



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

---

---

FACULTAD DE INGENIERÍA

**“PROGRAMA ABAP PARA VENTA DE  
REFACCIONES”**

**INFORME DE TRABAJO PROFESIONAL**

QUE PARA OBTENER EL TÍTULO DE:  
**INGENIERO EN COMPUTACIÓN**

PRESENTA:  
**BRENDA IVONNE ZEPEDA GUTIÉRREZ**



**Asesor Académico:**  
Ing. Cruz Sergio Aguilar Díaz

CIUDAD UNIVERSITARIA, MÉXICO D.F. Octubre 2015.

# Índice de Contenido

<b>ÍNDICE DE FIGURAS Y TABLAS</b>	<b>4</b>
<b>INTRODUCCIÓN</b>	<b>5</b>
<b>CAPÍTULO 1. MARCO DE REFERENCIA</b>	<b>7</b>
BREVE HISTORIA DE LA EMPRESA	7
PRODUCTOS	7
ORGANIGRAMA	8
EVOLUCIÓN Y CONCEPTO DEL LOS SISTEMAS ERP	9
PANORAMA GLOBAL DE LA EMPRESA SAP	12
HISTORIA DE LA EMPRESA SAP – LÍNEA DE TIEMPO	12
DEFINICIÓN Y DESCRIPCIÓN DEL SISTEMA SAP	15
INGRESO AL SISTEMA SAP	18
DEFINICIÓN DEL LENGUAJE DE PROGRAMACIÓN ABAP	19
<b>CAPÍTULO 2. DESCRIPCIÓN DE ACTIVIDADES PROFESIONALES</b>	<b>20</b>
EXPERIENCIA PROFESIONAL.	20
ACTIVIDADES COMO DESARROLLADOR ABAP	21
PARTICIPACIÓN EN PROYECTOS DE LA EMPRESA DONDE LABORO.	21
<b>CAPÍTULO 3. DESCRIPCIÓN DE PROYECTO: AGENCIAS PROPIAS</b>	<b>23</b>
PROBLEMÁTICA INICIAL	23
SOLUCIÓN REQUERIDA	23
OBJETIVOS ESPERADOS	23
DESCRIPCIÓN DEL PROCESO	24
FLUJO DEL PROCESO	28
<b>CAPÍTULO 4 RESULTADOS</b>	<b>35</b>
PRUEBAS	35
RESUMEN DE RESULTADOS	35
<b>CONCLUSIONES</b>	<b>37</b>
<b>GLOSARIO</b>	<b>39</b>
<b>BIBLIOGRAFÍA:</b>	<b>43</b>

<b>REFERENCIAS ELECTRÓNICAS:</b>	<b>44</b>
<b>ANEXOS</b>	<b>45</b>
<b>ANEXO A – SISTEMA SAP</b>	<b>45</b>
<b>ANEXO B - DESCRIPCIÓN DEL ENTORNO DE DESARROLLO ABAP</b>	<b>45</b>
<b>ANEXO C – INGENIERÍA DE SOFTWARE</b>	<b>45</b>

## Índice de Figuras y Tablas

### Índice de Figuras

<i>Figura 1. Organigrama del área de Sistemas SAP.</i>	8
<i>Figura 2. Evolución de los Sistemas Integrados de Gestión.</i>	11
<i>Figura 3. Evolución de la Empresa SAP.</i>	14
<i>Figura 4. Evolución de Sistemas SAP.</i>	16
<i>Figura 5. Versión de SAP ERP en la empresa.</i>	17
<i>Figura 6. Ejemplo de SAP Logon.</i>	18
<i>Figura 7. Inicio de Sesión de SAP.</i>	18
<i>Figura 8. Lenguaje Orientado a Eventos.</i>	19
<i>Figura 9. WEBGUI para el ambiente de Desarrollo.</i>	24
<i>Figura 10. Inicio de Sesión en Sistema de Agencias Propias.</i>	25
<i>Figura 11. Perfil ADMIN.</i>	25
<i>Figura 12. Perfil AUTORCC.</i>	25
<i>Figura 13. Perfil ALMACEN.</i>	26
<i>Figura 14. Perfil VENTAS.</i>	27
<i>Figura 15. Generación de Venta.</i>	28
<i>Figura 16. Procesos en la opción “Generar Venta”.</i>	29
<i>Figura 17. Procesos en la opción “Surtir Pedido”.</i>	30
<i>Figura 18. Procesos en la opción “Cancelar Pedido”.</i>	31
<i>Figura 19. Pantalla de Facturación.</i>	32
<i>Figura 20. Procesos en la opción “Crear Factura”.</i>	32
<i>Figura 21. Pantalla de Facturación Electrónica.</i>	33
<i>Figura 22. Pantalla de Devoluciones.</i>	33
<i>Figura 23. Pantalla de Proceso de Devolución.</i>	34
<i>Figura 24. Pantalla de Reportes de Ventas.</i>	34

### Índice de Tablas

<i>Tabla 1. Modelos de las motocicletas.</i>	7
<i>Tabla 2. Versiones de SAP R/3.</i>	16
<i>Tabla 3. Versiones de SAP ERP 6.0.</i>	17

## Introducción

El entorno empresarial ha cambiado de forma importante en los últimos años, estos cambios suelen relacionarse principalmente con la creciente globalización de la economía, la consecuente internacionalización de los mercados y la rapidez del cambio tecnológico.

Estas circunstancias han originado un aumento de la competencia entre empresas que las ha conducido a tener que adaptar sus sistemas de información para poder sobrevivir, mantener su posición en el mercado e incluso obtener una ventaja competitiva que les permita diferenciarse del resto.

Las empresas han ido incorporando las nuevas tecnologías de la información, que se han convertido en un instrumento clave para una gestión eficaz y eficiente, básicamente, por su capacidad de proceso de grandes cantidades de información en muy poco tiempo.

Anteriormente cada área o sector de una empresa solía contar con uno o más sistemas de información para su gestión, los cuales se han ido sustituyendo por uno sólo, un sistema de planificación de recursos empresariales (ERP, Enterprise Resource Planning) que integra todas las funciones e información de los diversos departamentos o áreas en una única base de datos común.

De este modo, cuando un dato es introducido en el sistema ERP, inmediatamente está disponible para ser empleado por cualquier área y así automatizan muchas de las prácticas asociadas con los aspectos operativos o productivos de la empresa.

Uno de los sistemas ERP es SAP (Systeme, Anwendungen und Produkte in der Datenverarbeitung - Sistemas, Aplicaciones y Productos en Procesamiento de Datos), que está constituido por diversos módulos, mediante los cuales lleva a cabo la gestión de los procesos de negocios de las empresas que lo utilizan.

SAP tiene su propio lenguaje de programación, ABAP (Advanced Business Application Programming - Programación de Aplicaciones de Negocios Avanzado), con el cual brinda la oportunidad de desarrollar nuevas funcionalidades en el sistema SAP y/o adecuar las existentes de modo que cumplan con las necesidades que la empresa requiere.

El objetivo del presente informe es dar a conocer la **experiencia profesional** adquirida en el área de desarrollos SAP mediante el lenguaje de programación ABAP, en la empresa que laboro.

El informe abarca cuatro capítulos. El primer capítulo es un marco de referencia sobre la historia de la empresa y la estructura del área de sistemas, así como un panorama general sobre los sistemas ERP, la historia y desarrollo de SAP, como empresa y como software, y el lenguaje de programación ABAP.

El segundo capítulo detalla las actividades realizadas durante más de cuatro años como consultor y desarrollador en aplicaciones ABAP.

El capítulo tres explica el proyecto desarrollado para el manejo del funcionamiento de ventas en las agencias propias, que son puntos de venta y servicios propios de la empresa donde trabajo.

En el cuarto capítulo muestro los resultados obtenidos con el desarrollo realizado en comparación con los objetivos planteados inicialmente y de qué manera ha impactado en el funcionamiento del negocio. De igual forma expongo los conocimientos adquiridos durante mi participación.

## Capítulo 1. Marco de Referencia

Para comprender mejor el desarrollo realizado daré un panorama general de la empresa donde laboro.

### Breve Historia de la empresa

La empresa inició operaciones en el año 2004. Se dedica a ensamblar y comercializar motocicletas y moto partes en México, Guatemala, Honduras, Panamá, Perú, Brasil y Costa Rica.

En 2008 se funda la planta de ensamblaje localizada en el parque industrial Toluca 2000 en la ciudad de Toluca, Estado de México, la cual tiene una capacidad de producción de 2100 motocicletas por día y al año más de 350,000, logrando así ser la planta ensambladora más grande en México y Latinoamérica. Tiene una superficie de instalaciones de 75,000 m<sup>2</sup> y cuenta con 3 líneas de ensamble.

En 2010 logra el 55% del mercado de motocicletas en México y en 2011 alcanza el 70%, en gran parte por su bajo costo inicial.

Su misión es dar acceso a un medio de transporte propio, que permita a sus clientes ser más eficientes en sus actividades diarias de una forma cómoda, confiable y divertida.

Inicialmente las motocicletas sólo se comercializaban en tiendas propias de la empresa, pero desde 2011 comenzaron a venderse en tiendas de autoservicio además de distribuidores externos e internos. Estos últimos son conocidos como Agencias Propias, donde también se pueden encontrar accesorios y refacciones.

### Productos

Cuenta con una amplia gama de modelos de motocicletas en sus diversas categorías:

Estilo	Modelos
De Trabajo	FT250, FT200 AMARILLA, FT180 CON LED, FT150 GTI, FT150 ROJA, FT150 DELIVERY, FT150, FT150 S, FT125 DELIVERY, FT125 ROJA, FT125 CLASICA, FT125 SPORT, AR110, AT110, AR110RT CON LED.
Motoneta	GSC175, WS175, TRN150, GS150 LED ELITE, WS150, DS150 NEGRO, XS125, CS125 LED, VITALIA125, VGO125.
Línea Z	250Z, 250SZ, 150Z.
Deportiva	RT200 2015.
Doble Propósito	DM150, DM200.
Cuatrimoto	ATV150 C/REVERSA, ATV250 C/REVERSA.
Chopper	TC200, TC250, RC150 GT.

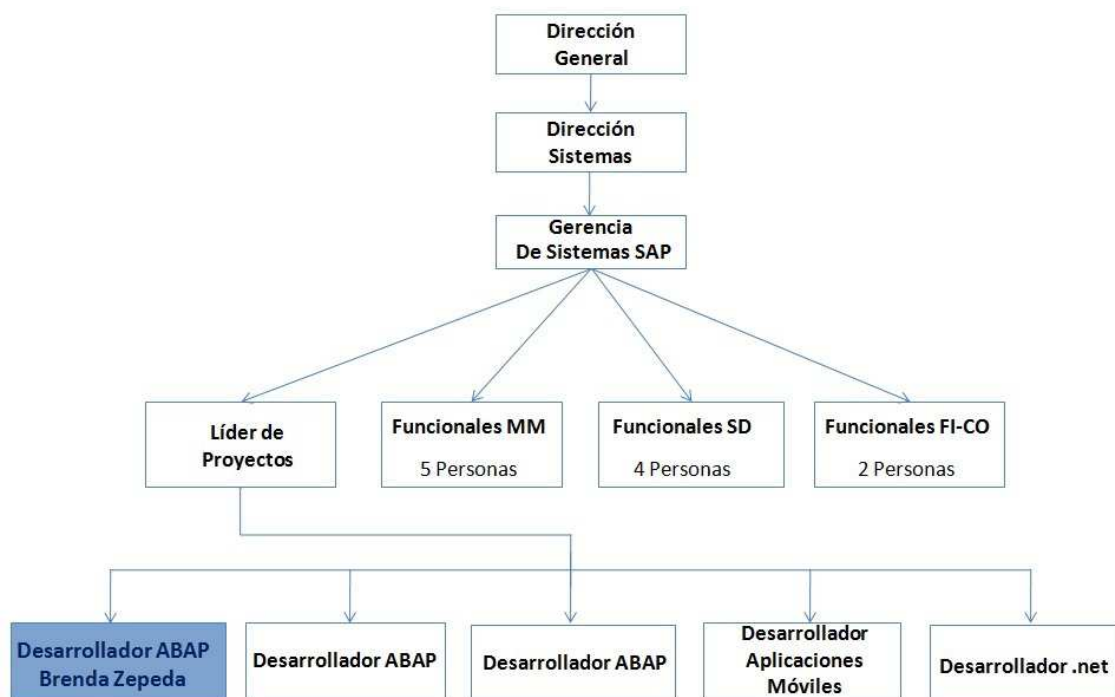
**Tabla 1. Modelos de las motocicletas.**

Además cuenta con las refacciones correspondientes a cada modelo, aceites y accesorios tanto para las motocicletas como para los conductores, como pueden ser: kits de seguridad, pintura, fundas, logos, cascos, rompevientos, playeras, etc.

## Organigrama

La estructura organizacional de una empresa u organización es la división de cargos y responsabilidades que deben cumplir los miembros de la misma, es decir, la forma en que se va a gestionar.

En la figura 1, se muestra el organigrama del área de sistemas SAP en el lugar donde trabajo.



**Figura 1. Organigrama del área de Sistemas SAP.**  
Fuente: Elaboración Propia.



## **Evolución y concepto del los sistemas ERP**

Los antecedentes de los ERP comenzaron en Estados Unidos durante la segunda guerra mundial, con el objetivo de apoyar la gestión de los recursos materiales que demandaba el ejército. Fueron llamados MRPS (Material Requirements Planning Systems).

Los ejércitos emplearon programas especializados que se ejecutaban en las enormes y complejas computadoras recién surgidas en el principio de la década de los 40's para controlar la logística y organización de sus unidades en acciones bélicas.

En la década de los 60's, las compañías manufactureras retomaron la idea de MRPS con el fin de gestionar y racionalizar sus inventarios y planificar el uso de recursos acorde a la demanda real de sus productos, por lo que los MRPS evolucionan a MRP (Manufacturing Resource Planning).

En los años 80 la utilización de estos sistemas incluían conceptos como "Just in Time", manejo de la relación con clientes y proveedores, entre otros. Es así como los MRP evolucionan hasta lo que se conoce como MRP II.

Su alcance fue distinto: permitían atender factores relacionados con la planeación de las capacidades de manufactura; un MRP II, a diferencia de los sistemas previos, reconocía que las empresas padecían interrupciones en la operación, cambios súbitos y limitaciones en recursos que iban más allá de la disponibilidad de materiales.

En la década de los 90's las empresas comenzaron a requerir de sistemas que apoyaran la gestión empresarial, cubrieran todas las áreas funcionales de una empresa de forma integrada (finanzas, producción, ventas, etc.), promovieran la eficiencia operativa y sirvieron de soporte a aspectos críticos de la administración.

Así, la industria de software desarrolló aplicaciones para integrar los distintos sistemas MRP I y MRP II, que años más tarde se transformaron en los sistemas empresariales integrados, conocidos actualmente como ERP (Enterprise Resource Planning).

Hasta la presentación en el mercado de los sistemas ERP se solía utilizar un software independiente para cada departamento, dando lugar a múltiples problemas derivados de la dispersión de datos y formatos, que dificultaban la integración de la información.

Un sistema ERP combina la funcionalidad de los distintos programas de gestión en uno solo, basándose en una única base de datos centralizada. Esto permite garantizar la integridad y unicidad de los datos a los que accede cada departamento, evitando que éstos tengan que volver a ser introducidos en cada aplicación o módulo funcional que los requiera.

Los ERP se conforman por paquetes de software que permiten a las empresas evaluar, implementar, automatizar, integrar y gestionar de forma eficiente las diferentes operaciones que se presentan en éstas.

Dichos paquetes disponen de módulos específicos para cubrir las exigencias de cada una de las áreas funcionales de la empresa, de tal manera que crean un flujo de trabajo (workflow) entre los distintos usuarios. Este flujo permite evitar tareas repetitivas y mejora la comunicación en tiempo real entre todas las áreas que integran la empresa.

Podemos definir un sistema ERP como un sistema integrado de software de gestión empresarial, compuesto por un conjunto de módulos funcionales (logística, compras, ventas, finanzas, recursos humanos, etc.) susceptibles de ser adaptados a las necesidades de cada cliente.

Es importante destacar que existen, o durante su evolución han existido, dos conceptos diferentes de ERP, que en muchas ocasiones han llevado a la confusión entre ellos.

Por una parte está el ERP que se denominó “tradicional”. Es el que surge tras el MRP II y recibe el nombre originalmente. Surgen de la evolución y sofisticación de los sistemas de gestión de inventarios durante los últimos cincuenta años.

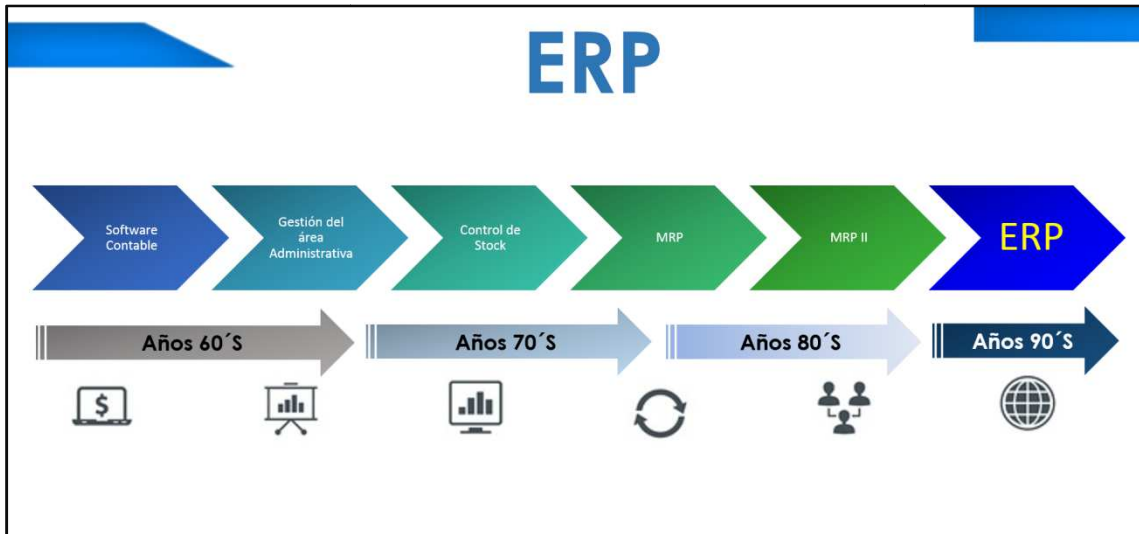
Es una aplicación para la gestión integrada de los diferentes módulos básicos que forman una empresa, es decir, desde los módulos más primarios como son contabilidad, compras, ventas y producción, hasta módulos secundarios como recursos humanos, gestión de costes, calidad, etc. Es una plataforma de comunicación entre todas las divisiones de la empresa que permite agilizar el trabajo. Por lo general, debe dar cobertura a todas las áreas de negocio **internas** de una empresa.

Por otra, está el ERP II, que es la evolución hacia las nuevas tendencias tecnológicas y de negocio, como pueden ser el e-business y los entornos colaborativos. Su principal característica es la diferencia de los procesos que contemplan y su naturaleza.

En el modelo del ERP II la empresa **interactúa** con su entorno (e-business) y se generan modelos de “integración virtual” de cadenas de valor de todos los “jugadores” del ecosistema empresarial.

Actualmente, las empresas no solo necesitan poder gestionar sus procesos internos, lo que se entiende como el “back-office”, sino que cada vez se requiere cubrir necesidades de conectividad con clientes, proveedores, empleados, etc.

En la figura 2 se muestra gráficamente la evolución de los sistemas de gestión empresarial que he mencionado.



**Figura 2. Evolución de los Sistemas Integrados de Gestión.**

Fuente: Sistemas Integrados de Gestión Empresarial. Evolución histórica y tendencias de futuro. Raúl Oltra Badenes. Edit. Universitat Politècnica de Valencia.

El mercado global de software ERP para las grandes empresas está dominado por tres compañías:

- Microsoft
- Oracle
- SAP

Debido a que el presente reporte se enfoca al ERP de SAP por ser el utilizado en el proyecto desarrollado, es en el que profundizaré, así como en su lenguaje de programación (ABAP).

## Panorama Global de la empresa SAP

SAP AG es una multinacional alemana que provee productos y servicios informáticos de gestión empresarial y tiene su sede en Walldorf, Baden-Württemberg, Alemania.

## Historia de la empresa SAP – Línea de tiempo

### 1972-1981: Los primeros años, SAP R/1

- En 1972 cinco antiguos empleados de IBM fundan una empresa llamada SAP. Su proyecto era desarrollar software de aplicación estándar para la gestión empresarial en tiempo real. La sede se encontraba en Weinheim, Alemania.
- En 1973 SAP completa el primer software para contabilidad financiera, el sistema RF, que sería la base para el **sistema SAP R/1**. Utiliza los servidores de IBM y el sistema operativo DOS.
- En 1976 se funda SAP GmbH (Gesellschaft mit beschränkter Haftung- Sociedad con Responsabilidad Limitada) como una empresa auxiliar de ventas y marketing.
- En 1977 la sede se traslada de Weinheim a Walldorf y por primera vez se instala SAP con clientes fuera de Alemania (Austria).
- En 1979 SAP utiliza su primer servidor propio, un siemens 7738 y se comienza el desarrollo de **SAP R/2**.
- En 1980 SAP aumenta su infraestructura de procesamiento de datos con el emparejamiento de su servidor existente con un IBM/370-148, que reemplazan en el mismo año por uno más potente, IBM 4341 que contaba con 4 MB de memoria.
- En 1981 la empresa privada fundada en 1972 (SAP) se disuelve y sus derechos se transfieren a SAP GmbH, con lo que la empresa adquiere el nombre actual. SAP R/2 alcanza un alto nivel de estabilidad.

### 1982-1991: La era de SAP R/2

- En 1984 SAP AG (Aktiengesellschaft – Sociedad Anónima) es fundada en Biel, Suiza.
- En 1986 SAP acude por primera vez a la feria de comercio de software CeBIT en Hannover, Alemania.
- En 1987 la nueva generación de servidores de IBM hace que el software de SAP esté ahora disponible para empresas de mediano tamaño. Se funda SAP Consulting dirigido a éste segmento de clientes. La estandarización de la producción de software da comienzo a la nueva generación: SAP R/3.
- Para 1988 SAP GmbH se convierte en SAP AG.
- En 1991 SAP presenta su sistema R/3 por primera vez en la CeBIT de Hannover. Tiene el concepto de cliente/servidor, aspecto uniforme de las interfaces gráficas, uso consistente de bases de datos relacionales y capacidad de ejecutarse en equipos de distintos proveedores.

### 1992-2001: La era de SAP R/3

- En 1992 **SAP R/3 se lanza al mercado general**, era compatible con una amplia gama de sistemas operativos, protocolos de comunicación y bases de datos. También la interfaz gráfica SAP GUI ayuda a expandir la base de sus clientes.
- En 1994 se funda SAP Latinoamérica y el Caribe (LAC).
- En 1996 se incorpora a **SAP R/3** la posibilidad de **trabajar en Internet**, empieza a elaborar soluciones específicas para cada sector comercial.
- En 1999 se anuncia la estrategia **mySAP.com**, herramienta de software de comercio electrónico con aplicaciones ERP existentes por medio de tecnología web la cual ofrece soluciones para el usuario en función de su rol dentro la empresa.
- En 2001 SAP adquiere Top Tier y forma SAP Portals, que representa la total expansión de la funcionalidad de SAP a internet.

### 2002-presente: Datos en Tiempo Real, donde y cuando los necesitan

- En 2003 mySAP evoluciona a SAP NetWeaver. Esta tecnología permite ofrecer aplicaciones de negocio rápidas, abiertas y flexibles que soportan los procesos de negocio "end-to-end", no importa si se basan en sistemas de SAP o de otros proveedores.
- En 2004 se aporta la **primera versión de SAP NetWeaver** al mercado. SAP planea su futuro en torno al concepto de la arquitectura orientada a servicios empresariales (Enterprise SOA-Service Oriented Architecture).
- En 2005 se centra en la adquisición de empresas más pequeñas cuyas soluciones específicas aumentan su cartera, entre ellas, los dos proveedores minoristas Triversity y Khimetrics.
- En 2006 SAP y Microsoft anuncian una alianza para integrar las aplicaciones ERP de SAP con las de Office de Microsoft bajo el nombre "Duet". En el primer evento SAPPHIRE del año en Orlando, Florida, SAP anuncia el lanzamiento general de su **aplicación insignia, SAP ERP**. Amplía notablemente su participación en el mercado con las soluciones SAP All-in-One y SAP Business One.
- En 2007 SAP compra Pilot Software, Yusa, OutlookSoft, Wicom y MaXware. En Nueva York revela SAP Business ByDesign, un producto diseñado específicamente para las pequeñas y medianas empresas.
- En 2008 SAP completa la adquisición de Business Objects, una empresa francesa especializada en aplicaciones de inteligencia de negocios, con lo que se convierte en el líder del mercado en software e inteligencia de negocios y gestión del rendimiento empresarial.
- En 2009 en Nueva York, SAP presenta su software *SAP Business Suite 7.0*, que está diseñado para ayudar a las empresas a optimizar su rendimiento y reducir los costes de TI.
- En mayo del 2010 SAP anuncia la compra de la compañía de California Sybase, el más grande software empresarial y el proveedor de servicios especializado exclusivamente en la gestión de información y el uso de datos móviles. El propósito es producir soluciones "inalámbricas". También existen numerosas innovaciones, la principal es la tecnología

"in-memory", que marca el comienzo de la era de procesamiento en tiempo real en las aplicaciones de negocio.

- En 2011 los primeros clientes implementan el primer producto *en memoria*, la plataforma SAP HANA, que permite analizar los datos en segundos. Siguen creciendo las aplicaciones empresariales móviles y también tiene planes de crecimiento en el mercado "cloud-computing" que está en auge. A finales del año anuncia su adquisición de SuccessFactors, el proveedor líder de aplicaciones en la nube.
- Hasta la fecha, SAP sigue apostando a aplicaciones en la nube, aplicaciones de movilidad y a procesamiento, análisis y reportes de grandes volúmenes de información en tiempo real (SAP HANA).

La figura 3 expone la cronología de la empresa SAP y algunos de los logros significativos que menciona:

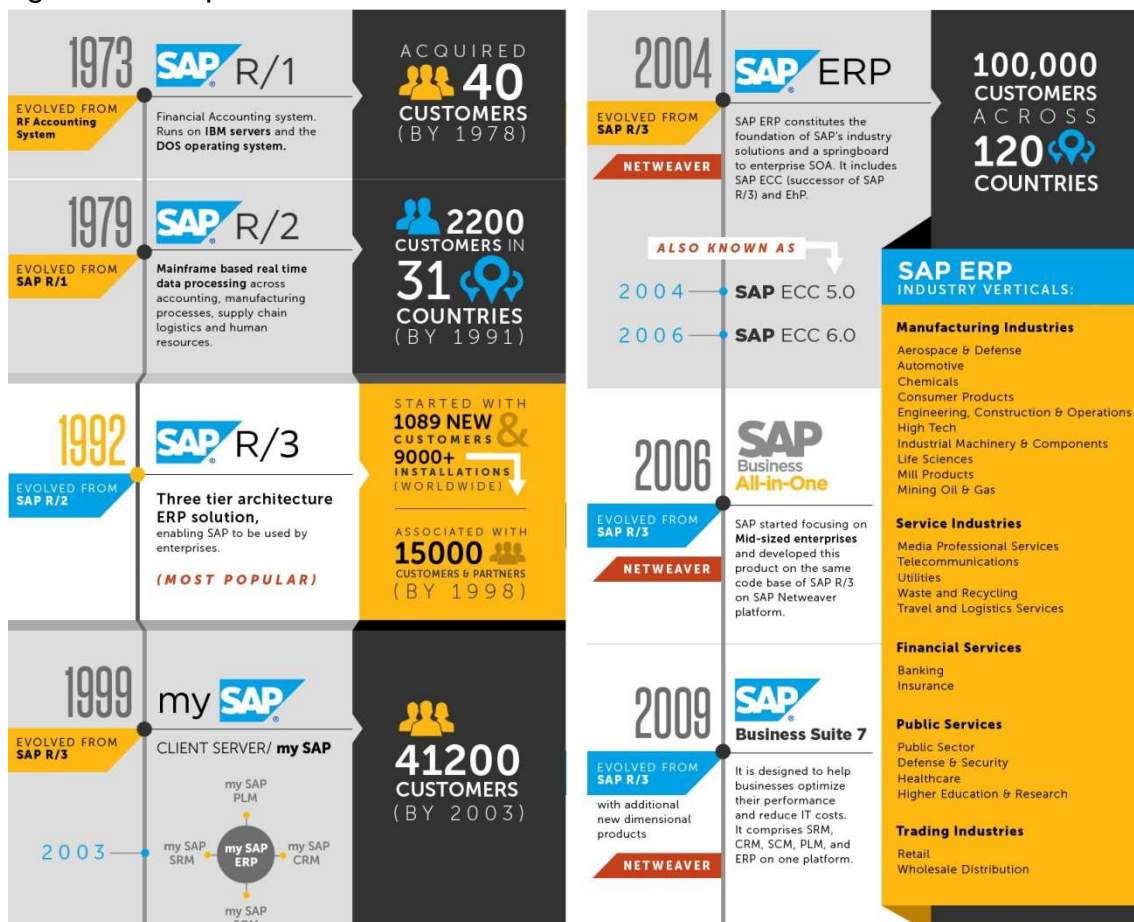


Figura 3. Evolución de la Empresa SAP.

Fuente: InSync Techfin Solutions, Enero 2014. SAP ERP Solutions - How It Evolved Over Time.

## Definición y Descripción del Sistema SAP

SAP es un sistema para gestión de negocios que cuenta con diferentes módulos que encapsulan los procesos de negocios de una organización. Se basa en combinar todas las actividades de negocio en una solución informática simple, integrada, robusta y fiable.

Al ser un sistema modular, combina muchísimas áreas de la organización entre sí formando un todo integrado que posibilita la comunicación e interacción de los datos, procesando grandes cantidades de ellos y obteniendo información útil para la toma de decisiones.

Uno de los principales productos es SAP ERP, que es una de las cinco aplicaciones empresariales en Business Suite <sup>1</sup> y brinda soporte a las funciones esenciales de los procesos y operaciones de una empresa.

Otras ofertas de productos SAP incluyen: Gobierno, Riesgo y Cumplimiento (GRC), Duet (oferta conjunta con Microsoft), soluciones de gestión del rendimiento, etc. También ofrece capacidades de arquitectura orientada a servicios (Enterprise SOA) en forma de servicios web que se envuelven alrededor de sus aplicaciones.

Durante mucho tiempo, se accedía al cliente exclusivamente a través de la interface gráfica SAP GUI. Alrededor de 1996 el uso de ITS (Internet Transaction Server) y el SAP GUI para HTML hizo posible tener acceso a la funcionalidad vía internet.

Entre los años 1999 y 2002, dentro del alcance de la iniciativa mySAP.com, la funcionalidad de internet tuvo un gran avance con aspectos como el soporte XML y el SAP Web Application Server (después fue renombrado como SAP NetWeaver Application Server) así como la integración de la tecnología JAVA.

SAP NetWeaver Portal constituye la total integración con las tecnologías basadas en WEB. La combinación del lenguaje ABAP con programación JAVA contribuye a un acceso directo a la funcionalidad de SAP con aplicaciones para internet.

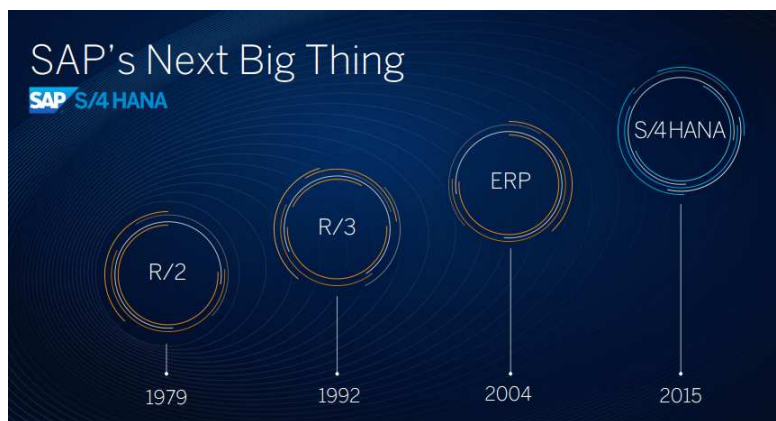
El nombre anterior de SAP ERP era R/3, el número 3 está relacionado con la arquitectura de programación por capas: base de datos, servidor de aplicaciones y cliente (SAP GUI). R/2, que se desarrolló en una arquitectura Mainframe (computadora central), fue el predecesor de R/3 y antes estuvo RF System, conocido como R/1.

En la figura 4 se observa cómo ha evolucionado del sistema de gestión empresarial que ofrece SAP hasta nuestros días.

---

<sup>1</sup> Ver Anexo A - SAP Business Suite.





**Figura 4. Evolución de Sistemas SAP.**

Fuente: Ser Consultor SAP. Publicación de Febrero 2015.  
"SAP S/4 HANA", el nuevo producto de SAP

A partir de 1992, se fueron lanzando distintas versiones de SAP R/3, desde la 1.0 hasta la 6.0 (SAP ERP 6.0). La última versión de SAP Business Suite es la 7.0 lanzada en 2009.

Versión	Fecha/Lanzamiento
SAP R/1, System RF	1972
SAP R/2, ran on a Mainframe architecture	1979
SAP R/3 Enterprise Edition 1.0 A	Jul 1992
SAP R/3 Enterprise Edition 2.0	1993
SAP R/3 Enterprise Edition 3.0	1995
SAP R/3 Enterprise Edition 4.0 B	Jun 1998
SAP R/3 Enterprise Edition 4.3	
SAP R/3 Enterprise Edition 4.5 B	Mar 1999
SAP R/3 Enterprise Edition 4.6 C	Abr 2001
SAP R/3 Enterprise Edition 4.6 F	
SAP R/3 Enterprise Release 4.70	Marz - Dic 2003
SAP R/3 Enterprise Edition 4.7	
SAP R/3 Enterprise Central Component (ECC) 5.0	2004
SAP R/3 Enterprise Central Component (ECC) 6.0	Oct 2005- Jun2006
SAP <b>ERP</b> 6.0 - Enhancement Packages (1,2,3,4,5,6,7)	

**Tabla 2. Versiones de SAP R/3.**

SAP presentó su paquete de mejoras (EHP) para SAP ERP como medio para que los clientes pudieran actualizar o instalar nuevas funcionalidades sin tener que migrar de sistema.

Son una forma efectiva, segura y económica de incorporar mejoras sin necesidad de instalar un nuevo release, preservando el core del producto y modularizando las actualizaciones. Los clientes pueden implementar selectivamente las innovaciones de software de SAP y activar el software a la demanda empresarial.



Como resultado, los clientes pueden aislar el impacto de las actualizaciones de software y poner en línea la nueva funcionalidad más rápido. Esto reduce el conflicto que se pueda crear entre innovación y estabilidad.

SAP ERP 6.0			
Basado en	Ultimo Support Package (SP) / Liberado en:	Enhancements Packages correspondientes (EhP)	Disponibilidad (Liberado al cliente)
SAP Netweaver 7.0	SP22 / Oct-12		Octubre 2005
	SP11 / Sept-10	SAP EhP 1 para SAP ERP 6.0	Diciembre 2006
	SP12 /Oct-12	SAP EhP 2 para SAP ERP 6.0	Julio 2007
	SP11 / Oct-12	SAP EhP 3 para SAP ERP 6.0	Diciembre 2007
	SP12 / Nov-12	SAP EhP 4 para SAP ERP 6.0	Noviembre 2008
SAP Netweaver 7.01	SP12 / Nov-12	SAP EhP 4 para SAP ERP 6.0 /NW 7.01	Mayo 2009
SAP Netweaver 7.0	SP09 / Nov-12	SAP EhP 5 para SAP ERP 6.0	Diciembre 2010
Enhancement Package 3 for SAP NetWeaver7.0 (SAP NW 7.03)	SP05 / Nov-12	SAP EhP 6 para SAP ERP 6.0	Noviembre 2011
	SP03 / Jul-13	SAP EhP 6 para SAP ERP 6.0, version para SAP HANA	Diciembre 2012
		SAP EhP 7 para SAP ERP 6.0	Agosto 2013

**Tabla 3. Versiones de SAP ERP 6.0.**

En la empresa donde laboro se cuenta con un sistema SAP ERP 6.0 EhP 5, como se muestra en la figura 5. Este paquete incluye mejoras en el SAP Business Objects Portfolio, herramientas de internet, workflow y en cada módulo del ERP, por ejemplo, se habilita la interfaz SAP NetWeaver Business Client 3.0 y el acceso al ERP desde Safari de Apple.

Detalles	
Denominación grupo	Conten.celda
Object ID	01200615320900003612
Produkt	EHP5 FOR SAP ERP 6.0
Release	EHP5 FOR ERP 6.0
Hersteller	sap.com
Kurzbeschreibung des Produktes	EHP5 FOR SAP ERP 6.0

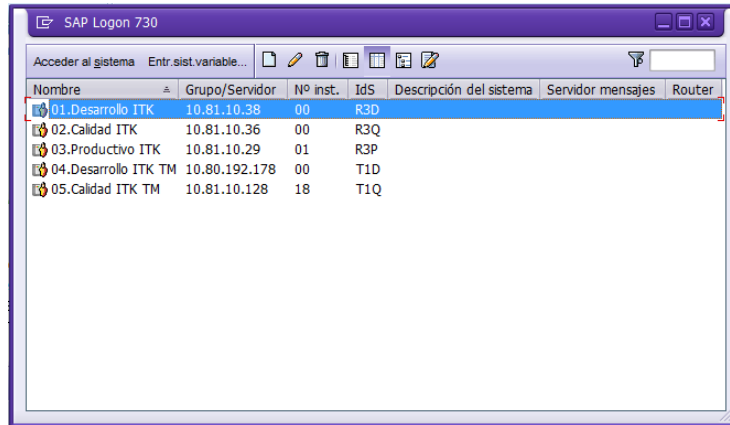
**Figura 5. Versión de SAP ERP en la empresa.**

Fuente: Impresión directa del sistema.

**Nota:** Para profundizar en el Sistema SAP ir al Anexo A.

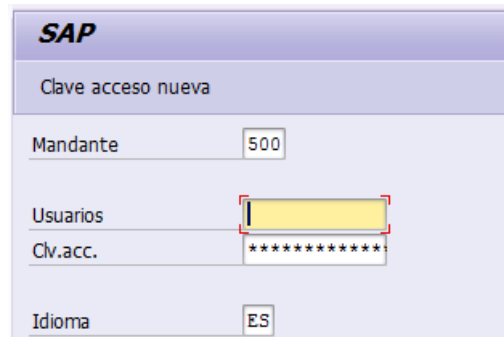
## Ingreso al Sistema SAP

Debemos considerar que el sistema SAP tiene por un lado, el servidor donde está la base de datos (el motor de SAP) y por otro el cliente SAP (SAP GUI) que está en cada computadora en la red de una empresa e incluye el programa SAP Logon para inicio de sesión al sistema SAP <sup>2</sup>. En la figura 6 muestro como se ven los accesos a los distintos ambientes de trabajo en una empresa al iniciar el SAP Logon.



**Figura 6. Ejemplo de SAP Logon.**  
Fuente: Impresión directa del sistema.

Al dar doble click en el ambiente al que se requiera ingresar, se solicita un usuario y contraseña (el cual tiene establecidos los privilegios permitidos) que asigna el administrador del sistema SAP (basis) y con el cual se mantiene el control de cualquier actividad realizada. En la figura 7 se observa la pantalla que solicita los datos indicados para poder ingresar al ambiente de desarrollo en la empresa.



The screenshot shows the SAP login screen for a new access key. The title is 'SAP' and the subtitle is 'Clave acceso nueva'. The form contains the following fields:

- Mandante: 500
- Usuarios: [Empty field]
- Civ.acc.: [Masked field with asterisks]
- Idioma: ES

**Figura 7. Inicio de Sesión de SAP.**  
Fuente: Impresión directa del sistema.

<sup>2</sup> Ver Anexo A – SAP GUI.

## Definición del Lenguaje de Programación ABAP

ABAP (Advanced Business Application Programming) es un lenguaje de programación, propiedad de SAP, para el desarrollo de sus aplicaciones de negocios. Como lenguaje de SAP, ABAP se utiliza como elemento de programación específico para la mayoría de sus productos y los programas ABAP son ejecutables únicamente dentro del ambiente SAP.

Originalmente fue definido como un lenguaje estructurado de cuarta generación y posteriormente ha incorporado elementos propios de lenguajes orientados a objetos. Se podría decir que ABAP se comporta como lenguaje de cuarta generación y como lenguaje orientado a objetos a la vez dentro de SAP.

## Modelo de programación

ABAP es un lenguaje estructurado orientado a eventos, es decir, la secuencia de las instrucciones depende del cumplimiento de una condición o evento. Se basa en subrutinas y módulos de función, y desde el release 6.10, el modelo de programación se basa en clases e interfaces. Ambos modelos son interoperables. En la figura 8 ejemplifico lo mencionado donde un programa abap realiza ciertas instrucciones en base al flujo y condiciones establecidos.

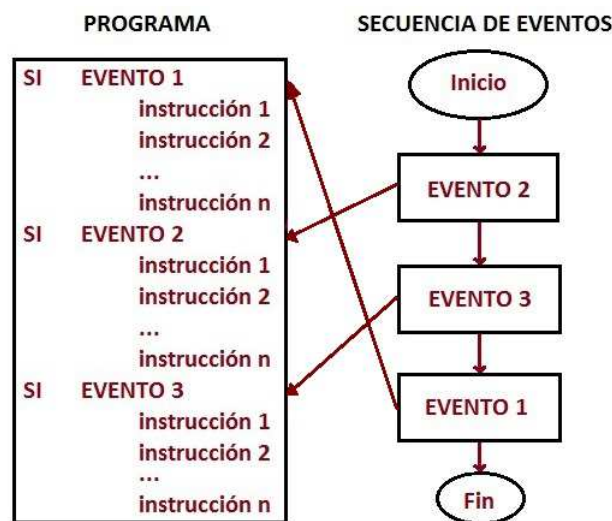


Figura 8. Lenguaje Orientado a Eventos.

Fuente: Elaboración Propia.

## El entorno de desarrollo ABAP

Los usuarios interactúan con SAP a través de transacciones, que representan procesos y funcionalidades propias del negocio de las empresas. Se pueden ejecutar desde menús definidos en el sistema, pero regularmente se ingresa el código de la transacción directamente en el campo de comandos. Dentro de SAP, el lenguaje ABAP posee su propio entorno de desarrollo denominado **ABAP Workbench**, que es un conjunto de transacciones necesarias para poder programar ABAP, todas ellas agrupadas bajo un área común.

**Nota:** Para profundizar en el lenguaje de programación ABAP ir al Anexo B.

## Capítulo 2. Descripción de Actividades Profesionales

### Experiencia profesional.

Comienza en el año 2007 al realizar el Servicio Social en la Unidad de Servicios de Cómputo Académico (UNICA) de la Facultad de Ingeniería, donde me encargaba de tareas como configuración de equipos de cómputo, mantenimiento preventivo y correctivo de los mismos, además de revisión de inventarios, apoyo a los usuarios en el préstamo de equipos y asesoría sobre su funcionamiento. Posteriormente estuve también un año como becaria.

En 2009 estuve por dos años en una consultoría del ramo energético, en el área de soporte técnico, donde brindaba asesoría técnica e intervenía en la resolución de problemas inherentes al funcionamiento del hardware y software. Asistía en la instalación y configuración de paquetería, así como en el mantenimiento de equipo de cómputo y redes.

Tuve la oportunidad de ingresar a una consultoría que brindaba soluciones en el área de SAP en 2011, donde conocí su lenguaje de programación y comencé mis actividades como consultor ABAP. Estuve cerca de un año desarrollando aplicaciones para diversas empresas en proyectos cortos de uno o dos meses. En éste tiempo aprendí las bases del lenguaje de programación ABAP y mis tareas principales fueron modificaciones a desarrollos Z (todos los nuevos objetos creados y cuyos nombres siempre deben iniciar con Z, pues el resto de letras están reservadas para los programas propios de SAP)<sup>3</sup>, smartforms (formularios para impresión) y reportes ALV (Abap List Viewer).

En marzo del 2012 trabajé en una editorial donde, por 8 meses, me desempeñé como analista y desarrollador ABAP. Algunas de mis actividades fueron el análisis de la operación de su almacén, para después desarrollar programas de diálogo que sirvieran para el control de indicadores de servicio del mismo, así como creación de funciones, implementación de badis, user exit y enhancement<sup>4</sup>, modificación de sapsript (formularios para impresión antiguos), creación y modificación de smartforms y reportes ALV, modificación de programas Z de interfaz entre SAP y SQL, principalmente.

Un programa de diálogo no es más que una pantalla (dynpro) relacionada a un programa ABAP subyacente que actúa según las elecciones del usuario sobre dicha pantalla.

En noviembre del mismo año ingresé a la empresa donde actualmente laboro en el área de sistemas SAP como desarrollador ABAP, donde he participado en diversos proyectos nuevos y mejoras a los ya existentes, como creación y modificación de reportes, funciones, RFC's (Remote Function Call), smartforms, consumos de web service para interactuar programas hechos en otro lenguaje de programación, implementación de badis, uso de bapis, etc.

---

<sup>3</sup> Ver Anexo B - Entorno de desarrollo ABAP.

<sup>4</sup> Ver Anexo B - Herramientas de ampliación de SAP.

El proyecto a exponer en el presente trabajo es la creación y modificación de programas de diálogo para el proceso completo de ventas de refacciones en Agencias Propias.

En 2013 curso la academia de ABAP obteniendo la certificación ABAP con SAP Netweaver 7.02 con ID 0010920699.

Como desarrollador ABAP he trabajado principalmente en los módulos de Ventas y Distribución (SD - Sales and Distribution), Gestión de Materiales (MM - Material Management), Gestión Financiera (FI - Financial Accounting), Costos (CO - Controlling), y ocasionalmente en Gestión de Calidad (QM - Quality Management), Planificación de Producción (PP - Production Planning) y Gestión del Transporte (TM - Transportation Management).

### **Actividades como Desarrollador ABAP**

Mis actividades se centran en crear soluciones en diversos módulos de SAP dependiendo de los requerimientos de la empresa. Principalmente han sido en los módulos de SD, FI, CO y MM.

Se comienza con el planteamiento de una problemática o de algún área de oportunidad por parte de los funcionales de los distintos módulos de SAP, que a su vez las ubican en la operación diaria del negocio o por el reporte o petición de los usuarios finales.

Como analista ABAP reviso con los funcionales el requerimiento para encontrar la mejor solución. Creo o modifico programas de tipo reporte o modulpool (programación de diálogo), pero también realizo “ampliaciones” al código estándar por medio de las herramientas estándar que provee SAP.

He trabajado con reportes tipo ALV's, smartforms, dynpros (pantallas) para programas de diálogo y funciones. Así mismo he creado y modificado diversos objetos en el diccionario de datos (dominios, elementos de datos, estructuras, vistas, tablas, etc.) y también he implementado batch input, call transactions y bapis, configurado web services, RFC's, etc.

### **Participación en proyectos de la empresa donde laboro.**

Los proyectos más relevantes que he tenido durante mi estancia en la empresa han sido:

- Agencias Propias: Se desarrolló un programa tipo modulpool con todas las opciones necesarias para que las agencias propias, puntos de venta de la empresa, realicen las actividades inherentes al área de refacciones sin que tengan que acceder directamente al sistema SAP, a cada transacción estándar requerida. El proyecto se hizo entre los tres desarrolladores ABAP que laboramos en la empresa e intervine en el flujo de ventas.

- Impresión de Etiquetas: Creé un programa que permite visualizar un reporte con los pedidos de autoservicios a nivel de detalle con la opción de ingresar el número de etiquetas que se requiera imprimir. Las etiquetas tienen datos generales del pedido en texto y en códigos de barra para que sea posible escanearlas fácilmente con la hand held.
- Porcentaje de Productividad: Anteriormente el proceso para el cálculo del porcentaje de productividad de Mano de Obra Directa (MOD) y Mano de Obra Indirecta (MOI) se hacía de forma manual mediante la obtención de información en distintas transacciones estándar en SAP y su posterior procesamiento en excel. Desarrollé un programa de tipo reporte que obtiene la información de tablas de SAP y la procesa para desplegar el reporte requerido y el resumen del proceso, lo cual redujo el tiempo de trabajo y errores humanos.
- Empaque y Volumetría: Para tener un correcto manejo de los materiales de empaque y volumetría dentro del módulo de TM desarrollé un programa de tipo modulpool que sugiere un material de empaque para las motocicletas que se van a embarcar de acuerdo al tipo de entrega que se va a trasladar del sistema ECC al de TM y bajo ciertas reglas establecidas por el negocio, el programa permite que el material de empaque sugerido pueda modificarse y/o agregar otros tipos de empaque cuando es necesario.
- Contabilidad Electrónica – Ingresos: Para poder recopilar la información de contabilidad de ingresos, y enviarla al SAT (Servicio de Administración Tributaria) hice una serie de programas que obtienen la información inicial de SAP, después se completa con información de otro sistema, con el que me comunico mediante un RFC (Remote Function Call) y posteriormente se envía mediante un web service.

Además de los proyectos mencionados, he trabajado en creaciones y modificaciones para diversas necesidades que han surgido en la empresa, que varían en su tiempo de desarrollo desde unas horas, hasta días o semanas.

Trabajo conjuntamente con los funcionales de los distintos módulos para comprender los requerimientos de los usuarios finales y así encontrar la mejor opción para solucionarlos.

## Capítulo 3. Descripción de Proyecto: Agencias Propias

### Problemática Inicial

En la empresa existen diversas áreas de negocio, una de ellas es el área de distribuidores que a su vez se dividen en dos: externos e internos. A los internos se les denomina como Agencias Propias, quienes pueden vender motocicletas, refacciones y accesorios. Son parte de la empresa pero tienen sus propios clientes finales.

Para poder realizar las ventas de refacciones en las agencias propias, el área de sistemas había desarrollado una solución externa a SAP, mediante la cual se realizaba la venta y posteriormente, por medio de RFC's, se actualizaba la información en el sistema SAP. Dicha solución tenía intermitencias de comunicación y por ello inconsistencias en los datos, por lo que la información no era consistente ni confiable.

Se pensó entonces en cambiar el sistema utilizado y manejar únicamente SAP, sin tener sistemas externos, pero no se quería que los usuarios finales tuvieran acceso al ambiente completo, sino únicamente a las tareas específicas que se realizan en las agencias, la parte de creación de ventas, compras y facturación.

SAP cuenta con soluciones propias pero tienen un costo elevado y se tenía que adquirir una licencia por cada agencia para que pudieran acceder al sistema, haciendo inviables los costos.

### Solución Requerida

Por lo planteado anteriormente se necesitó un programa que en cada agencia propia ingresara a los procesos necesarios para la venta y compra de refacciones mediante una pantalla de acceso administrado desde SAP, pero accediendo con una sola licencia para todos los usuarios en dicha agencia.

También se necesitaba agilizar los procesos y tener permisos de acuerdo a las actividades de cada usuario dentro de la agencia, que pueden ser 3 o 4 en cada una, pues no todos pueden realizar las mismas tareas. Actualmente se cuentan con 26 agencias propias en México, aunque se piensa expandir tanto en el país como en Latinoamérica.

### Objetivos Esperados

Se esperaba terminar con la inconsistencia en los datos de SAP y así tener un mejor control de inventarios y procesos de ventas y compras, reduciendo la venta muerta y el holdeo de material en las agencias propias.

Ventas muertas son las posibles ventas que no se realizaron por no tener material disponible. El holdeo se refiere a "apartar" material al generar



documentos de venta (cotizaciones) pero al no concluirse la venta y no cancelar dicho documento, el material no puede “liberarse” ni ser vendido.

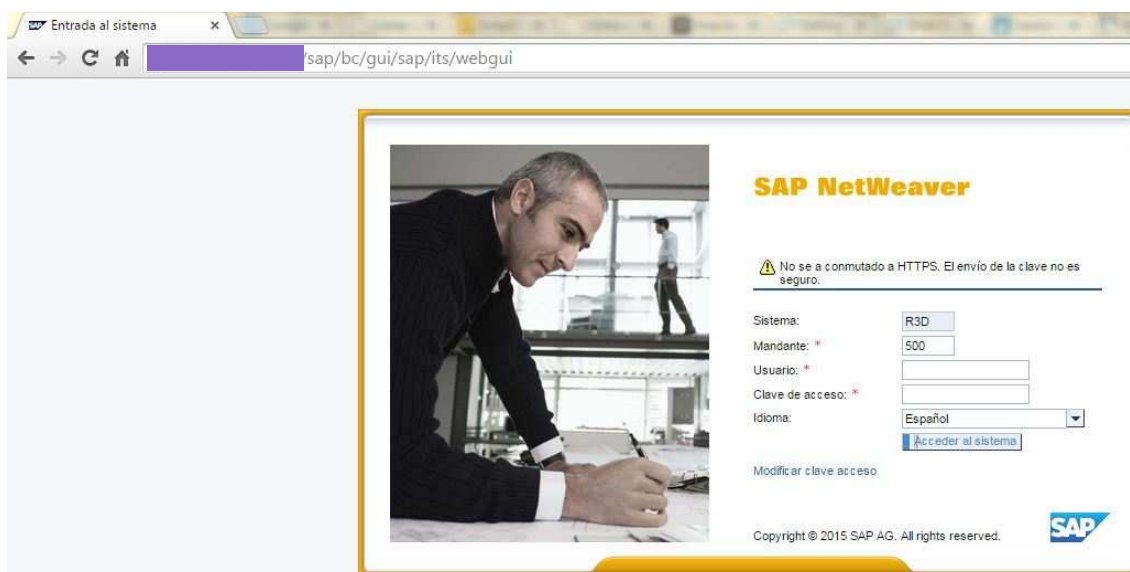
También se deseaba reducir tiempo de procesos al eliminar la existencia de un proceso intermedio y brindar un proceso sencillo a los usuarios finales.

## Descripción del Proceso

SAP cuenta con una herramienta para acceder al sistema desde un navegador web, sin necesidad de tener instalado el cliente SAPGUI, que se denomina **WEBGUI**, el cual se puede utilizar mediante la activación de ciertos servicios <sup>5</sup>.

Por medio de una url se ingresa a SAP desde internet, donde inician sesión y se direcciona automáticamente a la transacción asignada al programa creado, pues esto se establece en los permisos del usuario. Con esto no es necesario tener una conexión a intranet o VPN y cada agencia propia ingresa al programa en SAP.

En la figura 9 muestro la pantalla como se visualiza al ingresar a la url asignada en la empresa, donde solicita usuario y contraseña para iniciar sesión:



**Figura 9. WEBGUI para el ambiente de Desarrollo.**

Fuente: Impresión directa del sistema.

El programa creado es de tipo modulpool o programación de diálogo. Inicialmente el programa abap solicita un usuario y un password asignados de acuerdo al perfil de cada uno. Dichos usuarios son creados exclusivamente para ésta aplicación y se guardan en una tabla Z intrínseca al desarrollo. Es independiente al usuario con el que inician sesión en SAP.

<sup>5</sup> Ver Anexo A – SAP GUI para HTML.



En la figura 10 se observa la pantalla inicial al ingresar a la transacción asignada al programa creado, donde solicita usuario y password del usuario:

**Figura 10. Inicio de Sesión en Sistema de Agencias Propias.**

Fuente: Impresión directa del sistema.

Se establecieron ciertos perfiles de acuerdo a las tareas que cada usuario realiza.

**Perfiles establecidos:**

- **ADMIN** Administración de Usuarios y Menú Inicial
- **ALMACEN** Parte de Almacén de Menú Inicial
- **VENTAS** Parte de Ventas de Menú Inicial
- **AUTORCC** Administración de Usuarios, Menú Inicial y Autorizaciones de Descuentos

Al ingresar los datos para inicio de sesión, dependiendo de los privilegios del usuario, se desplegará una pantalla con las opciones atribuidas de acuerdo a su perfil. En la figura 11 se pueden ver las opciones disponibles para un usuario con perfil de administrador y en la figura 12 para un usuario con perfil de autorizador.

**Figura 11. Perfil ADMIN.**

Fuente: Impresión directa del sistema.

**Figura 12. Perfil AUTORCC.**

Fuente: Impresión directa del sistema.

- En la parte de Administración de usuarios se puede dar de alta, modificar o bloquear un usuario para el programa.
- La parte de Autorizaciones sirve para aceptar o denegar las peticiones de descuentos significativos.
- Las actividades que se pueden realizar en el área de refacciones de agencias propias se encuentran en la pantalla de menú inicial que está dividida en dos secciones, una para la parte de ventas y otra para la de almacén.
- El Menú Inicial nos muestra ambas secciones habilitadas para los perfiles de administrador y autorizador, pero para almacenista o cambaceador (ventas) sólo se habilitan las que les corresponde.

En las figuras 13 y 14 se observa el menú principal de las tareas de compra y venta, pero dependiendo de los permisos asignados al usuario, sólo se habilitan las tareas que le son permitidas.



**Figura 13. Perfil ALMACEN.**  
Fuente: Impresión directa del sistema.



**Figura 14. Perfil VENTAS.**  
Fuente: Impresión directa del sistema.

El desarrollo de la aplicación se dividió en tres partes, pues los tres desarrolladores ABAP de la empresa participamos en su realización:

- Administración de usuarios y facturación
- Actividades para el área de almacén
- Actividades para el área de ventas

**Mi participación fue principalmente en la parte de ventas, para generar el flujo completo y los reportes que utilizan.**

La parte de ventas tiene las siguientes opciones:

- a. Generar Venta
- b. Surtir Pedido
- c. Cancelar Pedido
- d. Crear Factura
- e. Crear Factura Electrónica
- f. Devoluciones
- g. Reportes de Ventas

## Flujo del Proceso

Las operaciones comerciales relacionadas con las ventas se registran en el sistema como “documentos de ventas”.

El flujo de ventas en la empresa es el siguiente:

- 1) Se genera cotización
- 2) Se genera pedido referenciado a la cotización
- 3) Se genera entrega para el pedido anterior
- 4) Se contabiliza dicha entrega
- 5) Se genera la factura

Describiendo brevemente cada opción del menú del programa:

- a. Generar Venta:** Direcciona a la pantalla mostrada en la figura 15, donde se ingresan las posiciones de un pedido (material, cantidad y descuento, si es que aplica):

No Material	Cantidad	% Descuento	UM	Descripción	Cant.Disponible [UN]	Precio Unitario	\$ Descuento	SubTotal	Total Posición	Moneda
<input checked="" type="checkbox"/>	<input type="checkbox"/>									
<input checked="" type="checkbox"/>	<input type="checkbox"/>									
<input checked="" type="checkbox"/>	<input type="checkbox"/>									
<input checked="" type="checkbox"/>	<input type="checkbox"/>									
<input checked="" type="checkbox"/>	<input type="checkbox"/>									
<input checked="" type="checkbox"/>	<input type="checkbox"/>									
<input checked="" type="checkbox"/>	<input type="checkbox"/>									
<input checked="" type="checkbox"/>	<input type="checkbox"/>									
<input checked="" type="checkbox"/>	<input type="checkbox"/>									

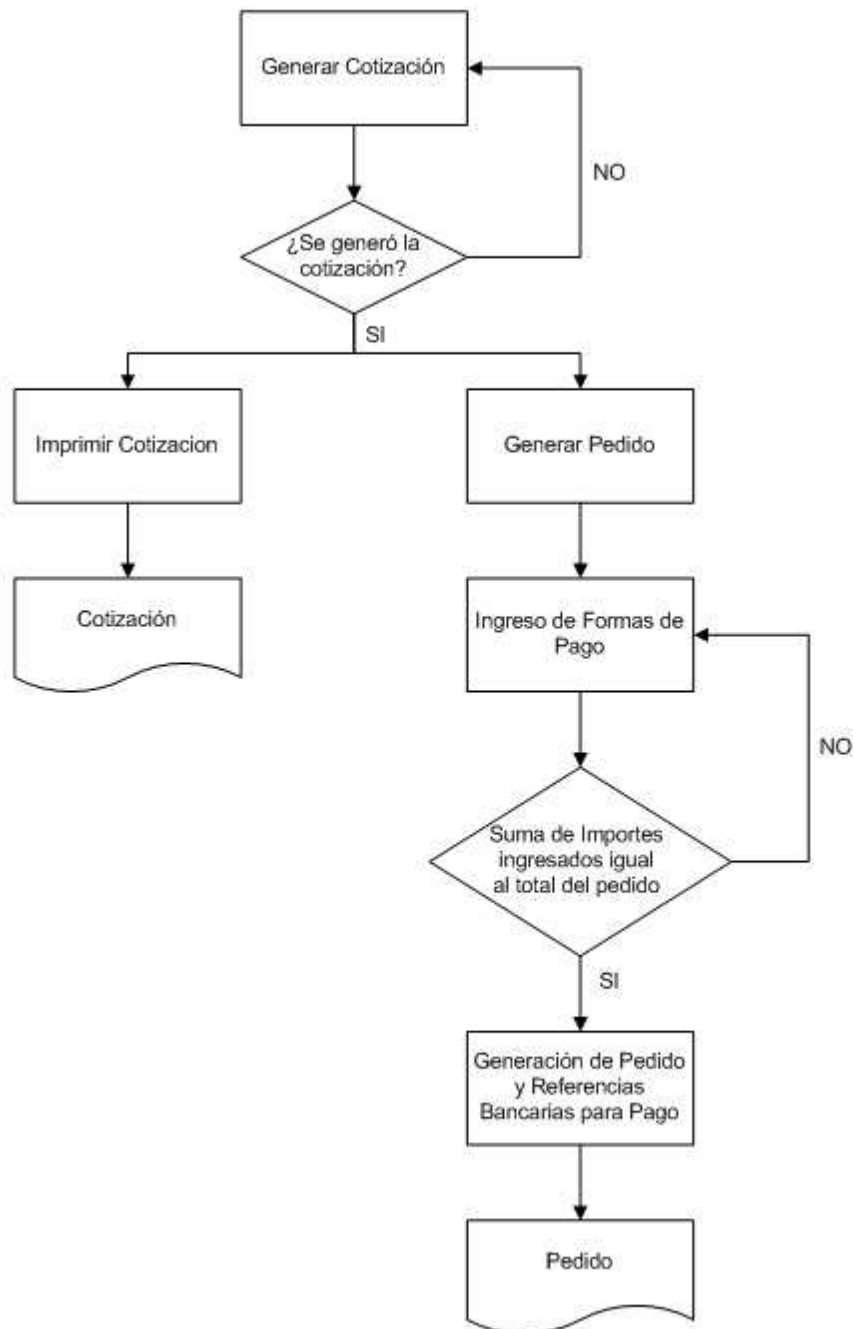
**Figura 15. Generación de Venta.**  
Fuente: Impresión directa del sistema.

Se cuenta con la opción de ingresar descuentos por posición (material):

- En venta al público en general se puede usar del 1 al 10 % de descuento.
- Para descuentos a CESIT (Centros de Servicio propios de la empresa) pueden tener de 10 a 40% de descuento, que requieren autorización previa.

Dichos descuentos se ingresan en una tabla Z para obtener los valores consultándola cada vez que se ingrese a la aplicación y que se puedan actualizar fácilmente cada vez que el negocio lo requiera.

En la figura 16 vemos los procesos que pueden ejecutarse en ésta opción:

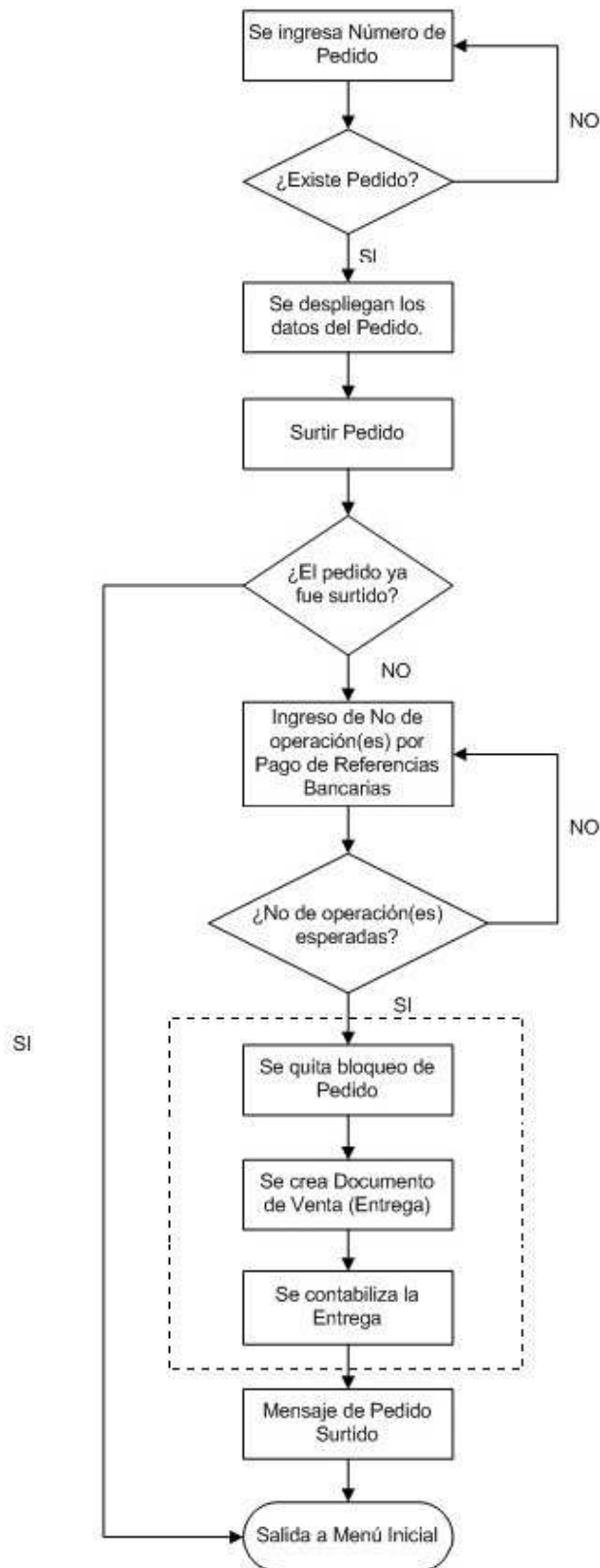


**Figura 16. Procesos en la opción “Generar Venta”.**

Fuente: Elaboración Propia.

Al generar el pedido, el programa consulta la disponibilidad de inventario de los materiales y en caso de que no haya la cantidad solicitada o sea insuficiente se manda un mensaje y termina el proceso. Si no hay problema de inventario se abre una pantalla pop-up que solicita ingresar el monto a pagar con varias opciones de pago: efectivo en banco propio de la empresa, efectivo en otro banco, cheque o tarjeta de crédito (pueden ingresar hasta 3). Se puede ingresar el monto total en una sola opción o distribuirlo, pero internamente se valida que se cumpla con el total a cobrar. El pedido se crea bloqueado (mediante BAPI).

b. **Surtir Pedido:** Direcciona a una pantalla donde se solicita el número de pedido y al presionar enter despliega su información, pero sin poder editarlo. Las opciones son: “Surtir Pedido” ó “Regresar”. En la figura 17 mostro el flujo que se realiza:



**Figura 17. Procesos en la opción “Surtir Pedido”.**  
Fuente: Elaboración Propia.

Al presionar el botón de surtir pedido se abre una pantalla pop-up que solicita el número de operación por cada forma de pago que se da al cliente al momento de pagar. Después de validarlo se modifica el pedido grabando el número de operación en la tabla Z donde se guardan los tipos de pago y se desbloquea el pedido.

Si no ocurren errores, se crea la entrega (mediante bapi) y se contabiliza (mediante batch input y call transaction a la Tx VL02N).

- c. **Cancelar Pedido:** Direcciona a una pantalla donde se solicita el número de pedido, al presionar enter despliega la información del mismo, pero sin poder editarlo. Las opciones disponibles son: “Cancelar Pedido” ó “Regresar”. La figura 18 señala los pasos que se realizan:

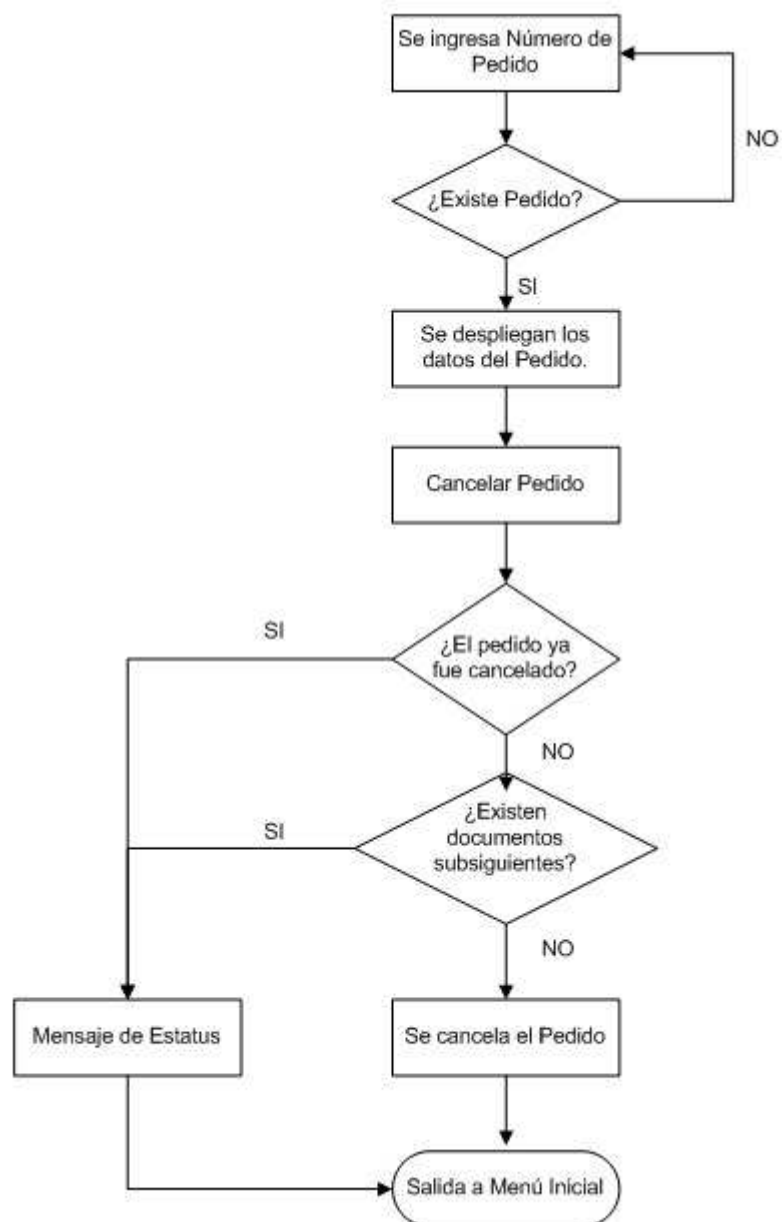


Figura 18. Procesos en la opción “Cancelar Pedido”.

Fuente: Elaboración Propia.

Al presionar el botón de cancelar pedido aparece un mensaje de advertencia preguntando si se quiere cancelar. Al aceptar se valida si el pedido no tiene documentos subsecuentes, es decir que no se le haya generado una entrega, o si no se ha cancelado anteriormente. Si no ocurren errores, el pedido se cancela (mediante bapi).

- d. **Crear Factura:** Direcciona a la pantalla mostrada en la figura 19, donde se debe ingresar el número de pedido. En la figura 20 se indican los pasos realizados por el programa.

**Figura 19. Pantalla de Facturación.**  
Fuente: Impresión directa del sistema.

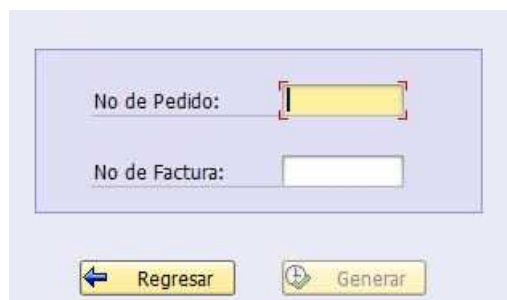


**Figura 20. Procesos en la opción “Crear Factura”.**  
Fuente: Elaboración Propia.



Al ingresar dicho pedido y presionar enter, busca el pedido en tablas estándar de SAP y si lo encuentra se habilitan las casillas para ingresar los datos del cliente (nombre, RFC, dirección, teléfono, etc.). Se crea la factura (mediante BAPI).

- e. **Crear Factura Electrónica:** Direcciona a la pantalla mostrada en la figura 21 donde se solicita el número de pedido y el número de factura. Al presionar enter se habilita la opción para generar la factura electrónica.



**Figura 21. Pantalla de Facturación Electrónica.**

Fuente: Impresión directa del sistema.

- f. **Devoluciones:** Direcciona a la pantalla mostrada en la figura 22 con dos opciones: devolución o venta con nota de crédito:



**Figura 22. Pantalla de Devoluciones.**

Fuente: Impresión directa del sistema.

En la opción de venta con nota de crédito direcciona a la pantalla mostrada (Fig. 15) para generar la venta, donde se tiene que ingresar el número de cotización y al oprimir enter despliega la información, para poder efectuar el proceso de ventas establecido, al generar el pedido para dicha cotización.

En la opción de devolución direcciona a una pantalla donde solicita el número de pedido. Al presionar enter se despliega el detalle del pedido y se deben seleccionar las posiciones e indicar la cantidad a devolver, como se muestra en la figura 23.

Pedido: 0220666916

Posición	No.Material	Descripción	Cantidad	D..	Cant A Devolver
10	XS125120119	XS125 NEGRO/AZUL	1.000	<input checked="" type="checkbox"/>	1
20	FT110110101	FT110 AZUL	1.000	<input type="checkbox"/>	
30	FT150130221	FT150GT grafito	1.000	<input type="checkbox"/>	

Regresar    Confirmar Devolución

**Figura 23. Pantalla de Proceso de Devolución.**

Fuente: Impresión directa del sistema.

**g. Reportes de Ventas:** Muestra los distintos reportes que se pueden generar, como podemos ver en la figura 24.

Reporte Ventas AGP  
 Solicitud de Descuentos  
 Lista de Pedidos  
 Reporte de Venta Muerta

Regresar    Aceptar

**Figura 24. Pantalla de Reportes de Ventas.**

Fuente: Impresión directa del sistema.

## Capítulo 4 Resultados

Como lo mencioné anteriormente, el programa se realizó entre los tres desarrolladores ABAP de la empresa y los funcionales de los módulos de SD, MM y FI, con quienes realizamos las pruebas necesarias de acuerdo a la configuración establecida para el negocio.

SAP en sí es un sistema robusto y consistente, que no permite “activar” un programa si tiene errores sintácticos y al ejecutarlo siempre usa la última versión activa. También cuenta con herramientas para evaluar el performance del código y sugiere como se puede mejorar cuando encuentra alguna advertencia.

### Pruebas

Primeramente se realizaron pruebas unitarias, donde validé la correcta funcionalidad de los módulos programados, probando los casos posibles y tratando de provocar los errores más comunes para asegurar que el manejo de los mismos fuera correcto.

Posteriormente se hicieron las pruebas de integración con los módulos que crearon los otros desarrolladores para comprobar que no existía interferencia entre los módulos y que el desarrollo era consistente.

Después se hicieron pruebas en conjunto con el funcional de ventas en el ambiente de desarrollo, corroborando en transacciones estándar que los documentos generados por el programa eran correctos y completos.

También se probó que las opciones habilitadas en las pantallas fueran las correctas de acuerdo a cada perfil de usuario y que se adaptaran cada que dicho usuario cambiara. Se probaron todas las validaciones en el programa verificando que enviara un mensaje de error significativo para el usuario cuando hubiera algún problema.

Posteriormente se pasó el programa al ambiente de calidad donde probamos los casos generales y en dicho ambiente fue donde se dio la capacitación a los usuarios finales, la cual fue a cargo de los funcionales de cada módulo.

**Nota:** Para profundizar en el tema de Pruebas de Software ir al Anexo C - Pruebas.

### Resumen de Resultados

Actualmente las ventas de refacciones en las agencias propias se realizan mediante éste programa y las inconsistencias que se presentaban de inventario se han resuelto.

Los usuarios tienen acceso a reportes que les simplifican las tareas diarias y les ayudan a tomar decisiones eficientes para la mejor administración de la agencia.

Se ha reducido el holdeo de inventario al existir procesos ejecutados por la noche diariamente, que eliminan las cotizaciones hechas no concluidas en venta, liberando así el material para cuando se requiera.

Se tiene un registro de venta muerta para mejorar el manejo de los materiales de mayor demanda y evitar pérdidas.

Pueden realizar pedidos de compra en base a las ventas realizadas y a los materiales que en requieren por lo que se tiene un mejor manejo de inventario.

El flujo de venta se realiza de manera sencilla y se tiene mejor control sobre los descuentos otorgados a cierto tipo de clientes, pues al ser mayores a cierto rango se debe tener autorización, la cual se solicita y se otorga mediante el mismo programa, enviando correos a las personas que tienen la facultad de hacerlo.

Desde su liberación en 2013 a la fecha se han realizado diversas modificaciones a las rutinas iniciales, así como eliminación o inserción de nuevas conforme las necesidades del negocio lo han requerido.

Se tiene pensado reestructurar todo el programa debido a dichos cambios para mejorar el rendimiento y poder quitar código que pueda ser redundante, afectando el performance y crear un sistema parecido para el manejo en Latinoamérica.

## Conclusiones

Durante la carrera en la facultad de ingeniería recibí educación en diversas áreas del conocimiento, desde las materias de ciencias básicas, las asignaturas formativas que corresponden a cada carrera, las asignaturas optativas e incluso materias humanísticas, es decir, una educación integral que nos permite acoplarnos más fácilmente a las diversas necesidades en la vida profesional.

Al inicio del proyecto de agencias propias mis conocimientos de la programación de diálogo eran básicos. El iniciar el desarrollo desde cero y el tener tantas necesidades para construir un sistema robusto que pudiera cubrir con todas las actividades de una agencia me permitió ampliar dichos conocimientos en el manejo del flujo de proceso de los module pool, que es muy diferente a la programación de reportes.

También me permitió conocer un poco más el aspecto funcional del área de ventas y de la configuración establecida en la empresa, más allá del aspecto técnico, ya que el flujo de venta de SAP es sugerido pero cada negocio emplea el que mejor se ajuste a sus necesidades.

Además de adquirir los conocimientos mencionados, me dio la satisfacción de trabajar en equipo con otros desarrolladores y, aunque hubo momentos complicados, ésta forma de trabajo brinda la ocasión de aprender de otras personas de la misma área y poder ayudar con mis aptitudes y habilidades, mejorando así las relaciones interpersonales.

Así mismo, puse en práctica las habilidades y conocimientos adquiridos durante la carrera en la Facultad de Ingeniería en el ambiente laboral, principalmente en ingeniería de software. Me permitió complementar y perfeccionar mis estudios académicos, aplicándolos en problemas reales, permitiendo tener una base más sólida y con ello un mejor desempeño en mis actividades. Me permitió la adquisición de nuevos conocimientos, ya que las exigencias propias de las actividades desarrolladas así lo exigieron.

Por último, partiendo de la base de que un ingeniero en computación debe ser capaz de analizar las diversas problemáticas que se presentan en su lugar de trabajo, diseñar y construir soluciones a dichas situaciones así como dar soporte a las mismas o mejorar las ya establecidas, considero que mi desempeño laboral ha contribuido a solucionar en muchas ocasiones los problemas presentados en la operación de SAP y del negocio en sí.

Varios de los reportes que he desarrollado han contribuido a disminuir considerablemente el tiempo invertido en obtener y manipular información sensible para poder tomar decisiones de negocio en base a ellas, como también han reducido los errores humanos que se presentaban. Programas creados para controlar movimientos de materiales entre almacenes han

disminuido los tiempos de surtimiento e influido en un mejor control de stock. Procesos que anteriormente se realizaban mediante soluciones externas a SAP se han reemplazado para ser cada vez más uniforme la información, pues el intercambio de datos con otros satélites contribuía a su inconsistencia.

El programa de agencias propias expuesto contribuyó a un fácil manejo de compras y ventas, un mejor control de inventarios de cada agencia, un fácil proceso de venta, traducándose a más pedidos realizados correctamente y a su vez más ingresos económicos a la empresa.

He podido analizar conjuntamente con los funcionales los diversos requerimientos que tienen los clientes finales y las problemáticas que se presentan, pudiendo discutir las opciones posibles, pues no siempre la mejor solución es un desarrollo ABAP, sino que se puede arreglar mediante configuración estándar de SAP o simplemente cambiando el procedimiento realizado.

## Glosario

**ABAP:** (Advanced Business Application Programming) es un lenguaje de programación propiedad de SAP para el desarrollo de sus aplicaciones de negocios.

**BADI:** Herramienta de ampliación en el código estándar de SAP de 3ra generación basada en programación ABAP orientada a objetos. Sirven para acomodar los requerimientos específicos de un cliente a las transacciones estándar de sap.

**BAPI:** (Business Application Programming Interface) son funciones que sirven para realizar acciones sobre los objetos de negocios de SAP. Son en realidad métodos de los objetos de negocios y gracias a ellos podemos realizar cargas evitando utilizar Batch Inputs por Call Transaction.

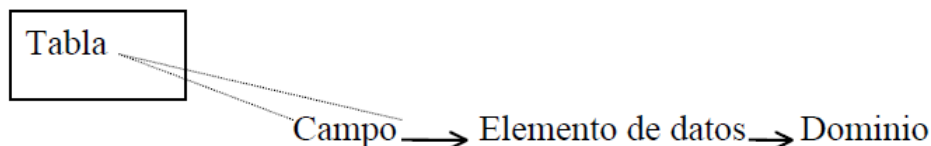
**Batch Input:** Es un método para de transferir grandes cantidades de datos a un sistema SAP, para hacer muchas altas, modificaciones o borrados. Se simula un proceso on-line (transacción donde interacciona el usuario), para someter a los datos a todos los chequeos y validaciones que sufrirían si se metieran manualmente, para salvaguardar la integridad de los mismos (cosa que no ocurriría con un MODIFY directo a una tabla del Diccionario de Datos, eso es lo importante de los batch-inputs). Pero en cambio no requieren interacción.

Hay 2 métodos de batch-input: "clásico" y "call transaction". En el método "clásico" se genera una "sesión batch-input". Se tiene un fichero con los datos y un programa ABAP de conversión que crea la sesión (datos, pantallas, transacciones, etc. es un juego de datos), que simulan la existencia de un usuario que introduciría los datos), que se almacena y se puede procesar. Este método es asíncrono: se procesan los datos ahora pero se actualizan más tarde. Permite múltiples transacciones. Se genera un log para cada sesión, pero no se pueden generar en paralelo desde el mismo programa (sólo puede abrirse un juego de datos cuando se cierra el anterior). En el método "call transaction" los datos se crean on-line al ejecutar el programa de conversión, en lugar de crear una sesión. Es mucho más rápido, pero poco útil para gran cantidad de datos (se perderían datos si hay errores, pues no se guardan en la sesión batch-input). Se usa para dar de alta rápidamente pocos datos. Es un método síncrono, válido para una transacción, rápido, pero no se genera log, ni pueden tratarse errores a posteriori.

**Customizing:** Cuando se implementa el sistema SAP se debe adaptar a las necesidades específicas de la empresa cliente. El proceso de parametrización del sistema, NO AL CODIGO FUENTE, se conoce como Customizing. Habitualmente son las opciones que se encuentran disponibles dentro de la transacción SPRO y por las cuales se cambia el comportamiento del sistema en su conjunto o de transacciones individuales para adecuar el sistema a la organización donde se lo implementa.

**Datos Maestros:** Información que representan en el sistema las entidades básicas de una empresa, por ejemplo cuentas de mayor, clientes, acreedores, proveedores, materiales, etc. Conjunto de datos almacenados sistemáticamente.

**Diccionario de Datos:** Es una fuente de información centralizada. Los distintos objetos del diccionario de datos están estructurados en:



**Dominio:** Describen el campo de valores posibles. Consiste la especificación técnica de un campo de la base de datos. Define lo que se conoce como estructura del campo. Se guarda en el diccionario y es reutilizable para otros campos.

**Dump (Error en tiempo de ejecución):** Es un log de terminación anormal de ejecución de cualquier programa. Se produce por una cancelación del programa que se está actualmente ejecutando y el sistema muestra una pantalla de log con la información acerca del error y su posible solución.

**Dynpro (DYNAMIC PROGRAM):** Es un programa dinámico que consiste en una pantalla y la lógica de proceso subyacente que la controla.

**E-Business:** Acrónimo del idioma inglés “electronic” y “business”, se refiere al conjunto de actividades y prácticas de gestión empresariales resultantes de la incorporación a los negocios de las tecnologías de la información y la comunicación (TIC) generales y particularmente de Internet, así como a la nueva configuración descentralizada de las organizaciones y su adaptación a las características de la nueva economía. El e-business, que surgió a mediados de la década de los años 1990, ha supuesto un notable cambio en el enfoque tradicional del capital y del trabajo, pilares fundamentales de la empresa, y en sus prácticas productivas y organizacionales. Las actividades que ponen en contacto clientes, proveedores y socios como la mercadotecnia y ventas, la producción y logística, gestión y finanzas tienen lugar en el e-business dentro de redes informáticas que permiten a su vez una descentralización en líneas de negocio.

**Elementos de dato:** Objeto ABAP que define el tipo de dato de un variable o campo y que puede utilizarse en un programa ABAP o en un campo de una tabla base de datos. También se puede definir a partir de un dominio.

**Enhancement:** Framework para el desarrollo de ampliaciones a partir de la versión 6.0 de SAP. Es una nueva tecnología que brinda diferentes técnicas de ampliación: Ampliaciones de código (plug-ins), Ampliaciones de grupo de función, Ampliaciones de clases y Ampliaciones Kernel-BADI. Dentro de este



Framework se diferencian las ampliaciones de tipo implícito, predefinidas y provistas por SAP en el código fuente, y las ampliaciones de tipo explícito que pueden implementarse en cualquier parte del código fuente a excepción de algunas áreas restringidas por SAP, a través de los ENHANCEMENT-POINTS y ENHANCEMENT-SECTION.

**ERP (Enterprise Resource Planning o sistema de planificación de recursos empresariales):** Integra todas las funciones e información de los diversos departamentos o áreas de un negocio o empresa en una única base de datos común. Los ERP se conforman por paquetes de software que permiten a las empresas evaluar, implementar, automatizar, integrar y gestionar de forma eficiente las diferentes operaciones que se presentan en éstas.

**Instancia:** Cada ambiente (desarrollo, testing, producción) es una instalación distinta de SAP, a cada instalación se las llama instancia.

**Mandante:** Es el concepto más importante dentro de SAP R/3, es una unidad dentro del sistema desde un punto de vista técnico, organizativo y comercial, un ambiente de trabajo dentro de un ambiente o entorno. Los mandantes tienen sus propios registros maestros y su propio conjunto de tablas. Es el nivel jerárquico más alto en el Sistema SAP. Se caracteriza por tener su propio entorno de datos empresariales, así como sus propios datos variables, maestros y de usuario. Desde el punto de vista lógico es la unidad organizativa divisoria de la empresa y permite que diferentes usuarios estén trabajando en el mismo sistema sin ningún tipo de interferencia y desde el punto de vista físico es el primer campo clave de la mayoría de las tablas que forman la base de datos relacional de SAP R/3.

**Metodología ASAP:** Es una metodología de implantación de proyectos denominada Accelerated SAP (ASAP) que fue desarrollada por la empresa SAP, y constituye uno de los puntos clave para maximizar los tiempos, la calidad y la eficiencia del proceso de implantación. Sus fases son: preparación del proyecto, plano empresarial (blueprint), realización, preparación final y entrada a productivo y soporte.

**Modo:** Cada una de las seis pantallas, como máximo, que puede abrir un usuario desde que abre una sesión con R/3.

**RFC (Remote Function Call):** Es un procedimiento para intercambiar datos entre un cliente y un servidor. Típicamente el cliente llama a un programa residente en el servidor y éste retorna los datos mediante una conexión TCP/IP. En SAP / ABAP, las RFC proveen una vía para acceder desde un programa externo (escrito en otro lenguaje) para usar datos devueltos por el servidor. También es posible llamar a una RFC desde el mismo sistema en que reside.

**Órdenes de Transporte:** El sistema R/3 dispone de una herramienta que nos permite pasar objetos de un ambiente (por ejemplo, desarrollo) a otro (por ejemplo, calidad o testing) conocida como sistema de transportes. Cuando hablamos de los distintos ambientes decimos que para pasar un desarrollo, parametrización, corrección, etc., de uno a otro lo hacemos por medio de una orden de transporte.

**Tabla de base de datos:** Objeto de SAP donde se almacena en forma permanente la información de los maestros de datos, documentos, parametrización, etc. Constituyen una estructura que en su conjunto conforman la denominada Base de datos de SAP.

**Tabla interna:** Objeto de datos que permite guardar en memoria grandes cantidades de registros de datos con la misma estructura. La cantidad de registros máxima está relacionada con los límites especificados en la instalación del sistema.

**Transacción:** Nomenclador para ejecutar un programa o una función en SAP R/3.

**SAP:** Es un sistema ERP para gestión de negocios que cuenta con diferentes módulos que encapsulan los procesos de negocios de una organización. Se basa en combinar todas las actividades de negocio en una solución informática simple, integrada, robusta y fiable.

**SAP GUI:** (Graphical User Interface) Programa de interfaz del usuario final que debe ser instalada en la máquina cliente para poder conectarse y trabajar con SAP. Programa mediante el cual el usuario puede intercambiar información con el ordenador de manera fácil e intuitiva.

**Vistas de bases de datos:** Es una tabla virtual en una base de datos relacional. Tienen la misma estructura que una tabla: filas y columnas, la única diferencia es que sólo se almacena de ellas la definición, no los datos.

**Workbench:** Entorno de desarrollo y todo lo que implique la manipulación de código fuente.

**Workflow:** Se refiere a un mecanismo que permite controlar una secuencia de acciones o un procedimiento que ejecutan varias personas. En general cuando se modela un proceso de negocios se definen las interacciones entre eventos, tareas y personas, pero no se considera explícitamente la secuencia de ejecución en términos de responsables y plazos. Es la automatización de un proceso de negocio, sea parcial o totalmente, durante el cual documentos, información o tareas son pasados desde un participante a otro para la ejecución de otra acción, de acuerdo a un conjunto de reglas procedurales.

## Bibliografía:

**Tema** Sistemas ERP  
**Título** Sistemas de Información Integrados (ERP)  
**URL** <http://www.elmostrador.cl/media/2015/05/nt6.pdf>  
**Autor** Asociación Española de contabilidad y Administración de Empresas (AECA)  
**Editorial** Gráficas ORMAG, 1era Edición, Madrid  
**Fecha/Consulta** mayo/2015

**Título** Implementación de sistemas ERP, su Impacto en la Gestión de la Empresa e Integración con otras TIC  
**URL** <http://www.capic.cl/capic/media/ART3Benvenuto.pdf>  
**Autor** Ángelo Benvenuto Vera, Universidad de Concepción, Chile  
**Editorial** Universidad de concepción, 2006.  
**Fecha/Consulta** mayo/2015

**Título** Sistemas Integrados de Gestión (ERP)  
**URL** [http://www.gcd.udc.es/subido/catedra/presentaciones/economia\\_competencia\\_ii/nota\\_tecnica\\_sistemas\\_de\\_gestion\\_erp\\_carlos\\_suarez\\_rey\\_17-03-2010.pdf](http://www.gcd.udc.es/subido/catedra/presentaciones/economia_competencia_ii/nota_tecnica_sistemas_de_gestion_erp_carlos_suarez_rey_17-03-2010.pdf)  
**Autor** Carlos Suárez Rey, 2010  
**Fecha/Consulta** mayo/2015

**Título** Sistemas Integrados de Gestión Empresarial. Evolución histórica y tendencias de futuro.  
**Autor** Raúl Oltra Badenes  
**Editorial** Universidad Politécnica de Valencia, 1era Edición, España, 2012  
**Fecha/Consulta** mayo/2015

**Título** Así es SAP R/3  
**Autor** Jose Antonio Hernandez Muñoz  
**Editorial** MacGraw Hill  
**Fecha/Consulta** junio/2015

**Tema** Ingeniería de software  
**Título** Ingeniería de software. Un enfoque práctico  
**Autor** Roger S. Pressman  
**Editorial** Mc Graw Hill  
**Fecha/Consulta** julio/2015

## Referencias Electrónicas:

**Tema** Historia, actividades y servicios de La empresa  
**URL** <http://www.la empresa.com.mx/>  
**Autor** Comercializadora de Motocicletas, S.A. de C.V. (La empresa)  
**Fecha/Consulta** mayo/2015

**Tema** Modelos de motocicletas La empresa  
**URL** <http://www.la empresa.com.mx/Refacciones/App/Default.aspx>  
**Autor** Comercializadora de Motocicletas, S.A. de C.V. (La empresa)  
**Fecha/Consulta** mayo/2015

**Tema** Historia de SAP  
**URL** <http://www.sap.com/corporate-en/about/our-company/history/index.html>  
**Autor** SAP  
**Fecha/Consulta** mayo/2015

**Tema** Descripciones diversas de SAP y ABAP  
**URL** <http://www.teknodatips.com.ar>  
**Autor** Teknoda (empresa de tecnología informática)  
**Fecha/Consulta** mayo/2015

**Tema** SAP GUI  
**URL** [http://help.sap.com/saphelp\\_nw73/helpdata/en/4d/69270ba5bd1745e10000000a42189b/frameset.htm](http://help.sap.com/saphelp_nw73/helpdata/en/4d/69270ba5bd1745e10000000a42189b/frameset.htm)  
**Autor** SAP  
**Fecha/Consulta** mayo/2015

**Tema** ABAP  
**URL** [http://help.sap.com/abapdocu\\_70/en/](http://help.sap.com/abapdocu_70/en/)  
**Autor** SAP  
**Fecha/Consulta** mayo/2015

## **Anexos**

**Anexo A - Sistema SAP**

**Anexo B - Descripción del Entorno de Desarrollo ABAP**

**Anexo C - Ingeniería de Software**

## Anexo A – Sistema SAP

---

### SAP Business Suite

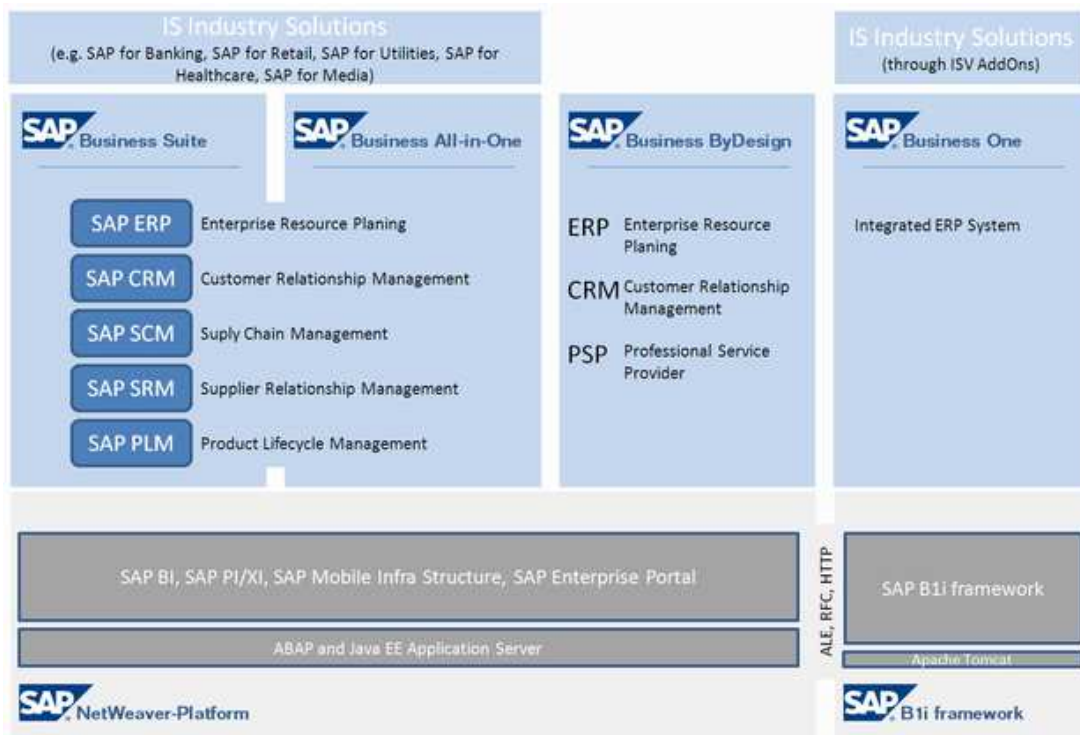
SAP Business Suite es un conjunto de programas que permiten a las empresas ejecutar y optimizar distintos aspectos como los sistemas de ventas, finanzas, compras, fabricación, etc. Ofrece la posibilidad de realizar procesos específicos de la empresa o crear módulos independientes para funcionar con otro software de SAP o de otros proveedores.

Está basado en una plataforma de tecnología integrada llamada NetWeaver. La suite puede soportar sistemas operativos, bases de datos, aplicaciones y componentes de hardware de casi cualquier proveedor.

SAP Business Suite está dividido en cinco módulos o aplicaciones, que son:

- Administración basada en la relación con los clientes (**CRM** - Customer Relationship Management) - ayuda a las empresas a adquirir y retener a los clientes, aumentando el marketing y la comprensión del cliente.
- Planificación de Recursos empresariales (**ERP** – Enterprise Resource Planning) - Da soporte a las funciones esenciales de los procesos y operaciones de la empresa.
- Administración del ciclo de vida de productos (**PLM** - Product Lifecycle Management) - ayuda a los fabricantes con información sobre el producto.
- Administración de la cadena de suministro (**SCM** - Supply Chain Management) - ayuda a las empresas con el proceso de asignación de recursos de su fabricación y procesos de servicio.
- Gestión de relaciones con los proveedores (**SRM** - Supplier Relationship Management) - permite a las empresas mejorar los mecanismos de suministro de sus proveedores.

## Anexo A – Sistema SAP



### SAP Business Suite

Fuente: Blog de Heinz Pauly, Enero 2011.

SAP Business One – A Member of the SAP Family.

## Área Laboral de SAP

En SAP existen diversas áreas de trabajo, en primera instancia se definen dos, por un lado están los que usan SAP como herramienta diaria y por otro están los consultores SAP:

### Usuarios SAP

Son las personas que usan SAP para ejecutar sus labores diarias dentro de la empresa. Por ejemplo:

- Una de las tareas de la gente de finanzas es manejar la facturación y pagos a proveedores por medio de SAP. Estos usuarios son catalogados como Usuarios de SAP FI por ser FI el módulo de finanzas.
- Una de las tareas de la gente de recursos humanos es calcular aguinaldos por medio de SAP. Estos usuarios son catalogados como Usuarios de SAP HCM por ser HCM el módulo de Recursos humanos que manejan.

Se limitan a usar el sistema y cómo podemos ver existen varias clases de usuarios de SAP, cada usuario especializado en su área de trabajo.

## Anexo A – Sistema SAP

### Consultores SAP

Es un profesional especializado en algún área de SAP quien administra, releva, configura, y prepara el programa para que SAP funcione perfectamente en una empresa, así como también da soporte y/o mantenimiento al sistema.

Existen a su vez dos clases de Consultores SAP:

- Los **Consultores Funcionales**, que se especializan en alguno de los módulos del sistema. SAP tiene muchos módulos y submódulos, regularmente un consultor SAP se especializa en UNO de esos módulos, del cual se encarga de configurar y administrar de acuerdo a las necesidades del cliente.
- Los **Consultores Técnicos**, que se especializan en alguna de las áreas técnicas del sistema. Participan en la programación o administración técnica del sistema SAP. Algunas áreas técnicas son: Basis/NetWeaver, ABAP, BI BW, XI/PI, etc.

Cada área tiene distintas tareas y responsabilidades que en conjunto logran que el sistema SAP funcione eficazmente.

Algunos de los módulos y submódulos funcionales SAP se muestran en la siguiente figura:

<b>SD VENTAS Y DISTRIBUCION</b> SD - MD Datos maestros SD-SLS Gestión de Ventas SD-GF Gestión Tarifas y Condiciones de Precio SD-SHP Gestión de Expediciones SD-BIL Facturación SD-IS Sistemas de Información SD-EDI Intercambio Electrónico de Datos	<b>MM GESTION DE MATERIALES</b> MM - MRP Planificación Necesidades Materiales MM-PUR Gestión de Compras MM-IM Gestión de Inventarios MM-WM Gestión de Almacenes MM-IV Verificación de Facturas MM-IS Sistema de Información MM-EDI Intercambio Electrónico de Datos Sistema Clasificación Gestión de Lotes	<b>PS GESTION DE PROYECTOS</b> PS-BD Datos Basicos PS-OS Planificación del proyecto PS-PLN Plan de Costes PS-APM Proceso de Aprobación PS-EXE Seguimiento y Progreso del Proyecto PS-IS Sistema de Información
<b>FI CONTABILIDAD FINANCIERA</b> FI-GL Cuentas de Mayor FI-LC Consolidación Sociedades FI-AR Cuentas a Cobrar FI-AP Cuentas a Pagar FI-AA Gestión de Activos FI-SL Special Ledger Cierres	<b>PP PRODUCCION</b> PP-BD Datos Basicos PP-SOP Gestión de la Demanda PP-MP Plan Maestro PP-CRP Plan de Capacidades PP-MRP Plan de Materiales PP-SFC Ordenes de Fabricación PP-PC Costes de producto PP-IS Sistema de Información PP-PI Industria de procesos PP-CFG Configuración de Producto	<b>PM GESTION DEL MANTENIMIENTO</b> PM-EQM Identificación Descripción PM-PRM Mantenimiento Preventivo PM-WOC Ordenes de Mantenimiento PM-PRO Proyectos de Mantenimiento PM-SM Gestión del Servicio
<b>QM CALIDAD</b> QM-PT Herramientas de planificación QM-IM Proceso de Inspección QM-QC Control de Calidad QM-CA Certificados de Calidad QM-QN Notificaciones de Calidad		<b>CO CONTROLLING</b> CO-CCA Contabilidad por Centros Coste Contabilidad Presupuestaria CO-PC Control de Costes del Producto CO-PA Analisis de Rentabilidad CO-OPA Ordenes Internas CO-ABC Costes Basados en Actividades

### Módulos y Submódulos de SAP

Fuente: Blog de Christian Camilo Jaimes Duran. Julio 2015.

SAP



## Anexo A – Sistema SAP

### Estructura del Sistema SAP

Cuando se implementa el sistema SAP se debe adaptar a las necesidades específicas de la empresa cliente. El proceso de adaptación del software se conoce como **Customizing**.

Se utilizan parametrizaciones del Customizing para definir las estructuras organizativas del cliente en el sistema, tales como mandantes, sociedades, canales de distribución, etc. y para fijar parámetros de transacciones SAP de tal forma que puedan reflejar procesos específicos de cliente.



#### Estructura de datos de sistemas SAP

Fuente: Elaboración Propia.

Un **Mandante** en SAP es una unidad dentro del sistema desde un punto de vista técnico, organizativo y comercial, un ambiente de trabajo dentro de un ambiente o entorno. Los mandantes tienen sus propios registros maestros y su propio conjunto de tablas. Es el nivel jerárquico más alto en el Sistema SAP. Se caracteriza por tener su propio entorno de datos empresariales, así como sus propios datos variables, maestros y de usuario.

El **Repository** es el almacén central para todos los componentes de desarrollo y es válido en todos los mandantes. Incluye los modelos de procesos, modelos de datos, programas ABAP, funciones, screens, etc. es decir, todos los datos y sus relaciones. También incluye el diccionario ABAP.

El **Diccionario ABAP** es el centro donde se almacenan sin redundancia y manejan todas las definiciones de todos los datos del sistema. Está completamente integrado en el Workbench ABAP. Soporta la definición de tipos definidos por el usuario (dominios, elementos de datos, estructuras, etc.).

## Anexo A – Sistema SAP

---

También se puede definir la estructura de objetos de la base de datos (tablas, índices y vistas). Proporciona automáticamente toda la información que se crea o modifica, asegurando así que los objetos en tiempo de ejecución son actuales y que los datos son consistentes y seguros.



**Diccionario de Datos - Tx SE11**

Fuente: Impresión directa del sistema.

En SAP, para poder ejecutar un programa se hace por medio de una **transacción**. Regularmente son invocados mediante el ingreso del código de transacción directamente en el campo de comandos:



**Campo de Comandos**

Fuente: Impresión directa del sistema.

**ABAP Workbench** es un conjunto de programas (transacciones) para el desarrollo y modificación de los programas ABAP, es el entorno gráfico para escribir programas de aplicación y aplicaciones web. La última versión incluye ABAP Objects, un lenguaje de programación orientado a objetos.

## Anexo A – Sistema SAP

---

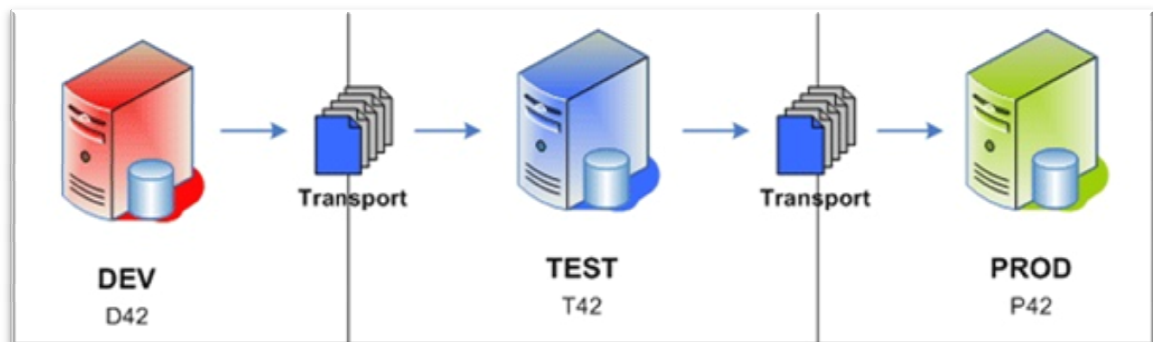
### Sistema de Transportes

Las empresas que tienen instalado SAP se encuentran realizando modificaciones, mejoras o implementando nuevas funcionalidades en su software además de continuar con sus actividades de negocio diarias. Por ello es necesario tener diferentes sistemas o entornos:

- Sistema de Desarrollo: entorno abierto específicamente para la parametrización y programación por parte del equipo de consultoría. No se tienen datos reales del negocio en la base de datos.
- Sistema de Test: donde los usuarios y el equipo de consultoría pueden verificar los nuevos procesos desarrollados en el sistema de desarrollo. Tampoco se tienen datos reales.
- Sistema Productivo: entorno de trabajo de los usuarios finales donde se encuentran datos reales de la empresa y se concentra su día a día. Es siempre el sistema de destino donde acaban las modificaciones iniciadas en el sistema de Desarrollo.

Cuando se crea un objeto, el sistema automáticamente pide una Orden de Transporte para guardarlo, ya sea nueva o existente.

En una arquitectura de dos o tres sistemas los transportes son necesarios para transferir de un sistema a otro los objetos desarrollados.



#### Sistema de Transportes en SAP

Fuente: BIZEC - Three-Tier System Landscape.

### SAP GUI

Para iniciar sesión en el sistema SAP se necesita SAP GUI (Graphical User Interface). Es un software que se puede ejecutar en Microsoft Windows, Apple Macintosh o en el escritorio Unix y permite al usuario acceder a la funcionalidad de SAP en aplicaciones como SAP ERP y SAP Business Intelligence.

## Anexo A – Sistema SAP

---

Es la plataforma utilizada para el acceso remoto al servidor central SAP desde una computadora en la red de una empresa.

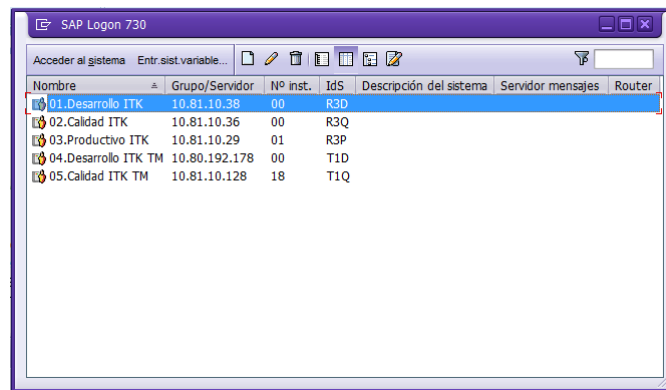
SAP GUI se instala en cada computadora que requiera ingresar a SAP y para iniciar sesión se utiliza el programa que lleva incluido: *SAP Logon*. Está entre el sistema SAP y la interfaz de usuario SAP GUI.



### Acceso Directo a SAP Logon

Fuente: Impresión directa del sistema.

Al ejecutar SAP Logon se muestra una lista de los sistemas SAP disponibles y se pueden añadir y/o eliminar elementos a la lista.



### Ejemplo de SAP Logon

Fuente: Impresión directa del sistema.

Para iniciar sesión se da doble click sobre el sistema al que se quiera ingresar y se solicita el usuario y la contraseña que el administrador del sistema (BASIS) nos proporciona.

Una captura de pantalla del formulario de inicio de sesión de SAP. El título es 'SAP'. Hay un campo 'Clave acceso nueva'. El campo 'Mandante' tiene el valor '500'. El campo 'Usuarios' tiene un cursor. El campo 'Clv.acc.' tiene asteriscos. El campo 'Idioma' tiene el valor 'ES'.

<b>SAP</b>	
Clave acceso nueva	
Mandante	500
Usuarios	
Clv.acc.	*****
Idioma	ES

### Inicio de Sesión de SAP

Fuente: Impresión directa del sistema.

## Anexo A – Sistema SAP

### SAP GUI para HTML

En ocasiones se requiere tener acceso a SAP desde cualquier lugar mediante un navegador web, sin necesidad de tener instalado el cliente SAP GUI, por ejemplo, los usuarios en itinerancia o que trabajan con aplicaciones muy específicas (grabación de gastos de viaje, horas de visitas a clientes, pedidos de venta, etc).

Para ello se utiliza SAP GUI para HTML. Es un elemento disponible en el sistema SAP al que se accede a través del ITS (Internet Transaction Server).

Desde las versiones ECC 5.0 y ECC 6.0, el ITS es un elemento integrado en el Application Server para ABAP (AS ABAP) y no hay que realizar ninguna instalación adicional, sólo se debe activar y configurar los servicios necesarios siguiendo los siguientes pasos:

1. Se activan los servicios web necesarios a través de la transacción (Tx) SICF. Al ingresar se localizan los servicios /sap/public/bc/its/mimes, /sap/public/bc/ur y /sap/bc/gui/sap/its/webgui y se activan utilizando la jerarquía SERVICES. Si falta algún servicio más por activar, el propio sistema nos avisará al hacer logon.



**Tx. SICF, Servicio: /sap/bc/gui/sap/its/webgui**

Fuente: Impresión directa del sistema.

2. Se prepara la publicación de servicios a través de la Tx SE80, en la opción de menú Utilidades -> Opciones -> pestaña Servidor ITS -> pestaña Publicar y se marca la opción Integrated ITS. Esto restringe la publicación de los servicios web únicamente al Site Internal del ITS integrado.

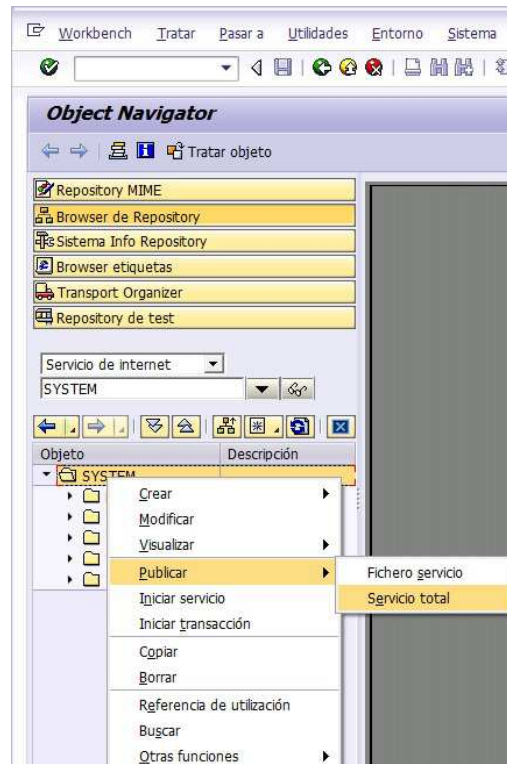
## Anexo A – Sistema SAP



**Tx. SE80, pestaña Servidor ITS.**

Fuente: Impresión directa del sistema.

3. Se publican los servicios desde la Tx SE80, donde se selecciona el tipo de objeto "Servicio de Internet" y se buscan los servicios SYSTEM y WEBGUI. Con el menú contextual, se selecciona la opción Publicar -> Servicio Total.



**Tx. SE80, Publicación de Servicios.**

Fuente: Impresión directa del sistema.

4. Puede ser necesario revisar algún parámetro de la instancia SAP, referente a los puertos de los servicios web. A través de la transacción RZ10 (parámetros icm/server\_port\_0 y otros que comienza por icm/).

Se puede acceder a SAP vía web por medio de la ruta genérica:  
**<http://<server>:<icmport>/sap/bc/gui/sap/its/webgui/>**:

## Anexo A – Sistema SAP

---



### Ejemplo de WEBGUI en un ambiente de Desarrollo

Fuente: Impresión directa del sistema.

El aspecto del entorno gráfico es muy similar al que se tiene con SAPGUI.



## Anexo B - Descripción del Entorno de Desarrollo ABAP

### Lenguaje de Programación ABAP

ABAP (Advanced Business Application Programming) es un lenguaje de programación propiedad de SAP para el desarrollo de sus aplicaciones de negocios. Con este lenguaje se desarrolló el sistema SAP R/3, pero además es el lenguaje con el que los clientes pueden agregar su propio código para crear reportes, interfaces o cualquier tipo de programa específico para su negocio.

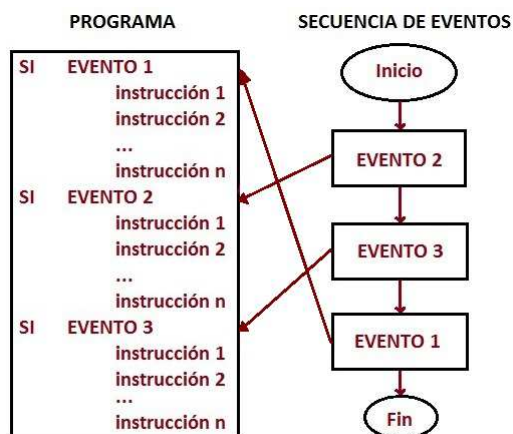
No es un lenguaje en el que podemos realizar un ejecutable para correr en cualquier equipo. Los programas ABAP son ejecutables únicamente dentro del ambiente SAP.

Originalmente nació como un lenguaje estructurado de cuarta generación (de ahí el antiguo nombre de ABAP/4) y sigue siendo el lenguaje para desarrollar en SAP R/3, pero con el avance del hardware se fue adaptando hasta que en 1999, con el lanzamiento de la versión 4.6 de R/3, SAP lanzó una extensión orientada a objetos denominada ABAP Objects. La última plataforma de desarrollo de SAP, NetWeaver, soporta ABAP y Java.

ABAP utiliza sentencias de Open SQL para conectarse con prácticamente cualquier base de datos sin importar cuál sea el administrador utilizado en el sistema (Oracle, IMB DB2, SQL Server, etc.) y cuenta con miles de funciones para el manejo de archivos, fechas, etc. así como conexiones RFC (Remote Function Calls) para conectar a los sistemas SAP con cualquier otro sistema o lenguaje de programación.

### Modelo de programación

ABAP es un lenguaje estructurado orientado a eventos, es decir que la secuencia de las instrucciones depende del cumplimiento de una condición o evento. Se basa en subrutinas y módulos de función, y desde el release 6.10 se basa en clases e interfaces. Ambos modelos son interoperables.



### Lenguaje Orientado a Eventos

Fuente: Elaboración Propia.



## Anexo B - Descripción del Entorno de Desarrollo ABAP

---

### Características del lenguaje ABAP

Como lenguaje estructurado:

- Contiene sentencias y palabras clave propias del lenguaje.
- Es un lenguaje orientado a eventos bien definidos.
- Es interpretado, no compilado.
- Se utiliza tanto en programación de informes como en programación de diálogo para SAP.
- Se encuentra completamente integrado dentro del entorno de desarrollo de SAP.

Como lenguaje orientado a objetos contiene:

- Objetos.
- Clases.
- Atributos.
- Métodos
- Interfaces.

Se podría decir que ABAP se comporta como lenguaje de cuarta generación y como lenguaje orientado a objetos a la vez dentro de SAP.

### Para qué sirve el lenguaje ABAP

Los usuarios interactúan con SAP a través de transacciones. Estas transacciones representan procesos y funcionalidades propias del negocio de las empresas.

ABAP es el lenguaje utilizado por programadores para crear transacciones que no existen en el estándar de SAP, pero también sirve para ampliar o modificar transacciones estándar existentes cuando la funcionalidad que proveen es insuficiente para el negocio.

En el caso particular de SAP, no se permite modificar el código fuente de un programa estándar. Por ello proporciona recursos e instrumentos que permiten intromisiones "controladas" al código, a través del concepto de ampliaciones o enhancements. La idea es expandir la funcionalidad dentro del sistema SAP para atender las necesidades adicionales del cliente, sin modificar el código fuente del programa estándar.

Las sucesivas versiones de SAP fueron incorporando nuevos y mejores instrumentos para llevar a cabo esta tarea. Ya que SAP está programado en ABAP, todas las funciones complementarias al estándar se han de programar también en ABAP mediante alguna de las formas que SAP permite (user exits, BADIs, BAPIs, enhancements, etc.).

## Anexo B - Descripción del Entorno de Desarrollo ABAP

---

### Herramientas de ampliación de SAP

- **USER EXITS:** Es la primera generación de ampliaciones. Son “subrutinas vacías” dentro de un programa estándar en las cuales se puede agregar código. Este método utiliza módulos de función mediante los cuales se puede ingresar código ABAP. El sistema provee las funciones donde los user-exits son llamados y se cuenta con una interfaz definida y un programa *Include* donde se agrega el código requerido. Los user exits NO están definidos dentro de un proyecto de ampliación, directamente se modifica el código estándar. Existen básicamente en el módulo de SD.
- **CUSTOMER EXITS:** Se definen dentro de un proyecto de ampliación, por lo tanto se gestionan con la transacción CMOD. Permiten ampliar código fuente (function module exit), menús (menu exit) y pantallas estándares (screen exit), y no permiten múltiples implementaciones.
- **BTE's:** Con la versión 4.0, aparecieron también los llamados BTE (Business Transaction Events). Son ampliaciones desarrolladas para el módulo de Finanzas (FI), que permiten agregar componentes adicionales en los programas estándares por medio de funciones. Una característica común es que son independientes de mandante a diferencia de las ampliaciones denominadas “Customer Exits”.
- **BADIS:** Con la versión 4.6C, y de la mano de la orientación a objetos, apareció la 3era generación de recursos de ampliación, a través de las BADIS (Business Add-ins), que son las nuevas técnicas de ampliación basadas en programación ABAP orientada a objetos. Usan instancias de ABAP Objects. Se invocan con CALL METHOD. Se crean con la transacción SE18, se implementan con la transacción SE19 y se puede hacer más de una vez al igual que las BTE's y a diferencia de los CUSTOMER EXITS.
- **ENHANCEMENT FRAMEWORK:** A partir de la versión 6.0 del ERP, SAP brinda un nuevo framework para el desarrollo de ampliaciones: Enhancement Framework. Es una nueva tecnología que brinda todas las técnicas de ampliación bajo un mismo techo. Las diferentes técnicas son las siguientes:
  - a. Ampliaciones de código (plug-ins)
  - b. Ampliaciones de grupo de función
  - c. Ampliaciones de clases
  - d. Ampliaciones Kernel-BADI

Dentro de este Framework se diferencian las ampliaciones de tipo implícito, predefinidas y provistas por SAP en el código fuente, y las ampliaciones de tipo explícito que pueden implementarse en cualquier

## Anexo B - Descripción del Entorno de Desarrollo ABAP

parte del código fuente a excepción de algunas áreas restringidas por SAP, a través de los ENHANCEMENT-POINTS y ENHANCEMENT-SECTION.

### Entorno de desarrollo ABAP

Dentro de SAP, el lenguaje ABAP posee su propio entorno de desarrollo denominado **ABAP Workbench**. Este entorno no es más que un conjunto de transacciones necesarias para poder programar ABAP, todas ellas agrupadas bajo un área común.



#### ABAP Workbench

Fuente: Impresión directa del sistema.

Algunas de las transacciones más importantes dentro del ABAP Workbench son:

#### SE11 – Diccionario de datos

Bajo la transacción SE11 se agrupan las herramientas necesarias para la creación, mantenimiento y ampliación de objetos de la base de datos. Tablas, estructuras, elementos de datos, dominios, etc. se generan a través del diccionario de datos. Pero también nos permite visualizar el contenido de las

## Anexo B - Descripción del Entorno de Desarrollo ABAP

---

diferentes tablas del sistema tanto aquellas creadas por nosotros como los que son propiedad de SAP.

### **SE37 – Biblioteca de Funciones**

Es la herramienta utilizada para crear, modificar o visualizar todas las partes de un módulo de función marcando las distintas opciones de objetos parciales.

### **SE38 – Editor ABAP**

Es la herramienta básica de programación. Permite la programación de informes y del código asociado a pantallas de diálogo. También nos permite realizar ampliaciones de contenido sobre el código propietario de SAP.

### **SE51 – Screen painter**

Es la herramienta utilizada en la creación y desarrollo de pantallas de diálogo en SAP (dynpros), sin embargo la programación ABAP de la estructura de pantallas se realiza conjuntamente con el editor ABAP.

### **SE80 – Object Navigator**

Agrupar todas o casi todas las transacciones de desarrollo ABAP en una sola transacción. Nos permite trabajar a la vez con diferentes herramientas de programación bajo un área común.

### **SE10/STMS – Sistema gestor de transportes**

Las modificaciones o nuevos programas en el entorno de desarrollo se deben “transportar” al entorno productivo a través de un sistema que comunica todos los entornos. Éste sistema es el gestor de transportes y las transacciones SE10 y STMS son sus transacciones más conocidas.

Finalmente, hay que decir que el ABAP Workbench es propio de SAP pero últimamente SAP con las opciones orientadas a objetos, también permite programar ABAP mediante herramientas no propietarias de SAP, sobre todo con Eclipse. Sin embargo, con estas plataformas se pierde mucha de la funcionalidad propia del ABAP Workbench.

ABAP es un lenguaje desarrollado específicamente para el procesamiento masivo de datos, por ello ofrece las siguientes ventajas:

- El acceso de base de datos utilizando sentencias de Open SQL para conectarse con prácticamente cualquier base.
- La optimización del rendimiento de la base de datos de accesos integrado en el entorno de ejecución ABAP mediante el uso de buffering SAP.
- Las tablas internas para el almacenamiento dinámico y tratamiento de los datos tabulares de masas en la memoria de trabajo.

## Anexo B - Descripción del Entorno de Desarrollo ABAP

---

- Cuenta con miles de funciones para el manejo de archivos, bases de datos, fechas, etc.
- Permite conexiones RFC (Remote Function Calls) para conectar a los sistemas SAP con cualquier otro sistema o lenguaje de programación.
- El lenguaje tiene una interfaz integrada a XML.

Se debe tener en cuenta que todo programa no estándar en ABAP tendrá que empezar SIEMPRE por la letra Z o Y ya que el resto de letras están reservadas para los programas propietarios de SAP.

## Anexo C – Ingeniería de Software

---

### Modelos de Proceso de Software

La ingeniería de software está formada por un conjunto de métodos, herramientas y técnicas que se utilizan en el desarrollo de los programas informáticos (software).

Esta disciplina trasciende la actividad de programación, que es el pilar fundamental a la hora de crear una aplicación, pues se encarga de toda la gestión del proyecto para que éste se pueda desarrollar en un plazo determinado y de manera correcta.

Incluye el análisis previo de la situación, el diseño del proyecto, el desarrollo del software, las pruebas necesarias para confirmar su correcto funcionamiento y la implementación del sistema.

El desarrollo de software implica lo que se conoce como ciclo de vida, que se utiliza para estructurar las actividades que se llevarán a cabo en el desarrollo de un producto.

Existen varios modelos de ciclo de vida y dependiendo de qué modelo se elija serán las etapas que lo componen y la forma en cómo se llevará a cabo la realización del proyecto. Independientemente del modelo que se trate, hay etapas en común:

- **Análisis:** Estudio de los requerimientos para determinar su viabilidad por parte de los desarrolladores y de la gente de negocio para establecer el alcance del producto.
- **Diseño:** Abstracción del sistema y elaboración de la interfaz de usuario.
- **Codificación:** Propiamente la elaboración del sistema como tal mediante algún lenguaje de programación.
- **Pruebas:** Verificación del correcto funcionamiento del producto.
- **Mantenimiento:** Correcciones y/o modificaciones al sistema.

### Modelos de Ciclos de Vida

Algunos de los modelos de ciclos de vida son:

**Modelo en cascada** o modelo lineal secuencial: Siguiendo el modelo de cascada de forma estricta, sólo cuando se finaliza una fase, comienza la siguiente. Una vez que se ha llegado al periodo de mantenimiento es posible regresar a etapas anteriores para hacer cambios, haciendo nuevamente el recorrido hasta llegar al final.

## Anexo C – Ingeniería de Software

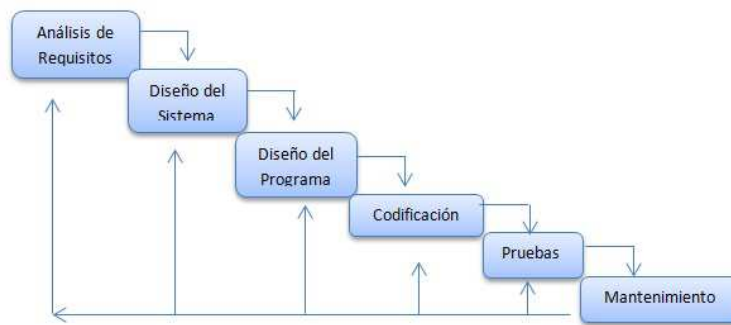


Fig. 1 fases del modelo en cascada

### Modelo en Cascada.

Fuente: <http://proyect-is.wikispaces.com/>.

**Modelo en V:** Está basado en el modelo en cascada y consta de las mismas fases, con la ventaja de que una fase también nos sirve para verificar o validar otras etapas. Este modelo dice que las pruebas necesitan empezarse lo más pronto posible para poder detectar errores.

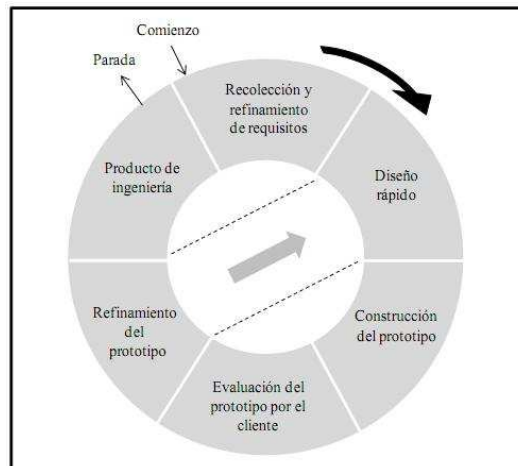


### Modelo en V.

Fuente: Issam El Bahri - Ciclo de vida del modelo V

**Modelo de prototipos:** define un conjunto de objetivos generales para el software, pero no identifica los requisitos detallados de entrada, proceso o salida.

Forma parte de los modelos de desarrollo evolutivo y es de gran ayuda cuando los requerimientos no están claramente establecidos, cuando no se sabe cómo implementar la interfaz de usuario o en algún otro caso que implique una gran cantidad de modificaciones y/o interacción con el usuario. Gracias a la retroalimentación se afinan detalles para empezar con el desarrollo formal del proyecto.

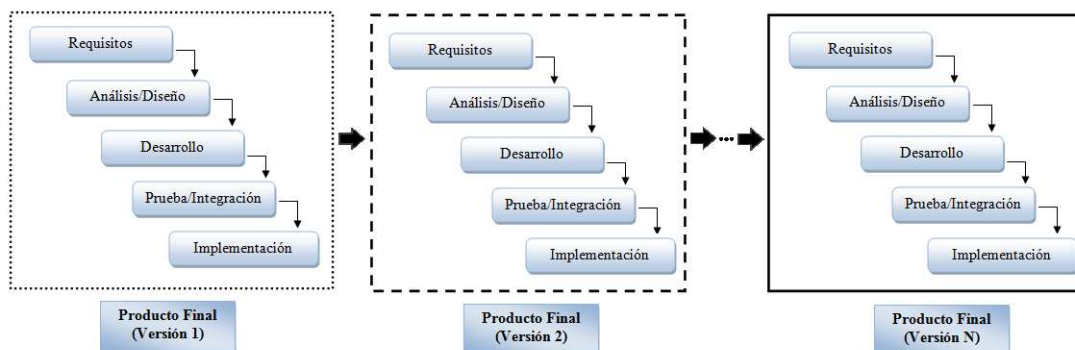


**Modelo de Prototipos.**

Fuente: <https://sisteminformacii.wikispaces.com/METODOLOG%C3%8DA+DE+ROGER+PRESSMAN>

**Modelo Incremental o Iterativo:** combina elementos del modelo en cascada con la filosofía interactiva de construcción de prototipos. Se basa en la filosofía de construir incrementando las funcionalidades del programa. Aplica secuencias lineales de forma escalonada mientras progresa el tiempo en el calendario.

Cada secuencia lineal produce un incremento del software. Los primeros incrementos son versiones incompletas del producto final, pero proporcionan al usuario la funcionalidad que precisa y también una plataforma para la evaluación.



**Modelo Incremental o Iterativo.**

Fuente: <http://procesosoftware.wikispaces.com/>

**Modelo en espiral:** Las actividades se conforman en una espiral, cada bucle representa un conjunto de actividades que se repiten dando como resultado un producto con mejoras respecto del anterior. Las actividades no están fijadas a priori, sino que las siguientes se eligen en función del análisis de riesgos, comenzando por el bucle anterior.



## Anexo C – Ingeniería de Software

---

Dicho análisis abarca aspectos como la mala interpretación de requerimientos, errores de diseño, problemas de implementación, etc. y es la diferencia al de prototipos.



**Modelo en Espiral.**

Fuente: <http://proyect-is.wikispaces.com/>.

La forma en que se desarrolló la aplicación de agencias propias corresponde al modelo en espiral, debido a que no se tenía establecido el requerimiento de manera completa inicialmente, pues sólo se tenía una visión general del proyecto y se fue perfeccionando durante la marcha.

### Mantenimiento del Software

El Mantenimiento del Software es la modificación de un producto software después de la entrega para corregir fallos, para mejorar el rendimiento u otros atributos, o para adaptar el producto a un entorno modificado.

El mantenimiento comienza cuando el producto se ha entregado, sin embargo se considera que algunas actividades de mantenimiento pueden comenzar antes de la entrega del producto.

Algunas de estas actividades son la planificación de las actividades posteriores a la entrega, así como toda actividad orientada a facilitar el mantenimiento, como la revisión de la documentación, aunque éstas pueden considerarse actividades de preparación para el mantenimiento más que de mantenimiento en sí.

Los tipos de mantenimientos existentes son:

**Correctivo:** es corregir un problema que tiene un software ante un funcionamiento incorrecto, deficiente o incompleto. También se puede definir como corrección de fallos detectados durante la explotación. Ejemplo de este mantenimiento, tenemos:

## Anexo C – Ingeniería de Software

---

1. Las actualizaciones que Windows hace para disminuir las vulnerabilidades.
2. Instalación de software antivirus para corregir daños que hayas sufrido con algún virus.
3. También implica buscar información inútil, programas residentes y demás software que no necesitas o que funciona incorrectamente.

**Preventivo:** es corregir un problema antes que se presente. También es facilitar el mantenimiento futuro del sistema (verificar precondiciones, mejorar legibilidad, etc.).

**Perfectivo:** son las acciones llevadas a cabo para mejorar la calidad interna de los sistemas en cualquiera de sus aspectos: reestructuración del código, definición más clara del sistema y optimización del rendimiento y eficiencia.

**Adaptativo:** son las modificaciones que afectan a los entornos en los que el sistema opera, por ejemplo, cambios de configuración del hardware, software de base, gestores de base de datos, comunicaciones, etc.

### Pruebas

Las pruebas son parte integral de un proyecto y del ciclo de vida de una aplicación. Dentro un proyecto de implementación, las pruebas siguen un enfoque estructurado que consiste en etapas de pruebas funcionales y no funcionales. Llevar a cabo una adecuada estrategia de pruebas, garantiza que la funcionalidad que se incorpore en el ambiente productivo cumpla con los requerimientos y expectativas de cada proceso. Adicionalmente reduce al mínimo las adecuaciones y ajustes que se tengan que realizar durante la fase de soporte.

- Las pruebas funcionales aseguran el correcto funcionamiento de los objetos de prueba y se suele realizar en varias etapas para cada entrega dentro del ciclo de vida de un proyecto.
- Las pruebas no funcionales lidian con el desempeño y el tiempo de ejecución del sistema.

### Niveles de Prueba del Software

En un proceso de pruebas formal, suelen confundirse con mucha facilidad, los niveles de pruebas con los tipos de prueba y a pesar de que se encuentren íntimamente relacionadas, tienen connotaciones diferentes en el proceso. Las pruebas pueden ejecutarse en cualquier punto del proceso de desarrollo de software y es aquí donde los niveles de prueba nos permiten entender con claridad los diferentes puntos o etapas en donde pueden ejecutarse ciertos tipos de prueba.

## Anexo C – Ingeniería de Software

---

Por lo anterior, es común que algunas personas se refieran a los niveles de pruebas o intenten clasificarlos como: pruebas de desarrollador, pruebas funcionales y pruebas de usuario final. Sin embargo, la terminología apropiada para referirse a los diferentes niveles corresponde a las siguientes cuatro clasificaciones que son: pruebas unitarias, pruebas de integración, pruebas de sistema y pruebas de aceptación. En cada uno de estos niveles de prueba, se podrán ejecutar diferentes tipos de prueba tales como: pruebas funcionales, no funcionales, de arquitectura y asociadas al cambio de los productos.

A continuación una breve descripción de cada nivel de prueba:

**Pruebas Unitarias o de Componente:** son ejecutadas normalmente por el equipo de desarrollo, básicamente consisten en la ejecución de actividades que le permitan verificar al desarrollador que los componentes unitarios están codificados bajo condiciones de robustez, esto es, soportando el ingreso de datos erróneos o inesperados y demostrando así la capacidad de tratar errores de manera controlada. Adicionalmente las pruebas sobre componentes unitarios suelen denominarse pruebas de módulos o pruebas de clases, siendo la convención definida por el lenguaje de programación la que influye en el término a utilizar.

**Pruebas de Integración:** son ejecutadas por el equipo de desarrollo y consisten en la comprobación de que elementos del software que interactúan entre sí, funcionan de manera correcta.

**Pruebas de Sistema:** deben ser ejecutadas idealmente por un equipo de pruebas ajeno al equipo de desarrollo, una buena práctica en este punto corresponde a la tercerización de esta responsabilidad. La obligación de este equipo, consiste en la ejecución de actividades de prueba en donde se debe verificar que la funcionalidad total de un sistema fue implementada de acuerdo a los documentos de especificación definidos en el proyecto. Los casos de prueba a diseñar en este nivel de pruebas deben cubrir los aspectos funcionales y no funcionales del sistema. Para el diseño de los casos de prueba en este nivel, el equipo debe utilizar como bases de prueba entregables tales como: requerimientos iniciales, casos de uso, historias de usuario, diseños, manuales técnicos y de usuario final, etc. Por último, es importante que los tipos de pruebas ejecutados en este nivel se desplieguen en un ambiente de pruebas / ambiente de pre-producción cuya infraestructura y arquitectura sea similar al ambiente de producción.

**Pruebas de Aceptación:** es indispensable que el cliente designe a personal que haga parte de los procesos de negocio para la ejecución de pruebas de aceptación, es incluso recomendable que los usuarios finales que participen en este proceso, sean independientes al personal que apoyó el proceso de desarrollo.

## Anexo C – Ingeniería de Software

---

Cuando las pruebas de aceptación son ejecutadas en instalaciones o ambientes proporcionados por la firma desarrolladora se les denominan **pruebas Alpha**, cuando son ejecutadas desde la infraestructura del cliente se les denomina **pruebas Beta**. En los casos en que las pruebas de aceptación del producto se hayan ejecutado en el ambiente del proveedor, el aplicativo no podrá salir a producción, sin que se hayan ejecutados las respectivas pruebas Beta en el ambiente del cliente. De lo anterior se concluye que las pruebas Alpha son opcionales, pero las pruebas Beta son obligatorias.

### Metodología de Pruebas

Los procesos de aseguramiento de calidad de un producto de software suelen dividirse en lo que respecta a su componente analítico en pruebas estáticas y dinámicas. La diferencia fundamental radica en que las pruebas estáticas se centran en evaluar la calidad con la que se está generando la documentación del proyecto por medio de revisiones periódicas, mientras que las dinámicas requieren de la ejecución del software con el fin de medir el nivel de calidad con la que este fue codificado y el nivel de cumplimiento en relación con la especificación del sistema.

Realizar pruebas dinámicas a un producto de software suele confundirse con una simple actividad de ejecución de pruebas y reporte de incidencias, sin embargo, para productos de complejidad media en adelante lo recomendable es implementar de manera formal una metodología de pruebas que se ajuste y acople uniformemente con la metodología de desarrollo seleccionada.

Para procesos de desarrollo basados en la metodología RUP o métodos tradicionales, implementar una metodología de pruebas es totalmente viable, teniendo en cuenta que estas metodologías están orientadas a la documentación y a la formalización de todas las actividades ejecutadas. Si por el contrario, el proceso se guía bajo lineamientos basados en metodologías ágiles será necesario reevaluar la conveniencia de ejecutar todas las actividades que implica un proceso de pruebas formal, lo que en la mayoría de los casos conlleva a reducir al mínimo las actividades relacionadas con un proceso de pruebas, circunstancia que naturalmente puede desencadenar en la liberación de productos con bajos niveles de calidad.

Un proceso de pruebas formal, está compuesto, cuando menos por las siguientes 5 típicas etapas:

- Planeación de pruebas.
- Diseño de pruebas.
- Implementación de pruebas.
- Evaluación de criterios de salida.
- Cierre del proceso.

## Anexo C – Ingeniería de Software

---

**Planeación de Pruebas:** Es la etapa en donde se ejecutan las primeras actividades correspondientes al proceso de pruebas y tiene como resultado un entregable denominado plan de pruebas el cual debe estar conformado cuando menos por aspectos tales como alcance de pruebas, tipos de pruebas, estrategia de pruebas, criterios de salida, entre otros aspectos.

**Diseño de Pruebas:** Una vez elaborado y aprobado el plan de pruebas, el equipo de trabajo debe iniciar el análisis de toda la documentación existente con respecto al sistema, con el objeto de iniciar el diseño de los casos de prueba. Los entregables claves para iniciar este diseño pueden ser: casos de uso, historias de usuario, arquitectura del sistema, diseños, manuales de usuario (si existen), manuales técnicos (si existen), etc. El diseño de los casos, debe considerar la elaboración de casos positivos y negativos. Los casos de prueba negativos permiten validar cómo se comporta el sistema ante situaciones atípicas y permite verificar la robustez del sistema, atributo que constituye unos de los requerimientos no funcionales indispensable para cualquier software. Por último, es necesario definir cuáles son los datos de prueba necesarios para la ejecución de los casos de prueba diseñados.

**Implementación y Ejecución de Pruebas:** La ejecución de pruebas debe iniciar con la creación de los datos de prueba necesarios para ejecutar los casos de prueba diseñados. La ejecución de estos casos, puede realizarse de manera manual o automatizada; en cualquiera de los casos, cuando se detecte un fallo en el sistema, este debe ser documentado. Una vez el defecto ha sido corregido es necesario realizar un re-test que permita confirmar que el defecto fue solucionado de manera exitosa. Por último, es indispensable ejecutar un ciclo de regresión que nos permita garantizar que los defectos corregidos no hayan desencadenado otros tipos de defectos en el sistema.

**Evaluación de Criterios de Salida:** Los criterios de salida son necesarios para determinar si es posible dar por finalizado un ciclo de pruebas. Para esto, es conveniente definir una serie de métricas que permitirán al finalizar un proceso de pruebas, comparar los resultados obtenidos contra las métricas definidas, si los resultados obtenidos no superan la métricas definidas, no es posible continuar con el siguiente ciclo de pruebas.

**Cierre del proceso:** Durante este periodo de cierre se deben cerrar las incidencias reportadas, se debe verificar si los entregables planeados han sido entregados y aprobados, se deben finalizar y aprobar los documentos de soporte de prueba, analizar las lecciones aprendidas para aplicar en futuros proyectos, etc.