

20 DE MAYO DE 2019

WS-GUARDIAN **CLIENT'S AGENT**



SOFTWARE REQUIREMENTS SPECIFICATION PROYECTO DE GRADO

CHRISTIAN GIOVANNY ROJAS DIAZ
PONTIFICIA UNIVERSIDAD JAVERIANA
Colombia, Bogotá D.C.

JUAN DAVID GAMA PEÑA
PONTIFICIA UNIVERSIDAD JAVERIANA
Colombia, Bogotá D.C.

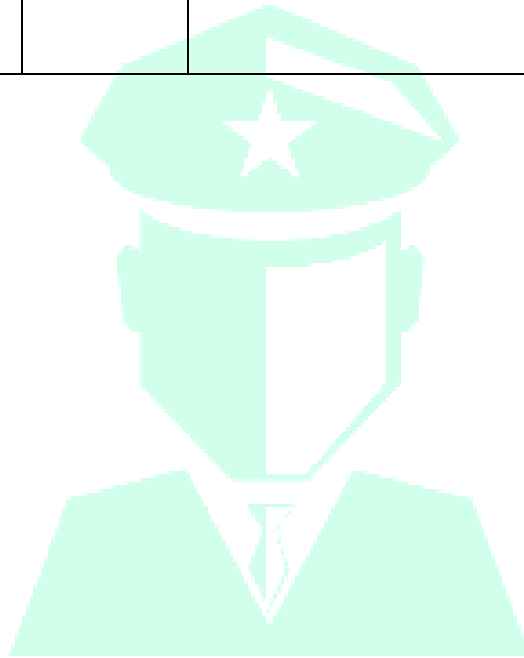
MICHAEL ANDRÉS VARGAS BUITRAGO
PONTIFICIA UNIVERSIDAD JAVERIANA
Colombia, Bogotá D.C.

RIE KANEKO BOJACÁ
PONTIFICIA UNIVERSIDAD JAVERIANA
Colombia, Bogotá D.C.

WILLIAM ALEXANDER MORENO PRIETO
PONTIFICIA UNIVERSIDAD JAVERIANA
Colombia, Bogotá D.C.

Historial de Cambios

Versión	Fecha	No. Sección	Descripción de cambios	Responsable
1.0	12/05/2019	Todo	Creación del documento	Todo



WSG-Agent



Contenido

Listas de Tablas	3
Anexos	3
1. Introducción	4
1.1. Propósito	4
1.2. Alcance	4
2. Descripción Global	5
2.1. Perspectiva del Producto	5
2.1.1. Operaciones	5
2.1.2. Requerimientos de adaptación	5
2.2. Funciones del Producto	6
2.3. Características del Usuario	6
2.4. Restricciones del Software	6
2.5. Suposiciones y Dependencias	6
2.6. Distribución de Requerimientos	7
3. Requerimientos Específicos	8
3.1. Requerimientos de Interfaces Externas	8
3.1.1. Interfaces con el usuario	8
3.1.2. Interfaces con el software	8
3.1.3. Interfaces de comunicación	8
3.2. Características del Producto de Software	9
3.3. Requerimientos No Funcionales	9
4. Referencias	10



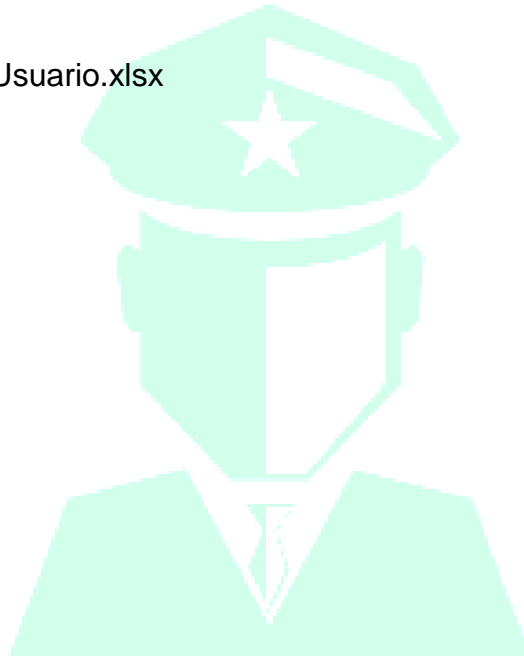
Listas de Tablas

TABLA 1: OPERACIONES DE USUARIOS.....	5
TABLA 2: RESTRICCIONES MÍNIMAS DE MEMORIA [1].....	5
TABLA 3: MAPEO Y DISTRIBUCIÓN DE REQUERIMIENTOS	7
TABLA 4: P-1.....	9

Anexos

Anexo 1: Glosario.pdf

Anexo 2: Historias de Usuario.xlsx



WSG-Agent



1. Introducción

1.1. Propósito

Este documento continúa con la planeación realizada en el documento anterior (*Software Project Management Plan*). Su objetivo es describir *WSG-Client's Agent* en materia de desarrollo a lo largo del documento, planteando el contexto y de dónde nace *WSG-Client's Agent*. Se plantean las interfaces a nivel de comunicación con sistemas externos y con el usuario; además, se muestran los roles previstos para el agente, finalizando con la descripción de sus funciones y requerimientos (funcionales y no funcionales).

El siguiente documento va dirigido a ITAC S.A, Pontificia Universidad Javeriana y a quien le interese el desarrollo del proyecto *WSG-Client's Agent*.

1.2. Alcance

Actualmente, cuando una empresa instala políticas de seguridad sobre sus *Web Services* (WS), la responsabilidad de implementar las técnicas de seguridad para consumir dichos WS queda a cargo del cliente. Esto hace que el aplicar políticas de este tipo en sus servicios, sea costoso en materia de tiempo, dinero y esfuerzo; al tener que hacer los cambios en sus aplicaciones consumidoras de estos WS.

Con *WSG-Client's Agent* se quiere reducir los costos del cliente (tiempo, dinero y esfuerzo), al momento de consumir un WS. Logrando conocer automáticamente qué políticas requiere un servicio para ser consumido (Políticas que pueden añadirse, modificarse y o ser removidas desde *WS-Guardian*) y aplicándolas sobre el mensaje, disminuyendo los costos del cliente al no tener que modificar su sistema.

WSG-Client's Agent contará con:

- Comunicación con *WS-Guardian* para conocer las políticas por servicio.
- Un Registro de logs con un tamaño de archivo variable definido por el cliente, con el fin de ayudar a visualizar el estado del agente.
- Una capa de administración para los módulos que implementan técnicas de seguridad.
- Dos módulos con la implementación de dos técnicas de seguridad,
- Capacidad de obtener la implementación de técnicas de seguridad, en tiempo de ejecución de las políticas que se requieran.
- Comunicación de con *WS-Guardian* para realizar una inicialización de *WS-Guardian Client's Agent*.

En el [Anexo 1: Glosario.pdf](#) se listan los conceptos clave del proyecto y producto.

2. Descripción Global

2.1. Perspectiva del Producto

WSG-Client's Agent será un software perteneciente a la familia de *WS-Guardian*, un producto desarrollado por la empresa ITAC S.A. Se desplegará junto al consumidor de *WS*, para asegurar la información a nivel de mensaje, desde el origen hasta el destino, como se menciona en [VFP, sección 1. Visión Global](#).

2.1.1. Operaciones

Modos de operación de usuarios: En la *Tabla 1: Operaciones de usuarios*, se describen las operaciones que cada usuario realizará dentro del sistema.

Usuario	Descripción
Administrador	Podrá crear y modificar las políticas de seguridad para cada servicio web. También podrá conceder o denegar el acceso de un <i>WSG-Agent</i> a un <i>WS</i> .
Desarrollador	Podrá ver el registro de eventos y acciones que se presentan en <i>WSG-Agent</i> .

Tabla 1: Operaciones de usuarios

2.1.2. Requerimientos de adaptación

Debido a que *WSG-Client's Agent* debe funcionar de forma independiente, este no tendrá que ajustarse a un modo operacional (administrador), pero sí deberá adaptarse a las políticas exigidas por *WS-Guardian*.

Para la instalación y ejecución, se debe contar con *JRE* v1.8 en la máquina del consumidor de *WS*. Adicional a esto, es necesario que cumpla con las restricciones mínimas de memoria mostradas en la *Tabla 2: Restricciones mínimas de memoria*.

Recurso	Tamaño	Justificación
RAM	128 MB	Requerido para usar el JRE
	1 GB	Requerido para ejecutar <i>WSG-Client's Agent</i>
Espacio en disco	124 MB	Requerido para usar el JRE
	1 GB	Espacio destinado para almacenar los JAR con las políticas de seguridad y archivos de log

Tabla 2: Restricciones mínimas de memoria [1]

2.2. Funciones del Producto

Las principales funciones de *WS-Guardian Client's Agent* son:

- Descargar la configuración y restricciones de seguridad de *WS-Guardian*, por medio de servicios *REST*.
- Descargar las técnicas de seguridad requeridas en los servicios, para almacenarlas y usarlas en tiempo de ejecución.
- Agregar a mensajes *SOAP* políticas de seguridad bajo el estándar *WS-Security* y *WSI Basic profile*.
- Retransmitir los mensajes *SOAP* con políticas de seguridad hacia *WS-Guardian*, para completar la transacción de consumo del cliente.
- Eliminar y validar las políticas de seguridad de una respuesta *SOAP* recibida de *WS-Guardian*, para retransmitirla al cliente.

2.3. Características del Usuario

Para *WS-Guardian Client's Agent* se tendrán los roles de: Administrador, Cliente y Desarrollador. La descripción y los privilegios de cada uno se muestran en el [Anexo 2: Historias de Usuario.xlsx](#) (hoja "Roles").

2.4. Restricciones del Software

A continuación, se listan las restricciones enunciadas por ITAC que limitan el alcance del desarrollo del proyecto:

- *WS-Guardian Client's Agent* se debe desarrollar sobre JDK 1.8 open-source.
- Las políticas de seguridad deben ser aplicadas de acuerdo con el estándar *WS-Security*.
- El código fuente debe estar escrito bajo lo establecido por Java Coding Standards y debidamente comentado.
- Las políticas se deben implementar bajo el estándar *WS-Security* y *WSI Basic profile 1.0*.

2.5. Suposiciones y Dependencias

A continuación, se listan los factores que afectan los requerimientos del proyecto:

- **Suposiciones:**
 - *WS-Guardian* está previamente instalado y puesto en funcionamiento para que *WSG-Client's Agent* se pueda comunicar.
 - La máquina del cliente tendrá instalado *JRE* v1.8 para ejecutar *WSG-Client's Agent*, si no lo tiene, este se instalará.
 - *WS-Guardian* tendrá implementadas las interfaces de comunicación usadas por *WSG-Client's Agent*.

- El cliente que consume los servicios web estará registrado en *WS-Guardian* antes de que *WS-Guardian Client's Agent* inicie su ejecución.
 - Los requerimientos del sistema son estables una vez aprobados por ITAC S.A., y no cambiarán durante el desarrollo del proyecto.
- **Dependencias:**
 - Se usará el modelo de datos de *WS-Guardian* para relacionar al *WSG-Client's Agent* con los clientes del sistema, y de este modo identificar los WS.
 - La arquitectura de diseño seguirá el modelo *Cliente/Servidor*. Por esto, la disponibilidad del sistema dependerá de la conexión entre la máquina donde reside *WSG-Client's Agent* y *WS-Guardian*.

2.6. Distribución de Requerimientos

La *Tabla 3: Mapeo y distribución de requerimientos*, muestra el mapeo y distribución de los requerimientos de *WS-Guardian Client's Agent*, relacionando funcionalidades y los siguientes módulos del sistema:

- Configuración: Definido para obtener y dar soporte a la información de configuración (servicios y su relación con las políticas de seguridad).
- Comunicación: Este módulo contendrá toda la lógica para atender las peticiones hechas por el cliente consumidor de WS.
- Políticas: Este módulo es destinado para el uso y gestión de las políticas de seguridad (archivos JAR).
- Auditoría: La finalidad de este módulo es generar las líneas de registro de eventos (logs) que ocurren en el sistema y que son almacenadas en los archivos planos con estructura definida.

	Módulos				Historias de Usuario
	Configuración	Comunicación	Políticas	Auditoría	Id
Descargar archivos de configuración	✓	✓		✓	F-6, F-10, F-13, F-16
Descargar los JAR con políticas de seguridad	✓	✓	✓	✓	F-7, F-9, F-13
Agregar políticas de seguridad a los mensajes SOAP			✓		F-5, F-15, F-17
Retransmitir mensaje SOAP con políticas hacia WS-Guardian		✓	✓		F-19
Quitar las políticas de seguridad a los mensajes SOAP recibidos de WS-Guardian			✓		F-17
Actualizar políticas de seguridad	✓	✓		✓	F-10, F-11, F-13
Registrar eventos ocurridos en el sistema				✓	F-14

Tabla 3: Mapeo y distribución de requerimientos

3. Requerimientos Específicos

3.1. Requerimientos de Interfaces Externas

3.1.1. Interfaces con el usuario

WSG-Client's Agent contará con dos interfaces dedicados a los diferentes roles entre el sistema, el rol administrador y el usuario.

- **Cliente:** Su interfaz será por consola y podrá ser accedida en la fase de configuración inicial de *WSG-Client's Agent*. Debe ser parametrizado y sólo permite entradas por teclado.
- **Administrador:** Al ser el encargado de administrar *WS-Guardian*, tendrá que contar con una interfaz web donde pueda registrar los diferentes agentes haciendo uso del respectivo ID. La interfaz estará contenida en *WS-Guardian*.

3.1.2. Interfaces con el software

Como se mencionó en el [SPMP, sección 3.2. Lenguajes y Herramientas](#), el sistema será desarrollado en el lenguaje Java sobre el servidor JRE, para implementar Java en entornos de servidor y disponibles para multiplataforma. [2]

De acuerdo al objetivo específico “*Implementar un prototipo del agente para dos o más plataformas*”, el sistema debe funcionar en diferentes plataformas, al menos las dos principales que son: Windows y Linux.

3.1.3. Interfaces de comunicación

WS-Guardian Client's Agent es un complemento de *WS-Guardian*. Este es un producto de la empresa ITAC S.A, quien es responsable de brindar mayor seguridad y gobernabilidad sobre arquitecturas orientadas a servicios (SOA). [3] Esto conlleva a que *WS-Guardian Client's Agent* cuente con interfaces para comunicarse con *WS-Guardian*:

- **Interfaz REST:** Con el fin de descargar los JAR necesarios para satisfacer las políticas requeridas por los clientes que ejecutan *WS-Guardian Client's Agent*. Esta interfaz será la encargada de responder a las peticiones (https) del agente cuando se soliciten dichas políticas.

WS-Guardian Client's Agent, además contará con una interfaz para recibir peticiones (https) del cliente al que está asignado para incorporar las técnicas de seguridad necesarias, al momento de consumir el *Web Service* protegido por *WS-Guardian*.

3.2. Características del Producto de Software

En esta sección se describen los requerimientos funcionales mediante historias de usuario. Se escogió el uso de historias de usuario para la especificación de los requerimientos debido a la metodología *SCRUM* que es manejada en el proyecto, y además porque la empresa ITAC S.A. solicitó detallarlos de esta manera.

En el [Anexo 2: Historias de Usuario.xlsx](#) (hoja “características”), se listan las historias de usuario funcionales del producto en detalle. Los diferentes roles que pueden aparecer a lo largo de la especificación de requerimientos son los ya mencionados en la [sección 2.3 Características del Usuario](#).

3.3. Requerimientos No Funcionales

En esta sección se mencionan las historias de usuario relacionadas con los requerimientos no funcionales, teniendo en cuenta los atributos de calidad mencionados en la [Versión Final Propuesta, sección 5.1.5.1 Principios de Seguridad](#).

En el [Anexo 2: Historias de Usuario.xlsx](#) (hoja “No funcionales”), se listan las historias de usuario no funcionales del producto en detalle teniendo en cuenta los roles mencionados en la [sección 2.3 Características del Usuario](#).

Requerimiento de Desempeño

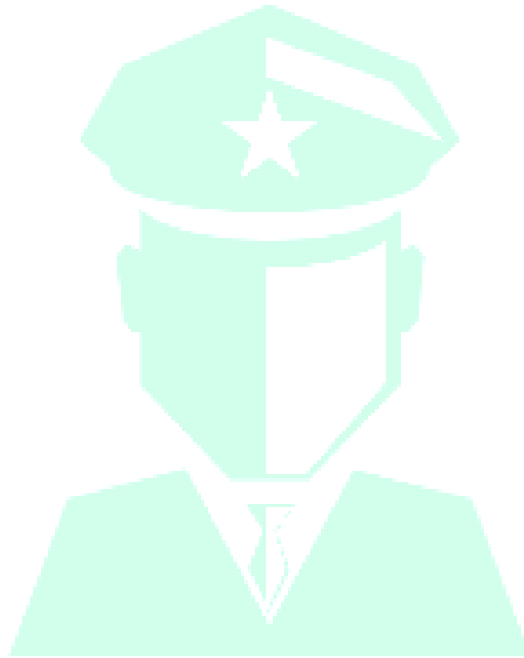
A continuación, se describen los requerimientos no funcionales del modelo FURPS+, correspondientes al atributo de disponibilidad del sistema.

ID	P-1	Requerimiento	El sistema no debe tardar más de 1 segundo aplicando una política de seguridad
Justificación	Aplicar una política de seguridad de manera que el usuario pueda solicitar un servicio sin mayor espera.		
Criterio de aceptación	Aplicar una política de seguridad correctamente en un tiempo menor o igual a 1 segundo.		
Restricciones	Limitación de recursos de hardware en la máquina del cliente que afecten el buen rendimiento del agente.		
Prioridad	Should have	Historia asociada	NF-6

Tabla 4: P-1

4. Referencias

- [1] «What are the system requirements for Java?» [En línea]. Disponible en: <https://www.java.com/en/download/help/sysreq.xml>. [Accedido: 18-may-2019].
- [2] «Server JRE (Java SE Runtime Environment) 8 Downloads». [En línea]. Disponible en: <https://www.oracle.com/technetwork/java/javase/downloads/server-jre8-downloads-2133154.html>. [Accedido: 06-may-2019].
- [3] «Home - wsguardian.co». [En línea]. Disponible en: <http://wsguardian.co/home>. [Accedido: 30-mar-2019].



WSG-Agent

