El siguiente cuadro nos ayuda para gestionar las diferentes modificaciones del contrato del servicio.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CONTROL DE VERSIONES** | | | | |
| **Versión** | **Elaborado por** | **Aprobado por** | **Fecha** | **Motivo** |
| 1.0 | Junior Mateo |  | 09/04/2019 | Versión inicial |

**DATOS DEL PROYECTO:**

|  |  |
| --- | --- |
| **Codigo del Proyecto** | **Nombre del Proyecto** |
| PROY-140067 | IDEA-140067 Motores de Promociones para ONE AMX |

**TABLA DE CONTENIDOS**

[**DATOS DEL PROYECTO:** 1](#_Toc498703416)

[1. Objetivo 3](#_Toc498703417)

[2. Audiencia 3](#_Toc498703418)

[3. Diagrama de Interfaces 3](#_Toc498703419)

[4. Plantilla de diseño del servicio 3](#_Toc498703420)

[4.1 Datos generales del servicio RegistrarConstancias 3](#_Toc498703421)

[4.2 Plantilla de especificación de cada método del servicio 4](#_Toc498703422)

[4.2.1 Método main 4](#_Toc498703423)

[4.2.1.1 Datos de entrada del método main 5](#_Toc498703424)

[4.2.1.2 Datos de salida del método main 5](#_Toc498703425)

[4.2.1.3 Diseño del método main del servicio RegistrarConstancias 5](#_Toc498703426)

[4.2.1.4 Recursos que usará el método main 8](#_Toc498703427)

[4.2.1.5 Códigos/Mensajes de respuesta del método main 8](#_Toc498703428)

[4.2.1.6 Métrica del negocio del método main del servicio RegistraConstancias 8](#_Toc498703429)

Especificación de diseño del servicio

# Objetivo

El objetivo del presente documento es presentar la plantilla que sirve para hacer el diseño de un servicio (servicio batch “asíncrono”, servicio de negocio o servicio compuesto) del área de Arquitectura e Integración, este diseño tiene un perfil funcional y técnico alineado a la propuesta de solución, sirviendo de Input para los desarrolladores.

Adicionalmente, este documento contiene la descripción de las diferentes características que debe contener el servicio al momento de ser diseñado, desde el modelado del servicio hasta la invocación de los servicios y/o otros recursos.

Este documento solo contiene datos del ambiente de desarrollo.

# Audiencia

Los estándares del presente documento están dirigidos a:

* Arquitectos de integración que requieran diseñar un nuevo servicio
* Desarrolladores de soluciones de tecnología de información que requieran consultar el detalle de la implementación del servicio.
* Directores y Líderes de proyectos que requieran compartir la funcionalidad del servicio.

# Diagrama de Interfaces

En el diagrama se muestra las interfaces del servicio y su dependencia con otros servicios.



# Plantilla de diseño del servicio

## Datos generales del servicio RegistraConstancias

Los detalles del servicio debe contemplar la especificación funcional y técnica **SOLO** de los servicios de negocio, servicios compuestos (orquestación) y/o servicios batch “asíncronos”, NO aplica para servicios de datos puntuales que exponen funcionalidades granulares (aquellos que no contienen un flujo, por ejemplo los servicios de OSB de Datos y Servicios de Reglas BRMS, estos solo serán detallados en el Contrato del Servicio).

|  |  |
| --- | --- |
| **Nombre del servicio** | SH02\_MOTPROM\_ValidaDes\_Vigencia |
| **Descripción del servicio** | JAR que permitirá ejecutar la desactivación de Bonos Por vigencia |
| **Beneficios del negocio** | desactivación de Bonos Por vigencia |
| **Objetivo estratégico relacionado** | desactivación de Bonos Por vigencia |
| **ID** |  |
| **Dueño del servicio** | Post Venta |
| **CLARITY** | PROY-140067 |
| **Número de métodos** | 1 |
| **Clasificación del servicio** | JAVA\_X\_\_ BPEL\_\_\_ OSB Compuesto\_\_\_ |
| **Analista De Integración** | Junior Mateo |
| **Analista Funcional** | Jose Tinoco |

## Plantilla de especificación de cada método del servicio

### Método SH02\_MOTPROM\_ValidaDes\_Vigencia

|  |  |  |
| --- | --- | --- |
| **Nombre del método:** | main | Nuevo \_ X\_ Modificado \_\_ |
| **Método:** | Síncrono (Request/Response) \_\_\_ Asíncrono(OneWay)\_X\_ Asíncrono (ACK) \_\_ | |
| **Descripción del Método:** | Este método se encargara de la desactivación de Bonos Por vigencia | |
| **Pruebas de Stress (AC: rendimiento):** | Si \_ \_ No\_X\_ | |
| **Máximo SLA permitido (AC: rendimiento):** | \_300\_\_ ms | |
| **Tipo de notificación ante errores :** | SMS\_\_\_ Mail\_\_\_ Otros\_\_\_\_ Ninguno \_\_X\_\_ | |
| **Estimación de transacciones por segundo** | \_3\_ tps | |
| **Disponibilidad del servicio**  **(AC: disponibilidad)** | \_\_24 x 7\_ | |
| **Aspectos de seguridad (AC: seguridad)** | **Acceso al servicio:**  Restricción por IP:\_X\_ Por autenticación HTTP: \_\_\_  **Canal:**  HTTP:\_X\_ HTTPS\_\_\_ | |
| **Métricas del servicio (uso de BAM)** | **Aplica:** SI\_\_\_ NO\_X\_\_ | |
| **Log de transacción (estados del servicio en tabla)** | Si\_\_\_ No\_X\_\_ | |
| **Palabras claves** | Procesar, archivo output | |

#### Datos de entrada del método main

No Aplica

**Diagrama request xsd.**

No Aplica

#### Datos de salida del método main

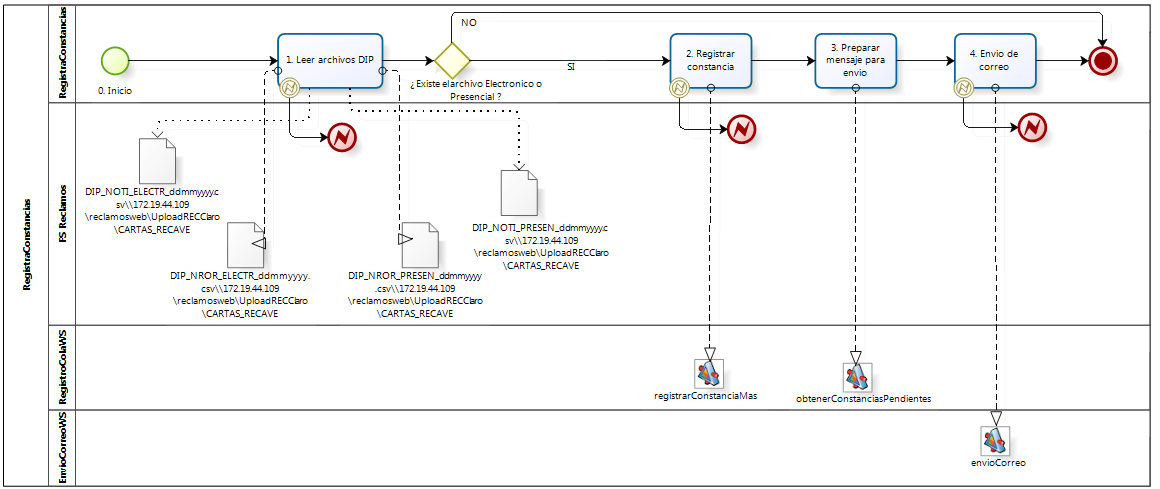
No Aplica

**Diagrama request xsd.**

No Aplica

#### Diseño del método main del servicio registrarConstanciaMas

1. **Modelamiento del proceso del método**



1. **Diagrama de Componentes:**



1. **Descripción de cada actividad del proceso:**

Actividad del proceso

Descripción del proceso

|  |  |  |
| --- | --- | --- |
| **Actividad del proceso** | **Descripción del proceso** | **Componente** |
| 0.Inicio | Se envía el idTransaccion y ruta de properties al jar. | RegistraConstancias.jar |
| 1. Leer los archivos DIP | Se precederá a leer los archivos DIP\_NOTI\_ELECTR\_yyyymmdd.csv  (ejm: DIP\_NOTI\_ELECTR \_20180214.csv)  Si se encuentra el archivo se procederá a leer el archivo resumen DIP\_NROR\_ELECTR\_yyyymmdd.csv (ejm: DIP\_NROR\_ELECTR\_20180214.csv)  Se precederá a leer los archivos DIP\_NOTI\_PRESEN \_yyyymmdd.csv  (ejm: DIP\_NOTI\_ PRESEN\_20180214.csv)  Si se encuentra el archivo se procederá a leer el archivo resumen DIP\_NROR\_PRESEN \_yyyymmdd.csv (ejm: DIP\_NROR\_ PRESEN \_20180214.csv)  Los archivos se encuentran en el servidor  [\\172.19.44.109\reclamosweb\UploadRECClaro\CARTAS\_RECAVE](file:///\\172.19.44.109\reclamosweb\UploadRECClaro\CARTAS_RECAVE) y nos permitira leer y generar copias en nuestro archivo local.   * Si se encuentra el archivo DIP\_NROR\_ELECTR\_yyyymmdd.csv o DIP\_NROR\_PRESEN \_yyyymmdd.csv el codigoRespuesta será 1. * Si no se encuentra el archivo DIP\_NROR\_ELECTR\_yyyymmdd.csv ni el archivo DIP\_NROR\_PRESEN \_yyyymmdd.csv el codigoRespuesta será 2. * Si el archivo DIP\_NROR\_ELECTR\_yyyymmdd.csv o el archivo DIP\_NROR\_PRESEN \_yyyymmdd.csv se encuentra vacio el codigoRespuesta también será 2.   Si el codigoRespuesta==1 significa que uno o ambos archivos de resumen se han encontrado y se enviaran a registrar en el **paso 2**.  Sino si codigoRespuesta==3  Se establece en el log el mensaje “No se encontro el archivo CSV” y se mostrara el siguiente mensaje  “No se encontro el archivo CSV DIP\_NOTI\_ELECTR\_20180214.dip  No se encontro el archivo CSV DIP\_NOTI\_PRESEN\_20180214.dip”  a ser enviado por correo en el **paso 4**.  Sino si codigoRespuesta==5 significa que ambos archivos de resumen no se han encontrado entonces se establece en el log el mensaje “El archivo CSV esta vacio”  Sino  Se establece en el log el mensaje “El archivo CSV contiene errores de datos” y el flujo finaliza. | RegistraConstancias.jar |
| 2. Registrar constancia | Se iteraran las constancias obtenidas y se adicionaran en una lista de tipo ListaCamposAdicionalesType para ser enviadas y registradas mediante el servicio RegistroColaWS. Dicho servicio va a ser invocado hasta una tercera vez en caso haya registros que no se hayan podido registrar.  **Servicio: RegistroColaWS**  http://172.19.122.140:10002/RegistroColaWS/ebsRegistroColaSB11?wsdlMétodo: registrarConstancias  **DATOS DE ENTRADA**   |  |  | | --- | --- | | **Parámetro de entrada** | **Descripción /Valor** | | auditRequest/idTransaccion | Id de transacción (Obtenido como argumento o generado en el Jar) | | auditRequest/ipAplicacion | Ip de la aplicación (Configurable en properties) | | auditRequest/nombreAplicacion | Nombre de la aplicación(Configurable en properties) | | auditRequest/usuarioAplicacion | Usuario de la aplicación(Configurable en properties) | | tipoReclamo | Identificador de tipo de reclamo (1) electronica (2) Presencial | | listaCamposAdicionalesType | Lista que contiene la información de los archivos tanto electrónica como presencial. |   **DATOS DE SALIDA**   |  |  | | --- | --- | | **Parámetro de salida** | **Descripción** | | auditResponse/idTransaccion | Id de transacción de la consulta | | auditResponse/codigoRespuesta | Codigo de respuesta | | auditResponse/mensajeRespuesta | Mensaje de respuesta | | listaCamposAdicionalesType | Lista que contiene la información de los registros que no se insertaron en la BD de Clarify. |   **Consideraciones:**   * Si el código de respuesta es diferente a 0, establecer en el log error de respuesta del servicio y el flujo finaliza. * Si existiera algún inconveniente de Timeout con el WS, establecer en el log error de Timeout y el flujo finaliza. * Si existiera algún inconveniente de no disponibilidad con el WS, establecer en el log error de no disponibilidad de WS y el flujo finaliza. * Si existiera un error de proceso interno, mostrar en el log error interno y el flujo finaliza. * Caso contrario continuar con la actividad 4   Nota: Este servicio se invocara hasta por tres veces en caso el resultado devuelva el cursor con registros.   * Si no se grabo ningún registro se elimina el archivo local que es la copia del archivo obtenido originalmente. * Si se grabaron algunos registros procedemos a crear un archivo con los registros pendientes y lo reemplazamos por el archivo en el servidor original. Finalmente eliminamos la copia del archivo local. * Si se grabaron todos los registros se elimina el archivo origen y el archivo local. | RegistraConstancias.jar |
| 3. Preparar el mensaje para el envio | Aquí se cogen las listas Electronicas y Presenciales realizando el conteo de los datos necesarios para presentar en reporte en el cual estarán la siguiente información como se muestra en el reporte de ejemplo:  Resultados del envío de constancias de los Reclamos con notificación por correo electrónico.  === REPORTE DE ELECTRONICOS ===  19 reclamos procesados  8 reclamos procesados con éxito  11 reclamos procesados sin éxito  3 reclamos que tienen un archivo  \* Lista de Reclamos procesados con exito : 180002128,180002129,180002130,180002141,180002134,180002135,180002153,18000215  \* Lista de Reclamos procesados sin exito  Reclamos con error averia : 180002140,180002142,180002146,180002147,180002148,180002151. Por favor, en caso de existir averias eliminarlas del repositorio de ONBASE:\172.19.44.109 eclamoswebUploadRECClaro;  Reclamos con error sin cierre: 180002136,180002154. Por favor, en caso de existir reclamos sin cierre procesarlo manualmente  Reclamos no llegaron a Onbase :  Reclamos con un archivo : 180002198,180045198,180445198  === REPORTE DE PRESENCIALES ===  20 reclamos procesados  9 reclamos procesados con éxito  11 reclamos procesados sin éxito  \* Lista de Reclamos procesados con exito : 180002125,180002127,180002137,180002138,180002131,180002132,180002152,180002155,18000214  \* Lista de Reclamos procesados sin exito  Reclamos con error averia : 180002126,180002143,180002144,180002145,180002150. Por favor, en caso de existir averias eliminarlas del repositorio de ONBASE \172.19.44.109 eclamoswebUploadRECClaro;  Reclamos con error sin cierre : 180002139,180002133,180002156. Por favor, en casos de existir cierres procesarlas manualmente  Reclamos no se tomaron en cuenta para el registro : 180312310,132423420,132343420  === REPORTE SIN DOCUMENTO===  Reclamos sin Documentos 5to sin fase PI : 180002148,180001900,180001893,180001885,180001882,180001919,180001883,180002140,180001917,180001904,180001903,180002126,180001902  Informe enviado del proceso de envío de constancias de los Reclamos con notificación por correo electrónico. | RegistraConstancias.jar |
| Envio de correo | Se enviara un correo para informar los erros funcionales en el proceso para ello se ejecuta el servicio EnvioCorreoWS.  <http://172.19.74.223:8909/EnvioCorreoWS/envioCorreoWSPortSB11?wsdl>  Método: procesarEnvioWS  Datos entrada:   |  |  | | --- | --- | | **Parámetro de entrada** | **Valor** | | auditRequest/idTransaccion | Id de transacción (Obtenido de la cola) | | auditRequest/ipAplicacion | Ip de aplicación (IP donde se encuentre instalado el MDB) | | auditRequest/codigoAplicacion | Codigo de aplicación (Obtenido de la cola) | | auditRequest/usrAplicacion | Usuario de aplicación (Obtenido de la cola) | | remitente | Correo remitente (Configurado en propeties) | | destinatario | Correo del cliente a enviar la notificación (Configurado en propeties) | | asunto | Asunto (Configurado en properties) | | mensaje | Mensaje (Configurado en propeties ) | | htmlFlag | Html Flag (Configurado en propeties ) |   Datos de salida:   |  |  | | --- | --- | | **Nombre** | **Descripción** | | auditResponse/idTransaccion | Id de transaccion | | auditResponse/codigoRespuesta | Código de respuesta (Éxito=”0”) | | auditResponse/mensajeRespuesta | Mensaje de respuesta |   **Consideraciones:**   * Si el código de respuesta es diferente a 0, establecer en el log error de respuesta del servicio y el flujo finaliza. * Si existiera algún inconveniente de Timeout con el WS, establecer en el log error de Timeout y el flujo finaliza. * Si existiera algún inconveniente de no disponibilidad con el WS, establecer en el log error de no disponibilidad de WS y el flujo finaliza. * Si existiera un error de proceso interno, mostrar en el log error interno y el flujo finaliza. * Caso contrario continuar con la actividad 4 |  |
| 5. Fin | El flujo termina con exito. | RegistraConstancias.jar |

#### Recursos que usará el método main

* Servidor de Shell :

SH\_OBTENER\_ARCHIVO\_OUTPUT.sh

* WS: RegistroColaWS
  + <http://172.19.122.140:10002/RegistroColaWS/ebsRegistroColaSB11?wsdl> Método: registrarConstanciaMas
* WS: EnvioCorreoWS
  + <http://172.19.74.223:8909/EnvioCorreoWS/envioCorreoWSPortSB11?wsdl> Método: procesarEnvioWS

#### Códigos/Mensajes de respuesta del método main

1. **Respuestas funcionales del método:**

No aplica.

1. **Respuestas técnicas del método:**

No aplica.

#### Métrica del negocio del método main del servicio RegistrarConstancia

No aplica