



CONGRESO
DE INTENDENTES

Proyecto UE - CI

Gestión Integral de Residuos y Economía Circular Fortalecimiento Institucional y Buenas Prácticas

SISTEMA DE INFORMACIÓN GESTIÓN INTEGRAL DE RESIDUOS DE INTENDENCIAS

REQUERIMIENTOS PARA EL DESARROLLO DE SOFTWARE

JULIO 2024

Ing. Gabriel Cebreiros
Especialista IT
Congreso de Intendentes



Contenido

1. Introducción.....	1
2. Requerimientos Funcionales	1
2.1. Indicadores.....	1
2.2. Ratios para Nivel de Desempeño	3
2.3. Usuarios y Roles	4
2.4. Frontend para Ingreso de Datos	4
2.5. Backend de Administración.....	4
3. Requerimientos Técnicos	5
3.1. Plan de Desarrollo.....	5
3.2. Interfaz Web y Móvil.....	5
3.3. Plataforma Tecnológica.....	6
3.4. Arquitectura de Software.....	6
3.5. Seguridad e Interoperabilidad	6
3.6. Derechos de Propiedad y Código Fuente	8
3.7. Implantación y Capacitación	8
3.8. Soporte y Mantenimiento	8
3.9. Performance y Calidad del Software.....	8
4. Referencias.....	9



SISTEMA DE INFORMACIÓN GESTIÓN INTEGRAL DE RESIDUOS DE INTENDENCIAS

Desarrollo de un Sistema de Información y Monitoreo (SIM) con datos desagregados por Departamentos (Intendencia) para la gestión integral de residuos sólidos urbanos de competencia de Intendencias según la Ley 19.829 de Gestión Integral de Residuos y según los lineamientos del Plan Nacional de Gestión de Residuos (PNGR).

REQUERIMIENTOS PARA EL DESARROLLO DE SOFTWARE

1. Introducción

El objetivo de este documento es presentar una serie de requerimientos funcionales y técnicos a tener en cuenta para la elaboración de la propuesta técnica y comercial del proyecto de solución de software.

El Sistema de Información a construir es una aplicación web que permite gestionar y monitorear un conjunto de indicadores para la Gestión Integral de Residuos de las Intendencias.

2. Requerimientos Funcionales

Se realiza una breve descripción funcional del software. Se brindará oportunamente documentación adicional para completar el análisis funcional del sistema. (4) (5)

2.1. Indicadores

En la primera fase del proyecto se requiere la implementación de diez (10) Bloques de Información con un total de veintiocho (28) Indicadores básicos. Esto es común a todas las Intendencias. Cada uno de estos indicadores se componen de una serie de parámetros y una formula de cálculo. El valor de cada parámetro deberá ser ingresado por los usuarios de las distintas Intendencias de forma periódica. Quince (15) indicadores son de tipo contestación inmediata (Si / No / Número) y doce (12) del tipo obtención de datos donde se ingresa algún dato o parámetro como por ejemplo las toneladas transportadas o generadas, y requieren que esta interfase de ingreso sea intuitiva y sencilla para los usuarios. A futuro se incorporarán e instrumentarán nuevos bloques de información con nuevos indicadores o nuevos indicadores sobre los bloques de información existentes. Por lo tanto el sistema debe admitir la ampliación e incorporación de nuevos bloques de información e indicadores. A continuación se muestra el esquema de los bloques de información con los indicadores.



ESQUEMA DE INDICADORES BÁSICOS

- **Indicador 1. Generación de residuos sólidos municipales per cápita y totales.**
 - I.1.1. Por tipo de residuo
 - I.1.2. Por zona de origen
- **Indicador 2. Tasa de recolección de residuos sólidos municipales.**
 - I.2.1. Por tipo de residuo
 - I.2.2. Por zona de origen
 - I.2.3. Por forma de colecta
- **Indicador 3. Uso y destino final de residuos.**
 - I.3.1. A relleno sanitario
 - I.3.2. A disposición inadecuada
 - I.3.3. Reciclados en tratamiento
 - I.3.4. Tratados en compostaje
 - I.3.5. Energía desde residuos
- **Indicador 4. Eficiencia en la gestión de residuos.**
 - I.4.1. Grados de ocupación de instalaciones
 - I.4.2. Rendimientos del trabajo y equipos
- **Indicador 5. Legislación, Institucional y Planificación.**
- **Indicador 6. Sociales.**
 - I.5.1. Inclusión social.
 - I.5.2. Capacitación, educación y formación.
- **Indicador 7. Clima y ambiente.**
- **Indicador 8. Economía y finanzas.**
 - I.8.1. Tarifas (SÍ-NO)
 - I.8.2. Ingresos y costes
 - I.8.3. Capacidad de pago
- **Indicador 9. Usuarios y operadores.**
 - I.9.1. Tasa de reclamos
 - I.9.2. Tasa de incidentes
 - I.9.2. Organización de empresas
- **Indicador 10. Operación y mantenimiento.**



A modo de ejemplo se muestran los parámetros considerando el bloque de información del Indicador 1:

Indicador 1. Generación de residuos sólidos municipales per cápita y totales.

TEMA / UNIDAD:	Generación de residuos per cápita
USOS	Planificación-Urbanización (construcción)- Gestión (operación y mantenimiento)
Código / Abreviatura	GenPC
Periodicidad:	Anual
Definición:	Es la producción promedio de kilogramos por habitante al año de residuos sólidos departamentales
Ecuación de cálculo:	$GenPC = TGR / PT$
Referencia de cálculo:	GenPC: Generación Per Cápita de Residuos Sólidos Departamentales
	TGR: Total de Residuos Sólidos Departamentales Generados
	PT: Población Total
Residuos incluidos:	Según Ley N° 19.829 de 18/09/2019
Residuos excluidos:	Según Ley N° 19.829 de 18/09/2019
Período de referencia:	Anual, registro mensual en la base de datos. Hay que revisar las variaciones estacionales, semanales y festivas.
Nivel geográfico de reporte:	Municipal
	Departamental
Origen de la información:	Por Intendencias. TGR desde encuestas y modelo estadístico. PT desde Censo.
Pertinencia (Relevancia):	Este indicador se emplea comúnmente como parámetro para el diseño de sistemas de manejo de residuos sólidos departamentales por parte de las autoridades locales. Es transversal en su aplicación, ya que permite calcular otros indicadores a partir de la determinación de la generación total de residuos sólidos departamentales

Desde él se puede calcular la Generación total simplemente (TGR).

También se puede dividir su cálculo en:

I.1.1. Por tipo de residuo (orgánico, papel, vidrio, ...)

En el caso de conocer o asumir los porcentajes se puede fácilmente calcular cada parte de ellos.

I.1.2. Por zona de origen

Para los diferentes municipios o áreas urbanas si se disponen datos y mapas.

2.2. Ratios para Nivel de Desempeño

La aplicación debe permitir definir ratios para evaluar el nivel de desempeño de cada uno de los indicadores presentados en el apartado anterior. Por ejemplo como se muestra en la imagen a continuación:

	VERDE	AMARILLO	ROJO
Generación Per Cápita de Residuos Sólidos Departamentales	< XX kg/Hab-año	X kg/Hab-año - X kg/Hab-año	> XX kg/Hab-año



Estos ratios deben desplegarse en el dashboard para conocer el nivel de desempeño pautado de cada indicador. (5)

2.3. Usuarios y Roles

La aplicación debe permitir la gestión de usuarios y roles. Deberá utilizar un esquema de autorización basado en roles (RBAC). Usuarios por cada Intendencia con distintos niveles de permisos con acceso al sistema y a determinadas funcionalidades. Para ello deberán existir roles predefinidos que habiliten ciertas funcionalidades. Por otro lado deben existir usuarios de administración para gestionar por completo el sistema tanto la creación de usuarios como indicadores y parámetros.

La aplicación deberá permitir la autenticación con Usuario Gub.uy, o en su defecto la configuración de una política de usuarios y contraseñas. Dicha política debe poder soportar funciones básicas ABM y bloqueo de usuarios, así como caducidad, longitud y complejidad de contraseñas. Los usuarios deben poder cambiar sus contraseñas.

Cierre de sesión: Se deben suministrar opciones de desconexión o cierre de sesión (logout) que permitan terminar completamente con la sesión o conexión asociada, las cuales deben encontrarse disponibles en todas las páginas protegidas por autenticación.

2.4. Frontend para Ingreso de Datos

Aplicación web donde accedan los usuarios finales de cada Intendencia para ingresar los valores de los parámetros que van a utilizarse para el cálculo de los indicadores. Deberá desarrollarse con una interfase gráfica sencilla, intuitiva y de fácil uso. Debe incluirse alguna consulta de ingreso y podría requerirse un dashboard que muestre algunos de los indicadores con el objetivo de monitorear.

2.5. Backend de Administración

Acceso a usuarios administrativos donde pueden configurar, controlar y gestionar los distintos parámetros, ratios, indicadores y usuarios. Dashboard configurable para control y monitoreo completo de los indicadores ya sea por Intendencia o global. Debe permitir realizar consultas de todas las entidades y exportar los datos.

3. Requerimientos Técnicos

Se detallan una serie de exigencias técnicas que se deben cumplir en el desarrollo de sistemas de software para organismos estatales y son fundamentales para asegurar la calidad del diseño, seguridad y robustez.

3.1. Plan de Desarrollo

Se deberá confeccionar y presentar un plan detallado del proyecto de desarrollo que describa la solución completa, con la estimación de tiempos y costos para las distintas etapas del ciclo de vida del desarrollo. Se debe incluir en el plan los distintos entregables que deberán ser validados por el Congreso de Intendentes. Es deseable que el tiempo máximo para la realización del proyecto no supere los cinco (5) meses.

3.2. Interfaz Web y Móvil

La interfaz de la aplicación debe ser 100% Web y funcional en Dispositivos Móviles. Todas las funcionalidades de la aplicación deben poder accederse y ejecutarse completamente desde un navegador web y en un dispositivo móvil.

Independencia de dispositivos. Debe soportar la navegación web al menos desde los siguientes navegadores: Firefox, Internet Explorer, Chrome, Safari, sin necesidad de plugin.

Móvil. Se deben seguir las “Buenas prácticas para dispositivos móviles” de la W3C. Debe pasar la validación automática utilizando herramientas provistas por W3C: <http://validator.w3.org/mobile/>

Pautas de diseño - Usabilidad. Se deben seguir las pautas de diseño establecidas en la “Guía para diseño e implementación de Portales Estatales” en http://www.agesic.gub.uy/innovaportal/v/548/1/agesic/guia_para_diseno_e_implementacion_de_portales_estatales.html.

Estándares Web. Debe pasar validación automática utilizando herramientas gratuitas provistas por W3C: <http://validator.w3.org/> y <http://jigsaw.w3.org/css-validator/>

Accesibilidad: nivel AA de WCAG 2.0

Accesibilidad: buenas prácticas de tipo AAA de WCAG 2.0 (2) (3)



3.3. Plataforma Tecnológica

La aplicación debe funcionar sobre una plataforma tecnológica de base Open Source y en lo posible gratuita sin costos de licenciamiento (sistema operativo, base de datos, servidor de aplicaciones y servidor web).

3.4. Arquitectura de Software

El sistema debe poder funcionar en esquemas de alta disponibilidad y funciones de balanceo de carga, explicitando la forma en que se logran estos requerimientos y cuales componentes de hardware y software están involucrados.

El sistema debe poder permitir configuraciones escalables. En particular, clúster y granja o aportar soluciones alternativas que permitan la escalabilidad.

En caso de extender las funcionalidades o cambiar las versiones se deberá asegurar la estabilidad de la solución. Es recomendable especificar detalladamente las consideraciones de diseño y los mecanismos empleados para su cumplimiento.

Es recomendable que los sistemas cuenten con herramientas de monitoreo que permitan evaluar la salud de la aplicación, detectando condiciones anómalas de funcionamiento de la aplicación y generando alertas. Éstas herramientas deberán ser complementarias y adicionales a las ofrecidas por el sistema operativo, el motor de base de datos y/o servidor de aplicaciones.

3.5. Seguridad e Interoperabilidad

La aplicación deberá publicarse en la red privada de organismos del estado REDuy. El sistema deberá cumplir con los protocolos de seguridad de REDuy en los puntos de comunicación con la misma.

Si fuera necesario publicarse, habilitar acceso o consumir servicios en Internet deberá utilizar canales de comunicación seguros entre módulos y/o componentes externos. Se debe garantizar la confidencialidad e integridad en la transmisión de datos entre componentes (Decreto 92/014). (3)

Proteger los sistemas expuestos a Internet. La aplicación deberá funcionar correctamente cuando ésta se encuentre publicando servicios a través de un WAF (Web Application Firewall) implementado típicamente sobre apache/mod_security con las reglas de filtrado provistas por la comunidad OWASP (OWASP Core Ruleset) y en modalidad bloqueante. (3)

El sistema deberá contar con mecanismos de auditoría para todas las transacciones llevadas a cabo por el mismo. Se debe poder establecer las operaciones a auditar, e información a registrar para las mismas, incluyendo si fueron exitosas o fallidas. Configurar conforme necesidades específicas los registros que se desean auditar. Se debe poder establecer la marca de tiempo para los registros de



auditoría de todas las operaciones. Los registros de logs siempre deben tener marca de tiempo para poder correlacionar datos. Se debe poder establecer para cada registro de auditoría la identificación unívoca del usuario que realiza las operaciones en los logs de auditoría; además la dirección IP y puerto de origen. Poder identificar el usuario que efectuó la operación. Poder establecer los tiempos de rotación y retención de los registros de auditoría.

Se deberá indicar cuál es la seguridad implementada a nivel de logs de auditoría. Se debe saber que tanto se cuida la integridad de los logs, en particular para saber qué tan modificables son y si queda registro de dichas modificaciones.

Se debe cumplir con las buenas prácticas estipuladas en el top 10 de OWASP. Es una buena práctica para asegurar el desarrollo de sistemas, en particular los web, y disminuir las vulnerabilidades inherentes a todas las aplicaciones.

Ante incidentes de seguridad informática que sufran los sistemas de información, se debe solicitar su notificación inmediata por parte del proveedor al Congreso de Intendentes. Quedando en manos del Congreso de Intendentes la notificación al CERTuy, según lo establece el decreto 451/009. (2)

Manejo de información sensible propia del sistema: Se debe garantizar que el software desarrollado no divulgue información sensible en respuestas de error, incluyendo detalles del sistema, identificadores de sesión o información de las cuentas de usuarios; así mismo, deben implementar mensajes de error genéricos. No se debe mostrar información innecesaria, como por ejemplo, en los encabezados de respuesta que se refieran a los sistemas operativos y versiones del software utilizado. (2)

El sistema deberá contar con las herramientas de seguridad para garantizar la confidencialidad, integridad y autenticidad de los datos. Se deberá explicar cómo se implementan los aspectos antes mencionados a nivel de operación y a nivel de administración. Conocer como el sistema almacena, transmite y manipula los datos garantizando la confidencialidad, integridad y autenticidad.

Integración con sistemas transaccionales internos y externos a través de Web Services y/o sistemas de mensajería. Los Web Services que involucren datos asociados a Personas, Direcciones y a los modelos de referencia publicados al momento de comenzar la implementación, deben estar diseñados siguiendo las especificaciones y XML Schemas disponibles por AGESIC según los modelos de metadatos correspondientes descritos en el Catálogo de Metadatos del Estado. (3)

3.6. Derechos de Propiedad y Código Fuente

El sistema desarrollado y otros productos entregados serán propiedad exclusiva del Congreso Intendentes y del Proyecto UE-CI, pudiendo éstos disponer libremente de su uso. La empresa no podrá utilizarlos o divulgar los mismos a terceros, salvo autorización por escrito.

El código fuente debe formar parte de la solución y debe ser un entregable más que deberá contar con un repositorio adecuado para mantenerse y actualizarse.

3.7. Implantación y Capacitación

La implantación se llevara a cabo en la infraestructura definida y acordada con el Congreso de Intendentes y del Proyecto UE-CI. Deberá presentarse un plan de implantación y capacitación con todas las Intendencias involucradas en el proyecto y con un cronograma para la puesta en marcha de la solución.

3.8. Soporte y Mantenimiento

Se deberá presentar las opciones de soporte y un plan de mantenimiento continuo con sus respectivos costos de toda la solución, tanto para la plataforma tecnológica si así fuera el caso, pero principalmente para toda la solución de software y sus componentes. El mantenimiento debe contemplar la evolución y así como también la actualización de todos los componentes de software de la solución.

3.9. Performance y Calidad del Software

Se valorará que se pueda comprobar el cumplimiento de atributos de calidad del software tales como:

- Funcionalidad a través del uso de herramientas de testeo que permitan automatizar las pruebas y/o verificar qué porcentaje del código está cubierto por las pruebas.
- Calidad arquitectónica del código a través del uso de herramientas de análisis estático del código, medida de la complejidad, análisis de dependencias, detección de código duplicado. (3)

4. Referencias

(1) Descripción y Guías de Uso de la Plataforma de Gobierno Electrónico (PGE) del Estado Uruguayo

<http://www.agesic.gub.uy/innovaportal/file/1295/1/>

[descripcion_y_guias_de_uso_de_la_plataforma_de_gobierno_electronico_del_estado_uruguayo.pdf](#)

(2) Web AGESIC

<https://www.gub.uy/agencia-gobierno-electronico-sociedad-informacion-conocimiento/>

(3) Requerimientos Técnicos para sistemas de Organismos y Agencias del Estado

<https://www.gub.uy/agencia-gobierno-electronico-sociedad-informacion-conocimiento/comunicacion/publicaciones>

(4) Sistema De Información De Gestión Integral De Residuos Sólidos Urbanos, Fase De Inicio (Primera Fase De Implementación)

Proyecto UE - CI - Gestión Integral de Residuos y Economía Circular Fortalecimiento Institucional y Buenas Prácticas

(5) Planilla Indicador Reducido V3

Proyecto UE - CI - Gestión Integral de Residuos y Economía Circular Fortalecimiento Institucional y Buenas Prácticas