<TITULO>

Análisis estático de código detallado

**{{ project\_name }}**

**Mes - año**

La información contenida en este documento es confidencial y propiedad de Sogeti.  
Copyright© 2020 Sogeti. Todos los derechos reservados.

La información relativa a los conceptos, ideas, metodologías y otros materiales utilizados están estrictamente limitados para ser utilizados por los colaboradores de Renfe, y únicamente para la consideración de la propuesta y la evaluación de las competencias y capacidades de Sogeti, sobre la colaboración que pueda dar Renfe dentro de este marco y presente propuesta de colaboración.

La copia total o parcial de los materiales que se incluyen en la presente propuesta no podrán ser facilitadas a otras entidades además de Renfe, sin la previa autorización de Sogeti.

Las referencias a Clientes o a otras entidades presentadas en la siguiente propuesta no deberán ser utilizadas sin la previa autorización de Sogeti.

**Sogeti España, S.L.**

Inscrita en el Registro Mercantil de Madrid, folio 1/M del tomo 16.838 del archivo. Hoja M-287.781 – N.I.F.: B-08377715

Sede social: **Calle Puerto de Somport, 9 - Edificio Oxxeo - 28050 Madrid**

Tel.: +34 91 657 70 00 l Fax.: +34 91 657 20 19 l www.es.Sogeti.com

Índice

[1. Introducción 4](#_Toc46235340)

[1.1. Objetivo 4](#_Toc46235341)

[1.2. Alcance 4](#_Toc46235342)

[1.2.1. Fuera de alcance 4](#_Toc46235343)

[1.3. Prerrequisitos y Condiciones 4](#_Toc46235344)

[1.4. Modelo de Calidad Aplicado 4](#_Toc46235345)

[1.5. Perfiles Implicados 6](#_Toc46235346)

[2. Resultados 7](#_Toc46235347)

[2.1. Características Generales de la Aplicación 7](#_Toc46235348)

[2.2. Calidad General de Código 7](#_Toc46235349)

[2.3. Incidencias Detectadas 8](#_Toc46235350)

[2.3.1. Incidencias clasificadas por prioridad 8](#_Toc46235351)

[2.3.2. Incidencias por característica de calidad 8](#_Toc46235352)

[2.4. Lista detallada de defectos 9](#_Toc46235353)

[3. Investigación y Falsos Positivos 10](#_Toc46235354)

[4. Evolución de la calidad del código 11](#_Toc46235355)

[5. Conclusiones 12](#_Toc46235356)

1. Introducción
   1. Objetivo

Los resultados expuestos en este documento tratan de reflejar el estado de la aplicación {{ project\_name }} analizada desde el punto de vista de la calidad de su código fuente.

* 1. Alcance

El personal de SOGETI se encarga de realizar el análisis del código entregado por el cliente para la aplicación {{ project\_name }} y presentar los resultados del mismo, así como las recomendaciones pertinentes (si las hubiera), para consideración por parte del cliente antes de la puesta en producción.

|  |  |  |
| --- | --- | --- |
| Aplicación | Release | Fichero |
| {{ project\_name }} | - | - |
|  |  |  |

El análisis estático de la calidad del código proporcionado en relación con el modelo de calidad estándar (ver capitulo siguiente) basado en las mejores prácticas de programación de la industria.

Las incidencias encontradas durante el proceso de análisis estático de código de la aplicación {{ project\_name }} del cliente son analizadas y el resultado es objeto de las consideraciones que se presentan en este informe.

* + 1. Fuera de alcance

Quedan fuera del alcance de este documento de resultados de análisis estático:

* El análisis de código de aplicaciones o librerías adyacentes.
* El análisis de código de otras aplicaciones distinta de {{ project\_name }}.
* Cualquier otro aspecto o tarea no especificada explícitamente en el apartado.
  1. Prerrequisitos y Condiciones

Para llevar a cabo las tareas relacionadas con este servicio es imprescindible que se cumplan los siguientes requisitos y condiciones:

* Acceso al código fuente a analizar de la aplicación {{ project\_name }}.

* 1. Modelo de Calidad Aplicado

El modelo de calidad de código empleado está basado en OWASP ASVS (Application Security Verification Standard), norma que define el alcance y las características de la calidad interna. El modelo empleado va más allá de la ASVS, centrándose en la calidad interna.

El modelo empleado se centra en la evaluación de diferentes características del código, permitiendo obtener una visión global de calidad del mismo código, así como focalizar los esfuerzos en aquellas características más importantes para Renfe, distinguiendo las siguientes:

* ***Fiabilidad***. Capacidad del producto Software para mantener un determinado nivel de servicio.
* ***Eficiencia***. Capacidad del producto Software para rendir adecuadamente bajo determinadas circunstancias y consumo de recursos.
* ***Mantenibilidad***. Flexibilidad del producto Software para ser modificado. Las modificaciones pueden incluir correcciones, mejoras o adaptaciones del Software a cambios en el entorno, los requisitos y las especificaciones funcionales.
* ***Portabilidad***. No se tiene en cuenta en este modelo de calidad
* ***Seguridad***. Capacidad del producto Software de proteger la información y los datos utilizados, de manera que personas o sistemas no autorizados no puedan acceder a ellos o modificarlos.

El modelo de calidad aplicado responde a las necesidades planteadas y trata de minimizar el impacto que supone la implantación de un servicio de análisis estático de código en un equipo de desarrollo. Se establece además una clasificación con respecto al tipo de problemas detectados atendiendo a la característica de calidad con la que están relacionados y este informe se referirá a ellos en sus mismos términos. De esta forma, surgen los siguientes tipos de problemas:

* ***Vulnerability***. Problemas relacionados con la seguridad de la aplicación y que pueden suponer un riesgo para la integridad de la misma.
* ***Bug***. Problemas relacionados con la fiabilidad de la aplicación, su capacidad para recuperarse de errores inesperados y mantener el servicio.
* ***Code smell***. Este tipo de problema se relaciona con la mantenibilidad del código fuente, su capacidad para cambiar de una forma eficiente y efectiva.

Así mismo, se detalla a continuación la terminología propia para determinados aspectos internos:

* ***Quality profile***. Conjunto de reglas de calidad de código que se aplican a un lenguaje en concreto.
* ***Quality gate***. Métrica o conjunto de métricas asociadas a ciertos valores límite en los cuales se basa la herramienta para valorar el resultado de un escaneo.
* ***Nivel de madurez.*** Definen los niveles de reglas aplicados y los requisitos que una aplicación debe cumplir en cuanto a seguridad, fiabilidad, mantenibilidad y duplicidad de código, para ir escalando en madurez. Cuanto más nivel se alcance, mejor salud gozará la aplicación.

Estos niveles quedan definidos mediante la siguiente tabla:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Nivel de madurez | Nivel de análisis | Requisitos para cumplir el nivel de madurez | | | | |  |
| Seguridad | Fiabilidad | Mantenibilidad | % Dup | Severidad |  |
| L0 | N/A | Aplicación no incluida en el servicio AECF | | | | |  |
| L0 | L1 | Aplicación incluida en el servicio AECF | | | | |  |
| L1 | L1 | N1=0 | %N1<50% | N1/NLOC<1% | < 40 | Blocker |  |
| L2 | L2 | N2=0 | %N2<25% | N2/NLOC<0,5% | < 20 | + Critical |  |
| L3 | L3 | N3=0 | %N3<10% | N3/NLOC<0,1% | < 15 | + Resto |  |
| L4 | L3 | N3=0 | N3=0 | N3/NLOC<0,05% | < 10 | + Resto |  |
| L5 | L3 | N3=0 | N3=0 | N3/NLOC<0,01% | < 5 | + Resto |  |

De forma adicional a estos conceptos, durante el informe se hará mención a otros aspectos técnicos de la calidad de código que se explican a continuación:

* Regla cumplida. Hace referencia a reglas de calidad de código que pertenecen al modelo de calidad y que no generan ningún problema en el análisis actual.
* Regla incumplida. Son reglas de calidad de código pertenecientes al modelo de calidad de código aplicado y que generan al menos un problema.
* Problema. Un problema hace referencia a un componente (proyecto, modulo, archivo, bloque de código, etc.) que no cumple con la especificación de una regla de calidad.
* Duplicidad. Métrica de calidad de código que recoge el porcentaje de código duplicado que se ha encontrado al analizar la aplicación.
* Bloques duplicados. Un bloque duplicado es un conjunto de líneas de código que se encuentran al menos dos veces escritos en la aplicación.
  1. Perfiles Implicados

La siguiente tabla muestra las personas involucradas en este proyecto:

|  |  |  |
| --- | --- | --- |
| Rol | Nombre | Empresa |
| Analista | Mario Peris Blanch | SOGETI |
|  |  |  |

1. Resultados

Esta sección muestra la información de partida para conocer la aplicación que se está valorando, su tamaño y los lenguajes implicados en la misma.

* 1. Características Generales de la Aplicación

|  |  |
| --- | --- |
| Datos Generales |  |
| Nombre del proyecto | {{r rt|e}} |
| Análisis base | {{ report\_date }} |
| Líneas de código | {{ ncloc }} LOC |
| Lenguajes  {% for language in lang\_metrics %} | {{ language[0] }}  {{ language[4] }} LOC ({{language[2] }} %) |
| {% endfor %} |  |
| Nivel Madurez | Nivel L0 |
| Nivel de Análisis | [Nivel L1](http://subdirecciones2013.central.sepg.minhac.age/sitios/DOIP/Calidad/Documentos%20compartidos/AN%C3%81LISIS%20EST%C3%81TICO%20DE%20C%C3%93DIGO/DOCUMENTACION/IGAE_Quality%20Model_v0.5.xlsx?Web=1) |
| Quality gates | No aplica |

* 1. Calidad General de Código

A continuación, se muestran métricas generales que otorgan una visión global de la calidad del código de la aplicación.

|  |  |
| --- | --- |
| Datos Calidad |  |
| Vulnerabilities | {{ vulnerabilities }} |
| Bugs | {{ bugs }} |
| Code smells | {{ code\_smells }} |
| Duplicidadd | {{ duplicity }} % |
| Bloques duplicados | {{ duplicated\_blocks }} |
| Nivel Madurez | Nivel L0 |
| Nivel de Análisis | [Nivel L1](http://subdirecciones2013.central.sepg.minhac.age/sitios/DOIP/Calidad/Documentos%20compartidos/AN%C3%81LISIS%20EST%C3%81TICO%20DE%20C%C3%93DIGO/DOCUMENTACION/IGAE_Quality%20Model_v0.5.xlsx?Web=1) |

<Pie Chart>

* 1. Nivel y modelo de calidad

{{ project\_name }} se encuentra actualmente en el **Nivel X de análisis del modelo de calidad** acordado entre los equipos de desarrollo y calidad. Dentro de este nivel, se muestran a continuación la relación de reglas cumplidas e incumplidas (reglas que generan al menos un problema):

< Stacked column chart >

Además, se muestra a continuación los problemas encontrados categorizados por tipo y lenguaje.

< Column datatable chart >

* 1. Incidencias Detectadas

Esta sección muestra el total de incidencias encontradas tras realizar el análisis de código estático. Primero atendiendo a su severidad y posteriormente clasificadas por característica de calidad.

* + 1. Incidencias clasificadas por prioridad

Para tener una visión más detallada de la distribución de los defectos detectados, en la siguiente tabla se muestran clasificadas por gravedad.

|  |  |
| --- | --- |
| Prioridad | Incidencias |
| Muy Alta | {{ hot\_spot }} |
| Alta | {{ vulns }} |
| Normal | {{ bugs }} |
| Baja | {{ code\_smells }} |

* + 1. Incidencias por característica de calidad

La siguiente tabla muestra las incidencias clasificadas por característica de calidad:

|  |  |
| --- | --- |
| Característica | Total Defectos |
| Mantenibilidad | {{ code\_smells }} |
| Fiabilidad | {{ bugs }} |
| Eficiencia | {{ duplicated\_blocks }} |
| Seguridad | {{ vulnerabilities }} |

* 1. Lista detallada de defectos

Se adjunta un Excel con la lista completa de defectos encontrados incluyendo su localización en el código para facilitar la búsqueda y resolución de las incidencias. Además, incluye una lista de reglas impactadas y un resumen ejecutivo que facilita la visión global de la calidad del código de la aplicación {{ application\_name }}.

<<Excel con reglas incumplidas>>

* 1. Listado de reglas incumplidas

Las reglas incumplidas y el número de problemas que ocasiona cada una, separadas por tipo, son las siguientes:

**Security Hotspots**

|  |  |  |
| --- | --- | --- |
| Lenguaje | Regla | Problemas |
| {% for lang in hotspot\_violations[2] %}{% for rule in lang[1] %}{{ lang[0] }} | {{ rule[0] }} | {{ rule[1] }} |
| {% endfor %}{% endfor %}Total |  | {{ hotspot\_violations[1] }} |

**Vulnerability**

|  |  |  |
| --- | --- | --- |
| Lenguaje | Regla | Problemas |
| {% for lang in vuln\_violations[2] %}{% for rule in lang[1] %}{{ lang[0] }} | {{ rule[0] }} | {{ rule[1] }} |
| {% endfor %}{% endfor %}Total |  | {{ vuln\_violations[1] }} |

**Bug**

|  |  |  |
| --- | --- | --- |
| Lenguaje | Regla | Problemas |
| {% for lang in bugs\_violations[2] %}{% for rule in lang[1] %}{{ lang[0] }} | {{ rule[0] }} | {{ rule[1] }} |
| {% endfor %}{% endfor %}Total |  | {{ bugs\_violations[1] }} |

**Code Smell**

|  |  |  |
| --- | --- | --- |
| Lenguaje | Regla | Problemas |
| {% for lang in cs\_violations[2] %}{% for rule in lang[1] %}{{ lang[0] }} | {{ rule[0] }} | {{ rule[1] }} |
| {% endfor %}{% endfor %}Total |  | {{ cs\_violations[1] }} |

1. Evolución de la calidad del código

En términos de calidad, la aplicación presenta un estado muy similar al del análisis anterior, incluyendo alguna mejora en fiabilidad:

No aplica debido a que es el primer análisis realizado de {{ project\_name }} y no existen datos suficientes para realizar la comparativa. En esta sección se realizará un análisis de la evolución de las métricas más destacadas y una valoración de la calidad a nivel histórico, hitos planteados e hitos conseguidos.

1. Análisis de la aplicación

Después de someter la aplicación {{ project\_name }} a un análisis más detallado basado en los resultados obtenidos por el análisis estático de código y teniendo en cuenta que este informe, por ser el primero, tiene como objetivo presentar una primera radiografía de la aplicación y establecer un punto de partida sobre el que comenzar a trabajar en la mejora de la calidad del código fuente, se exponen las siguientes conclusiones.

* 1. Vulnerabilidades
  2. Bugs
  3. Code Smells
  4. Duplicidad

1. Conclusiones

Conclusiones….

|  |  |
| --- | --- |
| Sobre Sogeti  Sogeti is a leading provider of technology and engineering services. Sogeti delivers solutions that enable digital transformation and offers cutting-edge expertise in Cloud, Cybersecurity, Digital Manufacturing, Digital Assurance & Testing, and emerging technologies. Sogeti combines agility and speed of implementation with strong technology supplier partnerships, world class methodologies and its global delivery model, Rightshore®. Sogeti brings together more than 25,000 professionals in 15 countries, based in over 100 locations in Europe, USA and India. Sogeti is a wholly-owned subsidiary of Capgemini SE, listed on the Paris Stock Exchange.  Para más información acceda a: [www.sogeti.com](https://capgemini-my.sharepoint.com/personal/mario_peris-blanch_sogeti_com/Documents/KiuwanTools%20(1)/kiuwan/templates/www.sogeti.com) |  |
| La información contenida en este documento es confidencial y propiedad de Sogeti © 2018 Sogeti. Todos los derechos reservados. | |

|  |
| --- |
|  |