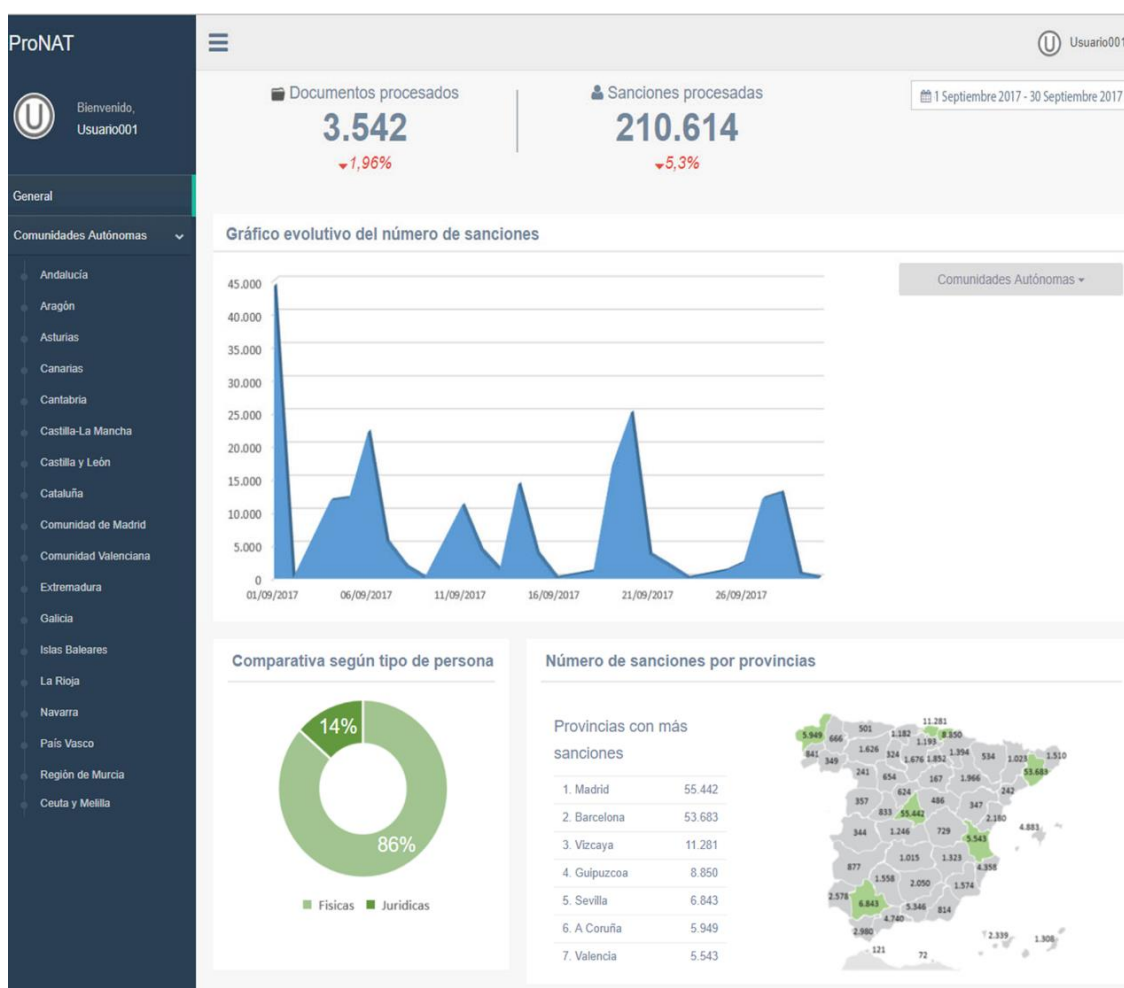


Figura 3-18. Interfaz de los usuarios analistas

El funcionamiento del Cuadro de Mando del sistema ofrece dos tipos de visiones de consulta, que se pueden seleccionar en el menú lateral de la interfaz:

- General, presenta los resultados agregados de todo el territorio español.
- Detalle, presenta los resultados por Comunidad Autónoma.

Adicionalmente, se ofrece la posibilidad de seleccionar un periodo temporal con el que filtrar los datos expuestos en los gráficos e indicadores que se visualizan.

A continuación se resumen las principales estadísticas que un usuario del sistema podrá visualizar a través de su Cuadro de Mando:

- Indicadores globales. En este apartado se incluyen los indicadores de número de documentos y sanciones procesados, junto con la comparativa de estos valores con un periodo de tiempo anterior.

- Evolución temporal de sanciones. En este gráfico se representará el evolutivo temporal del número de sanciones interpuestas por tráfico para cada día del periodo elegido por el usuario. Por defecto la información se presenta agregada para todas las Comunidades Autónomas, si bien, el usuario podrá seleccionar aquellas que desea.
- Número de sanciones por provincia. Incluye un mapa del territorio español en el que se compara el número de sanciones procesadas para cada provincia. Además, se listan las provincias con mayor número de sanciones en el periodo temporal seleccionado por el usuario.
- Comparativa de sanciones según tipo de persona. Este gráfico representa la distribución o porcentaje del total de sanciones que han sido interpuestas a personas físicas y jurídicas (empresas).

3.4. Pruebas y Resultados

3.4.1. Introducción

Tras haber diseñado e implementado el nuevo sistema ProNAT, Prospector de Notificaciones Administrativas de Tráfico, y con el fin de determinar los resultados globales de funcionamiento del aplicativo, detectar posibles limitaciones y conocer el rendimiento y escalabilidad futura del sistema, se han llevado a cabo un conjunto de pruebas cuyos resultados se expone en este apartado.

Estas pruebas se hacen en el entorno de ejecución final y en base a la configuración definitiva del sistema.

3.4.2. Resultados

Se recuerda que el objetivo de las pruebas realizadas es detectar las limitaciones del sistema ProNAT en función de la configuración final del entorno de ejecución. Para ello se considera necesario analizar y medir el volumen de datos que precisa cargar el sistema, así como los tiempos invertidos en su ejecución. En los dos primeros motores que forman parte del sistema las pruebas se han centrado en los tiempos y volumen de

datos procesados. Adicionalmente, en este apartado, se ha realizado la comprobación de la veracidad de los datos obtenidos y representados por el aplicativo.

3.4.2.1. *Pruebas de rendimiento*

En primer lugar, se analiza el rendimiento del componente Motor de Carga, cuya finalidad es realizar la prospección o carga de los ficheros PDF publicados en el TEU y que contienen las sanciones administrativas de tráfico a tratar por el sistema ProNAT.

En este apartado se realiza el análisis y pruebas de funcionamiento del sub-sistema de carga de datos de aplicativo ProNAT. Recordemos que este componente se encarga, diariamente y dentro de una ventana de ejecución previamente determinada por el usuario administrador, de realizar la búsqueda de los documentos PDF que son publicados en el Tablón Edictal Único y posteriormente procesados por el sistema.

Se ha realizado la medida del número de documentos PDF procesados para un periodo de un mes, en concreto, el mes de Septiembre de 2017. La siguiente tabla muestra los resultados obtenidos durante el funcionamiento del Motor de Carga en dicho periodo:

Fecha	Nº de Documentos PDF
01/09/2017	281
02/09/2017	12
04/09/2017	76
05/09/2017	323
06/09/2017	296
07/09/2017	136
08/09/2017	141
09/09/2017	9
11/09/2017	118
12/09/2017	241
13/09/2017	157
14/09/2017	238
15/09/2017	147
16/09/2017	20

Fecha	Nº de Documentos PDF
18/09/2017	84
19/09/2017	68
20/09/2017	169
21/09/2017	113
22/09/2017	140
23/09/2017	10
25/09/2017	80
26/09/2017	69
27/09/2017	240
28/09/2017	230
29/09/2017	133
30/09/2017	11

Tabla 3-3. Resultados prueba número de documentos procesados

En base a los resultados obtenidos en este primer punto, se concluyen que en el mes de Septiembre de 2017 se han publicado 3.542 documentos. Adicionalmente, se obtiene que en promedio por día el sistema extrae 136 documentos PDF.

El objetivo de la segunda prueba es analizar el tamaño de los ficheros que se procesan. En la siguiente tabla se presenta una muestra de ficheros PDF que han sido publicados en el Tablón Edictal Único (muestra seleccionada de ficheros de ejemplo que son cargados por el sistema). Por cada uno de los ficheros de prueba se ha medido su tamaño.

Fecha	Fichero de prueba	Tamaño fichero
01/11/2017	20171101_pdfs_BOE-N-2017-Jefatura de Tráfico de A CORUÑA.pdf	222 KB
01/11/2017	20171101_pdfs_BOE-N-2017-Jefatura de Tráfico de CANTABRIA.pdf	219 KB
01/11/2017	20171101_pdfs_BOE-N-2017-Ayuntamiento de Xativa-VALENCIA.pdf	374 KB

Fecha	Fichero de prueba	Tamaño fichero
07/11/2017	20171107_pdfs_BOE-N-2017-Ayuntamiento de Benidorm-ALICANTE.pdf	610 KB
07/11/2017	20171107_pdfs_BOE-N-2017-Ayuntamiento de Carmona-SEVILLA	309 KB
07/11/2017	20171107_pdfs_BOE-N-2017-Ayuntamiento de MADRID.pdf	2.221 KB
07/11/2017	20171107_pdfs_BOE-N-2017-Ayuntamiento de Tarrasa-BARCELONA.pdf	1.126 KB
14/11/2017	20171113_pdfs_BOE-N-2017-Ayuntamiento de BARCELONA.pdf	312 KB
14/11/2017	20171114_pdfs_BOE-N-2017-Ayuntamiento de Zamora-ZAMORA.pdf	1.301 KB
14/11/2017	20171114_pdfs_BOE-N-2017-Ayuntamiento de Zaragoza-ZARAGOZA.pdf	2.496 KB
21/11/2017	20171121_pdfs_BOE-N-2017-Ayuntamiento de Arroyo Encomienda-VALLADOLID.pdf	967 KB
21/11/2017	20171121_pdfs_BOE-N-2017-Ayuntamiento de Salamanca-dSALAMANCA.pdf	656 KB
21/11/2017	20171121_pdfs_BOE-N-2017-Jefatura de Tráfico de ALMERIA.pdf	925 KB

Tabla 3-4. Resultados prueba tamaño de documentos procesados

Este test sirve para determinar cuál es el tamaño medio de los ficheros PDF que tendría que procesar el sistema ProNAT diariamente para realizar la carga en la base de datos. En base a las pruebas realizadas, se observa que el volumen o tamaño de los ficheros origen es muy variable, publicándose documentos desde aproximadamente 100 KB que contienen muy pocos registros, a documentos de aproximadamente 3MB que contienen un elevado número de datos. A continuación se presentan los resultados finalmente obtenidos en esta prueba, teniendo en cuenta que la muestra queda limitada al mes de pruebas del sistema:

- Tamaño medio de un fichero PDF de datos origen: 952 KB.
- Volumen medio de carga al día: 129.494,67 KB (129 MB).

Con los números obtenidos en las anteriores pruebas se reproduce un escenario de estrés en el Motor de Carga. Para ello se compromete al sistema a prospeccionar un total de 646 documentos, el doble del máximo obtenido, con un tamaño de aproximadamente 5MB. El motor tarda 13 minutos en completar el prospeccionado de estos documentos. Este tiempo cumple el rango de valores que se considera adecuado para esta parte del sistema.

En segundo lugar, se analiza el rendimiento del componente Motor de Procesamiento, que se encarga de realizar el tratamiento de los datos obtenidos en los documentos PDF y almacenar los datos de las personas y sanciones encontradas.

Cada uno de los documentos PDF publicados por el TEU y cargados por el sistema ProNAT presenta un número muy diverso de registros de sanciones, dependiendo entre otros, del día de publicación y del organismo público que realice la publicación. A partir de la misma muestra de resultados del mes de septiembre 2017 se obtienen los siguientes resultados:

Fecha	Nº de sanciones
01/09/2017	44.196
02/09/2017	49
04/09/2017	11.800
05/09/2017	12.189
06/09/2017	22.160
07/09/2017	5.474
08/09/2017	1.786
09/09/2017	61
11/09/2017	11.072
12/09/2017	4.283
13/09/2017	1.207
14/09/2017	14.240
15/09/2017	3.733
16/09/2017	85

Fecha	N° de sanciones
18/09/2017	1.119
19/09/2017	17.098
20/09/2017	25.045
21/09/2017	3.585
22/09/2017	1.877
23/09/2017	63
25/09/2017	1.185
26/09/2017	2.472
27/09/2017	12.073
28/09/2017	12.973
29/09/2017	672
30/09/2017	117

Tabla 3-5. Resultados prueba número de sanciones procesadas

El número total de sanciones recopiladas por el sistema ProNAT durante el mes prefijado para las pruebas es de un total de 210.614 notificaciones de tipología “Tráfico, Circulación y Seguridad Vial”. Adicionalmente se obtiene como resultado, que el aplicativo procesará un número medio al día de aproximadamente 8.101 sanciones.

Además de sanciones, de estos documentos PDF, se extraen los datos de las personas y se realiza un tratamiento antes de almacenarlas en la base de datos. A partir del análisis de los datos obtenidos en el mes de Septiembre de 2017 se determinan los siguientes resultados:

Fecha	N° de personas
01/09/2017	33.199
02/09/2017	46
04/09/2017	8.962
05/09/2017	9.144
06/09/2017	19.665
07/09/2017	4.705

Fecha	N° de personas
08/09/2017	1.571
09/09/2017	53
11/09/2017	9.307
12/09/2017	4.212
13/09/2017	985
14/09/2017	11.924
15/09/2017	2.990
16/09/2017	72
18/09/2017	975
19/09/2017	15.071
20/09/2017	18.784
21/09/2017	2.701
22/09/2017	1.407
23/09/2017	60
25/09/2017	988
26/09/2017	1.859
27/09/2017	10.504
28/09/2017	11.005
29/09/2017	521
30/09/2017	98

Tabla 3-6. Resultados prueba número de personas procesadas

El número total de personas sancionadas que se han encontrado en los documentos por el sistema ProNAT durante el mes prefijado para las pruebas es de un total de 170.808 personas y una media al día de 6.570 personas.

Con los resultados obtenidos en las dos últimas pruebas, se somete al Motor de Procesamiento a un escenario de estrés en el que se realiza el tratamiento de serie de documentos que contienen un total de 88.392 sanciones interpuestas a 66.398 personas. Estos números son el doble de la volumetría máxima obtenida en las pruebas. En este escenario, el motor tarda 22 minutos en completar el procesado de estas personas y

sanciones. Este tiempo cumple el rango de valores que se considera adecuado para esta parte del sistema.

3.4.2.2. *Pruebas de certificación*

En este apartado se recogen los resultados de las pruebas realizadas sobre el Motor de Generación de Resultados y el Cuadro de Mando web del aplicativo. Previa carga y procesamiento de los datos obtenidos en el mes de muestra, Septiembre de 2017, se generan las estadísticas, posteriormente, se realizarán las consultas oportunas que permitirán comprobar que los datos de la tabla T_ESTADISTICAS son coherentes con el resto de tablas de la base de datos. Desde el Cuadro de Mando Web del sistema se visualizan los gráficos e indicadores y se comparará, mediante consultas ejecutadas directamente contra la base de datos, que las representaciones obtenidas concuerdan con la información almacenada. A continuación se presentan los resultados obtenidos para las pruebas expuestas.

Los primeros indicadores que se representan en el Cuadro de Mando son el número total de documentos y sanciones procesadas. Al realizar las consultas sobre la base datos obtenemos que se han procesado 3.504 documentos y 210.614 sanciones. Mismos resultados que se representan en la interfaz de usuario del sistema.

Justo debajo de estos valores se representan los porcentajes de variación de estos indicadores con respecto al periodo anterior. Seleccionando como periodo de visualización en el Cuadro de Mando los últimos 15 días de Septiembre, se obtiene que el porcentaje de variación entre la primera y la segunda quincena de ese mes. Al realizar los cálculos necesarios a través de consultas directas a la base de datos del sistema obtenemos los mismos resultados capturados en el Cuadro de Mando. Durante la segunda quincena se publicó un 37,15% menos de documentos y se procesó un 40,01% menos de sanciones.

Posteriormente, y para cada día del mes de Septiembre, se ha obtenido el conteo de las sanciones almacenados en las tablas de la base de datos. Comprobando que la generación de resultados del sistema ofrece la misma salida que la representada para los usuarios.

De los datos generados en la prospección y el tratamiento realizado por el sistema se obtiene que el número de personas físicas sancionadas es de 181.128, lo que

corresponde al 86% del total que se representa en el Cuadro de Mando. Mientras que el número de personas jurídicas o empresas sancionadas es de 29.486, es decir, un 14% del total, número que también corresponde con lo visualizado por los usuarios.

Por último, con la información almacenada en la tabla T_SANCION se ha conseguido una lista con las distintas provincias y el número de sanciones de ese territorio durante el periodo de muestra. Madrid, Barcelona y Vizcaya encabezan esta enumeración, mismo resultado que el obtenido y representado en el Cuadro de Mando.

Aunque también se han realizado pruebas para todas las comunidades autónomas por separado, en este apartado sólo se han expuesto los resultados obtenidos para el agregado de los datos del territorio español.

3.5. Planificación y presupuesto

3.5.1. *Introducción*

En esta sección, en primer lugar, se exponen la duración de cada una de las fases que se han llevado a cabo para realizar el sistema. Y posteriormente se define el presupuesto realizado basándose en las horas de trabajo necesarias para completar el aplicativo y la memoria de este.

3.5.2. *Planificación*

La duración de la realización del proyecto ha sido de dos meses y medio. A continuación se resumen las fases en las que se ha dividido el desarrollo del proyecto y el tiempo empleado en cada una de ellas, para disponer del sistema ProNAT implantado y en funcionamiento.

- Delimitación del Alcance

En esta fase se pretende centrar la idea general del proyecto. Se delimita el alcance y objetivo de la solución que se plantea como objeto del proyecto y se acuerdan las tecnologías que se emplearán en el desarrollo. El tiempo estimado para esta fase ha sido de 1 semana, 40 horas.

- Diseño del sistema

Fase en la que se definen los requisitos del sistema y la arquitectura de componentes que lo forman, así como la interacción entre ellos y la elección del diseño final. Entre las actividades que se incluyen en esta fase destacan el diseño funcional y técnico de la solución, así como el diseño del plan de pruebas a realizar. El tiempo estimado para esta fase ha sido de 3 semanas, 120 horas.

- Configuración del entorno de implementación

Fase previa al desarrollo que ha consistido en la instalación y configuración de las herramientas a utilizar para la implementación del sistema. Dentro del alcance de esta fase se incluye la instalación del entorno de desarrollo, preproducción y producción. El tiempo estimado para abordar esta fase de configuración de los entorno ha sido de 1 semana, 40 horas.

- Implementación o desarrollo de la solución

Escritura del código necesario para el correcto funcionamiento del sistema, cumpliendo los requisitos establecidos. El tiempo estimado para esta fase ha sido de 4 semanas, 160 horas.

- Implantación de la herramienta

Fase final destinada a implantar la herramienta en el entorno de producción para su utilización por parte de los usuarios finales. Dentro de esta fase se incluye también la generación de documentación para los usuarios. El tiempo dedicado para esta fase ha sido de 1 semana, 40 horas.

- Pruebas y conclusiones

Pruebas de validación del correcto funcionamiento del sistema en el entorno final. En esta fase, adicionalmente, se miden tiempos de ejecución, resultados de funcionamiento y en definitiva, se realizan las pruebas finales que certifican que la solución cumple con los requerimientos inicialmente planteados. El tiempo dedicado para esta fase ha sido de 1 semana, 40 horas.

- Redacción de la memoria

Documentación final de todo el trabajo realizado en el desarrollo completo del proyecto. La tarea de redacción de la memoria sobre el trabajo realizado, se ha llevado a cabo tras la realización del proyecto. El tiempo dedicado ha sido aproximadamente de 2 semanas, 80 horas.

A continuación, en la siguiente figura se muestra el diagrama de tiempos que se ha dedicado a cada una de las fases del proyecto anteriormente indicadas. En este se excluye la fase de redacción de memoria pues su realización se ha llevado a cabo posterior al cierre del proyecto.

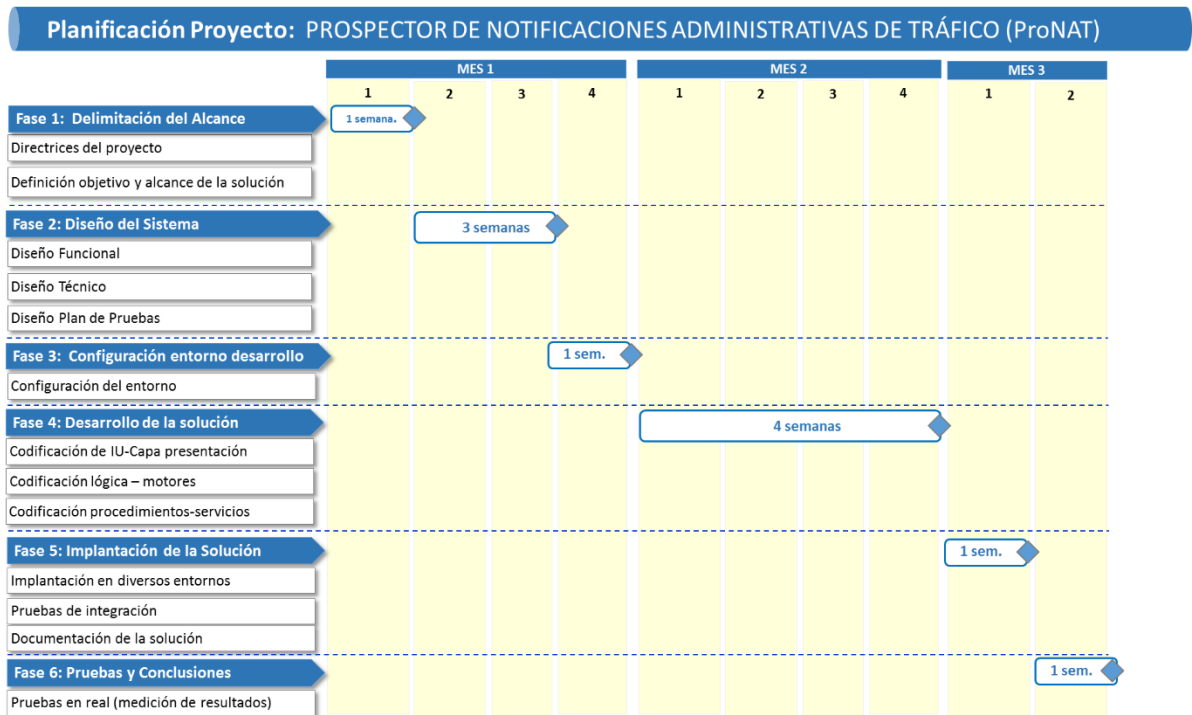


Figura 3-19. Planificación del proyecto

3.5.3. Presupuesto

Aunque en la realización del proyecto se han visto involucradas dos personas, para el cálculo del presupuesto del proyecto únicamente se considera de dedicación de un consultor junior con dedicación total al proyecto. Las tareas del segundo perfil, el consultor senior (jefe de proyecto), han sido únicamente de supervisión y control del proyecto. En la siguiente tabla se recogen el tiempo dedicado por cada fase del proyecto para poder realizar la estimación de coste:

Fase	Dedicación total (en días)	Dedicación total perfil junior (en horas)
Delimitación del Alcance	5	40
Diseño del sistema	10	80
Configuración del entorno de implementación	5	40
Implementación o desarrollo de la solución	20	160
Implantación de la herramienta	5	40
Pruebas y conclusiones	5	40
Total	50	400
Redacción de la memoria	10	80
Total	60	480

Tabla 3-7. Resumen de tareas y tiempo dedicado al proyecto

El presupuesto total asciende a 21600 €, considerando una estimación de 480 horas a una tarifa de 45 €/hora, según se muestra en la siguiente tabla.

Concepto	Dedicación en horas	Coste por hora (€)	Coste Total (€)
Consultor Junior	480	45	21.600€

Tabla 3-8. Costes del personal del proyecto

Capítulo 4

4. CONCLUSIONES Y FUTURAS LÍNEAS DE TRABAJO

4.1. Conclusiones

El objetivo de este Trabajo Fin de Grado era el desarrollo de un sistema de prospección de datos de interés disponibles en internet (en concreto notificaciones de sanciones de tráfico que deben publicar las Administraciones Públicas), que ofreciera la posibilidad al usuario de interactuar con el mismo pudiendo llevar a cabo de forma automatizada, la carga, el procesado y la visualización de los datos. En este caso, los datos cargados se correspondían a sanciones de tráfico, circulación y seguridad vial que deben publicar las Administraciones Públicas, para que los ciudadanos puedan consultarlo en cualquier momento.

Después de haber realizado el diseño y la implementación del sistema ProNAT, Prospector de Notificaciones de Administrativas de Tráfico, se puede concluir con el cumplimiento de los requisitos especificados. El resultado final es la disponibilidad de un sistema con capacidad para la carga de todos los datos publicados diariamente en el portal del Tablón Edictal Único del BOE y para el procesamiento automatizado de todos esos datos aislándose de la complejidad del tratamiento de mucha información y desestructurada. Y el objetivo último, ofrecer al usuario final un entorno analítico que muestre los principales indicadores y estadísticas de interés obtenidas tras el procesamiento masivo de los datos publicados diariamente sobre sanciones de tráfico.

A nivel personal, este trabajo no solo me ha permitido conocer una gran variedad de tecnologías y sus características, sino que me ha brindado la oportunidad seguir todos los pasos para la realización de un proyecto, desde la recogida de requisitos, pasando por el análisis del contexto y las tecnologías, el diseño, la implementación y, por último, las pruebas del sistema.

Además, me ha ofrecido una visión distinta de la cantidad de datos que en la actualidad son accesibles en una plataforma de libre acceso como es el TEU y, sobre todo, el valor de estos. Como se puede ver en este proyecto, a pesar de que esta información

puede encontrarse de manera desestructurada, tras un tratamiento específico, se puede explotar para un análisis por parte de usuarios expertos. Los resultados obtenidos permiten a las empresas focalizar los servicios ofrecidos, conocer en gran medida a sus usuarios actuales y ampliar su nicho de negocio hacía nuevos mercados o valores.

4.2. Futuras líneas de trabajo

A continuación se exponen algunas de las ideas, como trabajo a futuro, para la ampliación de funcionalidades del sistema ProNAT. Estas posibles mejoras se plantean desde dos puntos de vista, desarrollo de nuevas las funcionalidades para el aplicativo y modificación de la arquitectura del sistema.

4.2.1. Otras funcionalidades

Dentro de esta sección se plantean algunas futuras líneas de trabajo:

- Desarrollo de nuevas características de visualización de estadísticas extraídas en el procesamiento de los datos. En este caso, al tratarse de sanciones de tráfico, además de las estadísticas que ya proporciona el sistema (nº de documentos y sanciones procesadas por periodo temporal, distribución de sanciones por tipo de persona física/jurídica, ranking de provincias con mayor número de sanciones, etc.), el sistema puede evolucionar para presentar al usuario otros indicadores. Algunos ejemplos de estos son: el importe medio que paga una persona, la recaudación total de una provincia y/o comunidad autónoma en un periodo de tiempo, la distribución de sanciones según sexo de la persona sancionada, el ranking de provincias con menor número de sanciones, etc.
- Ampliación de las funcionalidades de procesamiento de los datos, añadiendo un nuevo componente al sistema que permita la búsqueda en bases de datos propias más información sobre las personas sancionadas (teléfono, dirección, edad, etc.). Las entidades usuarias de este sistema podrán utilizar esta información para por ejemplo ofrecer productos personalizados a sus clientes.
- Utilización de notificaciones de distinta naturaleza, es decir, ampliación del sistema para que permita la carga de otras notificaciones de interés que se

publican en el Tablón Edictal Único y que no están relacionadas con el tráfico, circulación y seguridad vial.

4.2.2. *Arquitectura del sistema*

Por otro lado, también se pueden planificar las siguientes mejoras desde el punto de vista de la arquitectura:

- Sustitución del Cuadro de Mando desarrollado mediante tecnologías web por una herramienta de Business Intelligence. Este tipo de soluciones supone un coste mayor, y dado que los gráficos e índices representados por el sistema ProNAT no son excesivamente complejos, en un primer momento se descartó su utilización. Si bien es verdad que, cómo se ha expuesto en los puntos anteriores, el análisis de estos datos se podría ampliar para realizar un estudio más detallado. Con el fin de facilitar la representación de las estadísticas más complejas de los datos extraídos del TEU se propone utilizar una de las numerosas herramientas BI que ofrece hoy en día el mercado. Dado que toda la tecnología base utilizada para la implementación del sistema ha sido desarrollada por Microsoft se considera conveniente hacer uso de la herramienta de análisis Power BI. Este framework está diseñado por la misma compañía por lo que facilita su administración e integración con el resto de componentes (Microsoft, 2017).
- Migración del sistema ProNAT desde la arquitectura tradicional implantada On-Premise planteada en este trabajo, a una arquitectura en Cloud utilizando los servicios de AWS, Amazon Web Services. A continuación se expone, a grandes rasgos, una posible implementación junto con los componentes necesarios para llevar a cabo el sistema:
 - Los elementos principales serían Instancias EC2, Elastic Compute Cloud. En estas máquinas de procesamiento se ejecutaría un código, que dependiendo del objetivo con el que se cree, la instancia se encargaría de extraer los documentos del TEU, tratar los datos o generar los resultados. De esta forma se podrá replicar el comportamiento de los distintos motores expuestos anteriormente en este trabajo. Estas tareas no se pueden realizar con una función

Lambda pues, a pesar de que podrían ejecutar el mismo código, se ha estimado que los tiempos necesarios superan los 5 minutos, límite fijado para estas funciones (Amazon Web Services, 2017).

- AWS factura el coste de las instancias EC2, entre otros criterios, en función del tiempo que estas están habilitadas. Para ahorrar los costes que supondría tener activas las instancias cuando no se necesitan, se crearían dos funciones Lambdas. La primera de ellas se encargaría de levantar la instancia EC2 periódicamente y para ello se utilizaría un CloudWatch Event como disparador. La otra función permitiría terminar con la instancia y se invocaría al final del “user data” de la propia EC2. Con estos elementos se reproduciría el comportamiento del Modo Horas que se implementa en la arquitectura On-Premise descrita anteriormente. Para poder reproducir el Modo Rango sería necesario habilitar un disparador, en la función que acaba con la instancia, al igual que el configurado para la otra Lambda. El Modo Flag se podría implementar de dos maneras: configurando otro disparador en la función que levanta la instancia, por ejemplo, un evento en S3, o permitir a los administradores del sistema lanzar manualmente la Lambda (Amazon Web Services, 2017).
- Para poder almacenar toda la información que maneja el sistema, se utilizaría una base de datos RDS. El esquema de esta base de datos se mantendría igual al diseñado y expuesto anteriormente en este trabajo (Amazon Web Services, 2017).
- Aunque existen herramientas en AWS, como QuickSight, que podrían explotar la información almacenada en la base de datos RDS, se propone utilizar en esta modalidad de arquitectura en Cloud, una herramienta de Business Intelligence externa. QuickSight todavía se encuentra en una fase de desarrollo en la que no se implementan todas las funcionalidades que si poseen otras herramientas BI, y que son necesarias para cumplir todos los requisitos del Cuadro de Mando deseado (Pereversina, 2017).
- Para gestionar la seguridad y la administración se crearían Roles IAM que limitarían los permisos de los componentes. Así, por ejemplo, la

instancia EC2 que simularía el Motor de Carga de datos tendría acceso al exterior para poder consultar las notificaciones publicadas en el TEU, en cambio, el resto de instancias no tendrían este permiso (Amazon Web Services, 2017).

En la Figura 4-1 se resume la interacción de los componentes que forman parte de esta solución tecnológica alternativa.

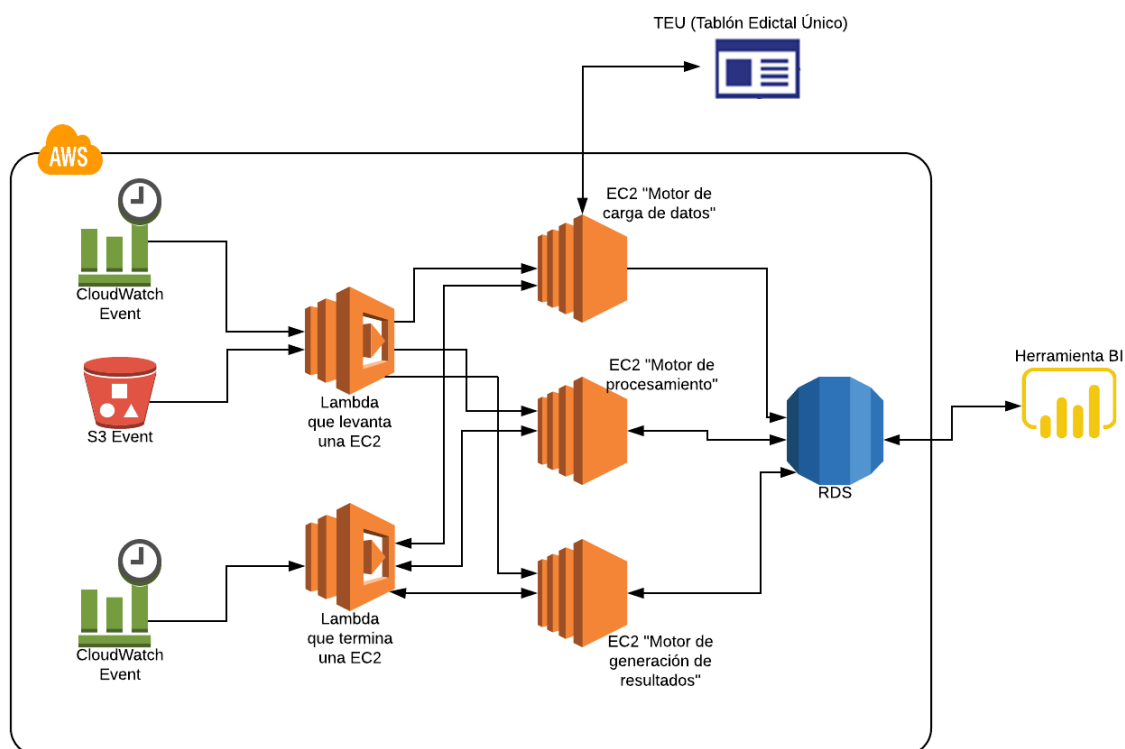


Figura 4-1. Infraestructura AWS del sistema

Para la configuración de las variables del sistema se recomienda permitir el acceso a todos los servicios descritos anteriormente desde la consola de AWS a los usuarios con perfil administrador y añadiendo a estos permisos los necesarios para generar los ficheros con la información almacenada en la base de datos RDS.

Otra recomendación que se puede plantear es la utilización plantillas de Cloud Formation para desplegar la infraestructura necesaria, simplificando así la administración de la arquitectura. Esto facilitaría en gran medida el despliegue tanto de los servicios que permanecen estáticos, Lambdas, Cloud Watch Events, base de datos RDS, etc, como de las distintas instancias EC2 (Amazon Web Services, 2017).

La arquitectura explicada en las líneas anteriores permitiría una reducción considerable de los costes de la infraestructura que es necesario desplegar para implementar el sistema ProNAT. Además, en comparación con el diseño expuesto anteriormente en este trabajo, facilitaría la administración del sistema.

ANEXO

Marco de Regulación Legal

Debido al carácter de los datos procesados y almacenados por el sistema ProNAT es preciso cumplir con la establecido en la Ley Orgánica 15/1999 de Protección de Datos de Carácter Personal. El objetivo de esta es proteger los datos personales, garantizando las libertades y derechos fundamentales de las personas físicas (Ministerio de la Presidencia y para las Administraciones Territoriales, 2011).

Si bien es verdad que se procesa información de carácter personal, la finalidad de este aplicativo es la representación de datos estadísticos previa disociación de los datos personales. En cumplimiento con la legislación, los datos personales que se almacenan serán cancelados cuando hayan dejado de ser necesarios para la finalidad por la cual han sido recabados. Durante el tiempo que los datos se almacenan se ofrece a las personas los derechos de acceso, rectificación y cancelación de estos. La Agencia de Protección de Datos es la encargada de velar por el cumplimiento de la legislación y garantizar a las personas los derechos anteriormente citados (Ministerio de la Presidencia y para las administraciones territoriales, 2008). Siguiendo las directrices impuestas por en la ley se notifica a este organismo público la obtención, tratamiento y finalidad de los datos procesados por el sistema, así como los responsables de seguridad de esta información. Los usuarios del aplicativo que tengan acceso a la información de carácter personal, a través de la funcionalidad del sistema de generación de ficheros, están obligados a cumplir tanto lo expuesto anteriormente, así como los requisitos de seguridad necesarios, evitando la alteración, pérdida, tratamiento o acceso no autorizado a los ficheros generados.

En el artículo 5 de esta ley orgánica anteriormente citada se detalla el derecho de información de la recogida de datos, por lo que es necesario notificar a los interesados el tratamiento y la finalidad de uso de los datos obtenidos. Por otro parte, en el apartado 5 de este mismo artículo, se excluye del cumplimiento de esta regla si el origen de los datos obtenidos es una fuente de acceso público. El sistema descrito en este trabajo cumple este requisito, pues extrae los datos de documentos publicado por distintas Administraciones Públicas en un boletín oficial del estado. Por tanto no será necesario realizar ningún tipo

de notificación a los interesados(Ministerio de la Presidencia y para las Administraciones Territoriales, 2011).

BIBLIOGRAFÍA

- Amazon Web Services. (Noviembre de 2017). *Administración de identidades / IAM / AWS*. Recuperado el Noviembre de 2017, de https://aws.amazon.com/es/iam/?nc2=h_m1
- Amazon Web Services. (Diciembre de 2017). *AWS - Cloud Computing - Servicios de informática en la nube*. Recuperado el Diciembre de 2017, de <https://aws.amazon.com/es/>
- Amazon Web Services. (Diciembre de 2017). *AWS / Elastic compute cloud (EC2) de capacidad modificable en la nube*. Recuperado el Diciembre de 2017, de https://aws.amazon.com/es/ec2/?nc2=h_m1
- Amazon Web Services. (Noviembre de 2017). *AWS / Lambda - Gestión de recursos informáticos*. Recuperado el Diciembre de 2017, de https://aws.amazon.com/es/lambda/?nc2=h_m1
- Amazon Web Services. (Diciembre de 2017). *AWS / Servicio de bases de datos relacionales (RDS)*. Recuperado el Diciembre de 2017, de https://aws.amazon.com/es/rds/?nc2=h_m1
- Amazon Web Services. (Diciembre de 2017). *AWS CloudFormation – Infraestructura como código y aprovisionamiento de recursos de AWS*. Recuperado el Diciembre de 2017, de https://aws.amazon.com/es/cloudformation/?nc2=h_m1
- Gallego, M. (Enero de 2014). *Tema2: Tecnologías de desarrollo web (Desarrollo Aplicaciones Web)*. Recuperado el Abril de 2018, de <https://es.slideshare.net/micaelgallego/tema2-tecnologas-de-desarrollo-de-aplicaciones-web>
- Highcharts. (Septiembre de 2017). Recuperado el Septiembre de 2017, de Highcharts: <https://www.highcharts.com/>
- INE. (Septiembre de 2017). *Listado de nombres más frecuentes en territorio español*. Obtenido de http://www.ine.es/dyngs/INEbase/es/operacion.htm?c=Estadistica_C&cid=1254736177009&menu=resultados&idp=1254734710990
- IngenioVirtual. (Abril de 2015). *Conceptos básicos sobre tecnologías de desarrollo web*. Recuperado el Abril de 2018, de <http://www.ingeniovirtual.com/conceptos-basicos-sobre-tecnologias-de-desarrollo-web/>
- jQuery Foundation. (Agosto de 2017). *jQuery*. Obtenido de <https://jquery.com/>

Microsoft. (Noviembre de 2007). *ASP.NET*. Recuperado el Agosto de 2017, de Microsoft:
[https://msdn.microsoft.com/es-es/library/4w3ex9c2\(v=vs.100\).aspx](https://msdn.microsoft.com/es-es/library/4w3ex9c2(v=vs.100).aspx)

Microsoft. (Septiembre de 2012). *Información general sobre ASP.NET MVC*.
Recuperado el Abril de 2018, de
[https://msdn.microsoft.com/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/library/dd381412(v=vs.108).aspx)

Microsoft. (Agosto de 2017). *Plataforma de datos de Microsoft / Microsoft*. Recuperado
el Agosto de 2017, de <https://www.microsoft.com/es-es/sql-server>

Microsoft. (Diciembre de 2017). *Power BI - Herramientas de BI para la visualización de
datos interactivos*. Obtenido de <https://powerbi.microsoft.com/es-es/>

Microsoft. (Abril de 2017). *Tutorial: Crear un sitio web con sintaxis de Razor en Visual
Studio*. Recuperado el Agosto de 2017, de [https://msdn.microsoft.com/es-es/library/gg606533\(v=vs.100\).aspx](https://msdn.microsoft.com/es-es/library/gg606533(v=vs.100).aspx)

Ministerio de la Presidencia y para las administraciones territoriales. (Noviembre de
2008). *Estatuto de la Agencia de Protección de Datos*. Recuperado el Julio de
2017, de Agencia Estatal, BOE (Boletín Oficial del Estado):
<https://www.boe.es/buscar/pdf/1993/BOE-A-1993-11252-consolidado.pdf>

Ministerio de la Presidencia y para las Administraciones Territoriales. (Marzo de 2011).
*Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter
Personal*. Recuperado el Julio de 2017, de Agencia Estatal, BOE (Boletín Oficial
del Estado): [https://www.boe.es/buscar/pdf/1999/BOE-A-1999-23750-
consolidado.pdf](https://www.boe.es/buscar/pdf/1999/BOE-A-1999-23750-consolidado.pdf)

Ministerio de la Presidencia y para las administraciones territoriales. (Junio de 2015).
TEU. Acceso Administraciones Públicas. Introducción. Recuperado el Julio de
2017, de Agencia Estatal, BOE (Boletín Oficial del Estado):
https://www.boe.es/tablon_edictal_unico/administraciones_publicas/index.php

Ministerio de la Presidencia y para las administraciones territoriales. (Noviembre de
2017). *20171114_pdfs_BOE-N-2017-Comunidad Autónoma BIZKAIA*.
Recuperado el Noviembre de 2017, de Agencia Estatal, BOE (Boletín Oficial del
Estado): [https://www.boe.es/boe_n/dias/2017/11/21/not.php?id=BOE-N-2017-
12142&fix_bug_chrome=foo.pdf](https://www.boe.es/boe_n/dias/2017/11/21/not.php?id=BOE-N-2017-12142&fix_bug_chrome=foo.pdf)

Ministerio de la Presidencia y para las administraciones territoriales. (Noviembre de
2017). *20171121_pdfs_BOE-N-2017-Ayuntamiento de Salamanca-SALAMANCA*. Recuperado el Noviembre de 2017, de Agencia Estatal, BOE
(Boletín Oficial del Estado):

https://www.boe.es/boe_n/dias/2017/11/21/not.php?id=BOE-N-2017-12139&fix_bug_chrome=foo.pdf

- Ministero de la Presidencia y para las administraciones territoriales. (Junio de 2015). *TEU. Búsqueda de notificaciones*. Recuperado el Julio de 2017, de Agencia Estatal, BOE (Boletín Oficial del Estado): https://www.boe.es/tablon_edictal_unico/notificaciones.php
- Mozilla Corporation. (Agosto de 2012). *AJAX*. Recuperado el Agosto de 2017, de <https://developer.mozilla.org/es/docs/Web/Guide/AJAX>
- Mozilla Corporation. (Julio de 2017). *JavaScript*. Recuperado el Agosto de 2017, de <https://developer.mozilla.org/es/docs/Web/JavaScript>
- Otto, Mark; Thornton, Jacob; Bootstrap contributors . (Agosto de 2017). *Bootstrap · The most popular HTML, CSS, and JS library in the world*. Obtenido de <https://getbootstrap.com/>
- Pereversina, J. (Marzo de 2017). *Power BI vs QuickSight - JP Blog*. Recuperado el Diciembre de 2017, de <http://pereversina.com/power-bi-vs-quicksight>
- Sautinsoft. (Febrero de 2017). *SDK to convert PDF to Word, DOCX, RTF, HTML, Excel, Text, XML, Images TIFF for C# and VB.Net*. Recuperado el Agosto de 2017, de <http://sautinsoft.com/products/pdf-focus/index.php>
- W3. (Agosto de 2017). *CSS*. Recuperado el Agosto de 2017, de <https://www.w3.org/Style/CSS/>
- W3. (Agosto de 2017). *HTML 5.2*. Recuperado el Agosto de 2017, de <https://www.w3.org/TR/html5/>