

Implementación de requerimiento de software para ERP SAP de operador móvil

Trabajo de grado para optar por el título de Ingeniero en Informática

Erik Restrepo Ramírez

Asesor
Juan Carlos Quintero Vallejo
Ingeniero de sistemas

Corporación Universitaria Lasallista
Facultad de ingeniería
Ingeniería informática
Caldas, Antioquia
2014

Dedicatoria

Dedicado a mis seres queridos que siempre han estado para apoyarme en el objetivo de convertirme en Ingeniero Informático.

Contenido

Glosario.....	8
Resumen.....	9
Introducción	10
Justificación	11
Impacto Tecnológico	11
Objetivos	12
Objetivo General.....	12
Objetivos Específicos	12
Marco Teórico.....	13
Tata Consultancy Services.....	13
Identidad Corporativa.....	13
<i>Misión</i>	13
<i>Valores Empresariales</i>	13
ERP (Enterprise Resource Planning).....	14
Tipos de ERP	14
<i>Software a medida</i>	14
<i>Soluciones estándar</i>	15
SAP	15
Tipos Consultores SAP	15
<i>Consultor Basis</i>	15

<i>Consultor Funcional</i>	16
<i>Consultor Abap</i>	16
Estructura SAP	16
<i>Ambiente de Desarrollo</i>	16
<i>Ambiente de Calidad</i>	17
<i>Ambiente de Producción</i>	17
ABAP	17
Objetos ABAP	17
<i>Módulos de Función</i>	17
<i>Módulo de Función RFC</i>	18
<i>Formularios</i>	18
<i>Sapscript</i>	18
<i>Smart Form</i>	18
<i>Web Dynpro</i>	18
<i>Clase de Mensajes</i>	19
<i>Tablas Internas</i>	19
Tipos de Tablas Internas.....	19
<i>Any Table</i>	19
<i>Index Table</i>	19
<i>Standard Table</i>	20
<i>Sorted Table</i>	20
<i>Hashed Table</i>	20
Diccionario de Datos	21

Programación Orientada a Objetos	21
Clase	21
Objeto	22
Herencia.....	22
Metodología	23
Interfaz Payment Request	24
Objetos Creados.....	25
Grupo de funciones.	25
Módulo de Funciones	26
Clase Global	28
Métodos	29
<i>Import_Data</i>	29
<i>Validate</i>	30
<i>Fetch_Data</i>	31
<i>Set_Log</i>	32
<i>Export_Data</i>	33
Clase Mensajes	34
Documentación.....	35
<i>Arquitectura y Diseño</i>	35
<i>Manual Técnico</i>	35
<i>Manual de Usuario</i>	35
<i>Documento de Pruebas</i>	36

Resultados Y Análisis De Los Resultados	37
Conclusiones	38
Recomendaciones.....	39
Bibliografía	40

Tabla de Ilustraciones

Ilustración 1. Estructura SAP	16
Ilustración 2. Tablas Internas	21
Ilustración 3. Interacción entre las tres interfaces.	24
Ilustración 4. Interfaz Payment Request	25
Ilustración 5. Grupo de Funciones	26
Ilustración 6. Módulo de Funciones (lógica).	27
Ilustración 7. Módulo de Funciones (Import y Export).	28
Ilustración 8. Súper Clase.	28
Ilustración 9. Clase Global Payment Request.	29
Ilustración 10. Método Import_Data.....	30
Ilustración 11. Método Validate.....	31
Ilustración 12. Método Fetch_Data.....	32
Ilustración 13. Método Set_Log.....	33
Ilustración 14. Método Export_Data.....	34
Ilustración 15. Clase de Mensajes.	34

Glosario

ERP (Planificación de Recursos Empresariales): es un conjunto de sistemas de información que permite la integración de ciertas operaciones de una empresa, como producción, logística, inventario, recursos humanos, contabilidad, etc.

SAP: es un sistema informático de gestión empresarial, que maneja los procesos de negocios de las empresas que lo usan, gestionando sus recursos humanos, sus finanzas, costos, logística, materiales, inventarios, etc.

ABAP: es un lenguaje de programación de cuarta generación el cual se usa para adaptar las posibilidades del sistema SAP estándar a los requisitos específicos del cliente.

Servicio Web: es un servicio ofrecido por una aplicación que expone su lógica a clientes de cualquier plataforma mediante una interfaz accesible a través de la red utilizando protocolos estándar de internet.

Requerimiento de software: es el pedido que realiza el usuario para que se realice o modifique un programa o aplicativo.

Modularización: es encapsular código de un programa y crear una función; la funcionalidad de este código se puede volver a utilizar sin necesidad de copiar el código, sino haciendo llamado a la función en la que se encapsulo.

Resumen

En este informe de prácticas empresarial se desea compartir la experiencia adquirida en la multinacional TCS Tata Consultancy Services, desempeñando el cargo de Consultor Sap-Abap cumpliendo con la tarea de implementar y modificar requerimientos de software de acuerdo a las necesidades de los clientes de la organización.

La nueva solución de software que se realizó permite al cliente (Operador Móvil) con sucursal en Colombia dar reporte sobre la información de pagos y cobros de pólizas a la sede central ubicada en México.

La funcionalidad del programa consiste en la comunicación de dos Servicios Web uno ubicado en México y el otro en Colombia, este último se comunica con SAP para extraer la información de las pólizas y así poderlas retornar hasta el servicio web ubicado en México.

Palabras Claves: ERP, SAP, Abap, software, servicio web, operador móvil, póliza.

Introducción

A medida que va pasando el tiempo aparecen nuevas tecnologías de la información, y estas se convierten en una herramienta fundamental e indispensable para el crecimiento de las empresas; el buen aprovechamiento de las tecnologías de la información son de gran ayuda a la hora de optimizar y mejorar los procesos empresariales.

Dentro de este abanico tan amplio de tecnologías se encuentra SAP, unos de los ERP más usados a nivel mundial, el cual permite integrar todas las áreas de una organización desde recursos humanos, finanzas, materiales, inventarios, contabilidad, entre otros.

En el presente informe se pretende describir la metodología usada para la creación de una aplicación en el sistema SAP de un Operador Móvil presente en el país y uno de los clientes más fuertes en la región de Tata Consultancy Services; la funcionalidad de esta implementación de software es la de permitir extraer la información de SAP acerca de los pagos y cobros asociados a las pólizas generadas en un servidor web localizado en México, este a su vez se comunica con otro servicio web ubicado en Colombia que es el encargado de ser el intermediario entre las comunicaciones entre el sistema SAP del cliente y el servicio web de México.

Se puede evidenciar que en el desarrollo de este sistema de software hay varias tecnologías implicadas y como son usados los recursos de cada una para mejorar un proceso empresarial.

Justificación

Impacto Tecnológico

Tata Consultancy Services (TCS) es una multinacional de la India cuya actividad económica está centrada la prestación de servicios de tecnología de la información (TI), consultoría, infraestructura e inteligencia de negocios; por ende tiene importantes y exigentes clientes a nivel mundial como Operadores de Telecomunicaciones, Bancos, Entidades Públicas y grandes empresas dedicada a diferentes sectores económicos. TCS hace presencia en nuestro país desde el año 2005 con sedes en Bogotá y en Medellín, actualmente TCS se encuentra en un gran crecimiento.

Debido a la cantidad de servicios que presta TCS se encuentra dividido por verticales para satisfacer las diferentes necesidades del cliente, una de las verticales con mayor número de empleados es la de Telecomunicaciones donde se prestan servicios de desarrollo web en el lenguaje JAVA y consultoría en SAP; SAP es un software de origen alemán que provee un abanico muy amplio de soluciones de software para cada uno de los aspectos de una empresa (CRM, ERP, PLM, SCM, SRM), TCS posee el personal capacitado para ofrecer solución y soporte a los diferentes aplicativos y funcionalidades de SAP, el personal está dividido en tres tipos diferentes de consultores: Basics, Funcionales y Abaps, estos últimos son los encargados de desarrollar ampliaciones a funcionalidades estándar de SAP, modificar y desarrollar nuevos programas para satisfacer una necesidad particular del cliente.

Adquirir el conocimiento y experiencia en SAP es muy conveniente para un ingeniero ya que el mercado de SAP está en crecimiento en Colombia y es muy bien pago y abre puertas para trabajar en exterior.

Objetivos

Objetivo General

Desarrollar requerimiento de software en ABAP (lenguaje nativo de ERP SAP) a partir de una especificación funcional, que permita dar reporte a un Operador Móvil en México sobre la información de pagos y cobros de pólizas del mismo Operador con Sucursal en Colombia.

Objetivos Específicos

- Analizar la especificación funcional para identificar la necesidad concreta que posee del cliente.
- Definir metodología de desarrollo de software, la cual permita dar pronta y satisfactoria solución a la necesidad del cliente.
- Realizar la documentación del desarrollo de software: Especificación Técnica, Manual de Usuario, Manual de Instalación y evidencia de Pruebas, exigidas por el cliente.
- Diseñar y Realizar casos de pruebas para verificar el correcto funcionamiento de la funcionalidad desarrollada.

Marco Teórico

Tata Consultancy Services

Fundada en 1968, TCS ha crecido hasta llegar a su posición actual como la principal empresa de servicios de TI de Asia; TCS Colombia inicio sus operaciones en el año 2005, trabajando de forma conjunta con TCS Uruguay, ese año se gana la licitación de un proyecto muy importante con una empresa de telecomunicaciones, al mismo tiempo que desarrolla el Sistema Integrado de Información Financiera para el Ministerio de Hacienda del gobierno de Colombia.

TCS cuenta con dos sedes en nuestro país una de ellas ubicada en la capital del país Bogotá y la otra en la ciudad de Medellín, entre éstas dos sedes suman más de 1000 recursos humanos que se destacan por sus capacidades técnicas, entusiasmo y compromiso para atender las necesidades de los clientes. (Tata Consultancy Services, s.f)

Identidad Corporativa

Misión

- Ayudas a los clientes a alcanzar sus objetivos de negocio proporcionando innovación, mejor clase de consultoría, soluciones TI y servicios.
- Que sea una alegría para todos interesados trabajar con nosotros.

Valores Empresariales

- Liderar el cambio
- Integridad

- Respeto por el individuo
- Excelencia
- Aprender y compartir (Tata Consultancy Services, s.f)

ERP (Enterprise Resource Planning)

Es un sistema integral de gestión empresarial que está diseñado para mejorar y automatizar los procesos empresariales (ventas, finanzas, producción, logística, etc.). Su objetivo principal es la planificación de todos los recursos de la organización, desde recursos humanos hasta materiales e insumos.

La funcionalidad primordial de un ERP es la de unificar y ordenar toda la información de la empresa en un solo lugar, por ende cualquier movimiento de ésta se puede observar inmediatamente por cualquier área de la compañía, ayudando a la toma de decisiones de una manera más rápida y eficiente.

Con la implementación de un ERP se tendrá la empresa bajo control y se incrementara la calidad de los servicios y productos que presta la empresa, además de eliminar las barreras entre los departamentos permitiendo que la información fluya y sea constante entre cada uno de las áreas.

Tipos de ERP

Existen dos tipos de ERP, el software a medida y la solución estándar.

Software a medida

Es un sistema creado específicamente para una empresa, el desarrollo comienza desde cero, la implementación es larga, costosa y el riesgo es alto debido a que si le llega a pasar algo a la consultora creadora del software se pierde toda la parte de soporte; una

de sus mayores ventajas es que la aplicación creada se adapta perfectamente al negocio y un porcentaje mayor al 90% de las funcionalidades serán de gran ayuda para la empresa.

Soluciones estándar

Son aplicaciones más baratas y fáciles de implementar, se puede comprar por módulos por ejemplo: el módulo de compras, finanzas, recursos humanos, etc. Una de sus ventajas es el soporte, se puede contratar fácilmente una consultora para brindar éste servicio; la dificultad más grande es definir cuál es solución adecuada para empresa.

SAP

Es el ERP tipo estándar más utilizado por las grandes empresas a nivel mundial ente un 60% y 70% usan éste sistema para integrar todos los procesos dentro de la organización, pero SAP no solo se enfoca en las grandes empresas, también tiene soluciones para las pequeñas y medianas empresas.

Cuando una empresa decide implementar SAP se instalan programas y transacciones estándar que pueden ser modificados y adaptados a las necesidades de la organización; también se pueden crear nuevas funcionalidades por medio del lenguaje de programación nativo de SAP que es ABAP o utilizar JAVA.

Tipos Consultores SAP

Existen tres tipos de consultores:

Consultor Basis

Son los encargados de la instalación, mantenimiento y optimización de los servidores SAP y las bases de datos.

Consultor Funcional

Son los responsables de la personalización y parametrización según las necesidades del cliente, también son los encargados de crear las especificaciones funcionales de las que se basan los consultores Abap para crear y adaptar los programas de acuerdo a la demanda del cliente.

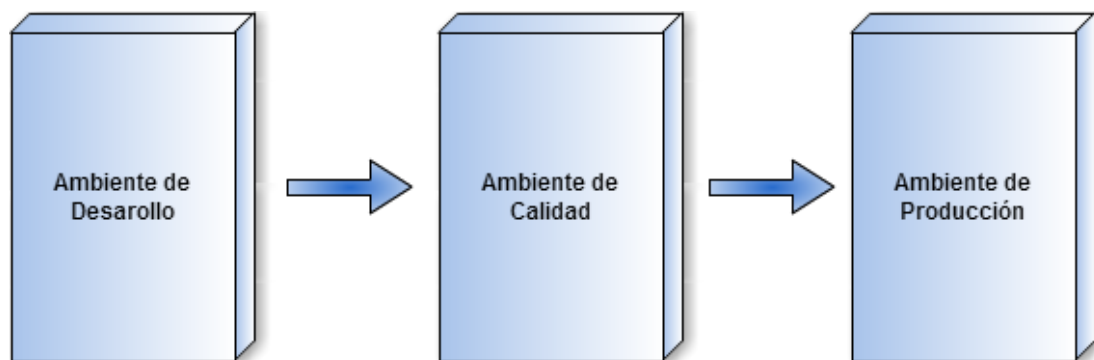
Consultor Abap

Son los responsables de la codificación Abap.

Estructura SAP

En general los sistemas SAP están compuestos por tres ambientes, desarrollo, calidad y producción:

Ilustración 1. Estructura SAP



Ambiente de Desarrollo

En este ambiente se realizan los desarrollos de aplicaciones y parametrizaciones del sistema, para que los programas y objetos creados o modificados puedan ser visibles en los demás ambientes se crea una orden de transporte.

Ambiente de Calidad

Los objetos tratados en desarrollo pasan calidad por medio de la orden de transporte; en éste ambiente, se realizan las pruebas integrales para verificar el correcto funcionamiento de los programas y parametrizaciones.

Ambiente de Producción

En este ambiente están los datos reales y es donde opera la empresa que tiene el sistema SAP.

ABAP

Lenguaje de cuarta generación el cual se usa para adaptar y crear nuevas funcionalidades en el sistema SAP para satisfacer las necesidades de la empresa. Es un lenguaje basado en eventos, es decir, que existen eventos que condicionan la secuencia de ejecución del programa.

Permite tener acceso directo a la base de datos mediante la utilización de OPEN SQL; es una herramienta que permite la creación de aplicaciones de diferente tipo como reportes, reportes interactivos, aplicativos web, formularios, etc.

Objetos ABAP

Este lenguaje permite crear diferentes tipos de objetos, los cuales permite tener diferentes funcionalidades dentro del sistema SAP.

Módulos de Función

Son objetos que realizan operaciones que pueden ser utilizadas en varios programas. Para utilizar la funcionalidad presente en un módulo de función en diferentes programas solo se necesita hacer un llamado al módulo en cada uno de los programas, lo anterior permite la reutilización y el fácil mantenimiento del código.

Dentro de SAP se encuentra un gran número de módulos de funciones estándar que son agrupados según su funcionalidad en grupos de funciones, además de esto se pueden crear módulos y grupos de funciones que son desarrollados a medida; los módulos que estén agrupados en el mismo grupo de función comparten las mismas definiciones de datos globales.

Módulo de Función RFC

Sirve para intercambiar información entre SAP y un sistema externo, éste sistema puede estar codificado en cualquier otro lenguaje.

Formularios

En SAP se manejan dos tipos de formularios, los cuales permiten el formato de impresión a los documentos creados dentro de los programas.

Sapscript

Por medio de un sapscript se puede definir el formato de las páginas y la estructura de la información que contiene.

Smart Form

Es una herramienta de impresión de formularios fácil de utilizar y muy sólida, permite la creación y modificación de formularios mediante la utilidad de diseño gráfico.

Web Dynpro

Se utiliza para desarrollar interfaces de usuario para aplicaciones profesionales del negocio, se basa en la metodología Modelo Vista Controlador (MVC) que permite que la lógica del negocio este separada de la presentación del negocio.

Clase de Mensajes

Permite almacenar diferente tipos de mensaje (Éxito, Error, Advertencia), estos mensajes se pueden acceder desde cualquier otro objeto ABAP como programas, módulos de función, clases globales y locales, etc. Cada mensaje es identificado por un Id único el cual permite retornar el mensaje.

Tablas Internas

Su principal funcionalidad es el procesamiento de datos en ABAP, hacen la función de los arrays en otros lenguajes de programación.

Su uso es el de almacenar en memoria los datos de una tabla de base de datos durante la ejecución de un programa ABAP, por ende, las tablas internas son memoria dinámica, es decir, son creadas en tiempo de ejecución del programa en memoria RAM y desaparece una vez ha terminado la ejecución.

Tipos de Tablas Internas

Any Table

Permiten definir una tabla interna con un tipo de parámetro genérico y realizar todas las operaciones permitidas para tablas internas.

Index Table

Es una tabla indexada a la cual puedes acceder mediante un índice, también es posible especificar en un tipo de parámetro genérico.

Standard Table

Para acceder a este tipo de tablas se hace mediante una búsqueda lineal, esto quiere decir que el tiempo en que se demora la búsqueda está relacionado con la cantidad de registros.

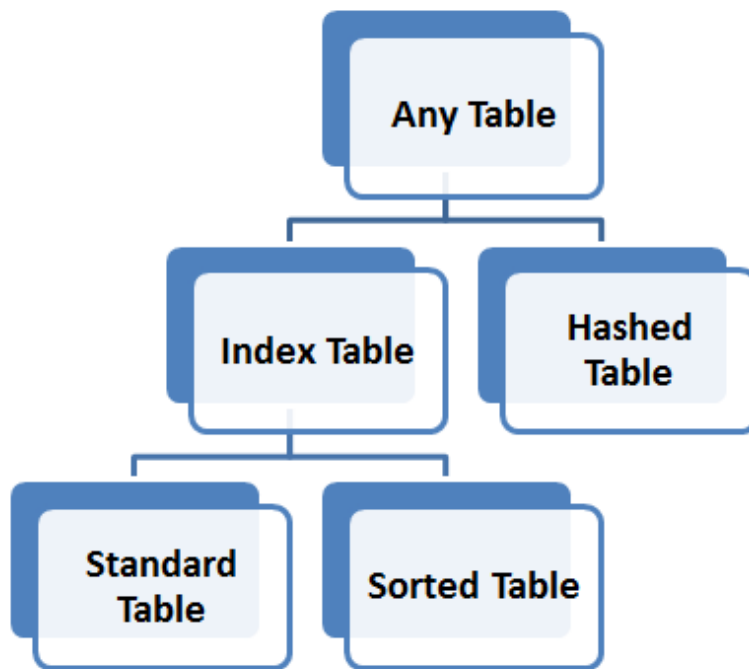
Sorted Table

Es una tabla que esta ordenada ascendentemente por su clave primaria, el acceso a este tipo de tabla se realiza mediante búsqueda binaria, el tiempo de demora de la consulta está directamente relacionado con la cantidad de registros que contiene la tabla.

Hashed Table

Todos los registros deben tener una clave única, donde se utiliza esta clave para poder acceder a la tabla, el tiempo que demora la consulta es constante, sin tener en cuenta la cantidad de registros en la tabla.

Ilustración 2. Tablas Internas



Diccionario de Datos

Es la utilidad y almacén central del entorno de desarrollo de ABAP, su función es la de proporcionar la definiciones de los datos y las relaciones de la información usadas en las aplicaciones.

Programación Orientada a Objetos

Facilita la reutilización, organización y mantenimiento del código.

Clase

Es la definición de las propiedades y comportamiento de un objeto; el comportamiento está dado por medio de los métodos los cuales definen la funcionalidad del programa.

Objeto

Es una instancia de una clase, esta se puede ver como una variable declarada del tipo de dato de la clase, y puede tomar todas las características y comportamiento de la clase.

Herencia

Permite la creación de nuevas clases a partir de unas ya existentes, donde las nuevas clases heredan los datos y métodos de la superclase, estos métodos se pueden sobrescribir e incluso agregar métodos nuevos.

Metodología

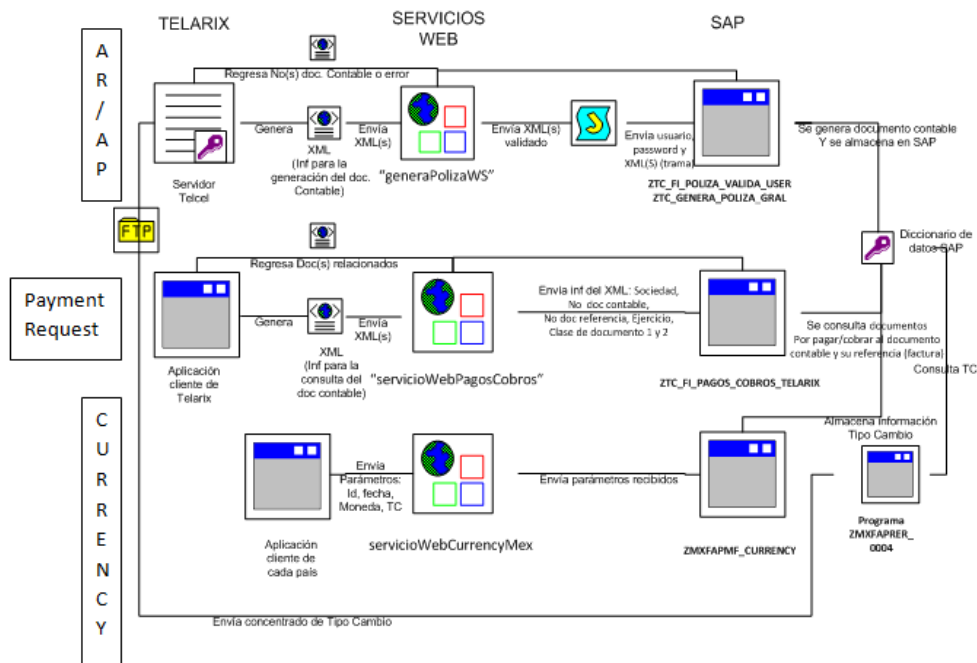
Durante el proceso de prácticas empresariales en Tata Consultancy Services se ha dado apoyo en el proyecto Telarix con el fin de cumplir los objetivos planteados desde el inicio de las prácticas, este proyecto surge para suplir la necesidad de software solicitada por el cliente (Operador Móvil).

Telarix es un proyecto que permite la comunicación e intercambio de información entre la sede principal del Cliente localizada en México y la sucursal de la misma en Colombia; el proyecto está dividido en tres interfaces diferentes que cumplen con una función específica cada una de ellas; las interfaces son: interfaz AR/AP, Payment Request y Currency; para la creación de cada interfaz se asignaron 3 Consultores Abap, donde soy el encargado de desarrollar la funcionalidad de la interfaz Payment Request.

La funcionalidad de cada interfaz es la siguiente:

- Interfaz AR/AP: es la encargada del proceso de contabilización y de la generación de las pólizas de cobros y pagos.
- Interfaz Payment Request: se encarga de alimentar el servidor Telarix con la información asociada a pagos y cobros de las pólizas creadas en la interfaz AR/AP.
- Interfaz Currency: diariamente consulta la tasa de cambio de dólar a peso colombiano y la envía al servidor Telarix.

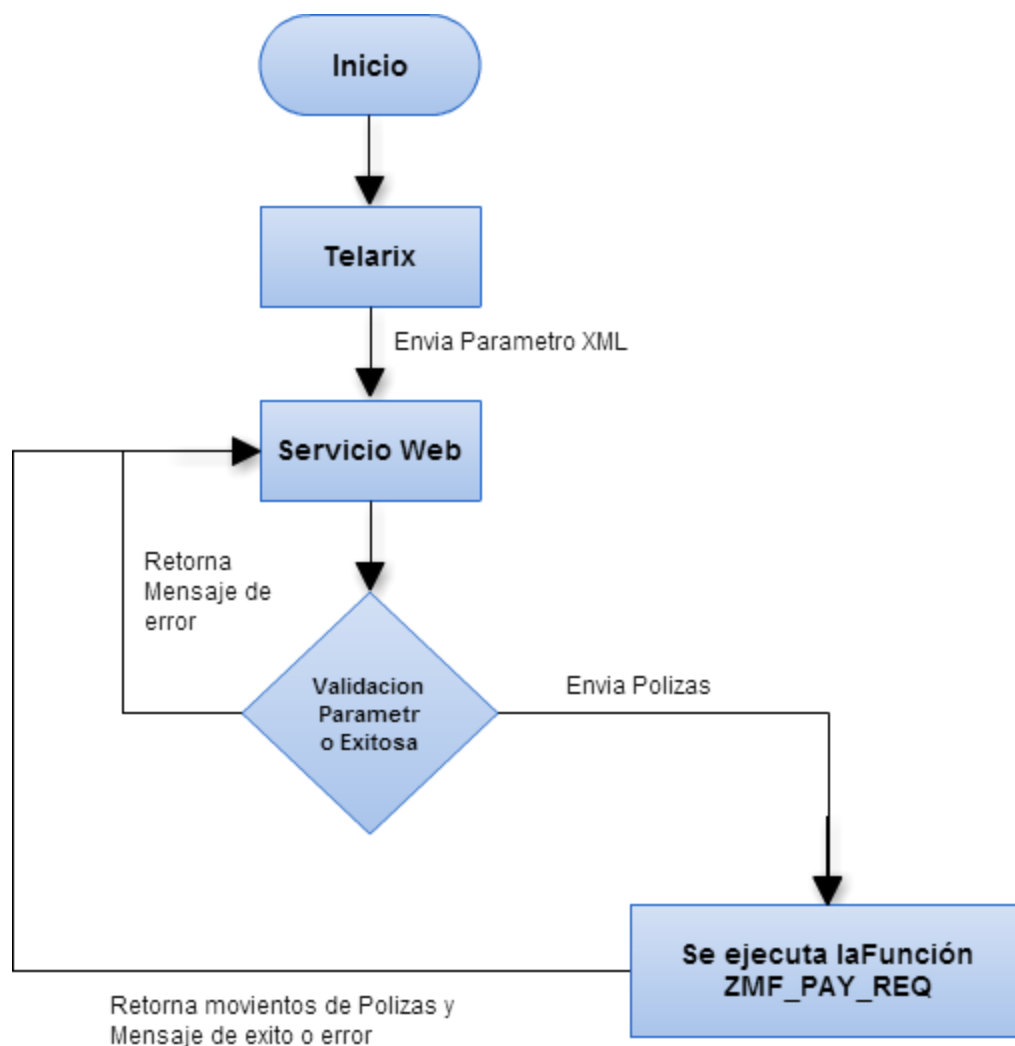
Ilustración 3. Interacción entre las tres interfaces.



Interfaz Payment Request

La interfaz Payment Request consiste en la realización de la consulta del documento contable generado en la interfaz AR/AP en el ERP local (SAP) y determinar el documento de compensación con el que se haya cruzado (con sus respectivos importes pagados/cobrados y por pagar/por cobrar), cada consulta se realiza hasta encontrar una compensación del 100% del documento, para realizar la consulta del documento contable Telarix envían un archivo XML a un Web Service(SW) con la información del documento contable, el SW es el encargado de la comunicación con SAP, en SAP se ejecuta la consulta donde se obtiene información del documento de compensación, esta consulta es realizada por un módulo de funciones; se retorna la información del documento de compensación y un mensaje ya sea de éxito o error a Telarix por medio del SW.

Ilustración 4. Interfaz Payment Request



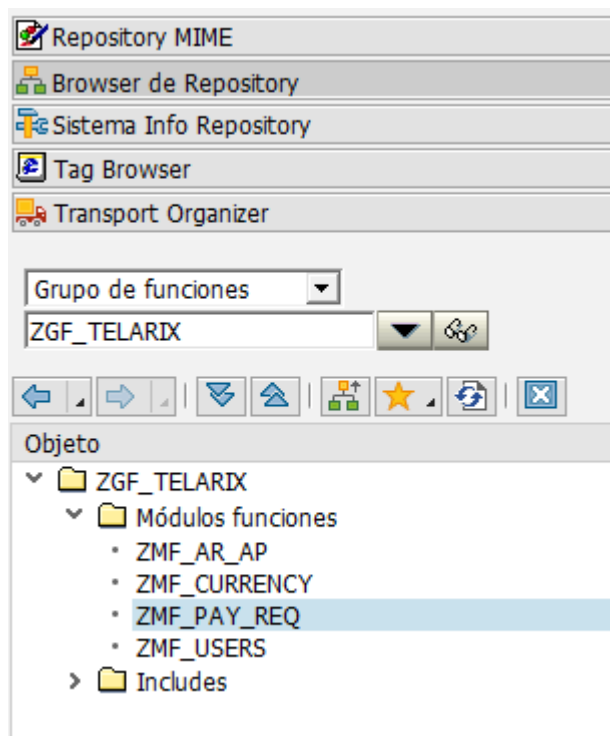
Objetos Creados

Son varios los objetos que se crearon para satisfacer el requerimiento del cliente para las tres interfaces, pero solo se va a tener en cuenta los objetos creados para la interfaz Payment Request, a continuación se van a nombrar cada uno:

Grupo de funciones.

En este grupo se almacena cada módulo de funciones de cada interfaz.

Ilustración 5. Grupo de Funciones



Módulo de Funciones

La función que cumple este módulo de funciones es de realizar la comunicación con Servicio Web (SW) local, recibiendo los números de pólizas y retornando al SW los movimientos encontrados en SAP con relación a las pólizas recibidas y el mensaje de éxito o error según el caso.

La lógica del módulo consiste en hacer el llamado a los métodos de una clase global que es donde se desarrolla la validación de parámetros y consulta de movimientos.

Ilustración 6. Módulo de Funciones (lógica).

```

1  function zmf_pay_req.
2  """-----
3  """Interfase local
4  """ IMPORTING
5  """     VALUE(I_DOCS) TYPE  CKMLI_MGV_STATISTIC
6  """ EXPORTING
7  """     VALUE(ET_MOVS) TYPE  ZTY_MOVS_PAY_REQ
8  """     VALUE(E_RESULTADO) TYPE  STRING
9  """-----
10
11     data: lo_pay_req type ref to zcl_pay_req.
12
13     *Instancia el objeto
14     create object lo_pay_req.
15
16     call method lo_pay_req->import_data
17         exporting
18             i_docs = i_docs.
19
20     call method lo_pay_req->process.
21
22     call method lo_pay_req->export_data
23         importing
24             e_movs = et_movs
25             e_msg  = e_resultado.
26
27     endfunction.

```

El módulo está compuesto por un parámetro import donde se reciben las pólizas y dos parámetros export donde en uno de ellos retorna una tabla con los movimientos encontrados y en el otro un mensaje informativo.

Ilustración 7. Módulo de Funciones (Import y Export).

The image shows two screenshots of the SAP Import and Export function modules. The top screenshot shows the 'Import' tab with a table of parameters. The bottom screenshot shows the 'Export' tab with a table of parameters.

Parámetro	Tipf...	Tipo ref.	Valor propuesta	Op...	Tr...	Texto breve	Te
I_DOCS	TYPE	CKMLI_MGV_STAT...		<input type="checkbox"/>	<input checked="" type="checkbox"/>	Documentos de entrada interface Tel...	

Parámetro	Tipificac.	Tipo ref.	Trasp.v...	Texto breve	Texto e...
ET_MOVS	TYPE	ZTY_MOVS_PAY_REQ	<input checked="" type="checkbox"/>	Movimientos para interface TELARIX	
E_RESULTADO	TYPE	STRING	<input checked="" type="checkbox"/>	Respuesta de la consulta	

Clase Global

Existe una súper clase de la cual heredan cada una de las clases asociadas a cada interfaz, cada clase utiliza y redefine los métodos que sean necesarios para cumplir con la funcionalidad de cada interfaz.

Ilustración 8. Súper Clase.

The image shows the SAP Class Inspector for the class 'ZCL_TELARIX_SUPER'. The 'Métodos' tab is selected, showing a list of methods with their properties and descriptions.

Métodos	Clase	Vis...	Sól...	T...	Descripción
PROCESS	Insta...	Pub...	<input type="checkbox"/>		Procesamiento RFC
WS_CONSUME	Insta...	Pub...	<input type="checkbox"/>		WebService Consume
CONSTRUCTOR	Insta...	Pub...	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Constructor
CALL_BAPI	Insta...	Pro...	<input type="checkbox"/>		Llamado BAPI
FETCH_DATA	Insta...	Pro...	<input type="checkbox"/>		Obtener datos
VALIDATE	Insta...	Pro...	<input type="checkbox"/>		Validaciones
SET_LOG	Insta...	Pro...	<input type="checkbox"/>		Manejo de Log

En la clase global donde se implementó la funcionalidad de la interfaz Payment Request se crearon cinco métodos donde cada uno permitía la ejecución de las diferentes funcionalidades requeridas por el cliente.

Ilustración 9. Clase Global Payment Request.

Interfase clases		ZCL_PAY_REQ		Realizado / activo			
Propiedades	Interfaces	Friends	Atributos	Métodos	Eventos	Tipos internos	Alias
<div style="display: flex; justify-content: space-between; align-items: center;"> Parámetros Excepcio... [Iconos] Filtro </div>							
Métodos	Clase	Vis...	Sól...	T...	Descripción		
PROCESS	Insta...	Pub...	<input type="checkbox"/>		Procesamiento RFC		
WS_CONSUME	Insta...	Pub...	<input type="checkbox"/>		WebService Consume		
CONSTRUCTOR	Insta...	Pub...	<input type="checkbox"/>	[Icono]	Constructor		
CALL_BAPI	Insta...	Pro...	<input type="checkbox"/>		Llamado BAPI		
FETCH_DATA	Insta...	Pro...	<input type="checkbox"/>		Obtener datos		
VALIDATE	Insta...	Pro...	<input type="checkbox"/>		Validaciones		
SET_LOG	Insta...	Pro...	<input type="checkbox"/>		Manejo de Log		
IMPORT_DATA	Insta...	Pub...	<input type="checkbox"/>		Import data		
EXPORT_DATA	Insta...	Pub...	<input type="checkbox"/>		Export data		

Métodos

Import_Data

La información proveniente del servicio web local es almacenada en una tabla interna para realizar las consultas de las pólizas con los registros almacenados en esta. La tabla interna es utilizada como atributo de la clase para poder ser utilizada en los demás métodos.

Ilustración 10. Método Import_Data.

Cl.	Parámetro	Tipificac.	Descripción
pa	I_DOCS	TYPE CKMLI_MGV_STATISTIC	Interfase interna: tabla estadística (aplicación)

Método	IMPORT_DATA	Actv.
<pre> 1 method import_data. 2 data: tl_convertida1 type standard table of string, 3 tl_convertida2 type truxs_t_text_data. 4 5 field-symbols: <fs_docs> type zst_docs_pay_req. 6 7 *Se limpian las tablas para convertir la cadena de caracteres en una tabla 8 clear: tl_convertida1, tl_convertida2. 9 10 *Se convierte la cadena de caracteres en una tabla interna 11 if i_docs is not initial. 12 split i_docs at '#' into table tl_convertida1. 13 if sy-subrc eq 0. 14 tl_convertida2 = tl_convertida1. 15 endif. 16 17 *Se convierte la tabla en un tipo en el cual se pueda seguir trabajando 18 call function 'TEXT_CONVERT_TEX_TO_SAP' 19 exporting 20 i_field_seperator = ' ' "cl_abap_char_utilities=>horizontal_tab 21 i_tab_raw_data = tl_convertida2 22 tables 23 i_tab_converted_data = t_docs 24 exceptions 25 conversion_failed = 1 </pre>		

Validate

En este método se valida que la tabla interna que contiene los registros de las pólizas no este vacía de estarlo se retorna un mensaje de error al servicio web, además de validar que cada campo de la tabla contenga la información correcta.

Ilustración 11. Método Validate.

Cl.	Parámetro	Tipificac.	Descr...

Método	VALIDATE	Actv.

```

1  method validate.
2
3      field-symbols: <fs_movs_log> type ztt_movs_pay_req.
4      data: tl_polizas_movs type zty_movs_pay_req_log.
5
6      * Verifica si se envian documentos
7      if t_docs is initial.
8          rc = 1.
9          exit.
10     endif.
11
12     * Elimina documentos que no pertenezcan a la Sociedad INFR
13     delete t_docs where bukrs ne c_bukrs.
14     if sy-subrc = 0 and t_docs is not initial.
15         rc = 2.
16     endif.
17
18     *Se verifica si se han realizado consultas con las polizas que envia Telarix
19     if t_docs is not initial.
20
21         select *
22             into table tl_polizas_movs
23             from ztt_movs_pay_req
24             for all entries in t_docs
25             where recordid = t_docs-recordid and
26                 docorigen = t_docs-docorigen.

```

Fetch_Data

Se realiza extracción de la información de las tablas estándar de SAP, esta se hace por medio de la consulta de los registros que se encuentran en la tabla interna en las tablas estándar, si la consulta es exitosa los registros obtenidos son almacenados en otra tabla interna que se usa específicamente para almacenar los movimientos de las pólizas.

Ilustración 12. Método Fetch_Data.

Cl.	Parámetro	Tipificac.	Descr...

Método	FETCH_DATA	Actv.
31	<i>*Consulta informacion documntos de origen</i>	
32	<code>select bukrs belnr gjahr blart bldat xblnr awkey</code>	
33	<code>into table t1_bkpf_ori</code>	
34	<code>from bkpf</code>	
35	<code>for all entries in t_docs</code>	
36	<code>where bukrs = t_docs-bukrs</code>	
37	<code>and belnr = t_docs-docorigen</code>	
38	<code>and gjahr = t_docs-gjahr .</code>	
39	<code>*</code>	
40	<i>**Consulta si el documento existe en la BKPF</i>	
41	<code>select bukrs belnr gjahr blart bldat xblnr awkey</code>	
42	<code>into table t_bkpf</code>	
43	<code>from bkpf</code>	
44	<code>for all entries in t_docs</code>	
45	<code>where bukrs = t_docs-bukrs</code>	
46	<code>and belnr = t_docs-docto</code>	
47	<code>and gjahr = t_docs-gjahr .</code>	
48		
49	<code>if sy-subrc ne 0.</code>	
50	<i>*Si no se encuentran los documentos contables en la BKPF, no existe informacion</i>	
51	<code>rc = 3.</code>	
52	<code>exit.</code>	
53	<code>endif.</code>	
54		
55	<i>*Elimina registros igules entre t_bkpf y t1_bkpf_ori</i>	
56	<code>loop at t_bkpf assigning <fs_bkpf>.</code>	

Alcance \METHOD fetch_data

Set_Log

En este método se obtiene el mensaje que se va retornar al servicio web, este mensaje es almacenado en una tabla estándar de SAP, la cual cumple la función de ser un log, donde el usuario puede observar el tipo de mensaje retornado, el mensaje y la fecha.

Ilustración 13. Método Set_Log.

Cl.	Parámetro	Tipificac.	Descr...

Método	SET_LOG	Activ.
10	case rc.	
11	when 0. "RC initial	
12	lv_msgty = msgty_s.	
13	read table lt_t100 assigning <fs_t100> with table key msgnr = '005'.	
14	result_msg = <fs_t100>-text.	
15		
16	when 1.	
17	read table lt_t100 assigning <fs_t100> with table key msgnr = '002'.	
18	result_msg = <fs_t100>-text.	
19		
20	when 2. ↓	
21	lv_msgty = msgty_s.	
22	read table lt_t100 assigning <fs_t100> with table key msgnr = '006'.	
23	result_msg = <fs_t100>-text.	
24		
25	when 3.	
26	read table lt_t100 assigning <fs_t100> with table key msgnr = '004'.	
27	result_msg = <fs_t100>-text.	
28		
29	endcase.	
30		
31	concatenate lv_msgty result_msg into result_msg separated by '- '.	
32		
33	call method o_log->set	
34	exporting	
35	i msgty = lv msgty	

Alcance \METHOD set_log

Export Data

Por medio de este método se retorna la tabla interna al servicio web con la información de los movimientos de pago o cobro asociado a las pólizas consultadas, también es retornado el mensaje informativo.

Ilustración 14. Método Export_Data.

Cl.	Parámetro	Tipificac.	Descripción
▣▶	E_MOVS	TYPE ZTY_MOVS_PAY_REQ	Movimientos para interface TELARIX
▣▶	E_MSG	TYPE STRING	Mensaje Retorno

Método EXPORT_DATA Actv.

```

1  method export_data.
2      e_movs = t_movs.
3      e_msg = result_msg.
4  endmethod.

```

Clase Mensajes

En esta clase se encuentran almacenados los mensajes retornados por cada una de las interfaces del proyecto Telarix, para acceder a cada mensaje se hace por medio del ID del mensaje.

Ilustración 15. Clase de Mensajes.

Clase de mensajes ZCM_TELARIX actv.

Propiedades Mensajes

Mensaje	Texto breve mensaje	Autoexplicativo
000	%1 %2 %3 %4	<input checked="" type="checkbox"/>
001	Datos Inválidos	<input checked="" type="checkbox"/>
002	E-Campos obligatorios	<input checked="" type="checkbox"/>
003	E-No hay cuentas en los bancos	<input checked="" type="checkbox"/>
004	E-No existe información	<input checked="" type="checkbox"/>
005	S-Existe información para todos los documentos	<input checked="" type="checkbox"/>
006	S-Existe información pero no para todos los documentos	<input checked="" type="checkbox"/>
010	E-Parámetro identificador de subsidiaria obligatorio	<input checked="" type="checkbox"/>
011	E-Parámetro fecha obligatorio	<input checked="" type="checkbox"/>
012	E-Parámetro moneda origen obligatorio	<input checked="" type="checkbox"/>
013	E-Parámetro moneda destino obligatorio	<input checked="" type="checkbox"/>
014	E-Parámetro tipo de cambio obligatorio	<input checked="" type="checkbox"/>
015	Sin Tipo Cambio para los parámetros enviados	<input checked="" type="checkbox"/>
016	Error al consumir el Servicio Web	<input checked="" type="checkbox"/>
050	E-Usuario inválido - %	<input checked="" type="checkbox"/>

Documentación

Dentro de la metodología utilizada en el desarrollo de requerimientos de software para el cliente (Operador Móvil) se realizan documentos para servir como guía o soporte para los programas creados, los documentos que se deben crear son impuestos por el cliente, que también dice los estándares que estos documentos deben llevar, estos son los documentos pedidos a la hora de desarrollar un programa:

Arquitectura y Diseño

Es creado por un Consultor Funcional, la función de este documento es explicar la funcionalidad, arquitectura y diseño del sistema el cual se pide desarrollar, donde se especifica el flujo del programa, elementos y componentes involucrados, modelo de base de datos y especificaciones de seguridad.

Manual Técnico

El Consultor Abap implementa el requerimiento de software es el encargado de la creación de este documento, el contenido de este corresponde a la descripción de cada uno de los objetos creados en el desarrollo del software, relación y funcionalidad de los objetos creados, descripción de los parámetros de entrada y salida.

Manual de Usuario

Creado por el Consultor Abap, se explica la funcionalidad de la aplicación creada y se especifica los pasos para que el usuario pueda ejecutar la función del programa.

Documento de Pruebas

Una vez terminado el desarrollo el Consultor Abap realiza las pruebas respectivas para verificar la funcionalidad del programa, estas pruebas son documentadas y enviadas al cliente para que realice sus pruebas.

Resultados Y Análisis De Los Resultados

El proyecto de Telarix se culminó satisfactoriamente, actualmente las interfaces desarrolladas se encuentra en el ambiente de calidad, donde a cada interfaz se realizó por parte del cliente la ejecución de las pruebas pertinentes para verificar el correcto funcionamiento, dichas pruebas resultaron satisfactorias para todas las interfaces incluyendo Payment Request.

En estos momentos se está a la espera de la programación por parte del cliente de la fecha para realizar el paso al ambiente de productivo del proyecto.

Conclusiones

- La utilización de las tecnologías de la información (TI) a cada día son más importantes y esenciales para el crecimiento de las organizaciones, ya no es un lujo contar con un área de TI, es una obligación.
- La integración de los procesos empresariales por medio de un ERP le aporta a la empresa beneficios económicos, calidad, competitividad, eficiencia y eficacia.
- La integración y utilización de diferentes tecnologías o lenguajes de programación permite sacarle al máximo al provecho a sus beneficios y así crear una aplicación para la satisfacción de las necesidades empresariales del cliente.
- La realización de las prácticas empresariales en TCS me permitió obtener nuevos conocimientos técnicos y humanos que me han ayudado a crecer como persona y profesionalmente.
- La realización de este proyecto permitió explotar y utilizar los conocimientos adquiridos a través de los semestres en la universidad

Recomendaciones

La etapa de prácticas fue exitosa el conocimiento adquirido es indispensable para el crecimiento como persona y profesional, pero sería bueno que dentro del pensum de Ingeniería Informática donde se ofrecen las materias electivas sea incluida una donde a los estudiantes se les dé un acercamiento más a fondo sobre un ERP específico como SAP, estos conocimientos ayudaran a los estudiantes a salir más preparados al mercado laboral.

Bibliografía

Ehret, Stefan. (2012). *Taw10_1 Fundamentos de Workbench Abap*. Walldorf: SAP AG.

Ehret, Stefan. (2012). *Taw12_1 Objetos Abap y áreas de aplicación*. Walldorf: SAP AG.

Ehret, Stefan. (2012). *Taw11 Detalles Abap*. Walldorf: SAP AG.

Ehret, Stefan. (2012). *Taw12_2 Objetos Abap y áreas de aplicación*. Walldorf: SAP AG.

Tata Consultancy Services. (s.f.). *Colombia*. Recuperado de <http://www.tcs.com/worldwide/es/es/colombia/Pages/default.aspx>

Tata Consultancy Services. (s.f.). *Corporate Facts*. Recuperado de http://www.tcs.com/about/corp_facts/Pages/default.aspx