



**Govern  
de les Illes Balears**

Vicepresidència i Conselleria  
d'Innovació, Recerca i Turisme  
Direcció General de Desenvolupament  
Tecnològic

una manera de hacer  
**europa**

Fondo Europeo de  
Desarrollo Regional



Unió Europea

# Análisis funcional SISTRA 2

Diciembre 2017

**Servicios de Administración Electrónica en el Govern de les Illes Balears**

Lot 2 (Servicios electrónicos para la ciudadanía)

**Oficina Técnica de Dirección de Proyecto**

## Control de versiones del documento

Control de Cambios			
Data	Autor	Versión	Cambios
25/10/2017	Indra	v1.0	Versión inicial del análisis funcional de SISTRA 2 (en elaboración)
21/11/2017	Indra	v1.1	Versión inicial del análisis funcional de SISTRA 2 (borrador a falta de validar y definir interfaz visual)
05/12/2017	Indra	v1.2	Versión inicial del análisis funcional de SISTRA 2 (se ha revisado las funcionalidades y queda pendiente de definir interfaz visual)
21/12/2017	Indra	v1.3	Se incluyen los <i>mockups</i> de las principales pantallas de los módulos GTE y HLP.
09/03/2018	Indra	V1.4	Modificación de nomenclatura de módulos y funcionalidades derivadas de revisión por DGD

Revisado por		
Nombre	Data	Área, departamento o empresa

Aprobado por		
Nombre	Data	Área, departamento o empresa

Lista de distribución		
Nombre	Área, departamento o empresa	Correo electrónico




## Índex

am



**Govern  
de les Illes Balears**  
Vicepresidència i Conselleria  
d'Innovació, Recerca i Turisme  
Direcció General de Desenvolupament  
Tecnològic

una manera de hacer  
**europa** 

Fondo Europeo de  
Desarrollo Regional



.....	1
<b>Control de versiones del documento .....</b>	<b>2</b>
<b>1. Introducción .....</b>	<b>5</b>
<b>1.1. Propósito .....</b>	<b>5</b>
<b>1.2. Alcance .....</b>	<b>5</b>
<b>1.3. Glosario de términos .....</b>	<b>5</b>
<b>1.4. Nuevo esquema de Administración Electrónica.....</b>	<b>7</b>
<b>2. Análisis y definición de especificaciones del Sistema .....</b>	<b>8</b>
<b>2.1. Caso de Uso Principal .....</b>	<b>8</b>
<b>2.2. División del Sistema .....</b>	<b>10</b>
<b>2.3. Funcionalidades del sistema .....</b>	<b>11</b>
2.3.1. Gestor de Trámites Electrónicos (SISTRAGES) .....	11
2.3.2. Asistente de Tramitación Electrónica (SISTRAMIT) .....	29
2.3.3. HELPDESK de Tramitación Electrónica (SISTRAHELP) .....	33
<b>2.4. Roles y permisos de acceso .....</b>	<b>34</b>
<b>2.5. Integración con otros sistemas.....</b>	<b>35</b>
2.5.1. Catálogo de procedimientos.....	35
2.5.2. Dominios remotos .....	36
2.5.3. Gestor externo de formularios .....	37
2.5.4. Firma Electrónica .....	38
2.5.5. Gestor pago .....	38
2.5.6. Validación representación .....	40
2.5.7. Registro Electrónico .....	40
2.5.8. Cuadro de mando.....	42
2.5.9. Carpeta ciudadana.....	43
<b>2.6. Interfaz de usuario.....</b>	<b>44</b>
2.6.1. Interfaz de usuario del Gestor de Trámites Electrónicos (SISTRAGES).....	44
2.6.2. Interfaz de usuario del Asistente de Tramitación Electrónica (SISTRAMIT) .....	44
2.6.3. Interfaz de usuario del HELPDESK de Tramitación Electrónica (SISTRAHELP).....	44

## 1. Introducció

### 1.1. Propósito

El objetivo de este documento es definir las funcionalidades que debe implementar el nuevo componente de tramitación electrónica SISTRA2, que debe estar integrado con el resto de componentes de Administración Electrónica (AE) que están desarrollándose tanto por el Govern Balear como por el Ministerio de Hacienda y Administraciones Públicas (MINHAP.).

### 1.2. Alcance

Respecto al anterior producto SISTRA, se limita el alcance funcional de SISTRA2 al ámbito de la tramitación electrónica de una solicitud de forma que una vez registrada dicha solicitud será gestionada por el resto de componentes AE (registro, gestor expedientes, archivo, etc.). SISTRA2 deberá ser multientidad de forma que se integre mediante *plugins* con el resto de componentes AE comentados anteriormente. Respecto a la utilización de los *plugins*, habrá un conjunto de estos que deberán ser definidos a nivel de instalación general de la plataforma, y otros que se definirán a nivel de entidad.

Desde el punto de vista del ciudadano el objetivo del SISTRA2 es ofrecer al ciudadano una nueva forma de interactuar con los servicios electrónicos que mejore su usabilidad, que sea *responsive* y permita tener una visión personalizada según la entidad.

Desde el punto de vista del administrador y desarrollador de trámites el objetivo del SISTRA2 es proporcionar un nuevo entorno de configuración y desarrollo con mejoras funcionales que simplifiquen las tareas de administración y desarrollo de los trámites.

### 1.3. Glosario de términos

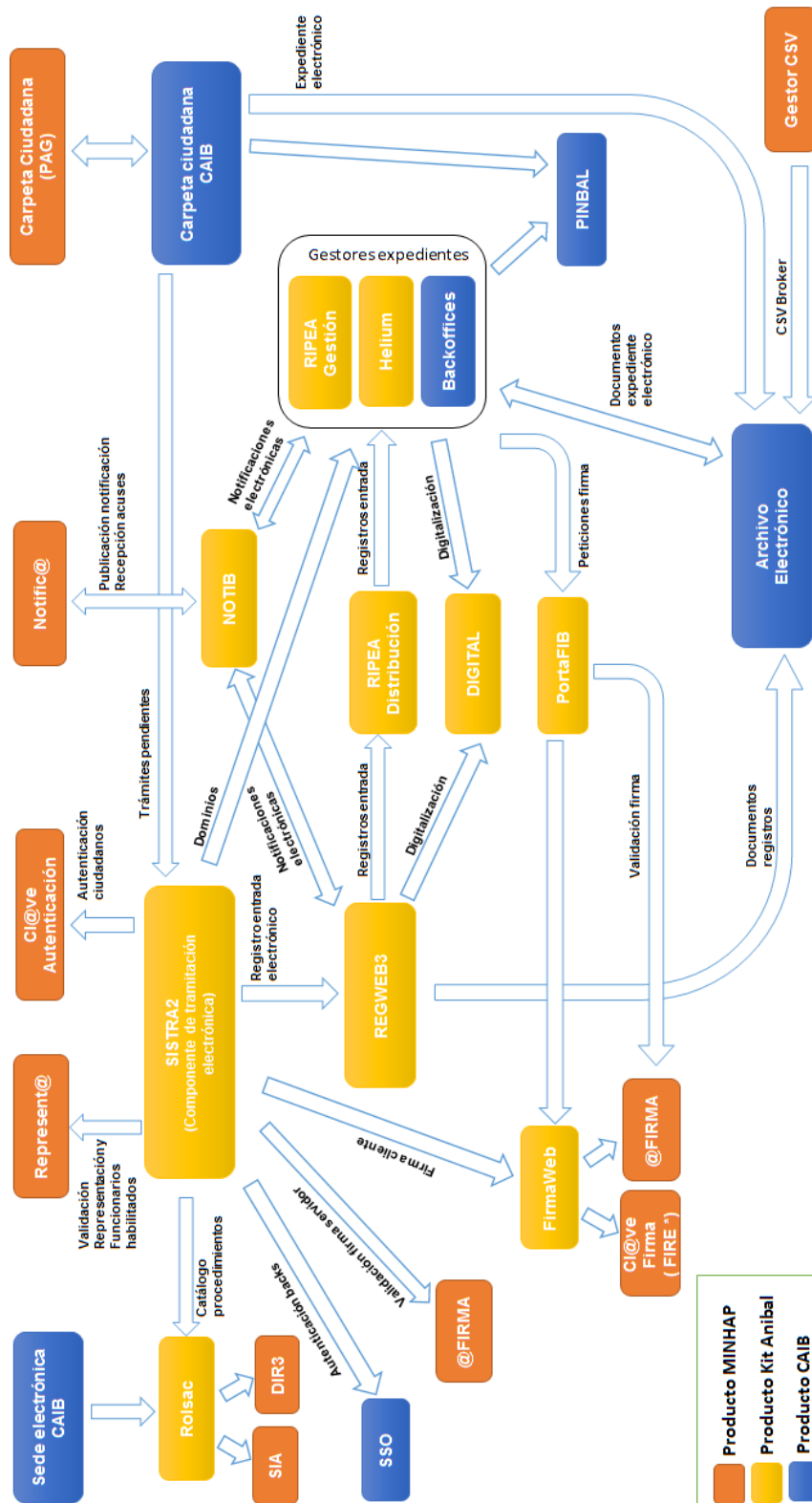
Término	Definición
AE	Administración Electrónica
SISTRA	Plataforma de tramitación electrónica SISTRA que contempla tramitación electrónica, registro, bandeja electrónica, gestión expedientes y carpeta ciudadana
SISTRA2	Componente de tramitación electrónica que contempla la presentación de solicitudes de forma electrónica
SISTRAMIT	Asistente de tramitación electrónica (subsistema de SISTRA2)
SISTRAGES	Gestor de trámites electrónicos (subsistema de SISTRA2)
SISTRAHELP	HELPDESK tramitación (subsistema de SISTRA2)
MINHAP	Ministerio de Hacienda y Administraciones Públicas
CI@ve autenticación	Solución desarrollada por el MINHAP para gestionar la autenticación ciudadana



CI@ve autenticación / FIRE	Solución desarrollada por el MINHAP para gestionar la firma (miniapplet/autofirma/certificados en la nube)
FirmaWeb	Solución desarrollada por la Fundación BIT que permite configurar plugins de firma de forma que aísla a las aplicaciones de la operativa de firma
Kit Anibal	Conjunto de productos open source que pueden ser utilizados por el resto de administraciones públicas.
PluginsIB	Repositorio de Plugins para las aplicaciones del Kit Anibal que permite reutilizar funcionalidades configurables mediante un plugin

## 1.4. Nuevo esquema de Administración Electrónica

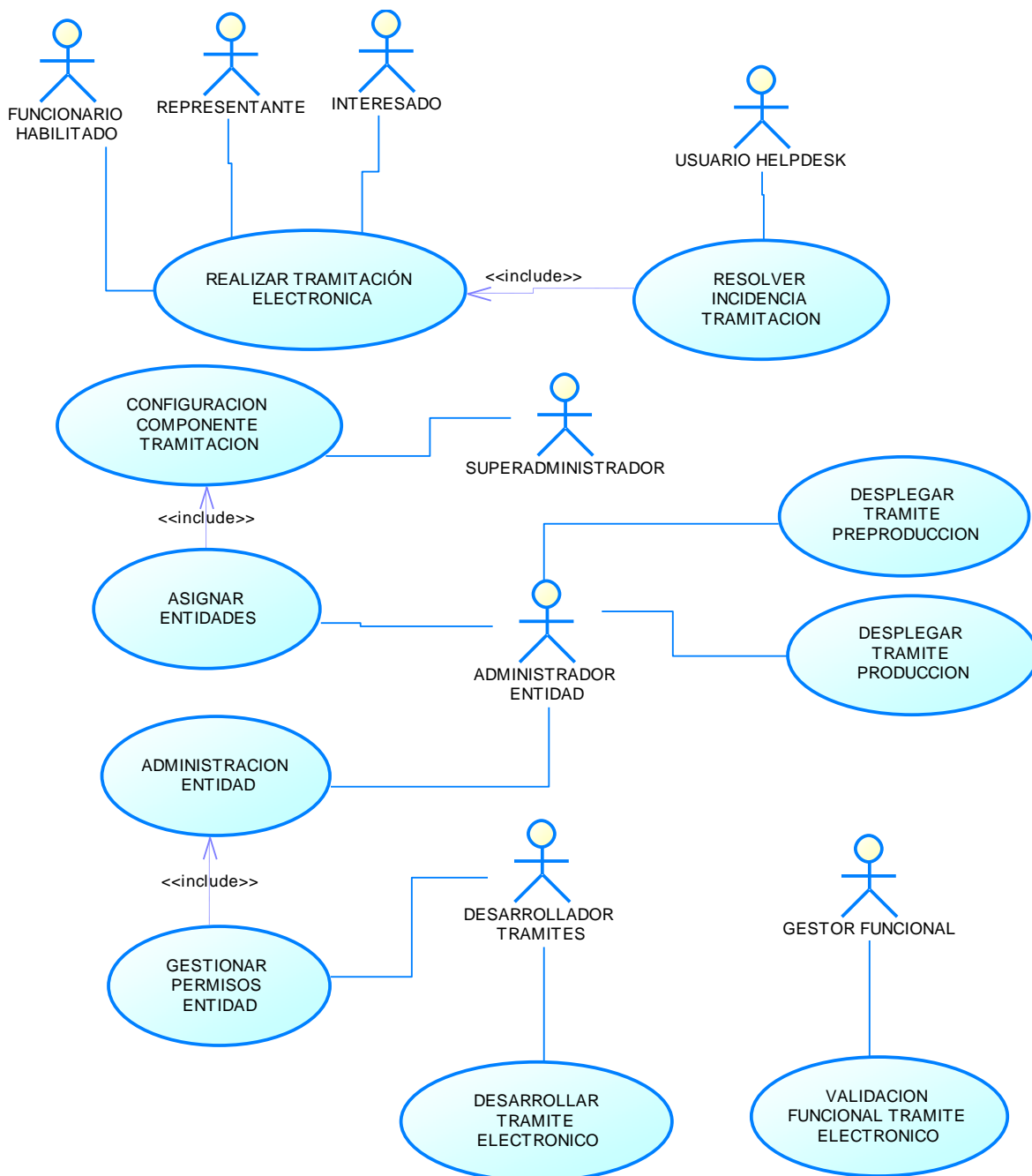
Para aclarar el nuevo alcance de SISTRA2 se presenta en el siguiente diagrama el nuevo esquema de AE definido en el Govern Balear, dónde SISTRA2 se integra con el resto de componentes de AE del MINHAP y propios de la entidad (en este caso el Govern Balear, dónde dispone de elementos del Kit Anibal así como componentes exclusivos del Govern).



## 2. Análisis y definición de especificaciones del Sistema

### 2.1. Caso de Uso Principal

El Caso de Uso Principal de este componente recoge la interacción de actores en el ámbito de desarrollo, pruebas, puesta en producción y explotación de un trámite electrónico.





## Los actores son:

<i>Superadministrador</i>	Usuario encargado de la configuración general del sistema.
Administrador Entidad	Usuario administrador de una Entidad y encargado de desplegar los trámites en los entornos de Preproducción y Producción
Interesado	Usuario que puede realizar una tramitación electrónica
Representante	Usuario que puede realizar una tramitación electrónica en representación del interesado
Funcionario habilitado	Funcionario que puede realizar una tramitación electrónica como si la estuviese realizando el interesado
Usuario HELPDESK	Servicio técnico para resolver incidencias de tramitación
Desarrollador	Usuario que puede desarrollar trámites electrónicos
Gestor Funcional	Gestor del procedimiento que valida funcionalmente el procedimiento y aprueba el pase a producción.

## Los casos de uso y la relación entre los actores es la siguiente:

Realizar Tramitación Electrónica	Proceso de completar un trámite telemático a través del asistente de tramitación
Resolver Incidencia Tramitación	A petición del ciudadano se resuelve una incidencia durante la tramitación electrónica. El usuario de HELPDESK dispone de una herramienta que le permite resolver las incidencias más comunes.
Configurar Componente Tramitación	El <i>superadministrador</i> realiza las configuraciones de administración de la plataforma.
Asignar Entidades	Una de las tareas de configuración que lleva a cabo el <i>superadministrador</i> es nombrar a los administradores de cada entidad.
Administrar Entidad	El administrador de entidad realiza acciones de configuración sobre la entidad.
Gestionar permisos Entidad	Una de las tareas de configuración que lleva a cabo el administrador de entidad es asignar permisos a los desarrolladores de trámites.
Desarrollar Trámite Electrónico	El desarrollador realiza el proceso de desarrollo de un trámite telemático y pruebas unitarias y de integración.
Promocionar a Preproducción	El administrador de entidad realiza la promoción de un trámite del entorno de desarrollo a preproducción para que el gestor funcional del trámite realice las pruebas funcionales
Validación funcional trámite electrónico	El gestor funcional del trámite realiza las pruebas funcionales dando la aprobación funcional para su subida a producción.
Promocionar a Producción	El administrador de entidad realiza la promoción a producción del trámite

## 2.2. División del Sistema

Podemos dividir el componente de tramitación electrónica en los siguientes subsistemas:

- **Gestor de Trámites Electrónicos (SISTRAGES)**  
Este subsistema permite al desarrollador definir trámites y formularios para su posterior ejecución en el asistente de tramitación (SISTRAMIT). Desde el punto de vista del administrador permitirá configurar el componente de tramitación electrónica.
- **Asistente de Tramitación Electrónica (SISTRAMIT)**  
Este subsistema se encarga de mostrar al ciudadano un asistente de tramitación en función de la configuración introducida en el SISTRAGES.
- **HELPDESK Tramitación Electrónica (SISTRAHELP)**  
Permite resolver de una forma rápida los principales problemas técnicos referentes a la tramitación electrónica de los ciudadanos (log de errores, auditoría trámites, problemas con pagos, etc.).

## 2.3. Funcionalidades del sistema

A continuación, se detallan las funcionalidades del sistema divididas por los subsistemas indicados anteriormente.

### 2.3.1. Gestor de Trámites Electrónicos (SISTRAGES)

#### Autenticación

Esta aplicación es considerada como una herramienta de backoffice por lo que deberá estar integrada dentro del sistema de identificación de usuarios del organismo en el que se implante. Además de verificar la autenticación se deberán recuperar los roles asociados al usuario, para gestionar el acceso a las distintas opciones de la aplicación.

Estará preparada para autenticarse vía JAAS

#### Multientidad

El SISTRAGES deberá permitir al *superadministrador* definir varias entidades, de forma que se puedan definir trámites electrónicos para diferentes entidades.

Los trámites electrónicos estarán compartimentados por entidad, de forma que existirá un administrador de entidad que se encargue de administrar los elementos de dicha entidad.

A nivel de entidad se podrá definir una hoja de estilos CSS de forma que se pueda personalizar el aspecto del asistente de tramitación (SISTRAMIT). Los elementos a personalizar se definirán en las funcionalidades específicas del asistente de tramitación (SISTRAMIT).

#### Plugins

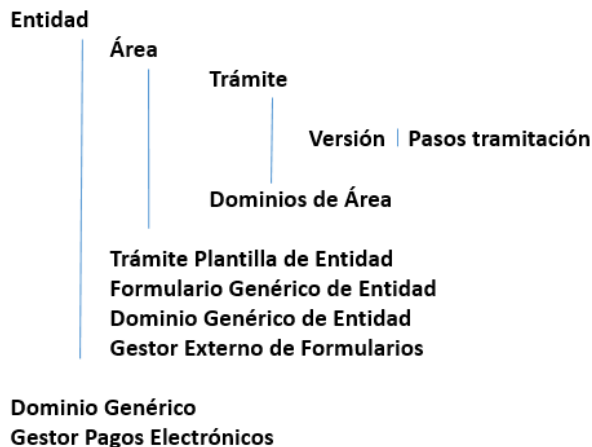
El SISTRAGES deberá permitir la configuración de los *plugins* a utilizar por SISTRA2 para la comunicación con el resto de componentes AE.

El *superadministrador* deberá establecer y configurar un conjunto de *plugins* mínimos a utilizar a nivel de instalación, y posteriormente el *administrador de entidad* establecerá y configurará otra serie de *plugins* que únicamente aplicarán a la entidad administrada.

El mantenimiento y configuración de los *plugins* permitirá fijar la tipología del *plugin*, el nombre y descripción, la clase implementadora y las propiedades de configuración.

#### Estructura elementos

Se distinguen el siguiente árbol de elementos dentro del SISTRAGES:



Entidad	Entidades definidas en el SISTRAGES que pueden tener trámites electrónicos.
Área	Un área permitirá agrupar procedimientos y trámites que guarden alguna relación (temática, orgánica, etc.). Este criterio será potestad del administrador de cada entidad en función de sus necesidades. Las áreas servirán para facilitar la gestión a los desarrolladores y gestores y tendrán un carácter interno.
Trámite	Trámite electrónico definido en el SISTRAGES que puede ser ejecutado en el Asistente de Tramitación (SISTRAMIT). Un trámite electrónico deberá ejecutarse obligatoriamente asociado a un trámite del catálogo de servicios de la entidad. A partir de esta asociación se podrá obtener el procedimiento asociado, y a partir de ahí el código SIA del procedimiento y el código DIR3 del departamento al que pertenece el procedimiento. La asociación de que un trámite del catálogo de procedimientos se ejecute a través de un trámite electrónico del SISTRAGES, se establecerá en el catálogo de procedimientos de la entidad. De esta forma un trámite definido en el SISTRAGES podría dar servicio a varios trámites del catálogo de procedimientos. Por tanto el código de trámite del catálogo de procedimientos deberá ser un parámetro de inicio en la ejecución de un trámite del SISTRAGES en el SISTRAMIT.
Versión	Un trámite podrá tener diferentes versiones, que pueden ser usadas por diferentes motivos (convocatorias distintas, cambios de especificaciones, etc.)
Pasos de tramitación	Una versión de un trámite constará de una serie de pasos de tramitación que determinará el comportamiento del trámite. Se distinguen 2 posibles flujos de pasos de tramitación: <ul style="list-style-type: none"> <li>Flujo normalizado: sería una tramitación clásica con los pasos de rellenar formularios, anexas documentos, pagar y registrar.</li> <li>Flujo personalizado: el desarrollador puede ir especificando los pasos a realizar. Permite establecer un número variable de pasos de tramitación. El flujo entre pasos se establecerá mediante scripts que permitan personalizar la navegación, indicando para un paso cual debe ser el siguiente</li> </ul>
Dominio de área	Dominios definidos en el ámbito de un área. Esta compartimentación permitirá a los usuarios con permisos sobre el área, realizar modificaciones de los dominios definidos.
Trámite Plantilla de Entidad	Trámite reusable dentro de una entidad para diferentes procedimientos (p. ej.: subsanación, aportación documentos).
Formulario Genérico de Entidad	Formulario reusable dentro de una entidad para diferentes trámites.
Dominio Genérico de Entidad	Dominios de carácter general a la entidad.
Gestor externo de formularios	Permite definir gestores de formularios distintos al del GTE. Un gestor de formulario externo tendría sentido si se necesita casuísticas o funcionalidades que no cubran el gestor de formularios del GTE.
Dominio Genérico	Dominio de valores de carácter general a toda la plataforma (países, provincias, municipios, etc.)
Gestor Pagos Electrónicos	Gestor que permite realizar pagos electrónicos.

A nivel de estructura de elementos no aparece el procedimiento ya que, como se explica más adelante, la relación entre el procedimiento y el trámite se establecerá en el catálogo de procedimientos de la entidad (p. ej.: *ROLSAC* para CAIB). Al iniciar un trámite deberá estar parametrizado por el procedimiento al que pertenece, de forma que en tiempo de ejecución en el asistente de tramitación recupere la información asociada al procedimiento (código SIA, código DIR3, plazos presentación, etc.) a través del plugin del catálogo de procedimientos correspondiente. Dentro de los parámetros de inicio se podrá encontrar o

no la versión del trámite, de manera que en caso de no indicarse se ejecutará la última versión (la mayor) activa.

Lo que sí se debe poder saber desde SISTRA2 es en qué procedimientos se está usando el trámite. A nivel de diseño de interfaz visual se establecerá como mostrar esta información.

## Entornos

Desde el punto de vista del ciclo de vida de un trámite electrónico distinguimos 3 entornos diferenciados:

- Desarrollo: entorno dónde se desarrollan los trámites. Para poder promocionar a preproducción, se deberá realizar una exportación del trámite.
- Preproducción: entorno dónde se realizan las validaciones funcionales. No se permitirá modificar un trámite en este entorno, sólo se permitirá importar trámites exportados desde desarrollo. Este entorno no será accesible por desarrolladores, sólo podrán importar trámites *superadministradores* y administradores de entidad y a nivel de área aquellos usuarios que dispongan de los permisos de Alta/Baja y Modificación. Una vez se valide un trámite se podrá exportar a producción.
- Producción: entorno de explotación. No se permitirá modificar un trámite en este entorno, sólo se permitirá importar trámites exportados desde preproducción. Este entorno será accesible por *superadministradores*, administradores de entidad y desarrolladores con permisos de Alta/Baja y Modificación sobre un área, teniendo visibilidad únicamente sobre los elementos de dicha área.

Adicionalmente debe existir un entorno extra para el testeo de nuevas versiones del SISTRAGES, de forma que se puedan probar nuevas funcionalidades sin afectar al desarrollo y explotación de los trámites. Una vez se valide una nueva versión del SISTRAGES se deberá instalar en los entornos de Desarrollo, Preproducción y Producción.

De esta forma, sólo se permitirá realizar modificaciones en la lógica del trámite en el entorno de Desarrollo, de manera que en este entorno se cerrarán las versiones que serán promocionadas a Preproducción y a Producción. En Preproducción y Producción sólo se permitirá modificar las propiedades de acceso al trámite: activar o desactivar el trámite (indicando el mensaje de desactivación), limitar el acceso al trámite, habilitar *debug* en caso de problemas, etc.

La exportación / importación de un trámite entre los distintos entornos será gestionada de forma que contemple toda la información relativa al trámite (formularios, dominios, fuentes de datos, etc.). La exportación de un trámite definirá una “*release*” del mismo, que identifique unívocamente los cambios que incluye dicha versión, la fecha de generación y el entorno desde el que se genera.

A la hora de importar un trámite se deberá verificar que todo esté correcto, como por ejemplo elementos compartidos como dominios de área o genéricos, etc., mostrando para ello al administrador una lista de advertencias con los elementos a revisar.

En el entorno de Desarrollo se podrá modificar completamente el trámite y en los entornos de Preproducción y Producción sólo se permitirá modificar los datos de acceso y plazo, de forma que se asegure que el cambio de lógica sobre un trámite se deba realizar sobre Desarrollo, asegurando la trazabilidad de versiones entre entornos. Adicionalmente en el entorno de Desarrollo se podrá duplicar un trámite, para usarlo como punto de inicio para desarrollar uno nuevo.

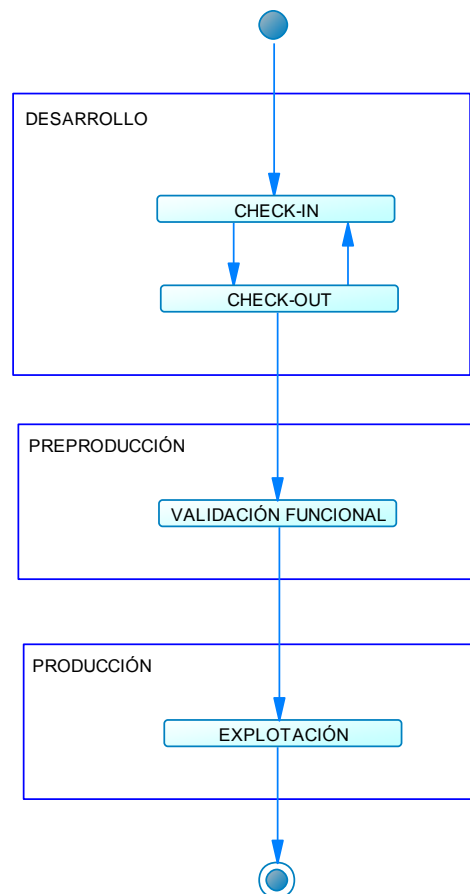
De esta manera se podrá gestionar un versionado y trazabilidad de versiones de trámites promocionados entre entornos, de forma que se puede conocer las *releases*

exportadas/importadas en cada entorno, así como el contenido de los cambios realizados en cada *release* y la fecha de exportación/importación de la *release*.

Se controlará la trazabilidad de las versiones promocionadas para verificar que versión está en cada entorno. Para ello en el entorno de Desarrollo se implementará una funcionalidad de *check-in* / *check-out*:

- **check-in**: habilitar trámite para modificación. Cada vez que se realice un *check-in* se generará un ID de desarrollo nuevo (*release*) para controlar los cambios realizados.
- **check-out**: una vez los cambios se han terminado se realizará el *check-out* para no permitir modificaciones y que se pueda promocionar a preproducción. Al realizar el *check-out* se permitirá al desarrollador introducir una descripción de los cambios realizados para disponer del historial de modificaciones.

Al promocionarse el trámite entre entornos se indicará qué *release* (ID de desarrollo) pertenece de forma que tendremos la trazabilidad de los cambios realizados.



## Dominios

Se permitirá la definición de dominios a 3 niveles:

- Genéricos plataforma: disponibles para todos los trámites (países, provincias, etc.). Mantenidos por el administrador general.
- Genéricos de entidad: disponibles para todos los trámites de una entidad. Mantenidos por el administrador de entidad.
- Genéricos a nivel de área: disponibles para los trámites de un área. Mantenidos por los desarrolladores con permisos de Alta/Baja y Modificación sobre el área.

Para intentar minimizar el uso de dominios remotos, se ofrecerán funcionalidades para definir dominios con sus valores directamente desde el gestor (listas fijas, fuentes de datos, ...).

Distinguiremos los siguientes tipos de dominios:

- Lista fija de valores: definir una lista fija de valores (lista de elementos fija con orden, código, valor).
- Consulta directa a base de datos: permitirá indicar el JNDI de un *datasource* (conexión de Base de datos) y la consulta SQL a realizar.
- Fuente de datos: consulta sobre una fuente de datos. A nivel de área se permitirá definir fuentes de datos, las cuales permiten definir mediante metadatos una tabla con diferentes campos cuyos contenidos pueden ser mantenidos a través del SISTRAGES. En el dominio se indicará la fuente de datos a usar y la consulta a realizar mediante una sentencia *pseudoSQL*. El formato de esta *pseudoSQL* debe ser el siguiente:

```
SELECT campo (, campo)*  
FROM fuenteDatos  
WHERE expresion ( [AND | OR] expresion ] ) *  
ORDER BY campo
```

donde "expresion": *campo* [= | LIKE] *parámetro*, siendo "*parámetro*" el código de parámetro a especificar.

Las partículas de la sentencia *pseudoSQL* (*SELECT*, *FROM*, *WHERE*, *AND*, *OR*, *LIKE*, *ORDER BY*) deben estar en mayúsculas.

Respecto a las comparaciones en el *WHERE* serán *ignorecase* (= y *LIKE*).

Respecto al *LIKE* se realizará automáticamente la comparación: *%parametro%*.

Respecto al *ORDER BY* sólo se puede indicar un campo. Además, como todos los campos son de tipo cadena la ordenación es literal por lo que si se quiere crear un campo en la fuente de datos que permita ordenar secuencialmente, mejor usar letras que números.

- Remoto: consulta remota a un backoffice mediante un *Web Service*. Como resultado de esta consulta remota se podrá obtener un resultado de valores o ficheros. Para realizar esta conexión se indicará la URL de conexión al *Web Service*.



Excepto para los dominios de tipo Lista fija de valores, se establecerán en la configuración del dominio la lista de parámetros a utilizar. De esta forma se tendrán especificados claramente cuáles son los parámetros de un dominio. Para cada parámetro se indicará un código de parámetro. En las consultas de los dominios de tipo Consulta directa a base de datos o Fuente de datos los parámetros de la consulta serán especificados con:

[#<CODIGO\_PARAMETRO>#]

dónde <CODIGO\_PARAMETRO> será el código del parámetro a establecer (p. ej.: *SELECT CAMPO1 FROM TABLA1 WHERE CAMPO2 = '[#<CODIGO\_PARAMETRO\_1>#]'*)

Para minimizar el impacto del uso de dominios remotos y mejorar rendimiento se habilitará un mecanismo de cacheo de dominios:

- Explícito: en la definición de un dominio se indicará si un dominio es *cacheable*. Si es *cacheable* el dominio se cacheará entre distintas peticiones. Después de 24 horas se forzará la renovación del dominio. A nivel de configuración también se ofrecerá la opción de forzar el refresco de un dominio cacheado.
- Implícito: para optimizar la ejecución de scripts todos los dominios se cachearán durante 1 minuto (forzando a renovar tras este tiempo). Esto permitirá que, aunque haya varias consultas seguidas a un dominio, no se penalice el rendimiento.

El acceso a dominios genéricos y dominios de área desde un trámite estará controlado en tiempo de ejecución, de forma que sólo se podrá acceder si se establece por configuración que dicho trámite usa ese dominio. De esta forma se controlarán los dominios usados por cada trámite, facilitando la gestión y promoción entre entornos, así como si se modifica un dominio genérico y de área poder saber el impacto del cambio en los trámites implicados.

Se dispondrá de la funcionalidad de testear el funcionamiento del dominio desde la pantalla de definición del dominio.

## Tràmites

Un trámite electrónico definido en el SISTRAGES permite la presentación electrónica de un trámite definido en el catálogo de servicios (p. ej.: *ROLSAC* para el Govern Balear). Desde el catálogo de servicios (p. ej.: *ROLSAC*) se indicará qué trámite electrónico definido en el SISTRAGES deberá usarse.

Un trámite electrónico definido en el SISTRAGES podría dar soporte a más de un trámite definido en el catálogo de servicios.

Por tanto, para ejecutar un trámite electrónico definido en el SISTRAGES en el Asistente de Tramitación Electrónica se deberá recibir como parámetro de inicio el identificador de trámite del catálogo de servicios (p. ej.: *ROLSAC*). El trámite se “*autoconfigurará*” de forma predeterminada con la información provista del catálogo de servicios (tipo de trámite, plazos presentación, código SIA procedimiento, código DIR3 departamento responsable, órgano instructor, etc.)

## Versiones trámite

Para un determinado trámite se permitirá definir diferentes versiones. A nivel de versión se especificarán las siguientes propiedades:

Versión	Número de versión del trámite
Flujo tramitación	Normalizado o Personalizado. Indicará si se sigue un flujo estándar de pasos o el flujo de pasos a seguir es personalizado para el trámite.
Autenticación	Indica nivel de autenticación soportado (autenticado o sin autenticación). En caso de autenticación se indicará el nivel QAA requerido para ejecutar el trámite.
Idiomas	Indica lista de idiomas soportados para el trámite. De esta forma se controlará que se introducen todas las traducciones necesarias.
Persistencia	Indica si admite persistencia y en caso afirmativo el número de días. Para trámites sin autenticación se ofrecerá una clave de acceso para poder retomar el trámite.
Script personalización	Este script se ejecutará al iniciar un trámite y permitirá particularizar las propiedades configurables por script a nivel de versión de trámite: título y plazo presentación. También se puede emplear este script para establecer una validación de permiso para iniciar un trámite.
Script parámetros iniciales	Permite establecer por script parámetros de inicio del trámite. Este script se ejecutará al iniciar el trámite.
Activo	Permite indicar si el trámite está activo. En caso de no estar activo, permitirá indicar un mensaje informativo multiidioma de la causa. Además se permitirá indicar un plazo de forma programada durante el que el trámite estará desactivado (p. ej.: tareas de mantenimiento sobre el trámite, paradas programadas, etc.)
Habilitar <i>debug</i>	Indica si se habilitan los mensajes de <i>debug</i> sobre el trámite. Por defecto no se generará información de <i>debug</i> pero podría habilitarse en caso de que se requiera solventar alguna incidencia.
Limitar tramitación	Permite limitar los inicios de trámite por intervalo de tiempo. Es útil para evitar una caída del sistema si se pone en marcha un trámite para el que se prevé que exista una fuerte demanda inicial. Se permite indicar el número de tramitaciones máxima por intervalo de tiempo fijado, en minutos.

Para una versión de un trámite se especificará el flujo de tramitación que debe seguir. Un flujo de tramitación está compuesto por pasos predefinidos (rellenar, anexar, etc.).

La herramienta dará soporte a dos tipos de flujos:

- Normal: trámite con los pasos normales de rellenar formularios, anexar documentos, realizar pago y registrar.
- Personalizado: el desarrollador puede ir especificando los pasos a realizar. Permite establecer un número variable de pasos de tramitación. El flujo entre pasos se establecerá mediante scripts que permitan personalizar la navegación, indicando para un paso cual debe ser el siguiente. Existirán 2 scripts:
  - *Script* de acción: Permite establecer que después de un paso se ejecute una acción (invocación a dominios, validaciones, etc.). Esta acción permite definir dentro del flujo una variable con una lista de propiedades. Después de ejecutar la acción se ejecutaría el script de navegación de un paso. La variable estará accesible desde el script de navegación y en los scripts de los siguientes pasos.
  - *Script* de navegación: Permite establecer cuál será el siguiente paso del flujo. Únicamente tendrá la lógica de navegación, no se debe realizar ninguna acción que modifique datos. Si no se especifica este script, el paso será el siguiente.

Debido a este diseño modular de la herramienta, en caso de detectar en un futuro más tipos de flujos o nuevos pasos de tramitación, la herramienta será lo suficiente flexible para poder incorporarlos.

En función del tipo de flujo el gestor de trámites dispondrá de pantallas de configuración específicas que permitirán personalizar el flujo. A nivel de paso de tramitación las pantallas de configuración serán comunes.

#### Pasos de tramitación

Se distinguen los siguientes pasos de tramitación:

- Rellenar
- Capturar
- Anexar
- Pagar
- Registrar
- Información

#### *Rellenar*

En este paso se permite rellenar 1 o n formularios. Una vez están rellenos correctamente todos los formularios se da el paso por finalizado.

Desde el punto de vista de la ejecución, un formulario esta implementado por un gestor de formularios, bien el interno de la plataforma o un gestor de formulario externo.

Como resultado de rellenar un formulario obtendremos un XML con los datos del formulario y un PDF con la vista asociada. La definición de formularios en el gestor de formularios interno y la integración con un formulario externo se especifica más adelante en el punto Gestor de formularios.

Un formulario tendrá las siguientes propiedades:

<b>Identificador</b>	Identificador
<b>Descripción</b>	Descripción
<b>Obligatoriedad</b>	Permite indicar la obligatoriedad: obligatorio, opcional o dependiente (en función script dependencia)
<b>Script establecer dependencia</b>	Este script se cumplimenta en caso de obligatoriedad dependiente, y permite establecer la obligatoriedad de cumplimentación del formulario de manera dinámica.
<b>Firmar digitalmente (Presentación electrónica)</b>	Indica si se debe firmar digitalmente. Por defecto el firmante será el iniciador del trámite, pero se podrá indicar por script quienes deben ser los firmantes.
<b>Script establecer firmantes (Presentación electrónica)</b>	En caso de habilitar la opción de firmar digitalmente el formulario permitirá establecer el firmante o firmantes de manera dinámica.
<b>Presentar en preregistro (Presentación presencial)</b>	Si debe presentar el formulario si el trámite es un preregistro.
<b>Script carga datos iniciales</b>	Mediante este script se podrán establecer datos iniciales del formulario. Este script se ejecutará al iniciar el trámite creando los datos iniciales de todos los formularios.
<b>Script parámetros de apertura de formulario</b>	Permite establecer parámetros cada vez que se acceda al formulario
<b>Script post-guardar formulario</b>	Sirve para: establecer una validación tras guardar, alimentar valores de otros formularios (los formularios quedarán como incorrectos en caso de que estén correctos) y para establecer como incorrectos otros formularios.

#### Anexar

Permite anexas 1 o n documentos. Se permitirá establecer dinámicamente la lista de documentos a anexas a través de un script de personalización, estableciendo dinámicamente las propiedades de cada anexo (descripción, obligatorio (si/no), plantilla, extensiones, etc.). Este script se ejecutará al iniciar este paso para que se tenga en cuenta si existen formularios anteriores que marquen la obligatoriedad de un anexo.

Desde el gestor se permitirá ir definiendo anexos y establecer sus propiedades:

<b>Identificador</b>	Identificador
<b>Descripción</b>	Descripción multiidioma del documento
<b>Obligatoriedad</b>	Permite indicar la obligatoriedad: obligatorio, opcional o dependiente (en función script dependencia)
<b>Script establecer dependencia</b>	Este script se cumplimenta en caso de obligatoriedad dependiente, y permite establecer la obligatoriedad de cumplimentación del documento de manera dinámica.
<b>Ayuda</b>	Se podrá indicar una modalidad de ayuda (fichero, URL o mensaje). En función de la modalidad seleccionada se permitirá indicar un fichero de ayuda, una URL o un texto multiidioma para un mensaje.
<b>Tipo presentación</b>	Indicará si se debe presentar de forma presencial (preregistro) o de forma electrónica.
<b>Número de instancias (Presentación electrónica)</b>	Número máximo de instancias que se pueden anexas de un documento

<b>Extensiones permitidas (Presentación electrónica)</b>	Listado de extensiones permitidas, que pueden ser todas o restringidas a unas concretas.
<b>Tamaño máximo (Presentación electrónica)</b>	Tamaño máximo del anexo
<b>Firmar (Presentación electrónica)</b>	Indica si se debe firmar digitalmente. Por defecto el firmante será el iniciador del trámite, pero se podrá indicar por script quienes deben ser los firmantes.
<b>Anexar firmado (Presentación electrónica)</b>	Indica si el documento deberá ir firmado. Se verificarán que las firmas sean válidas, pero no se validarán quienes deben ser los firmantes.
<b>Script establecer firmantes (Presentación electrónica)</b>	En caso de habilitar la opción de firmar digitalmente el documento permitirá establecer el firmante o firmantes de manera dinámica.
<b>Convertir a PDF (Presentación electrónica)</b>	Indica si el anexo se debe convertir a PDF. Se realizará la conversión a través de un plugin que permita externalizar este servicio (se podrá usar el plugin de conversión de documentos de <i>PluginsIB</i> ).
<b>Script validación (Presentación electrónica)</b>	Permite establecer una validación sobre el documento anexo. En este script estará disponible un plugin que permita acceder a datos de formularios PDF.

### Pagar

Permite la realización de pagos de tasas. Los pagos se realizarán con soluciones externas de pagos (ATIB, TPV Caixa,...) que estarán disponibles a través de diferentes *plugins* de pagos.

<b>Identificador</b>	Identificador
<b>Descripción</b>	Descripción multiidioma del documento de tasa
<b>Obligatoriedad</b>	Permite indicar la obligatoriedad: obligatorio, opcional o dependiente (en función script dependencia)
<b>Script establecer dependencia</b>	Este script se cumplimenta en caso de obligatoriedad dependiente, y permite establecer la obligatoriedad del pago de tasa de manera dinámica.
<b>Plugin pago</b>	<i>Plugin</i> de pago a utilizar (ATIB, TPV, ...)
<b>Script datos pago</b>	Indica los datos del pago a realizar (modelo, identificador de tasa, concepto, fecha devengo, importe, NIF interesado, nombre y órgano emisor).
<b>Tipo de pago</b>	Electrónico y/o Presencial (dependerá del tipo de plugin)
<b>Simular pago</b>	Indica si se ha de simular el pago de la tasa.

## Registrar

Implementa el paso de registrar y mostrar el justificante proporcionado por el Registro Electrónico. Tanto sea un registro como un preregistro, el Registro Electrónico proporcionará un justificante. Una vez realizado el registro se mostrará al ciudadano un enlace para descargar el justificante de registro y adicionalmente se le mostrará la lista de documentos aportados.

En este paso distinguimos tres perfiles:

- el que esta rellenando el trámite (el iniciador del trámite en la plataforma)
- el que lo debe registrar (presentador)
- el representado (en caso de representación, siendo en este caso el presentador el representante).

En este paso se aplicará la validación de representación que se realizará a través de un *plugin*, para poder externalizar esta validación en un sistema externo (en principio será Represent@).

Para la conexión con el Registro Electrónico se utilizará un *plugin* específico.

En este paso se configurarán las siguientes propiedades:

<b>Datos destino registro</b>	Se configuran los datos de registro: oficina, libro y órgano destino.
<b>Script datos destino registro</b>	Permite establecer dinámicamente los parámetros de registro anteriores (oficina, libro y órgano destino). Además permite establecer el número de expediente (sería obligatorio para trámites que no son de inicio).
<b>Instrucciones entrega preregistro (HTML)</b>	Cuando sea un preregistro y haya que entregar documentación presencial pueden establecerse instrucciones de entrega específicas.
<b>Instrucciones fin tramitación (HTML)</b>	Permite personalizar el mensaje de finalización del trámite. Este mensaje aparecerá al finalizar el trámite tras registrar.
<b>Script establecer presentador</b>	Permite indicar quién debe presentar el trámite (NIF, apellidos y nombre). En caso de que no se establezca este script se tomará que quién debe presentar el trámite es quién lo está rellenando (iniciador del trámite). Para trámites sin autenticar será necesario establecer este script.
<b>Admite representación</b>	Indica si el trámite admite representación. En este caso el presentador del trámite se tomará como el representante y el representado será el indicado en el script de referencia de representado.
<b>Script establecer representado</b>	En caso de admitir representación mediante este script se indicará el representado (NIF, apellidos y nombre)
<b>Validar representación</b>	En caso de representación se indica si se debe validar la representación. La validación de la representación se realizará a través de un <i>plugin</i> específico.
<b>Script permite registrar</b>	Permite establecer una validación antes de preparar los datos para registrar. Se ejecutará como precondition antes de mostrar este paso.

## Capturar

Permite obtener datos a partir de un formulario de captura de datos. Este paso se utiliza en el flujo de tramitación personalizado y está concebido para a partir de la obtención de datos en un formulario dirigir el flujo hacia un paso u otro. Este formulario será gestionado siempre por el gestor interno de la plataforma.

Las propiedades de un formulario de captura serán:

<b>Identificador</b>	Identificador
<b>Descripción</b>	Descripción
<b>Script datos iniciales</b>	Mediante este script se podrán establecer datos iniciales del formulario. Este script se ejecutará al iniciar el trámite creando los datos iniciales de todos los formularios.
<b>Script parámetros de apertura formulario</b>	Permite establecer parámetros cada vez que se acceda al formulario

## Información

Permite mostrar información (p. ej.: en un trámite con flujo personalizado puede mostrar el resultado del trámite).

Este paso constará de un script para calcular los datos a mostrar y de una plantilla para visualizar estos datos.

<b>Script datos información</b>	Script que permita obtener dinámicamente los datos a mostrar. Permitirá también recuperar tanto datos como ficheros binarios a partir de invocación de dominios (estos quedarán almacenados en la sesión).
<b>Plantilla visualización</b>	Establece plantilla de visualización de los datos ( <i>Freemaker</i> ), permitiendo realizar el <i>merge</i> con los datos del trámite (datos sesión, formularios, etc.) y los datos obtenidos mediante el script de obtención de datos de este paso.  Desde esta plantilla se permitirá descargar los ficheros obtenidos en el script de datos.

## Formularios

Desde el punto de vista de la ejecución un formulario está implementado por un gestor de formularios, bien el interno de la plataforma o uno externo del organismo.

Como resultado de rellenar un formulario se obtendrá:

- acción: indica que acción se ha realizado con el formulario:
  - Guardar formulario: se ha rellenado el formulario y se ha pulsado a guardar datos
  - Cancelar formulario: se ha cancelado la introducción de datos del formulario
  - Acciones personalizadas: permite establecer una acción personalizada. Las acciones personalizadas se utilizan en los formularios de un paso de captura de datos y permiten poder indicar



opciones distintas al usuario a guardar y cancelar (se usarán en formularios de pasos de tipo Captura)

- datos: XML con los datos del formulario
- PDF: vista PDF del formulario

### Gestor de formularios interno

El Gestor de Trámites ofrecerá un módulo para definir formularios incluyendo secciones y campos, de forma que se pueda ir diseñando visualmente el formulario.

Respecto a la lógica del formulario se diseñará de forma que la ejecución se realice en servidor y se minimicen las llamadas AJAX al servidor, para optimizar los tiempos de respuesta.

Un formulario podrá constar de distintas páginas. Cada página constará de diferentes elementos de formulario que tendrán distintas propiedades a configurar.

Cada formulario constará de unas propiedades de formulario y otras de página. Las propiedades de formulario configurables serán:

- Propiedades de la cabecera: habilitación/deshabilitación de cabecera, inclusión de logo, inclusión de título.
- Operaciones de página: *grid* que permitirá añadir, borrar y reordenar páginas.
- Plantillas de visualización: *grid* de plantillas de visualización del formulario en PDF

Respecto a propiedades de página, se posibilita:

- Fijar scripts de validación de página y navegación
- Marcar indicador de página final

A nivel de componentes, se distinguen los siguientes elementos de formulario:

- Cabecera: Cabecera del formulario
- Sección: Sección donde se introducen campos
- Etiqueta: Implementa los textos que aparecen en el formulario
- Imagen: Imágenes que aparecen en el formulario (logo, etc.)
- Campo de texto: Campo de introducción de datos
- Campo selector: Campo que permite la selección entre un grupo de valores. La selección podrá ser única o múltiple
- *Captcha*: Permite introducir un *captcha*
- Lista de elementos: componente que permite incorporar ítems de campos primarios

Adicionalmente se podrían definir secciones que sean estándar que puedan reutilizarse en los formularios (p. ej.: "sección datos interesado" que contenga los campos de nombre, apellidos y dirección). El objetivo de esta funcionalidad es facilitar la inclusión de algunos bloques



comunes de formularios. Estos bloques podrán ser incorporados por el desarrollador a un formulario y posteriormente adaptados a sus necesidades (modificables).

Para un elemento de tipo Campo a nivel general distinguimos las siguientes propiedades y los siguientes scripts:

Identificador	Identificador del elemento (dentro de la página)
Texto etiqueta	Indica si el campo ha de incluir una etiqueta. En caso afirmativo se permite indicar la etiqueta multiidioma
Obligatorio	Indica si el campo es obligatorio rellenarlo
Columnas ocupadas	Indica el número de columnas ocupadas por el campo la pantalla
Ayuda Online	Ayuda online multiidioma para la cumplimentación del campo
Sólo Lectura	Indica si un campo es sólo lectura
Oculto	Indica si el campo es o no visible. Únicamente aplica a campos de texto.
Tipo de campo	Indica tipo de campo: texto, <i>password</i> , número, etc.
Script autorrellenable	Permite indicar que un campo se rellene a partir de valores de otros campos (los campos de los que dependen deben ser anteriores a este campo).
Script sólo lectura	Permite indicar el estado de un campo (solo lectura y obligatorio) a partir de valores de otros campos (deben ser anteriores a este campo).
Script validación	Permite establecer validaciones personalizadas sobre un campo permitiendo mostrar un mensaje particularizado de error. Este script se ejecutará al modificar el campo y al guardar la página

Según el Tipo de campo, tendrá una serie de propiedades específicas. Inicialmente se distinguen las siguientes clases de campo de formulario:

- Campo de texto: campo para la introducción de texto por parte del usuario. Distinguimos los siguientes subtipos:
  - Normal: introducción de texto plano. Las propiedades que se pueden configurar son: tamaño máximo y si es multilínea (indicando número de líneas en ese caso).
  - Número: introducción de un campo numérico. Las propiedades a configurar son: precisión (número dígitos parte entera y parte decimal), separador miles/decimal, rango (mínimo/máximo) y si permiten números negativos.
  - Identificación: introducción de un campo identificativo: NIF, CIF, NIE y/o NSS.
  - Email: introducción de un campo email.
  - Fecha: campo para la introducción de un campo fecha. El formato del dato será: dd/mm/yyyy.
  - Código postal: introducción de un código postal.
  - Teléfono: introducción de un número de teléfono. Se configura si debe ser fijo y/o móvil.
  - Expresión regular: permite introducir una expresión regular que deberá cumplir el texto introducido.

- Campo selector: campo para que el usuario seleccione valores de una lista.  
La lista de valores posibles a seleccionar se podrá establecer de las siguientes formas:
  - Fija: lista fija de valores.
  - Dominio: lista a partir de un dominio
  - *Script*: permite establecer la lista de valores a partir de un scriptDistinguimos los siguientes subtipos de selector:
  - Selección única: permite elegir una opción de la lista de posibles valores a través de una lista de radios.
  - Selección múltiple: permite elegir una o varias opciones de la lista de posibles valores a través de una lista de *checks*.
  - Lista desplegable: muestra un *combobox* con la lista de opciones para seleccionar un único valor. Existirá la opción de habilitar autofiltro según se escribe.
  - Ventana selección: muestra una ventana pop-up con la lista opciones para seleccionar un valor.
- Campo casilla de verificación: implementa un *checkbox*.
- Botón de acción: Permite establecer la acción a realizar con los datos del formulario (guardar formulario, cancelar formulario o acción personalizada). Las acciones personalizadas están pensadas para formularios que se usan en pasos de tipo Captura, de forma que se le pueda indicar al usuario varias opciones para continuar. En caso de que el formulario no permita acciones personalizadas los botones de acción serán gestionados automáticamente y sólo se permitirá guardar o cancelar formulario. Para formularios que admitan acciones personalizadas permite establecer botones que indiquen la acción a realizar con los datos del formulario.

En un futuro podrían ampliarse con nuevos tipos de campos (mapas, tablas dinámicas, etc.).

Por otro lado, el diseñador de formularios incorporará un botón que permitirá consultar el árbol de elementos del formulario, mostrando la jerarquía formulario/página/campos a nivel de identificadores internos.

### Integración con un gestor de formularios externo

Se permitirán definir gestores de formularios externos a los que se redirigirá para rellenar un formulario. Para la integración con un gestor de formularios externo se definirá un *plugin* específico para gestionar la comunicación.

El gestor de formularios externo deberá implementar el siguiente servicio:

- Iniciar formulario: desde el Asistente de Tramitación (SISTRAMIT) se inicia la acción de mostrar un formulario gestionado por un Gestor Externo y como resultado de dicha invocación, el Gestor de Formularios Externo deberá devolver una URL de acceso al formulario (basada en ticket de acceso). El SISTRAMIT redirigirá el navegador del usuario a esta URL para iniciar el formulario en el gestor de formularios externo. Esta operación tiene los siguientes parámetros:
  - Id: identificador de formulario
  - Datos sesión: contendrá id sesión, idioma, nivel autenticación, datos autenticación,....
  - Datos formulario: será un XML con los datos actuales del formulario
  - URL *Callback*: será la URL de un *Web Service* de *callback* del ATE al que el Gestor de Formularios deberá retornar la respuesta una vez se haya finalizado el formulario.

El Asistente de Tramitación (SISTRAMIT) deberá implementar el servicio de *callback* para que el gestor de formularios devuelva el control a la plataforma:

- Finalizar formulario: desde el gestor de formularios externos se invocará este servicio cuando el usuario haya finalizado su cumplimentación. Como resultado de esta invocación el SISTRAMIT devolverá la URL a la que el gestor de formularios externos deberá redirigir el navegador del usuario para retomar el trámite (basada en ticket de acceso). Los parámetros de esta operación son:
  - acción: indica que acción se ha realizado con el formulario (guardar, cancelar o acción personalizada).
  - Id sesión: identificará la sesión del usuario en el SISTRAMIT.
  - XML formulario: será el XML que contendrá los datos del formulario rellenado.
  - PDF formulario: será la visión formateada del formulario rellenado.

### Ayudas desarrollador

El Gestor de Trámites Electrónico (SISTRAGES) deberá facilitar las tareas de desarrollo de trámites. Para ello se deberán implementar los mecanismos de ayudas necesarios para facilitar estas labores, principalmente distinguiendo:

- Asistente para introducción de script: ayuda que indique para qué sirve el script, qué debe devolver, acceso a los diferentes *plugins* con sus métodos disponibles, campos disponibles en los formularios, acceso a los dominios, etc.
- Ayuda online para cada pantalla, definiendo claramente el uso de cada campo con ejemplos prácticos.

### Configuración del componente

Las propiedades de configuración preferentemente serán establecidas en una pantalla de propiedades, de forma que estén almacenadas en base de datos, en lugar de en fichero de propiedades.

### 2.3.2. Asistente de Tramitación Electrónica (SISTRAMIT)

#### Asistente de tramitación

Al ciudadano se le ofrecerá un asistente de tramitación que le guiará a través de los pasos configurados a través del SISTRAGES. Este asistente deberá cumplir con los requisitos de accesibilidad (WAI-AA).

El enlace para iniciar un trámite vendrá parametrizado por:

- Identificador trámite SISTRAGES: identificador del trámite/versión a ejecutar definido en el SISTRAGES.
- Código trámite catálogo servicios: código de trámite definido en el catálogo de servicios. A partir de este código, se accederá al catálogo de servicios y se autoconfigurarán los plazos de presentación.
- Idioma: idioma de tramitación.
- Parámetros adicionales: otros parámetros adicionales que puedan ser requeridos por la lógica del trámite.

Un trámite estará basado en los pasos definidos en el SISTRAGES, de forma que cada paso tendrá su lógica independiente estando ejecutándose en cada momento el manejador de lógica del paso en cuestión permitiendo el modularidad entre los distintos pasos y optimizando la carga del sistema.

#### Autenticación

La autenticación en el SISTRAMIT vendrá determinada por la configuración del trámite definida en el SISTRAGES. Existen las siguientes opciones:

- No autenticado: se iniciará el trámite sin autenticación.
- Autenticado: se iniciará el trámite autenticándose mediante los sistemas permitidos. Inicialmente estará disponible la autenticación mediante Cl@ve. A nivel de configuración en el trámite se podrá indicar el nivel QAA de autenticación en Cl@ve.

El mecanismo de autenticación será propio del SISTRAMIT y no estará integrado dentro del sistema de autenticación propio del organismo.

#### Auditoría y monitorización errores

Para ofrecer un mejor servicio se ofrecerá una respuesta proactiva ante incidencias en la tramitación, no teniendo que esperar a que el ciudadano se ponga en contacto con el equipo de soporte para tener conocimiento de estos errores.

Para ello se implementarán las siguientes funcionalidades:

- Tipificar errores: revisar los errores que se generan en la tramitación tipificándolos mediante:
  - código de error: identificar unívocamente la causa del error.
  - un nivel de error: identifica el nivel del error:
    - Fatal: Error fatal. No deja continuar el flujo.
    - Error: Error, pero permite continuar el flujo
    - *Warning*: Alerta. Permite continuar el flujo.

- Generar alertas de administración (e-mail): al tener tipificados los errores y categorizados por niveles de error, se pueden crear reglas de aviso para avisar automáticamente de las incidencias a los equipos de soporte y *superadministradores*. De esta forma en cuanto se produzcan problemas significativos en la plataforma se tendrá conocimiento de ellos pudiendo tomar las acciones de corrección oportunas.
- Activación nivel *debug* a nivel de trámite: permitir poder activar el *debug* para un trámite concreto desde el SISTRAGES cuando existan incidencias sobre el mismo por si es necesario investigar las causas del problema.

Las funcionalidades anteriores además de suponer una mejora ante la respuesta de errores supondrán reducir el tamaño de los ficheros de log del servidor de aplicaciones.

#### Personalización diseño según entidad

En el SISTRAGES, el administrador de entidad puede establecer un CSS que personalice el diseño del asistente para adecuarse al estilo propio de la entidad.

Los elementos que podrán personalizarse serán el logo, colores, tipo de letra, cabecera y pie. En la fase de desarrollo se realizará un boceto del asistente y se fijarán los elementos a particularizar.

#### Diseño responsive

El SISTRAMIT tendrá que implementar un diseño *responsive* para ofrecer una correcta visualización del asistente en función del dispositivo. Básicamente se incorporarán 2 posibles visualizaciones: una visualización adaptada para modo escritorio y otra visualización adaptada para modo *tablet*. La visualización en modo móvil no se contempla ya que no es operativo mostrar formularios en este tipo de dispositivos, para estos dispositivos se utilizará el modo de visualización *tablet*.

#### Firma documentos

Para la firma de documentos se contemplarán dos tipos de firmas:

- Firma de documentos PDF: se realizarán mediante formato PAdEs
- Firma de otro tipo de documentos: se realizarán mediante formato establecido por configuración (XAeS, CAeS...)

Para los formularios se firmará el PDF (el XML no) y para los anexos se firmará según el tipo de documento (si es PDF o no).

La firma de documentos en cliente (navegador) se externalizará mediante el componente FirmaWeb. Este componente deberá proveer el sellado de las firmas.

Desde el punto de vista de servidor, en principio no se requerirá firma, tan sólo la verificación de las firmas y extracción de los datos del firmante para verificar la identidad del mismo.

### Recarga automática tras error

El comportamiento del SISTRAMIT ante un error que implique no dejar continuar, forzará la recarga del trámite para permitir al ciudadano continuar con la tramitación sin tener que ir a su carpeta personal para retomar el trámite.

### Almacenamiento temporal

Una vez registrado el trámite se purgarán los datos del mismo (formularios, anexos, pagos, etc.) ya que toda esa información se habrá trasladado al Registro Electrónico, que será el encargado de trasladar la información al archivo documental y al gestor de expedientes correspondiente.

Después de purgar estos datos se mantendrá la información de auditoría del trámite durante un tiempo por si es necesario conocer las operaciones y errores de una sesión de tramitación. El tiempo de almacenamiento de esta auditoría será configurable en las propiedades del sistema.

### Funcionario habilitado

Desde el punto de vista de realizar una tramitación electrónica surge un nuevo actor que es el funcionario habilitado.

Si un interesado no dispone de los medios electrónicos necesarios, su identificación o firma electrónica en el procedimiento administrativo podrá ser válidamente realizada por un funcionario público mediante el uso del sistema de firma electrónica del que esté dotado para ello. Para ello previamente el ciudadano debe identificarse ante el funcionario y consentir expresamente su identificación y autenticación por el funcionario público habilitado.

Una vez el ciudadano dé su consentimiento, el funcionario habilitado podría realizar la tramitación electrónica en su nombre.

La validación de si un funcionario es un funcionario habilitado se realizaría a través del plugin de representación.

### Generación estadísticas y auditoría

A la hora de generar estadísticas se utilizará el código procedimiento y código de trámite del catálogo de servicios para poder unificar los resultados.

Los eventos que se auditarán serán:

- Inicio de trámite
- Carga desde persistencia
- Pago electrónico
- Registro trámite
- Fin trámite
- Borrado de trámite
- Error
- *Debug* tramitación

Para un evento se almacenará:

- Código entidad: código de la entidad (DIR3 o código interno)

- Código procedimiento / Código trámite: código procedimiento, código SIA y código trámite del catálogo de servicios.
- Identificador trámite SISTRAGES: identificador trámite en el SISTRAGES (código y versión del trámite).
- Identificación tramitador: NIF/NIE/CIF
- Resultado: resultado del evento, si ha ido bien o mal.
- Detalles: dependiendo del evento se pueden registrar más detalles en caso necesario, p.ej. para un inicio de trámite se puede recoger: idioma tramitación, sistema operativo 8sistema operativo y versión), navegador (navegador utilizado y versión), etc.
- Fecha: fecha en la que se ha producido el evento.



### 2.3.3. HELPDESK de Tramitación Electrónica (SISTRAHELP)

#### Autenticación

Esta aplicación es considerada como una herramienta de *backoffice* por lo que deberá estar integrada dentro del sistema de identificación de usuarios del organismo en el que se implante. Además de verificar la autenticación se deberán recuperar los roles asociados al usuario, para gestionar el acceso a los trámites según las áreas a las que se hayan dado acceso.

#### Resolución de incidencias

Este módulo permite resolver de una forma rápida los principales problemas técnicos referentes a la tramitación electrónica de los ciudadanos. Está pensado para que sea usado por el personal de soporte de incidencias.

Básicamente este módulo será un frontal web que dará acceso controlado a la información sobre los trámites que se están realizando en un momento dado, de forma que se podrán realizar averiguaciones sobre el estado de un determinado trámite.

Las funcionalidades a las que dará soporte este módulo serán:

- Auditoría de trámites: con esta opción se podrán consultar los eventos almacenados pudiendo así determinar las acciones realizadas por un usuario o sobre un trámite en concreto.

El filtro de búsqueda será el siguiente:

- Id. Sesión: identificadora sesión de tramitación
- NIF: NIF del tramitador
- Identificador trámite: identificador del trámite del SISTRAGES (código trámite / versión)
- Rango horario
- Eventos: lista de eventos a buscar. Los eventos auditados son los indicados en el subsistema del Asistente de Tramitación Electrónica (SISTRAMIT) en el punto "Generación estadísticas y auditoría".
- Persistencia: con esta opción podremos consultar los datos de persistencia de los trámites en curso, permitiendo consultar los formularios y documentos anexados para verificar si existe algún tipo de problema. Además, permitirá consultar estados de trámite especiales a tener en cuenta, como trámites que tienen el pago realizado, pero no están finalizados.
- Estadísticas: con esta opción se permitirá obtener unos datos básicos estadísticos sobre la tramitación diaria en curso (registros correctos, registros incorrectos, trámites finalizados, pagos correctos, pagos incorrectos)

Este módulo minimizará las consultas a sistemas para la obtención de logs, consultas directas a base de datos, etc. que se vienen efectuando para la resolución de incidencias técnicas.

## 2.4. Roles y permisos de acceso

Existirán los siguientes roles a nivel general:

- **Superadministrador:** permitirá el acceso a todas las opciones, aunque básicamente sus tareas se centrarán en:
  - Configuración de los parámetros globales de la Plataforma de Tramitación.
  - Creación de Entidades indicando su nombre, descripción y el rol de administrador de la entidad.
- **Administrador entidad:** permitirá el acceso a todas las opciones de una entidad. Sus tareas se centrarán en:
  - Configurar los parámetros de la Entidad (estilos entidad, dominios genéricos entidad, etc.)
  - Creación de Áreas de la Entidad y definir los roles de acceso al Área. El administrador podrá definir los roles que considere necesarios a los que podrá asignar la combinación de los siguientes permisos:
    - Alta: permitirá realizar el alta y baja de trámites
    - Modificación: permitirá modificar trámites
    - Consulta: permitirá consultar trámites
    - HELPDESK: permitirá dar soporte en el HELPDESK a trámites realizados de esta área.
  - P. ej.: definir role "STR\_AREA1\_DEV" con los permisos de Alta y Modificación
  - Promocionar los trámites desde el entorno de Desarrollo a Preproducción y posteriormente a Producción, mediante la exportación/importación de trámites.
  - En el entorno de Preproducción/Producción modificar propiedades de acceso al trámite: activar o desactivar el trámite (indicando el mensaje de desactivación), limitar el acceso al trámite, habilitar *debug* en caso de problemas, etc.
- **Desarrollador de trámites:** acceso a todos trámites de una entidad o trámites determinados según los roles que tenga asignados.
- **Operador HELPDESK:** acceso al HELPDESK para resolver incidencias de las áreas según los roles que tenga asignados.

La modificación de trámites sólo será permitida en el entorno de Desarrollo, por lo que el rol de Desarrollador de trámites, sólo se permitirá en dicho entorno. En los entornos de Preproducción y Producción sólo accederán el *Superadministrador* y Administrador de Entidad.

## 2.5. Integración con otros sistemas

A continuación, se identifican los sistemas con los que se deberá integrar SISTRA2 que se realizarán a través del *plugin* correspondiente, para poder personalizar el funcionamiento según el organismo.

Como se ha apuntado en puntos previos de este documento, habrá un conjunto de *plugins* que se definirán y configurarán a nivel general de instalación de la aplicación, y otros que se definirán a nivel de entidad y por tanto solo aplicarán a esta.

A priori, los *plugins* a definir a nivel global y configurados por el superadministrador son:

- *Plugin* de login.
- *Plugin* de representación.
- *Plugin* de dominios remotos

A nivel de entidad, el administrador de entidad podrá configurar los *plugins* siguientes:

- *Plugin* de catálogo de procedimientos.
- *Plugin* de firma.
- *Plugin* de pagos
- *Plugin* de gestor de formularios externos
- *Plugin* de registro

Respecto a la definición de los *plugins* se analizará en la fase de desarrollo el uso del API definido para *PluginsIB* para ver cómo aprovechar la reutilización de componentes (p. ej.: *PluginsIB* ya dispone de un API de verificación de firma, pagos, etc.).

### 2.5.1. Catálogo de procedimientos

En el catálogo de procedimientos del organismo se enlazarán los trámites de los procedimientos a un trámite electrónico del Gestor de Trámites Electrónico (SISTRAGES). Se accederá al catálogo de servicios a través del *plugin* que esté establecido para la entidad.

En tiempo de ejecución del trámite en el Asistente de Tramitación Electrónico (SISTRAMIT) será necesario conocer los datos del trámite almacenados en el catálogo de procedimientos. Para ello se dispondrá de un *plugin* que permita acceder al catálogo de procedimientos para recuperar los siguientes datos de un trámite:

- Parámetros entrada:
  - Código del trámite en el catálogo de procedimientos
- Respuesta:
  - Código DIR3 entidad raíz
  - Código DIR3 entidad responsable
  - Identificador SIA procedimiento
  - Descripción procedimiento
  - Plazo presentación (fecha inicio y fecha fin)
  - URL de acceso a la consulta del trámite en el catálogo de servicios

Desde el punto de vista del Gestor de Trámites Electrónico (SISTRAGES) este *plugin* tendrá una función para conocer en que trámites del catálogo de procedimientos se está utilizando. Para ello el *plugin* puede disponer de una función para realizar esta consulta:

- Parámetros entrada:
  - Identificador del trámite electrónico en el SISTRAGES (código trámite y versión)
- Respuesta:
  - Lista de trámites del catálogo en los que se utiliza:
    - Identificador procedimiento catálogo
    - Descripción procedimiento catálogo
    - Identificador trámite catálogo
    - Descripción trámite catálogo
    - URL de acceso a la consulta del trámite en el catálogo de servicios

### 2.5.2. Dominios remotos

Para gestionar consultas remotas a un *backoffice* se realizarán a través de un *plugin* específico. Como resultado de esta consulta remota se podrá obtener un resultado de valores o ficheros.

Este *plugin* se implementará mediante un *Web Service*, en el que se fijará un contrato de servicio. De esta forma al definir un dominio se le deberá especificar la URL de acceso al *Web Service*.

Para ello el *plugin* puede disponer de una función para realizar esta consulta:

- Parámetros entrada:
  - Identificador del dominio: permitiría que un mismo *Web Service* resolviera distintos dominios.
  - Parámetros dominio: lista de tuplas código parámetro / valor parámetro. Si se requiere pasar un fichero como parámetro, se dispondrá de una función para poder convertir un fichero a Base64 de forma que pueda ser pasado como parámetro.
- Respuesta:
  - Error: indica si ha habido error.
  - Código error: indica código del error (código error en sistema remoto).
  - Descripción error: indica la descripción del error.
  - Registros: lista de registros con el resultado de la consulta (formato filas/columnas).
  - Archivos: lista de ficheros retornados. Cada fichero tendrá asociado un id de fichero y el nombre del fichero (con extensión).

Como se ha comentado en la funcionalidad de dominios, existirá la funcionalidad de poder testear el funcionamiento del dominio en la pantalla de definición del dominio.

### 2.5.3. Gestor externo de formularios

Para formularios cuya casuística no sea posible gestionar con el gestor de formularios ofrecido por SISTRA2, se contempla el uso de gestores externos de formularios, dónde se externalizará la operativa de rellenar el formulario y se obtendrá como resultado el formulario completado.

Este *plugin* se implementará mediante un *Web Service*, en el que se fijará un contrato de servicio. De esta forma al definir un gestor de formularios externo se le deberá especificar la URL de acceso al *Web Service*.

Para ello el *plugin* puede disponer de las siguientes funciones para interactuar un gestor de formularios externo:

- Iniciar sesión formulario: SISTRA2 invocaría a esta función para iniciar una sesión de rellenar formulario en el gestor de formularios externo.
  - Parámetros entrada:
    - Identificador del gestor de formularios: permitiría que un mismo *Web Service* diese cobertura a diferentes gestores de formularios.
    - Identificador del formulario: identificador del formulario a rellenar.
    - Datos actuales formulario: indica los datos actuales del formulario (XML).
    - Datos sesión tramitación: indica diferentes datos de la sesión actual de tramitación:
      - Identificador sesión tramitación: id sesión tramitación.
      - Identificador trámite: id y versión trámite.
      - Datos usuario autenticado
      - Idioma: idioma de tramitación.
      - Paso: identificador del paso actual en el que se encuentra el trámite y el formulario a rellenar.
      - Parámetros dominio: lista de tuplas código parámetro / valor parámetro
    - URL *callback* para retorno de datos: indicaría la URL de *callback* hacia SISTRA2 a la que el gestor de formularios deberá invocar tras finalizar la cumplimentación del formulario. En esta URL de *callback* se pasará como parámetro por POST el campo de ticket de sesión generado como resultado de esta llamada.
  - Respuesta:
    - Ticket de sesión formulario: indicaría la sesión de rellenar el formulario. Será necesaria para el retorno del gestor del formulario hacia SISTRA2.
    - URL redirección al gestor de formularios: retorna la URL a la que hay que redirigir el navegador del usuario para ir al gestor de formularios.
- Recuperar resultado formulario: Una vez se ha completado el formulario en el gestor de formularios externo, se invoca a la URL de Callback indicada en el inicio de forma que

SISTRA2 debe recoger el resultado del formulario y continuar con la tramitación en el asistente.

- Parámetros entrada:
  - Ticket de sesión formulario: identificaría la sesión de rellenar el formulario.
- Respuesta:
  - Cancelado: indica si se ha completado el formulario o se ha cancelado.
  - Datos formulario: datos rellenados del formulario (XML).
  - PDF formulario: PDF con la visualización formateada del formulario.

El formato de XML de los datos de formulario intercambiado entre SISTRA2 y el gestor de formularios externo tendrá un formato definido por SISTRA2, de forma que sea compatible este formato con el generado por el gestor de formularios interno de SISTRA2.

#### 2.5.4. Firma Electrónica

Desde el punto de vista de SISTRA2 será necesario validar firmas electrónicas, identificando el firmante del documento. Se deberá contemplar el uso de varios formatos de firma, en el que se incluirá PAdES para los documentos PDF y al menos otro formato de firma para otro tipo de documentos (CAAdES, XAdES, ...).

Esta validación de firmas será necesaria en la parte de servidor. No se prevé que desde SISTRA2 se deban realizar firmas, solo validarlas.

Para ello el *plugin* puede disponer de una función para realizar esta validación:

- Parámetros entrada:
  - Firma: firma realizada.
  - Formato: formato de la firma.
  - Documento original: documento sobre el que se realiza la firma. Necesario para firmas “detached”.
- Respuesta:
  - Validación: indica si se ha podido realizar el proceso de validación.
  - Correcta: indica si la firma es correcta.
  - Datos firmante: indica los datos del firmante.
  - Datos representado: en caso de representación, indica los datos del representado.

#### 2.5.5. Gestor pago

SISTRA2 deberá ser compatible con diversos gestores de pago (inicialmente la ATIB y TPV de La Caixa).

Este *plugin* se implementará mediante un *Web Service*, en el que se fijará un contrato de servicio. De esta forma al definir un gestor de pago se le deberá especificar la URL de acceso al *Web Service*.

Al disponer de una interfaz *Web Service* los gestores de pago podrán ser reutilizados por otras aplicaciones distintas a SISTRA2. En el diseño de interfaz se deberá tener en cuenta a nivel de diseño que el gestor de pago deberá poderse integrar tanto con SISTRA2 como con otras aplicaciones, de forma que se definan qué elementos podrán particularizarse a nivel de estilos.

El desarrollo de la integración con un gestor de pago, supondrá el desarrollo de una aplicación que contenga:

- *Web Service* para integración con la aplicación que quiera realizar el pago. Controlará el flujo de información y redirecciones entre la aplicación que quiere realizar el pago (p. ej.:SISTRA2) y el gestor de pagos.
- Frontal para realizar el proceso de pago.
- Backoffice en caso de que se requiera realizar algún tipo de administración sobre el gestor de pago (p. ej.: actualmente existe un backoffice para el pago de TPV de La Caixa).

Para ello el *plugin* puede disponer de las siguientes funciones:

- *iniciarSesionPago*: permitiría iniciar una sesión de pagos
  - Parámetros entrada:
    - Datos del pago: tipo de pagos permitidos (tarjeta, cargo en cuenta o presencial), modelo, tasa, importe, fecha devengo, datos sujeto pasivo y fecha límite para iniciar pago (opcional).
    - Parámetros adicionales: dependerán del gestor de pago (p. ej.: para TPV órgano emisor).
    - URL *callback* para retorno de datos: indicaría la URL de *callback* de la aplicación que invoca (p. ej.: SISTRA2) a la que el gestor de pagos deberá invocar tras finalizar la cumplimentación del pago. En esta URL de *callback* se pasará como parámetro por POST el campo de ticket de sesión de pago generado como resultado de esta llamada.
    -
  - Respuesta:
    - Ticket de sesión pago: indicaría la sesión de pago. Será necesaria para el retorno del gestor del formulario hacia aplicación que invoca (p. ej.: SISTRA2).
    - URL redirección al gestor de formularios: retorna la URL a la que hay que redirigir el navegador del usuario para ir al gestor de formularios.
- *reanudarSesionPago*: permitiría reanudar una sesión de pagos ya iniciada
  - Parámetros entrada:
    - Ticket de sesión pago: indicaría la sesión de pago. Será necesaria para el retorno del gestor del formulario hacia aplicación que invoca (p. ej.: SISTRA2).
  - Respuesta:
    - URL redirección al gestor de formularios: retorna la URL a la que hay que redirigir el navegador del usuario para ir al gestor de formularios.



- *comprobarEstadoSesionPago*: permitiría comprobar el estado de una sesión de pagos
  - Parámetros entrada:
    - Ticket de sesión pago: indicaría la sesión de pago.
  - Respuesta:
    - Estado: estado del pago (No existe / Pago en curso / Pago pendiente confirmar / Pago confirmado / Excedido tiempo de pago)
    - Descripción estado: mensaje descriptivo del estado del pago
    - Tipo de pago: en caso de que el pago se haya realizado, indica el tipo de pago (tarjeta, cargo en cuenta o presencial).
    - Identificador de pago: identificador del pago en el Gestor de pago (p. ej.: NRC).
    - Fecha de pago: fecha en la que se ha realizado el pago.
- *calcularImporteTasa*: permitiría calcular el importe de una tasa
  - Parámetros entrada:
    - Id tasa: indicaría el identificador de la tasa.
    - Parámetros consulta: lista de tuplas código/valor para especificar parámetros. Dependerán del gestor de pagos correspondiente.
  - Respuesta:
    - Importe de la tasa

#### 2.5.6. Validación representación

Para la validación de la representación se desarrollará un plugin que permita verificar si un representante tiene poder de representación para un representado para cierto procedimiento. Este plugin también se empleará para la validación de representación de funcionarios habilitados.

La implementación de este plugin se realizará con Represent@, aunque al estar basado en un plugin se podría personalizar según la instalación de SISTRA2.

Para ello el *plugin* puede disponer de una función para validar la representación:

- Parámetros entrada:
  - Colectivos: diferentes colectivos asociados (@podera, habilit@, ...)
  - NIF representante: documento identificativo del representante.
  - NIF representado: documento identificativo del representado.
  - Código SIA: código SIA del procedimiento.
- Respuesta:
  - Permitido: indica si puede ejercer la representación o no.

#### 2.5.7. Registro Electrónico

Para los trámites que tengan configurados el paso de registro se deberán registrar las solicitudes a través de un *plugin* que permita conectar con el registro del organismo. Se accederá al registro a través del *plugin* que esté establecido para la entidad.



El Asistente de Tramitación (SISTRAMIT) detectará en tiempo de ejecución según la lógica configurada en el trámite si debe realizar un registro de entrada o bien un preregistro. Un preregistro indica que el ciudadano aporta una serie de documentación electrónicamente, pero para confirmar la solicitud debe aportar presencialmente alguna documentación.

Para ello el *plugin* puede disponer de las siguientes funciones:

- *obtenerOficinasRegistroEntrada*: permitiría recuperar las oficinas de registro de entrada
  - Parámetros entrada:
    - Entidad: código DIR3 de la entidad raíz
  - Resultado:
    - Oficinas de registro disponibles (oficina y libro)
- *obtenerUnidadesDestinatarias*: permitiría recuperar las unidades destinatarias
  - Parámetros entrada:
    - Entidad: código DIR3 de la entidad raíz
  - Resultado:
    - Unidades destinatarias (en forma de árbol identificando al padre)
- *obtenerTiposAsunto*: permitiría recuperar los tipos de asunto
  - Parámetros entrada:
    - Entidad: código DIR3 de la entidad raíz
  - Resultado:
    - Tipos de asunto
- *registroEntrada*: permitiría realizar un registro de entrada de una solicitud
  - Parámetros entrada:
    - Entidad: código DIR3 de la entidad raíz
    - Oficina registro: oficina y libro
    - Datos asunto: extracto, tipo asunto, idioma,...
    - Datos interesado: NIF, nombre, apellidos,...
    - Datos representante: NIF, nombre, apellidos,...
    - Documentación anexada electrónicamente: lista de documentos que se han aportado electrónicamente
  - Resultado:
    - Número y fecha de registro
- *consultarJustificanteRegistroEntrada*: permitiría consultar el justificante de un registro de entrada
  - Parámetros entrada:
    - Entidad: código DIR3 de la entidad raíz
    - Número registro: número de registro de entrada
    - Fecha de registro: fecha de registro de entrada
  - Resultado:
    - Justificante de registro firmado por registro
- *preregistroEntrada*: permitiría realizar un preregistro de una solicitud

- Parámetros entrada:
  - Entidad: código DIR3 de la entidad raíz
  - Oficina registro: oficina y libro
  - Datos asunto: extracto, tipo asunto, idioma,...
  - Datos interesado: NIF, nombre, apellidos,...
  - Datos representante: NIF, nombre, apellidos,...
  - Documentación anexada electrónicamente: lista de documentos que se han aportado electrónicamente
  - Documentación a aportar presencialmente: lista de documentos que se han de aportar presencialmente, instrucciones de presentación y fecha límite de presentación.
- Resultado:
  - Número y fecha de preregistro
- *consultarJustificantePreregistro*: permitiría consultar el justificante de un preregistro
  - Parámetros entrada:
    - Entidad: código DIR3 de la entidad raíz
    - Número preregistro: número de preregistro
    - Fecha de preregistro: fecha de preregistro
  - Resultado:
    - Justificante de preregistro firmado por registro

#### 2.5.8. Cuadro de mando

La explotación estadística de la tramitación electrónica se externalizará fuera de SISTRA2 de forma que pueda ser tratada por bien por herramientas comerciales de BI (*Business Intelligence*) como por desarrollos realizados a medida (p. ej.: el anterior AUDITA de SISTRA).

Para ello se implementará un *Web Service* que permita extraer diariamente la información estadística y que sea tratada por el sistema externo de cuadro de mando.

Hay que definir la información estadística que será necesaria para la generación de este cuadro de mando:

- Entidad
- Tipo de evento: inicio trámite, registro trámite, preregistro trámite, fin trámite, pago, carga de trámite desde carpeta,...
- Procedimiento
- Trámite
- Fecha de evento
- Usuario
- Resultado evento (depende evento)
- Detalle evento (depende evento: p. ej.: para inicio de trámite el S.O. y navegador)

Hay que indicar que la información estadística de tramitación en SISTRA2 se irá purgando de forma periódica, por lo que los sistemas de cuadro de mando solo tendrán unos días para obtener la información de un determinado día. Lo normal será que diariamente estos

sistemas externos extraigan la información de SISTRA2 mediante el *Web Service* y consoliden los datos estadísticos en su sistema.

Para ello este *Web Service* puede disponer de una función para obtener los datos de tramitación diarios:

- Parámetros entrada:
  - Fecha: fecha de consulta
- Resultado:
  - Lista de eventos de tramitación con la información asociada

### 2.5.9. Carpeta ciudadana

Desde la carpeta ciudadana de la entidad se debe conocer que trámites tiene iniciados un ciudadano que están pendientes de finalizar.

Para ello se implementará un *Web Service* que permita desde la carpeta ciudadana conocer estos trámites pendientes.

Para ello este *Web Service* puede disponer de una función para obtener los trámites pendientes de un usuario:

- Parámetros entrada:
  - NIF: NIF ciudadano
- Resultado:
  - Lista de trámites pendientes

## **2.6. Interfaz de usuario**

Para este componente deberá definirse una interfaz de usuario por cada subsistema:

- SISTRAGES: Gestor de Trámites Electrónicos
- SISTRAMIT: Asistente de Tramitación Electrónica
- SISTRAHELP: HELPDESK de Tramitación Electrónica

Además, deberá definirse el diseño de los gestores de pago, teniendo en cuenta que su aspecto tendrá que ser adaptable no solo a SISTRA2, sino a otras aplicaciones.

### **2.6.1. Interfaz de usuario del Gestor de Trámites Electrónicos (SISTRAGES)**

En el documento *DGDT\_MockPantallasSISTRA2.pdf* se presentan *mockups* de las pantallas principales del módulo.

### **2.6.2. Interfaz de usuario del Asistente de Tramitación Electrónica (SISTRAMIT)**

A nivel de operativa el asistente de tramitación será similar al existente en la plataforma SISTRA actual, dividido en los diferentes pasos, etc.

En la fase de desarrollo se realizarán bocetos del asistente tanto en la versión de escritorio como en la versión para dispositivos móviles para validar el diseño.

### **2.6.3. Interfaz de usuario del HELPDESK de Tramitación Electrónica (SISTRAHELP)**

En el documento *DGDT\_MockPantallasSISTRA2.pdf* se presentan *mockups* de las pantallas principales del módulo.

