

UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE QUITO – CAMPUS SUR

CARRERA DE INGENIERÍA DE SISTEMAS

MENCIÓN TELEMÁTICA

**SISTEMA DE GESTIÓN Y CONTROL DE RECURSOS FÍSICOS Y
LÓGICOS PARA LA GERENCIA DE CONSTRUCCIÓN ÁREA
PRUEBAS DE OTECEL. S.A**

TESIS PREVIA A LA OBTENCIÓN DEL TÍTULO DE:

INGENIERO EN SISTEMAS

ROBERTO CARLOS ZUMBA NARANJO

DIRECTOR: ING. RENÉ ARÉVALO

Quito, Junio 2012

DECLARACIÓN

Yo, Roberto Carlos Zumba Naranjo, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Universidad Politécnica Salesiana, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normatividad institucional vigente.

Roberto Carlos Zumba Naranjo

AGRADECIMIENTO

Un proyecto de tesis culminado satisfactoriamente no habría sido posible gracias al apoyo, confianza y consideración de varias personas y profesionales; antes y durante el desarrollo del mismo.

A todos ellos; personal administrativo y docente de una de las universidades más prestigiosas del Ecuador, un pilar fundamental en mi vida como lo es mi familia que un día soñó con leer estas líneas, un profesional exitoso que llegó a mi vida laboral y por el cual pude conseguirlo, ese profesional Jefe del Área de Pruebas de OTECEL. S.A - Efren Eduardo Zambrano Escobar.

Y un agradecimiento especial a mi tutor de tesis; Alonso René Arévalo por las horas dedicadas, el esfuerzo y sus sabios consejos.

DEDICATORIA

Dedico este éxito en primer lugar a Ilda Beatriz Naranjo Morales; quien es, ha sido y será padre y madre para mí. Una mujer excepcional digna de admiración por su responsabilidad y coraje para salir adelante con dedicación y trabajo.

En segundo lugar; a Efren Eduardo Zambrano Escobar, una persona por la cual este proyecto fue posible y por su excelente calidad humana por la que día a día me motiva a crecer como persona y profesional.

CONTENIDO

CONTENIDO	1-7
CONTENIDO DE TABLAS	8-10
CONTENIDO DE GRAFICOS	11-14
RESUMEN	15
CAPITULO 1. ANTECEDENTES	16-21
1.1. Introducción	17
1.2. Planteamiento del problema	18
1.3. Objetivos	19
1.3.1. Objetivo general	19
1.3.2. Objetivos específicos	19
1.4. Justificación	19
1.5. Alcance	20
1.6. Limitaciones	21
CAPITULO 2. MARCO TEORICO	22-80
2.1. Metodología de desarrollo MSF V4.0.....	22
2.1.1. Introducción	22
2.1.2. Microsoft Solution Framework	22
2.1.3. Principios Fundacionales	23
2.1.4. Modelos	23
2.1.4.1. Modelo de procesos.....	23
2.1.4.2. Modelo de roles	24

2.2. Plataforma y Framework de .NET	26
2.2.1. Introducción	26
2.2.2. Framework de .NET.....	26
2.2.3. Versiones.....	27
2.3. Aplicaciones WEB Cliente - Servidor	27
2.3.1. Introducción	27
2.3.2. Antecedentes.....	28
2.3.3. Cliente.....	29
2.3.4. Servidor	29
2.3.5. Cliente - Servidor	30
2.3.5.1. Tipos de servidores.....	30
2.3.5.2. Estilos del modelo Cliente - Servidor	31
2.4. Metodología de programación orientada a objetos	34
2.4.1. Introducción	34
2.4.2. Ámbitos con nombre	34
2.4.3. Clases.....	36
2.4.4. Campos de datos.....	37
2.4.4.1. Propiedades.....	37
2.4.4.2. Métodos	39
2.4.4.3. Eventos.....	40
2.4.4.4. Constructores y destructores	40
2.4.4.5. Clases anidadas	41
2.4.5. Jerarquía de clases.....	42
2.4.5.1. Herencia	43
2.4.5.2. Polimorfismo	45
2.5. Servidor WEB para el Sistema Operativo Microsoft Windows (IIS)	47
2.5.1. Introducción	47
2.5.2. Arquitectura	47
2.5.2.1. Petición GET	47

2.5.2.2. Esquema de una petición GET	48
2.5.2.3. Petición GET pasiva	49
2.5.2.4. Procedimiento del navegador	50
2.5.2.5. Petición POST	50
2.5.2.6. Composición de una petición POST	51
2.5.2.7. IIS V6 – WINDOWS SERVER 2003	52
2.6. Bases de datos ORACLE sobre Microsoft Windows.....	55
2.6.1. Introducción	55
2.6.2 Términos de bases de datos.....	56
2.6.3 Arquitectura de ORACLE.....	61
2.6.3.1. La base de Datos.....	61
2.6.3.2. La capa física.....	61
2.6.3.3. La capa lógica.....	62
2.6.3.4. Tablespaces y Datafiles	62
2.6.3.5. Esquema de base de Datos.....	62
2.6.4 PL/SQL	64
2.6.4.1. Tipos de sentencias	64
2.6.4.2. Tipos de datos	65
2.6.4.3. Funciones	66
2.6.4.4. Variables.....	69
2.6.4.5. Constantes.....	69
2.6.4.6. Bloque PL/SQL	70
2.6.4.7. Tipos de bloques.....	70
2.6.4.7.1. Anónimos sin nombre	70
2.6.4.7.2. Subprogramas	71
2.6.4.7.3. Funciones	72
2.6.4.7.4. Triggers.....	72
2.7. Técnicas y uso de encriptación de datos	74
2.7.1. Introducción	74
2.7.2. Funcionamiento de TDE	74

2.7.3. Implementación de TDE	75
2.7.4. Respaldos encriptados y RMAN	78
2.7.5. Métodos encriptados disponibles.....	79
2.7.6. Diferencias entre el TDE y los packages de encriptación	79
2.7.7. Consideraciones de performance y storage	79
2.7.8. Restricciones para el uso de TDE	80
CAPITULO 3. ANALISIS.....	81-116
3.1. Introducción	81
3.1.1. Objetivo.....	81
3.1.2. Ámbito.....	81
3.1.3. Alcance	81
3.1.4. Definiciones	82
3.1.5. Referencias.....	85
3.1.6. Visión general	86
3.2. Descripción general	86
3.2.1. Perspectiva del producto	86
3.2.2. Funciones del producto.....	87
3.2.2.1. Actividades para la administración de recursos.....	89
3.2.2.2. Actividades para la verificación y consumos del inventario	90
3.2.3. Características del usuario	94
3.2.4. Limitaciones generales	96
3.2.4.1. Elementos de software y hardware.....	96
3.2.4.2. Lenguajes de programación	96
3.2.4.3. Seguridad de información	96
3.2.4.4. Notificaciones de correos.....	96
3.2.5. Supuestos.....	97
3.3. Requisitos específicos	97
3.3.1. Requisito funcional	97
3.3.1.1. Ingreso al sistema.....	97

3.3.1.2. Usuario solicitante.....	98
3.3.1.3. Usuario administrador del inventario.....	100
3.3.1.4. Jefe del inventario.....	109
3.3.1.5. Administrador del sistema.....	110
3.3.2. Requisitos de interfaces externas.....	111
3.3.2.1. Interfaces de usuario.....	111
3.3.2.2. Interfaces de hardware.....	112
3.3.2.3. Interfaces de software.....	114
3.3.2.4. Interfaces de comunicación.....	114
3.3.3. Requisitos de ejecución.....	115
3.3.4. Limitaciones de diseño.....	115
3.3.5. Atributos.....	115
3.3.5.1. Seguridad.....	115
3.3.5.2. Mantenibilidad.....	115
3.3.6. Otros requisitos.....	116
3.3.6.1. Bases de datos.....	116
3.3.6.2. Operaciones.....	116
CAPITULO 4. PROPUESTA.....	117-184
4.1. Diseño.....	117
4.1.1. Elementos de navegación.....	117
4.1.1.1. Estructura general del aplicativo.....	117
4.1.1.2. Módulos.....	117
4.1.1.3. Solución gráfica.....	118
4.1.2. Maqueta del sistema.....	119
4.1.2.1. Inicio de sesión.....	119
4.1.2.2. Página maestra.....	119
4.1.2.3. Principal reportes.....	120
4.1.2.4. Principal asignar insumos.....	120
4.1.2.5. Principal crear insumos.....	120

4.1.2.6. Principal mantenedor de insumos.....	121
4.1.2.7. Principal consultar solicitudes.....	121
4.1.2.8. Principal crear usuarios.....	121
4.1.2.9. Principal modificar usuarios.....	122
4.1.2.10. Principal crear solicitudes.....	122
4.1.2.11. Principal procesar solicitudes.....	122
4.1.2.12. Principal perfiles.....	123
4.1.3. Diagramas.....	123
4.1.3.1. Diagrama de casos de uso.....	123
4.1.3.1.1. Módulo administración.....	123
4.1.3.1.2. Módulo reportes.....	128
4.1.3.1.3. Módulo servicios.....	133
4.1.3.2. Diagrama navegacional.....	142
4.1.3.3. Diagrama de clases UML.....	134
4.1.3.4. Diagrama de secuencia UML.....	151
4.1.3.5. Diagrama de actividades UML.....	162
4.1.3.6. Diagrama físico entidad - relación.....	174
4.1.3.6. Diccionario de datos.....	175
4.2. Código fuente.....	187
4.2.1. Script de base de datos.....	187
4.2.2. Métodos y funciones.....	194
CAPITULO 5. PRUEBAS Y DOCUMENTACION.....	205-209
5.1. Pruebas.....	205
5.1.1. Introducción.....	205
5.1.2. Pruebas Internas.....	207
5.1.3. Pruebas formales.....	208
5.2. Documentación.....	209
5.2.1. Introducción.....	209
5.2.2. Manual de administración y usuario (MAU).....	209

CONCLUSIONES	212
RECOMENDACIONES.....	213
BIBLIOGRAFIA	214
ANEXOS.....	215-297
Anexo 1. Plan Maestro Construcción Sistema Inventarios	216-226
Anexo 2. Tabulación resultados casos de prueba	227-232
Anexo 3. Estrategia pruebas Sistema de Gestión Inventario de Pruebas.....	233-235
Anexo 4. Informe Pruebas Sistema de Gestión Inventario de Pruebas	236-238
Anexo 5. Manual Usuario Sistema de Gestión Inventario de Pruebas.....	235-280
Anexo 6. Visión Alcance Sistema de Gestión Inventario de Pruebas	281-287

CONTENIDO DE LAS TABLAS

CAPÍTULO 2. MARCO TEORICO

2.1. Tabla: Modelo de roles	26
2.2. Tabla: Listado de versiones Framework .NET	28
2.3. Tabla: Versiones IIS.....	56
2.4. Tabla: Tipos de sentencias PL/SQL.....	66
2.5. Tabla: Tipos de datos PL/SQL.....	66
2.6. Tabla: Tipos de operadores lógicos PL/SQL	67
2.7. Tabla: Tipos de operadores aritméticos PL/SQL	67
2.8. Tabla: Tipos de operadores de caracteres PL/SQL	67
2.9. Tabla: Funciones PL/SQL.....	68
2.10. Tabla: Funciones para cadenas de caracteres	68
2.11. Tabla: Funciones de manejo de fechas	69
2.12. Tabla: Funciones de conversión de tipos.....	69
2.13. Tabla: Funciones de agrupamiento.....	69
2.14. Tabla: Otras funciones	69

CAPÍTULO 3. ANALISIS

3.1. Tabla: Referencias de los estándares IEEE830.....	86
3.2. Tabla: Plantilla especificación de requerimientos para OTECEL.S.A	87
3.3. Tabla: Actividades para la administración de los recursos de pruebas.....	91
3.4. Tabla: Actividades para la verificación de la operatividad y consumos del inventario de pruebas	95
3.5. Tabla: Especificación de requerimiento: Autenticar usuario	99
3.6. Tabla: Especificación de requerimiento: Modificar clave	99
3.7. Tabla: Especificación de requerimiento: Crear solicitudes.....	100
3.8. Tabla: Especificación de requerimiento: Consultar solicitudes	100
3.9. Tabla: Especificación de requerimiento: Reporte de insumos	101
3.10. Tabla: Especificación de requerimiento: Asignar insumos.....	102

3.11. Tabla: Especificación de requerimiento: Procesar solicitudes	102
3.12. Tabla: Especificación de requerimiento: Crear usuarios	103
3.13. Tabla: Especificación de requerimiento: Modificar usuarios	104
3.14. Tabla: Especificación de requerimiento: Extender solicitudes	104
3.15. Tabla: Especificación de requerimiento: Re abrir solicitudes	105
3.16. Tabla: Especificación de requerimiento: Ingresar nuevos insumos	106
3.17. Tabla: Especificación de requerimiento: Mantenedor de insumos	106
3.18. Tabla: Especificación de requerimiento: Reporte devolución de insumos	107
3.19. Tabla: Especificación de requerimiento: Reporte situación actual del inventario	108
3.20. Tabla: Especificación de requerimiento: Reporte de insumos activos	108
3.21. Tabla: Especificación de requerimiento: Auditoria por insumo	109
3.22. Tabla: Especificación de requerimiento: Consultar existencias	110
3.23. Tabla: Especificación de requerimiento: Reporte situación actual del inventario	110
3.24. Tabla: Especificación de requerimiento: Reporte de existencias	111
3.25. Tabla: Especificación de requerimiento: Crear perfiles	112
3.26. Tabla: Especificación de requerimiento: Consultar perfiles	112
3.27. Tabla: Especificación de requerimiento: Vínculos de navegación	113

CAPÍTULO 4. PROPUESTA

4.1. Tabla: Especificación de casos de uso: Autenticar usuario	124
4.2. Tabla: Especificación de casos de uso: Modificar clave	125
4.3. Tabla: Especificación de casos de uso: Verificar opciones por perfil.....	125
4.4. Tabla: Especificación de casos de uso: Crear usuario	126
4.5. Tabla: Especificación de casos de uso: Modificar usuarios	127
4.6. Tabla: Especificación de casos de uso: Crear perfiles.....	127
4.7. Tabla: Especificación de casos de uso: Consultar perfiles	128
4.8. Tabla: Especificación de casos de uso: Consultar existencias	128
4.9. Tabla: Especificación de casos de uso: Consultar insumos activos	128

4.10. Tabla: Especificación de casos de uso: Consultar transacciones.....	130
4.11. Tabla: Especificación de casos de uso: Devolución de insumos	132
4.12. Tabla: Especificación de casos de uso: Situación actual del inventario.....	132
4.13. Tabla: Especificación de casos de uso: Insumos asignados	132
4.14. Tabla: Especificación de casos de uso: Crear solicitudes	134
4.15. Tabla: Especificación de casos de uso: Consultar solicitudes	135
4.16. Tabla: Especificación de casos de uso: Procesar solicitudes	136
4.17. Tabla: Especificación de casos de uso: Extender solicitudes	136
4.18. Tabla: Especificación de casos de uso: Re abrir solicitudes.....	137
4.19. Tabla: Especificación de casos de uso: Solicitudes pendientes	138
4.20. Tabla: Especificación de casos de uso: Registrar existencias	139
4.21. Tabla: Especificación de casos de uso: Ingresar nuevos insumos simcards.....	140
4.22. Tabla: Especificación de casos de uso: Ingresar nuevos insumos terminales	140
4.23. Tabla: Especificación de casos de uso: Actualizar datos de insumos	141
4.24. Tabla: Especificación de casos de uso: Cargar insumos.....	142
4.25. Tabla: Especificación de casos de uso: Eliminar insumos.....	142
4.26. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_AUDIT_INSUMOS.....	175
4.27. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_REPORTES.....	176
4.28. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_INSUMOS.....	178
4.29. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_SOLICITUDES.....	179
4.30. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_USUARIOS.....	181
4.31. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_PERFILES	182
4.32. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_MENU_I.....	183
4.33. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_CONFIG	184
4.34. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_ERRORES.....	185
4.35. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_TMP	185
4.36. Tabla: Diccionario de datos: Tabla OTC_T_SGIP_TMP_SIM.....	186
4.37. Tabla: Código duro script de base de datos	187
4.38. Tabla: Código duro métodos y funciones .NET.....	194

CONTENIDO DE GRAFICOS

CAPÍTULO 2. MARCO TEORICO

2.1. Figura: Microsoft Solution Framework v4.0.....	25
2.2. Figura: Modelo de roles	26
2.3. Figura: Framework de .NET.....	27
2.4. Figura: Evolución de los sistemas de información a través del tiempo.....	30
2.5. Figura: Modelo Cliente / Servidor.....	34
2.6. Figura: Datos encriptados con WALLET.....	76

CAPÍTULO 3. ANALISIS

3.1. Figura: Proceso manual de gestión y control de inventarios pruebas.....	90
3.2. Figura: Propiedades estación de trabajo portátil.....	114
3.3. Figura: Propiedades estación de trabajo escritorio	114
3.4. Figura: Propiedades servidor virtual	115

CAPÍTULO 4. PROPUESTA

4.1. Figura: Inicio de sesión	119
4.2. Figura: Página Maestra.....	119
4.3. Figura: Formulario de reportes.....	120
4.4. Figura: Formulario de asignar insumos.....	120
4.5. Figura: Formulario de crear insumos	121
4.6. Figura: Formulario mantenedor de insumos	121
4.7. Figura: Formulario consulta de solicitudes.....	121
4.8. Figura: Formulario crear usuarios	121
4.9. Figura: Formulario modificar usuarios.....	122
4.10. Figura: Formulario crear solicitudes.....	122
4.11. Figura: Formulario para procesar solicitudes	123
4.12. Figura: Formulario perfiles	123
4.13. Figura: Casos de uso del sistema – Ingresar al aplicativo	124

4.14. Figura: Casos de uso del sistema – Gestionar usuarios	126
4.15. Figura: Casos de uso del sistema – Verificar stock actual.....	128
4.16. Figura: Casos de uso del sistema – Obtener transacciones por existencia.....	130
4.17. Figura: Casos de uso del sistema – Gestionar información.....	131
4.18. Figura: Casos de uso del sistema – Gestionar solicitudes.....	134
4.19. Figura: Casos de uso del sistema – Gestionar insumos	138
4.20. Figura: Casos de uso del sistema – Actualizar bodega	139
4.21. Figura: Diagrama navegacional – Inicio de sesión.....	143
4.22. Figura: Diagrama navegacional – Administración.....	143
4.23. Figura: Diagrama navegacional – Reportes.....	144
4.24. Figura: Diagrama navegacional – Servicios.....	144
4.25. Figura: Diagrama de clases – Ingresar al aplicativo	145
4.26. Figura: Diagrama de clases – Gestionar usuarios	146
4.27. Figura: Diagrama de clases – Actualizar bodega.....	147
4.28. Figura: Diagrama de clases – Verificar stock actual	147
4.29. Figura: Diagrama de clases – Obtener transacciones por existencia	148
4.30. Figura: Diagrama de clases – Gestionar información devoluciones	148
4.31. Figura: Diagrama de clases – Gestionar información insumos asignados.....	149
4.32. Figura: Diagrama de clases – Gestionar solicitudes.....	150
4.33. Figura: Diagrama de clases – Gestionar insumos	150
4.34. Figura: Diagrama de clases – Crear insumos	151
4.35. Figura: Diagrama de secuencia – Autenticar usuario	151
4.36. Figura: Diagrama de secuencia – Modificar clave	152
4.37. Figura: Diagrama de secuencia – Verificar opciones por perfil.....	152
4.38. Figura: Diagrama de secuencia – Crear usuario.....	153
4.39. Figura: Diagrama de secuencia – Modificar usuario.....	153
4.40. Figura: Diagrama de secuencia – Crear perfiles.....	154
4.41. Figura: Diagrama de secuencia – Consultar perfiles	154
4.42. Figura: Diagrama de secuencia – Consultar existencias	155
4.43. Figura: Diagrama de secuencia – Consultar insumos activos.....	155

4.44. Figura: Diagrama de secuencia – Consultar transacciones.....	156
4.45. Figura: Diagrama de secuencia – Devolución de insumos	156
4.46. Figura: Diagrama de secuencia – Situación actual del inventario.....	157
4.47. Figura: Diagrama de secuencia – Insumos asignados	157
4.48. Figura: Diagrama de secuencia – Crear solicitudes.....	158
4.49. Figura: Diagrama de secuencia – Consultar solicitudes	158
4.50. Figura: Diagrama de secuencia – Procesar solicitudes	159
4.51. Figura: Diagrama de secuencia – Extender solicitudes	159
4.52. Figura: Diagrama de secuencia – Registrar existencias	160
4.53. Figura: Diagrama de secuencia – Crear insumos	160
4.54. Figura: Diagrama de secuencia – Modificar Insumos	161
4.55. Figura: Diagrama de secuencia – Eliminar Insumos.....	161
4.56. Figura: Diagrama de actividades – Autenticar usuario.....	162
4.57. Figura: Diagrama de actividades – Modificar clave.....	162
4.58. Figura: Diagrama de actividades – Verificar opciones por perfil	163
4.59. Figura: Diagrama de actividades – Crear usuario.....	163
4.60. Figura: Diagrama de actividades – Modificar usuario	164
4.61. Figura: Diagrama de actividades – Crear y consultar perfiles.....	164
4.62. Figura: Diagrama de actividades – Consultar existencias	165
4.63. Figura: Diagrama de actividades – Consultar insumos activos.....	165
4.64. Figura: Diagrama de actividades – Consultar transacciones	166
4.65. Figura: Diagrama de actividades – Devolución de insumos	166
4.66. Figura: Diagrama de actividades – Situación actual del inventario	167
4.67. Figura: Diagrama de actividades – Insumos asignados.....	167
4.68. Figura: Diagrama de actividades – Crear solicitudes.....	168
4.69. Figura: Diagrama de actividades – Consultar solicitudes	168
4.70. Figura: Diagrama de actividades – Procesar solicitudes	169
4.71. Figura: Diagrama de actividades – Extender solicitudes	169
4.72. Figura: Diagrama de actividades – Re abrir solicitudes	170
4.73. Figura: Diagrama de actividades – Solicitudes pendientes.....	170

4.74. Figura: Diagrama de actividades – Registrar existencias	171
4.75. Figura: Diagrama de actividades – Ingresar nuevos insumos simcards	171
4.76. Figura: Diagrama de actividades – Ingresar nuevos insumos terminales	172
4.77. Figura: Diagrama de actividades – Actualizar datos de insumos.....	172
4.78. Figura: Diagrama de actividades – Cargar insumos	173
4.79. Figura: Diagrama de actividades – Eliminar insumos	173
4.80. Figura: Diagrama entidad relación	174

CAPÍTULO 5. PRUEBAS Y DOCUMENTACION

5.1. Figura: Proceso de Pruebas OTECEL. S.A	205
5.2. Figura: Gestión de defectos	207

RESUMEN

El análisis, diseño, implementación y documentación del “SISTEMA DE GESTION Y CONTROL DE RECURSOS FÍSICOS Y LÓGICOS PARA LA GERENCIA DE CONSTRUCCION AREA PRUEBAS DE OTECEL.S.A” fue realizado a través del marco de referencia Microsoft Solution Framework v4.0. Al implementar una solución WEB se utilizó la metodología orientada a objetos. Para la creación de los casos de uso se utilizó el lenguaje de modelado UML.

La plataforma de programación que se empleó para desarrollar y ejecutar el sistema fue “Microsoft Visual WEB Developer 2005 Express Edition”, utilizando una arquitectura Cliente-Servidor, para la publicación de las páginas ASPX se utilizó el Internet Information Server 6 y las transacciones de datos se realizan utilizando una base de datos corporativa ORACLE 8i.

CAPITULO 1. ANTECEDENTES

1.1. INTRODUCCIÓN

La empresa OTECEL S.A¹ es una empresa multinacional que ofrece a sus clientes los servicios de telefonía y entretenimiento móvil en Ecuador, dentro de los cuales podemos citar; planes de voz, mensajes de texto, paquetes de datos, banda ancha móvil, equipos celulares cuyas prestaciones son variados como sus precios.

Existen procesos establecidos y definidos por los cuales todos los productos y servicios son comercializados en tiempo y calidad según lo planificado por las áreas internas del negocio. Sin embargo existe información interna que indica la existencia de evidentes demoras en un proceso de la compañía, el cual se conoce como PRUEBAS TI.

El área de PRUEBAS TI es la encargada de certificar la calidad de los productos previos a su comercialización y posterior puesta en producción, se apoya con usuarios de las demás áreas del negocio de las cuales podemos citar las siguientes:

- SVC: Servicio al cliente, quién representa al usuario final o consumidor quien se encarga de evaluar los productos de forma funcional.
- Mercadeo: Son los encargados de realizar los estudios y proponer los nuevos negocios y servicios.
- Aseguramiento de ingresos: Son los responsables de asegurar que no existan pérdidas o fuga de ingresos, tienen establecidos controles y procesos para este efecto.
- Facturación: Son los encargados de asegurar que la factura final del cliente sea clara, oportuna y precisa.

¹ Razón social de la empresa TELEFONICA MOVILES ECUADOR.

- Soporte: Esta área del negocio se encarga del monitoreo de los sistemas productivos, el mantenimiento preventivo y correctivo de los aplicativos a su cargo a demás de la gestión de incidentes.
- Producción: Esta área se encarga de la coordinación de los cambios en producción, mantienen un control de las versiones de software y tienen relación con los proveedores del servicio.
- Logística: Esta área provee a toda la compañía de los insumos físicos como son: terminales celulares y simcards; los cuales sirven para la venta de los productos y servicios.
- Terminales: Esta área se encarga de la evaluación de los equipos celulares previo a su comercialización; realizan tareas de pruebas, homologación, actualización de software, etc.
- Redes: Es un área medular en la compañía, es la encargada que mantener disponible el servicio de voz y datos; realizan tareas de mantenimiento preventivo y correctivo, monitoreo y control de todas las estaciones y nodos a nivel nacional.
- PMO: Por sus siglas; Oficina de proyectos es la encargada de proveer los estándares para seguimiento y control de proyectos en la compañía, así también de auditar el correcto uso de la documentación para los fines pertinentes.
- Auditoria: Esta área se encarga de aplicar controles frecuentes a los procesos establecidos en la compañía, la misma que emite observaciones las mismas que sirven para aplicar correctivos y mejoras por parte de los auditados y también emiten inconformidades las cuales tienen un significado con mucho mas valor que tiene que ser corregido en tiempo y plazo definido por los auditados.

La realización del presente trabajo pretende apoyar al cierre de las inconformidades y observaciones levantadas por parte del área de auditoria al proceso de pruebas, además de contribuir a la productividad y mejora continua de los procedimientos internos.

1.2. PLANTEAMIENTO DEL PROBLEMA

El tema de estudio propuesto en la presente tesis surge de una necesidad relacionada con la productividad identificada en la Vicepresidencia de Tecnología / Gerencia de Construcción Área Pruebas de la empresa OTECEL.S.A.

Hay un proceso de asignación o préstamo de recursos físicos y lógicos como son; SIMCARDS y TELEFONOS CDMA / GSM a los colaboradores del área, y el registro de esta información se realiza de forma manual y haciendo uso de una hoja de cálculo Excel.

La cantidad de artículos que se maneja en esta área es de aproximadamente 6000 y la cantidad de colaboradores que utilizan estos recursos llegan a 30 personas.

La información que se maneja en las hojas de cálculo Excel, ocasionan inconvenientes relacionados con la organización de los recursos, pérdida de información, no se tiene información histórica de los movimientos generados ya que estos insumos se reutilizan periódicamente entre los colaboradores del área.

Otro problema además de la información es el control en este proceso manual el mismo que ha ocasionado perdidas o mal uso de los recursos.

No existen reportes que permitan realizar una gestión adecuada para los usuarios de la Gerencia de Construcción.

1.3. OBJETIVOS

1.3.1. OBJETIVO GENERAL

Realizar la puesta en producción de un aplicativo que apoye a la gestión y control de los recursos físicos y lógicos asignados a los colaboradores de la Gerencia de Construcción Área Pruebas en OTECEL.S.A

1.3.2. OBJETIVOS ESPECÍFICOS

- Definir y diseñar el esquema de base de datos ORACLE que cubra el alcance aprobado por los usuarios.
- Automatizar el proceso de asignación y control de recursos físicos y lógicos.
- Diseñar las interfaces del sistema, cuya funcionalidad comprenda los módulos de: Administración, Reportes y Servicios.
- Desarrollar el sistema de Gestión y Control de Inventarios Pruebas, cuya funcionalidad se ajuste a las especificaciones requeridas por los usuarios.
- Implementar el sistema de Gestión y Control de inventarios Pruebas y realizar los ajustes requeridos durante el tiempo de estabilización.

1.4. JUSTIFICACIÓN

El presente estudio favorecerá directamente a la empresa OTECEL.S.A, permitiendo disponer de una herramienta informática que apoye a la gestión y control de los recursos que se proveen al área de pruebas. Contribuyendo a identificar casos que puedan ser sometidos a estudio y evitar posibles fraudes o mal uso de los recursos.

Este proyecto de tesis sirve de material de apoyo y consulta para los estudiantes de la carrera de Ingeniería de Sistemas de la Universidad Politécnica Salesiana, al crear un sistema que sea puesto como productivo en una área específica de la empresa OTECEL.S.A, le permite al estudiante

entender los procesos de diseño, implementación y ejecución reales en el mercado de las telecomunicaciones.

1.5. ALCANCE

A continuación se especifica el alcance funcional requerido en base al procedimiento de gestión y control del inventario de pruebas.

- Administración
 - Creación de usuarios
 - Creación de perfiles
 - Consultar recursos
 - Mantenedor de insumos
 - Modificar usuarios
- Reportes
 - Auditoria por insumo
 - Existencia actual de insumos
 - Devoluciones por usuario
 - Insumos Asignados
 - Situación actual del inventario
 - Insumos activos
- Servicios
 - Asignar insumos
 - Consultar solicitudes
 - Crear solicitudes
 - Crear insumos
 - Procesar solicitudes

El sistema debe ser construido en ambiente WEB bajo la plataforma Microsoft Windows, con una arquitectura en capas. Su aspecto visual será realizado utilizando Visual Basic NET y el código fuente en ASPX para los métodos y funciones.

Se deben realizar las pruebas de calidad y funcionalidad del producto final ya que la implementación será para la intranet de OTECEL.S.A.

1.6. LIMITACIONES

Se han identificado las siguientes limitaciones en el presente estudio:

- **Diseño:** El diseño de la solución informática para cubrir las necesidades de los usuarios del proceso debe ajustarse a las normativas internas de desarrollo, los mismos que definen las interfaces de usuario y software base.
- **Software:** El software base a utilizar debe ser Microsoft Visual Net 2003 para el desarrollo del aplicativo, la base de datos Oracle versión 8i
- **Hardware:** El hardware provisto por la compañía tiene un estándar que se aplica a todos los usuarios, el mismo que comprende los siguientes puntos:
 - Sistema Operativo Windows XP 2002 SP2 - Ingles
 - Memoria RAM 1GB
 - Procesador: Intel Core Duo 2.5 GHZ
 - Espacio en Disco: 80 GB
- **Aplicativos:** Microsoft Office 2003, Sistema Comercial Latam (SCL), Citrix, Distribuidores, KenanFX launcher, Quick Test Profesional, Cliente Oracle 8i, McAfee, SisciX.
- **Procedimientos:** Para el diseño del aplicativo se debe referir al procedimiento llamado: "PRTISS0601 Gestión² y Control del Inventario de Pruebas"
 - Documentación: Para la entrega del software se debe realizar un documento llamado MAU (manual de usuario) el cual debe contener la información base requerida por la compañía.

² Procedimiento original de la compañía OTECEL. S.A proporcionado para la construcción del Sistema de Inventarios Pruebas para el Área de Pruebas – TI

CAPITULO 2. MARCO TEORICO

2.1. METODOLOGÍA DE DESARROLLO MSF V4.0

2.1.1. INTRODUCCION

Microsoft Solutions Framework por sus siglas, podemos definirlo como un marco para desarrollo de sistemas de software basado en principios, modelos, disciplinas, conceptos, prácticas y recomendaciones propias, derivadas de la experiencia de Microsoft. Se autodefine como “marco” y no como metodología, porque considera que no hay una única estructura de procesos válida para todos los proyectos.

El marco MSF se adapta de forma flexible a las características de cada proyecto. Podríamos decir que es la alternativa de Microsoft ante RUP (Rational Unified Process), con quien tiene algunas similitudes, pero también diferencias.

Con la aparición del producto Microsoft Visual Studio Team System (VS2005), se ha actualizado MSF a la versión 4.0, produciendo dos variantes: MSF for Agile Software Development para el trabajo en entornos que emplean metodologías ágiles, y MSF for CMMI Process Improvement para el trabajo en entornos con el modelo CMMI.

2.1.2. MICROSOFT SOLUTIONS FRAMEWORK

La visión de Microsoft Solutions Framework³ es proveer una guía de procesos que sea:

- Productiva.- Soportada por medio de checklists, guidelines, en lugar de contenido detallado.
- Integrada.- Gracias a VS2005, integración de tareas y herramientas.
- Extensible.- Procesos y guías son adaptables, permitiendo a sus usuarios tomar un enfoque formal o ágil.

³ Fuente: <http://www.microsoft.com/msf>

2.1.3. PRINCIPIOS FUNDACIONALES

- Fomentar la comunicación abierta y compartida
- Potenciar la capacidad y responsabilidad
- Focalizar en entregar valor de negocio
- Mantenerse ágil, esperar el cambio
- Invertir en calidad
- Aprender de las experiencias

2.1.4. MODELOS

2.1.4.1 MODELO DE PROCESOS

Provee una estructura para el desarrollo de aplicaciones que consiste en 4 etapas distintas, cada una de las cuales culmina con una meta definida. Las 4 etapas son:

- **Visión.**- El propósito de esta etapa es definir el objetivo del proyecto y las restricciones del mismo.

El principal entregable de esta etapa es el Documento de Visión, que contiene un análisis del problema del cliente, la descripción de los objetivos de la solución, un esquema de la conceptualización de la solución, perfil de los usuarios de la solución y objetivos del diseño.

- **Planeación.**- En esta etapa el equipo hace las especificaciones funcionales, un plan maestro para el proyecto y un calendario maestro. Las especificaciones funcionales describen lo que se va a desarrollar, incluyendo contenido como metas de diseño, requerimientos, características y dependencias.

Esta etapa culmina con la Aprobación del Plan, lo que representa que el desarrollo puede iniciar.

- **Desarrollo.**- En esta etapa el equipo se enfoca en desarrollar y probar la solución.

Esta etapa involucra una serie de entregas internas de componentes del desarrollo, hechas en paralelo y en segmentos, para medir el

avance del desarrollo y asegurar que los componentes interactúen entre sí.

Las pruebas de calidad, son parte integral de esta etapa, pues el rol de pruebas no implica únicamente encontrar errores, sino asegurar la calidad y que el producto resuelva las necesidades del cliente.

- **Estabilización.**- Esta etapa comienza con pruebas beta de la solución y termina cuando se libera el producto al cliente.

Las pruebas en esta etapa se enfocan en uso en condiciones reales. El equipo se concentra en corregir posibles fallos de la solución y preparar el producto para su liberación.

Cuando la solución se libera, ésta se transfiere al administrador de la misma.



Figura 2.1. Microsoft Solution Framework v4.0⁴

2.1.4.2 MODELO DE ROLES

- Enfoque a la estructuración de las personas y sus actividades para facilitar el éxito de un proyecto.

⁴ Fuente: www.naturastock.com/rsdotnet/iic3140/materia/iic3140_04.pdf

- Define bloques de roles, áreas funcionales, responsabilidades, y dirección para los miembros del equipo.
- Metas únicas para los roles dentro del ciclo de vida del proyecto.

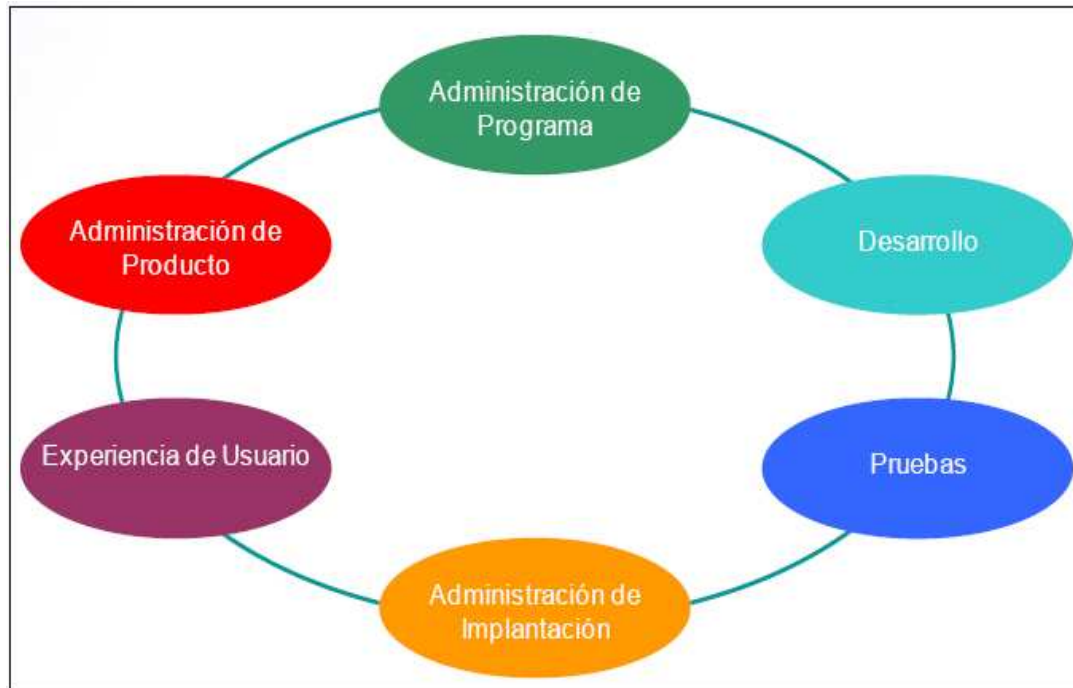


Figura 2.2. Modelo de roles⁵

Rol	Meta Clave de Calidad
Administración del Producto	Clientes Satisfechos
Administración del Programa	Entrega de la solución dentro de las restricciones del proyecto
Desarrollo	Construcción de acuerdo a la especificación
Pruebas	Aprobación para liberar la solución luego de certificar la calidad del producto
Experiencia del Usuario	Desempeño mejorado del usuario
Administración de Implantación	Gestión de puesta en marcha y operación continúa.

Tabla 2.1. Modelo de roles⁶

⁵ Fuente: www.naturastock.com/rsdotnet/iic3140/materia/iic3140_04.pdf

2.2. PLATAFORMA Y FRAMEWORK DE .NET

2.2.1. INTRODUCCION

.NET es una respuesta de Microsoft al creciente mercado de los negocios en entornos Web. Ofrece una manera rápida, económica, segura y robusta, de desarrollar aplicaciones.

2.2.2. FRAMEWORK DE .NET

La plataforma .NET es la propuesta de Microsoft para el desarrollo de aplicaciones completamente orientadas a objetos, seguras, sencillas de instalar y multiplataforma; no es un lenguaje de programación si no mas bien un conjunto de tecnologías de software que permite el desarrollo de aplicaciones de escritorio (WINDOWS), WEB (ASP.Net), bibliotecas de código (componentes reutilizables, Ej. DLLs), dispositivos móviles, controles de usuario, acceso a datos, reportes; que se ejecutan bajo el FRAMEWORK .Net siendo extensible al tener la posibilidad de heredar sus clases base y modificar el comportamiento o implementar sus interfaces para desarrollar clases nuevas.

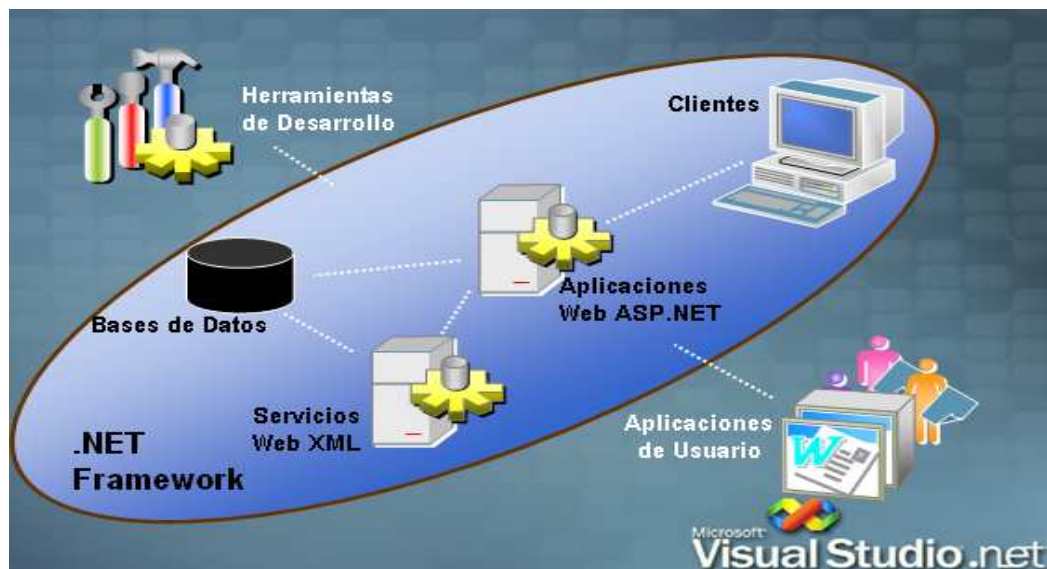


Figura 2.3. Framework de .NET⁷

⁶ Fuente: <http://www.microsoft.com/technet/itsolutions/msf/default.aspx>

⁷ Fuente: www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r46211.PPT

2.2.3. VERSIONES

El Framework de .Net ha evolucionado a través del tiempo desarrollando e incorporando mejoras en sus diferentes versiones, se describe un cuadro con el nombre de la versión, su código y fecha de lanzamiento oficial.

Nombre de la Versión	Número de Versión	Lanzamiento
Pre-beta	1.0.00000.00000	Julio 2000
1.0 Beta 1	1.0.?????.0	Noviembre 2000
1.0 Beta 2	1.0.2914.0	20/6/2001
1.0 RTM	1.0.3705.0	05-01-2002
1.0 SP1	1.0.3705.209	19-03-2002
1.0 SP2	1.0.3705.288	07-08-2002
1.0 SP3	1.0.3705.6018	31-08-2004
1.1 RTM	1.1.4322.573	01-04-2003
1.1 SP1	1.1.4322.2032	30-08-2004
1.1 SP1 (W2k3)	1.1.4322.2300	30-03-2005
2.0 RTM	2.0.50727.42	07-11-2005
2.0 RTM (Vista)	2.0.50727.312	30-01-2007
2.0 (KB928365)	2.0.50727.832	10-07-2007
2.0 SP1	2.0.50727.1433	19-11-2007
2.0 SP2	2.2.30729	¿?
3.0 RTM	3.0.4506.30	06-11-2006
3.0 RTM (Vista)	3.0.4506.26	30-01-2007
3.0 SP1	3.0.4506.648	19-11-2007
3.0 SP2	3.2.30729	¿?
3.5 RTM	3.5.21022.8	19-11-2007
3.5 SP1	3.5.30729.01	11-08-2008
4.0 RTM	4.0.30319.1	12-04-2010

Tabla 2.2. Listado de versiones Framework .NET⁸

2.3. APLICACIONES WEB CLIENTE – SERVIDOR

2.3.1. INTRODUCCION

El navegador es una especie de aplicación capaz de interpretar las órdenes recibidas en forma de código HTML y convertirlas en las páginas que son el resultado de dicha orden.

⁸ Fuente: http://es.wikipedia.org/wiki/Microsoft_.NET

Cuando un usuario pincha sobre un enlace de hipertexto, lo que pasa es que establece una petición de un archivo HTML residente en el servidor el cual es enviado e interpretado por el navegador del cliente.

Así es posible que desde el lado del servidor se pueda utilizar un lenguaje que es reconocido, ejecutado e interpretado por el propio servidor y que se envían al cliente en un formato comprensible para él.

2.3.2. ANTECEDENTES

El uso de las redes de computadoras motivó el uso de recursos lejanos entre sí mediante el uso de alguna interface de transmisión. Anteriormente primó la filosofía de Mainframes, máquinas las cuales contaban con terminales mediante acceso serial a un computador central.

En dichos terminales se realizaba ingreso y salida de información, que era procesada por la máquina.

Al aparecer Internet, se establecieron métodos de conexión en forma remota utilizando aplicaciones como telnet o rlogin⁹, dando igual acceso que el anterior, pero bajo un ambiente de trabajo TCP/IP.

Sin embargo al pasar el tiempo, estos servicios resultaban complicados y poco amigables para los usuarios con muy poco conocimiento de computación, con lo cual aparecen métodos de interfaces gráficas, que además de ser amigables, estaban orientadas a proveer información.

A esto, y además del aumento en la computación personal, y el afán de crear negocios en Internet, motivaron la idea del CLIENTE-SERVIDOR.

⁹ Referencia: http://docs.oracle.com/cd/E24842_01/html/E22524/remotehowtoaccess-79344.html

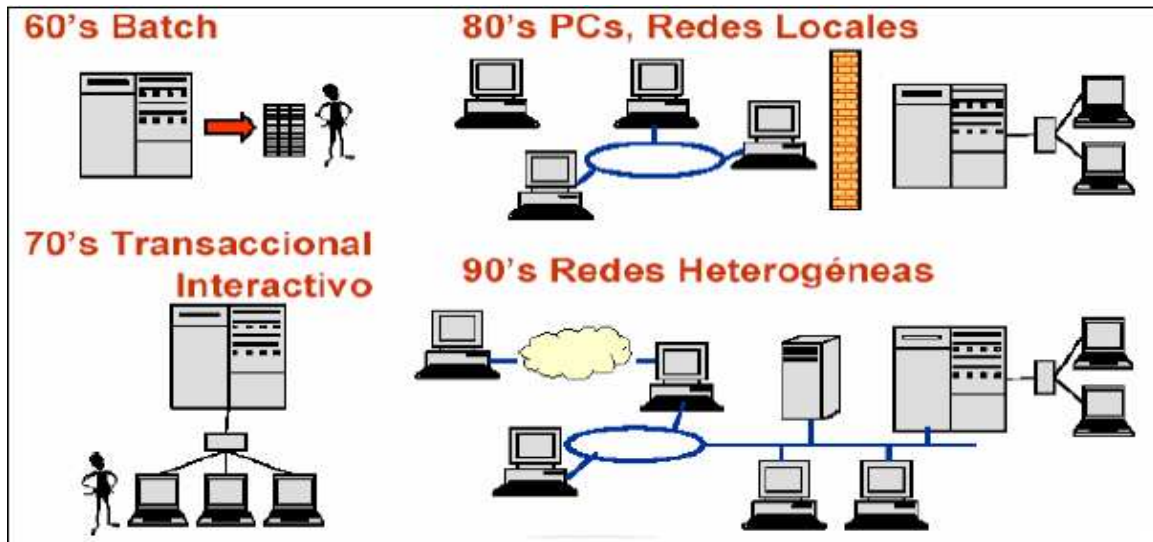


Figura 2.4. Evolución de los sistemas de información a través del tiempo¹⁰

2.3.3. CLIENTE

El cliente gestiona las peticiones del usuario y la recepción de las páginas que provienen del servidor.

Los lenguajes de lado cliente, entre los cuales no sólo se encuentra el HTML son Java, JavaScript, Visual Basic Script, Flash, CSS. Los cuales son simplemente incluidos en el código HTML

2.3.4. SERVIDOR

Es un ordenador que se encuentra continuamente conectado a la red y espera las peticiones del cliente.

En la aplicación del servidor hay páginas estáticas (documentos HTML), recursos multimedia (imágenes y documentos adicionales del sitio WEB), scripts o programas de servidor que al ser invocados se ejecutan y dan como resultado una página HTML generada (pueden acceder a recursos externos como una base de datos)

¹⁰ Fuente: <http://artemisa.unicauca.edu.co/~ecaldon/docs/apliweb/Elec-Ses2-2k5-ClienteServidor.pdf>

2.3.5. CLIENTE – SERVIDOR

El Cliente y el Servidor pueden actuar como una sola entidad o como entidades separadas, realizando actividades o tareas independientes, sin embargo un servidor da servicio a múltiples clientes en forma concurrente.

Las funciones de Cliente y Servidor pueden estar en plataformas separadas, o en la misma plataforma. Cada plataforma puede ser escalable independientemente.

Los cambios realizados en las plataformas de los Clientes o de los Servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.

La interrelación entre el hardware y el software están basados en una infraestructura poderosa, de tal forma que el acceso a los recursos de la red no muestra la complejidad de los diferentes tipos de formatos de datos y de los protocolos.

2.3.5.1 TIPOS DE SERVIDORES

- **Servidores de archivos.**- Servidor donde se almacena archivos y aplicaciones de productividad como por ejemplo procesadores de texto, hojas de cálculo, etc.
- **Servidores de bases de datos.**- Servidor donde se almacenan las bases de datos, tablas, índices. Es uno de los servidores que más carga tiene.
- **Servidores de transacciones.**- Servidor que cumple o procesa todas las transacciones. Valida primero y recién genera un pedido al servidor de bases de datos.
- **Servidores de Groupware.**- Servidor utilizado para el seguimiento de operaciones dentro de la red.
Servidores de objetos, Contienen objetos que deben estar fuera del servidor de base de datos. Estos objetos pueden ser videos, imágenes, objetos multimedia en general.

- **Servidores Web.-** Se usan como una forma inteligente para comunicación entre empresas a través de Internet. Este servidor permite transacciones con el acondicionamiento de un browser específico.

2.3.5.2 ESTILOS DEL MODELO CLIENTE SERVIDOR

- **Presentación distribuida.-**

Se distribuye la interfaz entre el cliente y la plataforma servidora. La aplicación y los datos están ambos en el servidor. Similar a la arquitectura tradicional de un Host y Terminales. El computador se aprovecha solo para mejorar la interfaz gráfica del usuario.

Ventajas:

- Revitaliza los sistemas antiguos.
- Bajo costo de desarrollo.
- No hay cambios en los sistemas existentes.

Desventajas:

- El sistema sigue en el Host.
- No se aprovecha la GUI y/o LAN.
- La interfaz del usuario se mantiene en muchas plataformas.

- **Presentación Remota.-**

La interfaz para el usuario esta completamente en el cliente. La aplicación y los datos están en el servidor.

Ventajas:

- La interfaz del usuario aprovecha bien la GUI y la LAN.
- La aplicación aprovecha el Host.
- Adecuado para algunos tipos de aplicaciones de apoyo a la toma de decisiones.

Desventajas:

- Las aplicaciones pueden ser complejas de desarrollar.

- Los programas de la aplicación siguen en el Host.
- El alto volumen de tráfico en la red puede hacer difícil la operación de aplicaciones muy pesadas.

- **Lógica Distribuida.-**

La interfaz esta en el cliente, la base de datos esta en el servidor.

La lógica de la aplicación esta distribuida entre el cliente y el servidor.

Ventajas:

- Arquitectura más corriente que puede manejar todo tipo de aplicaciones.
- Los programas del sistema pueden distribuirse al nodo mas apropiado.
- Pueden utilizarse con sistemas existentes.

Desventajas:

- Es difícil de diseñar.
- Difícil prueba y mantenimiento si los programas del cliente y el servidor están hechos en distintos lenguajes de programación.
- No son manejados por la GUI 4GL.

- **Administración de datos remota.-**

En el cliente residen tanto la interfaz como los procesos de la aplicación.

Las bases de datos están en el servidor, es lo que comúnmente imaginamos como aplicación cliente servidor.

Ventajas:

- Configuración típica de la herramienta GUI 4GL.
- Muy adecuada para las aplicaciones de apoyo a las decisiones del usuario final.
- Fácil de desarrollar ya que los programas de aplicación no están distribuidos.
- Se descargan los programas del Host.

Desventajas:

- No maneja aplicaciones pesadas eficientemente.

- La totalidad de los datos viaja por la red, ya que no hay procesamiento que realice el Host.

- **Base de datos distribuida.-**

La interfaz, los procesos de la aplicación, y parte de los datos de la base de datos están en cliente. El resto de los datos están en el servidor.

Ventajas:

- Configuración soportada por herramientas GUI 4GL.
- Adecuada para las aplicaciones de apoyo al usuario final.
- Apoya acceso a datos almacenados en ambientes heterogéneos.
- Ubicación de los datos es transparente para la aplicación.

Desventajas:

- No maneja aplicaciones grandes eficientemente.
- El acceso a la base de datos distribuida es dependiente del proveedor del software administrador de bases de datos.

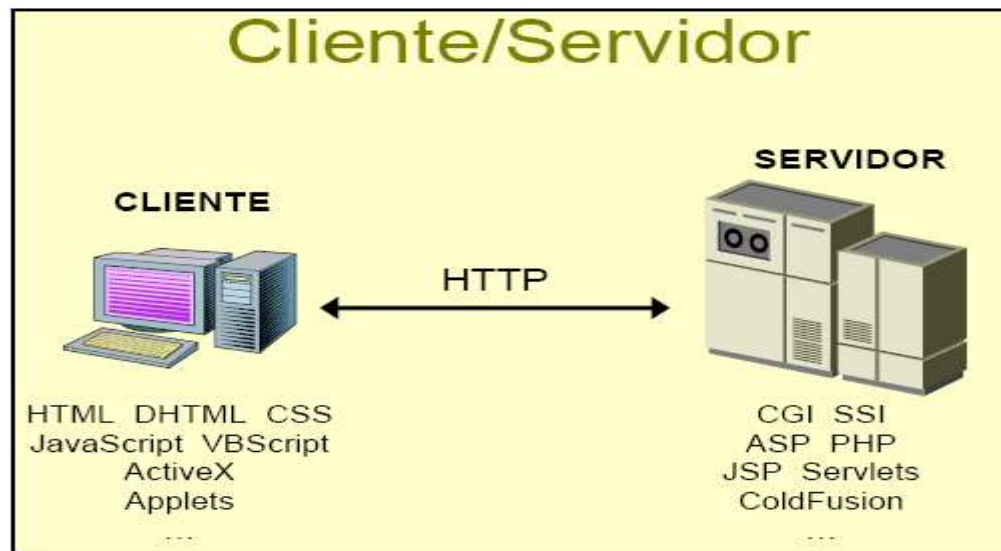


Figura 2.5. Modelo Cliente / Servidor¹¹

¹¹ Fuente: <http://rua.ua.es/dspace/bitstream/10045/4412/5/03c-AplicacionesWeb.pdf>

2.4. METODOLOGÍA DE PROGRAMACIÓN ORIENTADA A OBJETOS

2.4.1. INTRODUCCION

Visual Basic es un descendiente de Basic, que ha existido durante varias décadas. BASIC (el acrónimo de Beginners All-Purpose Symbolic Instruction Code)¹² fue originariamente desarrollado por la Dartmouth University en 1964 como lenguaje para programadores principiantes.

BASIC se convirtió en el primer lenguaje que la mayoría de los programadores aprendía para familiarizarse con los fundamentos de la programación antes de pasar a lenguajes más potentes.

Visual Basic apareció en mayo de 1991 y supuso una revolución en la metodología de desarrollo de aplicaciones, ya que permitía la creación de programas arrastrando y soltando componentes (objetos) en lugar de tener que codificar manualmente los elementos. Desde muchos puntos de vista Visual Basic .NET supone la madurez de ese proyecto que nació hace más de diez años.

La nueva versión, Visual Basic .NET, incorpora características de orientación a objetos de tal manera que podría considerarse un lenguaje totalmente diferente a sus predecesores y totalmente orientado a objetos, ya que en versiones anteriores Visual Basic era considerado un lenguaje de programación con sintaxis de objetos, pero no un verdadero lenguaje orientado a objetos.

2.4.2. AMBITOS CON NOMBRE

En una aplicación participan un buen número de componentes mediante los cuales se facilita el acceso a bases de datos, la conectividad en redes, comunicación con otras aplicaciones, etc. La existencia de tantos componentes puede causar conflictos si hay coincidencias de nomenclatura, y por ello, la plataforma .NET ha optado por el uso de espacios o ámbitos de nombre.

¹² <http://www.microsoft.com/express/Downloads/#2008-Visual-Basic>

Un ámbito con nombre, del inglés namespace¹³, es un ámbito delimitado explícitamente al que se ha asignado un identificador. En su interior es posible incluir tanto definiciones de tipos como otros ámbitos con nombre, creando una anidación que dará lugar a una jerarquía.

Para crear un ámbito con nombre, incluyendo en él las definiciones que nos interesen, se utiliza la palabra clave Namespace. Hay que tener en cuenta que todas las definiciones incluidas en su definición sólo existen dentro del espacio con nombres. Por ejemplo:

```
Namespace Espacio1
  Namespace Espacio2
    Public Class HolaMundo
      Public Shared Sub Main()
        System.Console.WriteLine("Hola desde Espacio1.Espacio2")
      End Sub
    End Class
  End Namespace
End Namespace
```

```
Namespace Espacio1.Espacio2
  Public Class HolaMundo
    Public Shared Sub Main()
      System.Console.WriteLine("Hola desde Espacio1.Espacio2")
    End Sub
  End Class
End Namespace
End Namespace
```

En ambos casos tenemos un ámbito con nombre, llamado Espacio1, que contiene en su interior otro, denominado Espacio2.

Al haberse definido la clase HolaMundo como pública, es posible crear un objeto de dicha clase desde otro programa, siempre y cuando se componga una referencia completa que permitiese al compilador identificarla de manera unívoca. Esto significaría anteponer al nombre de la clase el del ámbito o

¹³ <http://msdn.microsoft.com/vbasic/learning/windowsforms>

ámbitos con nombre donde se encuentra definida, como se hace en este ejemplo:

```
Class UsaHolaMundo
Shared Sub Main()
Dim MiHolaMundo As Espacio1.Espacio2.HolaMundo = New
Espacio1.Espacio2.HolaMundo()
End Sub
End Class
```

Una alternativa a esta composición de referencias cualificadas consiste en utilizar la sentencia Imports, seguida del ámbito cuya referencia no deseamos repetir continuamente. El resultado sería, como se puede ver a continuación, un código más legible.

```
Imports Espacio1.Espacio2
Class UsaHolaMundo
Shared Sub Main()
Dim MiHolaMundo As HolaMundo = New HolaMundo()
End Sub
End Class
```

2.4.3. CLASES

En la programación orientada a objetos (POO), utilizamos los objetos para encapsular la información asociada a las entidades con las que trabaja el programa.

Una clase¹⁴ es un molde que define los atributos y comportamientos de los objetos que se crean como instancias de esta clase. En Visual Basic .NET, la definición de clase es muy similar a la definición de la misma en cualquier lenguaje orientado a objetos. Esta versión .NET incluye como novedad que no es necesario crear un módulo independiente para cada clase.

Una clase puede ser base de otra, así como estar derivada de otra. En ella, puede definirse su comportamiento -métodos-, sus atributos -propiedades-, campos de datos y eventos, e incluso anidar unas clases dentro de otras.

¹⁴ Referencia On-Line [Programación Orientada a Objetos.pdf – Pag2 – Clases.]

2.4.4. CAMPOS DE DATOS

Los campos de datos son variables con un ámbito de visibilidad reducido a la clase. El acceso a dichas variables, por parte del usuario de la clase, puede ser directo o bien a través de métodos o propiedades, todo dependiendo de la visibilidad del campo. En su declaración, podemos controlar su visibilidad mediante unos modificadores, que no sólo pueden aplicarse a la definición de variables, sino también de clases, métodos, estructuras.

Destacan al mismo tiempo otros modificadores:

- Protected: Son accesibles internamente, y también por aquellas clases que hayan derivado de ésta.
- Friend: Permite el acceso desde cualquier punto del proyecto al que pertenece la clase.
- Protected Friend: Los miembros son visibles en el proyecto actual y en las clases derivadas, aunque estén en otro proyecto.

```
Public class ClaseBase
    Dim AmbitoDefecto As Integer
    Private AmbitoPrivado As Integer
    Protected AmbitoProtegido As Integer
    Friend AmbitoInterno As Integer
    Protected Friend AmbitoProtegidoInterno As Integer
    Public AmbitoPublico As Integer
End class
```

Desde cualquier método que pertenezca a la clase ClaseBase se puede acceder a todas las variables independientemente de su modificador. Si tuviésemos una clase derivada, la cual heredaría todas las variables, no podría acceder a las dos primeras por tener un ámbito privado.

Si tuviésemos una clase totalmente ajena a ClaseBase, sólo podríamos acceder a las tres últimas variables.

2.4.4.1 PROPIEDADES

Las clases almacenan información que, en muchas ocasiones, no sólo es para uso interno sino que también puede interesar al programador que las utilice.

Si, por ejemplo, se desea que el valor de una cierta variable pueda ser leído externamente pero no modificado, o bien que cada vez que se modifique su valor éste pueda ser controlado para saber si es válido, es adecuado definir una propiedad.

Una propiedad es una variable que tiene vinculado internamente unos métodos de acceso que son los que controlan la lectura o asignación de valores.

Dentro del bloque de código de propiedades existe un bloque Get, que devuelve al usuario de la clase el valor de la instancia privada de la variable; y un bloque Set. Al mismo tiempo, es posible restringir el acceso a las propiedades utilizando el “comando ReadOnly”¹⁵, y eliminando el bloque Set, para convertirla en una propiedad de sólo lectura; o utilizar el comando WriteOnly y eliminar el bloque Get, para convertir la propiedad en una propiedad de sólo escritura.

Por ejemplo:

```

Class Ficha
    Private Pnombre As String
    Private Pdireccion As String
    Private Pdepartamento As String

    //La propiedad Nombre sólo puede leerse, al igual que sucede //con
    Dirección

    Public ReadOnly Property Nombre() As String
    Get
        Return Pnombre
    End Get
    End Property

    //La propiedad Departamento puede leerse y escribirse, al igual //que
    sucede con Idempleado.

    Public Property Departamento() As String
    Get
        Return Pdepartamento
    End Get
    Set (ByVal Value As String)
        Pdepartamento = Value
    End Set
    End Property
End Class

```

¹⁵ <http://www.canalvisualbasic.net/manual-net/vb-net/>

2.4.4.2 METODOS

Los métodos definen el comportamiento de las clases. Son bloques de código delimitados, a los cuales se les asigna un identificador para poder efectuar una o más llamadas y ejecutar el código las veces necesarias.

En un método hay dos posibilidades, que devuelva algún dato, o que por el contrario no devuelva nada. Si del primer caso se trata se antepone la palabra Function, para el segundo caso se usa la palabra Sub.

Los parámetros que recibe el método pueden estar pasados por valor o por referencia:

- Valor: El método recibe una copia del valor almacenado en la variable original.
El paso por valor se usa por defecto, pero si queremos resaltarlo se antepone la palabra ByVal.
- Referencia: El método recibe una referencia a esa variable, lo cual permite modificarla. Para indicar que el paso es por referencia se antepone la palabra ByRef.

Ejemplo:

```
Public Sub asignacion ( ByVal Pnombre As String, ByVal Pdireccion As String)
    If Pnombre = "David" And Pdireccion = "Santa Marta" Then
        PEmpleado = 1
        Departamento = "IS"
    ElseIf Pnombre = "Jesus" And Pdireccion = "Salamanca" Then
        PEmpleado = 2
        Departamento = "AC"
    Else
        Throw New Exception ("Acceso incorrecto.")
    End If
End Sub
```

En cuanto a la devolución de parámetros desde un método, lo único que hay que hacer es sustituir la palabra Sub, que hemos puesto delante del nombre del método, por Function y añadir al final la palabra As seguida del tipo que corresponda. En el cuerpo del método usaríamos la sentencia Return acompañada del valor a devolver.

2.4.4.3 EVENTOS

Se entiende por evento una señal generada por un objeto o componente. Esta señal puede conectarse a un gestor de evento, un método que se ejecutaría automáticamente al recibir esa señal.

Los eventos proporcionan una forma de que un objeto ejecute el código escrito por el usuario del objeto. El objeto desencadena el evento mediante un controlador de eventos.

Para declarar un evento se utiliza la palabra clave Event, que indica a la clase el nombre del evento: `Public Event SalarioError (ByVal Error As String)`

Una vez se le ha indicado a la clase la posibilidad de producirse un evento, es necesario introducir el código para producir el evento:

```

Public Property Salario( ) As Decimal
    Get
        Return Psalario
    End Get
    Set (ByVal Valor As Decimal)
        If Valor >= 0 Then
            Psalario = Valor
        Else
            RaiseEvent SalarioError ("Salario no puede ser negativo.")
        End If
    End Set
End Property

```

2.4.4.4 CONSTRUCTORES Y DESTRUCTORES

Las clases definidas en Visual Basic .NET pueden contar con unos métodos específicos para controlar la construcción y destrucción de objetos. Una misma clase puede contar con múltiples constructores, siempre que cuenten con diferentes listas de parámetros; mientras que destructor sólo puede haber uno por clase.

Los constructores tienen por nombre `New()`, no cuentan con valor de retorno, pueden tomar o no parámetros según interese, y pueden inicializar las propiedades del objeto instancia o establecer una conexión con una base de datos.

El destructor tiene por nombre `Finalize()`, no cuenta con valor de retorno y no pueden tomar parámetro alguno.

Al crear un objeto de una clase, utilizando para ello el operador `New`, se ejecuta automáticamente el constructor que corresponda según los parámetros facilitados. Sin embargo, el destructor, se ejecuta automáticamente para un cierto objeto cuando ya no hay ninguna referencia hacia él, es decir, no necesitamos destruir los objetos explícitamente, de esto se encarga el sistema de recogida de basura de la plataforma .NET. La recogida de objetos no útiles se efectúa en momentos concretos en los que no hay ninguna otra actividad pendiente.

Ejemplo:

```

Class Agenda
Public Sub New() //Constructor sin parámetros
End Sub
Public Sub New (ByVal Parametro As String) //y con un parámetro
End Sub
Protected Sub Finalize() //Destructor
End Sub
End Class
//Creación de un objeto sin entregar ningún parámetro
Dim MiAgenda As Agenda = New Agenda()
//Creación de un objeto facilitando una cadena
Dim Miagenda As Agenda = New Agenda("Parámetro")

```

2.4.4.5 CLASES ANIDADAS

Consiste básicamente en definir otras clases en el interior de una clase, esto es útil cuando una cierta clase tan solo va a ser necesaria en el interior de otra, nunca fuera.

Al anidar una clase en otra es también una forma de ocultarla, ya que si no se la hace pública, sería imposible usarla desde el exterior aun poniendo una referencia completa.

Conseguimos la encapsulación de una clase dentro de otra.

Ejemplo:

```

Class Agenda
  Class Anotación
    Shared Sub New()
      Console.WriteLine("Constructor de la clase anotación")
    End Sub
    Public Sub New()
      Console.WriteLine("Constructor de anotación")
    End Sub
    Public Sub New(ByVal Parametro As String)
      Console.WriteLine("Constructor {0}",parámetro)
    End Sub
    Protected Sub Finalize()
      Console.WriteLine("Destructor de anotación")
    End Sub
  End class
  //Las siguientes declaraciones pertenecen a la clase agenda
  Private MiAnotacion As Anotacion
  Private Sub New()
    Console.WriteLine("Constructor agenda")
    Mianotacion = New Anotación ("Nueva anotación")
  End Sub
  Protected Sub Finalize()
    Console.WriteLine("Destructor agenda")
    Mianotacion = Nothing
  End Sub
End Class

//Creamos una que contenga el Main()
Class Principal
  Shared Sub Main()
    Dim MiAgenda As Agenda = New Agenda()
    Console.WriteLine("Objeto creado")
    MiAgenda = Nothing
    Console.WriteLine("Fin de la aplicación")
  End Sub
End Class

```

El resultado que se obtendría por pantalla sería:

```

Constructor de Agenda
Constructor de la Clase Anotación
Constructor de Nueva Anotación
Objeto creado
Fin de la aplicación
Destructor de Anotación

```

2.4.5. JERARQUIA DE CLASES

La herencia es una de las funciones más potentes y fundamentales de cualquier lenguaje de programación orientado a objetos, ya que es posible crear una clase base que encapsule las propiedades y métodos que serán necesarios en múltiples clases derivadas del mismo tipo.

Otra de estas características fundamentales es el polimorfismo, gracias al cual se pueden definir en una clase base métodos que serán implementados por las clases que se deriven de ella.

2.4.5.1 HERENCIA

Gracias a la herencia¹⁶ podemos crear clases base en las que se encapsulen las funciones más comunes. Posteriormente, podemos crear otras clases que se deriven de las clases base.

Las clases derivadas heredan las propiedades y métodos de las clases base y pueden ampliar o complementar sus funciones para adaptarse a los requerimientos del programa.

Para crear una clase derivada en Visual Basic .NET se incluye en su definición el comando Inherits junto con el nombre de la clase base.

Por ejemplo:

```

Public Class Cuenta
    Public PnumeroCuenta As Long
    Public Property NumeroCuenta() As Long
    Get
        Return PnumeroCuenta
    End Get
    Set (ByVal Value As Long)
        PnumeroCuenta = Value
    End Set
End Property
Public Function HacerBalance () As Double
    //Código para obtener el balance de la cuenta.
End Function
End Class

Public Class CuentaCorriente
    Inherits Cuenta

```

¹⁶ Referencia On-Line [Programación Orientada a Objetos.pdf – Pag28– Herencia]

```

        Private MinBalance As Double
        Public Sub Retirar (ByVal Cantidad As Double)
            //Código para retirar dinero de la cuenta.
        End Sub
    End Class

    Dim Ahorros As CuentaCorriente = New CuentaCorriente ()
    Ahorros.NumeroCuenta = 2000

    //Método definido por la clase Cuenta heredado Por la clase //CuentaCorriente.
    Ahorros.HacerBalance = ()

    //Método definido por la clase CuentaCorriente.
    Ahorros.Retirar (500)

```

En ciertas ocasiones, interesa que no se tenga autorización para crear instancias de una clase base, forzando que el acceso a los métodos y propiedades de la clase se realice a través de una clase derivada. En este caso, construiríamos la clase base utilizando el modificador `MustInherit`.

En el siguiente código se muestra la definición de la clase `Cuenta` con el modificador `MustInherit`:

```
Public MustInherit Class Cuenta
```

Esta definición convierte a `Cuenta` en una clase abstracta, ya que define las interfaces de los métodos y propiedades que serán heredadas por las clases derivadas. Así, para poder acceder al método `HacerBalance` será necesario crear una instancia de la clase derivada `CuentaCorriente`.

Por defecto, todas las clases pueden tener herencia. Si no se tiene cuidado, es posible que generemos cadenas de herencia muy complicadas que resultan difíciles de administrar y depurar. Utilizando el modificador `NotInheritable`, es posible crear clases con la total certeza de que no darán lugar a clases derivadas. Este tipo de clases suele denominarse clase sellada o final.

Ejemplo:

```
Public NotInheritable Class CuentaCorriente
```


2.4.5.2 POLIMORFISMO

El polimorfismo¹⁷ es la habilidad que tienen objetos basados en diferentes clases para responder a la misma llamada de método utilizando sus propias implementaciones. Los métodos heredados por las clases derivadas pueden someterse a sobrecarga, para ello utilizamos la palabra clave OverLoads.

La signatura de método de la clase sobrecargada debe utilizar el mismo nombre que el método sobrecargado, pero la lista de parámetros debe ser diferente. Es exactamente lo mismo que sobrecargar métodos de la misma clase, excepto la palabra clave Overloads es opcional y, normalmente, se omite.

Ejemplo:

```

Public Class Cuenta
    Public Sub Retirar (ByVal Cantidad As Double)
        //Código de implementación
    End Sub
End Class
Public Class CuentaCorriente
    Inherits Cuenta
    Public Overloads Sub Retirar (ByVal Cantidad As Double,
        ByVal MinnumBalance As Double)
        //Código de implementación
    End Sub
End Class

```

En el siguiente ejemplo, se realizará una iteración a través de la colección de clases de tipo cuenta, y el compilador determinará en tiempo de ejecución qué implementación concreta de tipo cuenta debe ejecutar. De este modo, no hay que preocuparse de a qué tipo de clase estamos haciendo referencia, pues los tipos de clase implementan las mismas interfaces de método. Así, nos interesa que todas las clases de cuentas contengan un método ObtenerInfoCuenta con la misma definición de interfaz, pero con diferentes implementaciones que dependerán del tipo de cuenta.

¹⁷ Referencia On-Line [Programación Orientada a Objetos.pdf – Pag28 – Polimorfismo.]

Para poder reemplazar un método heredado en la clase derivada, usamos la palabra clave `Overrides` en la definición del mismo.

```
Public MustInherit Class Cuenta
    Public MustOverride Function ObtenerInfoCuenta () As String
End Class
```

```
Public Class CuentaCorriente Inherits Cuenta
    Public Overrides Function ObtenerInfoCuenta () As String
    Return "Imprimiendo información de cuenta corriente"
End Function
End Class
```

```
Public Class CuentaAhorro Inherits Cuenta
    Public Overrides Function ObtenerInfoCuenta () As String
    Return "Imprimiendo información de cuenta de ahorros"
End Function
End Class
```

También se puede obtener un resultado similar utilizando una interfaz, que definirá las firmas de los métodos. En lugar de heredar de una clase base `Cuenta`, definiremos una interfaz `ICuenta`, y las clases que la implementen deben aportar el código de implementación de todos los métodos definidos por ella:

```
Public Interface ICuenta
    Function ObtenerInfoCuenta () As String
End Interface
```

```
Public Class CuentaCorriente Implements ICuenta
    Public Function ObtenerInfoCuenta () As String Implements
    ICuenta.ObtenerInfoCuenta
    Return "Imprimiendo información de cuenta corriente"
End Function
End Class
```

```
Public Class CuentaAhorro Implements ICuenta
    Public Function ObtenerInfoCuenta () As String Implements
    ICuenta.ObtenerInfoCuenta
    Return "Imprimiendo información de cuenta de ahorros"
End Function
End Class
```

2.5. SERVIDOR WEB PARA EL SISTEMA OPERATIVO MICROSOFT WINDOWS (IIS).

2.5.1. INTRODUCCION

Al hablar de “servidor” puede ser que se aluda al servidor mismo (hardware) o al paquete de software que el servidor contiene para sus funciones (software). En otra palabras, puede hacerse referencia a el como hardware si se lo entiende como equipo especializado en las tareas de un servidor o como software si se enfoca en las posibles funciones de control sobre las tareas que el servidor realiza.

Un servidor Web¹⁸ puede definirse como un programa que escucha las peticiones de los usuarios o navegantes y las atiende o satisface. Por medio de la especificación de la búsqueda, el servidor Web buscará una página específica o ejecutará un programa, pero, necesariamente, enviara algún resultado sobre la búsqueda recibida. Los sistemas operativos más utilizados por los servidores son Windows y Linux, siendo este ultimo más estable y por lo tanto de uso más frecuente.

2.5.2. ARQUITECTURA

2.5.2.1 PETICION GET

Un servidor WEB opera mediante el protocolo HTTP, de la capa de aplicación del Modelo OSI. Al protocolo HTTP se le asigna habitualmente el puerto TCP 80. Las peticiones al servidor suelen realizarse mediante HTTP utilizando el método de petición GET en el que el recurso se solicita a través de la URL al servidor WEB.

GET¹⁹ /index.html HTTP/1.1 HOST: www.host.com

¹⁸ Fuente: http://es.wikipedia.org/wiki/Servidor_web#Aplicaci.C3.B3n_del_lado_del_Servidor

¹⁹ Fuente: http://es.wikipedia.org/wiki/Servidor_web#Arquitectura

En la barra de URL de un navegador cualquiera la petición anterior sería análoga a la siguiente dirección Web: www.host.com/index.html

2.5.2.2 ESQUEMA DE UNA PETICION GET

Petición WEB.- El navegador por medio de la interfaz de usuario permite al usuario realizar una o varias peticiones WEB. La interfaz de usuario o entorno de usuario es el conjunto de elementos del navegador que permiten realizar la petición de forma activa. Una petición Web no sólo puede ser realizada mediante un navegador sino con cualquier herramienta habilitada para tal fin, como una consola de comandos.

Elementos del entorno de usuario más comunes en navegadores Web visuales:

- **Hipervínculo enlace o link.-** Es una porción de contenido Web, texto, imagen y otros elementos, que enlaza con una dirección Web. Al pulsar un hipervínculo el navegador genera una petición GET automática a la dirección URL de dicho link.
- **Formulario WEB.-** Al realizar el envío satisfactorio de los datos de un formulario, el navegador Web genera una petición GET o POST (comúnmente POST) automática a la par que envía los datos al servidor.
- **Barra de direcciones.-** Todos los navegadores incluyen una barra de direcciones mediante la cual puede accederse manualmente a cualquier dirección URL, de modo que el navegador generará una petición GET automática a dicha URL cada vez que el usuario lo desee.
- **Script activo o pasivo.-** Cualquier aplicación JavaScript tiene acceso al estado del navegador, cómo puede modificar los datos que describen tal estado, de forma pasiva (sin medio de la intervención del usuario) o de forma activa (mediante alguna acción del usuario).

Socket a dirección DNS.- Se produce una socket con un servidor dado en dirección IP mediante TCP. Por lo general las direcciones que el navegador

posee inicialmente son direcciones DNS (direcciones alfanuméricas) que deberá convertir a direcciones numéricas.

Resolución de DNS a IP.- Si la dirección dada es DNS y no existe una regla en la base de datos DNS, el [Host Resolver Request] solicita al servidor DNS la o las direcciones IPs correspondientes. El navegador crea una nueva regla y almacena la dirección IP junto a la dirección DNS en su base de datos de reglas.

Recuperación de la regla DNS.- Una vez almacenada la regla se realiza una petición a la base de datos DNS para recuperar los valores de la regla.

Socket a dirección IP.- Se produce una socket con la dirección IP mediante TCP. La dirección IP puede haberse recuperado en el paso anterior. [SOCKET 192.168.0.1]

Preparación de la petición.- Se crea la petición GET estableciendo la URL, un flag, la prioridad de la petición y el método (implícitamente GET).

Apertura Caché.- Se abre y/o se crea una entrada en el HTTP caché

Efectuación de la petición.- Se realiza la petición GET. Se leen las cabeceras HTTP de la [http transaction] y más tarde el cuerpo de la [http transaction] [GET /index.html HTTP/1.1]

Consulta en Caché.- Se consulta en el caché de disco si existe una entrada en el caché asociada al recurso que se ha solicitado. Los valores son created (true o false) y key (la URL del recurso).

Retribución booleana existencialista del recurso solicitado.- Si la entrada no existe (si el valor de created es false) se escriben los datos en el caché de disco. Si no, se lee directamente.

Presentación visual del recurso.- Se concluye la operación y se muestra en pantalla (si es preciso) la información.

2.5.2.3 PETICION GET PASIVA

JavaScript permite realizar modificaciones en el estado del navegador el cual viene definido por el array de objetos location del objeto global Window.

Se referencia a tal objeto con `window.location`. En concreto `window.location.href` contiene la dirección actual del navegador Web²⁰.

Si una parte del script ejecuta tal sentencia:

```
window.location.href='http://wikipedia.org';
```

El navegador hará tal petición Web sin que el usuario haya mediado en tal circunstancia o sus efectos. Del mismo modo se producirá una nueva petición GET si se altera el valor de `window.location.search` o `window.location.protocol`.

2.5.2.4 PROCEDIMIENTO DEL NAVEGADOR

La tarea del navegador Web es crear la petición a partir de los datos recogidos en el entorno de usuario de elementos del mismo, como enlaces, el valor del texto de la barra de búsqueda, los metatags.

```
<a href="http://es.wikipedia.org">Entrar</a>
```

Al pulsar en el enlace, el navegador crea automáticamente la petición GET y las cabeceras de la petición en base a los metatags (cabeceras definidas), los cookies y cabeceras automáticas del navegador, para luego enviarlas junto a la petición al Servidor.

2.5.2.5 PETICION POST

Es el segundo tipo de petición HTTP más utilizado. Los datos a enviar al servidor se incluyen en el cuerpo de la misma petición con las cabeceras HTTP asignadas correspondientemente respecto al tipo de petición. Generalmente se asocia con los formularios WEB en el que los datos suelen ser cifrados para enviarlos de manera segura al servidor.

Por motivos de convención se incluye en la petición la cabecera `[application/x-www-form-urlencoded]` que indica el formato o codificación de los datos a enviar; esta es **variable->valor** en el formato: **variable=valor** separada cada par **variable->valor** por **&**. Esta cabecera, en los formularios HTML se envía automáticamente, pero en otras tecnologías WEB tal como AJAX, si se desea

²⁰ Fuente: http://es.wikipedia.org/wiki/Servidor_web#Petici.C3.B3n_GET_pasiva

hacer correctamente una petición POST debe ser especificado o instanciado el objeto:

```
setRequestHeader("Content-type:application/x-www-form-urlencoded");  
ajax.send(data);
```

Si se utilizase el método GET los datos deberían de ser añadidos a la URL, lo que los expondría a ser vistos de forma directa.

2.5.2.6 COMPOSICION DE UNA PETICION POST

Las cabeceras más comunes que se envían en una petición POST²¹:

- **Petition type:** Especifica el tipo de petición HTTP. (Esta cabecera no tiene nombre, se envía tal cual)
- **Referer:** Especifica la URL desde la cual se hizo la petición POST.
- **Content-Length:** Especifica la longitud en bytes de los datos enviados en el cuerpo de la petición.
- **Origin:** Especifica la URL principal del sitio.[cita requerida]
- **User-Agent:** Especifica el identificador del navegador Web desde el cual se hizo la petición.
- **Content-Type:** Especifica el formato o MIME de los datos enviados en el cuerpo de la petición.
- **Accept:** Especifica el MIME que se espera en la respuesta.
- **Accept-Language:** Especifica el código del lenguaje esperado en la respuesta.
- **Accept-Charset:** Especifica la codificación que se espera en la respuesta.
- **Cookie:** Especifica un identificador de sesión en la petición derivado de un cookie.
- **Accept-Encoding:** Especifica el tipo de codificación (generalmente compresión) que se espera de la respuesta. (No todos los navegadores envían esta cabecera)

²¹ Fuente: http://es.wikipedia.org/wiki/Servidor_web#Composici.C3.B3n_de_una_petici.C3.B3n_POST

2.5.2.7 IIS V6 – WINDOWS SERVER 2003

Los Servicios de Internet Information Server (IIS) 6.0 con el sistema operativo Microsoft® Windows Server 2003™²² proporciona una funcionalidad de servidor Web integrada, confiable, escalable, segura y administrable a través de una intranet, Internet o una extranet.

IIS es una herramienta para crear una sólida plataforma de comunicaciones de aplicaciones de red dinámicas. Organizaciones de todos los tamaños utilizan IIS para alojar y gestionar páginas Web en Internet o en su intranet, para alojar y gestionar sitios FTP y para dirigir las noticias o el correo utilizando el Protocolo de transferencia de noticias en la red (NNTP) y el Protocolo simple de transferencia de correo (SMTP).

IIS 6.0 aprovecha los últimos estándares Web como Microsoft ASP.NET, XML y el Protocolo simple de acceso a objetos (SOAP) para el desarrollo, la implementación y la administración de aplicaciones Web. IIS 6.0 incluye nuevas funciones diseñadas para ayudar a las organizaciones, los profesionales de TI y los administradores de servidores a conseguir sus objetivos de rendimiento, confiabilidad, escalabilidad y seguridad para miles de sitios Web en un único servidor IIS o en varios servidores.

CARACTERISTICAS

- **Fiabilidad.**- IIS 6.0 utiliza una nueva arquitectura de proceso de peticiones y un entorno de aislamiento de aplicaciones que permite que las aplicaciones Web funcionen en un proceso de trabajo auto contenido. Este entorno evita que una aplicación o sitio Web detenga a otra u otro y reduce la cantidad de tiempo que emplean los administradores en reiniciar los servicios para corregir problemas relacionados con las aplicaciones. El nuevo entorno incluye también la Estado del grupo de aplicaciones.

²² Fuente: <http://www.microsoft.com/technet/itsolutions/msf/default.msp>

- **Escalabilidad.**- IIS 6.0²³ incorpora un nuevo controlador en modo de núcleo para el análisis y almacenamiento en caché de HTTP, ajustado específicamente para aumentar el rendimiento del servidor Web y la escalabilidad de los equipos multiproceso, con lo que se aumenta de forma significativa:
 - Número de sitios para alojar un único servidor IIS 6.0
 - Número de procesos de trabajo activos simultáneamente
- **Seguridad.**- IIS 6.0 ofrece un nivel de seguridad notablemente superior al de versiones anteriores de IIS. Para reducir la superficie de ataques de los sistemas, IIS no se instala de forma predeterminada en los sistemas operativos de la familia Windows Server 2003. Los administradores deben seleccionar e instalar IIS de forma explícita. IIS se instala de forma predeterminada en un estado de bloqueo, con capacidad para servir sólo contenido estático. Al utilizar el nodo de Extensiones de servicio Web, los administradores de sitios Web pueden Configurar IIS para contenido dinámico la funcionalidad IIS en función de las necesidades individuales de su organización. IIS 6.0 incluye diversas tecnologías y características de Seguridad en IIS 6.0 para garantizar la integridad del contenido del sitio Web y FTP, así como de los datos transmitidos a través de los sitios. Las funciones de seguridad de IIS incluyen las siguientes tareas relacionadas con la seguridad:
 - Autenticación en IIS 6.0
 - Controlar el acceso con IIS 6.0
 - Cifrado IIS 6.0
 - Certificates_IIS_SP1_Ops
 - Auditar en IIS 6.0

²³ Fuente: <http://support.microsoft.com>

- **Administración.-** IIS proporciona varias herramientas de administración para satisfacer las necesidades de un conjunto diverso de organizaciones. Los administradores pueden configurar un servidor IIS 6.0 mediante el Administrador IIS, al usar secuencias de comandos de administración desde la línea de comandos o directamente Habilitar la edición en ejecución en IIS 6.0.
- **Desarrollo mejorado.-** En comparación con los sistemas operativos anteriores, la familia Windows Server 2003 ofrece una mejor experiencia para el programador con Acerca de ASP.NET y la integración de IIS. ASP.NET reconoce la mayoría del código ASP al tiempo que proporciona una mayor funcionalidad para crear aplicaciones Web para la empresa que pueden funcionar como parte de Microsoft .NET Framework. Utilizar ASP.NET le permite aprovechar al máximo las funciones de Common Language Runtime, como la seguridad de tipos, la herencia, la interoperabilidad de lenguajes y las versiones. IIS 6.0 ofrece también compatibilidad con los últimos estándares Web, incluidos XML, SOAP y el Características de Protocolo Internet versión 6.
- **Compatibilidad de aplicaciones.-** IIS 6.0 es compatible con la mayoría de las aplicaciones existentes, basado en los comentarios de miles de clientes y proveedores de software independientes (ISV). Además, para asegurar la máxima compatibilidad, IIS 6.0 se puede configurar para que se ejecute en Modo de aislamiento de IIS 5.0 en IIS 6.0.

Versión	Obtenida	Sistema operativo
1.0	Se incluye con Windows NT 3.51 Service Pack 3 (o como una descarga independiente).	Windows NT Server 3.51
2.0	Incluida con Windows NT Server 4.0.	Windows NT Server 4.0
3.0	Incluido con Windows NT Server 4.0 Service Pack 3 (Internet Information Server 2.0 se actualiza automáticamente a Internet Information Server 3.0 durante la instalación del SP3 de).	Windows NT Server 4.0
4.0	Descarga independiente desde www.microsoft.com o el disco compacto de Windows NT Option Pack.	Windows NT Server 4.0 SP3 y Microsoft Internet Explorer 4.01
5.0	Componente integrado de Windows 2000.	Windows 2000
5.1	Componente integrado de Windows XP Professional.	Windows XP
6.0	Componente integrado de Windows Server 2003.	Windows Server 2003
7.0	Componente integrado de Windows Vista y Windows Server 2008.	Windows Vista y Windows Server 2008

Tabla 2.3. Versiones IIS²⁴

2.6. BASES DE DATOS ORACLE SOBRE MICROSOFT WINDOWS

2.6.1. INTRODUCCION

Ser un administrador de bases de datos ORACLE²⁵ puede ser una carrera exigente pero gratificante que conlleva una gran responsabilidad, sin embargo para emprender proyectos de desarrollo de software no es necesario ser un DBA.

Por ello los temas tratados en este apartado se enfocarán en obtener los conocimientos necesarios para entender la arquitectura de las bases de datos

²⁴ Fuente: Fuente: <http://support.microsoft.com/kb/224609/es>

²⁵ Oracle 8 Manual de Referencia - Koch, George. Osborne / McGraw-Hill 1999.

ORACLE, su funcionamiento, las ventajas que ofrece, sus interacciones con el sistema operativo y los componentes básicos para manipular los datos.

El enfoque que se tratará será bajo el sistema operativo WINDOWS y el software Oracle8.

2.6.2. TERMINOS DE BASES DE DATOS

- **SHARED POOL**²⁶.- Se encarga de recopilar, analizar, interpretar y ejecutar todas las sentencias SQL que van en contra de la base de datos ORACLE. Por lo tanto, el shared pool es un componente clave, contiene los siguientes componentes (The library cache, The dictionary cache, Control structures)
- **AD-HOC QUERY**.- Este uso del término en latín significa una consulta espontánea, sencilla.
- **BLOCK**.- El bloque es la unidad más pequeña de almacenamiento en una base de datos ORACLE. El bloque de base de datos contiene información de encabezado sobre el mismo bloque, así como el código de datos o PL / SQL.
- **BOTTLENECK**.- En términos informáticos, un cuello de botella es un componente del sistema que limita el rendimiento del sistema.
- **BUFFER**.- Este término se refiere a una cantidad de memoria utilizada para almacenar datos. Puede ser utilizado como una copia de los datos para el acceso de lectura rápida, se pueden modificar y escribir en el disco, o se pueden crear en la memoria como el almacenamiento temporal.
- **CACHE**.- La memoria caché es un área de almacenamiento para proporcionar acceso rápido a los datos. En términos de hardware, la memoria caché es una pequeña (en relación a los principales RAM) cantidad de memoria que es mucho más rápida que la memoria principal.

²⁶ Fuente: <http://creativecommons.org/licenses/by-nc-sa/2.0/>

- **CHECKPOINT.-** Un punto de control es una operación que obliga a todos los cambios en memoria de los bloques de datos que se escriben en el disco. Este es un factor clave en el tiempo la base de datos tarda en recuperarse en caso de un fracaso.
- **CLEAN BUFFER.-** Un buffer limpio es un búfer que no se ha modificado. Debido a que este buffer no se ha cambiado, no es necesario para el DBWR a escribir este buffer en el disco.
- **CONCURRENCY.-** Este término se refiere a la capacidad de realizar muchas funciones al mismo tiempo. ORACLE²⁷ proporciona para la concurrencia el permitir que muchos usuarios para acceder a la base de datos simultáneamente.
- **DATABASE.-** Una base de datos es un conjunto de datos organizados para un fácil acceso, son los datos reales. Es la base de datos que va a acceder cuando se necesita para recuperar los datos.
- **DATA DICTIONARY.-** El diccionario de datos es un conjunto de tablas de ORACLE utiliza para mantener información sobre la base de datos. El diccionario de datos contiene información sobre las tablas, índices, clusters, etc.
- **DBA (DATABASE ADMINISTRATOR).-** Es la persona responsable de la operación, configuración y rendimiento de la base de datos, se encarga de mantener el funcionamiento de las bases de datos sin problemas, garantiza que los backups y copias de seguridad se realizan de forma regular, además de la instalación de nuevo software.
- **DBMS OR RDBMS.-** Sistema de Gestión de Base de Datos, es el software y la colección de herramientas que gestiona la base de datos. El software de ORACLE es el DBMS. Un sistema de base de datos relacional es un DBMS que es de naturaleza relacional.
- **DDL (DATA DEFINITION LANGUAGE).-** Son comandos se utilizan en la creación y modificación de objetos de esquema. Estos comandos

²⁷ Oracle Manual de Referencia - Koch, George. Osborne/McGraw-Hill 1997.

proporcionan la capacidad de crear, modificar y borrar objetos, otorgar y revocar privilegios y roles, establecer las opciones de auditoria, y añadir comentarios al diccionario de datos.

- **DIRTY BUFFER.**- Es un buffer que ha sido modificado. Es el trabajo de la DBWR escribir eventualmente todos los bloques utilizados en el disco.
- **DML (DATA MANIPULATION LANGUAGE).**- Estos comandos le permiten consultar y modificar datos dentro de objetos de esquemas existentes. A diferencia de los comandos DDL, un COMMIT (confirmar grabar datos al disco) no está implícito. DML consiste en borrar, insertar, buscar y actualizar, declaraciones de planes de ejecución, y sentencias de bloqueo de tablas.
- **DYNAMIC PERFORMANCE TABLES.**- Estas tablas se crean en el inicio de la instancia y son utilizadas para almacenar información sobre el desempeño de la instancia. Esta información incluye información de conexión, E/S, los valores de parámetros de inicialización, y así sucesivamente.
- **FUNCTION - PROCEDURE.**- Una función es un conjunto de sentencias SQL o PL/SQL que se utilizan en conjunto para ejecutar una función particular. Procedimientos y las funciones son idénticas, excepto que las funciones siempre devuelven un valor (procedimientos no). Al procesar el código SQL en el servidor de base de datos, puede reducir el número de instrucciones que se envían a través de la red y volver de las sentencias SQL.
- **NETWORK COMPUTING ARCHITECTURE (NCA).**- El NCA es un estándar para la informática en la red. El ANC fue desarrollado en conjunto con ORACLE.
- **PHYSICAL MEMORY.**- Este término se refiere a la RAM del hardware actual (memoria de acceso aleatorio) disponible en el ordenador para su uso por el sistema operativo y aplicaciones.
- **PROGRAM UNIT.**- En ORACLE, la unidad de programa se utiliza para describir un paquete, un procedimiento almacenado, o una secuencia.

- **QUERY.-** Una consulta es una transacción de sólo lectura contra una base de datos. Una consulta se genera con la instrucción SELECT. Los usuarios suelen distinguir entre las consultas y otros tipos de transacciones, porque una consulta no modifica los datos la base de datos.
- **SCHEMA.-** Un esquema es una colección de objetos relacionados con la base de datos.
- **SCHEMA OBJECTS.-** Los objetos de esquema son abstracciones o estructuras lógicas que hacen referencia a objetos de base de datos o las estructuras. Los objetos de esquema consisten en cosas como los clusters, índices, paquetes, secuencias, procedimientos almacenados, sinónimos, tablas, vistas, etc.
- **SYSTEM GLOBAL AREA (SGA).-** Es una región de memoria compartida que utiliza ORACLE para almacenar datos e información de control de una instancia de ORACLE. El SGA se asigna cuando la instancia de ORACLE se inicia, sino que se cancela la asignación cuando la instancia de ORACLE se apaga. Cada instancia de ORACLE²⁸ que comienza tiene su propio SGA. La información en el SGA se compone de los buffers de base de datos, el buffer de redo log, y el shared pool, cada uno tiene un tamaño fijo y se crea cuando la instancia se inicia.
- **TRANSACTION.-** Es una unidad lógica de trabajo que consiste en una o más sentencias SQL, que termina en una confirmación o una reversión. Mediciones de desempeño utilizan a menudo el número de transacciones por segundo o por minuto como métrica del rendimiento.
- **TRIGGER.-** Un disparador es un mecanismo que permite escribir procedimientos que se ejecutan automáticamente cada vez que un INSERT, UPDATE, o DELETE se ejecuta en una tabla o vista. Los disparadores pueden ser utilizados para hacer cumplir las restricciones de integridad o automatizar algunas funciones.

²⁸ Mastering Oracle - Cronin, Daniel. Hayden Books 1999

- **VIRTUAL MEMORY.-** Este término se refiere a la memoria que pueden ser utilizados por los programas del sistema operativo. Para superar las limitaciones asociadas con suficiente memoria física, memoria virtual permite que los programas de ejecución que son más grandes que la cantidad de memoria física del sistema. Cuando no hay suficiente memoria física del sistema, estos programas se copian de la memoria RAM a un archivo de disco denominado paginación o archivo de intercambio
- **OLTP.-** Procesamiento de transacciones en línea, es probablemente el más común de las configuraciones de RDBMS. Sistemas OLTP tienen usuarios en línea que el acceden al sistema. Estos sistemas se utilizan normalmente para fines de la entrada de pedidos, como para la venta al por menor, la validación de tarjetas de crédito, transacciones en cajeros automáticos, etc.
- **DSS.-** Sistema de Soporte de Decisiones, se utilizan para ayudar en el proceso de toma de decisiones. Estas decisiones pueden basarse en información como la cantidad de ventas en una región en particular, que muestra representativa de clientes están comprando un producto en particular, a quiénes envían un correo. El sistema DSS se usa para ayudar a tomar decisiones, proporcionando datos de calidad.
- **DATA WAREHOUSE.-** Es generalmente considerado como un sistema a gran escala, que consta de dos componentes del DSS y OLTP. Estos sistemas son normalmente de cientos de GIGABYTES de tamaño y el apoyo de muchos usuarios.
- **DATA MART.-** Es una versión a menor escala de un DATA WAREHOUSE, suele ser de 100 GB o menor. Es compatible con muchos usuarios en línea, así como una función de soporte de decisiones.
- **OLAP.-** Online Analytical Processing, se utiliza generalmente en relación con los datos multidimensionales. Usuarios de OLAP podría ser analistas financieros o de personal de marketing que buscan en los datos globales.

2.6.3. ARQUITECTURA DE ORACLE

El Oracle Relational Database Management System, o RDBMS, está diseñado para permitir el acceso simultáneo a una gran cantidad de información almacenada. El RDBMS consiste en la base de datos (la información) y la instancia (la realización del sistema). La base de datos contiene los archivos físicos que residen en el sistema y las piezas lógicas, como el esquema de base de datos. La instancia es el método utilizado para acceder a los datos y se compone de los procesos y la memoria del sistema.

2.6.3.1 LA BASE DE DATOS

La base de datos ORACLE tiene una capa lógica y una capa física. La capa física se compone de los archivos que residen en el disco, los componentes de la capa lógica de los datos de estos componentes físicos.

2.6.3.2 LA CAPA FISICA

La capa física de la base de datos consta de tres tipos de archivos:

- Uno o más archivos de datos (datafiles), almacenan la información contenida en la base de datos. Se puede tener tan sólo un datafile o hasta cientos de datafiles. La información para una sola tabla puede abarcar muchos datafiles o muchas tablas pueden compartir un conjunto de datafiles.
- Dos o más archivos de registro (redo log files), mantienen la información utilizada para la recuperación en caso de un fallo del sistema. Redo log files, guardan un registro de todos los cambios realizados a la base de datos. Esta información se utiliza en el caso de un fallo del sistema para volver a aplicar los cambios que se han hecho y confirmado, pero que tal vez no se han sido hechos a los ficheros de datos.
- Uno o más archivos de control, contienen información que se utiliza para iniciar una instancia, como la ubicación de los archivos de datos y los archivos de registro, ORACLE necesita esta información para iniciar la instancia de base de datos. Los archivos de control deben ser protegidos.

ORACLE proporciona un mecanismo para almacenar múltiples copias de archivos de control.

2.6.3.3 LA CAPA LOGICA

La capa de lógica de la base de datos consta de los siguientes elementos:

- Uno o mas tablespaces
- El esquema de base de datos, que consiste en elementos tales como tablas, clusters, índices, vistas, procedimientos almacenados, disparadores de base de datos, secuencias, etc.

2.6.3.4 TABLESPACES Y DATAFILES

La base de datos está dividida en una o más piezas lógicas conocidas como tablespaces. Un tablespace se utiliza para agrupar lógicamente los datos. Por ejemplo, puede crear un tablespace para la contabilidad y un tablespace por separado para la compra. La segmentación de los grupos en diferentes tablespaces simplifica la administración de estos grupos (ver Figura 18). Tablespaces constan de uno o más datafiles. Al utilizar más de un datafile por tablespace, pueden transmitir datos a través de muchos discos diferentes para distribuir la carga de E/S y mejorar el rendimiento.

2.6.3.5 ESQUEMA DE BASE DE DATOS

Es una colección de objetos lógico-estructura, conocida como esquema de objetos, que definen cómo se ven los datos de la base de datos. Consisten en estructuras tales como tablas, clusters, índices, vistas, procedimientos almacenados, disparadores de base de datos, y las secuencias.

- **TABLA.**- consta de un nombre de tabla, filas y columnas de datos. Es la unidad básica de almacenamiento lógico en la base de datos ORACLE. Las columnas se definen por su nombre y tipo de datos. Una tabla se almacena dentro de un tablespace, a menudo, muchas tablas comparten un tablespace.

- **CLUSTER.**- Es un conjunto de tablas físicamente almacenados como una tabla en la que comparte una columna común. Si los datos de dos o más tablas con frecuencia se recuperan juntos sobre la base de datos de la columna común, utilizando una tabla agrupada puede ser muy eficiente.
- **INDICE.**- Es una estructura creada para ayudar a recuperar los datos más rápida y eficaz (al igual que el índice de un libro permite encontrar una sección específica con mayor rapidez). Un índice se declara en una columna o conjunto de columnas.
- **VISTA.**- Es una ventana dentro de una o más tablas. Una vista no almacena datos, sino que presenta datos de la tabla. Una vista puede ser consultada, actualizada y eliminada en forma de tabla sin restricciones.
- **PROCEDIMIENTO ALMACENADO.**- es una consulta SQL predefinida que se almacena en el diccionario de datos. Los procedimientos almacenados están diseñados para permitir consultas más eficaces. El uso de procedimientos almacenados, pueden reducir la cantidad de información que se debe pasar a la RDBMS y reducir así el tráfico de red y mejorar el rendimiento.
- **TRIGGER.**- Es un procedimiento que se ejecuta automáticamente cuando se produce un evento. Este procedimiento, que es definido por el administrador o desarrollador, factores desencadenantes, o ejecutarse cada vez que se produce este evento. Este procedimiento podría ser una inserción, eliminación, o incluso una selección de datos de una tabla.
- **SECUENCIA.**- El generador de secuencias de ORACLE se utiliza para generar automáticamente una secuencia única de números en la memoria caché. Al utilizar el generador de secuencias se puede evitar los pasos necesarios para crear esta secuencia por su cuenta, tales como bloquear el registro que tiene el valor último de la secuencia, lo que genera un nuevo valor, y luego desbloquear el registro.

2.6.4. PL/SQL (PROCEDURAL LANGUAGE/STRUCTURED QUERY LANGUAGE)

Es un lenguaje que añade al SQL las características de un lenguaje procedimental. Las siglas son Procedural Lenguaje - SQL. Este lenguaje tiene que ser resuelto por el sistema gestor de base de datos, pero hay herramientas que tienen un motor de PL-SQL para que solo vaya al sistema gestor de base de datos los datos necesarios mientras que el programa hará otro tipo de procedimientos.

PL/SQL soportara todas las consultas, ya que la manipulación de datos que se usa es la misma que en SQL, incluyendo nuevas características:

- El manejo de variables.
- Estructuras modulares.
- Estructuras de control de flujo y toma de decisiones.
- Control de excepciones.

Los programas o paquetes de PL/SQL se pueden almacenar en la base de datos como otro objeto, y todos los usuarios que estén autorizados tienen acceso a estos paquetes. Los programas se ejecutan en el servidor para ahorrar recursos a los clientes.

2.6.4.1 TIPOS DE SENTENCIAS

Las sentencias SQL pertenecen a dos categorías principales: Lenguaje de Definición de Datos, DDL y Lenguaje de Manipulación de Datos, DML. Estos dos lenguajes no son lenguajes en sí mismos, sino que es una forma de clasificar las sentencias de lenguaje SQL en función de su cometido. La diferencia principal reside en que el DDL crea objetos en la base de datos y sus efectos se pueden ver en el diccionario de la base de datos; mientras que el DML es el que permite consultar, insertar, modificar y eliminar la información almacenada en los objetos de la base de datos.

<i>Sentencia DDL</i>	<i>Objetivo</i>
Alter procedure	Recompilar un procedimiento almacenado.
Alter Table	Añadir o redefinir una columna, modificar la asignación de almacenamiento.
Analyze	Recoger estadísticas de rendimiento sobre los objetos de la BD para utilizarlas en el optimizador basado en costes.
Create Table	Crear una tabla.
Create Index	Crear un índice.
Drop Table	Eliminar una tabla.
Drop Index	Eliminar un índice.
Grant	Conceder privilegios o papeles, roles, a un usuario o a otro rol.
Truncate	Eliminar todas las filas de una tabla.
Revoke	Retirar los privilegios de un usuario o rol de la base de datos.
<i>Sentencia DML</i>	<i>Objetivo</i>
Insert	Añadir filas de datos a una tabla.
Delete	Eliminar filas de datos de una tabla.
Update	Modificar los datos de una tabla.
Select	Recuperar datos de una tabla.
Commit	Confirmar como permanentes las modificaciones realizadas.
Rollback	Deshacer todas las modificaciones realizadas desde la última confirmación.

Tabla 2.4. Tipos de sentencias PL/SQL²⁹

2.6.4.2 TIPOS DE DATOS

Existen varios tipos de datos en SQL. De esta manera, cada columna puede albergar una información de naturaleza distinta. Los tipos de datos más comunes y sus características se resumen en la siguiente tabla:

<i>Tipo de Dato</i>	<i>Descripción</i>
VARCHAR2 (tamaño)	Almacena datos de tipo carácter alfanumérico de longitud variable, con un tamaño máximo de 2.000.
CHAR (tamaño)	Almacena datos de tipo carácter alfanumérico de longitud fija, con un tamaño máximo de 255.
LONG	Almacena datos de tipo carácter alfanumérico de longitud variable con un tamaño máximo de hasta 2 Gb.
NUMBER (dig, dec)	Almacena datos numéricos de <i>dig</i> dígitos, de los cuales <i>dec</i> son decimales. El tamaño máximo es de 38 dígitos.
DATE	Almacena fechas desde el 1-Ene-4712 AC hasta el 31-Dic-4712 DC.
RAW (tamaño)	Almacena datos de longitud variable, con un tamaño máximo de 255 bytes.
LONG RAW	Almacena datos de longitud variable, con un tamaño máximo de 2 Gb.

Tabla 2.5. Tipos de datos PL/SQL³⁰

Operadores de Comparación:

²⁹ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

³⁰ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

<i>Operador</i>	<i>Operación</i>
=	Igualdad
!=, <>, ^=	Desigualdad
<	Menor que
>	Mayor que
<=	Menor o igual que
>=	Mayor o igual que
in	Igual a cualquiera de los miembros entre paréntesis
not in	Distinto a cualquiera de los miembros entre paréntesis
between	Contenido en el rango
not between	Fuera del rango
like '_abc%'	Contiene la cadena 'abc' a partir del segundo carácter y luego cualquier cadena de caracteres

Tabla 2.6. Tipos de operadores lógicos PL/SQL³¹

Operadores de Aritméticos:

<i>Operador</i>	<i>Operación</i>	<i>Ejemplo</i>
+	Suma	<code>select nombre, salario+comision from emp where oficio='VENDEDOR';</code>
-	Resta	<code>select nombre from emp where sysdate-fecha_alta > 365;</code>
*	Producto	<code>select nombre, salario*12 from emp;</code>
/	División	<code>select nombre, salario/31 from emp;</code>

Tabla 2.7. Tipos de operadores aritméticos PL/SQL³²

Operadores de Cadenas de Caracteres:

<i>Operador</i>	<i>Operación</i>	<i>Ejemplo</i>
	Concatenación	<code>select nombre oficio from emp;</code>

Tabla 2.8. Tipos de operadores de caracteres PL/SQL³³

2.6.4.3 FUNCIONES

³¹ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

³² Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

³³ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

Existen en SQL muchas funciones que pueden complementar el manejo de los datos en las consultas. Se utilizan dentro de las expresiones y actúan con los valores de las columnas, variables o constantes.

Se pueden incluir en las cláusulas SELECT, WHERE y ORDER BY.

Pueden anidarse funciones dentro de funciones. Y existe una gran variedad de funciones para cada tipo de datos:

Función	Cometido	Ejemplo	Resultado
ABS(n)	Calcula el valor absoluto de <i>n</i> .	<code>select abs(-15) from dual;</code>	15
CEIL(n)	Calcula el valor entero inmediatamente superior o igual a <i>n</i> .	<code>select ceil(10.7) from dual;</code>	11
FLOOR(n)	Calcula el valor entero inmediatamente inferior o igual a <i>n</i> .	<code>select floor(10.7) from dual;</code>	10
MOD(m,n)	Calcula el resto resultante de dividir <i>m</i> entre <i>n</i> .	<code>select mod(11,4) from dual;</code>	3
POWER(m,n)	Calcula la potencia <i>n</i> -ésima de <i>m</i> .	<code>select power(3,2) from dual;</code>	9
ROUND(m,n)	Calcula el redondeo de <i>m</i> a <i>n</i> decimales. Si <i>n</i> <0 el redondeo se efectúa a por la izquierda del punto decimal.	<code>select round(123.456,1) from dual;</code>	123.5
SQRT(n)	Calcula la raíz cuadrada de <i>n</i> .	<code>select sqrt(4) from dual;</code>	2
TRUNC(m,n)	Calcula <i>m</i> truncado a <i>n</i> decimales (<i>n</i> puede ser negativo).	<code>select trunc(123.456,1) from dual;</code>	123.4
SIGN(n)	Calcula el signo de <i>n</i> , devolviendo -1 si <i>n</i> <0, 0 si <i>n</i> =0 y 1 si <i>n</i> >0.	<code>select sign(-12) from dual;</code>	-1

Tabla 2.9. Funciones PL/SQL³⁴

Función	Cometido	Ejemplo	Resultado
CHR(n)	Devuelve el carácter cuyo valor codificado es <i>n</i> .	<code>select chr(65) from dual;</code>	A
ASCII(cad)	Devuelve el valor ASCII de <i>cad</i> .	<code>select ascii('A') from dual;</code>	65
CONCAT(cad1,cad2)	Devuelve <i>cad1</i> concatenada con <i>cad2</i> . Esta función es equivalente al operador	<code>select concat(concat(nombre,' as '),oficio) from emp;</code>	Casio es Presidente, etc
LOWER(cad)	Devuelve la cadena <i>cad</i> con todas sus letras convertidas a minúsculas.	<code>select lower('MiNUsCuLAs') from dual;</code>	minúsculas
UPPER(cad)	Devuelve la cadena <i>cad</i> con todas sus letras convertidas a mayúsculas.	<code>select upper('maYUsCuLAs') from dual;</code>	MAYUSCULAS
INITCAP(cad)	Devuelve <i>cad</i> con el primer carácter en mayúsculas.	<code>select initcap('isabel') from dual;</code>	Isabel
LPAD(cad1,n,cad2)	Devuelve <i>cad1</i> con longitud <i>n</i> , y ajustada a la derecha, rellenando por la izquierda con <i>cad2</i> .	<code>select lpad('P',5,'*') from dual;</code>	****P
RPAD(cad1,n,cad2)	Devuelve <i>cad1</i> con longitud <i>n</i> , y ajustada a la izquierda, rellenando por la derecha con <i>cad2</i> .	<code>select rpad('P',5,'*') from dual;</code>	P****
REPLACE(cad,ant,nie)	Devuelve <i>cad</i> en la que cada ocurrencia de la cadena <i>ant</i> ha sido sustituida por la cadena <i>nie</i> .	<code>select replace('diego','i','ie') from dual;</code>	diego
SUBSTR(cad,m,n)	Devuelve la subcadena de <i>cad</i> compuesta por <i>n</i> caracteres a partir de la posición <i>m</i> .	<code>select substr('ABCDEFC',3,2) from dual;</code>	CD
LENGTH(cad)	Devuelve la longitud de <i>cad</i> .	<code>select length('cadena') from dual;</code>	6

Tabla 2.10. Funciones para cadenas de caracteres³⁵

³⁴ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

³⁵ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

Función	Cometido	Ejemplo	Resultado
SYSDATE	Devuelve la fecha y hora actuales.	select sysdate from dual;	14-MAR-97
ADD_MONTHS(d,n)	Devuelve la fecha <i>d</i> incrementada en <i>n</i> meses.	select add_months(sysdate,4) from dual;	14-JUL-97
LAST_DAY(d)	Devuelve la fecha del último día del mes de <i>d</i> .	select last_day(sysdate) from dual;	31-MAR-97
MONTHS_BETWEEN(d1, d2)	Devuelve la diferencia en meses entre las fechas <i>d1</i> y <i>d2</i> .	select months_between(sysdate, '01-JAN-97') from dual;	2 41409424
NEXT_DAY(d,cad)	Devuelve la fecha del primer día de la semana <i>cad</i> después de la fecha <i>d</i> .	select next_day(sysdate, 'sunday') from dual;	18-MAR-97

Tabla 2.11. Funciones de manejo de fechas³⁶

Función	Cometido	Ejemplo	Resultado
TO_NUMBER(<i>cad</i> , <i>fmto</i>)	Convierte la cadena <i>cad</i> a un número, opcionalmente de acuerdo con el formato <i>fmto</i> .	select to_number('12345') from dual;	12345
TO_CHAR(<i>d</i> , <i>fmto</i>)	Convierte la fecha <i>d</i> a una cadena de caracteres, opcionalmente de acuerdo con el formato <i>fmto</i> .	select to_char(sysdate) from dual;	14-MAR-97
TO_DATE(<i>cad</i> , <i>fmto</i>)	Convierte la cadena <i>cad</i> de tipo varchar2 a fecha, opcionalmente de acuerdo con el formato <i>fmto</i> .	select to_date('01-JAN-97') from dual;	01-JAN-97

Tabla 2.12. Funciones de Conversión de Tipos³⁷

Función	Cometido
COUNT(<i>col</i>)	Cuenta el número de filas agrupadas.
AVG(<i>col</i>)	Calcula el valor medio de todos los valores de la columna <i>col</i> .
MAX(<i>col</i>)	Calcula el valor máximo de todos los valores de la columna <i>col</i> .
MIN(<i>col</i>)	Calcula el valor mínimo de todos los valores de la columna <i>col</i> .
SUM(<i>col</i>)	Calcula la suma de los valores de la columna <i>col</i> .
STDDEV(<i>col</i>)	Calcula la desviación típica de los valores de la columna <i>col</i> sin tener en cuenta los valores nulos.
VARIANCE(<i>col</i>)	Calcula la varianza de los valores de la columna <i>col</i> sin tener en cuenta los valores nulos.

Tabla 2.13. Funciones de agrupamiento³⁸

Función	Cometido	Ejemplo	Resultado
DECODE(<i>var</i> , <i>val1</i> , <i>cod1</i> , <i>val2</i> , <i>cod2</i> , ..., <i>defecto</i>)	Convierte el valor de <i>var</i> , de acuerdo con la codificación.	select decode(empno, 'Presidente', 'P', 'Director', 'D', 'X') from emp;	P, D, X, ...
GREATEST(<i>exp1</i> , <i>exp2</i> , ...)	Devuelve el mayor valor de una lista.	sin ejemplo.	sin ejemplo.
LEAST(<i>cad</i> , <i>fmto</i>)	Devuelve el menor valor de una lista.	sin ejemplo.	sin ejemplo.
NVL(<i>val</i> , <i>exp</i>)	Devuelve la expresión: <i>exp</i> si <i>val</i> es NULL, y <i>val</i> si en otro caso.	select sal+nvl(comm,0) from emp;	450000, 350000, ...

Tabla 2.14. Otras funciones³⁹

³⁶ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

³⁷ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

³⁸ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

2.6.4.4 VARIABLES

Las variables⁴⁰ son nombres para procesar los elementos de los datos.

Declaración:

Nombre_variable tipo [NOT NULL] [:= valor | DEFAULT valor]

:= y DEFAULT son lo mismo.

Si ponemos NOT NULL es obligatorio inicializar la variable.

Ejemplos:

```
num_dep NUMBER(2) NOT NULL :=20
nom_emple VARCHAR2(15) DEFAULT 'Pedro'
```

También se puede definir una variable a partir de un campo mediante los atributos %TYPE y %ROWTYPE, con esto damos el tipo y longitud a la variable de otra variable u objeto ya definido.

%TYPE es la que se utiliza normalmente, %ROWTYPE es para claves de registro. El NOT NULL y el valor inicial no se heredan, sólo el tipo de dato y longitud de ese dato.

Por ejemplo:

```
num_dep emple.dept_no%TYPE
```

2.6.4.5 CONSTANTES

Las constantes son como las variables pero no puede modificarse su valor. Se declaran de la siguiente manera:

nombre_constante CONSTANT tipo_de_dato := valor

Por ejemplo:

³⁹ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/sqlplus/sqlplus.html>

⁴⁰ Oracle 8 Manual de Referencia - Koch, George. Osborne/McGraw-Hill 1999.

El IVA es un valor fijo, y para declararlo lo haríamos de la siguiente manera:

```
Imp_iva constant number(2,2) := 12,5
```

2.6.4.6 BLOQUE PL/SQL

Bloque⁴¹ es la unidad de estructura básica en los programas PL/SQL. Supone una mejora en el rendimiento, pues se envían los bloques completos al servidor para ser procesados en lugar de enviar cada secuencia SQL.

Partes de un bloque:

- Zona de declaraciones: zona opcional. Se declaran los objetos locales (variables, constantes...).
- Zona de instrucciones: zona obligatoria.
- Zona de tratamiento de excepciones: zona opcional.

Forma de crear un bloque:

```
[ DECLARE | IS / AS ]
    <declaraciones>
BEGIN
    <instrucciones>
[ EXCEPTION ]
    <tratamiento de excepciones>
END;
/
```

La barra "/" siempre se pone al final para ejecutar el bloque.

2.6.4.7 TIPOS DE BLOQUES

2.6.4.7.1 ANONIMOS – SIN NOMBRE

Siempre comienza con DECLARE o directamente con BEGIN.

Ejemplo 1;

```
BEGIN
DBMS_OUTPUT.PUT_LINE ('Hola');
END;
```

⁴¹ Oracle 8 Manual de Referencia - Koch, George. Osborne/McGraw-Hill 1999.

/

DBMS_OUTPUT es un depurador de Oracle que sirve para visualizar cualquier cosa, pero antes lo debemos tener activado (en SQL*plus, se activa con el comando SET SERVEROUTPUT ON):

```

SET
DECLARE
    fecha date;
BEGIN
select sysdate into fecha from dual;
dbms_output.put_line (to_char(fecha,
'day', "dd" de "month" de "yyyy", a las "hh24:mi:ss"));
END;
/

```

2.6.4.7.2 SUBPROGRAMAS

Se pueden almacenar en la base de datos.

Existen dos tipos de subprogramas: Procedimientos (PROCEDURE) y Funciones (FUNCTION⁴²)

Los procedimientos tienen la utilidad de fomentar la reutilización de programas que se usan comúnmente. Una vez compilado, queda almacenado en la base de datos (por eso es también llamado 'Procedimiento almacenado') y puede ser utilizado por múltiples aplicaciones.

La sintaxis es la siguiente:

```

CREATE [OR REPLACE] PROCEDURE nombre_procedimiento
    [nombre_parametro modo tipodatos_parametro ]
IS | AS
    bloque de código

```

Para utilizar un procedimiento almacenado de la base de datos simplemente se lo llama desde un bloque anónimo (desde la línea de comandos), previamente habiendo inicializado el/los parámetro/s (en caso que existan).

⁴² Oracle 8 Manual de Referencia - Koch, George. Osborne/McGraw-Hill 1999.

```

DECLARE
  nombre_parametro tipodatos_parametro;
BEGIN
  nombre_parametro tipodatos_parametro := valor_de_inicializacion;
  nombre_procedimiento (nombre_parametro => nombre_parametro);
END;
/

```

2.6.4.7.3 FUNCIONES

Una función⁴³ es un bloque de código PL/SQL que tiene las mismas características que un procedimiento almacenado. La diferencia estriba que una función devuelve un valor al retornar. Al devolver un valor pueden ser llamadas como parte de una expresión.

La sintaxis es:

```

CREATE [OR REPLACE] FUNCTION nombre_función
  [nombre_parámetro modo tipodatos_parametro ]
RETURN tipodatos_retorno IS | AS
  bloque de código

```

Los procedimientos y funciones se pueden agrupar en unas estructuras llamadas Paquetes.

2.6.4.7.4 TRIGGERS

Un trigger⁴⁴ o disparador se ejecuta ante un determinado evento de manera automática. Generalmente se utilizan para garantizar que una determinada acción siempre se realiza después de realizar una tarea determinada. Se debe

⁴³ Oracle 8 Manual de Referencia - Koch, George. Osborne/McGraw-Hill 1999.

⁴⁴ Oracle 8 Manual de Referencia - Koch, George. Osborne/McGraw-Hill 1999.

tener cuidado con este tipo de estructuras puesto que un uso excesivo puede dar lugar a dependencias difíciles de mantener. La sintaxis, A nivel de sentencia:

```
CREATE [OR REPLACE] TRIGGER nombre_trigger
    momento_ejecución evento [evento] ON nombre_tabla
    bloque PLSQL;
```

A nivel de registro:

```
CREATE [OR REPLACE] TRIGGER nombre_trigger
    momento_ejecución evento [evento] ON nombre_tabla
    [REFERENCING OLD AS old | NEW AS new]
    FOR EACH ROW
    [WHEN condición]
    bloque PLSQL;
```

2.7. TÉCNICAS Y USO DE ENCRIPCIÓN DE DATOS

2.7.1. INTRODUCCION

La pieza más sensible e importante en una empresa son los datos, cualquier mal uso de esta información puede desencadenar una debacle dentro de una empresa, por eso se le da tanta importancia al proteger los datos de ataques externos e internos en una empresa.

En ese tenor de gravedad, se utilizan los métodos de encriptación, que en ORACLE tienen su máximo exponente en el Oracle Advanced Security⁴⁵ Transparent Data Encryption (TDE), que tuvo su primera versión en Oracle10gr2, la principal ventaja de este tipo de encriptación física de datos es que las aplicaciones (código) no deben ser modificadas en lo absoluto, por lo cual el impacto sobre los sistemas empresariales es mínimo.

Nunca se debe dejar de lado que TDE es solamente una solución a nivel físico, no se realizan encriptaciones a nivel de cliente, sino, que a nivel de data almacenada.

2.7.2. FUNCIONAMIENTO DE TDE

TDE encripta los datos antes de que sean escritos a disco y los desencripta antes de que sean retornados a la aplicación, lo cual provoca que el impacto en el desarrollo/modificación de las aplicaciones sea mínimo. Todo lo que sea respaldo físico de las bases de datos estarán afectados a esta encriptación, opcionalmente TDE puede ser usado en conjunto con Recovery Manager (RMAN). Cabe mencionar que TDE forma parte del Oracle Advanced Security Option, por lo tanto se debe considerar ante temas de licenciamiento.

Un dato importante a tener en cuenta es que para encriptar o desencriptar una tabla con columnas encriptadas, Oracle almacena la llave de encriptación en la SGA.

⁴⁵ Fuente: <http://www.infor.uva.es/~jvegas/cursos/bd/oraseg/oraseg.html#2>

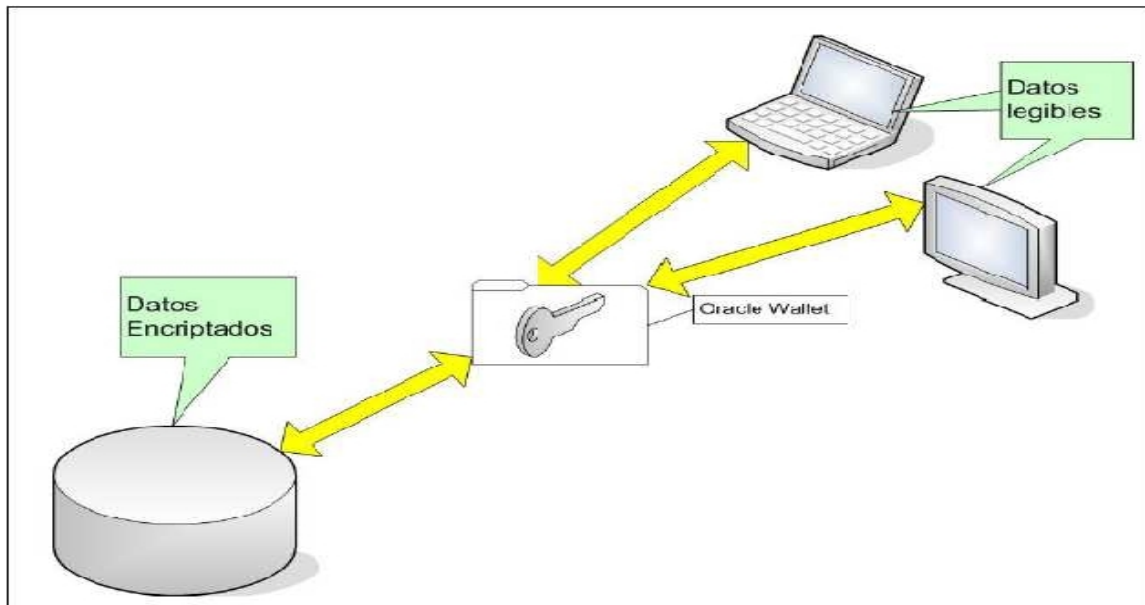


Figura 2.6. Datos encriptados con WALLET⁴⁶

El Wallet se genera en disco, es un archivo que se debiese respaldar siempre.

MANEJO DE LAS LLAVES EN TDE

TDE automáticamente crea una llave de encriptación cuando una columna de una tabla es encriptada. La llave de encriptación (table key) se usará para todas las columnas de esa tabla, por ende una tabla con 10 mil columnas encriptadas, solamente generará una llave, cada una de estas (table key) es almacenada en el diccionario de datos (dentro de la base de datos) y encriptada usando la llave maestra de encriptación del TDE (TDE Master Encryption key).

El TDE Master Encryption Key es almacenado fuera de la base de datos, en un Oracle Wallet, este Wallet es un archivo encriptado según el nombre ingresado por el DBA durante la creación del Wallet, esta clave también puede ser modificada con el Oracle Wallet Manager.

Este Wallet solamente debiese ser accedido por el usuario Oracle y la llave de creación del Wallet debiese tener un mínimo de 8 caracteres.

2.7.3. IMPLEMENTACION DEL TDE

⁴⁶ Fuente: <http://www.oracleyyo.com> [Introducción a TDE PAG 4 de 15]

Para llevar a cabo la implementación de TDE, se debe realizar lo siguiente:

- Identificar la llave maestra:

Una llave maestra sólo se puede utilizar para una sola base de datos a la vez, esta (master key) debe ser creada antes de encriptar las columnas de las tablas, la sintaxis para inicializar la (master key) es:

SQL> alter system set key identified by “password”

Este commando crea el Wallet⁴⁷ y usa ese nombre para encriptarlo basado en las recomendaciones del estándar PKCS#5.

El wallet queda por defecto en la siguiente ruta para Linux y Unix /etc/ORACLE/WALLETS/<usuario dueño de la instalación: oracle>

Lo directorios indicados, deben ser creados por root y deben ser asignados como carpetas del usuario dueño de la instalación.

A parte de poder encriptar las tablas comunes y corrientes (heap organizad) Oracle también puede encriptar tablas externas.

- Apertura y cierre del ORACLE Ballet

El Wallet contiene la Master Encryption Key, que debe ser abierta antes de que la base de datos pueda desencriptar los datos en las tablas, la idea principal es que el Wallet permanezca abierto, ya que un Wallet cerrado implica que no se tiene acceso a los datos en ese momento.

SQL> alter system set encryption wallet open identified by “password”;

Para cerrar el Wallet

SQL> alter system set encryption wallet close identified by “password”;

Si el Wallet se encuentra cerrado, no hay acceso a los datos encriptados, o dicho de otra forma, si no hay Wallet, no hay datos.

- Cambiar la Master Key

⁴⁷ Fuente: <http://www.oracleyyo.com> [Introducción a TDE PAG 4 de 15]

La Master Key puede ser cambiada usando el siguiente comando:

```
SQL> alter system set key identified by "password"
```

El hecho de cambiar la llave maestra implica que se van a reencriptar las (tables Keys) en el diccionario de datos ORACLE.

- Tipos de datos soportados por TDE

Se incluyen: VARCHAR2, NUMBER, RAW, BINARY_DOUBLE, CHAR, NVARCHAR2, BINARY_FLOAT, DATE, NCHAR, SECUREFILES (LOBS)

- Revisión del uso de llaves foraneas

TDE está imposibilitado de encriptar columnas que estén siendo usadas como llaves foráneas, lo cual implica una restricción bastante grande al momento de implementar TDE en cualquier industria. Para identificar aquellas columnas que forman parte de una Foreign Key, se puede utilizar la siguiente consulta:

```
select A.owner, A.table_name, A.column_name, A.constraint_name
from dba_cons_columns A, dba_constraints B
where A.table_name = B.table_name
and A.column_name = 'NOMBRE DE LA COLUMNA'
and B.constraint_type = 'R';
```

- Encriptar columnas con TDE

Para encriptar una columna mediante TDE, se utiliza el siguiente comando:

```
SQL> alter table customers modify (columna encrypt);
```

- Crear tablas con columnas en criptazas

```
SQL> CREATE TABLE ejemplo (
id NUMBER(5) PRIMARY KEY
nombre VARCHAR2(15) NOT NULL,
rut NUMBER(9) ENCRYPT
)
```

- Cambiar la table/Column name

La Table o Column key (es una por tabla independiente de la cantidad de columnas encriptadas), el tamaño de la llave y el algoritmo de encriptación pueden ser modificados de forma independiente ejecutando cualquiera de estos comandos:

```
SQL> alter table nombre_tabla rekey;
```

```
SQL> alter table nombre_tabla rekey using 'AES256';
```

```
SQL> alter table nombre_tabla encrypt using 'AES128';
```

Cambiando el Table/Column key, implica que se van a reasignar y re-encriptar todos los datos en la tabla.

2.7.4. RESPALDOS ENCRYPTADOS Y RMAN

Por motivos de seguridad, los respaldos siempre debiesen tener un grado de respaldo, los backup sets de RMAN⁴⁸ por ende siempre debiesen estar encriptados usando el Oracle Advanced Security. Cualquier backup sets de RMAN puede ser encriptado, no así los backups copies. Esto implica por ende, que cuando se hace un respaldo de datos que ya físicamente estaban respaldados, el resultado final, serán archivos con 2 grados de encriptación.

En todas las operaciones de restauración y recuperación, los datos se desencriptan de forma automática, por ende cumple con todas las normas de seguridad básicas relacionadas con los respaldos.

A pesar de llevar a cabo respaldos con RMAN, para que los datos sean legibles (se puedan abrir y consultar las tablas), se necesita del Wallet, este archivo se debe respaldar como una copia física, si por motivo de alguna pérdida o corrupción, este archivo se pierde, se pierden también todos los datos que han sido encriptados.

⁴⁸ Fuente: <http://www.oracleyyo.com> [Introducción a TDE PAG 4 de 15]

2.7.5. METODOS DE ENCRIPCIÓN DISPONIBLES

Los métodos disponibles de encriptación que provee el TDE son:

- AES 192 (valor por omisión)
- AES 256
- AES 128
- 3DES (2 and 3 keys; 168 bits)

2.7.6. DIFERENCIAS ENTRE EL TDE Y LOS PACKAGES DE ENCRIPCIÓN

La encriptación fue introducida por Oracle en la versión Oracle8i, esta encriptación era llevada a cabo mediante el package DBMS_OBFUSCATION_TOOLKIT.

En Oracle10gr1, aparece un nuevo package , mucho más potente , llamado DBMS_CRYPTO.

Las anteriores APIs sirven para encriptar, pero tienen una gran desventaja, implican cambio de código, toda la encriptación se debe hacer de forma manual y cada vez que se genera una encriptación, se debe guardar la llave de está encriptación para poder desencriptar los datos, el hecho de extraviarla implica inmediatamente la pérdida de información.

A diferencia de los anteriores métodos, el TDE no necesita, bajo ningún punto de vista, la modificación del código existente, sólo la alteración de las columnas de la tabla.

2.7.7. CONSIDERACIONES DE PERFORMANCE Y STORAGE

Una de las cosas a tener en cuenta cuando se implementa TDE es el grado de exigencia extra (OverHead⁴⁹) que puede tener una actividad normal.

El OverHead para la encriptación de columnas apunta al momento de encriptar y

⁴⁹ Fuente: <http://www.oracleyyo.com> [Introducción a TDE PAG 4 de 15]

desencriptar los datos (dato que viaja desde la aplicación a la base de datos y viceversa), se estima que las actividades normales ven incrementado su trabajo en un 5% , este porcentaje puede variar de acuerdo a la cantidad de columnas encriptadas dentro de una tabla, el largo de la columna, etc.

Por lo anterior se debe considerar que solamente un número muy reducido de columnas debiese ser candidata a la encriptación, a mayor seguridad, es menor la performance y a menor seguridad, la performance se ve incrementada.

Oracle extiende un cálculo basado en todas las experiencias recopiladas en años e indica que en un sistema no debiesen existir más de 0.3% de columnas encriptadas. Ya que la performance se ve afectada por el TDE, se recomienda siempre instalar un motor **Oracle10gr2 (10.2.0.4)**, lo anterior se debe a que los algoritmos internos de manejo de TDE se han visto mejorados con respecto a las versiones anteriores.

2.7.8. RESTRICCIONES PARA EL USO DE TDE

- No se puede utilizar en ambientes distribuidos, pues esta última utiliza SQL Asíncronos.
- En arquitecturas de RAC en una base de datos, las llaves tienen que ser creadas a través de las instancias.
- Los utilitarios export e import no pueden trabajar sobre columnas encriptadas,
- solamente lo pueden hacer los utilitarios expdp e impdp , de Datapump
- En el mecanismo de Transportable Tablespace, no se puede utilizar TDE
- Las tablas particionadas, no pueden tener columnas encriptadas
- La encriptación no puede ser realizada en columnas tipo LOB
- Las llaves foráneas en una tabla no pueden ser encriptadas
- Cualquier índice que no sea del tipo B*Tree puede ser encriptado
- Los tipos de objetos (object type) tampoco pueden ser creados en base a una columna encriptada

CAPITULO 3. ANALISIS

3.2. INTRODUCCIÓN

3.2.1. OBJETIVO

El presente apartado tiene como objetivo describir cuales son las especificaciones funcionales, no funcionales y del sistema para la implementación de una aplicación WEB que permita controlar, administrar, gestionar y solucionar oportunamente las actividades del proceso de gestión y control de los recursos físicos y lógicos en la Vicepresidencia de Tecnología - Área Pruebas dentro de la Gerencia de Proyectos y Desarrollo de OTECEL S.A.

3.2.2. AMBITO

Diseñar, desarrollar e implementar el sistema SGCIP (Sistema de Gestión y Control del Inventario de Pruebas)

El SGCIP será una aplicación que funcionará en un entorno WEB que permitirá administrar, controlar y clasificar la información de los recursos físicos y/o lógicos asignados al Área de Pruebas para el uso de los usuarios internos. Esta aplicación ofrecerá soporte a las actividades definidas en el procedimiento interno: "PRTISS0601 Gestión y Control del Inventario de Pruebas.pdf".

3.2.3. ALCANCE

Diseñar, desarrollar e implementar el sistema SGCIP (Sistema de Gestión y Control del Inventario de Pruebas) para cubrir las siguientes funcionalidades.

- Verificar disponibilidad de inventario
- Controlar la asignación de inventario
- Verificar la rotación y reutilización de inventario
- Registrar situación del inventario
- Devolver el Inventario
- Agrupar Inventario
- Confirmar disponibilidad en sistemas

- Identificar estado y funcionalidad del inventario
- Clasificar inventario
- Registrar estado del inventario de Terminales
- Registrar estado del inventario de Simcards

El SGCIP ofrecerá información confiable sobre el estado del inventario de pruebas, así también de procedimientos óptimos y adecuados para los usuarios del sistema.

El SGCIP tendrá la funcionalidad de gestión de claves para los usuarios que utilizarán el sistema, es decir que podrán cambiar su clave de acceso si son usuarios permitidos en el sistema.

El SGCIP contará con reportes que permitirán disponer de información de primera mano para el análisis y control de la gestión que se lleva a cabo y el acceso será a través de perfiles para cumplir los lineamientos de seguridad de información de la compañía.

3.2.4. DEFINICIONES

Del negocio:

- Recursos.- elementos presentes o ausentes que permiten el desarrollo de una actividad en particular, pueden ser dinero, materiales, tecnología, personas que desempeñan un cargo.
- Inventario.- recursos físicos y lógicos; son agrupados físicamente y/o virtualmente en un espacio destinado para el uso de los mismos. Sean estos teléfonos celulares, simcards, pines electrónicos, números celulares.
- Pines.- códigos electrónicos de 16 dígitos que tienen asociado un valor comercial en dólares, son utilizados para realizar incrementos de saldo en las líneas celulares permitidas a través del ingreso de estos códigos mediante un canal de servicio del proveedor.

- Simcards.- recurso tecnológico electrónico que está compuesto por un número de serie y un número de IMEI, puede o no tener un número de celular activo.
- Recursos físicos.- elementos de telecomunicaciones; simcards que pueden o no tener asociado un servicio de voz y/o datos, equipos celulares móviles vigentes comercialmente como son CDMA y GSM.
- Recursos lógicos.- elementos de telecomunicaciones; códigos o series de las simcards utilizadas virtualmente y/o los códigos o series de los equipos celulares móviles utilizados virtualmente.
- Líneas.- producto de usuario final, puede tener un servicio de voz y/o datos. Consta de una simcard para el caso de la tecnología GSM y un ESN para el caso de la tecnología CDMA.

Del sistema:

- SGCIP.- Sistema de Gestión y Control del Inventario de Pruebas, en donde se hace referencia el proceso de gestión y recursos de pruebas.
- AMPS.- por sus siglas, sistema telefónico móvil avanzado. Es un sistema de telefonía móvil de primera generación desarrollado por los laboratorios BELL.
- ESN.- por sus siglas, número de serie electrónico asociado a equipos celulares CDMA para acceder a una red AMPS.
- GSM.- por sus siglas sistema global para comunicaciones móviles, tecnología de segunda generación permite que los usuarios con estos dispositivos móviles puedan usarlos en todo el mundo.
- IMEI.- por sus siglas, Identidad Internacional de Equipo Móvil es un código pre-grabado en los teléfonos móviles GSM
- SCL.- Sistema comercial Latino Americano por sus siglas, en donde se encuentra la base de clientes de la operadora, cuenta con 3 instancias que son: PREPRODUCCION – PRODUCCION – PROYECTOS.
- Administrar: Acción de agregar, modificar, eliminar y consultar la información de un determinado objeto o persona.

- Usuario: Persona que puede ingresar al SGCIP, a través de un proceso de autenticación utilizando nombre de usuario y contraseña.
- Permiso: Parámetro que especifica si su poseedor dispone de acceso a una determinada función del sistema o a una parte de la interfaz de usuario del sistema.
- Rol: Es un conjunto de permisos que puede asignarse a un usuario.
- Administrador del Sistema: Persona encargada de ofrecer el soporte técnico y operativo al SGCIP.
- Pruebas: Proceso mediante el cual se realizan actividades para verificar la óptima funcionalidad del sistema.

De tecnología:

- WEB.- sistema de documentos interconectados por enlaces de hipertexto.
- Dominio.- Nombre base que agrupa a un conjunto de equipos o dispositivos y que permite proporcionar nombres de equipo más fácilmente recordables en lugar de una dirección IP numérica.
- Servidor WEB.- Es un programa que implementa el protocolo HTTP (hypertext transfer protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas WEB o páginas HTML (hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música.
- URL.- Significa Uniform Resource Locator, es decir, localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos, como documentos e imágenes en Internet, por su localización.
- HTTP.- El protocolo de transferencia de hipertexto (HTTP, HyperText Transfer Protocol) es el protocolo usado en cada transacción de la Web (WWW).
- Enlaces.- una referencia en un documento de hipertexto a otro documento o recurso.

- Base de Datos.- Es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.
- Aplicación.- Es un programa informático diseñado para facilitar al usuario la realización de un determinado tipo de trabajo.
- Browser o Motor de Navegación.- Es una aplicación que permite al usuario recuperar y visualizar documentos de hipertexto, comúnmente descritos en HTML, desde servidores WEB de todo el mundo a través de Internet.
- DNS.- Es una base de datos distribuida y jerárquica que almacena información asociada de una dirección IP a nombres de dominio en redes como Internet.
- HTML.- HyperText Markup Language, es un lenguaje de marcación
- diseñado para estructurar textos y presentarlos en forma de hipertexto, que es el formato estándar de las páginas web.
- Internet.- Es un método de interconexión descentralizada de redes de computadoras implementado en un conjunto de protocolos denominado TCP/IP.
- ORACLE.- Sistema de gestión de base de datos relacional. Comúnmente utilizados en aplicaciones Web.
- IIS.- Internet Information Server, servidor de páginas WEB para el sistema operativo Microsoft.

3.2.5. REFERENCIAS

Titulo del Documento	Referencia
Standard IEEE 830 – 1998	IEEE, upm 2000
Standard IEEE 830 – 1998	IEEE, Chalmeta 2000 Modelo 1 para la sección 3
PRTISS0601 Gestión y Control del Inventario de Pruebas - 2007	Procedimiento interno de OTECEL.S.A

Tabla 3.1. Referencias de los estándares IEEE830⁵⁰

⁵⁰ Fuente: IEEE830

3.2.6. VISION GENERAL

En el presente documento se encontrará la información acerca de las características del producto de software, interfases del usuario, interfases del sistema, características de los usuarios, descripción de los requerimientos funcionales, no funcionales, los cuales serán representados en las siguientes fichas diseñadas para la compañía.

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF: Funcional RFN: No funcional RI: Interfaz	Nombre del requerimiento	Fecha de especificación	Importancia del requerimiento, codificada como: Crítica, alta, media y baja.	
Descripción	Descripción del requerimiento			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Datos	Objeto	Acción Notificación	Acción Notificación	Pre requisitos para el éxito
Proceso	Descripción de las actividades realizadas en esta especificación.			
Efecto	Efectos generados a otros proceso o sistemas, si es el caso			

Tabla 3.2. Plantilla especificación de requerimientos⁵¹

3.2. DESCRIPCION GENERAL

3.2.1. PERSPECTIVA DEL PRODUCTO

El SGCIP debe permitir organizar, almacenar y administrar la información de manera óptima y eficiente, de tal forma que todo recurso nuevo o existente sea clasificado y actualizado para su posterior uso y distribución. Además debe proveer datos relevantes y de interés para los usuarios, como el inventario

⁵¹ ROBERTO CARLOS ZUMBA NARANJO – DISEÑO DE SISTEMA DE GESTION Y CONTROL INVENTARIO DE PRUEBAS - 2012

actual, disponible, reportes de uso por insumo, historia de uso por insumo, usuario.

EL SGCIP debe interactuar con las bases de datos de pruebas y de desarrollo para obtener la información de la situación de las líneas celulares, excluyendo de las consultas aquellas líneas que se encuentren canceladas.

3.2.2. FUNCIONES DEL PRODUCTO

Las funciones que debe realizar el producto se relacionan con el proceso realizado a la asignación de recursos físicos y/o lógicos. El mismo que se describe a continuación:

- Verificar disponibilidad de inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde obtiene información del estado o situación de los insumos a su cargo.
- Controlar la asignación de inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde realiza actividades de control como son; fechas vencidas, devolución de recursos tecnológicos obsoletos, compra de nuevos recursos, validación del estado funcional de los recursos tecnológicos.
- Verificar la rotación y reutilización de inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde se realiza una actualización periódica de los recursos tecnológicos registrados en el sistema.
- Registrar situación de inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde se actualiza masivamente o uno por uno los recursos tecnológicos a su cargo en base a parámetros como estado, usuario anterior, fecha de entrega.
- Devolver el Inventario.- proceso que es realizado por un recurso humano con el perfil de usuario solicitante del inventario. En donde

realiza la entrega física o virtual de los recursos tecnológicos a él asignados en un plazo establecido.

- Agrupar Inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde toma el inventario físico de los equipos celulares separándolos por tecnología de comunicación y toma el inventario físico de simcards separándolos por los ambientes del sistema comercial.
- Confirmar disponibilidad en sistemas.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde verifica el estado de los recursos tecnológicos en las bases de datos de los sistemas vigentes en la compañía.
- Identificar estado y funcionalidad del inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde verifica uno por uno el estado funcional de los insumos tecnológicos cuando realiza el proceso de recepción de recursos.
- Registrar estado del inventario de Terminales.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde se reciben los recursos tecnológicos y se realizan las actas respectivas en donde los usuarios dejan por escrito la entrega – recepción de los recursos tecnológicos.
- Registrar estado del inventario de Simcards.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde se reciben los recursos tecnológicos y se realizan las actas respectivas en donde los usuarios dejan por escrito la entrega – recepción de los recursos tecnológicos.

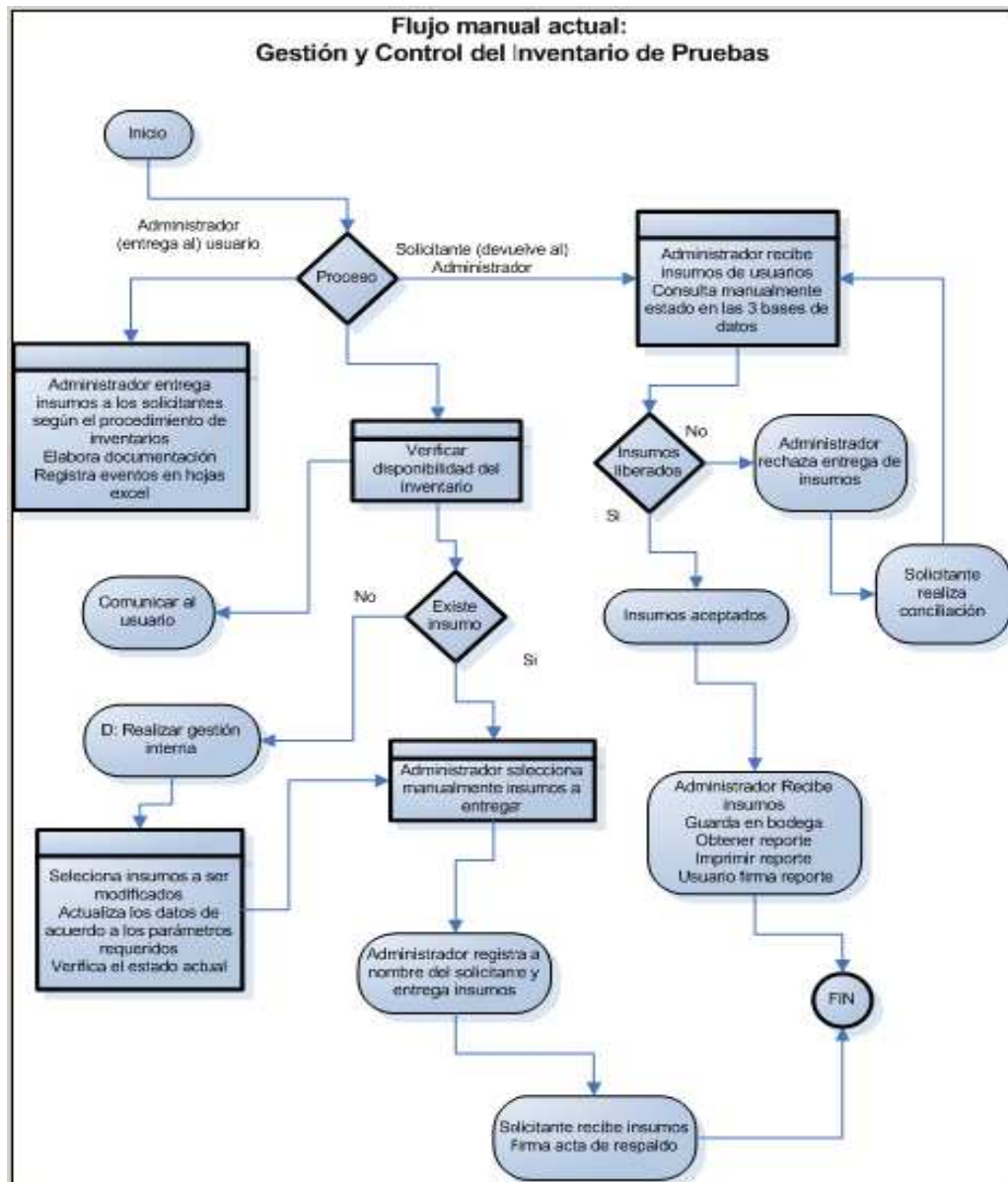


Figura 3.1. "Proceso manual de gestión y control de inventarios pruebas"⁵²

3.2.2.1 ACTIVIDADES PARA LA ADMINISTRACION DE LOS RECURSOS FISICOS Y LOGICOS DEL AREA DE CONSTRUCCION Y PRUEBAS TI.

⁵² PRTISS0601Gestión y Control del Inventario de Pruebas.doc

No.	ACTIVIDAD	DESCRIPCION	RESPONSABLE
01	Verificar disponibilidad de inventario	Verificar periódicamente la disponibilidad de los recursos físicos (simcards y terminales) y lógicos (numeración, pines electrónicos y ajustes desde BOWEB) para pruebas conforme con las políticas del presente documento.	Administradores de inventarios
02	Gestionar nuevo inventario	Solicitar, gestionar y recibir los nuevos recursos físicos y lógicos para pruebas.	Administradores de inventarios
03	Controlar la asignación de inventario	Controlar la asignación de recursos a los Ingenieros de Construcción y los Ingenieros de Construcción - de Pruebas.	Administradores de inventarios
04	Verificar la rotación y reutilización de inventario	Verificar la reutilización y reasignación de recursos físicos y lógicos a los Ingenieros de Construcción – Pruebas.	Administradores de inventarios
05	Almacenar Inventario	Almacenar el inventario de pruebas en el espacio físico destinado para este fin asegurando su correcta ubicación y protección.	Administradores de inventarios
06	Registrar situación de inventario	Mantener y controlar los registros de recepción, asignación, verificación y almacenamiento de los recursos físicos y lógicos de pruebas.	Administradores de inventarios

Tabla 3.3. Actividades para la administración de los recursos de pruebas (numeral 6.1).⁵³

3.2.2.2 ACTIVIDADES PARA LA VERIFICACIÓN DE LA OPERATIVIDAD Y CONSUMOS DEL INVENTARIO DE PRUEBAS

⁵³ PRTISS0601 Gestión y Control del Inventario de Pruebas.doc

No.	ACTIVIDAD	DESCRIPCIÓN	RESPONSABLE
01	Agrupar Inventario	Tomar el inventario de los equipos celulares separándolos por tecnología de comunicación.	Administrador de inventario (Terminales)
02	Agrupar Inventario	Tomar el inventario de simcards separándolos por los ambientes del sistema comercial.	Administrador de inventario (Simcards)
03	Realizar prueba de encendido	Ejecutar las pruebas de encendido y enlace a la señal de los equipos celulares y simcards	Administrador de inventario (Terminales / Simcards)
04	Realizar prueba de funcionalidad	Ejecutar la llamada de prueba de comunicación a los servicios gratuitos de soporte al cliente para verificar el enlace.	Administrador de inventario (Terminales / Simcards)
05	Confirmar disponibilidad en sistemas	Verificar el estado de los equipos celulares y Simcards en los sistemas de gestión comercial vigentes.	Administrador de inventario (Terminales / Simcards)
06	Identificar estado y funcionalidad del inventario	<p>Determinar el estado del equipo celular y/o Simcards considerando las siguientes categorías:</p> <p>Equipos Celulares:</p> <p>a) Funciona: Cuando el equipo ha respondido adecuadamente en la comunicación y está hábil en el sistema de gestión comercial vigente.</p> <p>b) No Funciona: Cuando el equipo no permitió el enlace adecuado para la comunicación y/o presenta fallos de conectividad durante la comunicación y/o está inhábil en el sistema de gestión comercial vigente.</p> <p>Simcards:</p> <p>a) Dañada: Cuando la Simcard no supera el paso de PIN y PUK que permita el enlace.</p> <p>b) Activa: Cuando la Simcard permita la comunicación sin inconvenientes.</p> <p>c) Suspendida: Cuando la Simcard recibe el mensaje de suspensión.</p> <p>d) Dada de Baja: Cuando la Simcard no permita la comunicación y se verifique en el sistema de gestión comercial dada de</p>	Administrador de inventario (Terminales / Simcards)

No.	ACTIVIDAD	DESCRIPCIÓN	RESPONSABLE
		baja. e) Disponible / Libre: Cuando la Simcard no permita la comunicación y se verifique en el sistema de gestión comercial en estado disponible para asignar.	
07	Clasificar inventario	Clasificar el inventario no válido para la ejecución de las pruebas del disponible para el uso en pruebas.	Administrador de inventario (Terminales / Simcards)
08	Registrar estado del inventario de Terminales	Registrar el inventario disponible para el uso de las pruebas en el formato "Inventario de Terminales" detallando lo siguiente: a) Tecnología de comunicación, No. de Serie / IMEI y modelo. b) Estado de la última verificación. c) Fecha de la última verificación. d) Responsable de la última verificación.	Administrador de inventario (Terminales)
09	Registrar estado del inventario de Sincards	Registrar el inventario disponible para el uso de las pruebas en el formato "Inventario de Simcards" detallando por cada ambiente del Sistema Comercial lo siguiente: a) ICC, Plan, Fecha de Alta, Fecha de Baja, Código de cliente. b) Estado de la última verificación. c) Fecha de la última verificación. d) Responsable de la última verificación.	Administrador de inventario (Simcards)
10	Almacenar e identificar inventario	Etiquetar el inventario no válido de manera legible y ubicarlo en el espacio físico destinado para la baja del mismo.	Administrador de inventario (Terminales / Simcards)

No.	ACTIVIDAD	DESCRIPCIÓN	RESPONSABLE
01	Coordinar la ejecución de Notas de Crédito y/o Ajustes	Una vez concluidas las pruebas y siempre que amerite, elaborar los formularios de ajustes por facturas en el ambiente de producción, previo a la validación de la información contenida en las facturas y coordinar la correspondiente Nota de crédito y/o Ajuste	Ingeniero de Construcción – Pruebas

No.	ACTIVIDAD	DESCRIPCIÓN	RESPONSABLE
02	Devolver el Inventario	Siempre que hayan concluido las pruebas, devolver los recursos para su reutilización, estos son: equipos celulares, pines electrónicos, números y simcards del inventario de pruebas.	Ingeniero de Construcción – Pruebas
03	Reposición de Simcards	<p>Las simcards asignadas al Ingeniero de Construcción – Pruebas que no consten en su poder, deben ser repuestas al inventario. Los pasos a seguir son:</p> <ul style="list-style-type: none"> a) Dar de baja a la simcard en todos los ambientes b) Para simcards libres: Solicitar en el formato “SOLICITUD DE EGRESO DE INVENTARIO Y DE ACTIVACION DE PLANES COMERCIALES Y NO COMERCIALES .xls” (Anexo H) a logística la asignación de una nueva simcard libre en el que se especifica que se debe facturar al Ing. de Pruebas c) Para simcards prepago o postpago: Comprar una simcard prepago en el CAV para proceder con la reposición d) Realizar una Acta de Entrega – Recepción indicando que se realiza la reposición de la simcard extraviada por la nueva simcard e) Ingreso en el inventario de la nueva simcard f) Baja del inventario de la simcard extraviada 	<p>Ingeniero de Construcción – Pruebas</p> <p>Administrador de inventario (Simcards)</p>
04	Reposición de Terminales	<p>Los terminales asignados al Ingeniero de Construcción – Pruebas que no consten en su poder, deben ser repuestos al inventario. Los pasos a seguir son:</p> <ul style="list-style-type: none"> g) Dar de baja al terminal en todos los ambientes h) Reponer un terminal de las mismas características al extraviado, en 	Ingeniero de Construcción – Pruebas

No.	ACTIVIDAD	DESCRIPCIÓN	RESPONSABLE
		<p>caso de no existir el mismo modelo en el mercado, devolver un terminal de similares características al extraviado</p> <p>i) Realizar una Acta de Entrega – Recepción indicando que se realiza la reposición del terminal extraviado por el nuevo terminal</p> <p>j) Ingreso en el inventario del nuevo terminal</p> <p>k) Baja del inventario del terminal extraviado</p>	Administrador de inventario (Terminales)

Tabla 3.4. Actividades para la verificación de la operatividad y consumos del inventario de pruebas (numeral 6.3)⁵⁴

3.2.3. CARACTERISTICAS DEL USUARIO

El SGCIP contendrá cuatro tipos de usuarios: Solicitante, Administrador de Recursos, Jefe de Recursos y Administrador del Sistema.

Cada usuario tendrá un perfil específico para su correcta interacción con el sistema.

Solicitante: Su rol en el sistema es interactuar como visitante permitido, son los usuarios que inician los procesos de solicitar o entregar recursos del inventario. Son usuarios que tienen conocimientos básicos en el manejo de páginas WEB.

El perfil para este usuario debe tener las siguientes funcionalidades:

- Consultar.- para conocer el estado de las peticiones o solicitudes creadas y el detalle de los recursos físicos y/o lógicos a él asignados.
- Crear.- llenar un formulario en donde especifique la necesidad.

Administrador del Inventario: Su rol en el sistema es cumplir con el proceso de gestión de recursos.

Son usuarios que tienen conocimientos intermedios en el manejo de páginas WEB y paquetes informáticos.

⁵⁴ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

El perfil para este usuario debe tener las siguientes funcionalidades:

- Consultar.- estado de las solicitudes, el detalle y estado de los recursos físicos y/o lógicos asignados a todos los usuarios del sistema, el estado de los recursos físicos y/o lógicos disponibles para asignar en el sistema, el estado de los recursos físicos y/o lógicos en el sistema comercial (tres bases de datos).
- Aprobar o negar.- las peticiones de recursos solicitadas por los usuarios internos.
- Asignar.- en el sistema podrá asignar, recibir o reasignar los recursos físicos y lógicos a los usuarios internos.
- Exportar.- el estado actual de los recursos físicos y/o lógicos, es decir una foto del inventario para un posterior análisis.
- Crear o modificar.- simcards, terminales, usuarios.

Jefe del Inventario: Su rol en el sistema es verificar el cumplimiento del proceso de gestión de recursos.

Son usuarios que tienen conocimientos intermedios en el manejo de páginas WEB y paquetes informáticos.

El perfil para este usuario debe tener las siguientes funcionalidades:

- Consultar.- revisar a través de cuadros resumen la información del estado de los recursos físicos y/o lógicos asignados a todos los usuarios del sistema, el estado de los recursos físicos y/o lógicos disponibles para asignar.

Administrador del Sistema de Inventarios: Su rol en el sistema es proporcionar soporte, capacitación, mantenimiento y solución de problemas.

Son usuarios que tienen conocimientos avanzados en el manejo de páginas WEB, paquetes informáticos, bases de datos y programación entre otros.

El perfil para este usuario debe tener las siguientes funcionalidades:

- Crear perfiles.- durante el proceso de pruebas se podría identificar la necesidad de crear nuevos perfiles para la gestión y control.
- Soporte.- a los elementos creados para el sistema.

3.2.4. LIMITACIONES GENERALES

3.2.4.1 ELEMENTOS DE SOFTWARE Y HARDWARE

Los elementos de hardware serán proporcionados por la compañía auspiciante, en donde existe un departamento; Infraestructura TI, quienes proveen los elementos de hardware a través de un proceso interno para cubrir las necesidades internas de los usuarios.

Los elementos de software serán proporcionados por la compañía auspiciante, en donde existe un departamento; Mesa de Ayuda, quienes internamente tienen el control del software y licencias que se maneja en la compañía, proveen el software a través de un proceso interno para cubrir las necesidades internas de los usuarios.

3.2.4.2 LENGUAJE DE PROGRAMACION

El aplicativo para la gestión y control de inventarios será desarrollada en su totalidad sobre la plataforma .NET, utilizando el lenguaje Visual Basic .NET 2005.

Para el sistema de gestión de bases de datos se utilizara la versión 8i de ORACLE.

3.2.4.3 SEGURIDAD DE INFORMACION

La compañía auspiciante dispone de un sistema de autenticación global, llamado IDM (Identity Management⁵⁵) el cual centraliza la gestión de usuarios, claves y perfiles a los aplicativos productivos.

El aplicativo a ser desarrollado, no se ajustará a las políticas de IDM, ya que esto implicaría un desarrollo adicional a través de proveedores externos.

El aplicativo manejará su propio esquema de autenticación de usuarios.

⁵⁵ Sistema de Gestión de Identidades propietaria de la empresa OTECEL. S.A

3.2.4.4 NOTIFICACIONES DE CORREO

El aplicativo no dispondrá de ningún módulo que realice validaciones a nivel de base de datos para enviar notificaciones vía correo electrónico, este tipo de funcionalidades son realizados siguiendo un estándar a través de la línea de desarrollo externo.

3.2.5. SUPUESTOS

El aplicativo será diseñado utilizando los elementos de software para ser ejecutado bajo el sistema operativo WINDOWS, las estaciones de trabajo de los usuarios dispondrán del navegador (BROWSER) Microsoft Internet Explorer 6 o superior para la correcta ejecución del aplicativo.

El aplicativo debe ser instalado en un servidor que disponga del software Internet Información Service 5 o superior.

Los equipos o estaciones de trabajo de los usuarios deberán estar correctamente estandarizadas y conectadas al dominio de la compañía, además deben tener acceso a las bases de datos y aplicativos involucrados en el proceso de gestión de inventarios. Esto incluye también las impresoras.

3.3 REQUISITOS ESPECIFICOS

3.3.1. REQUISITO FUNCIONAL

3.3.1.1. INGRESO AL SISTEMA

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3311.1	Autenticar usuario	7-Sep-2011	Crítica	
Descripción	Acceso al sistema por medio del explorador de Internet.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
URL	Red	Pantalla para ingresar el	Mensajes de error tipo HTTP	El ingreso lo realiza con un

	usuario/clave	perfil asignado en el sistema
Proceso	El sistema deberá tener una URL la misma que al ser digitada en la barra del navegador permita el ingreso.	
Efecto	N/A	

Tabla 3.5. Especificación de requerimiento: Autenticar usuario⁵⁶

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3311.2	Modificar clave	7-Sep-2011	Alta	
Descripción	Los usuarios ya creados en el sistema podrán tener la posibilidad de cambiar su clave de acceso.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Usuario	Formulario de cambio clave	El sistema presenta mensaje de cambio clave exitosa.	Manejo de errores definidos para el aplicativo	El ingreso lo realiza con un perfil asignado en el sistema
Proceso	El usuario selecciona la opción de cambio de clave, ingresa el usuario y clave actual; ingresa la nueva clave y selecciona grabar.			
Efecto	N/A			

Tabla 3.6. Especificación de requerimiento: Modificar clave⁵⁷

3.3.1.2. USUARIO SOLICITANTE

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A			
Código	Nombre	Fecha	Grado Necesidad
RF_3312.1	Crear solicitudes	7-Sep-2011	Crítica
Descripción	El sistema debe permitir al usuario y/o administrador del		

⁵⁶ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

⁵⁷ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

	inventario ingresar los datos necesarios para crear una solicitud de recursos físicos y/o lógicos.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Usuario solicitante Tipo de recurso Cantidad de recursos Fecha devolución Justificación Ambiente de trabajo	Formulario solicitud de recursos	Presenta al usuario el número de solicitud	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario de solicitud de recursos, ingresa la información requerida y presiona un botón grabar.			
Efecto	El sistema despliega en pantalla el número de solicitud creada.			

Tabla 3.7. Especificación de requerimiento: Crear solicitudes⁵⁸

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3312.2	Consultar solicitudes	7-Sep-2011	Media	
Descripción	El sistema debe proporcionar al usuario un listado con el estado de las solicitudes por él creadas.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Fecha inicio Fecha fin	Formulario reporte de solicitudes	Presenta al usuario una lista con la información definida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario reporte de solicitudes creadas; por defecto se muestra las últimas 10 solicitudes. Puede ingresar una fecha inicio y fecha fin en donde se busca por fecha de solicitud creada, la información histórica se debe obtener hasta 90 días atrás.			
Efecto	El sistema despliega en pantalla una tabla con: el número de solicitud, fecha de creación, estado (pendiente, cancelado y cerrado) y notas.			

Tabla 3.8. Especificación de requerimiento: Consultar solicitudes⁵⁹

⁵⁸ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3312.3	Reporte de insumos	7-Sep-2011	Alta	
Descripción	El sistema debe proporcionar al usuario un listado con el detalle de los recursos físicos y lógicos a él asignados.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Número de solicitud Fecha inicio – Fecha fin	Formulario insumos asignados	Presenta al usuario una lista con la información definida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario reporte de insumos; por defecto se muestran todos los insumos del usuario o ingresa un número de solicitud en donde se busca todos los insumos asignados a ese código de solicitud. Si el usuario que selecciona el reporte es coordinador o administrador podrá seleccionar los insumos por cualquier usuario del sistema.			
Efecto	El sistema despliega en pantalla una tabla con: el secuencial, tipo de insumo, código de insumo, fecha de asignación, fecha devolución, justificación y notas			

Tabla 3.9. Especificación de requerimiento: Reporte de insumos⁶⁰

3.3.1.3. USUARIO ADMINISTRADOR DEL INVENTARIO

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.1	Asignar insumos	7-Sep-2011	Crítica	
Descripción	El sistema debe proporcionar al usuario la posibilidad de asignar recursos físicos y/o lógicos			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones

⁵⁹ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

⁶⁰ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Número de solicitud Código de insumo	Formulario asignar recursos	Presenta al usuario mensaje de confirmación exitosa del proceso.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario asignar recursos, en donde asigna los códigos de insumos a una solicitud cuyo estado sea igual a pendiente. El usuario recibe los recursos y firma un documento con el detalle de lo recibido.			
Efecto	N/A			

Tabla 3.10. Especificación de requerimiento: Asignar recursos⁶¹

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre		Fecha	Grado Necesidad
RF_3313.2	Procesar solicitudes		7-Sep-2011	Crítica
Descripción	El sistema debe proporcionar al usuario un listado con las solicitudes cuyo estado sean iguales a pendiente; podrá cambiar el estado ha cerrado o cancelado.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Número de solicitud Usuario solicitante Fecha de creación	Formulario solicitudes por procesar	Presenta al usuario una lista con la información definida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario solicitudes por procesar, es decir cuyo estado es pendiente; por defecto se muestra las últimas 10 solicitudes o ingresa una fecha de creación o número de solicitud o usuario solicitante.			
Efecto	El sistema despliega en pantalla una tabla con: número de solicitud, usuario solicitante, fecha de creación, tipo de insumo, cantidad, ambiente y justificación.			

Tabla 3.11. Especificación de requerimiento: Procesar solicitudes⁶²

⁶¹ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

⁶² PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.3	Crear usuarios	7-Sep-2011	Alta	
Descripción	El sistema debe proporcionar al usuario la posibilidad de crear usuarios para ingresar al sistema.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Cédula Nombre Apellido Usuario de red Proveedor Fecha de creación Perfil Contacto Correo	Formulario creación de usuarios	Presenta al usuario un mensaje de creación exitosa.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario creación de usuarios, ingresa todos los datos requeridos y selecciona grabar.			
Efecto	N/A			

Tabla 3.12. Especificación de requerimiento: Crear usuarios⁶³

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.4	Modificar usuarios	8-Sep-2011	Alta	
Descripción	El sistema debe proporcionar al usuario la posibilidad de modificar usuarios ya creados en el sistema.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Cédula Usuario de red	Formulario modificar usuarios	Presenta al usuario un mensaje de modificación	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está

⁶³ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

		exitosa.		definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario modificación de usuarios, ingresa todos los datos requeridos y selecciona grabar. En este formulario es posible asignar una clave por defecto, suspender o habilitar un usuario pero no eliminarlo.			
Efecto	N/A			

Tabla 3.13. Especificación de requerimiento: Modificar usuarios⁶⁴

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.5	Extender solicitudes	8-Sep-2011	Baja	
Descripción	El sistema debe proporcionar al usuario la posibilidad de aplicar una prórroga o alargue de la fecha de entrega a una solicitud.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Número de solicitud	Formulario solicitudes por procesar	Presenta al usuario un mensaje de prórroga aplicada exitosa.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario solicitudes por procesar, busca una solicitud y le aplica una prórroga o alargue a la fecha de devolución, le suma un valor de hasta 30 días. Se podrá aplicar una sola prórroga por solicitud, y debe estar en estado cerrado.			
Efecto	N/A			

Tabla 3.14. Especificación de requerimiento: Extender solicitudes.⁶⁵

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A			
Código	Nombre	Fecha	Grado Necesidad

⁶⁴ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

⁶⁵ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

RF_3313.6	Re abrir solicitudes	8-Sep-2011	Baja
Descripción	El sistema debe proporcionar al usuario la posibilidad de re abrir una solicitud.		
Entradas	Fuente	Salida-Éxito	Salida-Fracaso
Número de solicitud	Formulario solicitudes por procesar	Presenta al usuario un mensaje de solicitud abierta.	Manejo de errores definidos para el aplicativo
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario solicitudes por procesar, busca una solicitud y le cambia el estado de cerrado a pendiente. Las solicitudes canceladas no pueden ser re abiertas.		
Efecto	N/A		

Tabla 3.15. Especificación de requerimiento: Re abrir solicitudes⁶⁶

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A			
Código	Nombre	Fecha	Grado Necesidad
RF_3313.7	Ingresar nuevos insumos	8-Sep-2011	Alto
Descripción	El sistema debe proporcionar al usuario la posibilidad de ingresar nuevos insumos al sistema.		
Entradas	Fuente	Salida-Éxito	Salida-Fracaso
Tipo de insumo Serie del insumo Código - caja Modelo Tecnología Central Tipo de uso Ambiente Estado Fecha de creación Fecha de actualización Notas	Formulario nuevos insumos	Presenta al usuario un mensaje de insumos creados correctamente.	Manejo de errores definidos para el aplicativo
			La disponibilidad de este formulario está definida por un perfil.

⁶⁶ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Proceso	El usuario permitido ingresa al sistema y selecciona el formulario nuevos insumos, ingresa los datos requeridos y guarda la información en el sistema.
Efecto	N/A

Tabla 3.16. Especificación de requerimiento: Ingresar nuevos insumos⁶⁷

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.8	Actualizar datos de insumos	9-Sep-2011	Alto	
Descripción	El sistema debe proporcionar al usuario la posibilidad de realizar modificaciones a los insumos ingresados en el sistema.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Tipo de uso Ambiente Estado Fecha de actualización Notas Caja Central Usuario Solicitud	Formulario mantenedor de insumos	Presenta al usuario un mensaje de insumos modificados correctamente.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario mantenedor de insumos, modifica los datos requeridos y guarda la información en el sistema. También debe ser posible eliminar un insumo.			
Efecto	N/A			

Tabla 3.17. Especificación de requerimiento: Mantenedor de insumos⁶⁸

⁶⁷ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

⁶⁸ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.9	Devolución de insumos.	9-Sep-2011	Alto	
Descripción	El sistema debe proporcionar al usuario un reporte con la información de las devoluciones de insumos.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Fecha inicio Fecha fin Usuario Solicitud	Formulario reporte devolucion es por usuario	Presenta al usuario la información requerida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario reporte devolución de insumos, ingresa los parámetros de entrada (Fecha y Usuario) y obtiene la información: Secuencial, tipo de insumo, serie, solicitud, fecha asignación, fecha devolución prevista, notas.			
Efecto	N/A			

Tabla 3.18. Especificación de requerimiento: Reporte devolución de insumos.⁶⁹

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.10	Situación actual del inventario.	9-Sep-2011	Alto	
Descripción	El sistema debe proporcionar al usuario un reporte en Excel con todos los insumos que existen en el sistema.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
N/A	Formulario foto del inventario	Presenta al usuario la información requerida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.

⁶⁹ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Proceso	El usuario permitido ingresa al sistema y selecciona el formulario foto del inventario y selecciona exportar a Excel. En donde guarda o abre el archivo el mismo que contiene la información de los insumos existentes en el sistema: Insumo, Caja, Central, Ambiente, Fecha de creación, Situación, Modelo, Tecnología, Fecha de revisión, Responsable revisión, Notas.
Efecto	N/A

Tabla 3.19. Especificación de requerimiento: Reporte situación actual del inventario.⁷⁰

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.11	Reporte de los insumos activos.	12-Sep-2011	Medio	
Descripción	El sistema debe proporcionar al usuario un cuadro con el detalle de todos aquellos insumos que existen en el sistema comercial de producción.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Series	Formulario líneas activas producción	Presenta al usuario la información requerida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario reporte del estado actual de los insumos en los ambientes. En donde se presenta una tabla con la siguiente información: CAJA, SERIE, ABONADO, CELULAR, CLIENTE, CUENTA, CODIGO PLAN, DESCRIPCION, PLAN, CICLO, FECHA ALTA, IMEI			
Efecto	N/A			

Tabla 3.20. Especificación de requerimiento: Reporte de insumos activos.⁷¹

⁷⁰ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

⁷¹ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.12	Auditoria por insumo	12-Sep-2011	Alto	
Descripción	El sistema debe proporcionar al usuario un detalle de los movimientos realizados sobre un insumo.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Series	Formulario reporte auditoria por insumo	Presenta al usuario la información requerida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario reporte auditoria por insumo y exporta en Excel el detalle de las transacciones de ese insumo: Insumo, Serie, Caja, Central, Ambiente, Fecha de creación, Estado, Modelo, Tecnología, Fecha de revisión, Responsable, Notas, Fecha de evento, Tipo de evento (creación, eliminación, modificación.)			
Efecto	N/A			

Tabla 3.21. Especificación de requerimiento: Consultar transacciones.⁷²

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3313.13	Consultar existencias.	12-Sep-2011	Alto	
Descripción	El sistema debe proporcionar al usuario un detalle totalizado de los insumos asignados y disponibles.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Series	Formulario Stock actual de	Presenta al usuario la información	Manejo de errores definidos para	La disponibilidad de este

⁷² PRTISS0601Gestión y Control del Inventario de Pruebas.doc

	insumos	requerida.	el aplicativo	formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario stock de insumos y visualiza la siguiente información: Tipo, ambiente, estado, cantidad.			
Efecto	N/A			

*Tabla 3.22. Especificación de requerimiento: Consultar existencias.*⁷³

3.3.1.4. JEFE DEL INVENTARIO

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3314.1	Situación actual del inventario.	9-Sep-2011	Alto	
Descripción	El sistema debe proporcionar al usuario un reporte en Excel con todos los insumos que existen en el sistema.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
N/A	Formulario foto del inventario	Presenta al usuario la información requerida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario foto del inventario y selecciona exportar a Excel. En donde guarda o abre el archivo el mismo que contiene la información de los insumos existentes en el sistema: Insumo, Caja, Central, Ambiente, Fecha de creación, Situación, Modelo, Tecnología, Fecha de revisión, Responsable revisión, Notas.			
Efecto	N/A			

*Tabla 3.23. Especificación de requerimiento: Reporte situación actual del inventario.*⁷⁴

⁷³ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

⁷⁴ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3314.2	Reporte de existencias.	12-Sep-2011	Alto	
Descripción	El sistema debe proporcionar al usuario un detalle totalizado de los insumos asignados y disponibles.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Series	Formulario Stock actual de insumos	Presenta al usuario la información requerida.	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario stock de insumos y visualiza la siguiente información: Solicitante, tipo, ambiente, estado, cantidad.			
Efecto	N/A			

Tabla 3.24. Especificación de requerimiento: Reporte de existencias.⁷⁵

3.3.1.5. ADMINISTRADOR DEL SISTEMA

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3315.1	Crear perfiles	12-Oct-2011	Alto	
Descripción	El sistema debe proporcionar al usuario la posibilidad de crear los perfiles para el acceso a los formularios del sistema.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Perfiles Usuarios Formularios	Formulario mantenedor de perfiles	Presenta el formulario de usuario	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un

⁷⁵ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

				perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario mantenedor de perfiles y modifica o crea los perfiles de acuerdo a los requerimientos del área.			
Efecto	N/A			

Tabla 3.25. Especificación de requerimiento: Crear perfiles.⁷⁶

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RF_3315.2	Consultar perfiles	12-Oct-2011	Alto	
Descripción	El sistema debe proporcionar al usuario la posibilidad de consultar los perfiles existentes en el sistema.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Perfiles Usuarios Formularios	Formulario mantenedor de perfiles	Presenta el formulario de usuario	Manejo de errores definidos para el aplicativo	La disponibilidad de este formulario está definida por un perfil.
Proceso	El usuario permitido ingresa al sistema y selecciona el formulario mantenedor de perfiles y consulta o crea los perfiles de acuerdo a los requerimientos del área.			
Efecto	N/A			

Tabla 3.26. Especificación de requerimiento: Consultar perfiles.⁷⁷

3.3.2 REQUISITOS DE INTERFACES EXTERNAS

3.3.2.1 INTERFACES DE USUARIO

El usuario podrá interactuar con el sistema por medio de formularios ASPX publicados en un servidor de páginas WEB.

⁷⁶ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

⁷⁷ PRTISS0601Gestión y Control del Inventario de Pruebas.doc

Estos formularios serán diseñados tomando lineamientos de desarrollo de la compañía auspiciante.

Sistema de Gestión y Control del Inventario de Pruebas (SGCIP) Especificación de Requerimientos para OTCEL S.A				
Código	Nombre	Fecha	Grado Necesidad	
RI_3321.1	Vínculos de navegación	7-Sep-2011	Crítica	
Descripción	El sistema debe presentar o contener vínculos de navegación entendibles para el usuario.			
Entradas	Fuente	Salida-Éxito	Salida-Fracaso	Restricciones
Vínculos - Textos – Iconos – botones	Página principal (masterpage) páginas de gestión (treeview)	Formulario de usuario	Mensajes de error definidos para el aplicativo	La disponibilidad de formularios está definido por un perfil asignado en el sistema
Proceso	El sistema presenta al usuario los formularios sobre los cuales puede tomar acción.			
Efecto	El usuario puede navegar de una manera fácil y dirigida.			

Tabla 3.27. Especificación de requerimiento: Vínculos de navegación⁷⁸

3.3.2.2 INTERFACES DE HARDWARE

Esta definido por la compañía auspiciante y consta de características predefinidas.

Se adjunta las características por tipo:

- Usuario Portátil:

⁷⁸ PRTISS0601Gestión y Control del Inventario de Pruebas.doc



Figura 3.2. Propiedades estación de trabajo portátil ⁷⁹
Fuente: Empresa auspiciante.

- Usuario Escritorio:

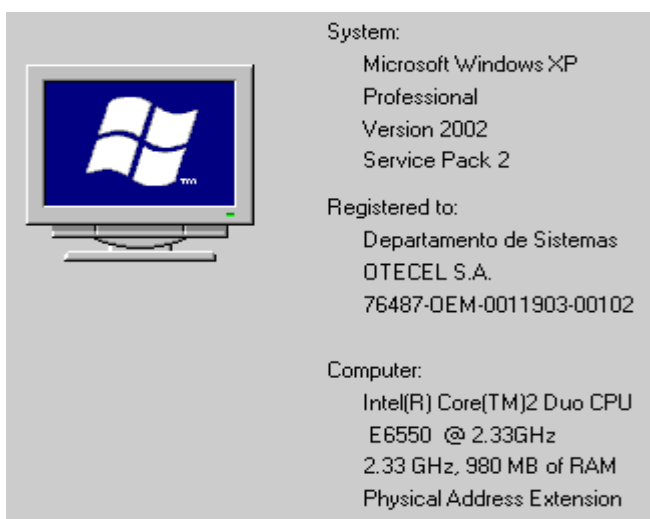


Figura 3.3. Propiedades estación de trabajo escritorio ⁸⁰

- Servidor Virtual:

⁷⁹ Fuente: Empresa auspiciante OTECEL. S.A

⁸⁰ Fuente: Empresa auspiciante OTECEL. S.A



Figura 3.4. Propiedades servidor virtual⁸¹

3.3.2.3 INTERFACES DE SOFTWARE

Está definido por la compañía auspiciante y consta de características predefinidas, como son:

- Versión en Ingles
 - Windows XP / 7
 - Internet Explorer 6 / 7
 - Oracle 8i
 - Microsoft office 2003
 - Sisci_x
 - Kenan_FX
 - Quick Test Profesional
 - McAfee
 - Sistema comercial Latino Americano
 - Acrobat 9

3.3.2.4 INTERFACES DE COMUNICACIÓN

⁸¹ Fuente: Empresa auspiciante OTECEL. S.A

Se tiene previsto que el aplicativo interactúe con un elemento externo, scanner de código de barras.

3.3.3 REQUISITOS DE EJECUCION

Para que los usuarios internos puedan acceder al aplicativo, este tiene que estar correctamente instalada en el servidor WEB y operativo la base de datos.

Al ser un aplicativo que se ejecutará dentro de la INTRANET de la empresa, no es requerida una conexión a Internet o banda ancha.

Para que los usuarios externos puedan acceder al aplicativo, deben tener acceso a los aplicativos de la INTRANET de la empresa, y además es requerida una conexión directa tipo VPN con los permisos y reglas correctamente aplicadas.

3.3.4 LIMITACIONES DE DISEÑO

El sistema no dispondrá de políticas de administración de claves, es decir que el sistema no realizará acciones controlará intentos fallidos de ingreso, caducidad de claves, notificaciones automáticas, solicitud automatizada de claves.

El sistema no contará con reportes personalizados, que permitan realizar consultas avanzadas para desplegar información en línea y/o con campos o columnas parametrizables.

3.3.5 ATRIBUTOS

3.3.5.1 SEGURIDAD

El aplicativo manejará un modulo de autenticación de usuarios independiente, sin apegarse a las políticas de seguridad de la compañía.

3.3.5.2 MANTENIBILIDAD

Por determinar.

3.3.6 OTROS REQUISITOS

3.3.6.1 BASES DE DATOS

En la base de datos se debe almacenar la información relacionada con:

- Solicitudes de usuario
- Terminales o equipos celulares
- Simcards
- Modelos de terminales
- Usuarios
- Información histórica de las transacciones de los terminales y simcards

3.3.6.2 OPERACIONES

El aplicativo se inicia creando los usuarios, configurando los perfiles y realizando el llenado de las tablas de simcards y terminales con la información provista por el coordinador de recursos.

Posteriormente los usuarios crean las solicitudes, las mismas que son revisadas por el coordinador de recursos quien ejecuta el proceso de asignación de recursos físicos y lógicos.

Entregable MSF "Visión Alcance Sistema de Gestión Inventario Pruebas"

CAPITULO 4. PROPUESTA

3.2. DISEÑO

4.1.1. ELEMENTOS DE NAVEGACION

4.1.1.1 ESTRUCTURA GENERAL DEL APLICATIVO

Entre los elementos más relevantes que conforman el sistema de navegación se cuentan los siguientes:

- Menú General: permite el acceso a cada una de las funcionalidades de sistema, se ubicará en la parte lateral izquierda de la página maestra.
- Pie de Página: ubicado en la parte inferior de la página maestra, indicará el nombre de la institución, teléfonos, dirección física y de correo electrónico.
- Barra Corporativa: ubicado en la parte superior de la página maestra, muestra el logotipo de la empresa, versión del aplicativo, usuario conectado y fecha del sistema.
- Principal: Ubicado en la parte derecha de la página maestra, en donde se desplegará la información de los elementos de las funcionalidades seleccionadas en el menú general.

4.1.1.2 MODULOS

Para cubrir el alcance del proyecto, se crean tres módulos para el menú general:

- Administración
 - Crear usuarios
 - Crear perfiles
 - Mantenedor de insumos
 - Modificar usuarios
- Reportes
 - Auditoria por insumo
 - Existencia actual de insumos
 - Devoluciones por usuario

- Insumos asignados
- Situación actual del inventario
- Insumos activos
- Servicios
 - Asignar insumos
 - Consultar solicitudes
 - Crear solicitudes
 - Crear insumos
 - Procesar solicitudes

4.1.1.3 SOLUCION GRAFICA

El aplicativo contendrá elementos con estándares corporativos, de los cuales se listan los siguientes:

- Colores de fondo de pantalla: azul y blanco
- Colores de fondo barras y menú: azul y blanco.
- Logotipo corporativo: representa la marca de la compañía, debe ser ubicado en la barra corporativa.
- Tipos de letra:
 - Títulos principales: Arial 16, Italic
 - Títulos secundarios: Arial 12, Bold e Italic
 - Contenido: Arial 9 e Italia
 - Mensajes de alerta: Arial 9, Bold e Italia, rojo.
- Tamaño de los botones: Ajustable al texto, mínimo 20 %
- Colores de los botones: Azul
- Mensaje de bienvenida: ubicado en la barra corporativa
- Fecha y hora del sistema: ubicado en la barra corporativa
- Versión vigente del sistema: ubicado en la barra corporativa, con el siguiente formato: año.mes.día (yy.mm.dd)
- Usuario conectado a sistema: ubicado en la barra corporativa
- Selección de fechas: calendario WEB y/o el usuario digita los valores manualmente.

4.1.2. MAQUETA DEL SISTEMA

4.1.2.1 INICIO DE SESION

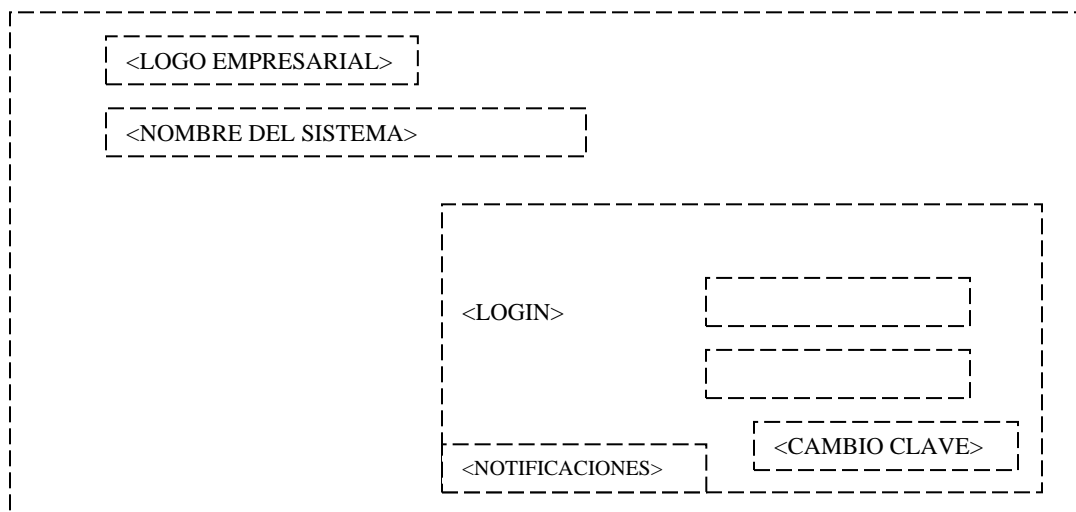


Figura 4.1. Inicio de sesión

4.1.2.2 PAGINA MAESTRA

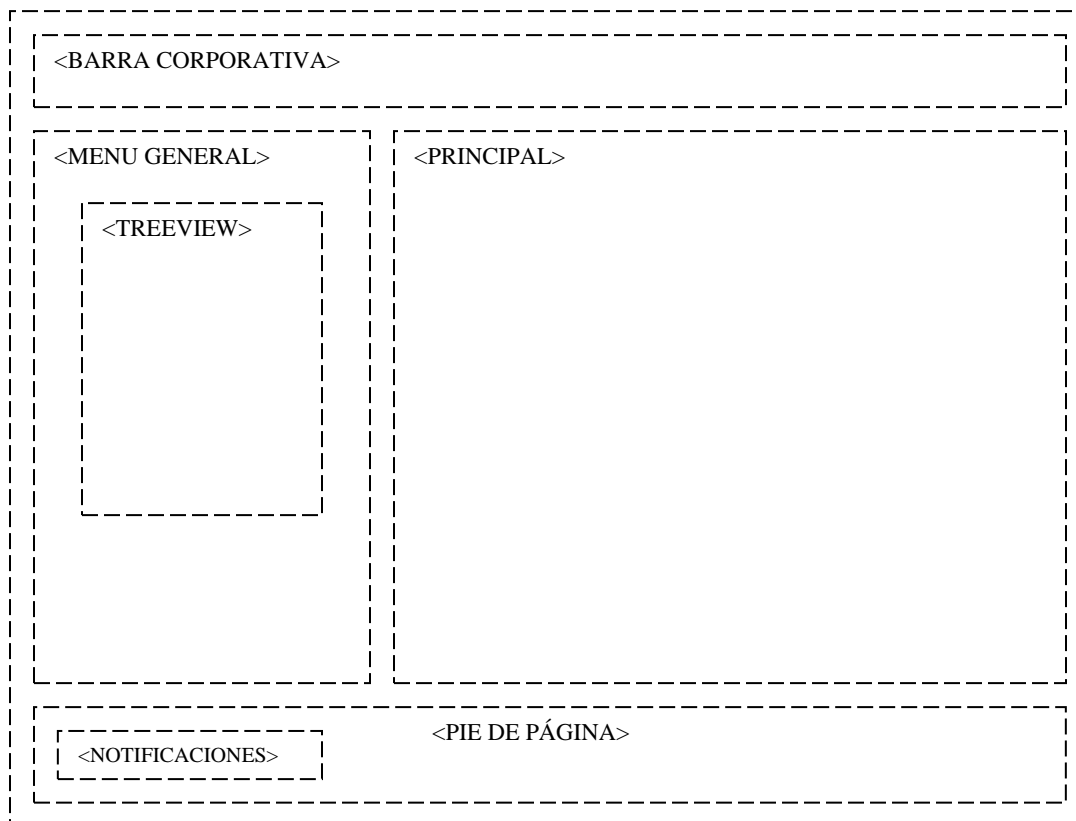
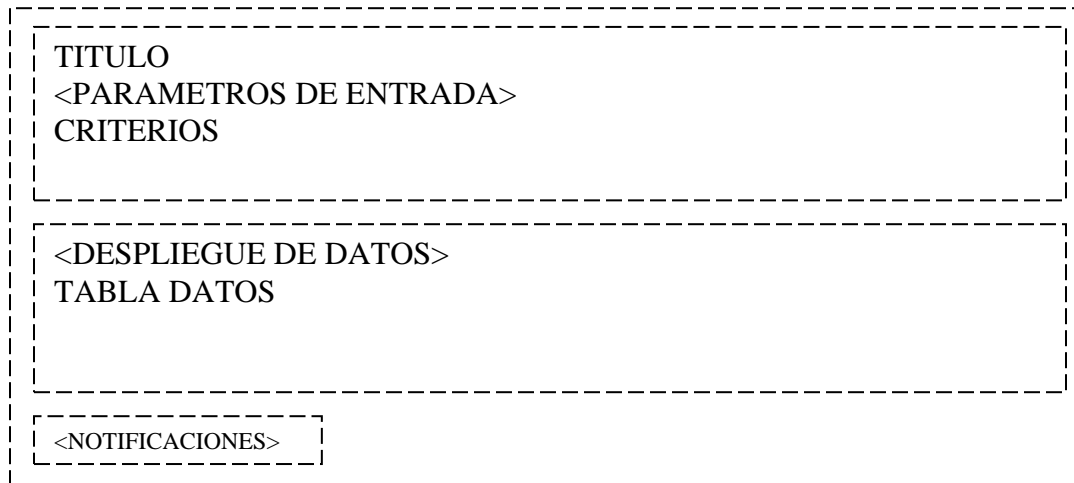


Figura 4.2. Página maestra

4.1.2.3 PRINCIPAL – REPORTES

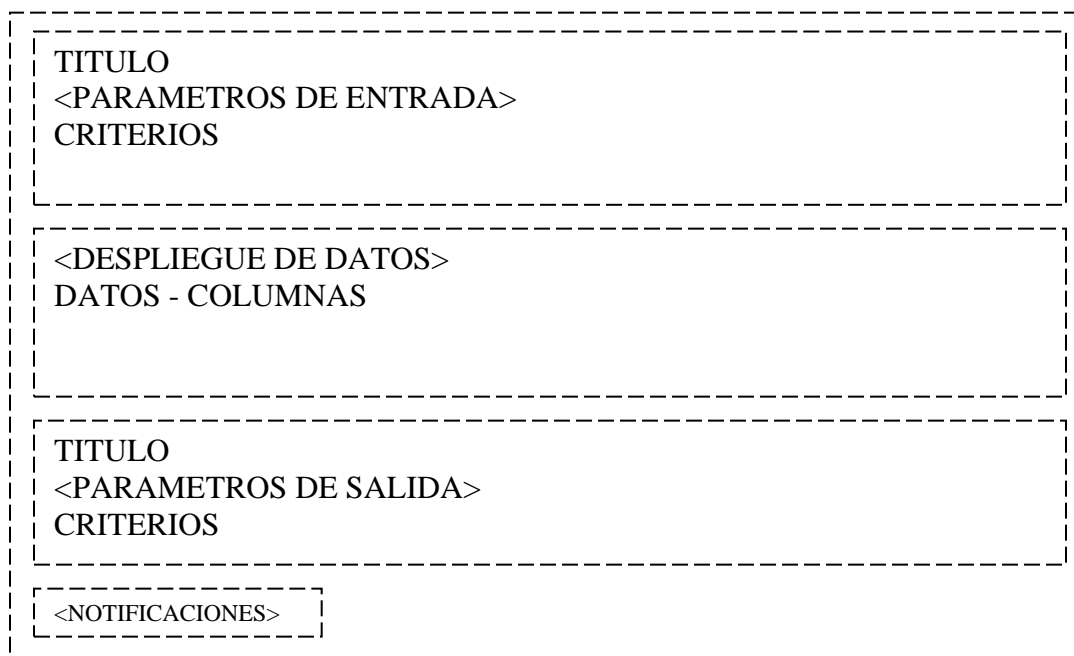


The diagram illustrates the structure of the Report Form, enclosed in a dashed border. It is divided into three main sections:

- Top Section:** Contains the text: TITULO, <PARAMETROS DE ENTRADA>, and CRITERIOS.
- Middle Section:** Contains the text: <DESPLIEGUE DE DATOS> and TABLA DATOS.
- Bottom Section:** Contains the text: <NOTIFICACIONES>.

Figura 4.3. Formulario de reportes

4.1.2.4 PRINCIPAL – ASIGNAR INSUMOS



The diagram illustrates the structure of the Assign Inputs Form, enclosed in a dashed border. It is divided into three main sections:

- Top Section:** Contains the text: TITULO, <PARAMETROS DE ENTRADA>, and CRITERIOS.
- Middle Section:** Contains the text: <DESPLIEGUE DE DATOS> and DATOS - COLUMNAS.
- Bottom Section:** Contains the text: TITULO, <PARAMETROS DE SALIDA>, and CRITERIOS.

At the very bottom, there is a separate dashed box containing the text: <NOTIFICACIONES>.

Figura 4.4. Formulario de asignación de insumos

4.1.2.5 PRINCIPAL – CREAR INSUMOS

TITULO
<PARAMETROS DE ENTRADA>
DATOS - COLUMNAS

<NOTIFICACIONES>

Figura 4.5. Formulario de nuevos insumos

4.1.2.6 PRINCIPAL – MANTENEDOR DE INSUMOS

TITULO
<PARAMETROS DE ENTRADA>
CRITERIOS

<NOTIFICACIONES>

Figura 4.6. Formulario mantenedor de insumos

4.1.2.7 PRINCIPAL – CONSULTAR SOLICITUDES

TITULO
<PARAMETROS DE ENTRADA>
CRITERIOS

<DESPLIEGUE DE DATOS>
TABLA DATOS

<NOTIFICACIONES>

Figura 4.7. Formulario consulta de solicitudes

4.1.2.8 PRINCIPAL – CREAR USUARIOS

The diagram shows a large dashed rectangular container representing the form. Inside this container, there is a smaller dashed rectangular box at the top containing the text: TITULO, <PARAMETROS DE ENTRADA>, and CRITERIOS. Below this box, there is another smaller dashed rectangular box containing the text: <NOTIFICACIONES>.

Figura 4.8. Formulario creación de usuarios

4.1.2.9 PRINCIPAL – MODIFICAR USUARIOS

The diagram shows a large dashed rectangular container representing the form. Inside this container, there is a smaller dashed rectangular box at the top containing the text: TITULO, <PARAMETROS DE ENTRADA>, and CRITERIOS. Below this box, there is another smaller dashed rectangular box containing the text: <DESPLIEGUE DE DATOS> and DATOS - COLUMNAS. At the bottom of the container, there is a third smaller dashed rectangular box containing the text: <NOTIFICACIONES>.

Figura 4.9. Formulario modificación de usuarios

4.1.2.10 PRINCIPAL – CREAR SOLICITUDES

The diagram shows a large dashed rectangular container representing the form. Inside this container, there is a smaller dashed rectangular box at the top containing the text: TITULO, <PARAMETROS DE ENTRADA>, and DATOS - COLUMNAS. Below this box, there is another smaller dashed rectangular box containing the text: <NOTIFICACIONES>.

Figura 4.10. Formulario creación de solicitudes

4.1.2.11 PRINCIPAL – PROCESAR SOLICITUDES

The diagram shows a form layout for request processing. It consists of a large outer dashed rectangle containing three smaller dashed rectangles. The top rectangle contains the text: TITULO, <PARAMETROS DE ENTRADA>, and DATOS - COLUMNAS. The bottom-left rectangle contains the text: <NOTIFICACIONES>.

Figura 4.11. Formulario para procesamiento de solicitudes

4.1.2.12 PRINCIPAL – PERFILES

The diagram shows a form layout for profile management. It consists of a large outer dashed rectangle containing three smaller dashed rectangles. The top rectangle contains the text: TITULO, <PARAMETROS DE ENTRADA>, and CRITERIOS. The middle rectangle contains the text: <DESPLIEGUE DE DATOS> and DATOS - COLUMNAS. The bottom-left rectangle contains the text: <NOTIFICACIONES>.

Figura 4.12. Formulario para manejo de perfiles

4.1.3. DIAGRAMAS

4.1.3.1 DIAGRAMA DE CASOS DE USO

4.1.3.1.1 MODULO ADMINISTRACION

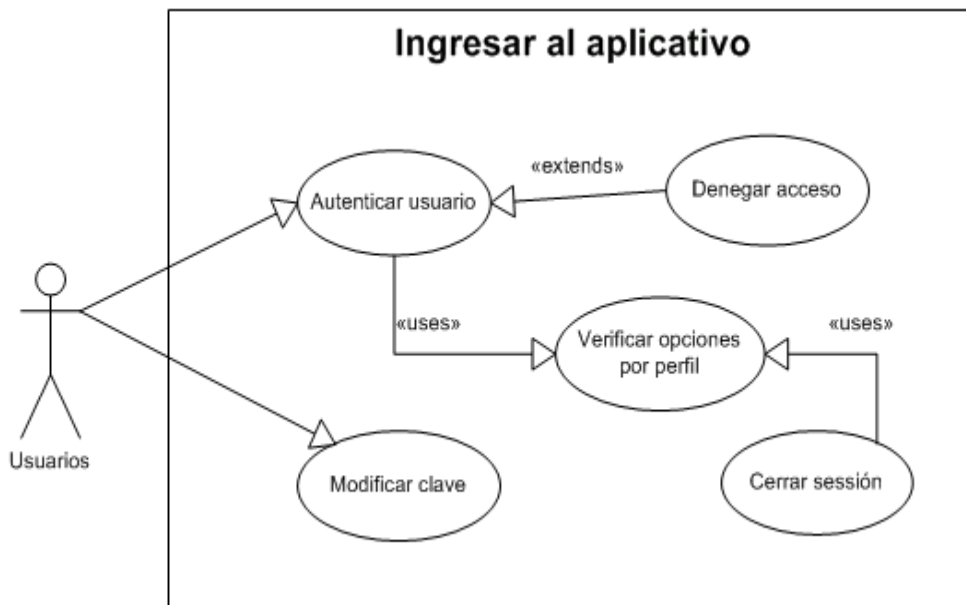


Figura 4.13. Caso de uso del Sistema: Ingresar al aplicativo
Fuente: Ficha de especificación de requerimientos 3.3.1.1

Caso de Uso	Autenticar usuario
Autor	Roberto Zumba
Fecha	03-Nov-2011
Objetivo	Acceder al sistema por medio de la pantalla de inicio de sesión.
Actores	Usuarios (Solicitante – Administrador del Inventario – Jefe del Inventario – Administrador del sistema)
Precondiciones	Los usuarios están previamente registrados y con un perfil asignado.
Flujo Normal	<ol style="list-style-type: none"> 1) Ingresar al sistema. 2) El actor ingresa la contraseña en la pantalla de inicio de sesión. 3) El sistema valida los datos ingresados. 4) El sistema otorga permisos de acceso al sistema. 5) El sistema presenta la pantalla “Página Maestra”. 6) En el menú de opciones se cargan los formularios de acuerdo al perfil.
Flujo Alternativo	3) El sistema valida los datos ingresados, si los datos no son correctos el sistema presenta el mensaje para que corrija la información.
Poscondiciones	El acceso al sistema es satisfactorio.

Tabla 4.1. Especificación de caso de uso: Autenticar usuario
Fuente: Ficha de especificación de requerimientos RF_3311.1

Caso de Uso	Modificar clave
Autor	Roberto Zumba
Fecha	04-Nov-2011
Objetivo	Actualizar la clave de acceso al sistema por medio de la pantalla de inicio de sesión.
Actores	Usuarios (Solicitante – Administrador del Inventario – Jefe del Inventario – Administrador del sistema)
Precondiciones	Los usuarios están previamente registrados y con un perfil asignado.
Flujo Normal	<ol style="list-style-type: none"> 1) Ingresar al sistema. 2) El actor ingresa la clave en la pantalla de inicio de sesión. 3) El sistema valida los datos ingresados. 4) El sistema muestra una caja de texto para que ingrese la nueva clave. 5) El sistema presenta un mensaje confirmando el cambio de clave exitosa.
Flujo Alternativo	N/A
Poscondiciones	N/A

*Tabla 4.2. Especificación de caso de uso: Modificar clave.
Fuente: Ficha de especificación de requerimientos RF_3311.2*

Caso de Uso	Verificar opciones por perfil
Autor	Roberto Zumba
Fecha	07-Nov-2011
Objetivo	Asegurar que los formularios asignados al perfil del usuario se carguen correctamente en la página principal del sistema.
Actores	Usuarios (Solicitante – Administrador del Inventario – Jefe del Inventario – Administrador del sistema)
Precondiciones	El usuario debe haberse autenticado en el sistema.
Flujo Normal	<ol style="list-style-type: none"> 1) El usuario navega por el menú de opciones del sistema. 2) El usuario selecciona los formularios que se encuentran disponibles en el menú de opciones. 3) Los formularios se cargan correctamente.
Flujo Alternativo	<ol style="list-style-type: none"> 4) Cerrar sesión.
Poscondiciones	Se cargan las opciones del perfil asignado al usuario.

*Tabla 4.3. Especificación de caso de uso: Verificar opciones por perfil
Fuente: Ficha de especificación de requerimientos RF_3311.1*

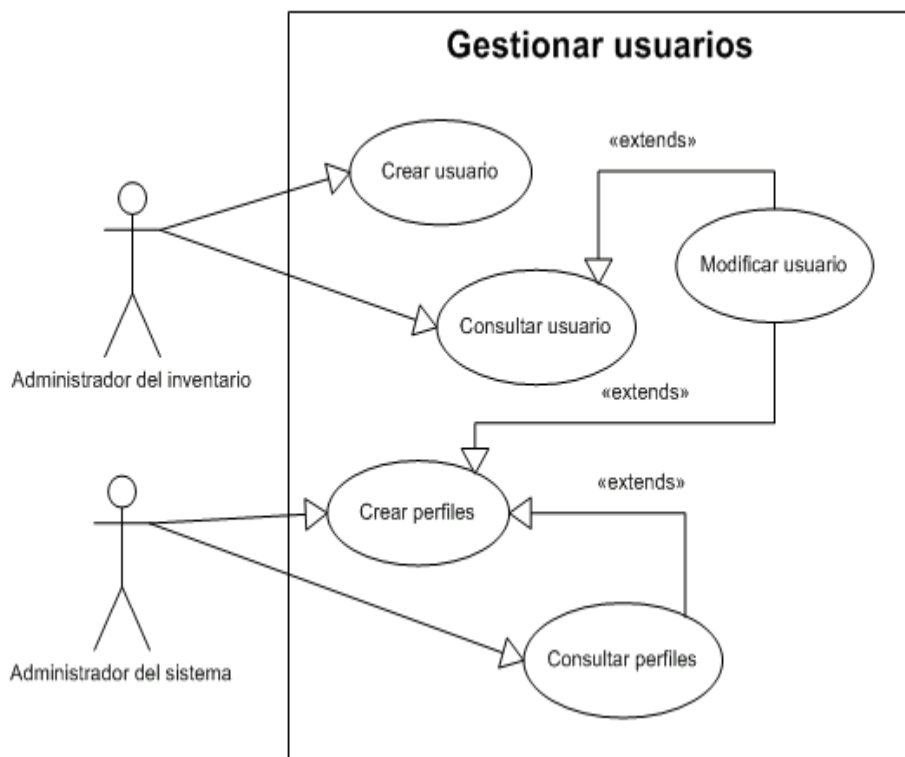


Figura 4.14. Caso de uso del Sistema: Gestionar usuarios
 Fuente: Ficha de especificación de requerimientos 3.3.1.3 – 3.3.1.5

Caso de Uso	Crear usuarios
Autor	Roberto Zumba
Fecha	08-Nov-2011
Objetivo	Crear un nuevo usuario para acceder al sistema.
Actores	Administrador del Inventario
Precondiciones	El actor debe tener permisos para crear un usuario.
Flujo Normal	<ol style="list-style-type: none"> 1) Ingresar a la opción de creación de usuarios en el modulo de administración del sistema. 2) Seleccionar el perfil para el usuario. 3) Ingresar los datos: nombre - apellido, correo, usuario de red, proveedor, contacto móvil, contacto fijo, extensión. 4) El sistema solicita confirmación de datos. 5) Validar datos. 6) Guardar datos.
Flujo Alternativo	5) Validar datos, si el usuario existe presentar el mensaje informativo.
Poscondiciones	El sistema presenta mensaje de creación exitosa.

Tabla 4.4. Especificación de caso de uso: Crear usuarios
 Fuente: Ficha de especificación de requerimientos RF_3313.3

Caso de Uso	Modificar usuarios
Autor	Roberto Zumba
Fecha	09-Nov-2011
Objetivo	Modificar los datos de un usuario existente.
Actores	Administrador del Inventario
Precondiciones	El actor debe tener permisos para modificar un usuario.
Flujo Normal	<ol style="list-style-type: none"> 1) Ingresar a la opción de modificar usuarios en el modulo de administración del sistema. 2) Consultar usuarios por apellido. 3) El sistema despliega las coincidencias si existe. 4) Modificar los datos: nombre - apellido, correo, proveedor, contacto móvil, contacto fijo, extensión. El sistema solicita confirmación de datos. 5) Validar datos. 6) Guardar datos.
Flujo Alternativo	<ol style="list-style-type: none"> 2) Consultar usuario, si el usuario no existe presentar el mensaje informativo. 3) Modificar alguno de los datos: Perfil del usuario
Poscondiciones	El sistema presenta mensaje de modificación exitosa.

*Tabla 4.5. Especificación de caso de uso: Modificar usuarios
Fuente: Ficha de especificación de requerimientos RF_3313.4*

Caso de Uso	Crear perfiles
Autor	Roberto Zumba
Fecha	10-Nov-2011
Objetivo	Crear un perfil nuevo en el sistema.
Actores	Administrador del Sistema
Precondiciones	El actor debe tener permisos para crear un perfil.
Flujo Normal	<ol style="list-style-type: none"> 1) Ingresar a la opción de crear perfiles en el modulo de administración del sistema. 2) El sistema presenta al usuario los perfiles existentes. 3) Ingresar el nombre para crear el perfil 4) El sistema solicita confirmación de datos. 5) Validar datos. 6) Guardar datos.
Flujo Alternativo	4) Validar datos, si el perfil existe presentar el mensaje informativo.
Poscondiciones	El sistema presenta mensaje de creación exitosa.

*Tabla 4.6. Especificación de caso de uso: Crear perfiles
Fuente: Ficha de especificación de requerimientos RF_3315.1*

Caso de Uso	Consultar perfiles
Autor	Roberto Zumba
Fecha	11-Nov-2011
Objetivo	Consultar los perfiles existentes en el sistema.
Actores	Administrador del Sistema
Precondiciones	El actor debe tener permisos para visualizar un perfil.
Flujo Normal	1) Ingresar a la opción perfiles en el modulo de administración del sistema. 2) El sistema presenta los perfiles existentes.
Flujo Alternativo	2) Si no existen perfiles se presenta el mensaje informativo.
Poscondiciones	N/A

Tabla 4.7. Especificación de caso de uso: Consultar perfiles
Fuente: Ficha de especificación de requerimientos RF_3315.2

4.1.3.1.2 MODULO REPORTES

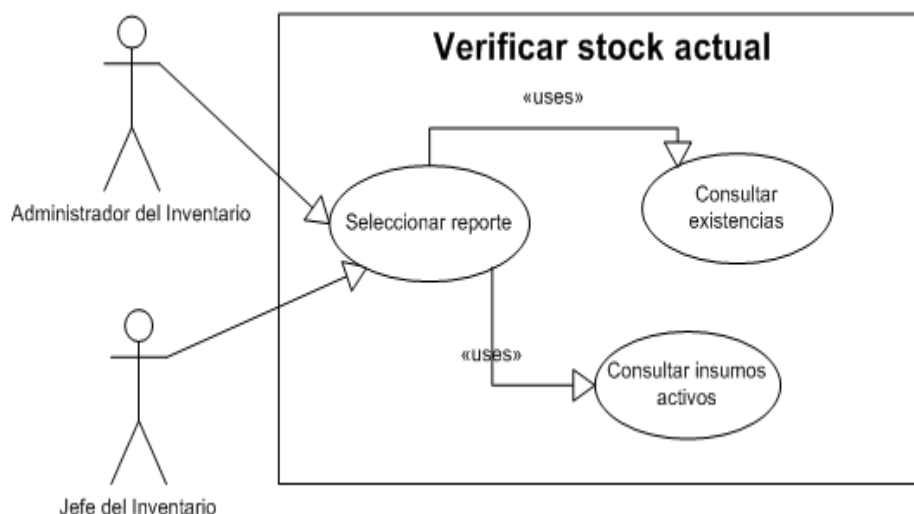


Figura 4.15. Caso de uso del sistema: Verificar stock actual
Fuente: Ficha de especificación de requerimientos 3.3.1.3

Caso de Uso	Consultar existencias
Autor	Roberto Zumba
Fecha	14-Nov-2011
Objetivo	Disponer de información relacionada con la cantidad totalizada de existencias disponibles y asignadas a los solicitantes.
Actores	Administrador del Inventario – Jefe del Inventario
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	1) El actor selecciona el módulo de reportes. 2) El actor selecciona el submenú Stock actual de

	insumos 3) El sistema presenta una tabla con el totalizado de existencias <ul style="list-style-type: none"> ○ Tipo ○ Ambiente ○ Estado ○ Cantidad
Flujo Alternativo	N/A
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

*Tabla 4.8. Especificación de caso de uso: Consultar existencias
Fuente: Ficha de especificación de requerimientos RF_3313.13*

Caso de Uso	Consultar insumos activos
Autor	Roberto Zumba
Fecha	15-Nov-2011
Objetivo	Disponer de información actualizada con el detalle de los insumos activos en producción.
Actores	Administrador del Inventario
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	1) El actor selecciona el módulo de reportes. 2) El actor selecciona el submenú insumos activos. 3) El sistema presenta una tabla con el detalle de las existencias, de acuerdo a los criterios entregados por el administrador del inventario. <ul style="list-style-type: none"> ○ CAJA ○ SERIE ○ ABONADO ○ CELULAR ○ CLIENTE ○ CUENTA ○ CODIGO PLAN ○ DESCRIPCION PLAN ○ CICLO ○ FECHA ALTA ○ IMEI
Flujo Alternativo	N/A
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

*Tabla 4.9. Especificación de caso de uso: Consultar Insumos activos.
Fuente: Ficha de especificación de requerimientos RF_3313.11*

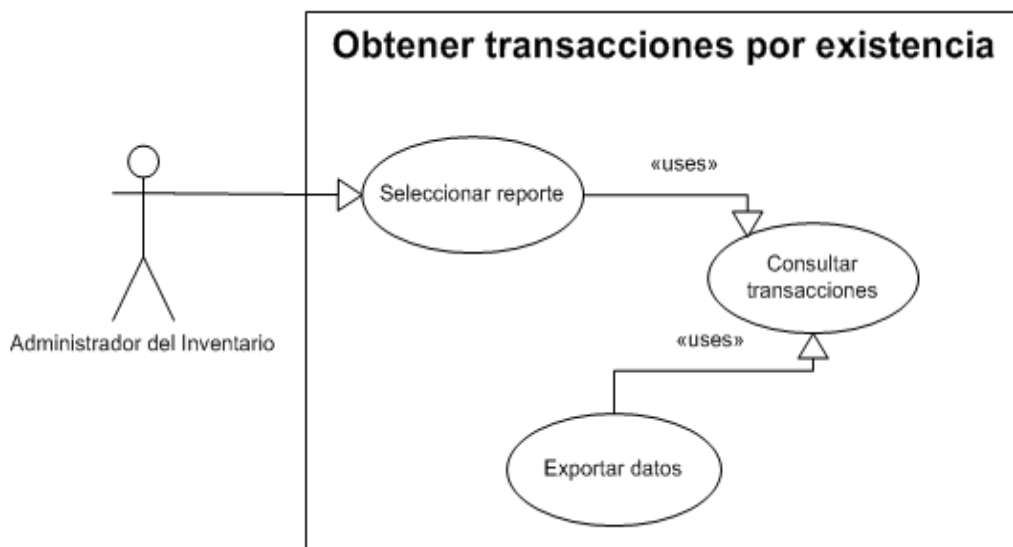


Figura 4.16 Caso de uso del sistema: Obtener transacciones por existencia.
Fuente: Ficha de especificación de requerimientos 3.3.1.3

Caso de Uso	Consultar transacciones
Autor	Roberto Zumba
Fecha	15-Nov-2011
Objetivo	Disponer de información histórica de los movimientos realizados sobre un insumo.
Actores	Administrador del Inventario
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo de reportes. 2) El actor selecciona el submenú Auditoria por insumo. 3) El actor consulta un insumo en el sistema. 4) El sistema presenta un reporte en Excel con la siguiente información <ul style="list-style-type: none"> ○ Insumo ○ Serie ○ Caja ○ Central ○ Ambiente ○ Fecha de creación ○ Estado ○ Modelo ○ Tecnología ○ Fecha de revisión ○ Responsable ○ Notas ○ Fecha de evento ○ Tipo de evento (creación, eliminación, modificación.)

Flujo Alternativo	N/A
Pos condiciones	Si no existe información el sistema muestra mensaje respectivo.

Tabla 4.10 Especificación de caso de uso: Consultar transacciones

Fuente: Ficha de especificación de requerimientos RF_3313.12

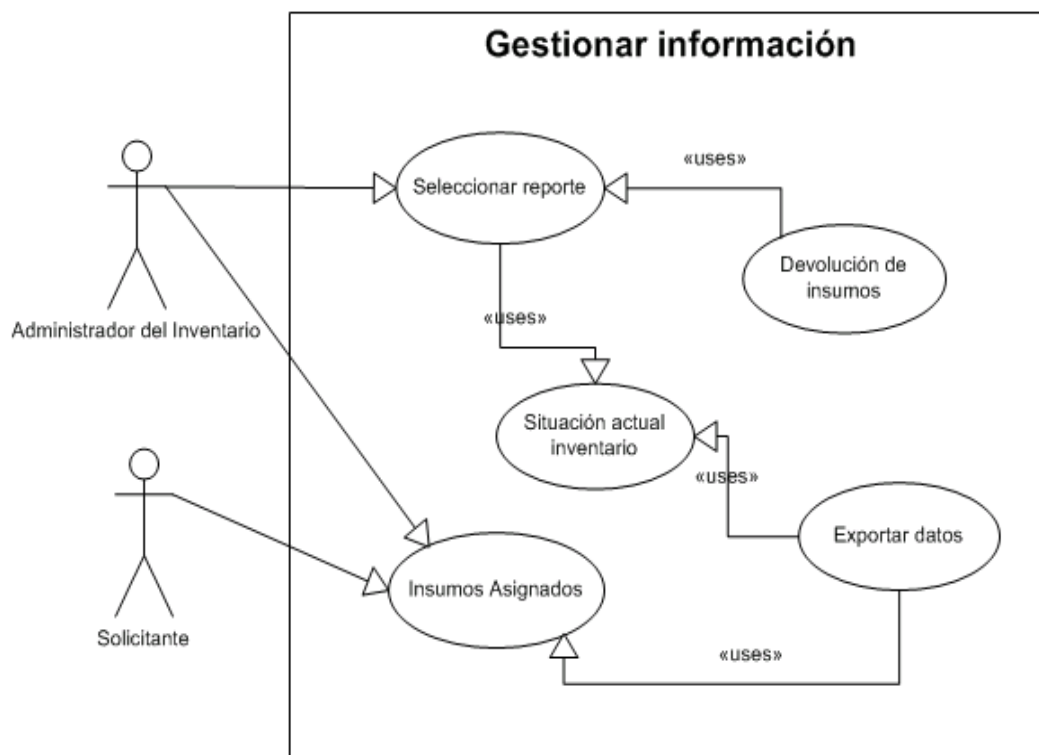


Figura 4.17 Caso de uso del sistema: Gestionar información.

Fuente: Ficha de especificación de requerimientos 3.3.1.3

Caso de Uso	Devolución de insumos
Autor	Roberto Zumba
Fecha	17-Nov-2011
Objetivo	Obtener un reporte con la información de las devoluciones de insumos realizadas por los usuarios.
Actores	Administrador del Inventario
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo de reportes. 2) El actor selecciona el submenú devoluciones por usuario. 3) El sistema presenta un cuadro con la siguiente información: <ul style="list-style-type: none"> ○ Físico (si/no) ○ Insumo

	<ul style="list-style-type: none"> ○ Requerimiento PPM ○ Solicitud ○ Usuario ○ Fecha devolución
Flujo Alternativo	N/A
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

*Tabla 4.11 Especificación de caso de uso: Devolución de insumos
Fuente: Ficha de especificación de requerimientos RF_3313.9*

Caso de Uso	Situación actual del inventario
Autor	Roberto Zumba
Fecha	18-Nov-2011
Objetivo	Obtener un reporte en Excel con la información de todos los insumos y el estado actual.
Actores	Administrador del Inventario
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo de reportes. 2) El actor selecciona el submenú situación actual del inventario. 3) El sistema presenta un archivo Excel con la siguiente información: <ul style="list-style-type: none"> ○ Insumo ○ Caja ○ Central ○ Ambiente ○ Fecha de creación ○ Situación ○ Modelo ○ Tecnología ○ Fecha de revisión ○ Responsable revisión ○ Notas
Flujo Alternativo	N/A
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

*Tabla 4.12 Especificación de caso de uso: Situación actual del inventario
Fuente: Ficha de especificación de requerimientos RF_3313.10*

Caso de Uso	Insumos Asignados
Autor	Roberto Zumba
Fecha	21-Nov-2011
Objetivo	Obtener un reporte visual y/o exportable a Excel con la

	información de todos los insumos asignados a un solicitante.
Actores	Administrador del Inventario – Solicitante
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo de reportes. 2) El actor selecciona el submenú insumos asignados. 3) El sistema presenta una tabla con la siguiente información: <ul style="list-style-type: none"> ○ Insumo ○ Físico (si/no) ○ Central ○ Solicitud ○ Ambiente ○ Usuario ○ Modelo ○ Tecnología ○ Fecha devolución ○ Requerimiento PPM
Flujo Alternativo	<p>3) El sistema presenta una tabla con la información solicitada, para el caso del solicitante se presenta por defecto el reporte con la información existente. Se requiere disponer de los siguientes parámetros de entrada adicionales: Número de solicitud [actor: Administrador y Solicitante] Solicitante [Administrador]</p> <p>4) Un solicitante no puede consultar las solicitudes / insumos de otro solicitante, esto debe estar permitido unidamente para el administrador del inventario.</p> <p>5) El reporte para el administrador del inventario se debe desplegar en una tabla ordenada por fecha de devolución más antigua.</p>
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

*Tabla 4.13 Especificación de caso de uso: Insumos Asignados.
Fuente: Ficha de especificación de requerimientos RF_3312.3*

4.1.3.1.3 MODULO SERVICIOS

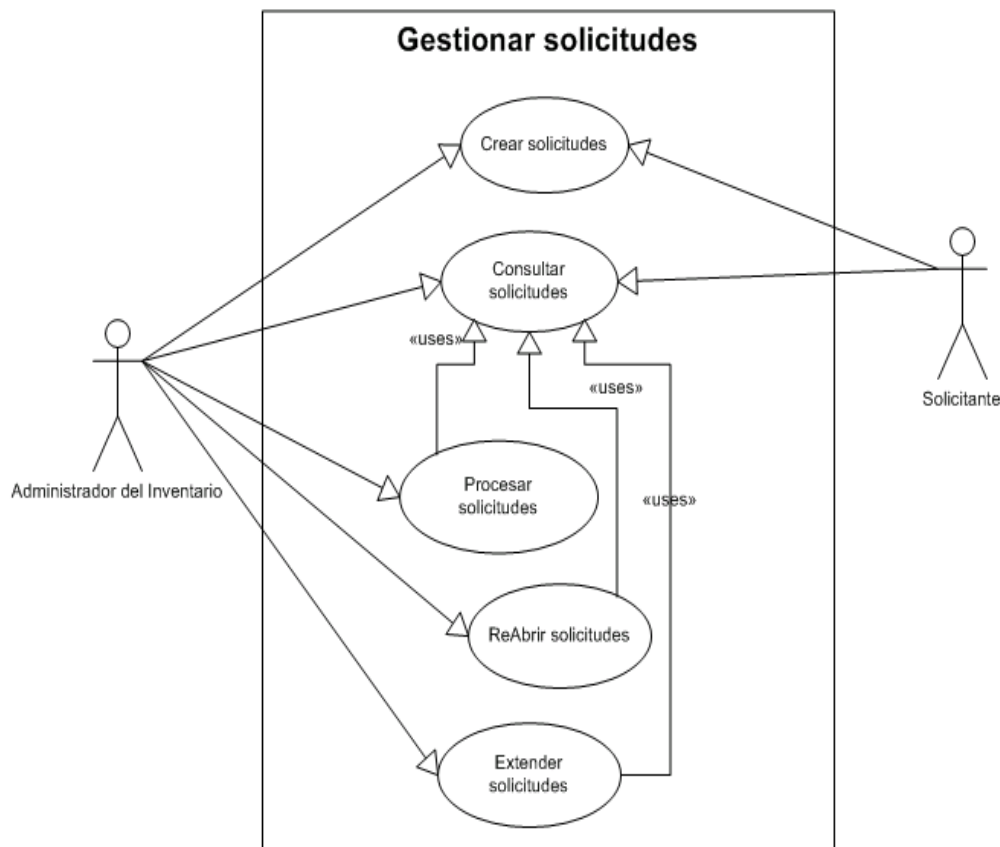


Figura 4.18 Caso de uso del sistema: Gestionar solicitudes.
Fuente: Ficha de especificación de requerimientos 3.3.1.3

Caso de Uso	Crear solicitudes
Autor	Roberto Zumba
Fecha	22-Nov-2011
Objetivo	Crear una solicitud de insumos
Actores	Solicitante – Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo servicios. 2) El actor selecciona el submenú crear solicitudes. 3) El sistema presenta un formulario para ingresar la siguiente información: <ul style="list-style-type: none"> ○ Recurso ○ Físico o Lógico ○ Cantidad ○ Justificación ○ Requerimiento interno (PPM) ○ Fecha prevista devolución ○ Ambiente. 4) El sistema controla datos requeridos 5) El sistema guarda la información.

Flujo Alternativo	3) El administrador de inventarios podrá crear solicitudes seleccionado una lista de usuarios permitidos. 4) El sistema controla datos requeridos, si no está completa la información no graba los datos.
Pos condiciones	El sistema asigna un número de solicitud interno e indica que la transacción se realizó correctamente.

Tabla 4.14 Especificación de caso de uso: Crear solicitudes

Fuente: Ficha de especificación de requerimientos RF_3313.1

Caso de Uso	Consultar solicitudes
Autor	Roberto Zumba
Fecha	23-Nov-2011
Objetivo	Revisar el estado de las solicitudes.
Actores	Administrador de inventarios, solicitante
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo servicios. 2) El actor selecciona el submenú consultar solicitudes. 3) El sistema presenta un cuadro con el detalle las solicitudes pendientes de procesar, con la siguiente información: <ul style="list-style-type: none"> ○ Número de solicitud ○ Fecha de solicitud ○ Usuario ○ Fecha de asignación ○ Usuario ○ Recurso ○ Cantidad ○ Justificación ○ Requerimiento interno (PPM) ○ Fecha prevista devolución ○ Ambiente ○ Física o lógica ○ Estado ○ Cantidad de días por prórroga ○ Observaciones o notas
Flujo Alternativo	<ol style="list-style-type: none"> 4) El actor (solicitante) ingresa los parámetros de entrada: Fecha Inicio y Fecha Fin. 5) El actor (coordinador) ingresa el parámetro de entrada: solicitud 6) El sistema presenta un cuadro con el detalle las solicitudes y su estado actual.
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

Tabla 4.15 Especificación de caso de uso: Consultar solicitudes

Fuente: Ficha de especificación de requerimientos RF_3312.2

Caso de Uso	Procesar solicitudes
Autor	Roberto Zumba
Fecha	23-Nov-2011
Objetivo	Aprobar o Negar las solicitudes creadas por los solicitantes.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema La solicitud debe estar en estado pendiente
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo servicios. 2) El actor selecciona el submenú procesar solicitudes. 3) El sistema presenta una caja de texto para buscar una solicitud. 4) El sistema presenta las acciones disponibles. 5) El actor selecciona aprobar o negar, además ingresa una nota descriptiva de la acción tomada. 6) El sistema presenta un mensaje de transacción exitosa.
Flujo Alternativo	3) El sistema presenta una caja de texto para buscar una solicitud, si no encuentra la solicitud no presenta las acciones disponibles
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

*Tabla 4.16 Especificación de caso de uso: Procesar solicitudes
Fuente: Ficha de especificación de requerimientos RF_3313.2*

Caso de Uso	Extender solicitudes
Autor	Roberto Zumba
Fecha	26-Nov-2011
Objetivo	Aplicar una prórroga a una solicitud existente.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema La solicitud debe estar en estado procesado
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo servicios. 2) El actor selecciona el submenú Extender solicitudes. 3) El sistema presenta una caja de texto para buscar una solicitud. 4) El sistema presenta las opciones disponibles. 5) El actor selecciona la cantidad de días (desde 0 hasta 30) que serán sumadas a la fecha devolución. 6) El actor selecciona actualizar. 7) El sistema presenta un mensaje de transacción exitosa.
Flujo	3) El sistema presenta una caja de texto para buscar

Alternativo	una solicitud, si no encuentra la solicitud no presenta las acciones disponibles.
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

*Tabla 4.17 Especificación de caso de uso: Extender solicitudes
Fuente: Ficha de especificación de requerimientos RF_3313.5*

Caso de Uso	Re abrir solicitudes
Autor	Roberto Zumba
Fecha	26-Nov-2011
Objetivo	Cambiar el estado de una solicitud
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema La solicitud debe estar en estado aprobado o negado
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo servicios. 2) El actor selecciona el submenú Re abrir solicitudes. 3) El sistema presenta una caja de texto para buscar una solicitud. 4) El actor selecciona reabrir la solicitud. 5) El sistema presenta un mensaje de transacción exitosa.
Flujo Alternativo	<ol style="list-style-type: none"> 1) El sistema presenta una caja de texto para buscar una solicitud. 2) El sistema no encuentra la solicitud 3) El actor selecciona reabrir solicitud. 4) El sistema presenta un mensaje solicitando que busque una solicitud primero.
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

*Tabla 4.18 Especificación de caso de uso: Re abrir solicitudes
Fuente: Ficha de especificación de requerimientos RF_3313.6*

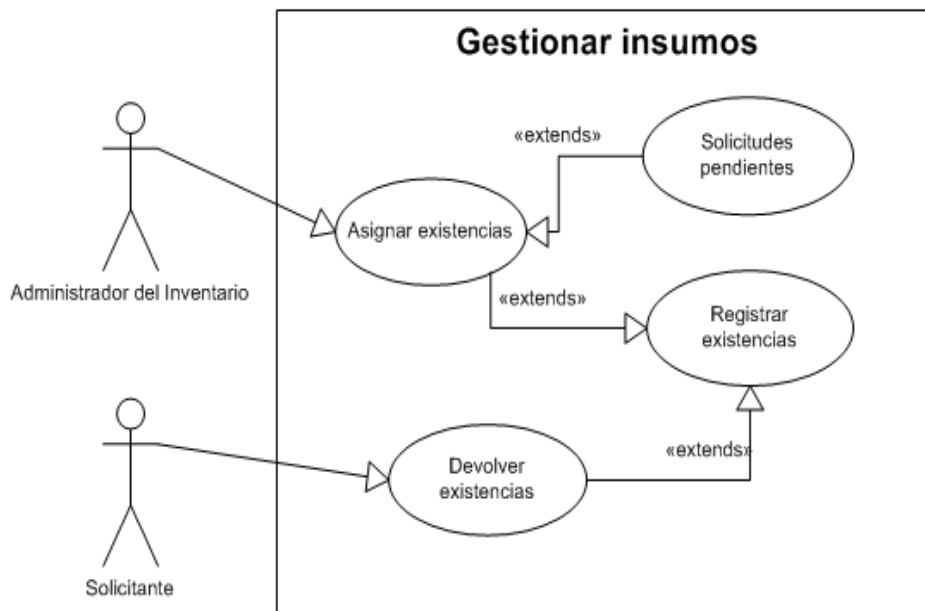


Figura 4.19 Caso de uso del sistema: Gestionar insumos.

Fuente: Ficha de especificación de requerimientos 3.3.1.3

Caso de Uso	Solicitudes pendientes
Autor	Roberto Zumba
Fecha	28-Nov-2011
Objetivo	Revisar las solicitudes pendientes por procesar.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo servicios. 2) El actor selecciona el submenú consultar solicitudes. 3) El sistema presenta una lista con las solicitudes pendientes de procesar, con la siguiente información: <ul style="list-style-type: none"> ○ Número de solicitud ○ Fecha de pedido ○ Usuario - Nombres ○ Insumo ○ Cantidad ○ Justificación ○ Req_PPM ○ Fecha devolución ○ Ambiente ○ Física (si/no)
Flujo Alternativo	N/A
Pos condiciones	Si no existe información el sistema muestra el mensaje respectivo.

Tabla 4.19 Especificación de caso de uso: Solicitudes pendientes
Fuente: Ficha de especificación de requerimientos RF_3312.2

Caso de Uso	Registrar existencias
Autor	Roberto Zumba
Fecha	28-Nov-2011
Objetivo	Asignar o devolver insumos a una solicitud.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema Disponer de una solicitud en estado pendiente
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo servicios, submenú asignar insumos. 2) El actor selecciona las existencias físicas. 3) El actor obtiene los códigos de las existencias. 4) El actor registra los códigos a una solicitud. 5) El actor selecciona el número de solicitud 6) El actor selecciona actualizar / asignar. 7) El sistema presenta un mensaje de asignación exitosa.
Flujo Alternativo	N/A
Pos condiciones	Los actores revisan las existencias por solicitante y/o solicitud y el sistema presenta las existencias asignadas.

Tabla 4.20 Especificación de caso de uso: Registrar existencias.
Fuente: Ficha de especificación de requerimientos RF_3313.1

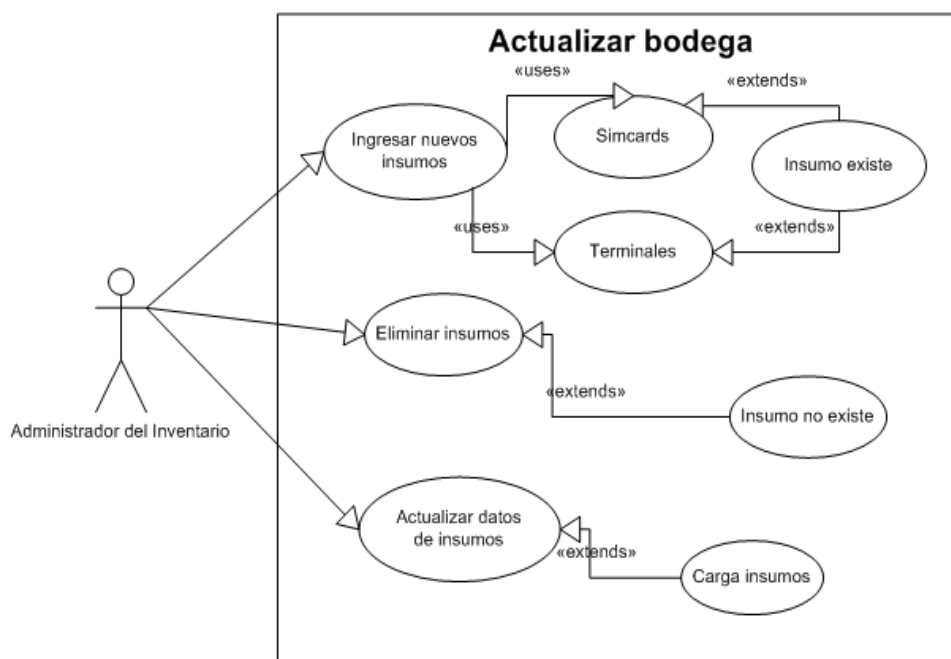


Figura 4.20 Caso de uso del sistema: Actualizar bodega.

Fuente: Ficha de especificación de requerimientos 3.3.1.3

Caso de Uso	Ingresar nuevos insumos – simcards
Autor	Roberto Zumba
Fecha	01-Dic-2011
Objetivo	Crear nuevas existencias en el sistema.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona el módulo servicios, submenú crear insumos. 2) El actor selecciona los datos de las existencias nuevas. 3) Ingresa los siguientes datos. <ul style="list-style-type: none"> • Código de existencia • Caja • Tipo de uso • Ambiente • Estado 4) El sistema valida los datos 5) El sistema guarda los datos
Flujo Alternativo	<ol style="list-style-type: none"> 1) El sistema valida los datos, si los datos no son completos el sistema solicita que se complete la información. 2) El sistema no guarda los datos
Pos condiciones	El sistema presenta un mensaje de creación exitosa

Tabla 4.21 Especificación de caso de uso: Ingresar nuevos insumos simcards.

Fuente: Ficha de especificación de requerimientos RF_3313.7

Caso de Uso	Ingresar nuevos insumos - terminales
Autor	Roberto Zumba
Fecha	01-Dic-2011
Objetivo	Crear nuevas existencias en el sistema.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ol style="list-style-type: none"> 1) El actor selecciona las existencias nuevas. 2) Ingresa los siguientes datos. <ul style="list-style-type: none"> • Código de existencia • Serie • Modelo • Estado • Tecnología

	<ul style="list-style-type: none"> • Responsable de revisión 3) El sistema valida los datos 4) El sistema guarda los datos
Flujo Alternativo	3) El sistema valida los datos, si los datos no son completos el sistema solicita que se complete la información. 4) El sistema no guarda los datos
Pos condiciones	El sistema presenta un mensaje de creación exitosa

Tabla 4.22 Especificación de caso de uso: Ingresar nuevos insumos terminales.

Fuente: Ficha de especificación de requerimientos RF_3313.7

Caso de Uso	Actualizar datos de insumos
Autor	Roberto Zumba
Fecha	01-Dic-2011
Objetivo	Actualizar existencias en el sistema.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	1) El actor selecciona el menú administración, submenú mantenedor de insumos. 2) El actor selecciona las existencias. 3) Modifica los datos. 4) El sistema valida los datos 5) El sistema guarda los datos
Flujo Alternativo	2) Modifica los datos: Por caja, ambiente, estado. 3) El sistema valida los datos, si los datos no son completos el sistema solicita que se complete la información. 4) El sistema no guarda los datos
Pos condiciones	El sistema presenta un mensaje de modificación exitosa

Tabla 4.23 Especificación de caso de uso: Actualizar datos de insumos.

Fuente: Ficha de especificación de requerimientos RF_3313.8

Caso de Uso	Cargar insumos
Autor	Roberto Zumba
Fecha	01-Dic-2011
Objetivo	Cargar un archivo con las existencias al sistema.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	1) El actor selecciona el menú servicios, submenú mantenedor de insumos. 2) El actor selecciona las existencias.

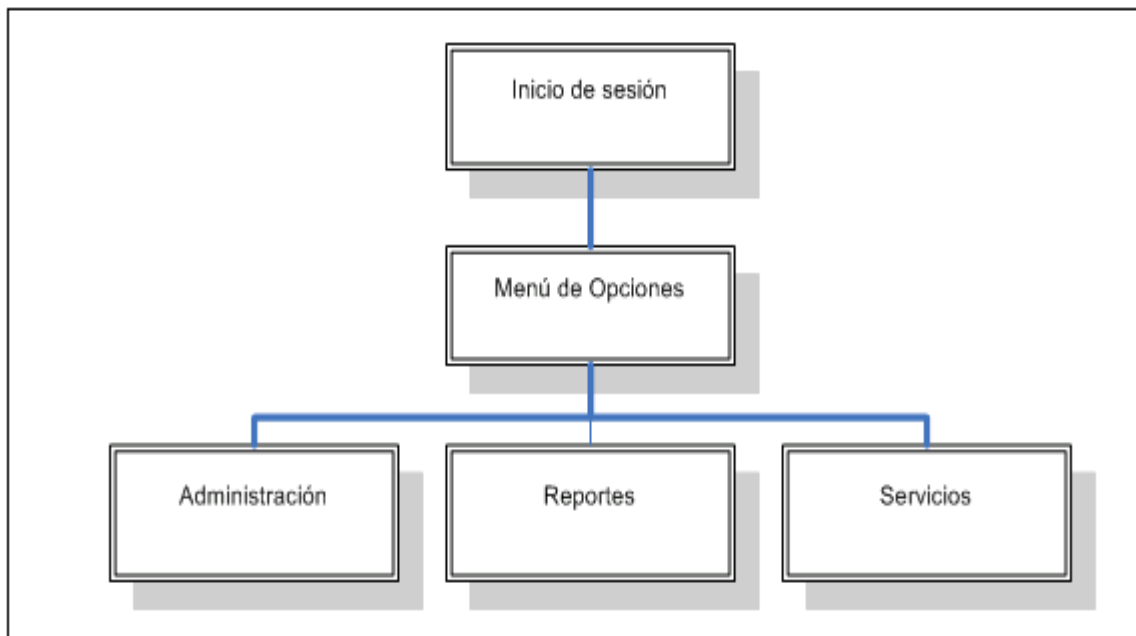
	<ul style="list-style-type: none"> 3) Edita un archivo de texto y guarda los códigos de las existencias. 4) El actor busca el archivo de texto con la opción examinar y carga la información al sistema solicita. 5) El sistema solicita confirmación de datos. 6) El actor confirma cargar datos 7) El sistema guarda los datos.
Flujo Alternativo	3) El sistema solicita confirmar, si no confirma cargar el sistema no guarda las existencias.
Pos condiciones	El sistema presenta un mensaje de eliminación exitosa

*Tabla 4.24 Especificación de caso de uso: Cargar insumos.
Fuente: Ficha de especificación de requerimientos RF_3313.8*

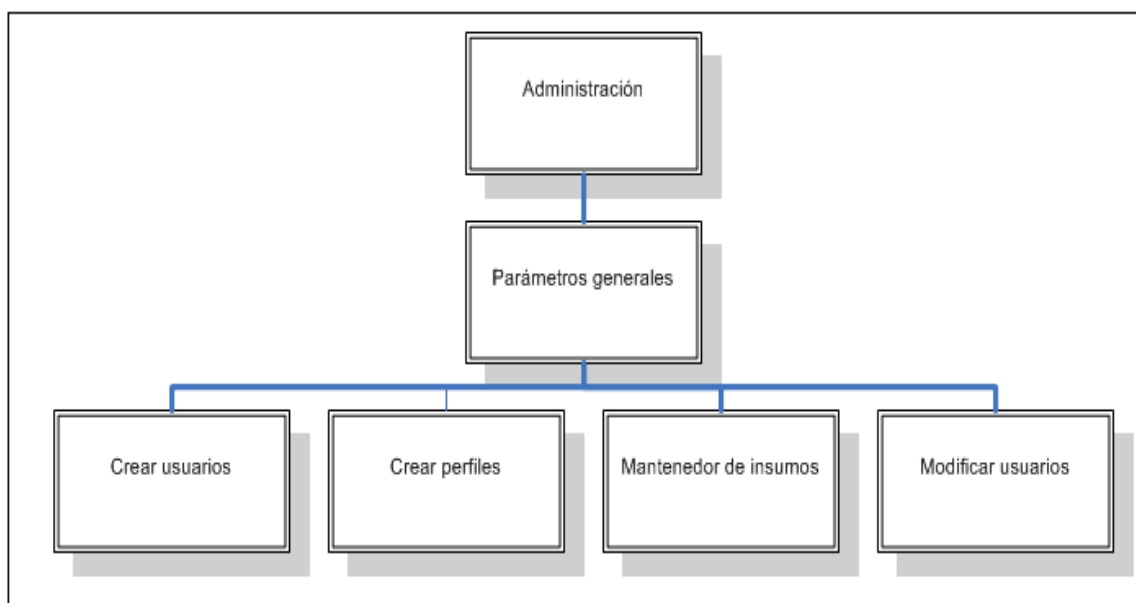
Caso de Uso	Eliminar insumos
Autor	Roberto Zumba
Fecha	01-Dic-2011
Objetivo	Eliminar existencias del sistema.
Actores	Administrador de inventarios
Precondiciones	Los actores disponen del perfil para acceder a esta información y deben haberse autenticado en el sistema
Flujo Normal	<ul style="list-style-type: none"> 1) El actor selecciona las existencias. 2) Selecciona eliminar existencias 3) El sistema solicita confirmar 4) El actor confirma eliminación
Flujo Alternativo	3) El sistema solicita confirmar, si no confirma eliminar el sistema no elimina la existencia.
Pos condiciones	El sistema presenta un mensaje de eliminación exitosa

*Tabla 4.25 Especificación de caso de uso: Eliminar insumos.
Fuente: Ficha de especificación de requerimientos RF_3313.8*

4.1.3.2 DIAGRAMA NAVEGACIONAL



*Figura 4.21 Diagrama Navegacional: Inicio de sesión.
Descripción: El usuario permitido tendrá acceso a los 3 módulos del sistema.*



*Figura 4.22 Diagrama Navegacional: Administración.
Descripción: El usuario permitido tendrá acceso a los 4 sub-módulos del módulo de administración del sistema.*

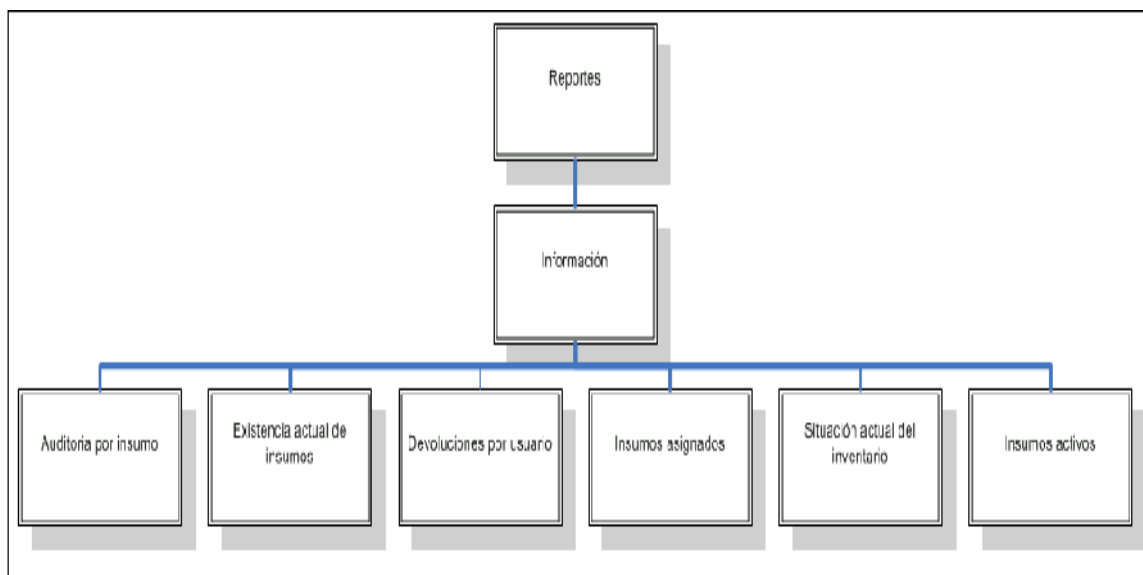


Figura 4.23 Diagrama Navegacional: Reportes.

Descripción: El usuario permitido tendrá acceso a los 6 sub-módulos del módulo de reportes del sistema.

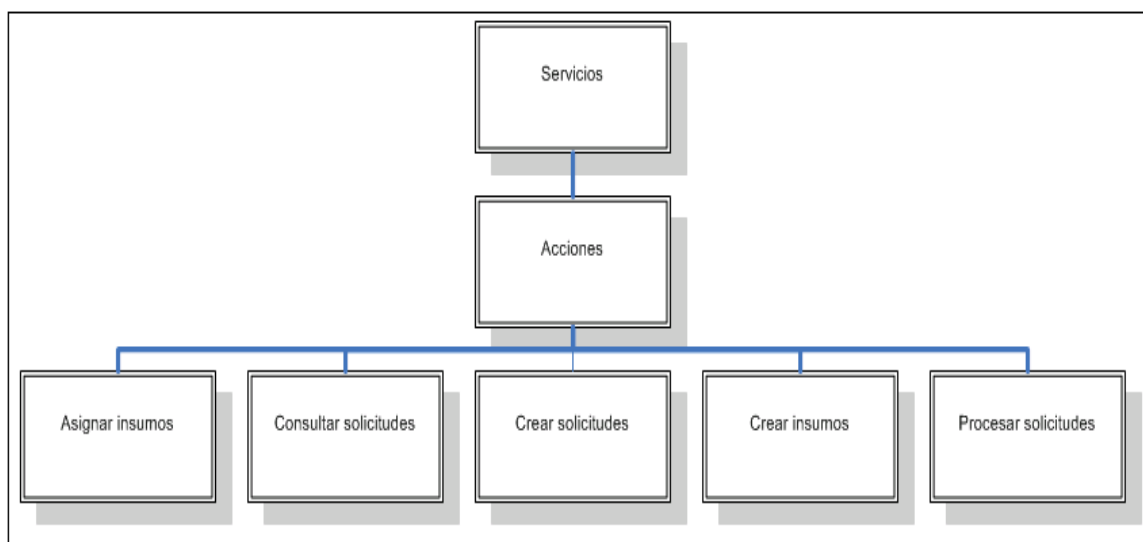


Figura 4.24 Diagrama Navegacional: Servicios.

Descripción: El usuario permitido tendrá acceso a los 5 sub-módulos del módulo de servicios del sistema.

4.1.3.3 DIAGRAMA DE CLASES UML

INGRESO AL APLICATIVO

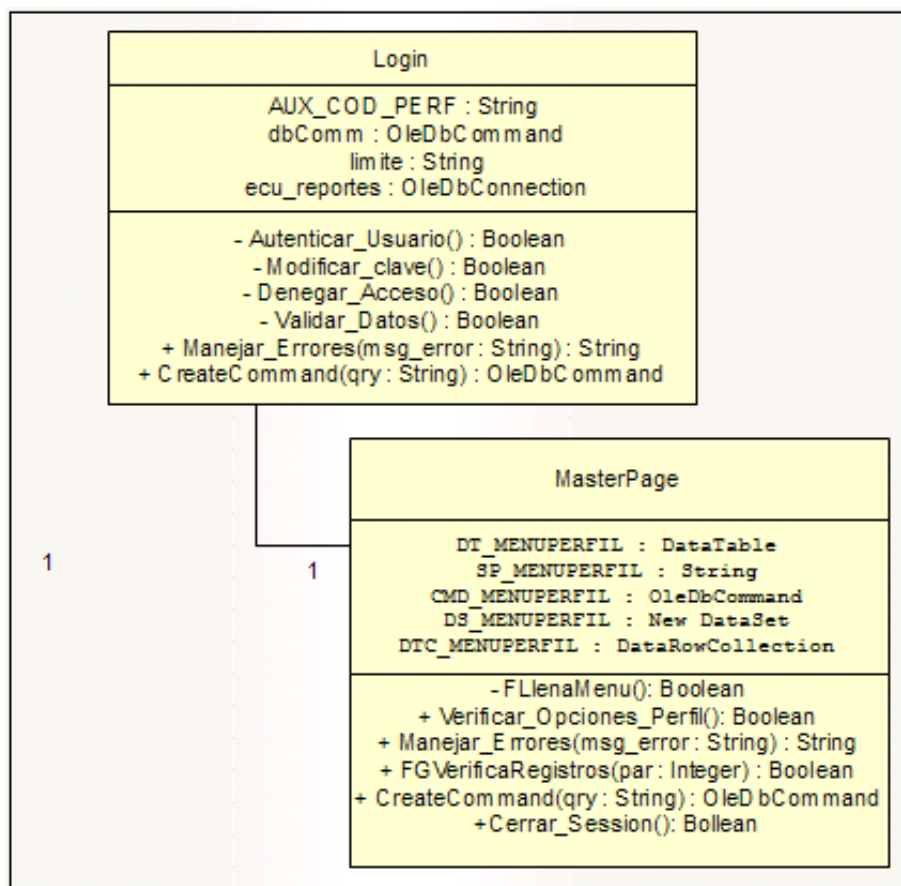


Figura 4.25 Diagrama de clases: Ingresar al aplicativo.
Descripción de variables a ser utilizadas en las clases login y master page.

MODULO ADMINISTRACION

- GESTIONAR USUARIOS

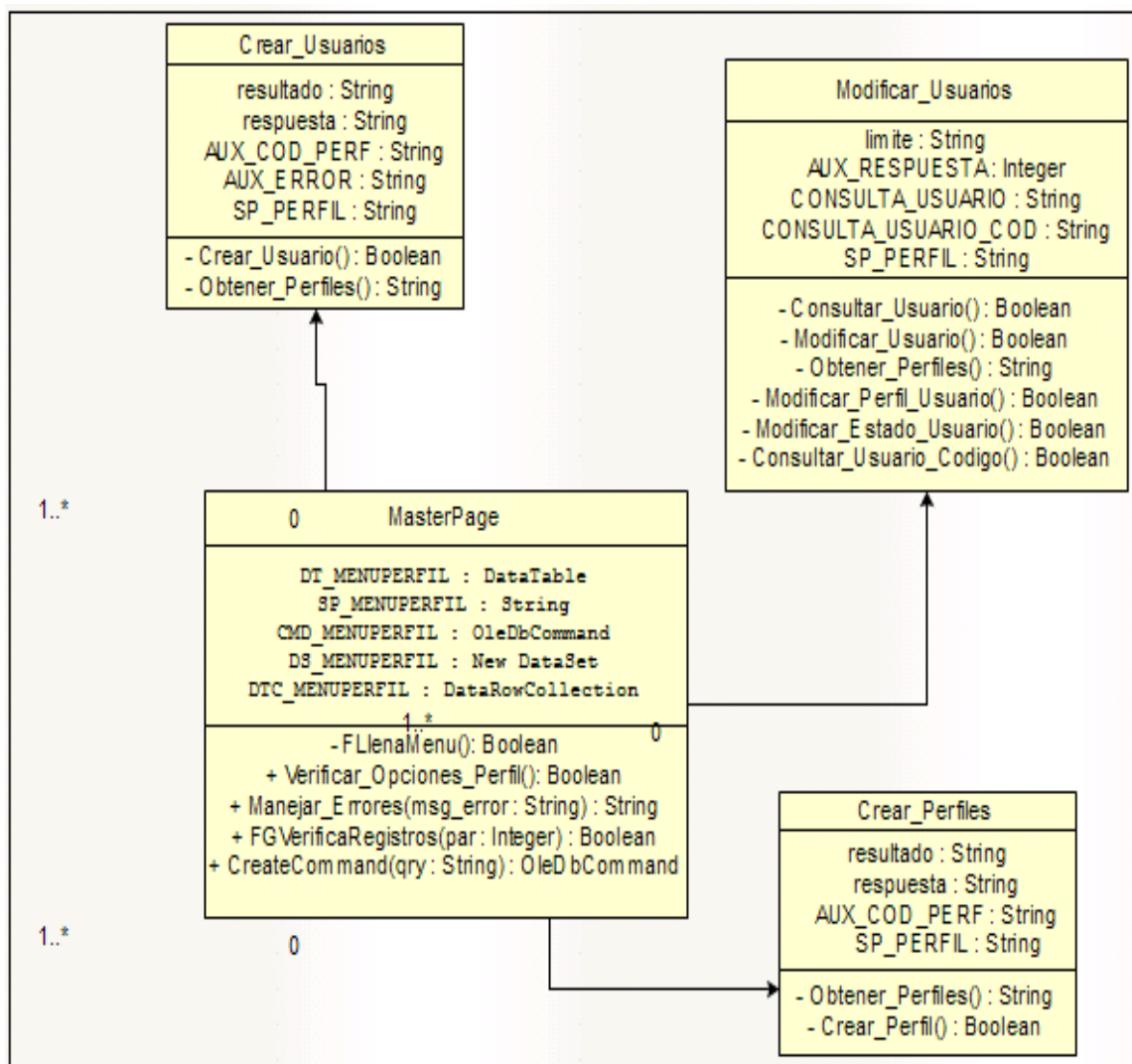


Figura 4.26 Diagrama de clases: Gestionar usuarios.

- **ACTUALIZAR BODEGA**

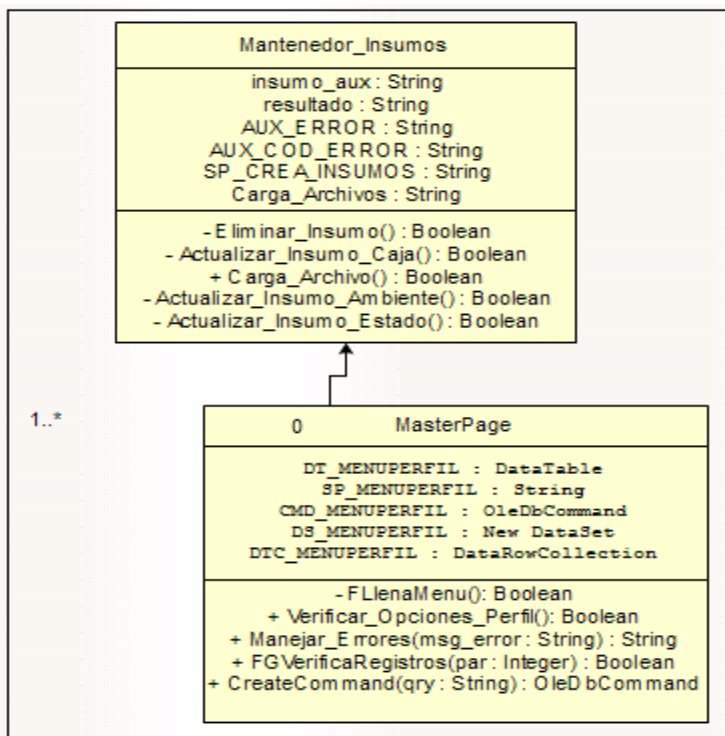


Figura 4.27 Diagrama de clases: Actualizar bodega.

MODULO REPORTES

- VERIFICAR STOCK ACTUAL

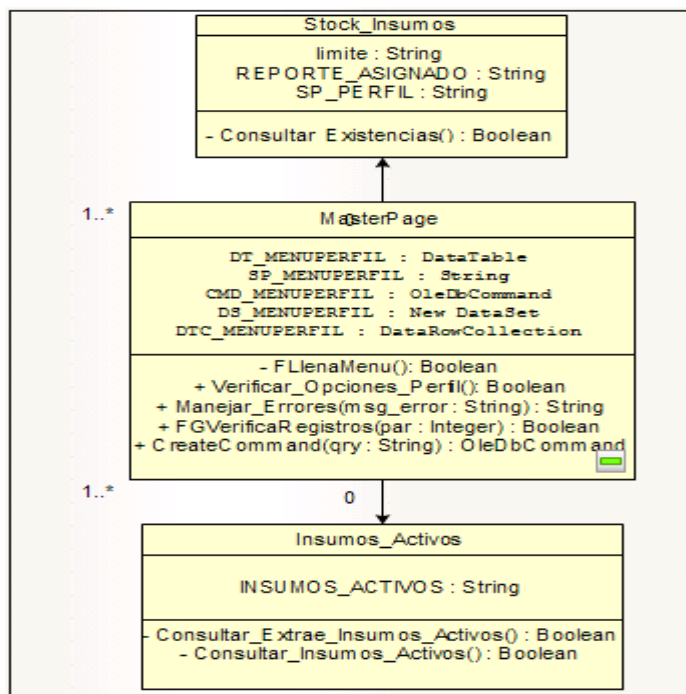


Figura 4.28 Diagrama de clases: Verificar stock actual.

- **OBTENER TRANSACCIONES POR EXISTENCIA**

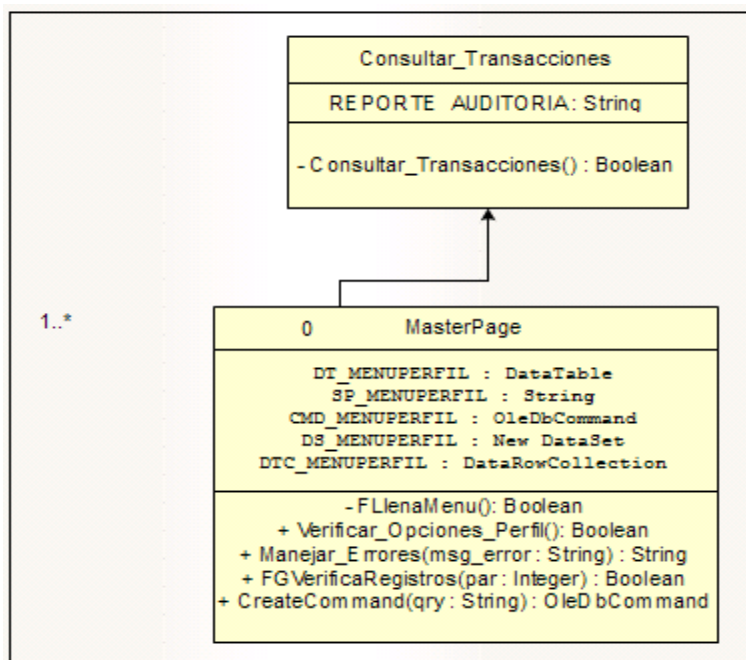


Figura 4.29 Diagrama de clases: Obtener transacciones por existencia.

- **GESTIONAR INFORMACION**

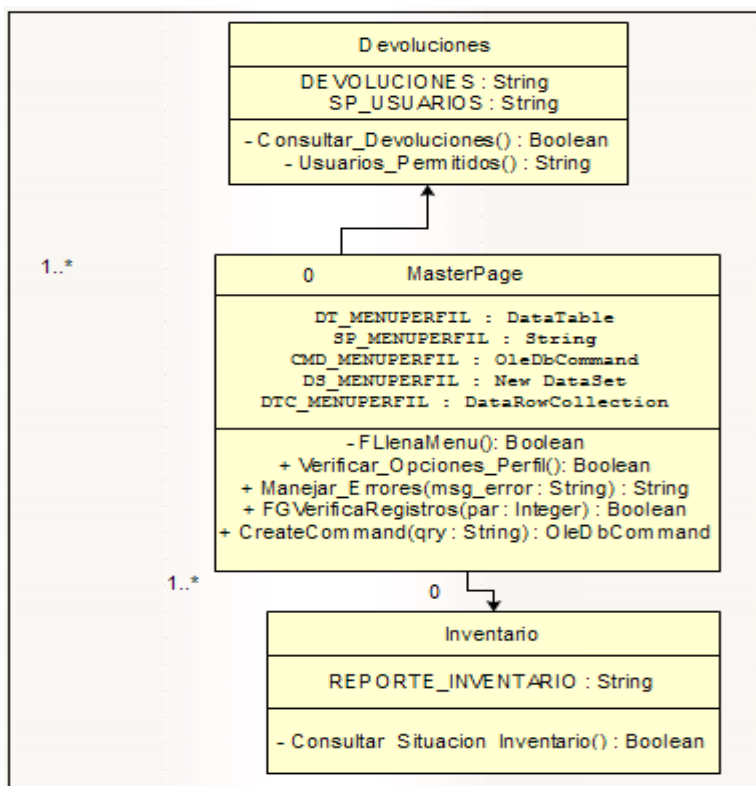


Figura 4.30 Diagrama de clases: Gestionar Información devoluciones.

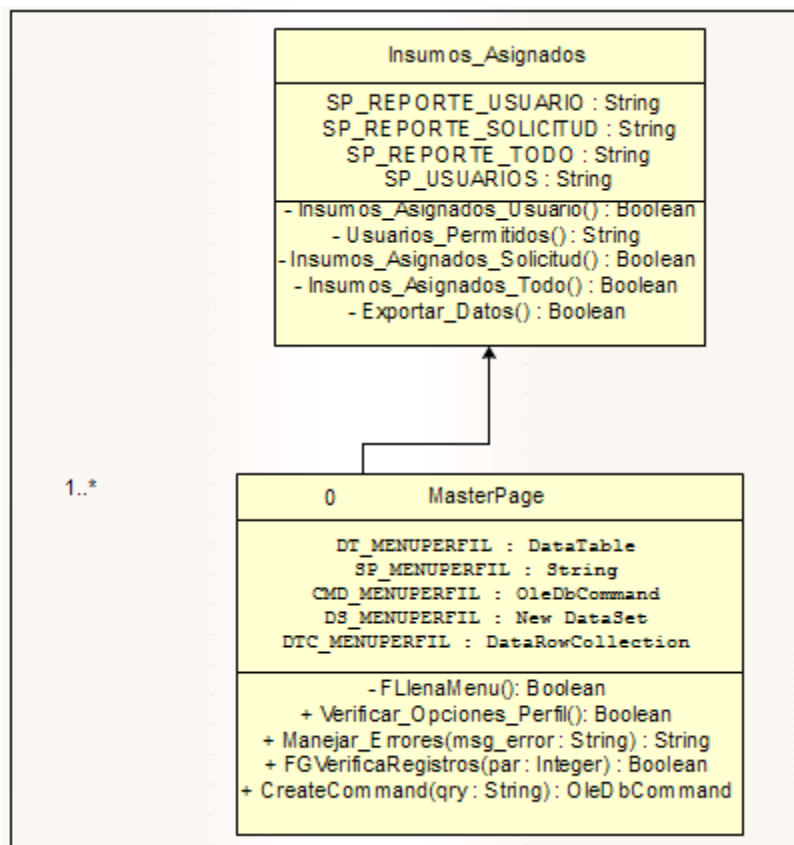


Figura 4.31 Diagrama de clases: Gestionar Información Insumos asignados.

MODULO SERVICIOS

- GESTIONAR SOLICITUDES

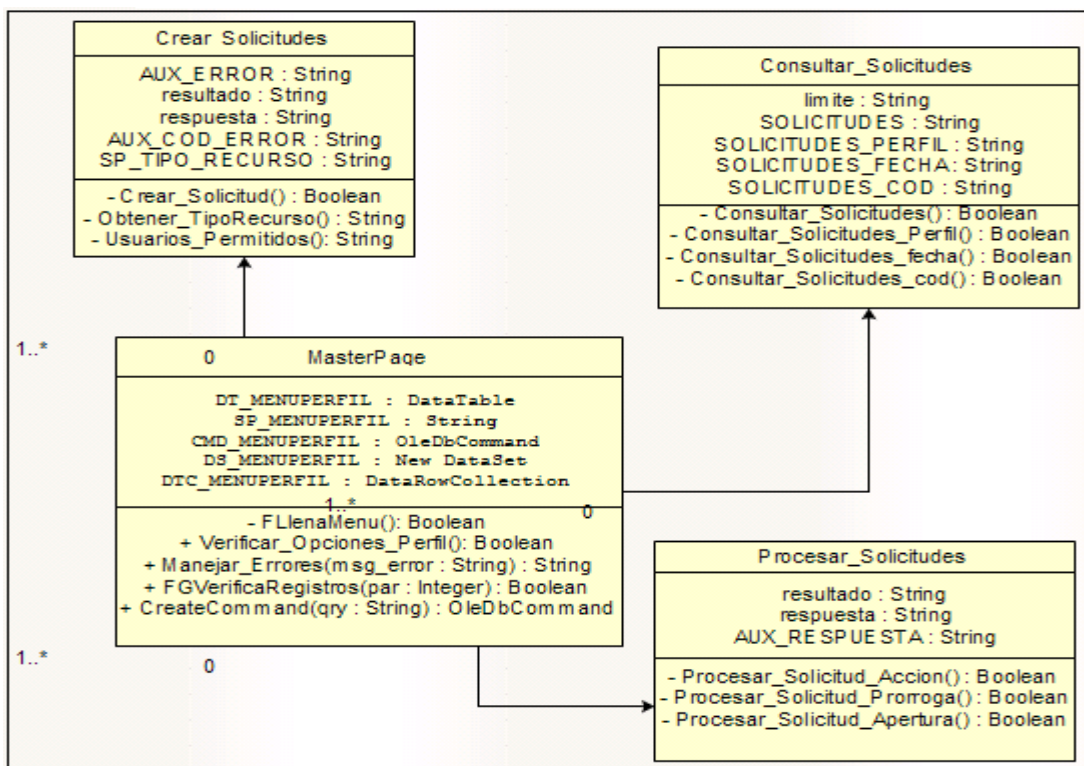


Figura 4.32 Diagrama de clases: Gestionar Solicitudes.

• GESTIONAR INSUMOS

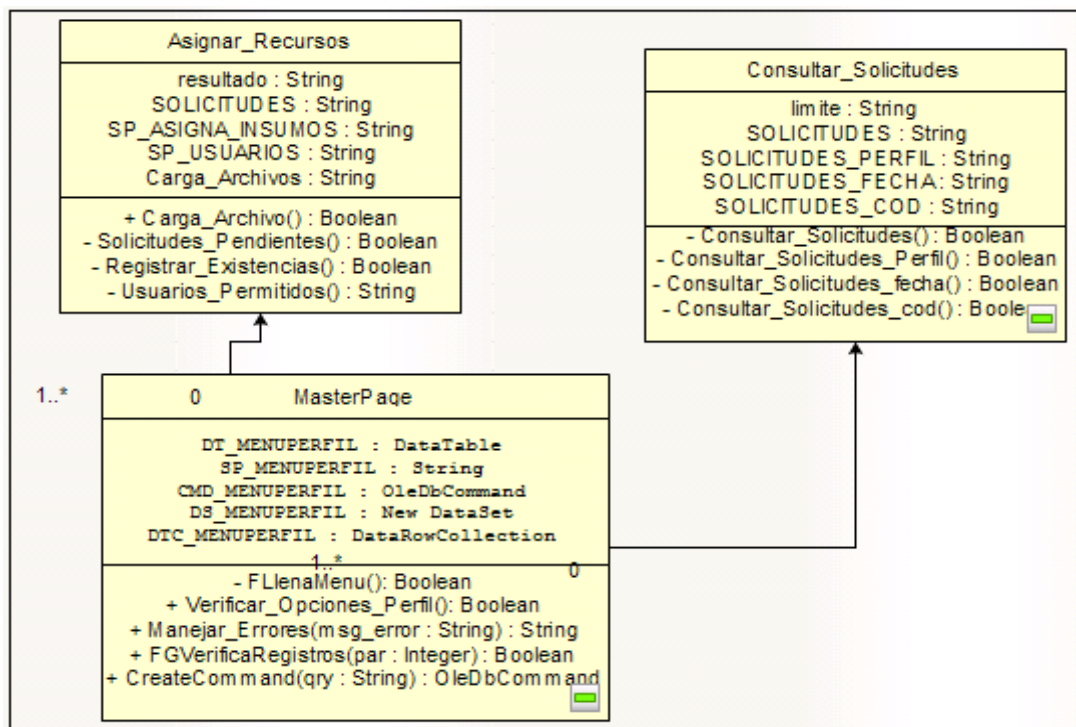


Figura 4.33 Diagrama de clases: Gestionar insumos.

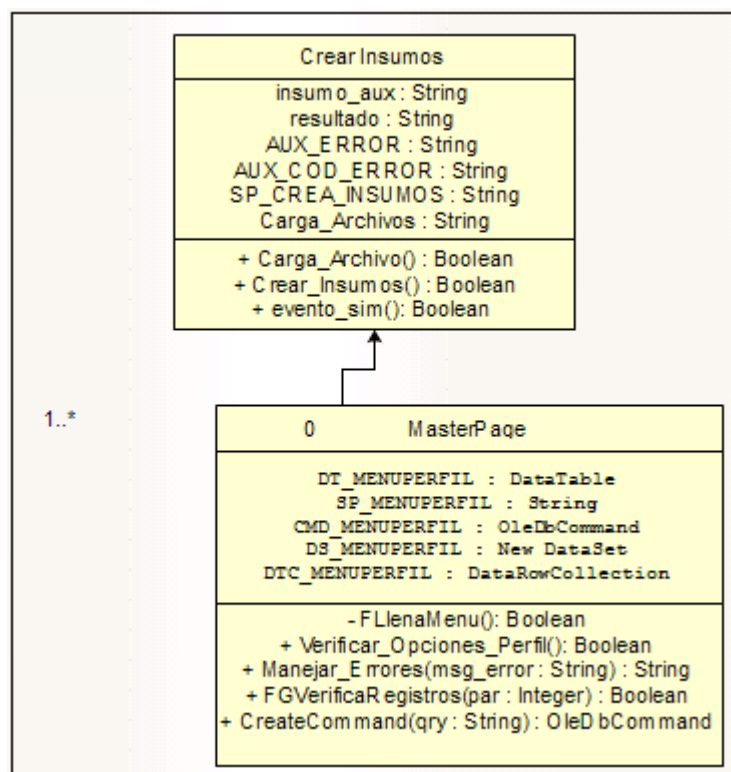


Figura 4.34 Diagrama de clases: Crear insumos.

4.1.3.4 DIAGRAMA DE SECUENCIA UML

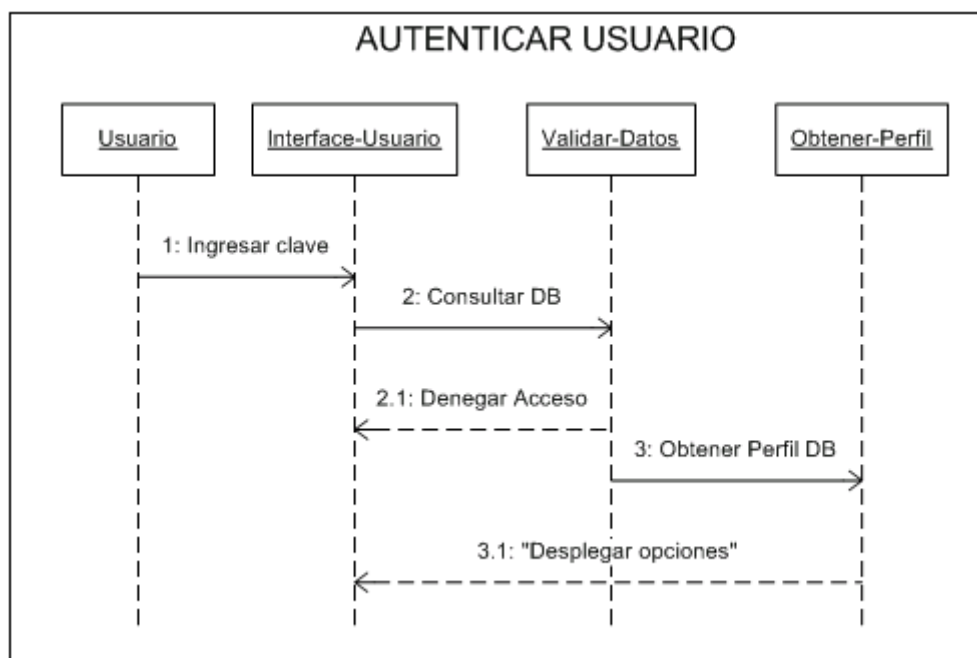


Figura 4.35 Diagrama de secuencia, Caso de uso: Autenticar usuario. Usuario ingresa parámetros de entrada y el sistema verifica, retornando las opciones por perfil o mensaje de error.

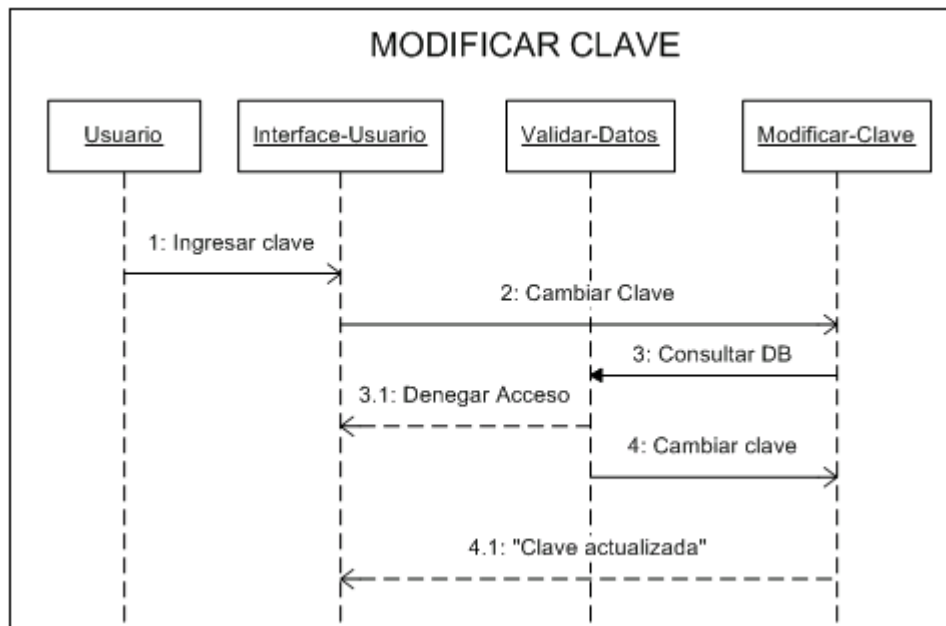


Figura 4.36 Diagrama de secuencia, Caso de uso: Modificar clave. Usuario solicita al sistema un cambio de clave, ingresa parámetros de entrada. Retorna confirmación exitosa o error.

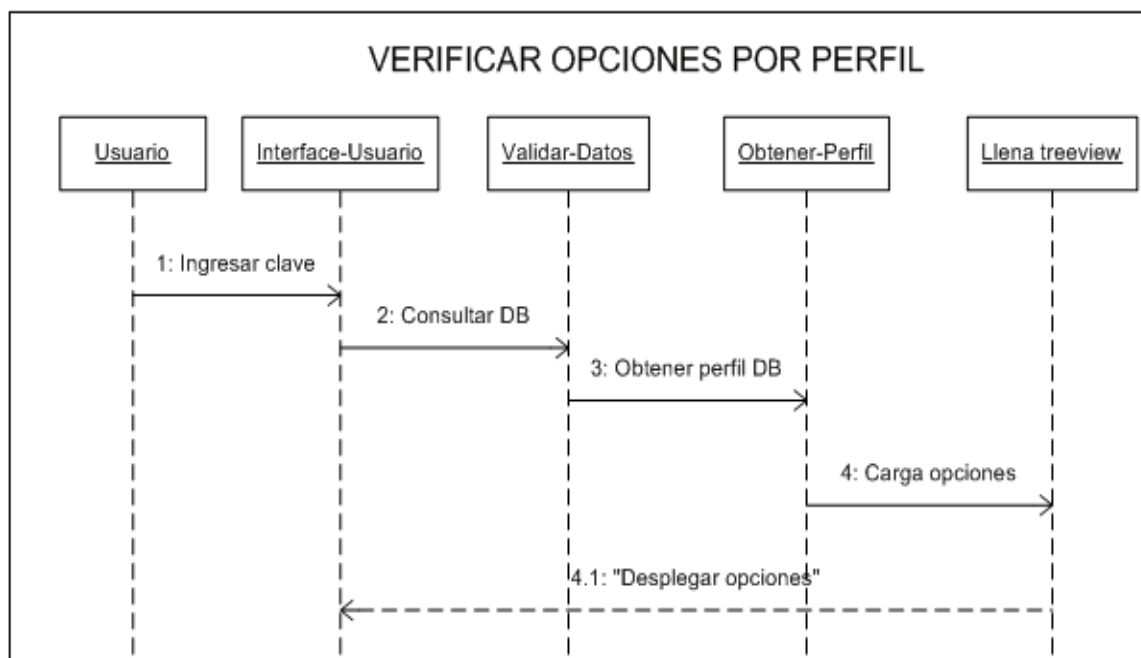


Figura 4.37 Diagrama de secuencia, Caso de uso: Verificar opciones por perfil. Usuario ingresa usuario y clave, el sistema verifica el perfil y las opciones asociadas al perfil. Publica el contenido de las opciones en un objeto TreeView.

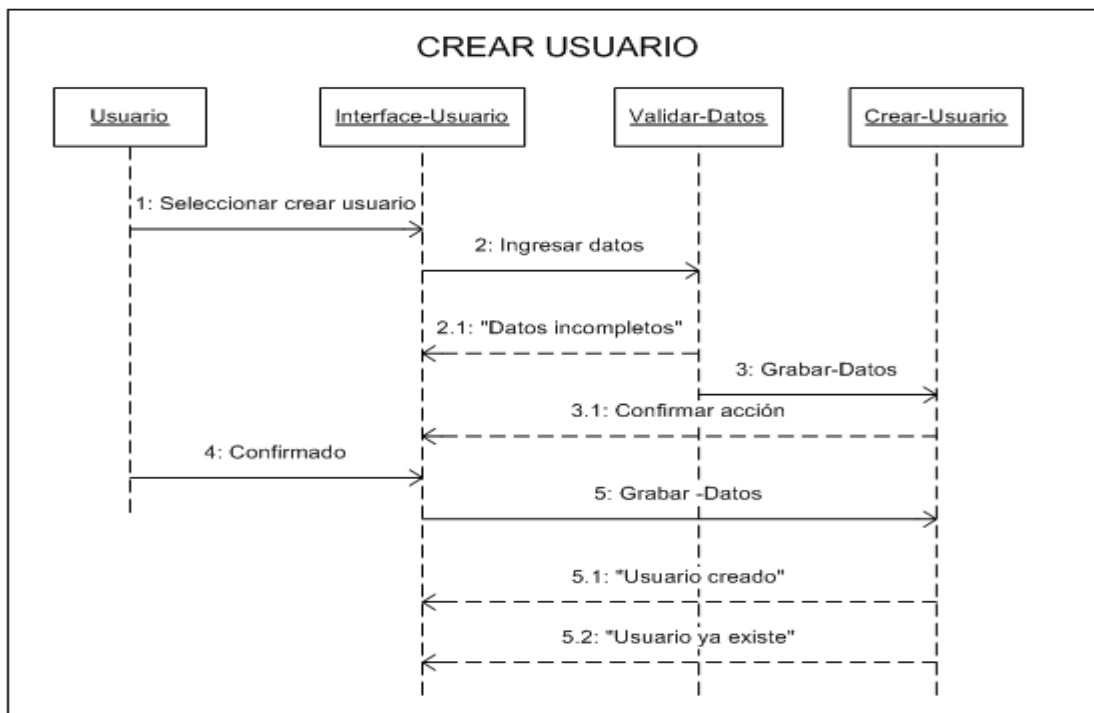


Figura 4.38 Diagrama de secuencia, Caso de uso: Crear usuario. Usuario ingresa parámetros de entrada requeridos para crear un usuario. Sistema retorna confirmación exitosa o mensajes de error / validación.

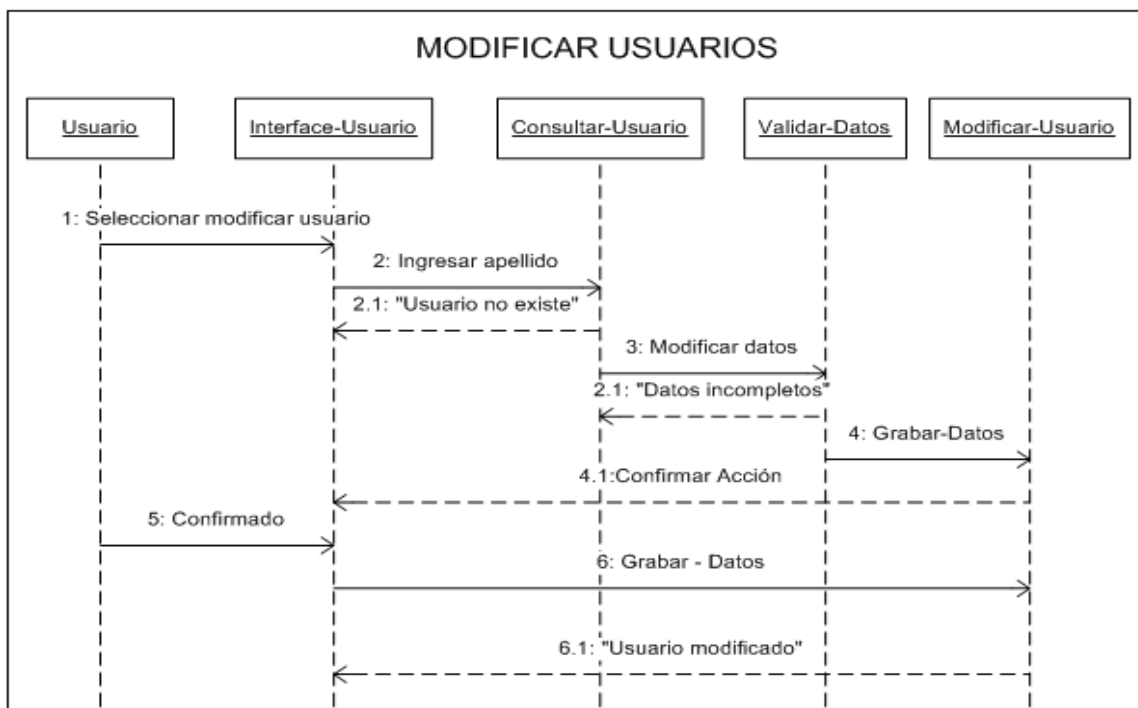


Figura 4.39 Diagrama de secuencia, Caso de uso: Modificar usuario. Usuario consulta un usuario, sistema carga la información actual, la cual puede ser modificada por el usuario.

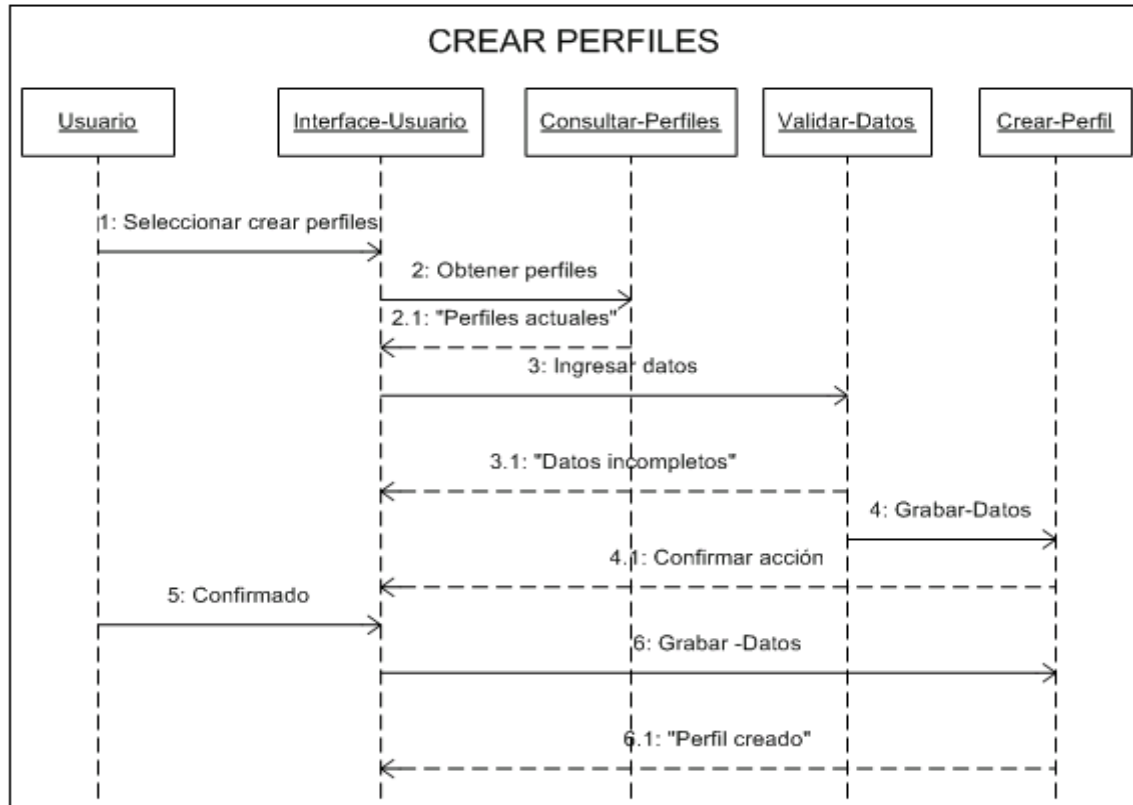


Figura 4.40 Diagrama de secuencia, Caso de uso: Crear perfiles. Usuario ingresa parámetros de entrada para crear un perfil nuevo, previamente visualiza los existentes.

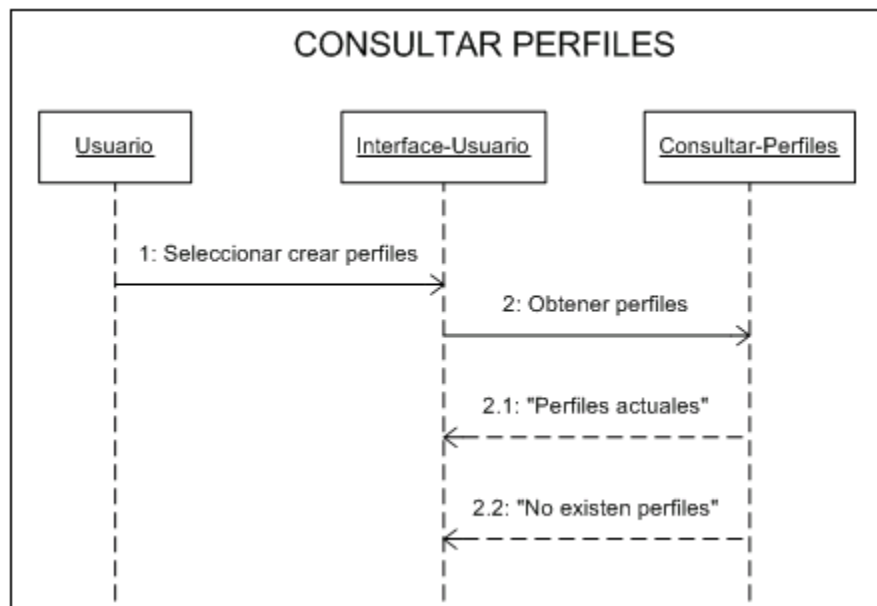


Figura 4.41 Diagrama de secuencia, Caso de uso: Consultar perfiles. Sistema presenta los perfiles existentes y vigentes.

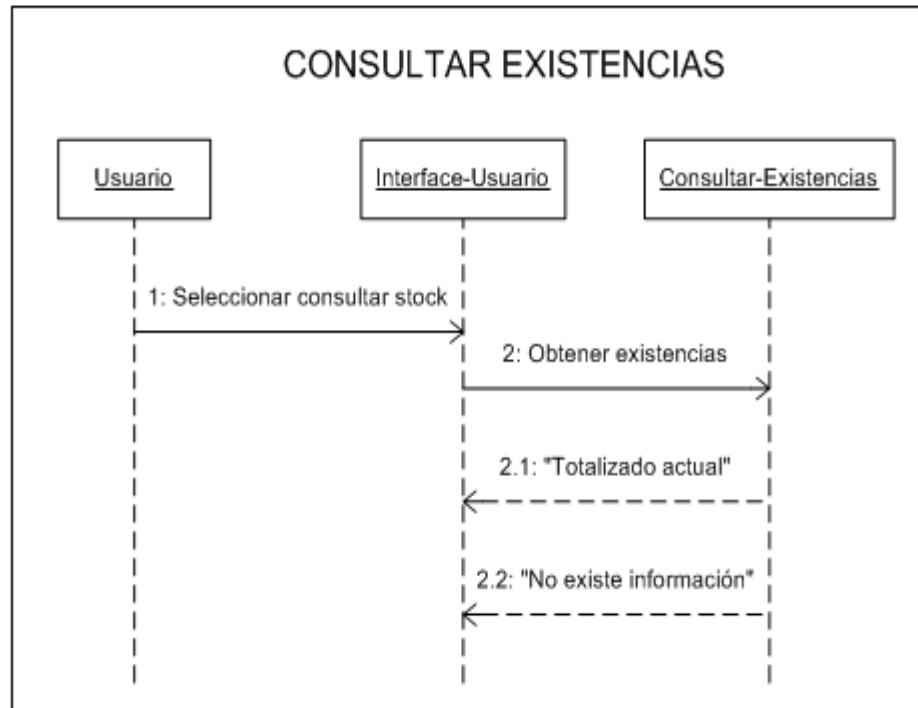


Figura 4.42 Diagrama de secuencia, Caso de uso: Consultar existencias. Usuario solicita al sistema la información actual, presenta una tabla con la información actual.

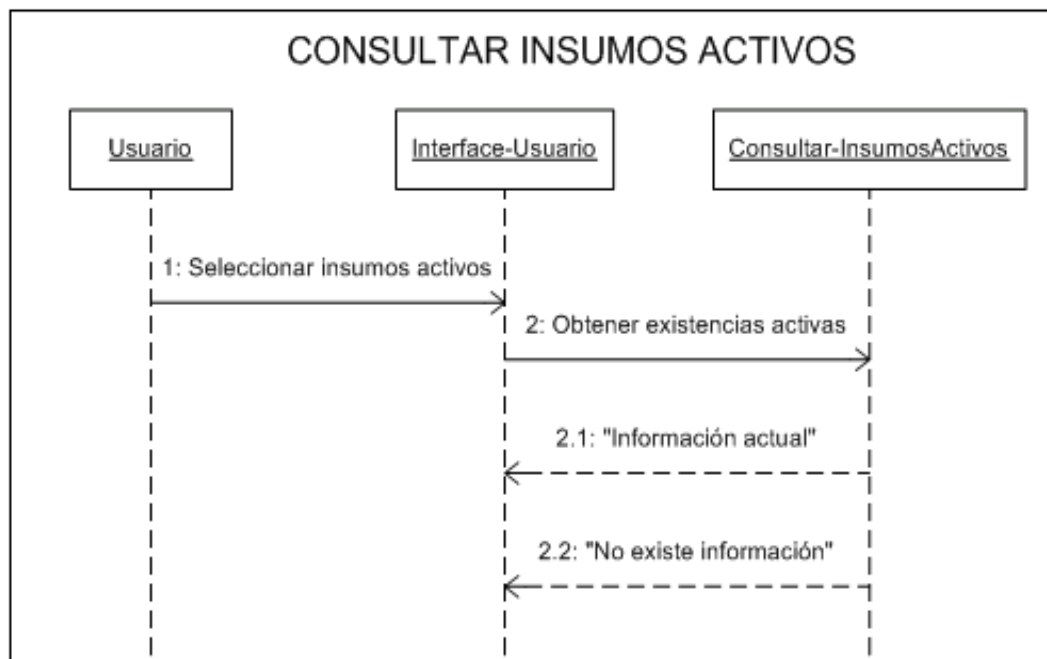


Figura 4.43 Diagrama de secuencia, Caso de uso: Consultar insumos activos. Usuario solicita reporte actual del estado de los registros en el sistema.

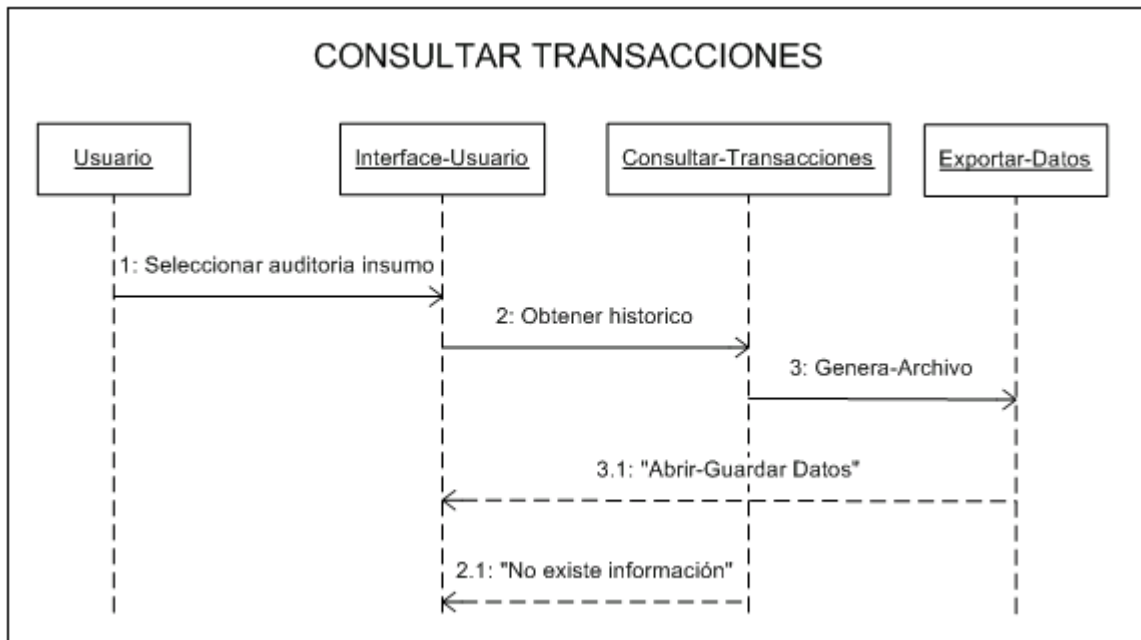


Figura 4.44 Diagrama de secuencia, Caso de uso: Consultar transacciones. Usuario realiza consulta de históricos en el sistema, ingresa los parámetros de entrada definidos.

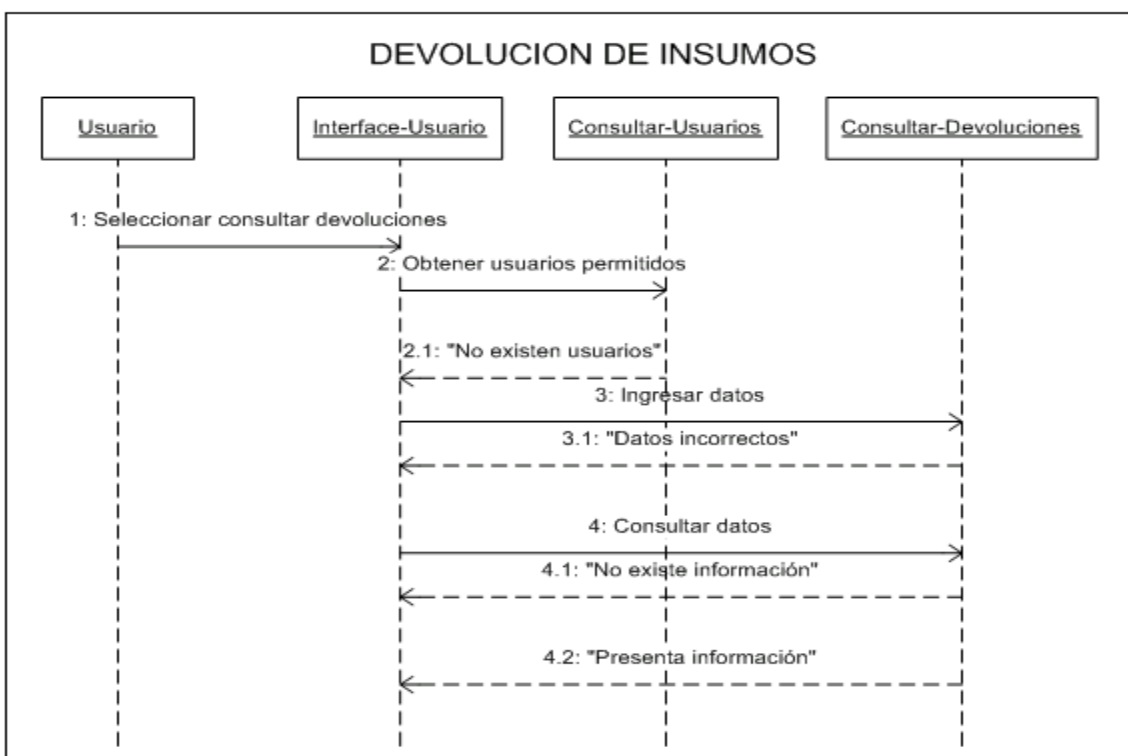


Figura 4.45 Diagrama de secuencia, Caso de uso: Devolución de insumos. Usuario ingresa los insumos devueltos por los usuarios, ingresa los parámetros de entrada permitidos.

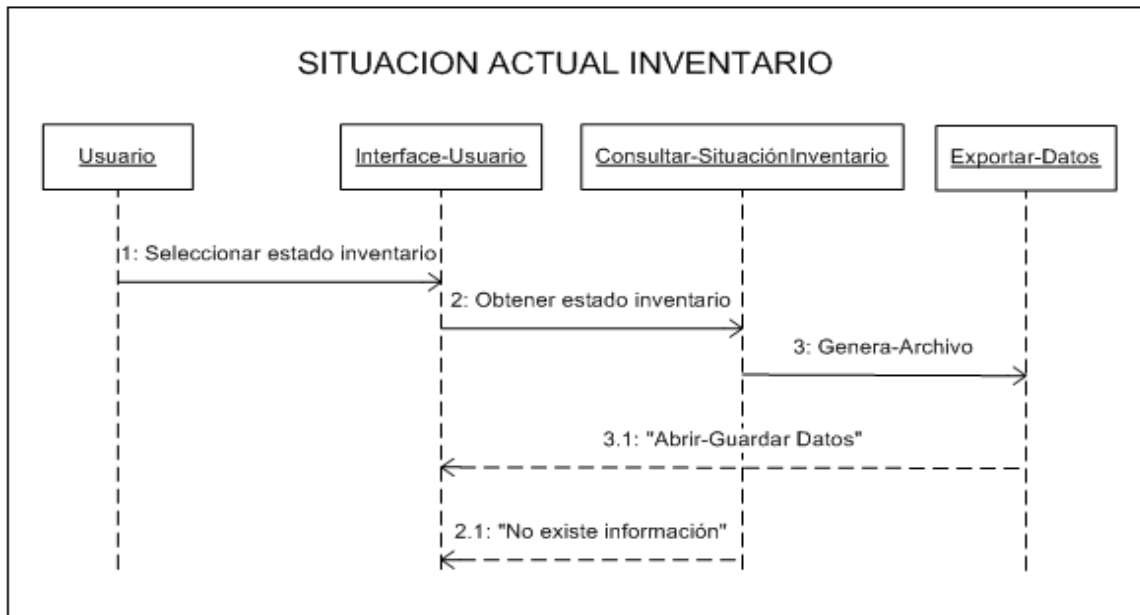


Figura 4.46 Diagrama de secuencia, Caso de uso: Situación actual inventario. Usuario solicita un reporte al sistema, con el estado actual de los insumos. Se genera un reporte en Excel.

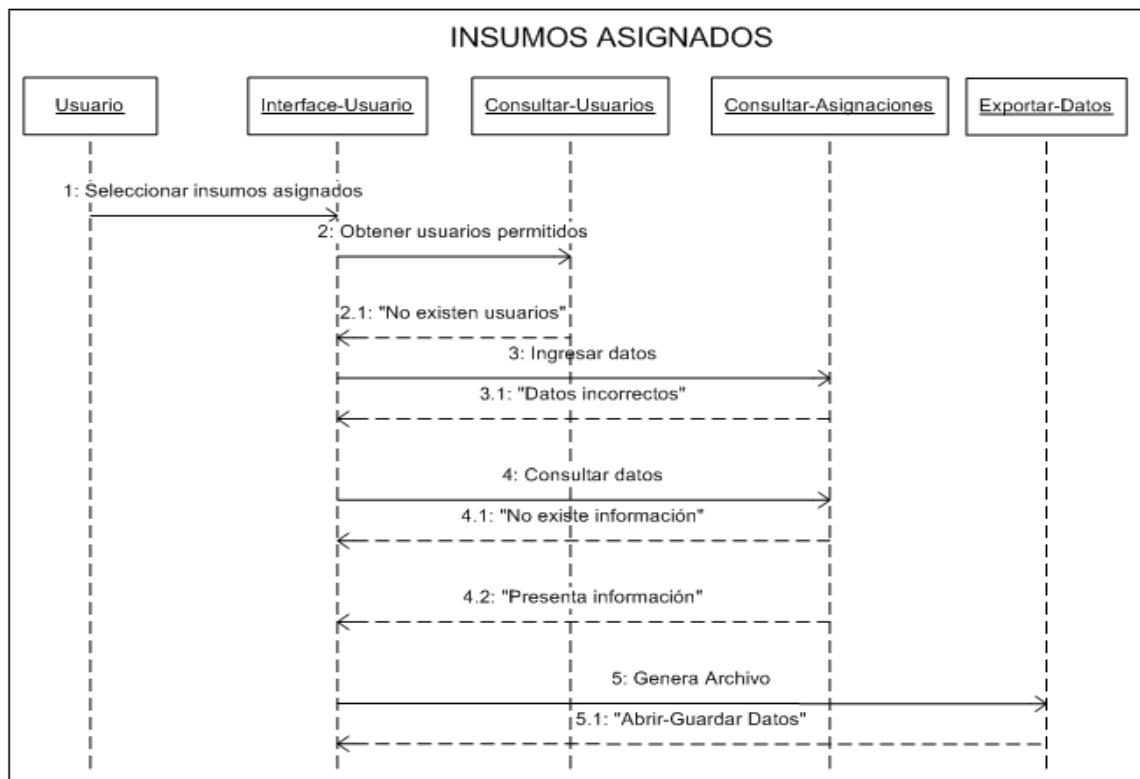


Figura 4.47 Diagrama de secuencia, Caso de uso: Insumos asignados. Usuario realiza consulta de insumos asignados a los usuarios, ingresa los parámetros de entrada permitidos.

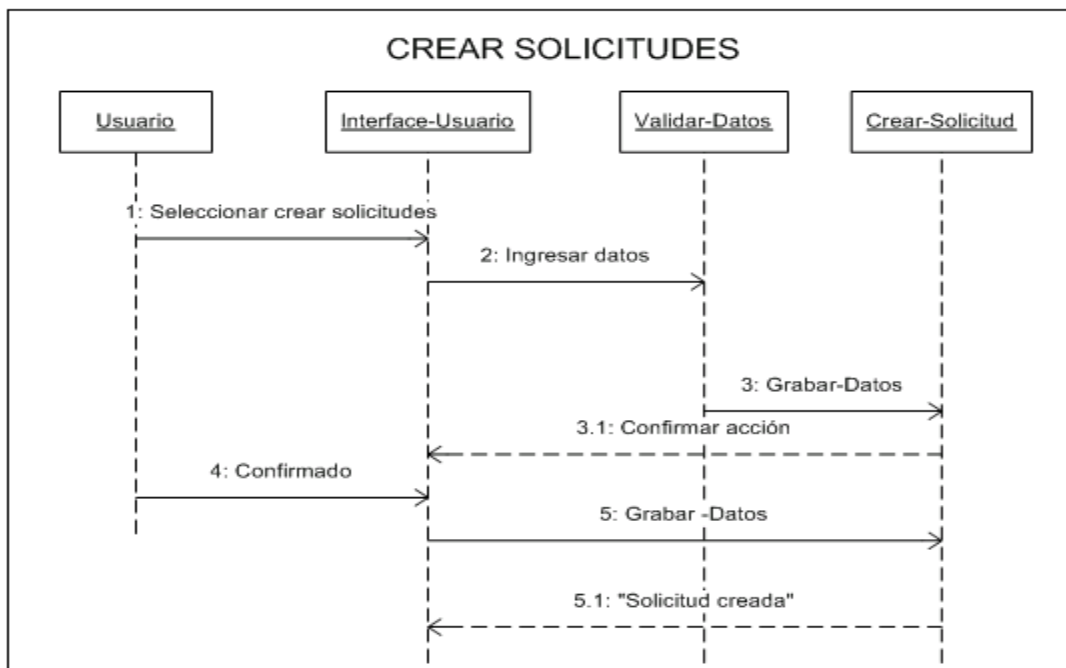


Figura 4.48 Diagrama de secuencia, Caso de uso: Crear solicitudes. Usuario permitido crea las solicitudes para asignación re recursos, el sistema retorna un número de solicitud.

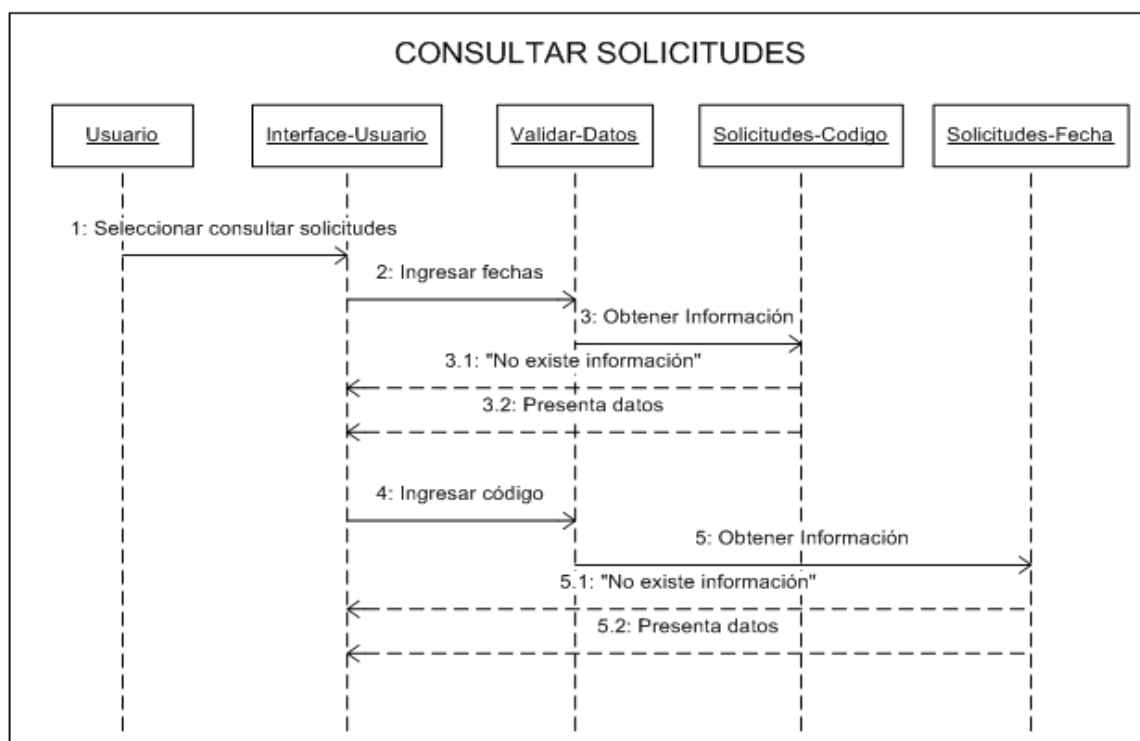


Figura 4.49 Diagrama de secuencia, Caso de uso: Consultar solicitudes. Usuario consulta solicitudes en el sistema, ingresa parámetros disponibles.

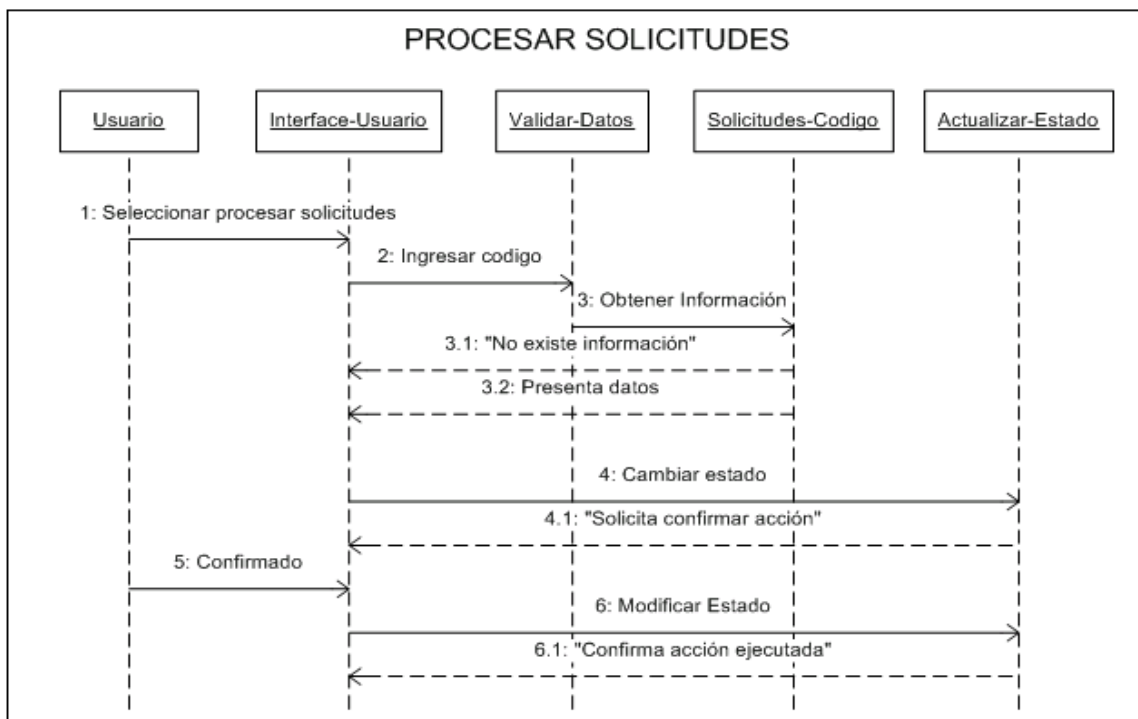


Figura 4.50 Diagrama de secuencia, Caso de uso: Procesar solicitudes. Usuario realiza la asignación de recursos y cierra el proceso cerrando la solicitud.

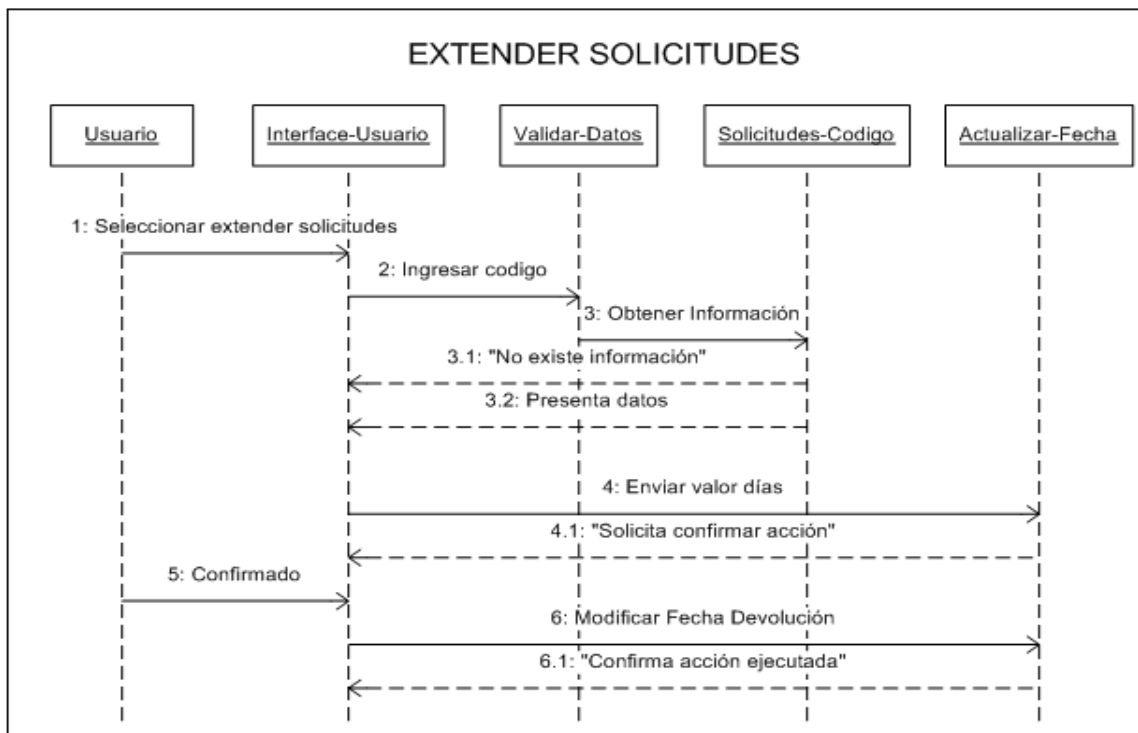


Figura 4.51 Diagrama de secuencia, Caso de uso: Extender solicitudes. Usuario modifica la fecha fin de entrega de un insumo.

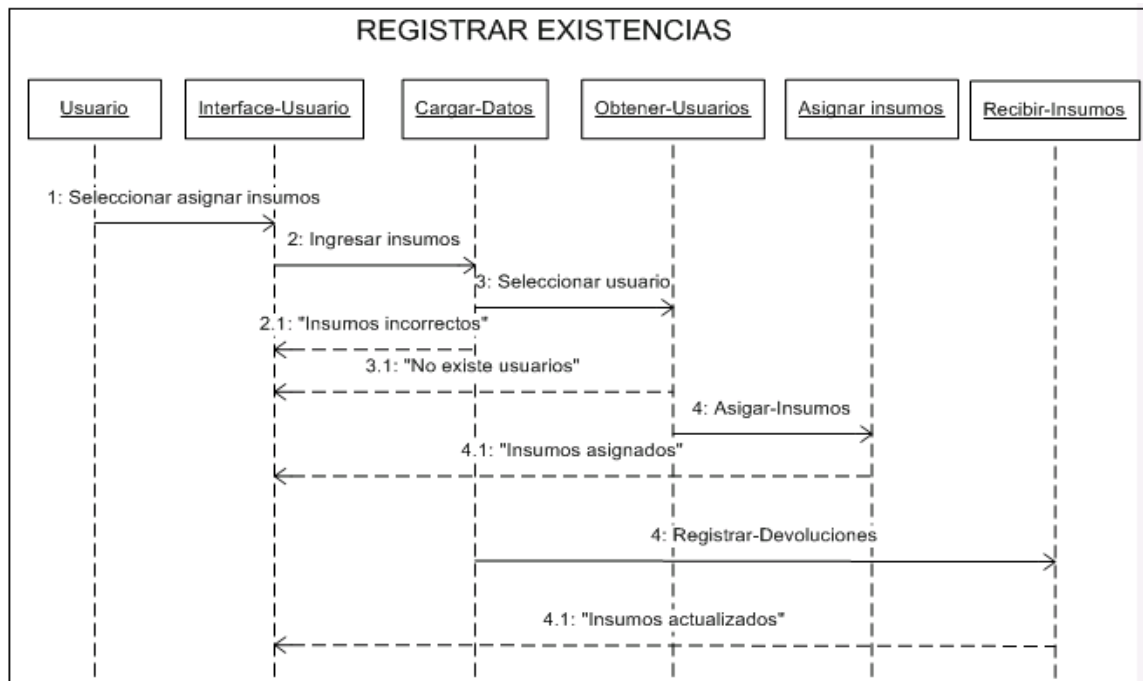


Figura 4.52 Diagrama de secuencia, Caso de uso: Registrar existencias. Usuario realiza la actualización de datos a través de este proceso.

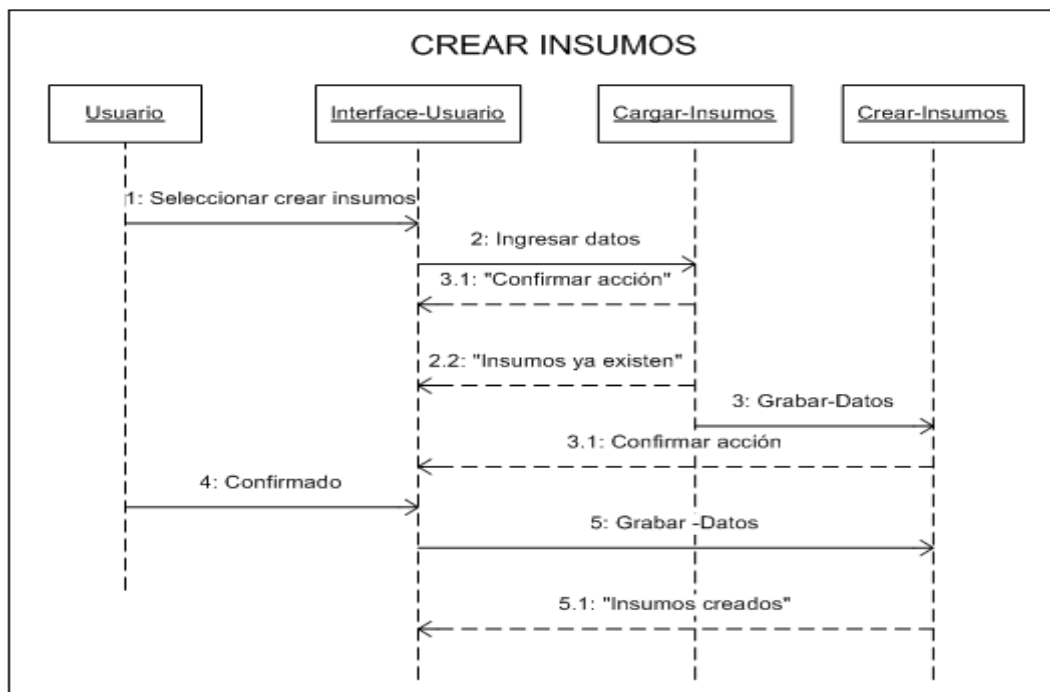


Figura 4.53 Diagrama de secuencia, Caso de uso: Crear insumos. Usuario permitido crea nuevos insumos en el sistema, a través de la carga de archivos.

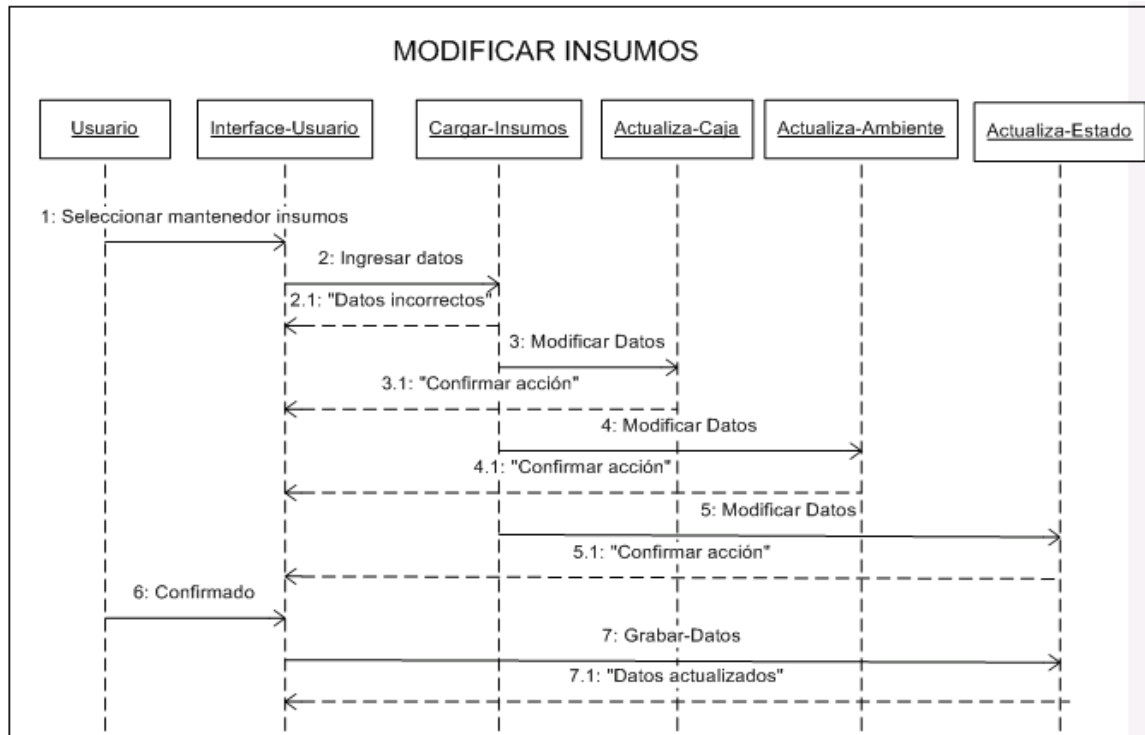


Figura 4.54 Diagrama de secuencia, Caso de uso: Modificar insumos. Usuario modifica los datos en el sistema.

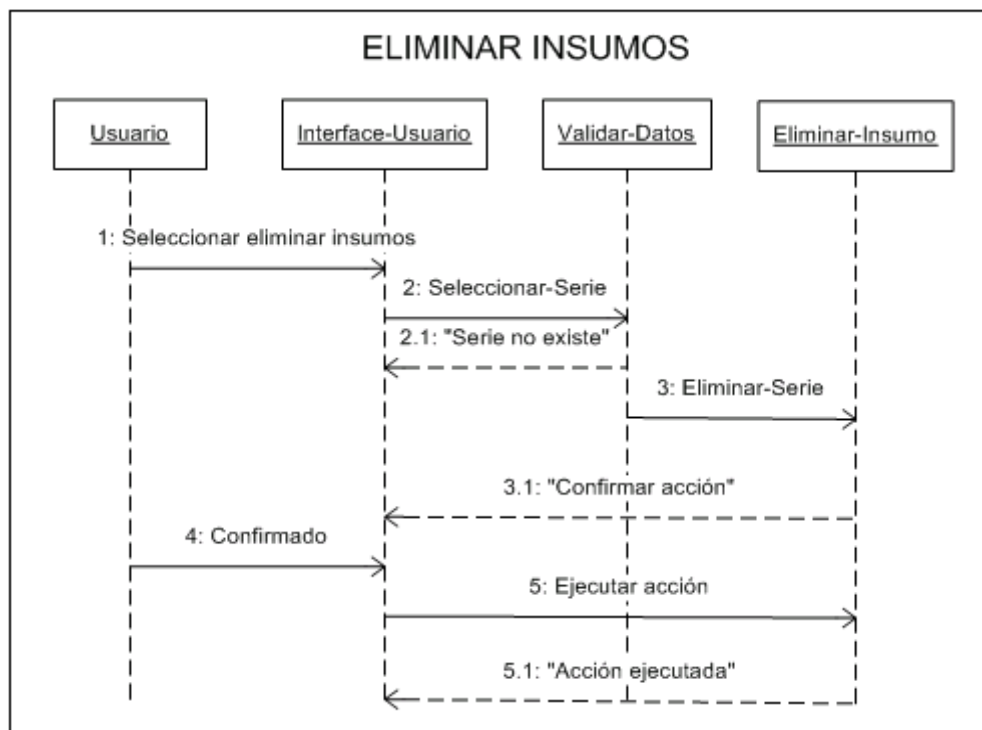


Figura 4.55 Diagrama de secuencia, Caso de uso: Eliminar insumos. Usuario elimina insumos del sistema.

4.1.3.5 DIAGRAMA DE ACTIVIDADES UML

Definir el comportamiento de los casos de uso

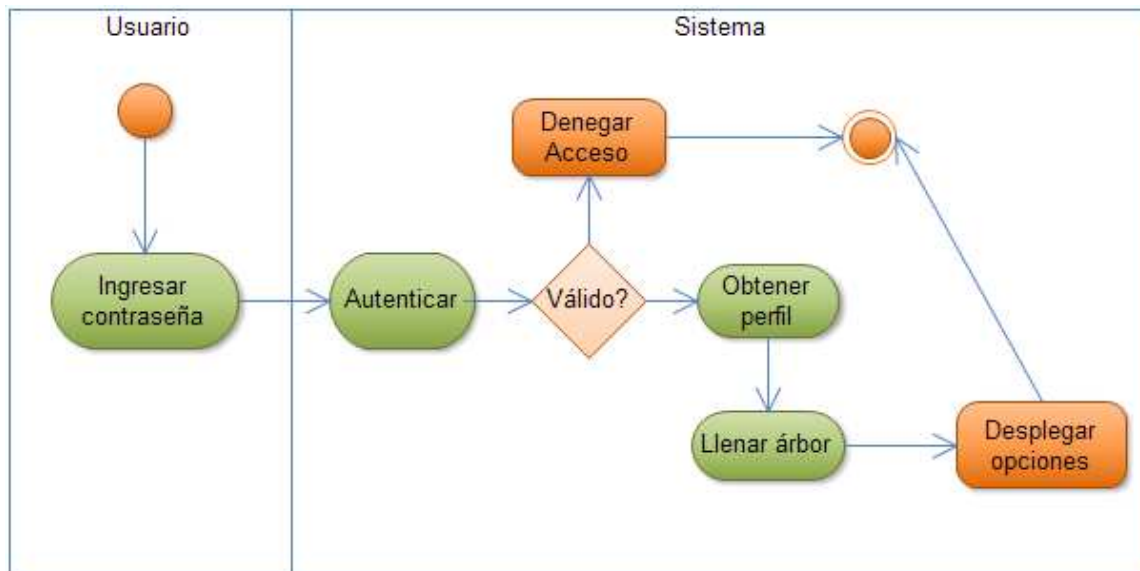


Figura 4.56 Diagrama de actividades, Caso de uso: Autenticar usuario. Usuario ingresa la clave, sistema verifica los datos y no permite al acceso o despliega las opciones para el perfil.

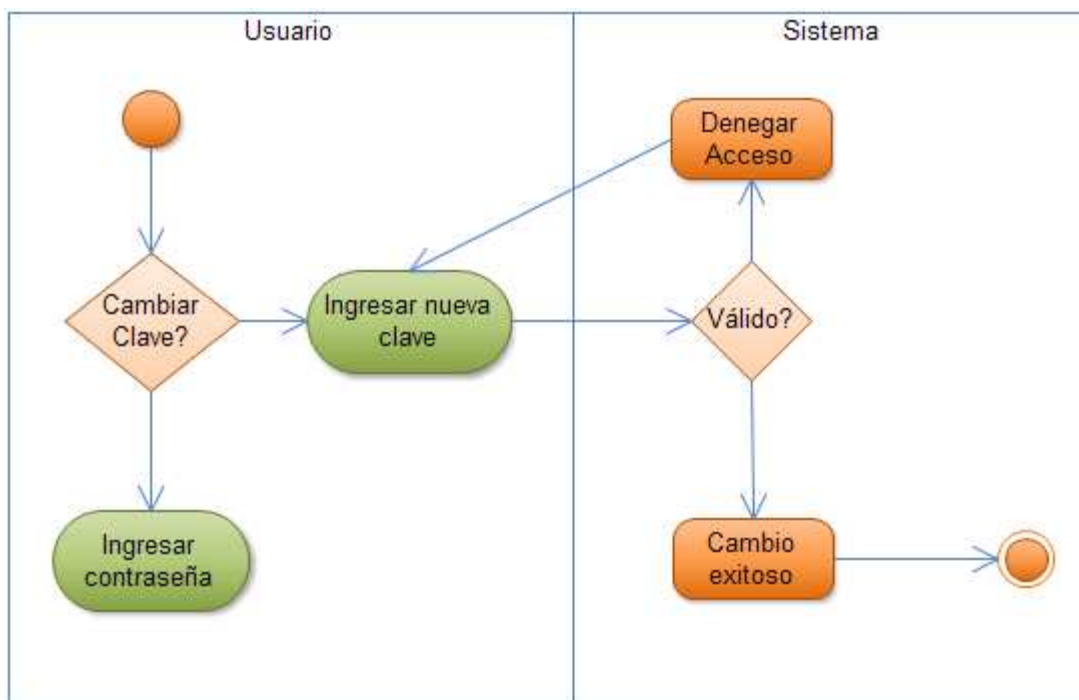


Figura 4.57 Diagrama de actividades, Caso de uso: Modificar Clave. Usuario selecciona cambio de clave, sistema valida información. Si los datos son correctos realiza el cambio.

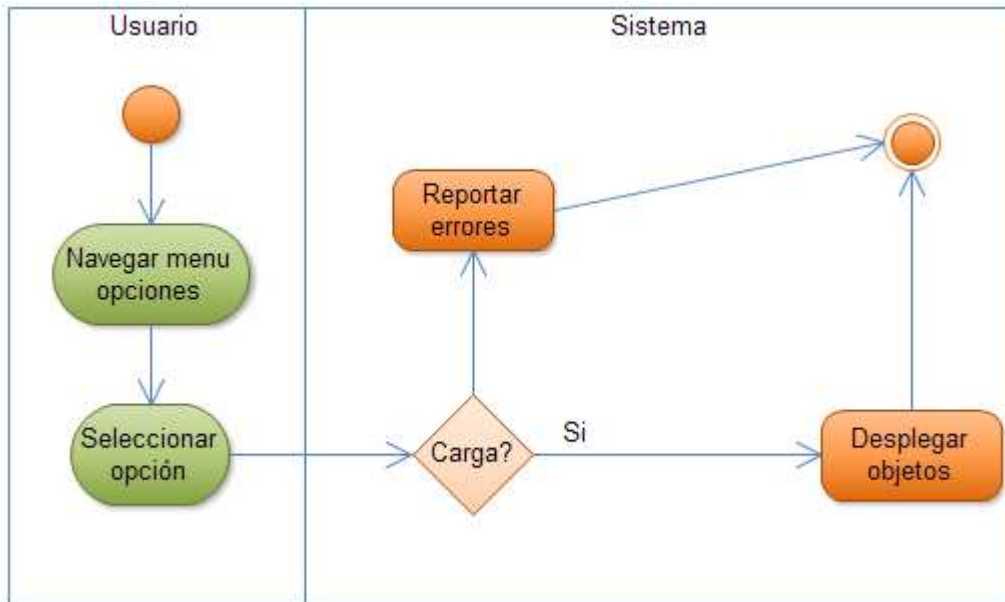


Figura 4.58 Diagrama de actividades, Caso de uso: Verificar opciones por perfil. Usuario navega por el árbol de opciones del sistema, con las opciones permitidas en su perfil.

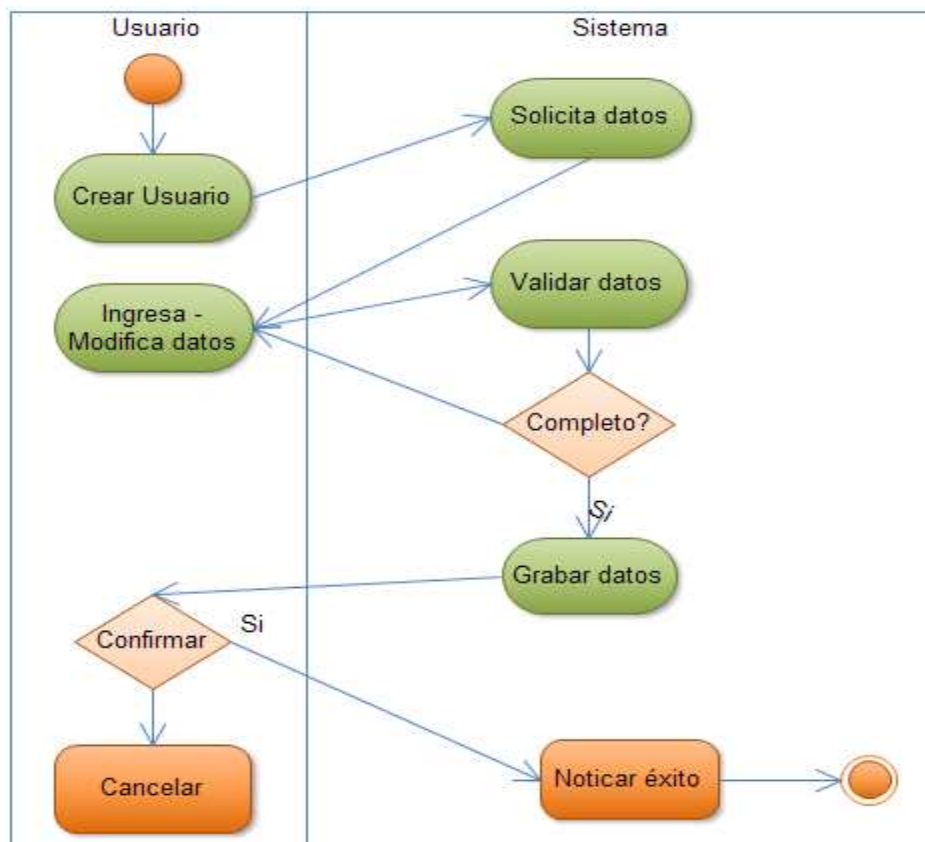


Figura 4.59 Diagrama de actividades, Caso de uso: Crear Usuario. Usuario selecciona crear usuario en el sistema.

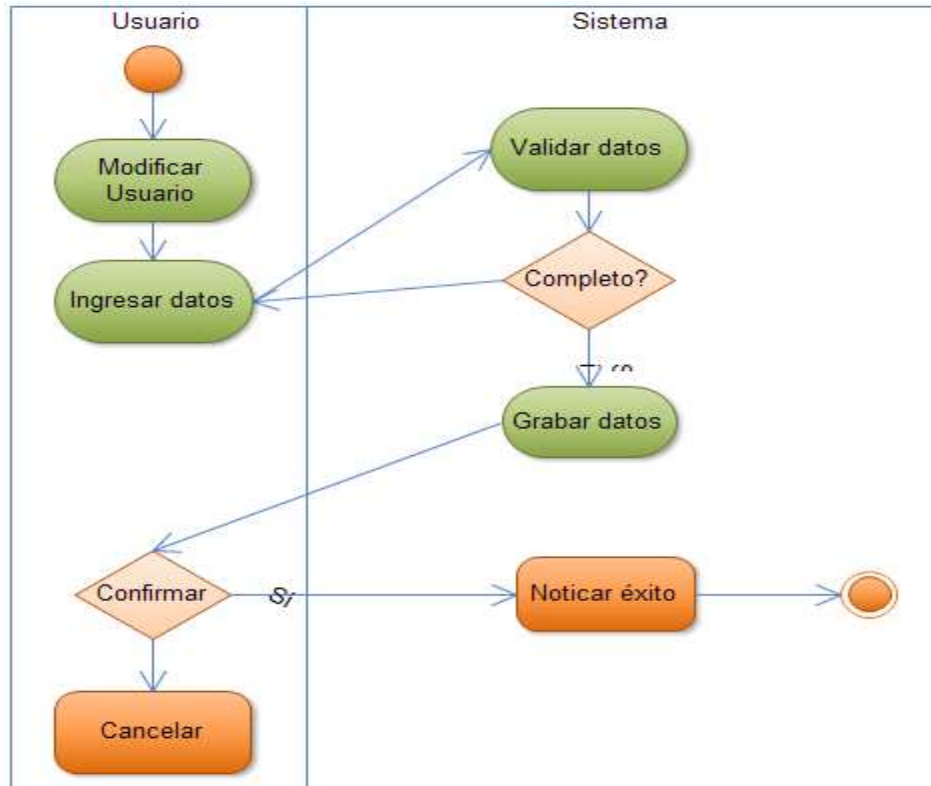


Figura 4.60 Diagrama de actividades, Caso de uso: Modificar Usuario. Usuario modifica los datos del usuario en el sistema.

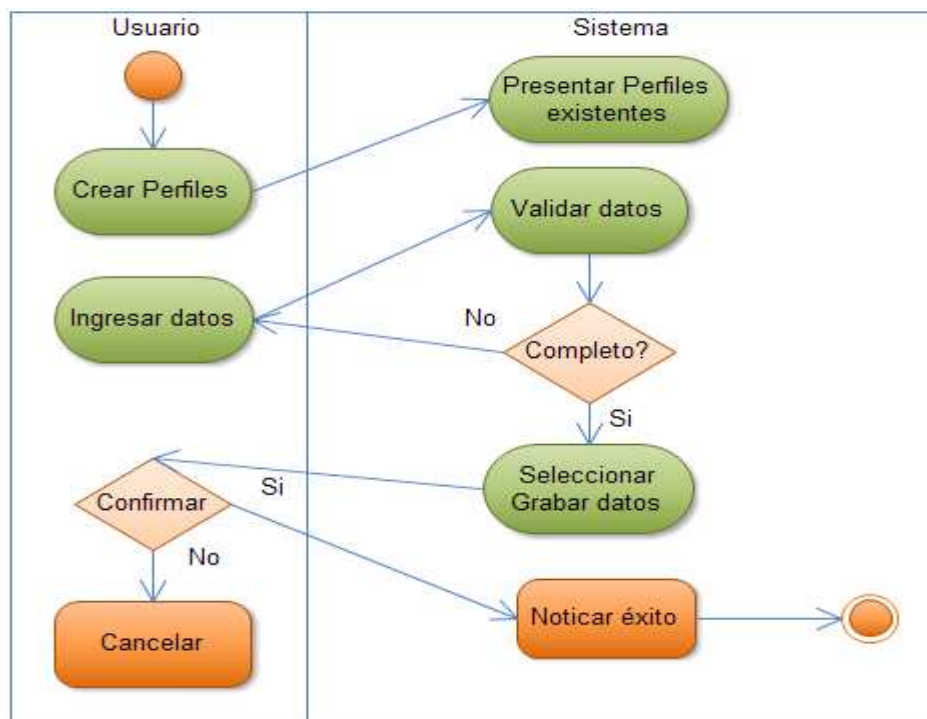


Figura 4.61 Diagrama de actividades, Caso de uso: Crear y Consultar perfiles. Usuario consulta visualiza los perfiles existentes en el sistema.

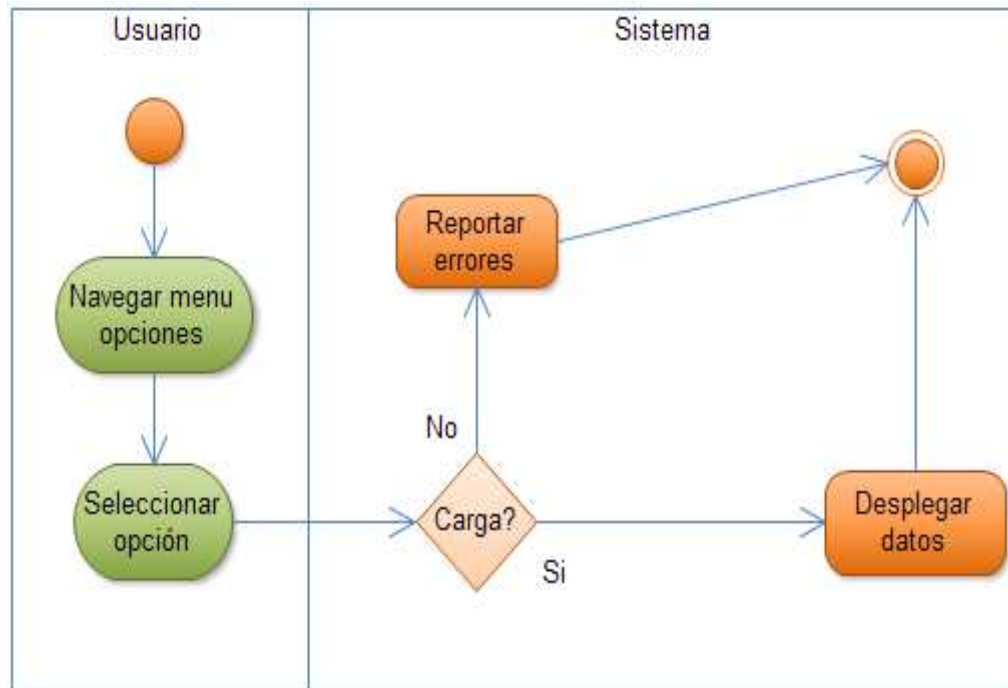


Figura 4.62 Diagrama de actividades, Caso de uso: Consultar existencias. Usuario consulta los datos existentes en el sistema.

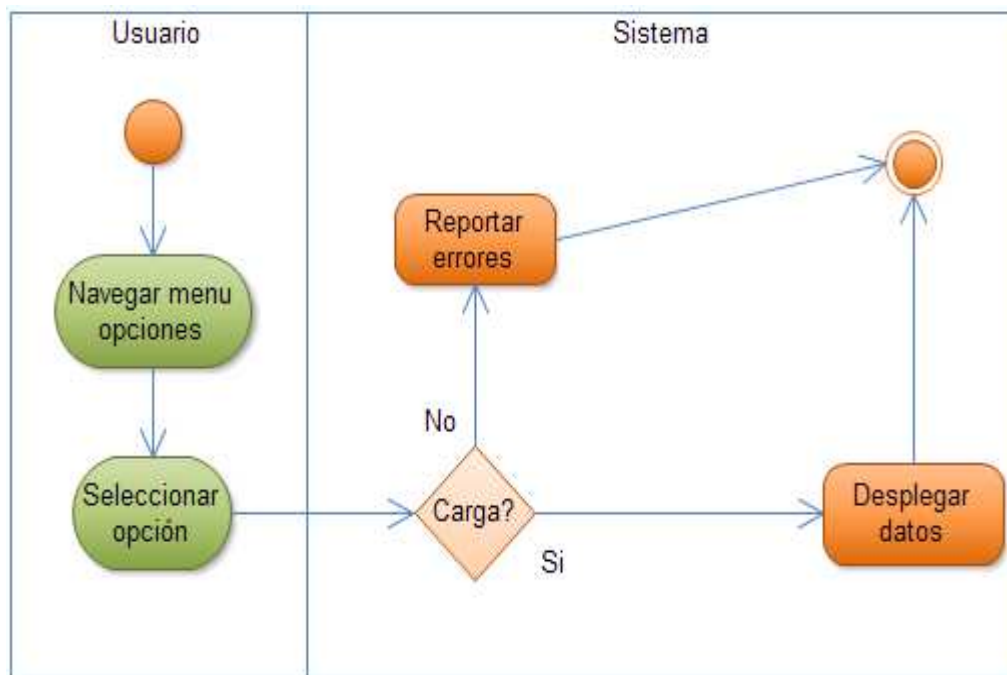


Figura 4.63 Diagrama de actividades, Caso de uso: Consultar insumos activos. Usuario selecciona reporte de insumos activos, el sistema realiza las consultas y presenta datos si existen.

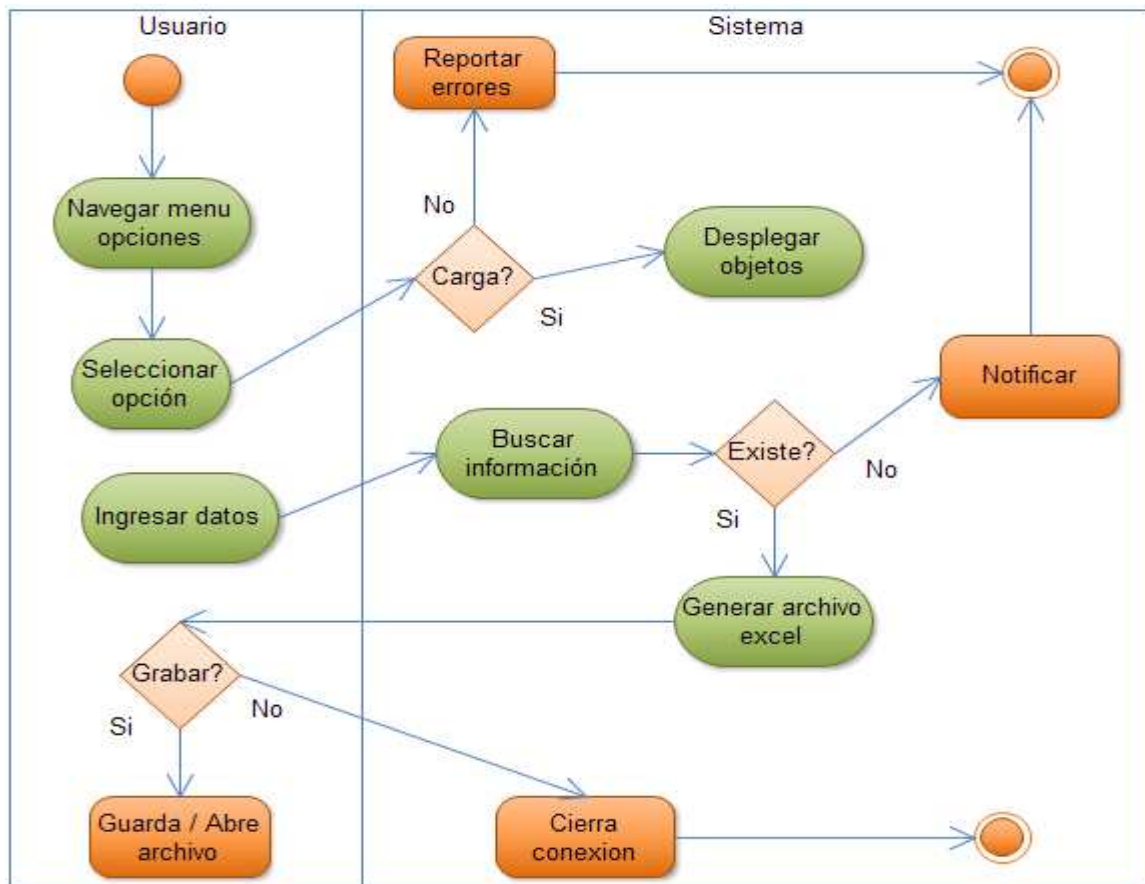


Figura 4.64 Diagrama de actividades, Caso de uso: Consultar Transacciones. Usuario ingresa un parámetro de entrada para buscar datos históricos en el sistema.



Figura 4.65 Diagrama de actividades, Caso de uso: Devolución de insumos. Usuario carga los insumos en el sistema para registrar la entrega realizada por el usuario.

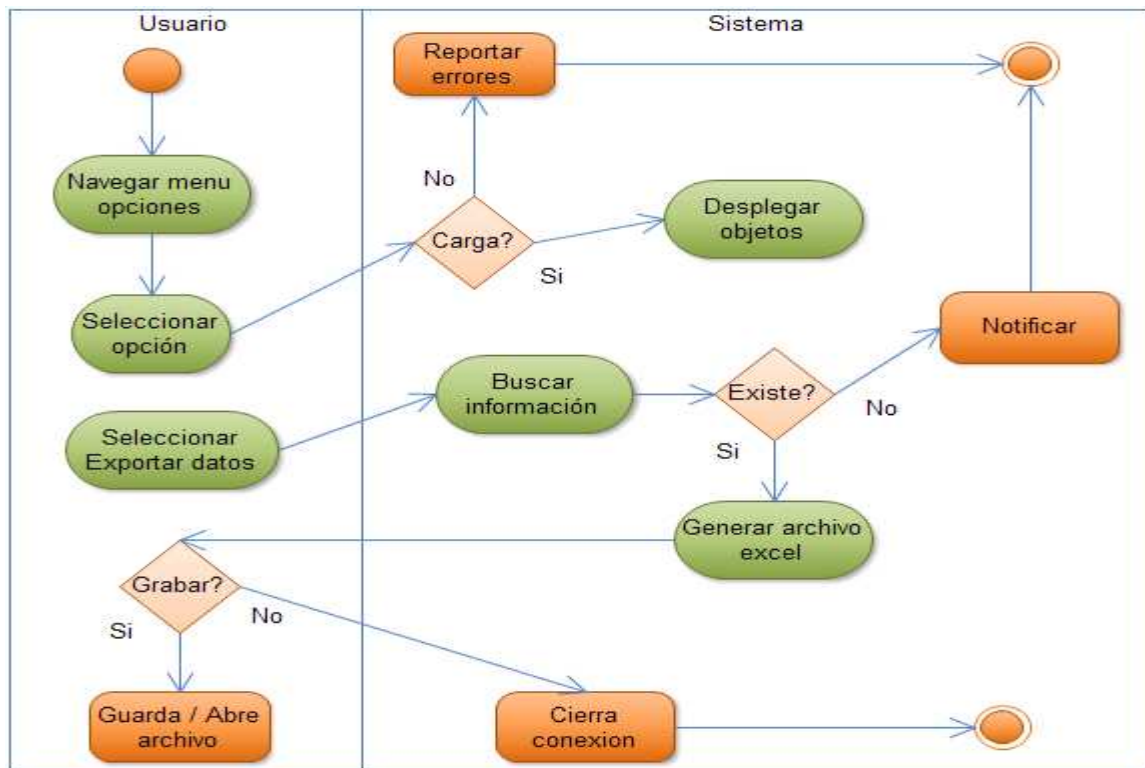


Figura 4.66 Diagrama de actividades, Caso de uso: Situación actual del inventario. Usuario selecciona reporte para obtener el estado actual de los insumos en el sistema.

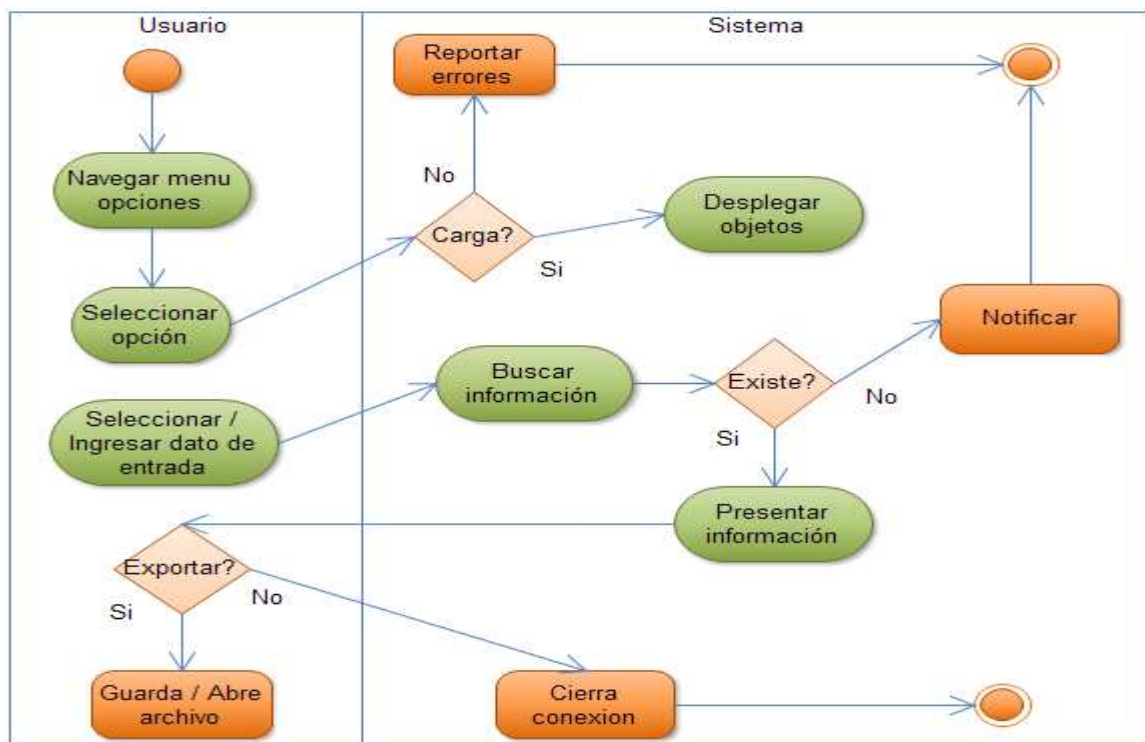


Figura 4.67 Diagrama de actividades, Caso de uso: Insumos Asignados. Usuario realiza la consulta de los insumos asignados a los usuarios.

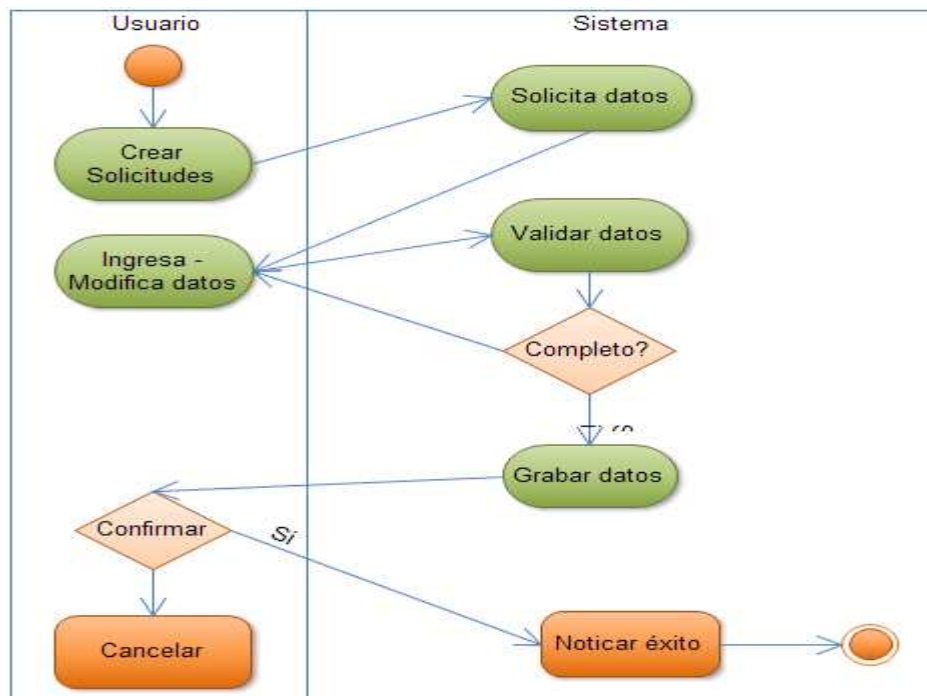


Figura 4.68 Diagrama de actividades, Caso de uso: Crear solicitudes. Usuario crea solicitudes en el sistema, ingresando los parámetros definidos.

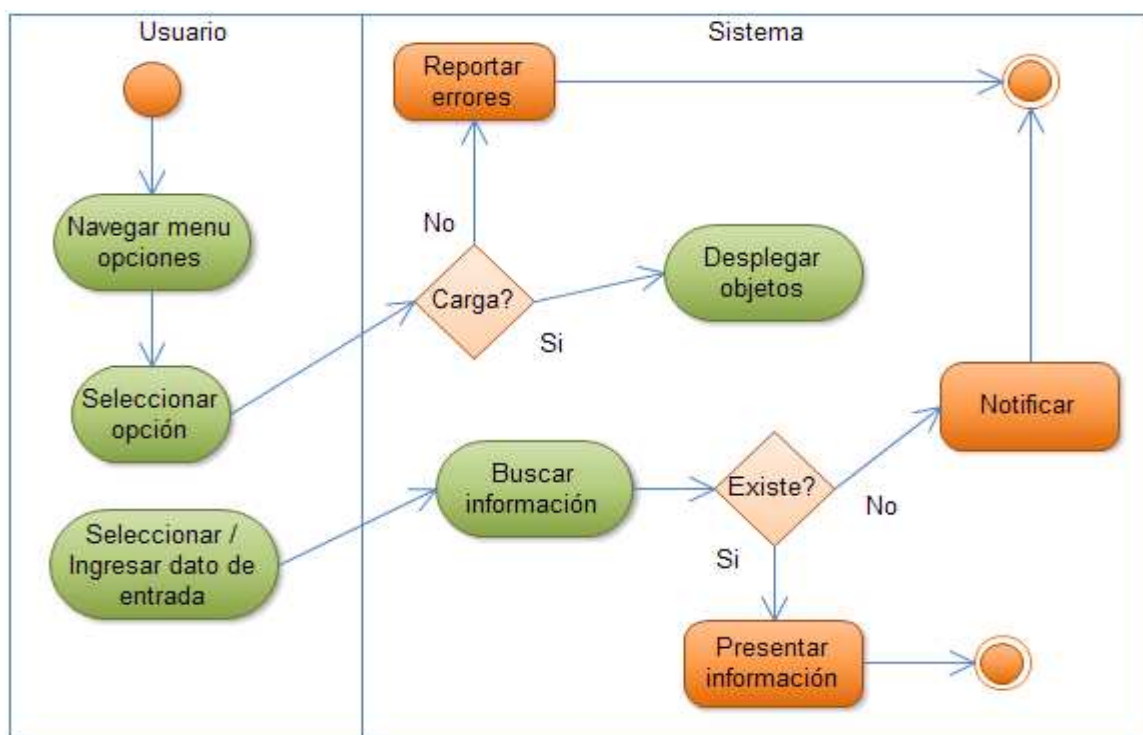


Figura 4.69 Diagrama de actividades, Caso de uso: Consultar solicitudes. Usuario consulta solicitudes existentes en el sistema.

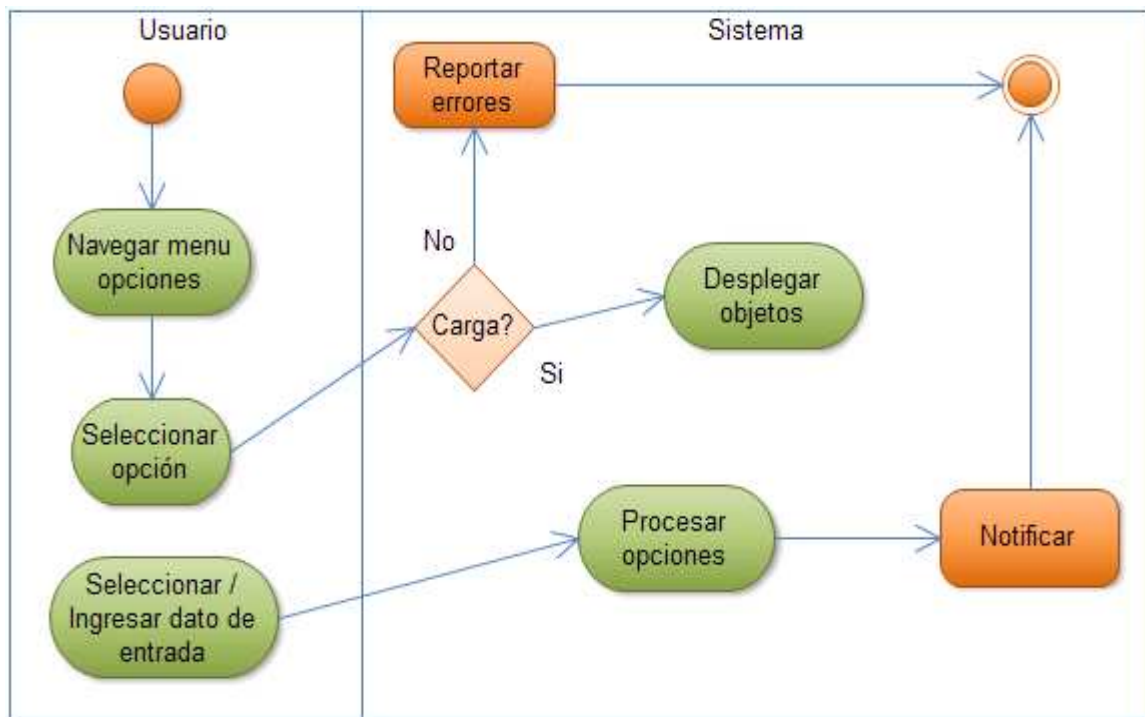


Figura 4.70 Diagrama de actividades, Caso de uso: Procesar solicitudes. Usuario realiza la asignación de recursos en el sistema y procede a cerrar la solicitud.

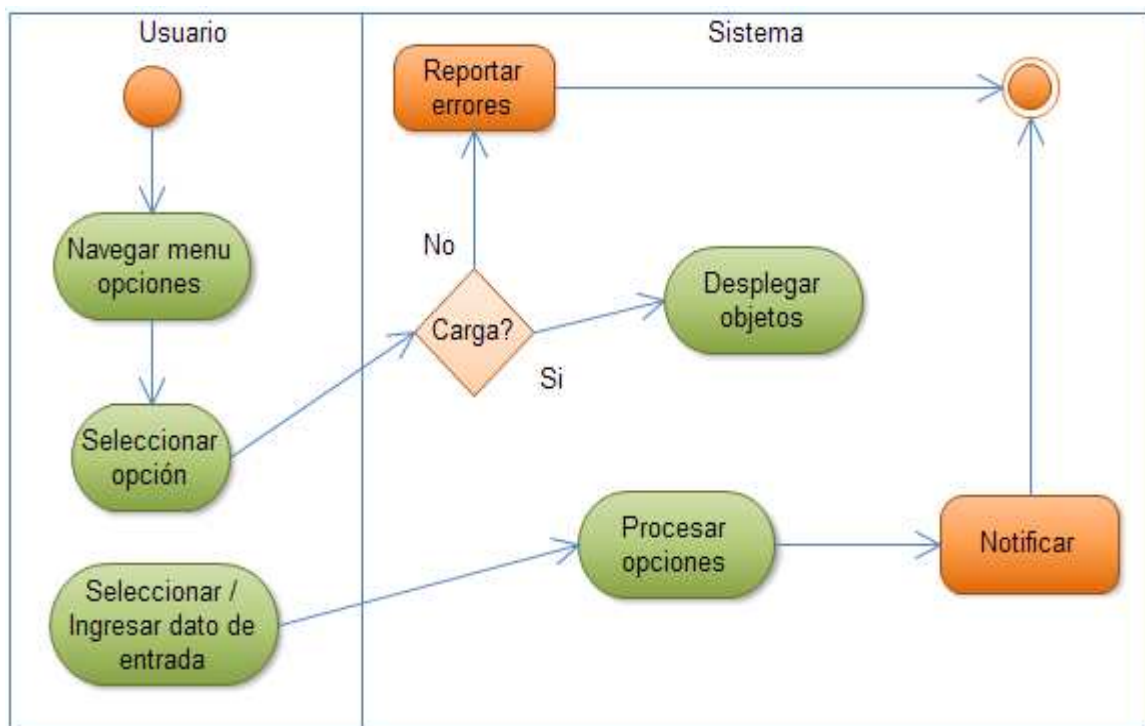


Figura 4.71 Diagrama de actividades, Caso de uso: Extender solicitudes. Usuario modifica la fecha de entrega de un insumo, modificando la fecha.

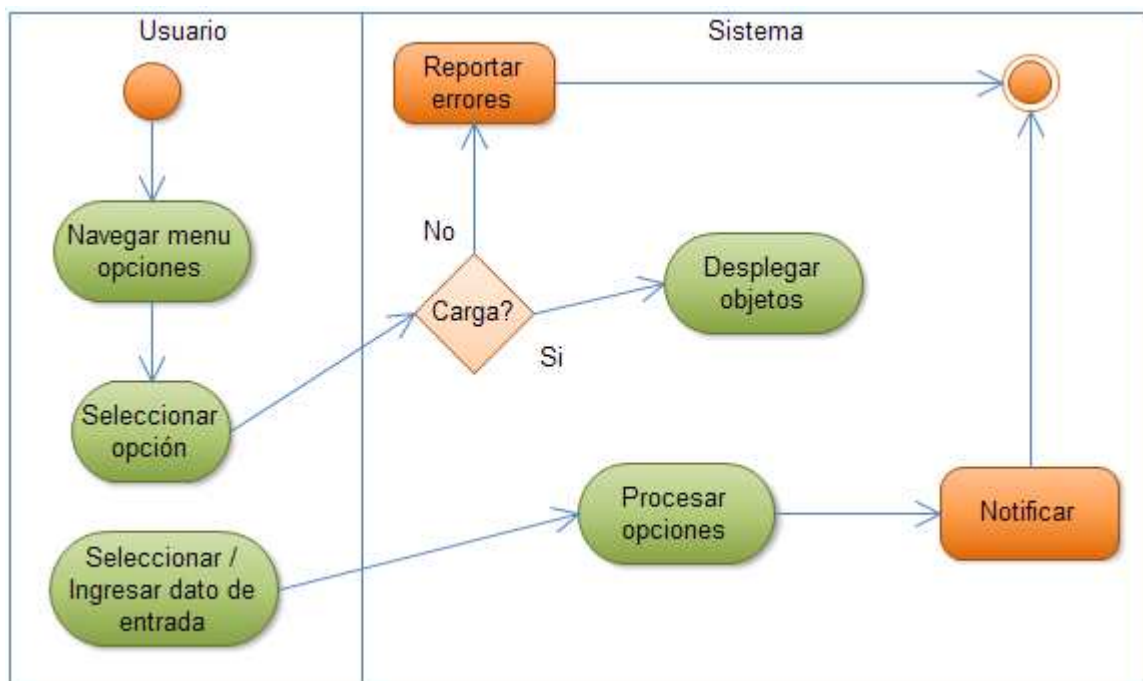


Figura 4.72 Diagrama de actividades, Caso de uso: Re abrir solicitudes. Usuario realiza la apertura de una solicitud ya cerrada en el sistema.

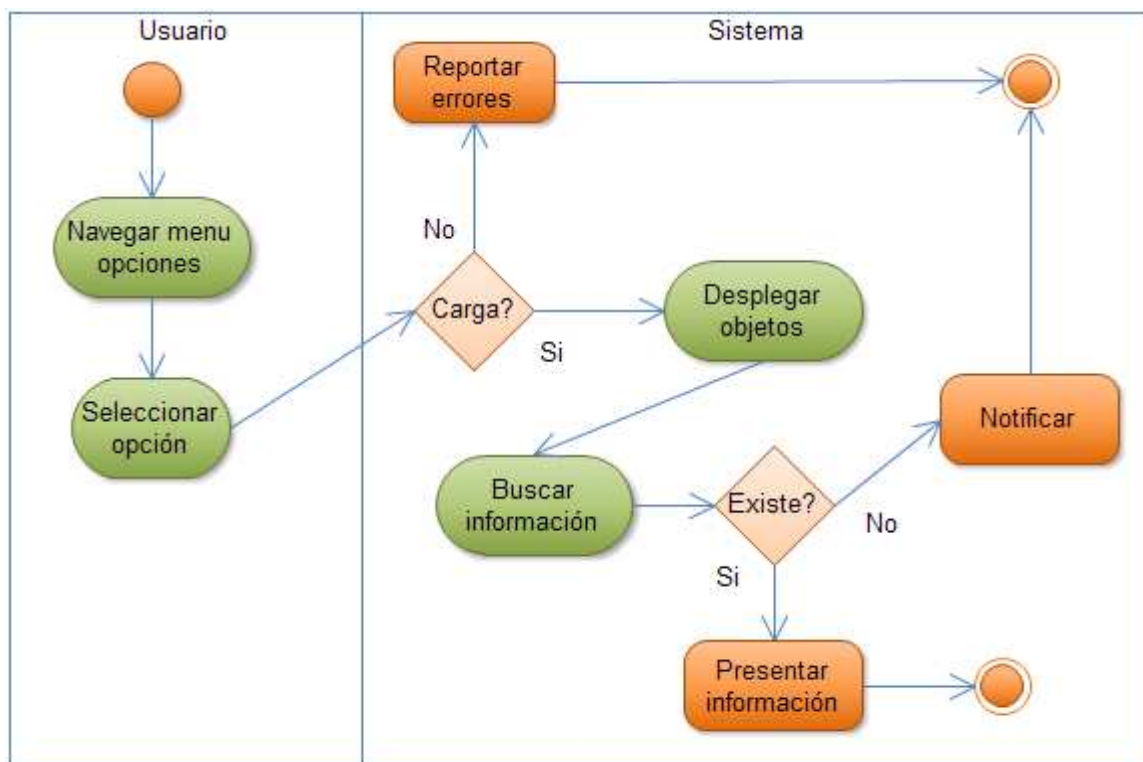


Figura 4.73 Diagrama de actividades, Caso de uso: Solicitudes Pendientes. Usuario visualiza las solicitudes pendientes de procesar en el sistema.

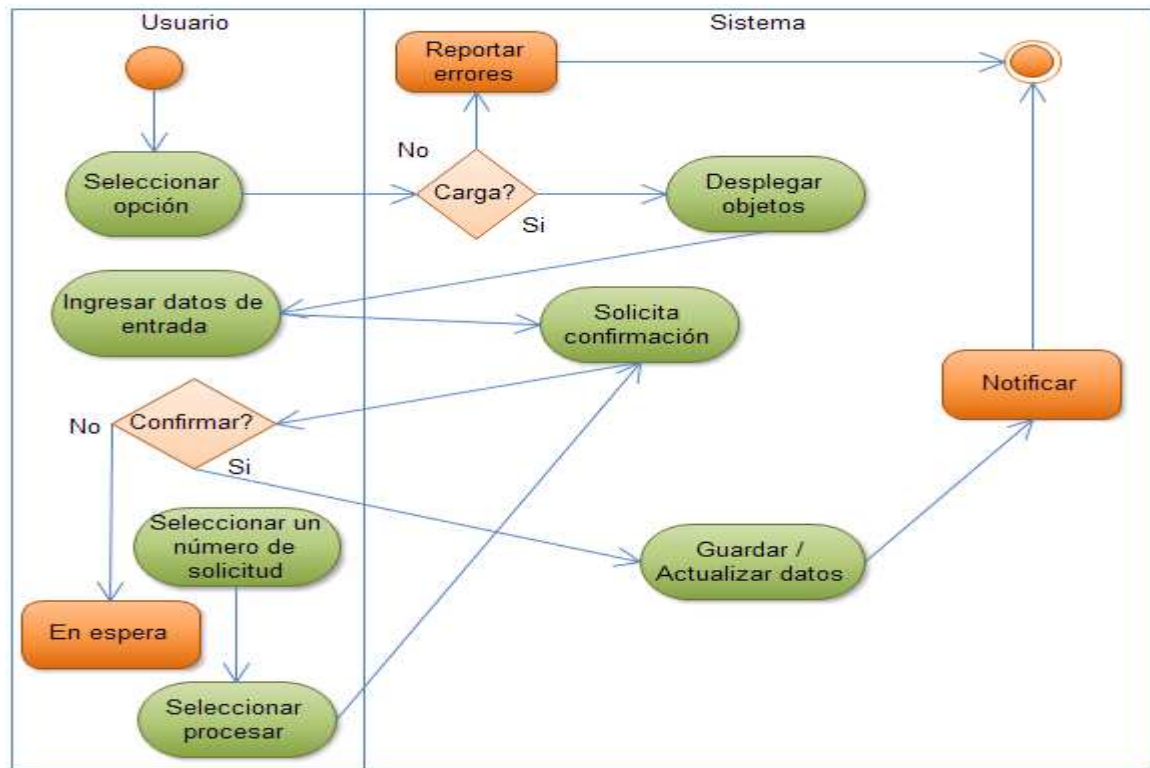


Figura 4.74 Diagrama de actividades, Caso de uso: Registrar Existencias. Usuario realiza la actualización de datos.

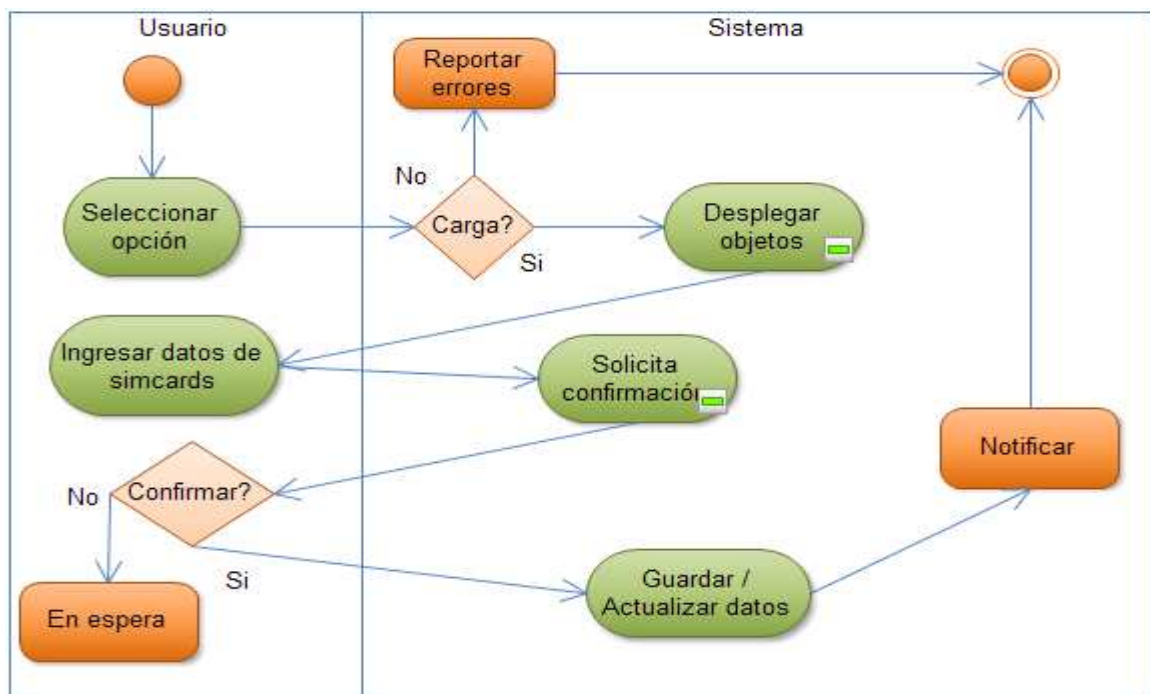


Figura 4.75 Diagrama de actividades, Caso de uso: Ingresar nuevos insumos - Simcards. Usuario crea nuevos registros en el sistema.

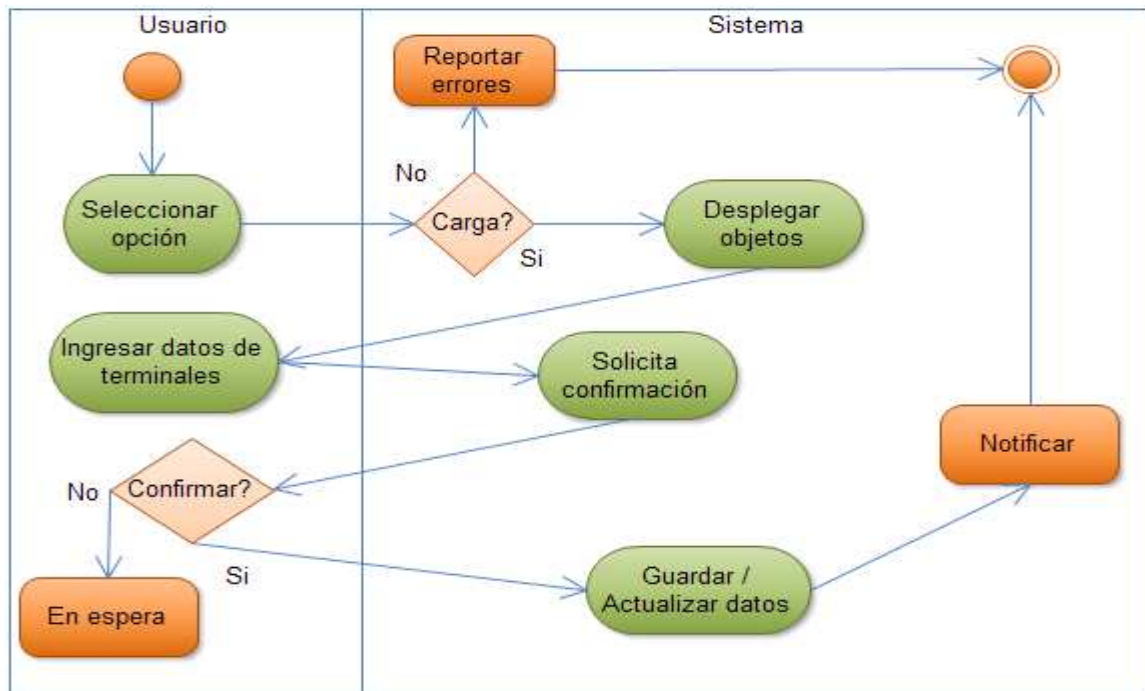


Figura 4.76 Diagrama de actividades, Caso de uso: Ingresar nuevos insumos - Terminales. Usuario crea nuevos registros en el sistema.

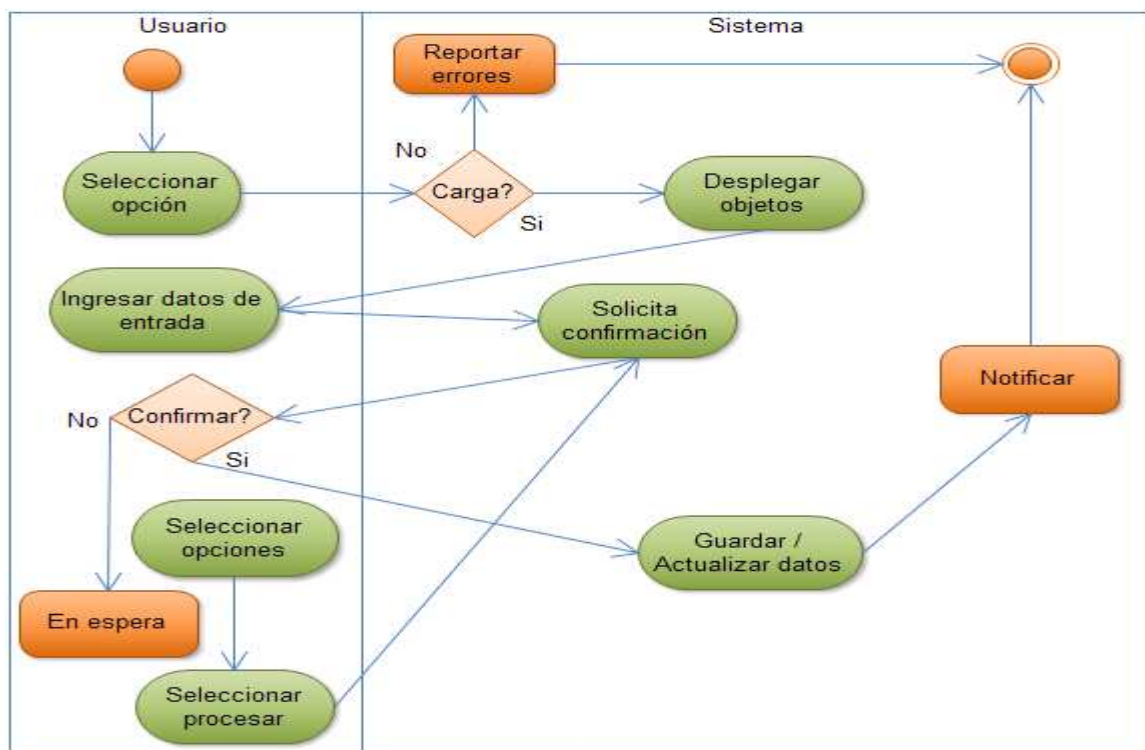


Figura 4.77 Diagrama de actividades, Caso de uso: Actualizar datos de insumos. Usuario modifica los datos en el sistema.

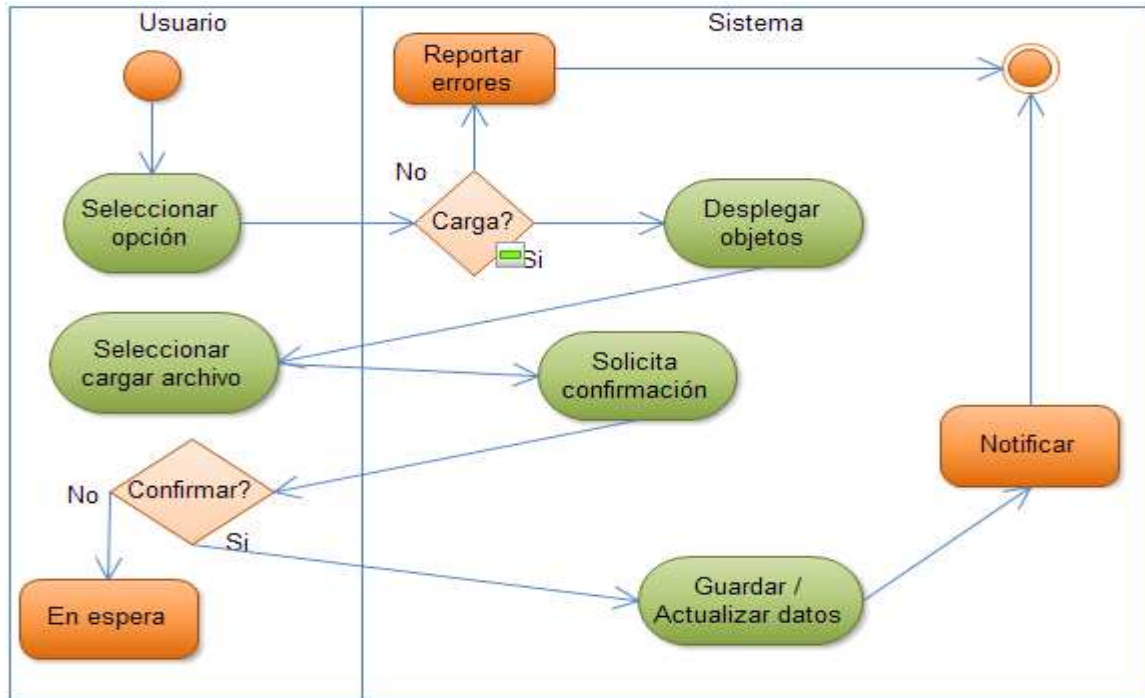


Figura 4.78 Diagrama de actividades, Caso de uso: Cargar insumos. El usuario ingresa un archivo de texto con los códigos de las existencias a ingresar en el sistema.

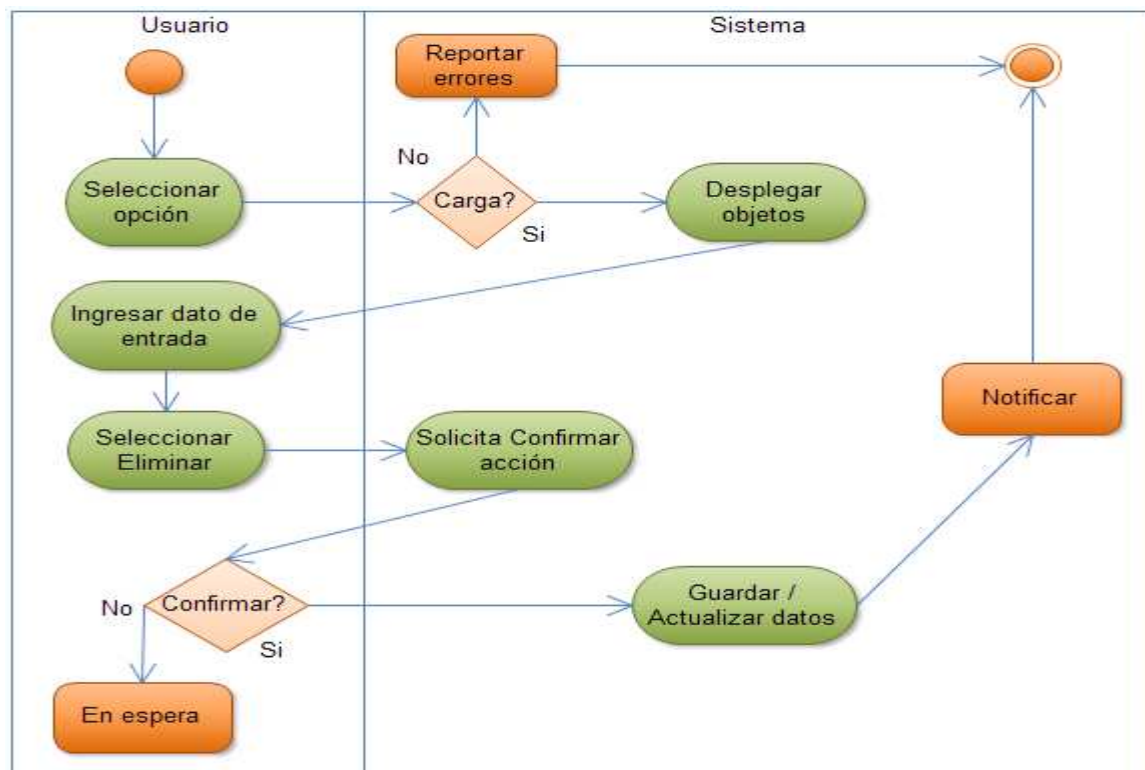


Figura 4.79 Diagrama de actividades, Caso de uso: Eliminar insumos. Usuario elimina registros del sistema a través de esta funcionalidad.

4.1.3.6 DIAGRAMA FISICO ENTIDAD – RELACION

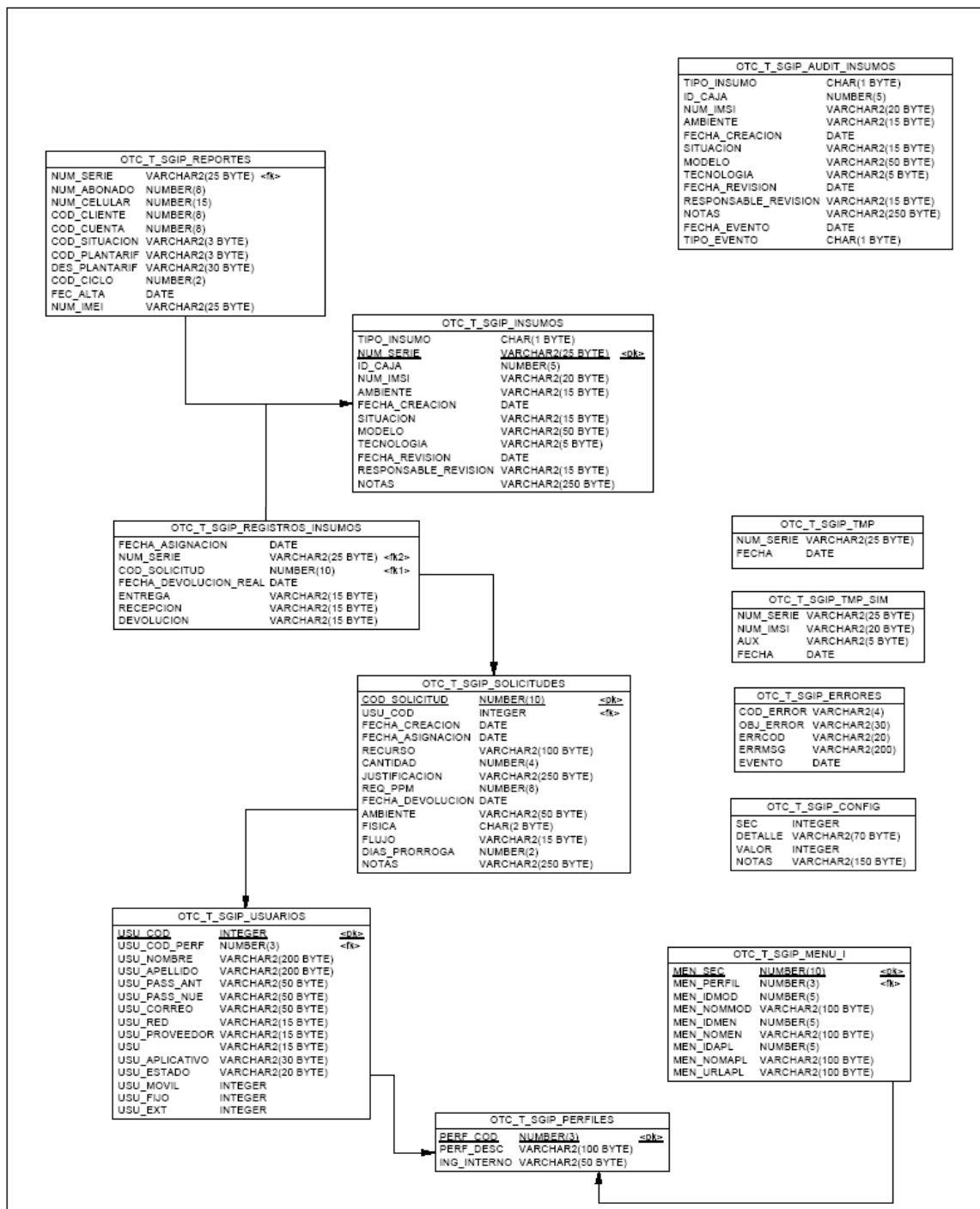


Figura 4.80 Diagrama Entidad Relación.
Diagrama de base de datos en donde se muestra la lógica de negocio para el Sistema de Gestión y Control de Inventario de Pruebas.

4.1.3.7 DICCIONARIO DE DATOS

TABLA: OTC_T_SGIP_AUDIT_INSUMOS [Almacena acciones realizadas sobre la tabla OTC_T_SGIP_INSUMOS]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
TIPO_INSUMO	Contiene un identificador asignado para codificar el tipo de existencia.	1 - Texto	Char	ninguna	Solo Texto #Carac. > 0 && <2
NUM_SERIE	Contiene el código de la simcard o serie de un equipo celular.	25 -Texto	Varchar2	Índice	Solo Texto #Carac. > 0 && <26
ID_CAJA	Contiene el código de caja en el cual se agruparan las series	5 - Números	Number	ninguna	Solo números #Carac. >0 && <6
NUM_IMSI	Contiene el código interno de una simcard, llamado también central o HLR.	20 - Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <21
AMBIENTE	Contiene el tipo de base de datos, para el caso de la empresa hay tres.	15 - Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <16
FECHA_CREACION	Contiene la fecha de cuando fue creado el registro.	Fecha	Date	ninguna	Solo fechas #Carac. Sysdate
SITUACION	Contiene el último estado del registro, el mismo que se obtiene del ambiente.	15-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <16
MODELO	Contiene el modelo de un teléfono celular.	50-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <51
TECNOLOGIA	Contiene el tipo de tecnología del Teléfono celular, siendo GSM o CDMA.	5-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <6
FECHA_REVISION	Contiene la última fecha de modificación del registro. Esto significa que la simcard o serie estuvo en manos de un responsable.	Fecha	Date	ninguna	Solo fechas #Carac. Sysdate

RESPONSABLE_REVISION	Contiene el usuario responsable de la revisión de una simcard o equipo celular.	15-Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <16
NOTAS	Contiene una descripción detallada que permite aclarar el por que de la creación del registro.	250-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <251
FECHA_EVENTO	Contiene la última fecha en la cual se realiza una modificación al registro.	Fecha	Date	ninguna	Solo fechas #Carac. Sysdate
TIPO_EVENTO	Contiene un identificador asignado para distinguir que acción fue realizada sobre el registro, siendo SELECT – INSERT – UPDATE – DELETE.	1-Texto	Char	ninguna	Solo Texto #Carac. >0 && <2

Tabla 4.26 Diccionario de datos: Tabla OTC_T_SGIP_AUDIT_INSUMOS.

TABLA: OTC_T_SGIP_REPORTES [Almacena los datos de las existencias activas]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
NUM_SERIE	Contiene el código de la simcard o serie de un equipo celular.	25 -Texto	Varchar2	FK - Índice	Solo Texto #Carac. > 0 && <26
NUM_ABONADO	Contiene el código interno del sistema comercial de la operadora. Se obtiene de otra base de datos mediante un DBL.	8 - Números	Number	Indice	Solo números #Carac. >0 && <8
NUM_CELULAR	Contiene el número celular de un cliente en formato 8 dígitos.	15 - Números	Number	ninguna	Solo números #Carac. >0 && <16

COD_CLIENTE	Contiene el código del cliente, el mismo que tiene relación con el num_celular y num_abonado.	8 - Números	Number	ninguna	Solo números #Carac. >0 && <9
COD_CUENTA	Contiene la cuenta del cliente, el mismo que tiene relación con el cod_cliente. Se obtiene de otra base de datos mediante un DBL.	8 - Números	Number	ninguna	Solo números #Carac. >0 && <9
COD_SITUACION	Contiene el último estado del registro, el mismo que se obtiene del ambiente. Se obtiene de otra base de datos mediante un DBL.	3-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <4
COD_PLANTARIF	Contiene el código del plan tarifario del cliente. Se obtiene de otra base de datos mediante un DBL.	3-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <4
DES_PLANTARIF	Contiene el nombre o descripción del plan tarifario, tiene relación con el cod_plantarif. Se obtiene de otra base de datos mediante un DBL.	30-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <31
COD_CICLO	Contiene el código relacionado con el período de facturación del cliente, tiene relación con el num_abonado. Se obtiene de otra base de datos mediante un DBL.	2- Números	Number	ninguna	Solo números #Carac. >0 && <3
FEC_ALTA	Contiene la fecha en la que fue creado el registro, o la fecha de alta del cliente en el sistema comercial de la operadora.	Fecha	Date	ninguna	Solo fechas #Carac. Sysdate
NUM_IMEI	Contiene la serie del teléfono celular, sobre el cual fue activado un cliente.	15-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <16

Tabla 4.27 Diccionario de datos: Tabla OTC_T_SGIP_REPORTES

TABLA: OTC_T_SGIP_INSUMOS [Almacena las existencias del área de pruebas]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
TIPO_INSUMO	Contiene un identificador asignado para codificar el tipo de existencia.	1 - Texto	Char	Índice	Solo Texto #Carac. > 0 && <2
NUM_SERIE	Contiene el código de la simcard o serie de un equipo celular.	25 - Texto	Varchar2	PK-Índice No nulo	Solo Texto #Carac. > 0 && <26
ID_CAJA	Contiene el código de caja en el cual se agruparan las series	5 - Números	Number	ninguna	Solo números #Carac. >0 && <6
NUM_IMSI	Contiene el código interno de una simcard, llamado también central o HLR.	20 - Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <21
AMBIENTE	Contiene el tipo de base de datos, para el caso de la empresa hay tres.	15 - Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <16
FECHA_CREACION	Contiene la fecha de cuando fue creado el registro.	Fecha	Date	Índice	Solo fechas #Carac. Sysdate
SITUACION	Contiene el último estado del registro, el mismo que se obtiene del ambiente.	15-Texto	Varchar2	Índice	Solo Texto #Carac. >0 && <16
MODELO	Contiene el modelo de un teléfono celular.	50-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <51
TECNOLOGIA	Contiene el tipo de tecnología del Teléfono celular, siendo GSM o CDMA.	5-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <6
FECHA_REVISION	Contiene la última fecha de modificación del registro. Esto significa que la simcard o serie estuvo	Fecha	Date	ninguna	Solo fechas #Carac. Sysdate

	en manos de un responsable.				
RESPONSABLE_REVISION	Contiene el usuario responsable de la revisión de una simcard o equipo celular.	15-Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <16
NOTAS	Contiene una descripción detallada que permite aclarar el por que de la creación del registro.	250-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <251

Tabla 4.28 Diccionario de datos: Tabla OTC_T_SGIP_INSUMOS

TABLA: OTC_T_SGIP_SOLICITUDES [Almacena las peticiones de existencias en el área de pruebas]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
COD_SOLICITUD	Contiene el número de solicitud asignado por el sistema, para atender una petición de recursos.	10 - Números	Number	PK – No Nulo	Solo números #Carac. >0 && <11
USU_COD	Contiene el código del usuario que realiza una solicitud de recursos en el sistema.	3 - Números	Number	FK	Solo números #Carac. >0 && <4
FECHA_CREACION	Contiene la fecha de creación del registro, la cual corresponde a la fecha de solicitud de recursos.	Fecha	Date	Índice	Solo fechas #Carac. Sysdate
FECHA_ASIGNACION	Contiene la fecha de procesamiento del registro, la cual corresponde a la fecha de entrega de recursos.	Fecha	Date	Índice	Solo fechas #Carac. Sysdate
RECURSO	Contiene el tipo de insumo / recurso que requiere un solicitante / usuario.	100 - Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <101
CANTIDAD	Contiene la cantidad de ítems que requiere el solicitante.	4 - Números	Number	ninguna	Solo números #Carac. >0 && <5

JUSTIFICACION	Contiene un detalle del por que y para que requiere lo detallado en la solicitud.	250-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <251
REQ_PPM	Contiene un código que se obtiene de un aplicativo externo llamado Project and Portafolio Managenemt.	8- Números	Number	ninguna	Solo números #Carac. >0 && <9
FECHA_DEVOLUCION	Contiene una fecha estimada o supuesta en la que el solicitante devolverá los recursos a la bodega.	Fecha	Date	Índice	Solo fechas #Carac. Sysdate
AMBIENTE	Contiene el tipo de base de datos, para el caso de la empresa hay tres.	50 - Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <16
FISICA	Contiene un identificador para definir si un recurso es digital o físico.	2 - Texto	Char	ninguna	Solo Texto #Carac. >0 && <3
FLUJO	Contiene el estado de la solicitud, es utilizado para las diferentes acciones para el proceso de asignación de recursos.	15 - Texto	Varchar2	ninguna	Solo Texto Todo mayúsculas #Carac. >0 && <16
DIAS_PRORROGA	Contiene un valor numérico en días, esto permite sumar una cantidad en días a la fecha devolución.	2- Números	Number	ninguna	Solo números #Carac. >0 && <3
NOTAS	Contiene una descripción detallada que permite aclarar el por que de la creación del registro.	250-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <251

Tabla 4.29 Diccionario de datos: Tabla OTC_T_SGIP_SOLICITUDES

TABLA: OTC_T_SGIP_USUARIOS [Almacena los usuarios permitidos para acceder al sistema de inventarios]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
USU_COD	Contiene el número del registro asignado por el sistema, al crear un usuario.	3 - Números	Number	PK – No Nulo	Solo números #Carac. >0 && <4
USU_COD_PERF	Contiene el código del perfil, relacionado con la tabla perfiles.	3 - Números	Number	FK	Solo números #Carac. >0 && <4
USU_NOMBRE	Contiene el nombre del usuario	200 - Texto	Varchar2	No nulo	Solo Texto #Carac. >0 && <201
USU_APELLIDO	Contiene el apellido del usuario	200 - Texto	Varchar2	No nulo	Solo Texto #Carac. >0 && <201
USU_PASS_ANT	Contiene la clave inicial asignada por el sistema / administrador. En este caso es 0	50 - Texto	Varchar2	No nulo	Solo números #Carac. >0 && <51
USU_PASS_NUE	Contiene la clave que el usuario registra al realizar un cambio de clave.	50 - Texto	Varchar2	ninguna	Alfanuméricos #Carac. >0 && <51
USU_CORREO	Contiene la dirección de correo electrónica del usuario.	50-Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <51
USU_RED	Contiene el código de red asignado por el administrador de TI, a través de un sistema externo.	15-Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <16
USU_PROVEEDOR	Contiene el código de red asignado por el administrador del proveedor.	15-Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <16
USU	Contiene el código de red asignado por el administrador interno del sistema de Inventarios.	15-Texto	Varchar2	ninguna	Todo mayúsculas #Carac. >0 && <16
USU_APLICATIVO	Contiene un identificador para definir si un usuario	30 - Texto	Varchar2	ninguna	Solo Texto

	tiene acceso a uno o más aplicativos si aplica.				#Carac. >0 && <31
USU_ESTADO	Contiene el estado – situación del usuario. Puede ser Activo – Suspendido.	20 - Texto	Varchar2	ninguna	Solo Texto Todo mayúsculas #Carac. >0 && <21
USU_MOVIL	Contiene el número celular móvil del usuario.	10- Números	Number	ninguna	Solo números #Carac. >0 && <11
USU_FIJO	Contiene el número de contacto fijo del usuario.	10- Números	Number	ninguna	Solo números #Carac. >0 && <11
USU_EXT	Contiene el número de extensión del contacto fijo del usuario.	5- Números	Number	ninguna	Solo números #Carac. >0 && <6

Tabla 4.30 Diccionario de datos: Tabla OTC_T_SGIP_USUARIOS

TABLA: OTC_T_SGIP_PERFILES [Almacena los perfiles disponibles para los usuarios del sistema]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
USU_COD_PERF	Contiene el código del perfil, asignado por el sistema. Relacionado con la tabla usuarios.	3 - Números	Number	PK – No nulo	Solo números #Carac. >0 && <4
PERF_DESC	Contiene el nombre del perfil de usuario.	100 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <101
ING_INTERNO	Contiene el nombre de un responsable del perfil de usuario. Esto lo define el área de la operadora.	50 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <51

Tabla 4.31 Diccionario de datos: Tabla OTC_T_SGIP_PERFILES

TABLA: OTC_T_SGIP_MENU_I [Almacena el árbol de opciones que está configurado para el sistema y los usuarios]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
MEN_SEC	Contiene el código del menú, asignado por el sistema. Relacionado con la tabla perfiles.	10 - Números	Number	PK – No nulo	Solo números #Carac. >0 && <11
MEN_PERFIL	Contiene el código del perfil para asignar a un usuario.	3 - Números	Number	FK – Índice	Solo números #Carac. >0 && <4
MEN_IDMOD	Contiene el código del menú padre a ser creado en el objeto TREE VIEW para desplegar los módulos del sistema.	5 - Números	Number	ninguno	Solo números #Carac. >0 && <6
MEN_NOMMOD	Contiene el nombre del menú padre a ser creado en el objeto TREE VIEW para desplegar los módulos del sistema.	100 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <101
MEN_IDMEN	Contiene el código del sub menú padre a ser creado en el objeto TREE VIEW para desplegar los eventos del sistema.	5 - Números	Number	ninguno	Solo números #Carac. >0 && <6
MEN_NOMEN	Contiene el nombre del sub menú padre a ser creado en el objeto TREE VIEW para desplegar los eventos del sistema.	100 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <101
MEN_IDAPL	Contiene el código del menú hijo a ser creado en el objeto TREE VIEW para desplegar los eventos del sistema y el orden.	5 - Números	Number	ninguno	Solo números #Carac. >0 && <6
MEN_NOMAPL	Contiene el nombre del menú hijo a ser creado en el objeto TREE VIEW para desplegar los eventos	100 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <101

	del sistema. Este nombre es visualizado por el usuario.				
MEN_URLAPL	Contiene el nombre del formulario del menú hijo.	100 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <101

Tabla 4.32 Diccionario de datos: Tabla OTC_T_SGIP_MENU_I

TABLA: OTC_T_SGIP_CONFIG [Almacena datos de configuraciones adicionales para el sistema]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
SEC	Contiene el código de una configuración, asignado por el administrador del sistema.	5 - Números	Number	ninguno	Solo números #Carac. >0 && <6
DETALLE	Contiene el nombre del parámetro de configuración.	70 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <71
VALOR	Contiene el código definido a una configuración, asignado por el administrador del sistema.	5 - Números	Number	Índice	Solo números #Carac. >0 && <6
NOTAS	Contiene una descripción detallada que permite aclarar el por que de la creación del registro.	150-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <151

Tabla 4.33 Diccionario de datos: Tabla OTC_T_SGIP_CONFIG

TABLA: OTC_T_SGIP_ERRORES [Almacena errores producto de invocaciones fallidas en los stored procedures]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
COD_ERROR	Contiene el código de error, definido en el paquete de base de datos. El mismo que está asignado a cada procedimiento almacenado.	4- Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <5
OBJ_ERROR	Contiene el nombre del procedimiento almacenado.	30 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <31
ERRCOD	Contiene el código del error ORACLE.	20 - Texto	Varchar2	ninguno	Solo Texto #Carac. >0 && <21
ERRMSG	Contiene el detalle del error ORACLE.	200-Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <201
EVENTO	Contiene la fecha del registro creado.	Fecha	Date	ninguna	Solo fechas #Carac. Sysdate

Tabla 4.34 Diccionario de datos: Tabla OTC_T_SGIP_ERRORES

TABLA: OTC_T_SGIP_TMP [Almacena datos temporales, como series o simcards para el proceso de inventarios]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
NUM_SERIE	Contiene el código de la simcard o serie de un equipo celular.	25 -Texto	Varchar2	Índice	Solo Texto #Carac. > 0 && <26
FECHA	Contiene la fecha de creación del registro.	Fecha	Date	ninguna	Solo fechas #Carac. Sysdate

Tabla 4.35 Diccionario de datos: Tabla OTC_T_SGIP_TMP

TABLA: OTC_T_SGIP_TMP_SIM [Almacena datos temporales para la creación de series o simcards]

Campo	Descripción	Longitud Tipo	Formato	Restricción	Validación
NUM_SERIE	Contiene el código de la simcard o serie de un equipo celular.	25 - Texto	Varchar2	Índice	Solo Texto #Carac. > 0 && <26
NUM_IMSI	Contiene el código interno de una simcard, llamado también central o HLR.	20 - Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <21
AUX	Contiene una descripción del registro.	25 - Texto	Varchar2	ninguna	Solo Texto #Carac. >0 && <26
FECHA	Contiene la fecha de creación del registro.	Fecha	Date	Índice	Solo fechas #Carac. Sysdate

*Tabla 4.36 Diccionario de datos: Tabla OTC_T_SGIP_TMP_SIM**Entregable MSF "Plan Maestro Construcción Sistema Inventarios"*

4.2. CODIGO FUENTE

4.2.1. SCRIPT DE BASE DE DATOS

A continuación se copia fragmentos de código representativos de los objetos de esta sección.

```

/* Formatted on 2012/05/18 11:55 (Formatter Plus v4.8.8) */
CREATE OR REPLACE PACKAGE internet.otc_pkg_sgip AS
  TYPE tabla_number IS TABLE OF NUMBER (38)
    INDEX BY BINARY_INTEGER;

  TYPE tabla_string IS TABLE OF VARCHAR2 (250)
    INDEX BY BINARY_INTEGER;

  PROCEDURE otc_t_sp_insumos_activos (existe OUT NUMBER);

  PROCEDURE otc_sp_modifica_estado_usuario (
    aux_usu_cod IN otc_t_sgip_usuarios.usu_cod%TYPE,
    aux_estado IN otc_t_sgip_usuarios.usu_estado%TYPE,
    retorno OUT NUMBER
  );

  PROCEDURE otc_t_sp_consultar_usuario_cod (
    aux_usu_cod IN otc_t_sgip_usuarios.usu_cod%TYPE,
    codigo OUT tabla_number,
    nombre OUT tabla_string,
    apellido OUT tabla_string,
    correo OUT tabla_string,
    proveedor OUT tabla_string,
    estado OUT tabla_string,
    movil OUT tabla_number,
    fijo OUT tabla_number,
    ext OUT tabla_number
  );

  PROCEDURE otc_t_sp_reporte_activos (
    caja OUT tabla_number,
    serie OUT tabla_string,
    abonado OUT tabla_number,
    celular OUT tabla_number,
    cliente OUT tabla_number,
    cuenta OUT tabla_number,
    codigo OUT tabla_string,
    PLAN OUT tabla_string,
    ciclo OUT tabla_number,
    alta OUT tabla_string,
    imei OUT tabla_string
  );

  PROCEDURE otc_t_sp_up_asignado_notas (
    existe OUT NUMBER,
    num_serie OUT tabla_string
  );
END otc_pkg_sgip;
/

```

```

/* Formatted on 2012/05/18 13:38 (Formatter Plus v4.8.8) */
CREATE OR REPLACE PACKAGE BODY internet.otc_pkg_sgip
AS

```



```

errcod NUMBER;
errmsg VARCHAR2 (200);

PROCEDURE otc_sp_cod_perfil (
  aux_usu_red      IN      otc_t_sgip_usuarios.usu_red%TYPE,
  aux_usu_pass_nue IN      otc_t_sgip_usuarios.usu_pass_nue%TYPE,
  aux_cod_perf     OUT     otc_t_sgip_usuarios.usu_cod_perf%TYPE
)
IS
BEGIN
  SELECT COUNT (usu_cod_perf)
  INTO aux_cod_perf
  FROM otc_t_sgip_usuarios
  WHERE usu_red = UPPER (aux_usu_red)
  AND usu_pass_nue = aux_usu_pass_nue
  AND usu_aplicativo = 'INVENTARIOS'
  AND usu_estado = 'ACTIVO'
  AND ROWNUM = 1;

  IF aux_cod_perf = 0
  THEN
    SELECT usu_cod_perf
    INTO aux_cod_perf
    FROM otc_t_sgip_usuarios
    WHERE usu_red = UPPER (aux_usu_red)
    AND usu_pass_nue = aux_usu_pass_nue
    AND usu_aplicativo != 'INVENTARIOS'
    AND usu_estado = 'ACTIVO'
    AND ROWNUM = 1;
  ELSE
    SELECT usu_cod_perf
    INTO aux_cod_perf
    FROM otc_t_sgip_usuarios
    WHERE usu_red = UPPER (aux_usu_red)
    AND usu_pass_nue = aux_usu_pass_nue
    AND usu_aplicativo = 'INVENTARIOS'
    AND usu_estado = 'ACTIVO'
    AND ROWNUM = 1;
  END IF;
EXCEPTION
  WHEN NO_DATA_FOUND
  THEN
    aux_cod_perf := 0;
  WHEN OTHERS
  THEN
    errmsg := SUBSTR (SQLERRM, 1, 150);
    errcod := SUBSTR (TO_CHAR (SQLCODE), 1, 15);

    INSERT INTO otc_t_sgip_errores
    VALUES ('1001', 'OTC_SP_COD_PERFIL', errmsg, errcod, SYSDATE);
END;

PROCEDURE otc_t_sp_crear_insumos (
  aux      OUT      tabla_string
)
IS
  existe      NUMBER;

  CURSOR c_num_serie
  IS
    SELECT DISTINCT (num_serie), num_imsi, aux
    FROM otc_t_sgip_tmp_sim

```

```

WHERE num_serie NOT IN (SELECT num_serie
                        FROM otc_t_sgip_insumos)
AND fecha > SYSDATE - 1;

t_num_serie_count NUMBER DEFAULT 1;
BEGIN
SELECT COUNT (num_serie)
  INTO existe
  FROM otc_t_sgip_tmp_sim
 WHERE num_serie NOT IN (SELECT num_serie
                        FROM otc_t_sgip_insumos);

IF existe > 0
THEN
  IF aux_insumo = 'S'
  THEN
    FOR all_t_num_serie IN c_num_serie
    LOOP
      num_serie (t_num_serie_count) := all_t_num_serie.num_serie;
      num_imsi (t_num_serie_count) := all_t_num_serie.num_imsi;

      INSERT INTO otc_t_sgip_insumos
        VALUES (aux_insumo, num_serie (t_num_serie_count),
                aux_caja, num_imsi (t_num_serie_count),
                aux_ambiente, SYSDATE, 'NUEVO', aux_modelo,
                aux_tecnologia, "", aux_responsable, aux_notas);

      t_num_serie_count := t_num_serie_count + 1;
      EXIT WHEN c_num_serie%NOTFOUND;
    END LOOP;
  ELSE
    FOR all_t_num_serie IN c_num_serie
    LOOP
      num_serie (t_num_serie_count) := all_t_num_serie.num_serie;
      num_imsi (t_num_serie_count) := all_t_num_serie.num_imsi;
      aux (t_num_serie_count) := all_t_num_serie.aux;

      INSERT INTO otc_t_sgip_insumos
        VALUES (aux_insumo, num_serie (t_num_serie_count),
                aux (t_num_serie_count), "", SYSDATE,
                'NUEVO', num_imsi (t_num_serie_count),
                aux_tecnologia, "", aux_responsable, aux_notas);

      t_num_serie_count := t_num_serie_count + 1;
      EXIT WHEN c_num_serie%NOTFOUND;
    END LOOP;
  END IF;
ELSE
  NULL;
END IF;
EXCEPTION
WHEN OTHERS
THEN
  errmsg := SUBSTR (SQLERRM, 1, 150);
  errcod := SUBSTR (TO_CHAR (SQLCODE), 1, 15);

  INSERT INTO otc_t_sgip_errores
    VALUES ('1024', 'OTC_T_SP_CREAR_INSUMOS_M', errcod, errmsg,
            SYSDATE);
END;

PROCEDURE otc_sp_modifica_perfil_usuario (

```

```

aux_usu_cod IN otc_t_sqip_usuarios.usu_cod%TYPE,
aux_perfil IN otc_t_sqip_usuarios.usu_cod_perf%TYPE,
retorno OUT NUMBER
)
IS
valida NUMBER;
BEGIN
SELECT COUNT (*)
INTO valida
FROM otc_t_sqip_usuarios
WHERE usu_cod = aux_usu_cod AND usu_estado = 'ACTIVO';

IF valida > 0
THEN
UPDATE otc_t_sqip_usuarios
SET usu_cod_perf = aux_perfil
WHERE usu_cod = aux_usu_cod;

retorno := 1;
ELSE
retorno := 0;
END IF;
EXCEPTION
WHEN NO_DATA_FOUND
THEN
retorno := 0;
WHEN OTHERS
THEN
errmsg := SUBSTR (SQLERRM, 1, 150);
errcod := SUBSTR (TO_CHAR (SQLCODE), 1, 15);

INSERT INTO otc_t_sqip_errores
VALUES ('1026', 'OTC_SP_MODIFICA_PERFIL_USUARIO', errcod,
errmsg, SYSDATE);
END;

PROCEDURE otc_t_sp_reporte_inventario (
tipo_insumo OUT tabla_string,
num_serie OUT tabla_string,
id_caja OUT tabla_number,
num_imsi OUT tabla_string,
ambiente OUT tabla_string,
fecha_creacion OUT tabla_string,
situacion OUT tabla_string,
modelo OUT tabla_string,
tecnologia OUT tabla_string,
fecha_revision OUT tabla_string,
responsable_revision OUT tabla_string,
notas OUT tabla_string
)
IS
CURSOR c_reporte_inventario
IS
SELECT DECODE (tipo_insumo,
'S', 'SIMCARD',
'T', 'TERMINAL'
) AS tipo_insumo,
num_serie, id_caja, num_imsi, ambiente,
TO_CHAR (fecha_creacion,
'MM/DD/YYYY HH24:MI:SS'
) AS fecha_creacion,
situacion, modelo, tecnologia,

```

```

        TO_CHAR (fecha_revision,
                'MM/DD/YYYY HH24:MI:SS'
        ) AS fecha_revision,
        responsable_revision, notas
FROM otc_t_sgip_insumos
WHERE tipo_insumo IN ('S', 'T')
AND situacion IN (SELECT detalle
                  FROM otc_t_sgip_config
                  WHERE valor = 101);

t_reporte_inventario_count NUMBER DEFAULT 1;
existe NUMBER;
BEGIN
otc_pkg_sgip.otc_t_sp_up_asignado_notas (existe, num_serie);

FOR all_reporte_inventario IN c_reporte_inventario
LOOP
    tipo_insumo (t_reporte_inventario_count) :=
        all_reporte_inventario.tipo_insumo;
    num_serie (t_reporte_inventario_count) :=
        all_reporte_inventario.num_serie;
    id_caja (t_reporte_inventario_count) :=
        all_reporte_inventario.id_caja;
    num_imsi (t_reporte_inventario_count) :=
        all_reporte_inventario.num_imsi;
    ambiente (t_reporte_inventario_count) :=
        all_reporte_inventario.ambiente;
    fecha_creacion (t_reporte_inventario_count) :=
        all_reporte_inventario.fecha_creacion;
    situacion (t_reporte_inventario_count) :=
        all_reporte_inventario.situacion;
    modelo (t_reporte_inventario_count) := all_reporte_inventario.modelo;
    tecnologia (t_reporte_inventario_count) :=
        all_reporte_inventario.tecnologia;
    fecha_revision (t_reporte_inventario_count) :=
        all_reporte_inventario.fecha_revision;
    responsable_revision (t_reporte_inventario_count) :=
        all_reporte_inventario.responsable_revision;
    notas (t_reporte_inventario_count) := all_reporte_inventario.notas;
    t_reporte_inventario_count := t_reporte_inventario_count + 1;
    EXIT WHEN c_reporte_inventario%NOTFOUND;
END LOOP;
EXCEPTION
WHEN NO_DATA_FOUND
THEN
    NULL;
WHEN OTHERS
THEN
    errmsg := SUBSTR (SQLERRM, 1, 150);
    errcod := SUBSTR (TO_CHAR (SQLCODE), 1, 15);

END;

PROCEDURE otc_t_sp_insumos_activos (existe OUT NUMBER)
IS
BEGIN
    DELETE FROM otc_t_sgip_reportes;

    INSERT INTO otc_t_sgip_reportes
    SELECT a.num_serie, a.num_abonado, a.num_celular, a.cod_cliente,
           a.cod_cuenta, a.cod_situacion, b.cod_plantarif,
           b.des_plantarif, a.cod_ciclo, a.fec_alta, a.num_imei

```

```

FROM ga_abocel@otc_t_dbl_misc3_sclpreecu a,
     ta_plantarif@otc_t_dbl_misc3_sclpreecu b
WHERE a.cod_plantarif = b.cod_plantarif
      AND a.cod_situacion = 'AAA'
      AND a.num_serie IN (
          SELECT num_serie
          FROM otc_t_sgip_insumos
          WHERE tipo_insumo = 'S'
                AND ambiente = 'PRODUCCION')

UNION
SELECT a.num_serie, a.num_abonado, a.num_celular, a.cod_cliente,
       a.cod_cuenta, a.cod_situacion, b.cod_plantarif,
       b.des_plantarif, a.cod_ciclo, a.fec_alta, a.num_imei
FROM ga_aboamist@otc_t_dbl_misc3_sclpreecu a,
     ta_plantarif@otc_t_dbl_misc3_sclpreecu b
WHERE a.cod_plantarif = b.cod_plantarif
      AND a.cod_situacion = 'AAA'
      AND a.num_serie IN (
          SELECT num_serie
          FROM otc_t_sgip_insumos
          WHERE tipo_insumo = 'S'
                AND ambiente = 'PRODUCCION');

EXCEPTION
WHEN NO_DATA_FOUND
THEN
    NULL;
WHEN OTHERS
THEN
    errmsg := SUBSTR (SQLERRM, 1, 150);
    errcod := SUBSTR (TO_CHAR (SQLCODE), 1, 15);

INSERT INTO otc_t_sgip_errores
VALUES ('1034', 'OTC_T_SP_INSUMOS_ACTIVOS', errcod, errmsg,
        SYSDATE);

END;

PROCEDURE otc_sp_modifica_estado_usuario (
    aux_usu_cod IN otc_t_sgip_usuarios.usu_cod%TYPE,
    aux_estado IN otc_t_sgip_usuarios.usu_estado%TYPE,
    retorno OUT NUMBER
)
IS
    valida NUMBER;
BEGIN
    SELECT COUNT (*)
    INTO valida
    FROM otc_t_sgip_usuarios
    WHERE usu_cod = aux_usu_cod;

    IF valida > 0
    THEN
        UPDATE otc_t_sgip_usuarios
        SET usu_estado = aux_estado
        WHERE usu_cod = aux_usu_cod;

        retorno := 1;
    ELSE
        retorno := 0;
    END IF;
EXCEPTION
WHEN NO_DATA_FOUND
THEN

```

```

    retorno := 0;
    WHEN OTHERS
    THEN
        errmsg := SUBSTR (SQLERRM, 1, 150);
        errcod := SUBSTR (TO_CHAR (SQLCODE), 1, 15);

        INSERT INTO otc_t_sgip_errores
            VALUES ('1035', 'OTC_SP_MODIFICA_ESTADO_USUARIO', errcod,
                errmsg, SYSDATE);
    END;

    PROCEDURE otc_t_sp_reporte_activos (
        caja    OUT  tabla_number,
        serie   OUT  tabla_string,
        abonado OUT  tabla_number,
        celular OUT  tabla_number,
        cliente OUT  tabla_number,
        cuenta  OUT  tabla_number,
        codigo  OUT  tabla_string,
        PLAN   OUT  tabla_string,
        ciclo  OUT  tabla_number,
        alta   OUT  tabla_string,
        imei   OUT  tabla_string
    )
    IS
        CURSOR c_activos
        IS
            SELECT i.id_caja AS caja, r.num_serie AS serie,
                r.num_abonado AS abonado, r.num_celular AS celular,
                r.cod_cliente AS cliente, r.cod_cuenta AS cuenta,
                r.cod_plantarif AS codigo, r.des_plantarif AS PLAN,
                r.cod_ciclo AS ciclo, r.fec_alta AS alta, r.num_imei AS imei
            FROM otc_t_sgip_reportes r, otc_t_sgip_insumos i
            WHERE r.num_serie = i.num_serie
                AND i.tipo_insumo IN ('S', 'T')
                AND r.num_abonado > 0;

        activos_count NUMBER DEFAULT 1;
    BEGIN
        FOR all_activos IN c_activos
        LOOP
            caja (activos_count) := all_activos.caja;
            serie (activos_count) := all_activos.serie;
            abonado (activos_count) := all_activos.abonado;
            celular (activos_count) := all_activos.celular;
            cliente (activos_count) := all_activos.cliente;
            cuenta (activos_count) := all_activos.cuenta;
            codigo (activos_count) := all_activos.codigo;
            PLAN (activos_count) := all_activos.PLAN;
            ciclo (activos_count) := all_activos.ciclo;
            alta (activos_count) := all_activos.alta;
            imei (activos_count) := all_activos.imei;
            activos_count := activos_count + 1;
            EXIT WHEN c_activos%NOTFOUND;
        END LOOP;
    END;
    /

```

Tabla 4.37 Código duro paquete de base de datos.

4.2.2. METODOS Y FUNCIONES

A continuación se copia fragmentos de código representativos de los objetos de esta sección.

```

MASTER_PAGE

Imports System.Data
Imports System.Data.OleDb
Public Module maestro
    Dim limite As String
    Public COMANDO As OleDbCommand
    Public DROWS As DataRowCollection
    Public DRIDER As OleDbDataReader
    Public DSET As New DataSet()
    Public DTABLE As DataTable
    Public DROWC As DataRowCollection
    Public ecu_reportes As New _
        OleDbConnection(ConfigurationManager.ConnectionStrings("ecu_reportes").ConnectionString)
    Public Function Manejar_Errores(ByVal msg_error As String) As String
        limite = msg_error.Length
        If limite > 100 Then
            msg_error = msg_error.Substring(0, 50)
            Return msg_error
        Else
            msg_error = msg_error.Substring(0, limite)
            Return msg_error
        End If
    End Function
    Public Function FGVerificaRegistros(ByVal par As Integer) As Boolean
        Dim mensaje_error As String
        Try
            If (par <= 0) Then
                Return False
            Else
                Return True
            End If
        Catch ex As Exception
            mensaje_error = "Error al retornar # de registros: FGVerificaRegistros: " &
                Manejar_Errores(ex.Message)
            lblmsg.BackColor = Drawing.Color.Navy
        End Try
    End Function
    Public Function CreateCommand(ByVal qry As String) As OleDbCommand
        Dim ecu_reportes As New _
            OleDbConnection(ConfigurationManager.ConnectionStrings("ecu_reportes").ConnectionString)
        Return New OleDbCommand(qry, ecu_reportes)
    End Function

End Module

Partial Class MasterPage
    Inherits System.Web.UI.MasterPage
    'Dim limite As String
    Dim DT_MENUPERFIL As DataTable
    Dim SP_MENUPERFIL As String = (ConfigurationManager.AppSettings("SP_MENUPERFIL"))
    Dim CMD_MENUPERFIL As OleDbCommand
    Dim DS_MENUPERFIL As New DataSet()
    Dim DTC_MENUPERFIL As DataRowCollection
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load

```

```

Verificar_Opciones_Perfil()
I_fecha.Text = Now()
I_usuario_conectado.Text = Session("user")
DSET.Clear()
End Sub
Private Function FLlenaMenu() As Boolean
Dim rows As DataRowCollection
Dim row As DataRow
Dim i As Integer
Try
If TreeView1.Nodes.Count < 1 Then
rows = DT_MENUPERFIL.Rows
If Not FGVerificaRegistros(rows.Count) Then
lblmsg.Text = "No existen aplicaciones que desplegar: FLlenaMenu"
lblmsg.BackColor = Drawing.Color.Navy
Return False
End If

Dim nodo1 As New TreeNode("Menú de Opciones")
TreeView1.Nodes.Add(nodo1)
TreeView1.NodeStyle.ForeColor = Drawing.Color.Black
TreeView1.NodeStyle.Font.Size = 9
TreeView1.ExpandDepth = 3

Dim grupo As String = ""
Dim modulo As String = ""

Dim nodoG As TreeNode = New TreeNode()
Dim nodoM As TreeNode = New TreeNode()

For i = 0 To rows.Count - 1
row = rows.Item(i)
If modulo <> row(1).ToString Then
grupo = row(3).ToString()
nodoG = New TreeNode(grupo, row(2).ToString())

modulo = row(1).ToString()
nodoM = New TreeNode(modulo, row(0).ToString())

nodoG.ChildNodes.Add(New TreeNode(row(5).ToString(), row(4).ToString(), "",
row(6).ToString(), "_self"))
nodoM.ChildNodes.Add(nodoG)
Nodo1.ChildNodes.Add(nodoM)

Else
If grupo <> row(3).ToString() Then
grupo = row(3).ToString()
nodoG = New TreeNode(grupo, row(2).ToString())
nodoM.ChildNodes.Add(nodoG)

End If
nodoG.ChildNodes.Add(New TreeNode(row(5).ToString(), row(4).ToString(), "",
row(6).ToString(), "_self"))
End If
Next i

End If
Return True
Catch ex As Exception
lblmsg.Text = "Error al crear menu: FLlenaMenu: " & Manejar_Errores(ex.Message)
lblmsg.BackColor = Drawing.Color.Navy

```



```

End Try
End Function
Protected Sub Button1_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Button1.Click
Try
Response.Redirect("login.aspx")
Session("user") = ""
Session("rol") = ""
Catch ex As Exception
lblmsg.Text = "Error al cerrar sesión: " & Manejar_Errores(ex.Message)
lblmsg.BackColor = Drawing.Color.Navy
End Try
End Sub

Public Function Verificar_Opciones_Perfil() As Boolean
'sp compuesto por un cursor interno y un bucle
'llama al sp definido en el web.config esto para reutilizar codigo y minimizar errores por cambios
'funcion generica para crear el comando
CMD_MENUPERFIL = CreateCommand(SP_MENUPERFIL)
CMD_MENUPERFIL.Parameters.Add("AUX_COD_PERF", OleDbType.Integer, 2).Direction =
ParameterDirection.Input
CMD_MENUPERFIL.Parameters("AUX_COD_PERF").Value = Session("rol")
CMD_MENUPERFIL.CommandType = CommandType.Text
Dim ADAPTER_MENUPERFIL As New OleDbDataAdapter(CMD_MENUPERFIL)
If Session("user") = "DynamicString" Then
Response.Redirect("login.aspx")
Else
Try
ADAPTER_MENUPERFIL.Fill(DS_MENUPERFIL, "tabla")
DT_MENUPERFIL = DS_MENUPERFIL.Tables(0)
'ecu_reportes.Close()
FLlenaMenu()
DTC_MENUPERFIL = DT_MENUPERFIL.Rows
'funcion generica para verificar registros
If Not FGVerificaRegistros(DTC_MENUPERFIL.Count) Then
lblmsg.Text = "Alerta: Usuario no tiene opciones en su perfil: " & " Funcion
Verificar_Opciones_Perfil"
lblmsg.BackColor = Drawing.Color.Navy
End If
Catch ex As Exception
lblmsg.Text = "Error en Function Verificar_Opciones_Perfil: " & Manejar_Errores(ex.Message)
lblmsg.BackColor = Drawing.Color.Navy
End Try
End If
End Function
End Class

```

PERFILES

```

Imports System.Data.OleDb
Imports System.Data
Imports maestro
Partial Class _Perfiles
Inherits System.Web.UI.Page
Dim resultado, respuesta, AUX_COD_PERF As String
Dim AUX_ERROR As String = ""
Dim SP_PERFIL As String = (ConfigurationManager.AppSettings("SP_PERFIL"))
Private Function validar(ByVal msg_error As String) As String
If txt_nombre_perfil.Text = "" Then
msg_error = "Alerta: Ingresar la descripción del nuevo perfil"

```

```

End If
Return msg_error
End Function
Private Function Obtener_Perfiles() As String
COMANDO = CreateCommand(SP_PERFIL)
COMANDO.CommandType = CommandType.Text
Dim DA_PERFIL As New OleDbDataAdapter(COMANDO)
Dim dset_perfil As New DataSet
Try
dset_perfil.Clear()
DA_PERFIL.Fill(dset_perfil, "tabla")
GridView1.DataSource = dset_perfil.Tables("tabla")
GridView1.DataBind()
DTABLE = dset_perfil.Tables(0)
DROWC = DTABLE.Rows
'funcion generica para verificar registros
If Not FGVerificaRegistros(DROWC.Count) Then
msg.Text = "Aleta: No existen registros" & " Obtener_Perfiles"
End If
ecu_reportes.Close()
Return ""
Catch ex As Exception
msg.Text = "Error en Function Obtener_Perfiles:" & Manejar_Errores(ex.Message)
Return ""
End Try
End Function
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
If Not (IsPostBack) Then
Obtener_Perfiles()
End If
End Sub
Private Function Crear_Perfil() As Boolean
resultado = validar(respuesta)
If resultado <> "" Then
msg.Text = resultado
Else
Try
ecu_reportes.Open()
COMANDO = ecu_reportes.CreateCommand
COMANDO.Parameters.Add("AUX_PERF_DESC", OleDbType.VarChar).Direction =
ParameterDirection.Input
COMANDO.Parameters("AUX_PERF_DESC").Value = txt_nombre_perfil.Text
COMANDO.Parameters.Add("AUX_COD_ERROR", OleDbType.Integer, 4).Direction =
ParameterDirection.Output
COMANDO.Parameters.Add("ERRMSG", OleDbType.VarChar, 150).Direction =
ParameterDirection.Output
COMANDO.CommandText = "OTC_PKG_SGIP.OTC_T_SP_CREAR_PERFIL"
COMANDO.CommandType = CommandType.StoredProcedure
COMANDO.ExecuteNonQuery()
ecu_reportes.Close()
txt_nombre_perfil.Text = ""
AUX_COD_PERF = COMANDO.Parameters("AUX_COD_ERROR").Value
AUX_ERROR = COMANDO.Parameters("ERRMSG").Value
If AUX_COD_PERF = 99 Then
msg.Text = "Perfil creado correctamente"
msg.ForeColor = Drawing.Color.Blue
Elseif AUX_COD_PERF = 1007 Then
msg.Text = AUX_ERROR.Substring(0, 100)
End If
Catch ex As Exception
msg.Text = "Error en Function Crear_Perfil:" & Manejar_Errores(ex.Message)
End Try

```

```

End If
End Function
Protected Sub Button3_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Button3.Click
    Crear_Perfil()
End Sub
End Class

```

INSUMOS_ACTIVOS

```

Imports System.Data.OleDb
Imports System.Data
Imports maestro
Partial Class _Default
    Inherits System.Web.UI.Page
    Dim INSUMOS_ACTIVOS As String = (ConfigurationManager.AppSettings("INSUMOS_ACTIVOS"))
    Public Function CreateCommand(ByVal qry As String) As OleDbCommand
        Return New OleDbCommand(qry, ecu_reportes)
    End Function
    Private Function Consultar_Extrae_Insumos_Activos() As Boolean
        'sp ejecuta un select insert con un DBL sin parámetros de entrada
        Try
            ecu_reportes.Open()
            COMANDO = ecu_reportes.CreateCommand
            COMANDO.Parameters.Add("EXISTE", OleDbType.Integer, 3).Direction =
ParameterDirection.Output
            COMANDO.CommandText = "OTC_PKG_SGIP.OTC_T_SP_INSUMOS_ACTIVOS"
            COMANDO.CommandType = CommandType.StoredProcedure
            COMANDO.ExecuteNonQuery()
            ecu_reportes.Close()
        Catch ex As Exception
            msg.Text = "Error en Function Consultar_Insumos_Activos:" & Manejar_Errores(ex.Message)
        End Try
    End Function
    Private Function Consultar_Insumos_Activos() As Boolean
        'sp compuesto por un cursor interno y un bucle
        'ejecuta un sp sin parámetros para obtener una tabla y publicar contenido a un gridview
        'llama al sp definido en el web.config esto para reutilizar codigo y minimizar errores por cambios
        Consultar_Extrae_Insumos_Activos()
        COMANDO = CreateCommand(INSUMOS_ACTIVOS)
        COMANDO.CommandType = CommandType.Text
        Dim ADAPTER_SOLICITUDES As New OleDbDataAdapter(COMANDO)
        Dim dset_insumos_activos As New DataSet
        Try
            dset_insumos_activos.Clear()
            ADAPTER_SOLICITUDES.Fill(dset_insumos_activos, "tabla")
            GridView1.DataSource = dset_insumos_activos.Tables("tabla")
            GridView1.DataBind()
            DTABLE = dset_insumos_activos.Tables(0)
            DROWS = DTABLE.Rows
            'funcion generica para verificar registros
            If Not FGVerificaRegistros(DROWS.Count) Then
                msg.Text = "Alerta: No existen registros:" & " Consultar_Insumos_Activos"
                msg.ForeColor = Drawing.Color.Red
            Else
                msg.Text = ""
            End If
        Catch ex As Exception
            msg.Text = "Error en Function Consultar_Insumos_Activos: " & Manejar_Errores(ex.Message)
        End Try
        ecu_reportes.Close()
    End Function
End Class

```

```

Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
    Consultar_Insumos_Activos()
End Sub
End Class

```

CREAR_USUARIOS

```

Imports System.Data.OleDb
Imports System.Data
Imports maestro
Partial Class _Default
    Inherits System.Web.UI.Page
    Dim resultado, respuesta, AUX_COD_PERF As String
    Dim AUX_ERROR As String = ""
    Dim SP_PERFIL As String = (ConfigurationManager.AppSettings("SP_PERFIL"))
    Public Function CreateCommand(ByVal qry As String) As OleDbCommand
        Return New OleDbCommand(qry, ecu_reportes)
    End Function
    Public Function clear() As Boolean
        txt_usu_cod.Text = ""
        txt_nombre.Text = ""
        txt_apellido_actualiza.Text = ""
        txt_correo.Text = ""
        txt_apellido_actualiza.Text = ""
        txt_usu_red.Text = ""
        txt_ext.Text = ""
        txt_fijo.Text = ""
        txt_movil.Text = ""
    End Function
    Protected Sub Button3_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles
    Button3.Click
        Crear_Usuario()
    End Sub
    Private Function Crear_Usuario() As Boolean
        resultado = validar(respuesta)
        If resultado <> "" Then
            msg.Text = resultado
        Else
            Try
                ecu_reportes.Open()
                COMANDO = ecu_reportes.CreateCommand
                COMANDO.Parameters.Add("AUX_USU_COD_PERF", OleDbType.Integer, 3).Direction =
                ParameterDirection.Input
                COMANDO.Parameters("AUX_USU_COD_PERF").Value = I_perfil.SelectedValue
                COMANDO.Parameters.Add("AUX_USU_NOMBRE", OleDbType.VarChar).Direction =
                ParameterDirection.Input
                COMANDO.Parameters("AUX_USU_NOMBRE").Value = txt_nombre.Text.ToUpper
                COMANDO.Parameters.Add("AUX_USU_APELLIDO", OleDbType.VarChar).Direction =
                ParameterDirection.Input
                COMANDO.Parameters("AUX_USU_APELLIDO").Value = txt_apellido_actualiza.Text.ToUpper
                COMANDO.Parameters.Add("AUX_USU_CORREO", OleDbType.VarChar).Direction =
                ParameterDirection.Input
                COMANDO.Parameters("AUX_USU_CORREO").Value = txt_correo.Text.ToUpper
                COMANDO.Parameters.Add("AUX_USU_RED", OleDbType.VarChar).Direction =
                ParameterDirection.Input
                COMANDO.Parameters("AUX_USU_RED").Value = txt_usu_red.Text.ToUpper
                COMANDO.Parameters.Add("AUX_USU_PROVEEDOR", OleDbType.VarChar).Direction =
                ParameterDirection.Input
                COMANDO.Parameters("AUX_USU_PROVEEDOR").Value = I_proveedor.SelectedValue
                COMANDO.Parameters.Add("AUX_USU_ESTADO", OleDbType.VarChar).Direction =
                ParameterDirection.Input
                COMANDO.Parameters("AUX_USU_ESTADO").Value = "ACTIVO"
            Catch
            End Try
        End If
    End Function

```

```

        COMANDO.Parameters.Add("AUX_MOVIL", OleDbType.Integer).Direction =
ParameterDirection.Input
        COMANDO.Parameters("AUX_MOVIL").Value = txt_movil.Text
        COMANDO.Parameters.Add("AUX_FIJO", OleDbType.Integer).Direction =
ParameterDirection.Input
        COMANDO.Parameters("AUX_FIJO").Value = txt_fijo.Text
        COMANDO.Parameters.Add("AUX_EXT", OleDbType.Integer).Direction =
ParameterDirection.Input
        COMANDO.Parameters("AUX_EXT").Value = txt_ext.Text
        COMANDO.Parameters.Add("AUX_COD_ERROR", OleDbType.Integer, 4).Direction =
ParameterDirection.Output
        COMANDO.Parameters.Add("ERRMSG", OleDbType.VarChar, 150).Direction =
ParameterDirection.Output
        COMANDO.CommandText = "OTC_PKG_SGIP.OTC_SP_CREAR_USUARIOS"
        COMANDO.CommandType = CommandType.StoredProcedure
        COMANDO.ExecuteNonQuery()
        ecu_reportes.Close()
        clear()
        AUX_COD_PERF = COMANDO.Parameters("AUX_COD_ERROR").Value
        AUX_ERROR = COMANDO.Parameters("ERRMSG").Value
        If AUX_COD_PERF = 99 Then
            msg.Text = "Usuario creado correctamente"
            msg.ForeColor = Drawing.Color.Blue
        ElseIf AUX_COD_PERF = 1006 Then
            msg.Text = AUX_ERROR.Substring(0, 100)
        End If
        Catch ex As Exception
            msg.Text = "Error en Function Crear_Usuario:" & Manejar_Errores(ex.Message)
        End Try
    End If
End Function
Private Function validar(ByVal msg_error As String) As String
    If txt_nombre.Text = "" Then
        msg_error = "Alerta: Ingresar el nombre del usuario"
    ElseIf txt_apellido_actualiza.Text = "" Then
        msg_error = "Alerta: Ingresar el apellido del usuario"
    ElseIf txt_correo.Text = "" Then
        msg_error = "Alerta: Ingresar el correo del usuario"
    ElseIf txt_usu_red.Text = "" Then
        msg_error = "Alerta: Ingresar el usuario"
    End If
    Return msg_error
End Function
Private Function Obtener_Perfiles() As String
    COMANDO = CreateCommand(SP_PERFIL)
    COMANDO.CommandType = CommandType.Text
    Dim DA_USUARIO As New OleDbDataAdapter(COMANDO)
    Try
        DA_USUARIO.Fill(DSET, "tabla")
        DTABLE = DSET.Tables(0)
        DROWC = DTABLE.Rows
        'funcion generica para verificar registros
        If Not FGVerificaRegistros(DROWC.Count) Then
            msg.Text = "Aleta: No existen registros" & " Obtener_Perfiles"
        End If
        ecu_reportes.Open()
        DRIDER = COMANDO.ExecuteReader
        While DRIDER.Read()
            _perfil.Items.Add(New ListItem(DRIDER.Item("DESCRIPCION"), DRIDER.Item("CODIGO")))
        End While
        ecu_reportes.Close()
        Return ""
    End Try
End Function

```

```

Catch ex As Exception
    msg.Text = "Error en Function Obtener_Perfiles:" & Manejar_Errores(ex.Message)
    Return ""
End Try
End Function
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Load
    If Not (IsPostBack) Then
        Obtener_Perfiles()
    End If
End Sub
End Class

```

FOTO_INVENTARIO

```

Imports System.Data.OleDb
Imports System.Data
Imports maestro
Imports System.IO
Partial Class _Default
    Inherits System.Web.UI.Page
    Dim REPORTE_INVENTARIO As String =
(ConfigurationManager.AppSettings("SP_REPORTE_INVENTARIO"))

    Private Function Consultar_Situacion_Inventario() As Boolean
        Dim sb As StringBuilder = New StringBuilder()
        Dim sw As StringWriter = New StringWriter(sb)
        Dim htw As HtmlTextWriter = New HtmlTextWriter(sw)
        Dim pagina As Page = New Page
        Dim form = New HtmlForm
        'sp compuesto por un cursor interno y un bucle
        'ejecuta un sp con un parámetro para obtener una tabla y publicar contenido a un gridview
        'llama al sp definido en el web.config esto para reutilizar código y minimizar errores por cambios
        COMANDO = CreateCommand(REPORTE_INVENTARIO)
        COMANDO.CommandType = CommandType.Text
        Dim ADAPTER_REPORTE As New OleDbDataAdapter(COMANDO)
        Dim dset_rep_inventario As New DataSet
        Try
            dset_rep_inventario.Clear()
            ADAPTER_REPORTE.Fill(dset_rep_inventario, "tabla")
            GridView1.DataSource = dset_rep_inventario.Tables("tabla")
            GridView1.DataBind()
            DTABLE = dset_rep_inventario.Tables(0)
            DROWS = DTABLE.Rows
            'funcion generica para verificar registros
            If Not FGVerificaRegistros(DROWS.Count) Then
                msg.Text = "Alerta: No existen registros:" & " Consultar_Situacion_Inventario"
                msg.ForeColor = Drawing.Color.Red
            Else
                msg.Text = ""
                pagina.EnableEventValidation = False
                pagina.DesignerInitialize()
                pagina.Controls.Add(form)
                form.Controls.Add(GridView1)
                pagina.RenderControl(htw)
                Response.Clear()
                Response.Buffer = True
                Response.ContentType = "application/vnd.ms-excel"
                Response.AddHeader("Content-Disposition", "attachment;filename=Reporte_Foto_Inventario_" &
Now & ".xls")
                Response.Charset = "@"
                Response.ContentEncoding = Encoding.Default
            End If
        Catch ex As Exception
            msg.Text = "Error en Function Obtener_Perfiles:" & Manejar_Errores(ex.Message)
        End Try
    End Function
End Class

```

```

        Response.Write(sb.ToString().ToUpper)
        Response.End()
        GridView1.Visible = False
    End If
    Catch ex As Exception
        msg.Text = "Error en Function Consultar_Situacion_Inventario: " & Manejar_Errores(ex.Message)
    End Try
    ecu_reportes.Close()
End Function
Protected Sub GridView1_RowDataBound(ByVal sender As Object, ByVal e As
System.Web.UI.WebControls.GridViewRowEventArgs) Handles GridView1.RowDataBound
'coloca un caracter al final de la columna 1 del datagrid
'esto para evitar que la información de la columna pierda el formato de texto y la informacion
If e.Row.RowType = DataControlRowType.DataRow Then
    e.Row.Cells(1).Text = e.Row.Cells(1).Text & ""
    e.Row.Cells(3).Text = e.Row.Cells(3).Text & ""
End If
End Sub
Protected Sub Button2_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Button2.Click
    Consultar_Situacion_Inventario()
End Sub
End Class

```

ASIGNAR_RECURSOS

```

Imports System.Data.OleDb
Imports System.Data
Imports System.IO
Imports maestro
Partial Class _Default
    Inherits System.Web.UI.Page
    Dim resultado As String
    Dim SOLICITUDES As String = (ConfigurationManager.AppSettings("SP_SOLICITUDES"))
    Dim SP_ASIGNA_INSUMOS As String =
(ConfigurationManager.AppSettings("SP_ASIGNA_INSUMOS"))
    Dim SP_USUARIOS As String = (ConfigurationManager.AppSettings("SP_USUARIOS"))
    Dim Carga_Archivos As String = (ConfigurationManager.AppSettings("Carga_Archivos"))
    Public Function CreateCommand(ByVal qry As String) As OleDbCommand
        Return New OleDbCommand(qry, ecu_reportes)
    End Function
    Public Function Carga_Archivo() As Boolean
        Dim saveDir_server As String = Server.MapPath(".")
        Dim extrernal_appPath As String = "\tmp"
        If (FileUpload1.HasFile) Then
            FileUpload1.SaveAs(saveDir_server & extrernal_appPath & "\" & FileUpload1.FileName)
            Dim client_FileName As String = System.IO.Path.GetFileName(FileUpload1.PostedFile.FileName)
            Dim server_FileDir As String = Carga_Archivos
            Dim sContent As String = vbNullString
            Dim archivo As String = server_FileDir & client_FileName
            Dim oSW As New StreamReader(archivo)
            Dim SPath As String = archivo
            Dim aux As Integer = 0
            Dim aux2 As Integer = 0
            Try
                ecu_reportes.Open()
                Dim delete As String = "delete from OTC_T_SGIP_TMP"
                COMANDO = New OleDbCommand(delete, ecu_reportes)
                resultado = COMANDO.ExecuteNonQuery

                If oSW.ReadLine.Contains(",") Then
                    msg.Text = "Error en datos internos del archivo, existe más de una columna"
                End If
            Catch ex As Exception
                Manejar_Errores(ex.Message)
            End Try
        End If
    End Function
End Class

```

```

Else
    With My.Computer.FileSystem
        ' verifica si existe el path
        If .FileExists(SPath) Then
            Do While Not oSW.EndOfStream
                sContent = oSW.ReadLine()
                If sContent <> "" Then
                    aux += 1
                    Dim count_serie = "select count(*) from OTC_T_SGIP_TMP where num_serie=" &
sContent & ""

                    COMANDO = New OleDbCommand(count_serie, ecu_reportes)
                    Dim val As Integer = COMANDO.ExecuteScalar
                    If val = 1 Then
                        aux2 += 1
                    Else
                        Dim insert As String = "insert into OTC_T_SGIP_TMP values (" & sContent &
",sysdate)"

                        COMANDO = New OleDbCommand(insert, ecu_reportes)
                        resultado = COMANDO.ExecuteNonQuery
                    End If
                End If
            Loop
        End With
        b_asignar.Visible = True
        ecu_reportes.Close()
        Dim total As Integer = aux - aux2
        msg.Text = "[" & total & "]" & " Insumos cargados correctamente, actualizar la información
según opciones disponibles"
        msg.ForeColor = Drawing.Color.Blue

        End If
        oSW.Close()
        ecu_reportes.Close()
    Catch ex As Exception
        ecu_reportes.Close()
        msg.Text = "Error en Function Carga_Archivo:" & Manejar_Errores(ex.Message)
        msg.ForeColor = Drawing.Color.Red
    End Try
Else
    msg.Text = "Alerta: Seleccione un archivo para cargar"
    msg.ForeColor = Drawing.Color.Red
End If
End Function

Protected Sub Button2_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles
Button2.Click
    Carga_Archivo()
End Sub

Private Function Registrar_Existencias() As Boolean
    'sp compuesto por un cursor interno y un bucle
    'utiliza la funcionalidad de carga de archivo para insertar los registros a la base
    Dim usuario As Integer = I_solicitud.Selected.Value
    msg.Text = ""
    COMANDO.Parameters("AUX_SOLICITUD").Value = usuario
    If usuario = 1 Then 'Entrega al administrador
        COMANDO.Parameters.Add("AUX_RECIBE", OleDbType.VarChar).Direction =
ParameterDirection.Input
        COMANDO.Parameters("AUX_RECIBE").Value = Session("user")
        COMANDO.Parameters.Add("AUX_ENTREGA", OleDbType.VarChar).Direction =

```



```

ParameterDirection.Input
    COMANDO.Parameters("AUX_ENTREGA").Value = I_usuarios.SelectedVale
Else
    COMANDO.Parameters.Add("AUX_ENTREGA", OleDbType.VarChar).Direction =
ParameterDirection.Input
    COMANDO.Parameters("AUX_ENTREGA").Value = Session("user")
    COMANDO.Parameters.Add("AUX_RECIBE", OleDbType.VarChar).Direction =
ParameterDirection.Input
    COMANDO.Parameters("AUX_RECIBE").Value = I_usuarios.SelectedVale
End If

Dim DA_INSUMO As New OleDbDataAdapter(COMANDO)
Try
    DSET.Clear()
    DA_INSUMO.Fill(DSET, "tabla")
    DTABLE = DSET.Tables(0)
    ecu_reportes.Close()
    msg.Text = "Proceso realizado correctamente"
Catch ex As Exception
    msg.Text = "Error en Function Asignar_Recursos: " & Manejar_Errores(ex.Message)
End Try
ecu_reportes.Close()
End Function

Private Function Usuarios_Permitidos() As String
    COMANDO = CreateCommand(SP_USUARIOS)
    COMANDO.CommandType = CommandType.Text
    Dim DA_USUARIO As New OleDbDataAdapter(COMANDO)
    Try
        DSET.Clear()
        DA_USUARIO.Fill(DSET, "tabla")
        DTABLE = DSET.Tables(0)
        maestro.DROWC = DTABLE.Rows
        'funcion generica para verificar registros
        If Not FGVerificaRegistros(DROWC.Count) Then
            msg.Text = "Aleta: No existen registros" & " Usuarios_Permitidos"
        End If
        ecu_reportes.Open()
        DRIDER = COMANDO.ExecuteReader
        While DRIDER.Read()
            I_usuarios.Items.Add(New ListItem(DRIDER.Item("USU_NOMBRE"),
DRIDER.Item("USU_RED")))
        End While
        ecu_reportes.Close()
        Return ""
    Catch ex As Exception
        msg.Text = "Error en Function Usuarios_Permitidos:" & Manejar_Errores(ex.Message)
        Return ""
    End Try
End Function
End Class

```

Tabla 4.38 Código duro metodos y funciones .NET

Entregable MSF "Informe de pruebas Sistema de Inventarios"

CAPITULO 5. PRUEBAS Y DOCUMENTACION

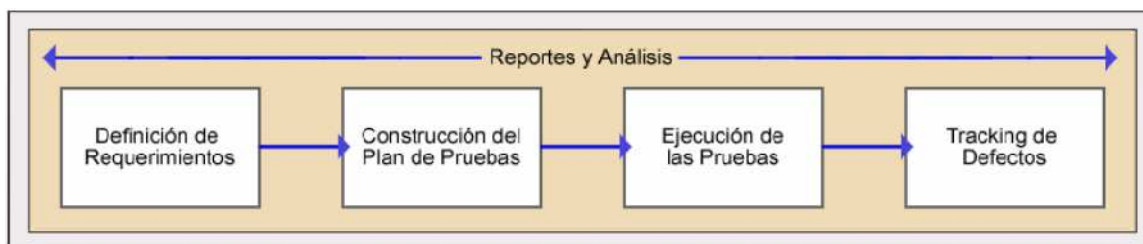
5.1. PRUEBAS

5.1.1. INTRODUCCION

La presente sección pretende describir de forma general el una parte del proceso de pruebas realizado en la compañía OTECEL.S.A, por la cual todos los productos de software son evaluados.

Los requerimientos de pruebas proveen una descripción detallada de las necesidades de testing de una aplicación.

Estas necesidades son identificadas inicialmente en el detalle funcional del usuario (DFU) el mismo que es elaborado en el proceso de gestión de cambios de la compañía. La estrategia de pruebas es elaborada tomando como base el DFU aprobado por los involucrados en la necesidad planteada.



*Figura 5.1 Proceso de pruebas OTECEL.S.A⁸²
Ciclo de vida de la gestión de cambios.*

Una estrategia de pruebas se construye considerando los siguientes elementos:

- **Definición de Requerimientos**

Se toma como información de entrada el detalle funcional de usuario, además de otros documentos anexos como plantillas, actas de reunión, etc. En donde se define el alcance de las pruebas y los recursos necesarios para la ejecución; en donde están usuarios, tiempos estimados y otros elementos que pueden variar según el requerimiento.

⁸² Fuente: Gráfico de gestión de Cambios propietaria de OTECEL S.A

- **Construcción del Plan de Pruebas**

Los planes de prueba o llamado también estrategia de pruebas, se construyen utilizando la metodología de casos de uso.

Un escenario de pruebas es creado a través de un caso de uso.

Un plan de pruebas puede tener varios escenarios.

Un escenario de pruebas puede tener uno o más pasos

- **Ejecución de las Pruebas**

Existen herramientas automatizadas o manuales que dependerán de la empresa, los escenarios de prueba son ejecutados a partir de los pasos definidos en él.

Un paso contiene un secuencial, una descripción del paso, un resultado esperado y un resultado obtenido u observación.

Para realizar una prueba, se ejecuta un paso a continuación de otro y se registra el resultado obtenido.

Si el resultado esperado no se cumple, respecto al resultado obtenido produce la creación de un defecto. El mismo que deberá ser anotado con la suficiente información para que el responsable de revisar el defecto lo pueda solucionar.

- **Tracking de Defectos⁸³**

En el proceso de pruebas, un defecto cumple un ciclo de vida, esto permite la colaboración y compromiso de los miembros del equipo de trabajo. Desde que el defecto es creado hasta que es resuelto.

⁸³ Para el proceso de pruebas de OTECEL. S.A un Defecto es cuando un caso de prueba no cumple el resultado esperado.

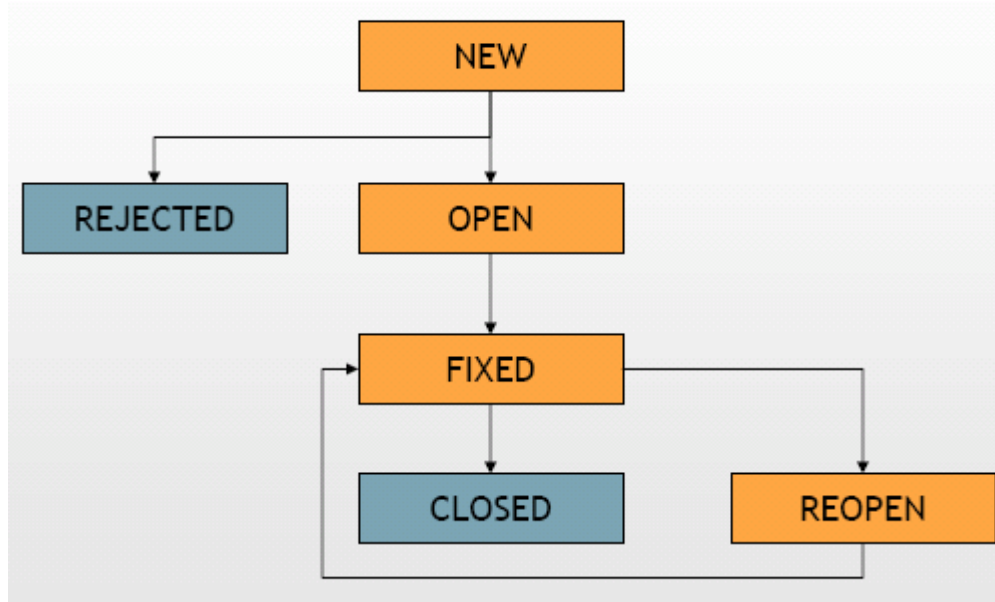


Figura 5.2 Gestión de defectos ⁸⁴
Ciclo de vida para el cierre y gestión de defectos.

- **NEW.-** el ingeniero de pruebas abre un defecto u observación y asigna a un responsable.
- **OPEN.-** el responsable acepta el defecto u observación.
- **FIXED.-** el responsable confirma haber revisado, corregido y verificado que el defecto u observación fue corregido.
- **REOPEN.-** el ingeniero de pruebas verifica que un defecto que fue corregido, continúa con el error.
- **CLOSED.-** el ingeniero de pruebas certifica que el defecto u observación está solucionado.
- **REJECTED.-** el responsable rechaza el defecto u observación y lo reasigna a otro responsable.

5.1.2. PRUEBAS INTERNAS

Las pruebas internas en el proceso de gestión de cambios de OTECEL.S.A corresponden a pruebas de funcionalidad del producto, de acuerdo a un

⁸⁴ Fuente: Gráfico de gestión de Cambios propietaria de OTECEL S.A

requerimiento específico. El mismo que debe cumplir un alcance definido y detallado.

No está permitido que un alcance se solicite verificar “toda” la funcionalidad de un producto, es muy general y no tiene un límite.

Las pruebas internas pueden iniciar solamente cuando la construcción y/o configuración del requerimiento ha sido concluido, el entregable que evidencia lo anteriormente expuesto se denomina MAU (Manual de Administración y Usuario) el cual contiene la evidencia de las pruebas realizadas por los desarrolladores, en donde certifican la correcta funcionalidad del producto.

El área de TEST o pruebas, toma este documento (puede existir más documentos adjuntos dependiendo del tipo de requerimiento) e inicia el proceso de pruebas internas el cual consiste en verificar la autenticidad de lo documentado en el MAU.

Estas pruebas son realizadas en una primera fase únicamente por el TESTER, si todo está correcto realiza una segunda fase involucrando al usuario solicitante del requerimiento.

Si el plan de pruebas ejecutado en conjunto con el usuario está correcto entonces se aprueba la fase de pruebas internas y en entonces cuando puede pasar a la siguiente fase llamada pruebas formales.

Por el contrario si el proceso de pruebas internas existe un defecto, este se crea y es asignado a un responsable para su tratamiento, el requerimiento retorna a la fase anterior que es construcción y pasará una vez solucionado el o los defectos.

5.1.3. PRUEBAS FORMALES

El proceso de pruebas formales consiste en reunir a todos o a la mayoría de los involucrados en el requerimiento solicitado por el usuario.

En un requerimiento pueden existir: administradores de plataformas, usuarios del área comercial, ventas, facturación, servicio al cliente entre otros.

Se ejecuta el mismo plan de pruebas de la fase anterior (pruebas internas) y se demuestra a todos los involucrados la correcta funcionalidad del producto según el plan de pruebas.

Al concluir la sesión de pruebas, el requerimiento es aprobado y pasa a las subsiguientes fases que comprenden versionamiento de fuentes, comité para paso a producción, revisión de documentos, elaboración de documentos adicionales del proceso, calendarización del paso a producción, cierre del requerimiento y pruebas del producto en ambientes productivos si así lo requiere.

5.2. DOCUMENTACION

5.2.1. INTRODUCCION

En esta sección se pretende describir de manera general el propósito y contenido del manual de administración y usuario. Este tipo de documentos varían de acuerdo a cada empresa sin embargo el objetivo final es el mismo para todos.

5.2.2. MANUAL DE ADMINISTRACION Y USUARIO (MAU)

El manual de administración y usuario de OTECEL. S.A es un documento fundamental en el proceso de gestión de cambios de la compañía.

Sin este documento no hay requerimiento alguno que pueda ser puesto como productivo.

Este documento tiene varias versiones en el proceso de gestión de cambios, el MAU⁸⁵ versión cero es elaborado por el proveedor de desarrollo cuando un usuario crea una petición de cambio. Para esto existen varias reuniones de factibilidad técnica en donde se analizan aspectos de requisitos para desarrollo, ambientes, limitaciones, riesgos, aclaración y/o modificación del alcance.

⁸⁵ Fuente – Documento de procesos propietario de OTECEL. S.A

Este documento es verificado y certificado por los coordinadores de construcción, pudiendo existir varias versiones si la propuesta no cumple con las necesidades del usuario.

La versión aprobada toma el nombre de versión uno y es con esta que se crean los planes de prueba iniciales.

A continuación se describe el contenido de un MAU de OTECEL. S.A

- Propuesta de solución funcional.- Esta sección contiene el alcance específico de la petición de cambio descrita en el requerimiento del usuario.
- Propuesta de solución técnica.- Esta sección explica cual es la alternativa más adecuada y recomendada para cubrir el alcance. Existen diagramas de arquitectura, de flujo, pseudocódigo. En la versión uno, esta sección comprende más detalle como objetos de bases de datos y/o código duro.
- Planificación.- En esta sección se plantea un cronograma macro del tiempo que toma implementar la solución; desde el análisis, construcción, pruebas unitarias, documentación. Es sujeta a modificación de fechas, presupuesto, aprobación.
- Construcción.- Esta sección es documentada cuando el requerimiento es construido, comprende los ambientes de desarrollo, bases de datos, diagrama de clases, procesos dependientes, descripción de alarmas, archivos de entrada / salida, archivos de configuración, código duro.
- Pruebas Unitarias.- Esta sección contiene planes de ejecución si el requerimiento tiene desarrollo en base de datos, capturas de pruebas de código, pantallas, envío de notificaciones, correos, logs. Toda la evidencia necesaria que demuestre que la construcción fue verificada.
- Paso a producción.- Esta sección explica el como debe ser instalado el producto de software, el orden en el cual se deben ejecutar los scripts, copia de archivos, instalación de servicios, lugar en donde están los fuentes. Tiene además una sección de reversa, esto como medida de seguridad el momento del paso a producción para que si los operadores

detectan el mínimo error se ejecutará el plan de reversa que consiste en dejar los aplicativos y plataformas tal como estaban antes de ejecutar el plan de paso a producción.

- Manual de usuario.- Esta sección explica de manera detallada como funciona el producto de software creado o modificado. Esta sección es de mucha ayuda para los usuarios de negocio que no tienen conocimientos técnicos.
- Anexos.- Esta sección puede tener archivos complementarios que se relacionen con el requerimiento de cambio.

CONCLUSIONES.

- El esquema de base de datos desarrollado se ajusta a las necesidades del proceso de gestión y control de inventarios.
- Se han utilizado los conceptos más importantes de programación PL/SQL, como son Stored Procedures, Funciones, Triggers; permitiendo que la lógica de negocio sea segura y centralizada en la base de datos.
- Con la instalación del Sistema de Inventarios Pruebas, el administrador del proceso logró mayores beneficios, ya que los procesos por él manejados fueron automatizados. Obteniendo tiempos de respuesta adecuados para los usuarios y un control adecuado de la información.
- Las interfaces de usuario fueron desarrolladas tomando los lineamientos básicos de la compañía, en cuanto a colores, logotipos y funcionalidad.
- Se cumplió el objetivo de desarrollar y poner en producción, un aplicativo para el apoyo a la gestión y control de los recursos físicos y lógicos asignados a los colaboradores de la Gerencia de Construcción Área Pruebas en OTECEL.S.A

RECOMENDACIONES.

- Los procesos manuales y recurrentes de negocio, deben ser creados con una lógica codificada y manejada a nivel de base de datos, esto proporcionará alta seguridad.
- Los aplicativos WEB deben ser livianos y con la menor cantidad de sentencias PL/SQL como SELECT, INSERT, UPDATE y DELETE. En su lugar se deben invocar STORED PROCEDURES de los paquetes de bases de datos.
- Disponer de un proceso de respaldos, deben ser recurrentes para evitar pérdidas de información, en cuanto a objetos de bases de datos y fuentes a nivel a aplicativo WEB.
- Se debe contar con un proceso de recuperación, es decir; disponer de un servidor de respaldo en donde se pueda instalar el servidor WEB, esto permitirá una adecuada y rápida recuperación del servicio al usuario en caso de una falla inesperada en el servidor.

BIBLIOGRAFIA.

Referencias online:

- MSDN Developer Centers
<http://msdn.microsoft.com/developercenters>
- Visual Studio® Team System Developer Center
<http://msdn.microsoft.com/vstudio/teamsystem>
- Introduction to Windows Forms Development
<http://msdn.microsoft.com/vbasic/learning/windowsforms>
- Programación en Visual Basic .NET” Editorial Eidos. 2002.
<http://guille.costasol.net/NET/cursoVB.NET>
- Microsoft’s MSF@TechNet
<http://www.microsoft.com/technet/itsolutions/msf/default.aspx>
- Microsoft’s MSF Home
<http://www.microsoft.com/msf>
- White Paper Microsoft Solutions Framework
http://www.uml.org.cn/softwareprocess/MSF_v3_OverviewWhitepaper.pdf

Libros:


- David E. Davila Fontecha. Programación Orientada a Objetos con VS.NET 2005
- Robin A. Reynolds-Haertle. OOP With Microsoft Visual Basic .Net and Microsoft Visual C# .Net
- Francisco Charte Ojeda. “Microsoft Visual Basic .NET” Ediciones Anaya Multimedia
(Grupo Anaya, S.A.). 2001.
- Benjamin Rosenzweig. Oracle PL/SQL by Example, Third Edition
- Jason Price. Oracle Database 10g SQL (Osborne ORACLE Press Series)
- Donald K. Burleson. Oracle Internals: Tips, Tricks, and Techniques for DBAs

ANEXOS.

Anexo 1.

Plan Maestro Construcción Sistema Inventarios

Anexo 2.**Tabulación resultados casos de pruebas**

OTECEL S.A. VP de Tecnología de Información	Tabulación Resultados Pruebas de Proyectos	CÓDIGO:								
		Fecha de emisión:								
		ene-12								
		VERSIÓN:								
		1.0.5								
Proyecto:	Sistema de Gestión y Control de Recursos Físicos y Logicos.									
Líder de Proyecto:	Efren Zambrano									
Usuario Funcional:	Gabriel Carvajal									
Ingeniero Desarrollo:	Roberto Zumba									
Casos de Prueba		Paso # - Plan #								
1.1_SGIP_Administración_AutenticarUsuario	1 - 1	2 - 1	3 - 1	1 - 1	4 - 1					
1.2_SGIP_Administración_ModificarClave	1 - 1	2 - 2	2 - 1	1 - 1	2 - 1	4 - 1				
1.3_SGIP_Administración_CrearPerfiles	1 - 1	2 - 1	3 - 4	4 - 4						
1.4_SGIP_Administración_CrearUsuarios	1 - 1	2 - 1	3 - 5	4 - 5						
1.5_SGIP_Administración_ModificarUsuario	1 - 1	2 - 1	3 - 3	4 - 3	6 - 3	7 - 3	8 - 3	9 - 3	10 - 3	
1.6_SGIP_Administración_MantenedorInsumos	1 - 1	2 - 2	3 - 6	4 - 6	5 - 6	6 - 6	5 - 6	7 - 6	8 - 6	
1.7_SGIP_Servicios_CrearSolicitudes	1 - 1	2 - 1	3 - 7	4 - 7						
1.8_SGIP_Servicios_ConsultarSolicitudes	1 - 1	2 - 1	3 - 8	4 - 8	5 - 8					
1.9_SGIP_Servicios_AsignarInsumos	1 - 1	2 - 1	3 - 9	5 - 6	4 - 9	3 - 8				
2.1_SGIP_Servicios_ProcesarSolicitudes	1 - 1	2 - 1	3 - 10	4 - 10	5 - 10	6 - 10	7 - 10			
2.2_SGIP_Servicios_IngresarNuevosInsumos	1 - 1	2 - 1	3 - 11	5 - 6	4 - 11					

2.3_SGIP_Reportes_SituacionInventario	1 - 1	2 - 1	3 - 12	4 - 12	5 - 12	6 - 12			
2.4_SGIP_Reportes_InsumosActivos	1 - 1	2 - 1	3 - 13	6 - 12					
2.5_SGIP_Reportes_AuditorialInsumos	1 - 1	2 - 1	3 - 14	4 - 14	6 - 12				
2.6_SGIP_Reportes_ConsultarExistencias	1 - 1	2 - 1	3 - 15	6 - 12					
2.7_SGIP_Reportes_DevolucionesUsuario	1 - 1	2 - 1	3 - 16	4 - 16	6 - 12				
2.8_SGIP_Reportes_InsumosAsignados	1 - 1	2 - 1	3 - 17	4 - 17	6 - 12	5 - 8	6 - 12	6 - 17	4 - 12
2.9_SGIP_Sistema_VinculosNavegacion	1 - 1	2 - 1	3 - 18	6 - 12					
Resultado ejecución plan de prueba - paso									
Casos de Prueba	Plan	Plan	Plan	Plan	Plan	Plan	Plan	Plan	Plan
1.1_SGIP_Administración_AutenticarUsuario	OK	OK	OK	OK	OK				
1.2_SGIP_Administración_ModificarClave	OK	OK	OK	OK	OK	OK			
1.3_SGIP_Administración_CrearPerfiles	OK	OK	OK	OK					
1.4_SGIP_Administración_CrearUsuarios	OK	OK	OK	OK					
1.5_SGIP_Administración_ModificarUsuario	OK	OK	OK	OK	OK	OK	OK	OK	OK
1.6_SGIP_Administración_MantenedorInsumos	OK	OK	OK	OK	OK	OK	OK	OK	OK
1.7_SGIP_Servicios_CrearSolicitudes	OK	OK	OK	OK					
1.8_SGIP_Servicios_ConsultarSolicitudes	OK	OK	OK	OK	OK				
1.9_SGIP_Servicios_AsignarInsumos	OK	OK	OK	OK	OK	OK			
2.1_SGIP_Servicios_ProcesarSolicitudes	OK	OK	OK	OK	OK	OK	OK		
2.2_SGIP_Servicios_IngresarNuevosInsumos	OK	OK	OK	OK	OK				
2.3_SGIP_Reportes_SituacionInventario	OK	OK	OK	OK	OK	OK			
2.4_SGIP_Reportes_InsumosActivos	OK	OK	OK	OK					

Descripción de los pasos por cada plan de prueba.

Descripción	Pasos	Plan
Ingresar al explorador web y digitar dirección	1	1
Digitar la clave	2	1
Presionar botón cerrar sesión	3	1
Digitar clave no válida	4	1
Cambiar contraseña	2	2
Seleccionar formulario "Modificar Usuario"	3	3
Buscar un usuario existente por apellido	4	3
Buscar un usuario existente ingresando combinación de letras del apellido.	5	3
busqueda de un usuario no existente por apellido	6	3
El usuario permitido ingresa un código de usuario al cual modificará sus datos y presiona TAB	7	3
Modificar datos del usuario	8	3
Modificar estado del usuario	9	3
Modificar perfil del usuario	10	3
Seleccionar formulario "Crear Perfiles"	3	4
Crear perfil.	4	4
Seleccionar formulario "Crear Usuarios"	3	5
Ingresar datos y crear un nuevo usuario	4	5
Selecciona formulario "Mantenedor de Insumos"	3	6
ingresar una serie (simcard o terminal) y selecciona eliminar	4	6
Realizar la carga de un archivo plano	5	6
Ingresar un número de caja y seleccionar actualizar 1	6	6
Seleccionar del combobox el ambiente deseado y seleccionar el botón actualizar 2	7	6
Seleccionar del combobox el estado deseado y seleccionar el botón actualizar 3	8	6
Seleccionar formulario "Crear Solicitudes"	3	7
Ingresar los datos y selecciona crear solicitud	4	7
Selecciona formulario "Consultar Solicitudes"	3	8
Buscar solicitudes ingresando fecha inicio y fecha fin	4	8
Buscar solicitudes ingresando una solicitud	5	8
Seleccionar formulario "Asignar Insumos"	3	9
Realiza asignación o recepción de insumos	4	9
Seleccionar formulario "Procesar Solicitudes"	3	10
Negar y/o procesar una solicitud	4	10
Aplicar una prórroga de 10 días a una solicitud	5	10
Verifica fecha devolución de una solicitud	6	10
Re abrir una solicitud	7	10
Seleccionar formulario "Crear Insumos"	3	11
Completar datos y selecciona crear insumos	4	11
Seleccionar formulario "Situación actual del inventario"	3	12
Seleccionar exportar datos	4	12
Seleccionar abrir archivo y/o guardar	5	12

Verificar información	6	12
Seleccionar formulario "Insumos Activos"	3	13
Seleccionar formulario "Auditoria por Insumo"	3	14
Ingresar un número de serie y selecciona exportar datos.	4	14
Seleccionar formulario "Existencia Actual de Insumos"	3	15
Seleccionar formulario "Devoluciones por usuario"	3	16
Seleccionar una fecha y un usuario, luego consultar	4	16
Seleccionar formulario "Insumos asignados"	3	17
Seleccionar un usuario y consultar	4	17
Seleccionar opción buscar todo	6	17
Ingresar a todos los formularios del sistema	3	18

Anexo 3.**Estrategia pruebas Sistema de Gestión Inventario de Pruebas**

Anexo 4.

Informe Pruebas Sistema de Gestión Inventario de Pruebas

Anexo 5.

Manual de Usuario Sistema de Inventario de Pruebas

RF / OTT	114524	Proyecto	N/A
Proveedor	N/A	Coordinador	N/A
Coordinador	N/A	Movistar	
Proveedor		Fecha último cambio:	11 de Enero 2012
Preparado por:	Roberto Zumba		
Descripción	Sistema de Gestión y Control del Inventario de Pruebas		
Datos	Roberto Zumba		
Desarrollador:	Roberto.Zumba@telefonica.com		
	Teléfono: 2227700 Ext. 3237		

1. PROPUESTA DE SOLUCIÓN FUNCIONAL

DENTRO DEL ALCANCE

Implementar el sistema SGCIP (Sistema de Gestión y Control del Inventario de Pruebas)

El SGCIP será una aplicación que funcionará en un entorno WEB que permitirá administrar, controlar y clasificar la información de los recursos físicos y/o lógicos asignados al Área de Pruebas para el uso de los usuarios internos. Esta aplicación ofrecerá soporte a las actividades definidas en el procedimiento interno: "PRTISS0601 Gestión y Control del Inventario de Pruebas.pdf", del cual se cubrirán las siguientes:

- Verificar disponibilidad de inventario
- Controlar la asignación de inventario
- Verificar la rotación y reutilización de inventario
- Registrar situación del inventario
- Devolver el Inventario
- Agrupar Inventario
- Confirmar disponibilidad en sistemas
- Identificar estado y funcionalidad del inventario
- Clasificar inventario
- Registrar estado del inventario de Terminales
- Registrar estado del inventario de Simcards

El SGCIP ofrecerá información confiable sobre el estado del inventario de pruebas, así también de procedimientos óptimos y adecuados para los usuarios del sistema.

El SGCIP tendrá la funcionalidad de gestión de claves para los usuarios que utilizarán el sistema, es decir que podrán cambiar su clave de acceso si son usuarios permitidos en el sistema.

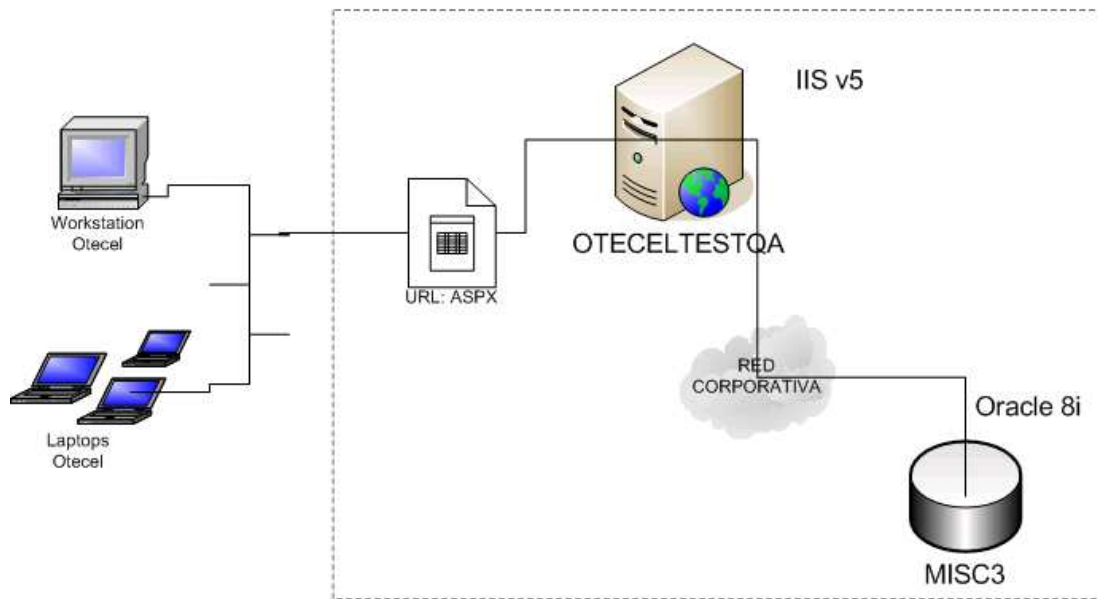
El SGCIP contará con reportes que permitirán disponer de información de primera mano para el análisis y control de la gestión que se lleva a cabo y el acceso será a través de perfiles para cumplir los lineamientos de seguridad de información de la compañía.

FUERA DE ALCANCE

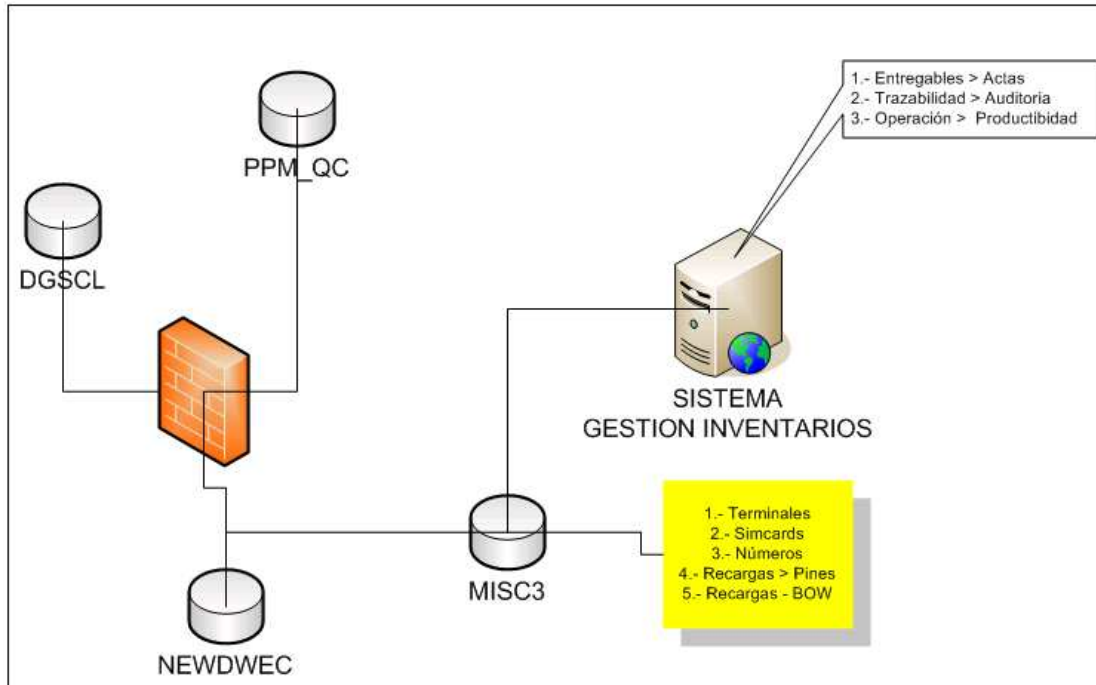
- Notificaciones de correo automáticas.
- Funcionalidad adicional a las definidas en el detalle de requerimientos.

2. PROPUESTA DE SOLUCIÓN TÉCNICA

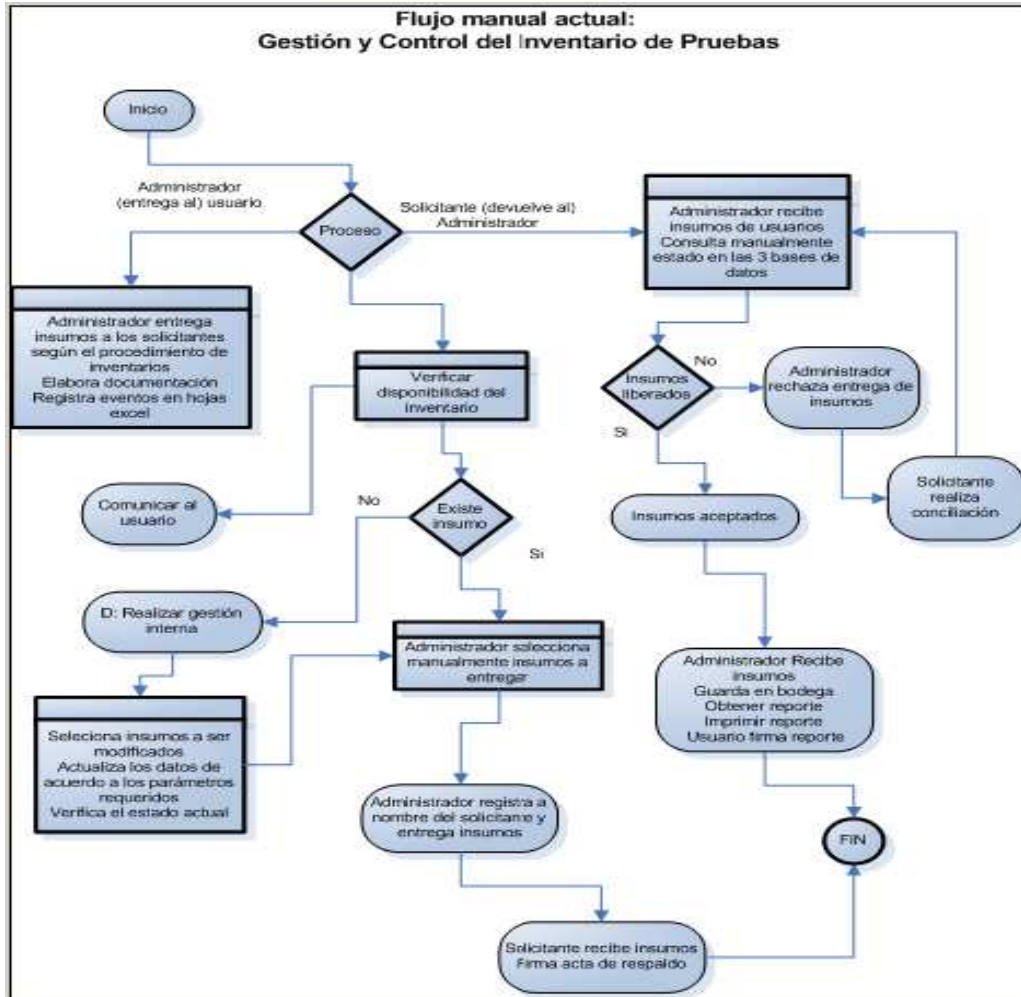
DIAGRAMA DE LA SOLUCION



ARQUITECTURA



FLUJO



ALGORITMO – SEUDOCODIGO

- Desarrollar un aplicativo que permita a los usuarios del dominio OTECEL, autenticarse a través de perfiles.
- El aplicativo dispondrá de formularios ASPX que permitirá realizar acciones enmarcadas al proceso de asignación de recursos en el área de pruebas.
- La funcionalidad del aplicativo esta definida de acuerdo a los casos de uso levantados en las especificaciones de requerimientos.

SCRIPTS RELEVANTES

No Aplica

OBJETOS DE BDD

NOMBRE	MISC3
DESCRIPCION	Base de datos Oracle 8i – Repositorio de Miscelaneos
TNS	<pre> MISC3= (DESCRIPTION = (AADDRESS = (PROTOCOL = TCP)(Host = 10.5.1.215)(Port = 1521)) (CONNECT_DATA = (SERVER = DEDICATED) (SID = MISC3)) </pre>

TIPO	Producción
------	------------

NUEVO / EXISTENTE	N	E	NOMBRE	DESCRIPCION	TIPO	PERMISOS				
						S	I	U	D	E
X			OTC_T_SGIP_PERFILES	Tabla donde se almacena los perfiles de usuario.	Tabla	x	x	x	x	
X			OTC_T_SGIP_USUARIOS	Tabla donde se almacena la información de los usuarios permitidos para acceder al sistema de inventarios.	Tabla	x	x	x	x	
X			OTC_T_SGIP_MENU_I	Tabla donde se almacena la información de los formularios del menú de opciones del sistema de inventarios.	Tabla	x	x	x	x	
X			OTC_T_SGIP_ERRORES	Tabla donde se almacenan los errores producidos en la ejecución del paquete OTC_PKG_SGIP.	Tabla	x	x	x	x	
X			OTC_T_SGIP_SOLICITUDES	Tabla donde se almacena la información de las solicitudes de recursos iniciadas por los usuarios y/o el administrador del inventario.	Tabla	x	x	x	x	
X			OTC_T_SGIP_INSUMOS	Tabla donde se almacena el detalle de los recursos del área de pruebas, simcards y terminales. Utiliza la función OTC_F_EXISTE_INSUMO	Tabla	x	x	x	x	
X			OTC_T_SGIP_REGISTROS_INSUMOS	Tabla donde se almacena las asignaciones y devoluciones de recursos de acuerdo a un número de solicitud de usuario.	Tabla	x	x	x	x	
x			OTC_T_SGIP_REPORTES	Tabla donde se almacena la información de los recursos activos, para que esta tabla se llene debe estar operativo el DBL (OTC_T_DBL_MISC3_SCLPRECU)	Tabla	x	x	x	x	
x			OTC_T_SGIP_AUDIT_INSUMOS	Tabla donde se almacena los eventos [INSERT - DELETE - UPDATE] realizados sobre la tabla [OTC_T_SGIP_INSUMOS]	Tabla	x	x	x		
x			OTC_T_SGIP_TMP	Tabla temporal para alojar las series que serán modificadas por el mantenedor de insumos.	Tabla	x	x	x	x	
x			OTC_T_SGIP_TMP_SIM	Tabla temporal para alojar las series que serán creadas por el formulario de creación de insumos.	Tabla	x	x	x	x	
x			OTC_PKG_SGIP	Ejecución del proceso gestión de inventario de pruebas.	Paquete	x	x	x	x	x
x			OTC_F_EXISTE_INSUMO	Función que valida si un recurso existe.	Función	x	x		x	
x			OTC_T_SGIP_TRG_INSUMOS_DELETE	Trigger que copia datos en la tabla OTC_T_SGIP_AUDIT_INSUMOS	Trigger	x	x	x	x	x
x			OTC_T_SGIP_TRG_INSUMOS_INSERT	Triger que copia datos en la tabla OTC_T_SGIP_AUDIT_INSUMOS	Trigger	x	x	x	x	x
x			OTC_T_SGIP_TRG_INSUMOS_UPDATE	Triger que copia datos en la tabla OTC_T_SGIP_AUDIT_INSUMOS	Trigger	x	x	x	x	x

FUENTES

Aplicativo nuevo, no hay fuentes o versiones anteriores

EQUIVALENCIAS FUNCIONALES - TECNICAS

No Aplica

ALARMAS, LOGS, AUDITORIAS, CONTROL M

No Aplica

DEPURACIÓN DE INFORMACIÓN HISTORICA

No Aplica

SOLUCIONES DESCARTADAS*No Aplica***FUERA DE ALCANCE***No Aplica***PROTOTIPO***Definido en la especificación de requerimientos.***PRERREQUISITOS NECESARIOS PARA EL DESARROLLO.**

NECESIDAD	RESPONSABLE	FECHA COMPROMISO
Esquema de base de datos	Jefe de Area	01-Julio-2011
Servidor de desarrollo	Jefe de Area	01-Julio-2011
Software Visual Net 2005	Jefe de Area	01-Julio-2011

PROTOCOLO BÁSICO DE PRUEBAS

ACTIVIDAD	RESULTADO ESPERADO
Definido en los casos de uso levantados posterior al detalle de requerimientos.	

MANEJO DE RIESGOS*No Aplica***3. PLANIFICACIÓN****CRONOGRAMA**

Task Name	Duration	Start	Finish	Predecessors
<input type="checkbox"/> Construcción - Sistema de Inventarios	42 days?	Wed 23/11/11	Thu 19/01/12	
<input type="checkbox"/> Crear de tablas en base de datos	3 days	Wed 23/11/11	Fri 25/11/11	
Implementar diseño entidad - relación	2 days	Wed 23/11/11	Thu 24/11/11	
Parametrizar tablas previo desarrollo de funciones	1 day	Fri 25/11/11	Fri 25/11/11	3
<input type="checkbox"/> Crear Pagina Maestra	8 days?	Mon 28/11/11	Wed 07/12/11	4
<input type="checkbox"/> Ingreso al aplicativo	5 days?	Thu 08/12/11	Wed 14/12/11	10
<input type="checkbox"/> Modulo Administración	6 days?	Thu 15/12/11	Thu 22/12/11	
<input type="checkbox"/> Modulo Servicios	5 days?	Fri 23/12/11	Thu 29/12/11	
<input type="checkbox"/> Modulo Reportes	6 days?	Fri 30/12/11	Fri 06/01/12	
Pruebas unitarias ciclo completo	1 day	Mon 09/01/12	Mon 09/01/12	136
Pruebas Formales Usuario	2 days	Tue 10/01/12	Wed 11/01/12	137
Realizar ajustes y depuraciones	1 day	Thu 12/01/12	Thu 12/01/12	138
Afinar documentación final y Elaborar MAU	5 days	Fri 13/01/12	Thu 19/01/12	139

4. CONSTRUCCIÓN**AMBIENTE DE DESARROLLO**

*Servidor: Estación de escritorio
 Ip: 10.112.199.101
 Usuario: ROZUMBA
 Password: red*

BASES DE DATOS

NOMBRE	SCLPREECU
DESCRIPCION	Base de datos test
TNS	<pre>SCLPREECU = (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = 10.80.1.109)(PORT = 1521))) (CONNECT_DATA = (SID = SCLPREECU))))</pre>
TIPO	Test

DIAGRAMA DE CLASES

Definido en el diseño del aplicativo. Anexo

PROCESOS DEPENDIENTES

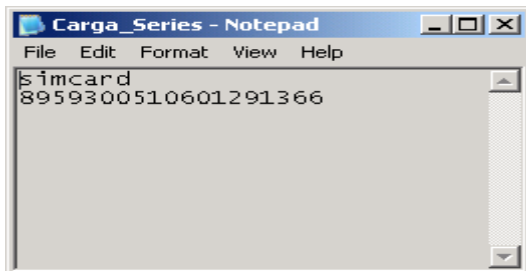
No Aplica

DESCRIPCIÓN DE ALARMAS

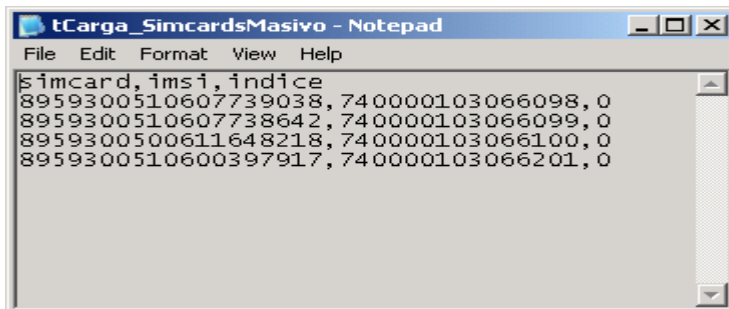
No Aplica

ARCHIVOS ENTRADA – SALIDA

Archivo para realizar el proceso de asignación y entrega de recursos



Archivo para realizar el proceso de creación de nuevos recursos (Simcards)



Archivo para realizar el proceso de creación de nuevos recursos (Terminales)

```

tCarga_TerminalesMasivo - Notepad
File Edit Format View Help
imei, modelo, codigo
740000103066098, E71, 20
740000103066099, Samsung, 10

```

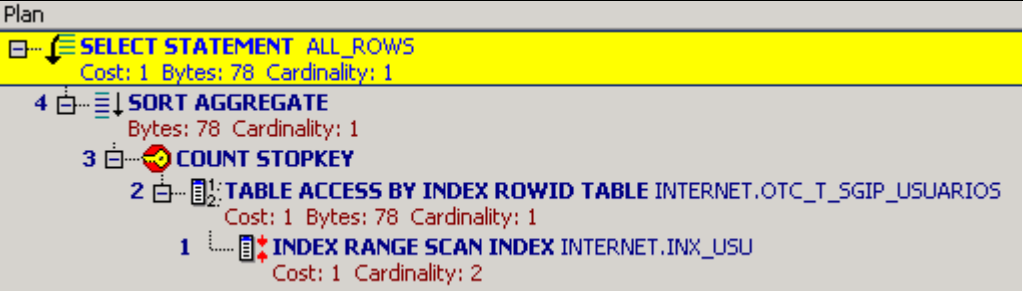
ARCHIVO DE CONFIGURACIÓN

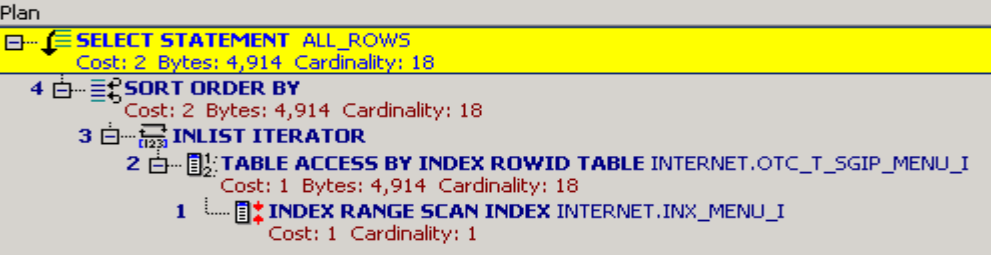
PARAMETRO	DESCRIPCION
key="Carga_Archivos" value ="D:\ServiciosWeb\sistema_inventarios\tmp\"	Ruta temporal donde se guarda el archivo cargado por el usuario desde el GUI del aplicativo.
sessionState mode="InProc" cookieless="false" timeout="25"	Parámetro para definir el tiempo de cerrar la sesión automáticamente.
key=" SP_MENUPERFIL"	Invocación al procedimiento: OTC_T_SP_MENUPERFIL
key="SP_CONSULTA_USUARIO"	OTC_T_SP_CONSULTAR_USUARIO
key="SP_PERFIL"	OTC_T_SP_OBTENER_PERFIL
key="SP_TIPO_RECURSOS"	OTC_T_SP_OBTENER_TIPO_RECURSOS
key="SP_SOLICITUDES"	OTC_T_SP_CONSULTAR_SOLICITUDES
key="SP_SOLICITUDES_PERFIL"	OTC_T_SP_SOLICITUDES_PERFIL
key="SP_SOLICITUDES_FECHA"	OTC_T_SP_SOLICITUDES_FECHA
key="SP_SOLICITUDES_COD"	OTC_T_SP_SOLICITUDES_COD
key="SP_ASIGNA_INSUMOS"	OTC_T_SP_ASIGNA_INSUMOS
key="SP_USUARIOS"	OTC_T_SP_USUARIOS_ACTIVOS
key="SP_CREA_INSUMOS"	OTC_T_SP_CREAR_INSUMOS
key="SP_SOLICITUDES SR"	OTC_T_SP_CONSULTARSOLICITUDES
key="SP_REPORTE_ASIGNADO"	OTC_T_SP_REPORTE_ASIGNADO
key="SP_REPORTE_AUDITORIA"	OTC_T_SP_REPORTE_AUDITORIA
key="SP_REPORTE_DEVOLUCIONES"	OTC_T_SP_REPORTE_DEVOLUCIONES
key="SP_REPORTE_INVENTARIO"	OTC_T_SP_REPORTE_INVENTARIO
key="SP_REPORTE_USUARIO"	OTC_T_SP_REPORTE_USUARIO
key="SP_REPORTE_SOLICITUD"	OTC_T_SP_REPORTE_SOLICITUD
key="SP_REPORTE_TODO"	OTC_T_SP_REPORTE_TODO
key="CONSULTA_USUARIO_COD"	OTC_T_SP_CONSULTAR_USUARIO_COD
key="INSUMOS_ACTIVOS"	OTC_T_SP_REPORTE_ACTIVOS

CÓDIGO DURO

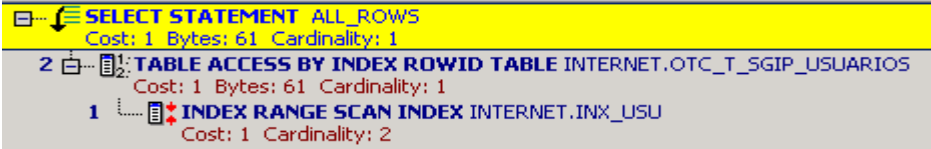
No Aplica

5. PRUEBAS UNITARIAS**PLANES DE EJECUCIÓN**

Script: OTC_SP_COD_PERFIL
<pre> /* Formatted on 2012/06/01 16:27 (Formatter Plus v4.8.8) */ SELECT COUNT(USU_COD_PERF) --INTO AUX_COD_PERF FROM OTC_T_SGIP_USUARIOS WHERE USU_RED = UPPER('ROZUMBA') AND USU_PASS_NUE = 0 AND USU_APLICATIVO='INVENTARIOS' AND USU_ESTADO='ACTIVO' AND ROWNUM =1; </pre>
Plan de Ejecución
<p>Plan</p> 

Script: OTC_T_SP_MENUPERFIL
<pre> /* Formatted on 2012/06/01 16:28 (Formatter Plus v4.8.8) */ SELECT men_idmod, men_nommod, men_idmen, men_nomen, men_idapl, men_nomapl, men_urlapl FROM otc_t_sgip_menu_i WHERE men_sec >= 1 AND men_perfil IN (2, 3) ORDER BY men_idapl; </pre>
Plan de Ejecución
<p>Plan</p> 

Script: OTC_SP_MODIFICA_CLAVE
<pre> /* Formatted on 2012/06/01 16:28 (Formatter Plus v4.8.8) */ SELECT usu_cod_perf --INTO AUX_COD_PERF FROM otc_t_sgip_usuarios WHERE usu_red = 'ROZUMBA' AND usu_pass_ant = 0 AND usu_cod_perf IN (2, 3) AND usu_estado = 'ACTIVO'; </pre>
Plan de Ejecución

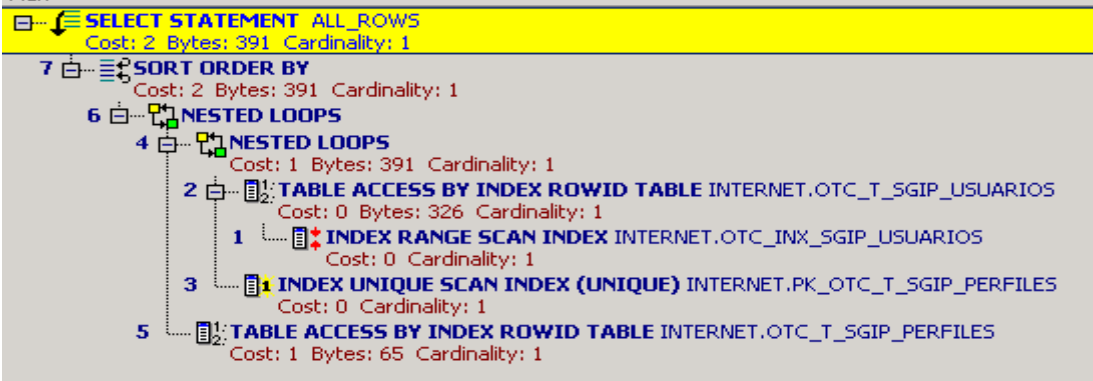
Plan
 <pre> SELECT STATEMENT ALL_ROWS Cost: 1 Bytes: 61 Cardinality: 1 2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_USUARIOS Cost: 1 Bytes: 61 Cardinality: 1 1 INDEX RANGE SCAN INDEX INTERNET.INX_USU Cost: 1 Cardinality: 2 </pre>

Script: OTC_T_SP_CONSULTAR_USUARIO

```

/* Formatted on 2012/06/01 19:03 (Formatter Plus v4.8.8) */
SELECT  u.usu_cod AS codigo, UPPER (p.perf_desc) AS perfil,
        u.usu_nombre AS nombre, u.usu_apellido AS apellido,
        u.usu_correo AS correo, u.usu_red AS red,
        u.usu_proveedor AS proveedor, u.usu_estado AS estado,
        usu_movil AS movil, usu_fijo AS fijo, usu_ext AS ext
FROM    otc_t_sgip_usuarios u, otc_t_sgip_perfiles p
WHERE   p.perf_cod = u.usu_cod_perf
        AND u.usu_cod IN (SELECT usu_cod
                          FROM otc_t_sgip_usuarios
                          WHERE usu_apellido LIKE ' ' || UPPER ('RO') || ' ')
ORDER BY 1;
  
```

Plan de Ejecución

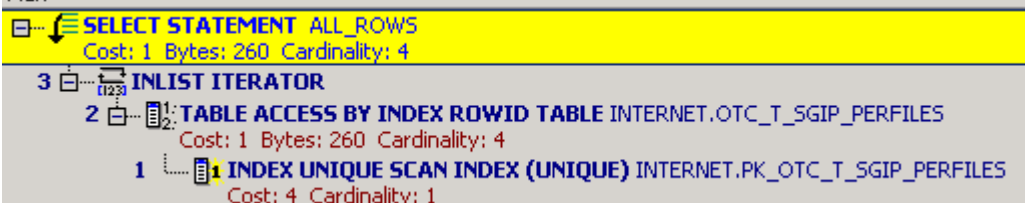
Plan
 <pre> SELECT STATEMENT ALL_ROWS Cost: 2 Bytes: 391 Cardinality: 1 7 SORT ORDER BY Cost: 2 Bytes: 391 Cardinality: 1 6 NESTED LOOPS Cost: 1 Bytes: 391 Cardinality: 1 4 NESTED LOOPS Cost: 0 Bytes: 326 Cardinality: 1 2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_USUARIOS Cost: 0 Bytes: 326 Cardinality: 1 1 INDEX RANGE SCAN INDEX INTERNET.OTC_INX_SGIP_USUARIOS Cost: 0 Cardinality: 1 3 INDEX UNIQUE SCAN INDEX (UNIQUE) INTERNET.PK_OTC_T_SGIP_PERFILES Cost: 0 Cardinality: 1 5 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_PERFILES Cost: 1 Bytes: 65 Cardinality: 1 </pre>

Script: OTC_T_SP_OBTENER_PERFIL

```

/* Formatted on 2012/06/01 16:35 (Formatter Plus v4.8.8) */
SELECT  perf_cod AS codigo, perf_desc AS descripcion
FROM    otc_t_sgip_perfiles
WHERE   perf_cod IN (1, 2, 3, 4)
ORDER BY perf_cod;
  
```

Plan de Ejecución

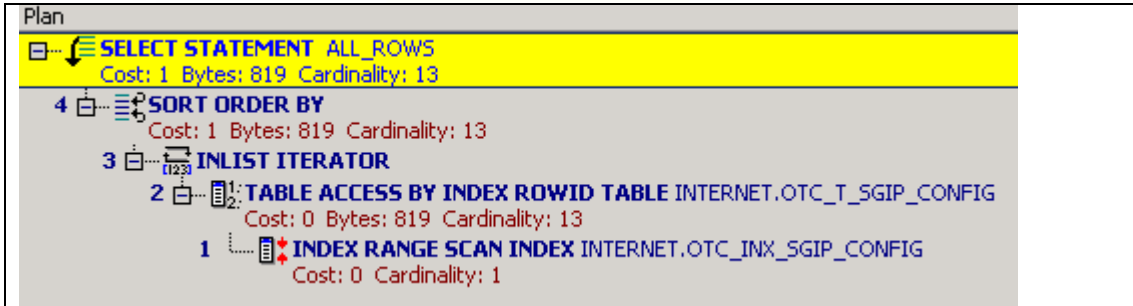
Plan
 <pre> SELECT STATEMENT ALL_ROWS Cost: 1 Bytes: 260 Cardinality: 4 3 INLIST ITERATOR Cost: 1 Bytes: 260 Cardinality: 4 2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_PERFILES Cost: 1 Bytes: 260 Cardinality: 4 1 INDEX UNIQUE SCAN INDEX (UNIQUE) INTERNET.PK_OTC_T_SGIP_PERFILES Cost: 4 Cardinality: 1 </pre>

Script: OTC_T_SP_OBTENER_TIPO_RECURSOS

```

/* Formatted on 2012/06/01 16:37 (Formatter Plus v4.8.8) */
SELECT  sec, detalle
FROM    otc_t_sgip_config
WHERE   valor IN (100, 101)
ORDER BY detalle;
  
```

Plan de Ejecución

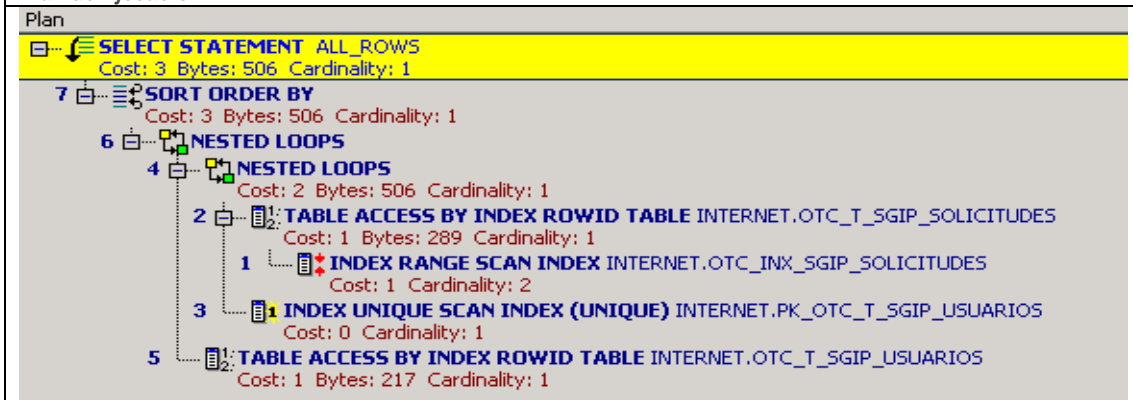


Script: OTC_T_SP_CONSULTAR_SOLICITUDES

```

/* Formatted on 2012/06/01 16:39 (Formatter Plus v4.8.8) */
SELECT
  s.cod_solicitud AS solicitud,
  TO_CHAR (s.fecha_creacion, 'MM/DD/YYYY HH24:MI:SS') AS pedido,
  u.usu_nombre || '-' || u.usu_apellido AS usuario, s.recurso,
  s.cantidad, s.justificacion, s.req_ppm,
  TO_CHAR (s.fecha_devolucion, 'MM/DD/YYYY HH24:MI:SS') AS
devolucion,
  s.ambiente, s.fisica
FROM otc_t_sgip_solicitudes s, otc_t_sgip_usuarios u
WHERE s.usu_cod = u.usu_cod
      AND s.cod_solicitud > 1
      AND s.fecha_creacion > SYSDATE - 30
      AND s.flujo = 'NUEVO'
ORDER BY s.cod_solicitud;
  
```

Plan de Ejecución

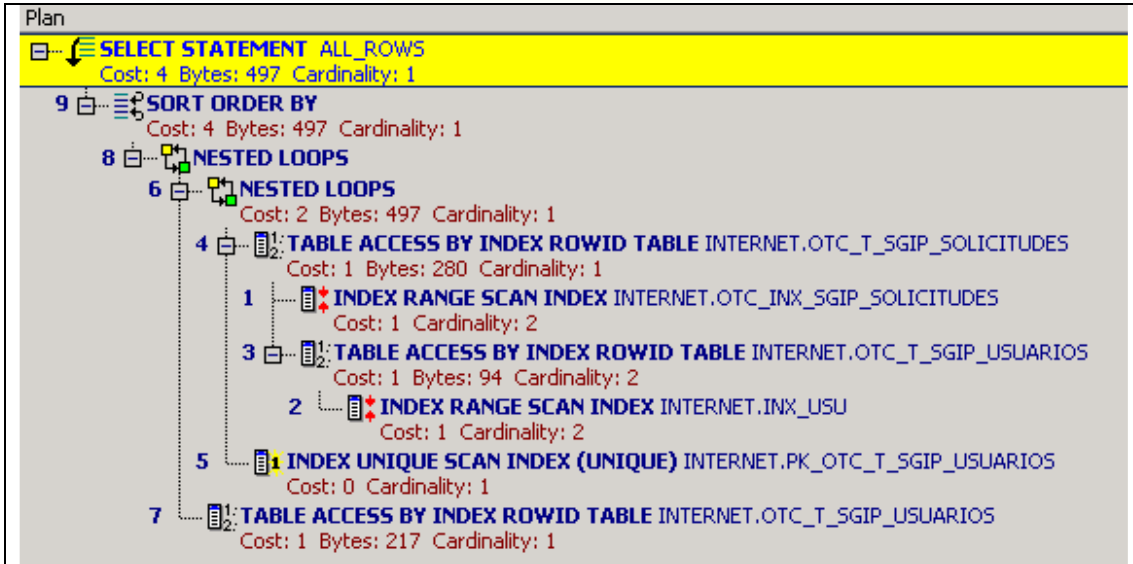


Script: OTC_T_SP_SOLICITUDES_PERFIL

```

/* Formatted on 2012/06/01 16:42 (Formatter Plus v4.8.8) */
SELECT
  s.cod_solicitud AS solicitud,
  TO_CHAR (s.fecha_creacion, 'MM/DD/YYYY HH24:MI:SS') AS pedido,
  u.usu_nombre || '-' || u.usu_apellido AS usuario, s.recurso,
  s.cantidad, s.justificacion, s.req_ppm,
  TO_CHAR (s.fecha_devolucion, 'MM/DD/YYYY HH24:MI:SS') AS
devolucion,
  s.ambiente, s.fisica
FROM otc_t_sgip_solicitudes s, otc_t_sgip_usuarios u
WHERE s.usu_cod = u.usu_cod
      AND s.cod_solicitud > 1
      AND s.fecha_creacion > SYSDATE - 30
      AND s.usu_cod =
        (SELECT usu_cod
         FROM otc_t_sgip_usuarios
         WHERE usu_red = 'ROZUMBA'
              AND usu_estado = 'ACTIVO'
              AND usu_cod_perf IN (2, 3))
ORDER BY s.cod_solicitud;
  
```

Plan de Ejecución

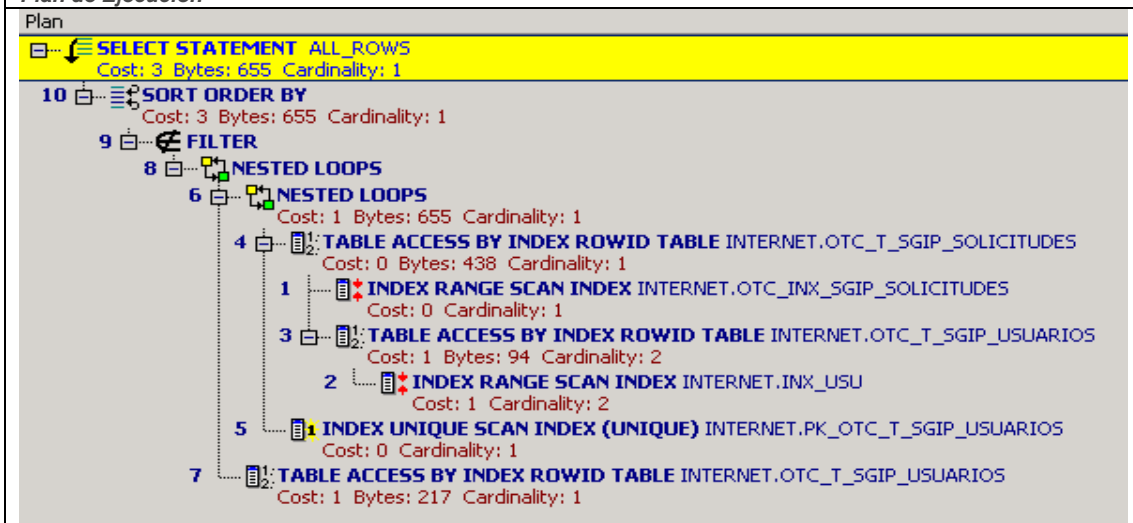


Script: OTC T_SP_SOLICITUDES_FECHA

```

/* Formatted on 2012/06/01 16:44 (Formatter Plus v4.8.8) */
SELECT
  s.cod_solicitud AS solicitud,
  TO_CHAR (s.fecha_creacion, 'MM/DD/YYYY HH24:MI:SS') AS pedido,
  u.usu_nombre || '-' || u.usu_apellido AS usuario,
  s.fecha_asignacion AS entrega, s.recurso, s.cantidad,
  s.justificacion, s.req_ppm,
  TO_CHAR (s.fecha_devolucion, 'MM/DD/YYYY HH24:MI:SS') AS
devolucion,
  s.ambiente, s.fisica, s.flujo AS estado, s.dias_prorroga, s.notas
FROM otc_t_sgip_solicitudes s, otc_t_sgip_usuarios u
WHERE s.usu_cod = u.usu_cod
  AND s.cod_solicitud > 1
  AND s.fecha_creacion BETWEEN SYSDATE AND SYSDATE
  AND s.usu_cod =
    (SELECT usu_cod
     FROM otc_t_sgip_usuarios
     WHERE usu_red = 'ROZUMBA'
       AND usu_estado = 'ACTIVO'
       AND usu_cod_perf IN (2, 3))
ORDER BY s.cod_solicitud;
  
```

Plan de Ejecución

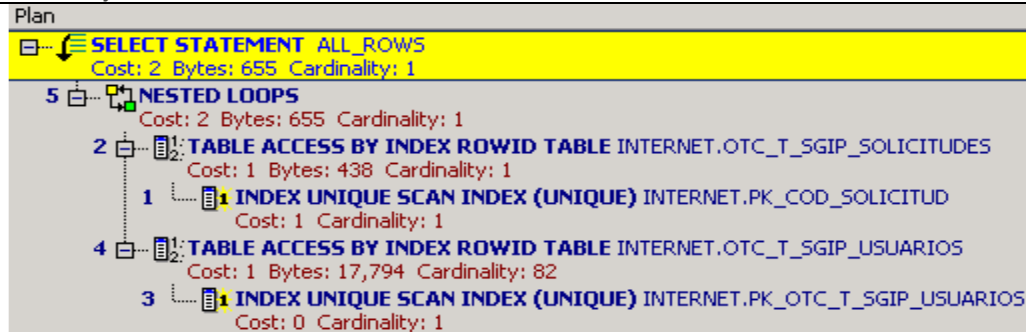


Script: OTC_T_SP_SOLICITUDES_COD

```

/* Formatted on 2012/06/01 16:46 (Formatter Plus v4.8.8) */
SELECT  s.cod_solicitud AS solicitud,
        TO_CHAR (s.fecha_creacion, 'MM/DD/YYYY HH24:MI:SS') AS pedido,
        u.usu_nombre || '-' || u.usu_apellido AS usuario,
        s.fecha_asignacion AS entrega, s.recurso, s.cantidad,
        s.justificacion, s.req_ppm,
        TO_CHAR (s.fecha_devolucion, 'MM/DD/YYYY HH24:MI:SS') AS
devolucion,
        s.ambiente, s.fisica, s.flujo AS estado, s.dias_prorroga, s.notas
FROM    otc_t_sqip_solicitudes s, otc_t_sqip_usuarios u
WHERE   s.usu_cod = u.usu_cod AND s.cod_solicitud = 2
ORDER  BY s.cod_solicitud;

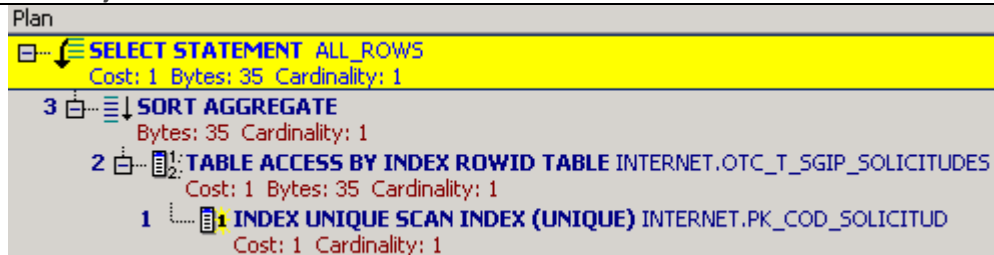
```

Plan de Ejecución**Script: OTC_T_SP_PRO_SO_PRORROGA**

```

/* Formatted on 2012/06/01 16:47 (Formatter Plus v4.8.8) */
SELECT COUNT (cod_solicitud)
  --INTO secuencial
FROM    otc_t_sqip_solicitudes
WHERE   cod_solicitud = 5
        AND dias_prorroga = 0
        AND flujo NOT IN ('NUEVO', 'NEGADO')
        AND cod_solicitud > 1;

```

Plan de Ejecución**Script: OTC_T_SP_PRO_SO_APERTURA**

```

/* Formatted on 2012/06/01 16:49 (Formatter Plus v4.8.8) */
SELECT COUNT (cod_solicitud)
  --INTO secuencial
FROM    otc_t_sqip_solicitudes
WHERE   cod_solicitud = 5
        AND flujo NOT IN ('NUEVO')
        AND cod_solicitud > 1;

```

Plan de Ejecución

Plan
<pre> SELECT STATEMENT ALL_ROWS Cost: 1 Bytes: 22 Cardinality: 1 3 SORT AGGREGATE Bytes: 22 Cardinality: 1 2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_SOLICITUDES Cost: 1 Bytes: 22 Cardinality: 1 1 INDEX UNIQUE SCAN INDEX (UNIQUE) INTERNET.PK_COD_SOLICITUD Cost: 1 Cardinality: 1 </pre>

Script: OTC_T_SP_ASIGNA_INSUMOS <pre> /* Formatted on 2012/06/01 16:51 (Formatter Plus v4.8.8) */ SELECT DISTINCT (num_serie) FROM otc_t_sgip_tmp WHERE num_serie IN (SELECT num_serie FROM otc_t_sgip_insumos); </pre>		
Plan de Ejecución <table border="1"> <tr> <td>Plan</td> </tr> <tr> <td> <pre> SELECT STATEMENT ALL_ROWS Cost: 2 Bytes: 28 Cardinality: 1 4 SORT UNIQUE NOSORT Cost: 2 Bytes: 28 Cardinality: 1 3 NESTED LOOPS Cost: 1 Bytes: 28 Cardinality: 1 1 INDEX FULL SCAN INDEX INTERNET.OTC_INX_OTC_T_SGIP_TMP Cost: 1 Bytes: 14 Cardinality: 1 2 INDEX UNIQUE SCAN INDEX (UNIQUE) INTERNET.PK_NUMSERIE Cost: 0 Bytes: 14 Cardinality: 1 </pre> </td> </tr> </table>	Plan	<pre> SELECT STATEMENT ALL_ROWS Cost: 2 Bytes: 28 Cardinality: 1 4 SORT UNIQUE NOSORT Cost: 2 Bytes: 28 Cardinality: 1 3 NESTED LOOPS Cost: 1 Bytes: 28 Cardinality: 1 1 INDEX FULL SCAN INDEX INTERNET.OTC_INX_OTC_T_SGIP_TMP Cost: 1 Bytes: 14 Cardinality: 1 2 INDEX UNIQUE SCAN INDEX (UNIQUE) INTERNET.PK_NUMSERIE Cost: 0 Bytes: 14 Cardinality: 1 </pre>
Plan		
<pre> SELECT STATEMENT ALL_ROWS Cost: 2 Bytes: 28 Cardinality: 1 4 SORT UNIQUE NOSORT Cost: 2 Bytes: 28 Cardinality: 1 3 NESTED LOOPS Cost: 1 Bytes: 28 Cardinality: 1 1 INDEX FULL SCAN INDEX INTERNET.OTC_INX_OTC_T_SGIP_TMP Cost: 1 Bytes: 14 Cardinality: 1 2 INDEX UNIQUE SCAN INDEX (UNIQUE) INTERNET.PK_NUMSERIE Cost: 0 Bytes: 14 Cardinality: 1 </pre>		

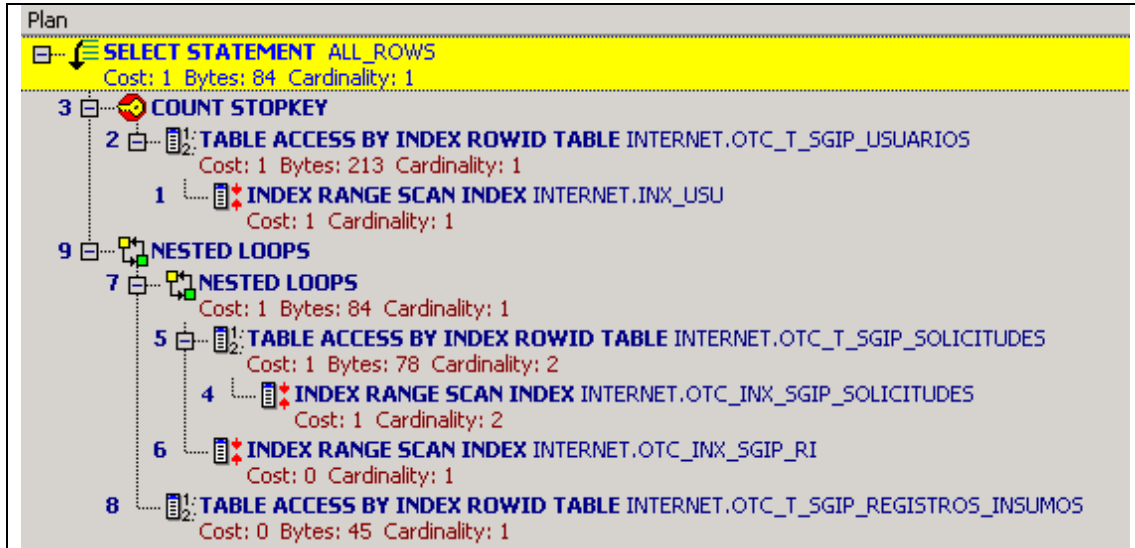
Script: OTC_T_SP_USUARIOS_ACTIVOS <pre> /* Formatted on 2012/06/01 16:53 (Formatter Plus v4.8.8) */ SELECT DISTINCT (usu_nombre ' ' usu_apellido) AS usu_nombre, usu_red FROM otc_t_sgip_usuarios WHERE usu_cod_perf IN (1, 2, 3, 4) AND usu_estado = 'ACTIVO' ORDER BY usu_nombre; </pre>		
Plan de Ejecución <table border="1"> <tr> <td>Plan</td> </tr> <tr> <td> <pre> SELECT STATEMENT ALL_ROWS Cost: 3 Bytes: 8,330 Cardinality: 35 5 SORT ORDER BY Cost: 3 Bytes: 8,330 Cardinality: 35 4 HASH UNIQUE Cost: 2 Bytes: 8,330 Cardinality: 35 3 INLIST ITERATOR 2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_USUARIOS Cost: 1 Bytes: 8,330 Cardinality: 35 1 INDEX RANGE SCAN INDEX INTERNET.INX_USU_PERF Cost: 1 Cardinality: 1 </pre> </td> </tr> </table>	Plan	<pre> SELECT STATEMENT ALL_ROWS Cost: 3 Bytes: 8,330 Cardinality: 35 5 SORT ORDER BY Cost: 3 Bytes: 8,330 Cardinality: 35 4 HASH UNIQUE Cost: 2 Bytes: 8,330 Cardinality: 35 3 INLIST ITERATOR 2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_USUARIOS Cost: 1 Bytes: 8,330 Cardinality: 35 1 INDEX RANGE SCAN INDEX INTERNET.INX_USU_PERF Cost: 1 Cardinality: 1 </pre>
Plan		
<pre> SELECT STATEMENT ALL_ROWS Cost: 3 Bytes: 8,330 Cardinality: 35 5 SORT ORDER BY Cost: 3 Bytes: 8,330 Cardinality: 35 4 HASH UNIQUE Cost: 2 Bytes: 8,330 Cardinality: 35 3 INLIST ITERATOR 2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_USUARIOS Cost: 1 Bytes: 8,330 Cardinality: 35 1 INDEX RANGE SCAN INDEX INTERNET.INX_USU_PERF Cost: 1 Cardinality: 1 </pre>		

Script: OTC_T_SP_CREAR_INSUMOS <pre> /* Formatted on 2012/06/01 16:55 (Formatter Plus v4.8.8) */ SELECT DISTINCT (num_serie), num_imsi, aux FROM otc_t_sgip_tmp_sim WHERE num_serie NOT IN (SELECT num_serie FROM otc_t_sgip_insumos) </pre>

AND fecha > SYSDATE - 1 ;
Plan de Ejecución
Plan
SELECT STATEMENT ALL_ROWS Cost: 2 Bytes: 53 Cardinality: 1
5 HASH UNIQUE Cost: 2 Bytes: 53 Cardinality: 1
4 NESTED LOOPS ANTI Cost: 1 Bytes: 53 Cardinality: 1
2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_TMP_SIM Cost: 1 Bytes: 39 Cardinality: 1
1 INDEX RANGE SCAN INDEX INTERNET.INX_TMP_SIM Cost: 1 Cardinality: 1
3 INDEX UNIQUE SCAN INDEX (UNIQUE) INTERNET.PK_NUMSERIE Cost: 0 Bytes: 14 Cardinality: 1

Script: OTC_T_SP_REPORTE_AUDITORIA
<pre> /* Formatted on 2012/06/01 17:22 (Formatter Plus v4.8.8) */ SELECT DECODE (ai.tipo_insumo, 'T', 'TERMINAL', 'S', 'SIMCARD') AS insumo, ai.num_serie AS serie, ai.id_caja AS caja, ai.num_imsi AS central, ai.ambiente, TO_CHAR (ai.fecha_creacion, 'MM/DD/YYYY HH24:MI:SS') AS fecha_creacion, ai.situacion AS estado, ai.modelo, ai.tecnologia, TO_CHAR (ai.fecha_revision, 'MM/DD/YYYY HH24:MI:SS') AS fecha_revision, ai.responsible_revision AS responsable, ai.notas, TO_CHAR (ai.fecha_evento, 'MM/DD/YYYY HH24:MI:SS') AS fecha_evento, ai.tipo_evento AS evento FROM otc_t_sgip_audit_insumos ai WHERE ai.num_serie = '89' ORDER BY ai.fecha_evento DESC; </pre>
Plan de Ejecución
Plan
SELECT STATEMENT ALL_ROWS Cost: 2 Bytes: 257 Cardinality: 1
3 SORT ORDER BY Cost: 2 Bytes: 257 Cardinality: 1
2 TABLE ACCESS BY INDEX ROWID TABLE INTERNET.OTC_T_SGIP_AUDIT_INSUMOS Cost: 1 Bytes: 257 Cardinality: 1
1 INDEX RANGE SCAN INDEX INTERNET.INX_AUDIT_INSUMOS Cost: 1 Cardinality: 1

Script: OTC_T_SP_REPORTE_DEVOLUCIONES
<pre> /* Formatted on 2012/06/01 17:30 (Formatter Plus v4.8.8) */ SELECT so.fisica AS digital, ri.num_serie AS serie, ri.cod_solicitud AS solicitud, (SELECT usu_nombre ' ' usu_apellido FROM otc_t_sgip_usuarios WHERE usu_red = ri.devolucion AND ROWNUM = 1) AS usuario, TO_CHAR (ri.fecha_devolucion_real, 'MM/DD/YYYY HH24:MI:SS') AS fecha_devolucion, so.req_ppm FROM otc_t_sgip_registros_insumos ri, otc_t_sgip_solicitudes so WHERE so.cod_solicitud = ri.cod_solicitud AND ri.fecha_devolucion_real IS NOT NULL AND ri.devolucion = 'ROZUMBA' AND ri.fecha_devolucion_real >= (SYSDATE) AND so.fecha_creacion < SYSDATE; </pre>
Plan de Ejecución

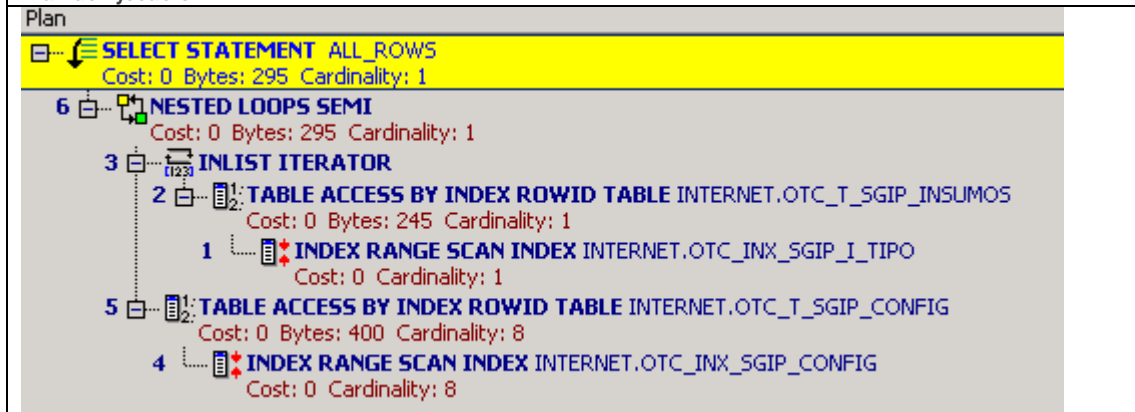


Script: OTC T SP REPORTE INVENTARIO

```

/* Formatted on 2012/06/01 17:46 (Formatter Plus v4.8.8) */
SELECT DECODE (tipo_insumo, 'S', 'SIMCARD', 'T', 'TERMINAL') AS tipo_insumo,
       num_serie, id_caja, num_imsi, ambiente,
       TO_CHAR (fecha_creacion, 'MM/DD/YYYY HH24:MI:SS') AS fecha_creacion,
       situacion, modelo, tecnologia,
       TO_CHAR (fecha_revision, 'MM/DD/YYYY HH24:MI:SS') AS fecha_revision,
       responsable_revision, notas
FROM   otc_t_sgip_insumos
WHERE  tipo_insumo IN ('S', 'T') AND situacion IN (SELECT detalle
                                                    FROM otc_t_sgip_config
                                                    WHERE valor = 101);
  
```

Plan de Ejecución



Script: OTC T SP REPORTE USUARIO

```

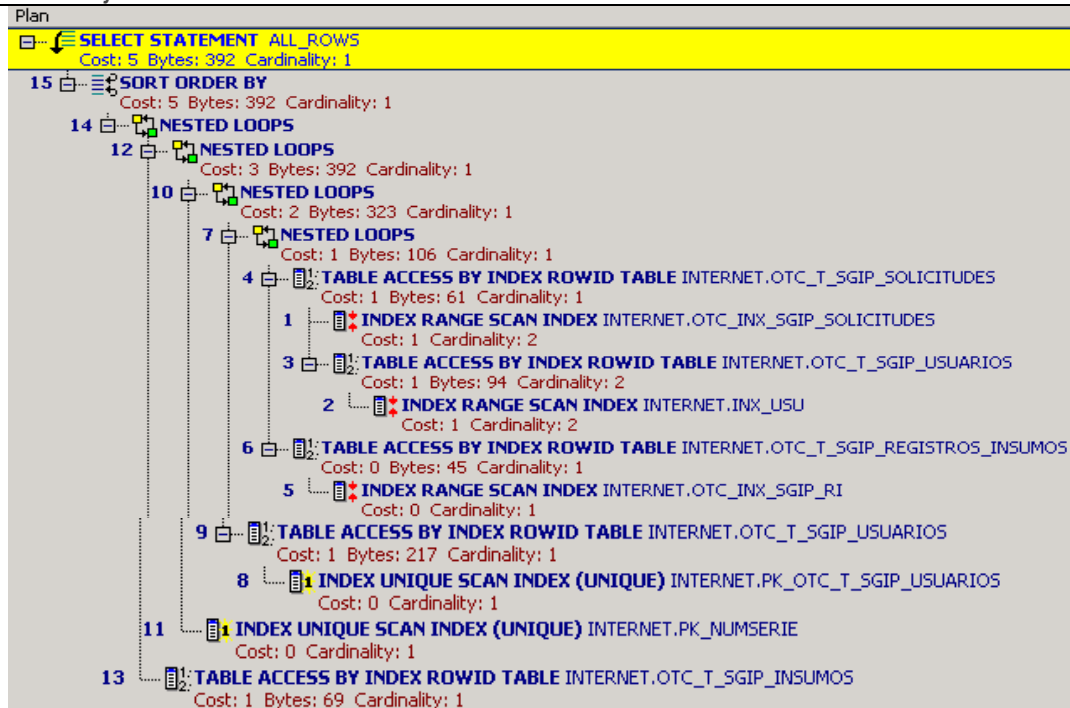
/* Formatted on 2012/06/01 19:38 (Formatter Plus v4.8.8) */
SELECT DECODE (i.tipo_insumo,
              'S', 'SIMCARD',
              'T', 'TERMINAL'
              ) AS tipo_insumo,
       i.num_serie, s.fisica, NVL (SUBSTR (i.num_imsi, 7, 1), 0) AS
central,
       s.cod_solicitud AS solicitud, i.ambiente,
       (u.usu_nombre || ' ' || u.usu_apellido) AS usuario, i.modelo,
       i.tecnologia,
  
```

```

TO_CHAR (ri.fecha_asignacion, 'MM/DD/YYYY HH24:MI:SS') AS
asignacion,
TO_CHAR (s.fecha_devolucion, 'MM/DD/YYYY HH24:MI:SS') AS
devolucion,
s.req_ppm
FROM otc_t_sgip_insumos i,
otc_t_sgip_solicitudes s,
otc_t_sgip_registros_insumos ri,
otc_t_sgip_usuarios u
WHERE i.num_serie = ri.num_serie
AND ri.cod_solicitud = s.cod_solicitud
AND u.usu_cod = s.usu_cod
AND ri.devolucion IS NULL
AND s.usu_cod =
(SELECT usu_cod
FROM otc_t_sgip_usuarios
WHERE usu_red = 'ROZUMBA'
AND usu_estado = 'ACTIVO'
AND usu_cod_perf IN (2, 3))
AND s.fecha_creacion < SYSDATE
ORDER BY s.fecha_devolucion ASC;

```

Plan de Ejecución

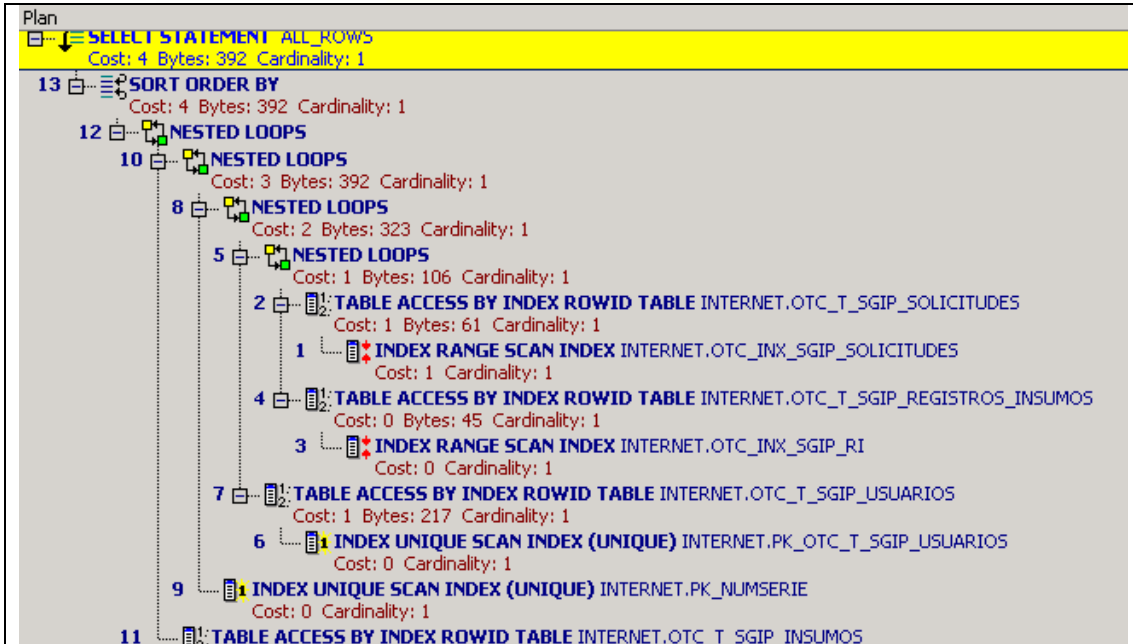


Script: OTC_T_SP_REPORTS_SOLICITUD

```

/* Formatted on 2012/06/01 19:40 (Formatter Plus v4.8.8) */
SELECT DECODE (i.tipo_insumo,
               'S', 'SIMCARD',
               'T', 'TERMINAL'
              ) AS tipo_insumo,
i.num_serie, s.fisica, NVL (SUBSTR (i.num_imsi, 7, 1), 0) AS
central,
s.cod_solicitud AS solicitud, i.ambiente,
(u.usu_nombre || ' ' || u.usu_apellido) AS usuario, i.modelo,
i.tecnologia,
TO_CHAR (ri.fecha_asignacion, 'MM/DD/YYYY HH24:MI:SS') AS
asignacion,
TO_CHAR (s.fecha_devolucion, 'MM/DD/YYYY HH24:MI:SS') AS
devolucion,
s.req_ppm
FROM otc_t_sgip_insumos i,

```

Script: OTC_T_SP_INSUMOS_ACTIVOS

```

/* Formatted on 2012/06/01 20:11 (Formatter Plus v4.8.8) */
SELECT a.num_serie, a.num_abonado, a.num_celular, a.cod_cliente,
a.cod_cuenta,
  a.cod_situacion, b.cod_plantarif, b.des_plantarif, a.cod_ciclo,
  a.fec_alta, a.num_imei
FROM ga_abocel@otc_t_dbl_misc3_sclpreecu a,
  ta_plantarif@otc_t_dbl_misc3_sclpreecu b
WHERE a.cod_plantarif = b.cod_plantarif
  AND a.cod_situacion = 'AAA'
  AND a.num_serie IN (SELECT num_serie
                      FROM otc_t_sgip_insumos
                      WHERE tipo_insumo = 'S' AND ambiente = 'PRODUCCION')
UNION
SELECT a.num_serie, a.num_abonado, a.num_celular, a.cod_cliente,
a.cod_cuenta,
  a.cod_situacion, b.cod_plantarif, b.des_plantarif, a.cod_ciclo,
  a.fec_alta, a.num_imei
FROM ga_aboamist@otc_t_dbl_misc3_sclpreecu a,
  ta_plantarif@otc_t_dbl_misc3_sclpreecu b
WHERE a.cod_plantarif = b.cod_plantarif
  AND a.cod_situacion = 'AAA'
  AND a.num_serie IN (SELECT num_serie
                      FROM otc_t_sgip_insumos
                      WHERE tipo_insumo = 'S' AND ambiente =
'PRODUCCION');

```

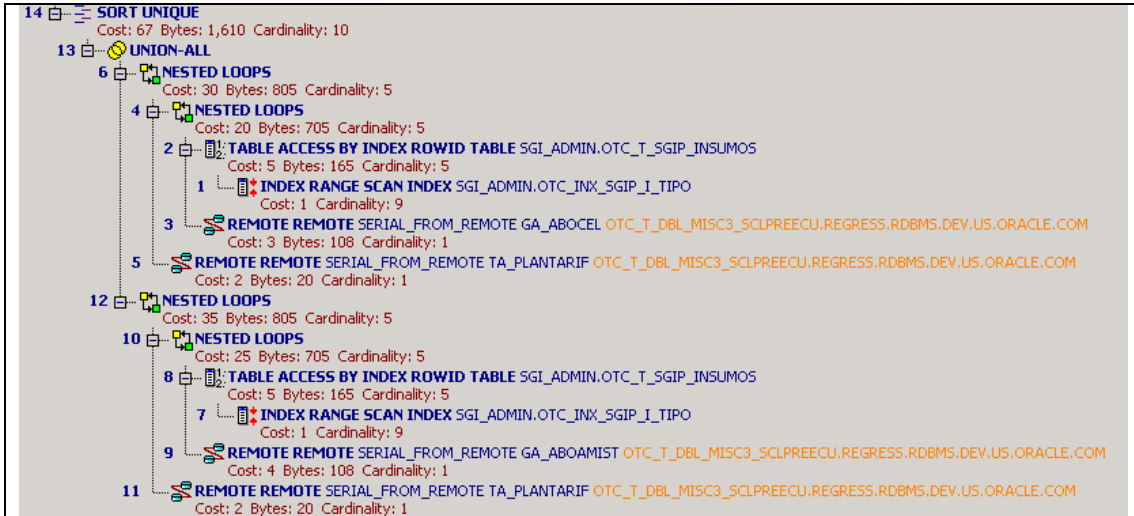
Plan de Ejecución

Plan

```

SELECT STATEMENT ALL_ROWS
  Cost: 67 Bytes: 1,610 Cardinality: 10

```

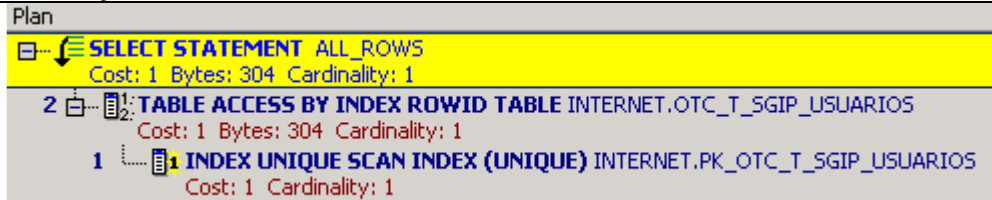


Script: OTC_T_SP_CONSULTAR_USUARIO_COD

```

/* Formatted on 2012/06/01 17:51 (Formatter Plus v4.8.8) */
SELECT u.usu_cod AS codigo, u.usu_nombre AS nombre,
       u.usu_apellido AS apellido, u.usu_correo AS correo,
       u.usu_proveedor AS proveedor, usu_estado AS estado, usu_movil AS
movil,
       usu_fijo AS fijo, usu_ext AS ext
FROM otc_t_sgip_usuarios u
WHERE u.usu_cod = 2;
  
```

Plan de Ejecución

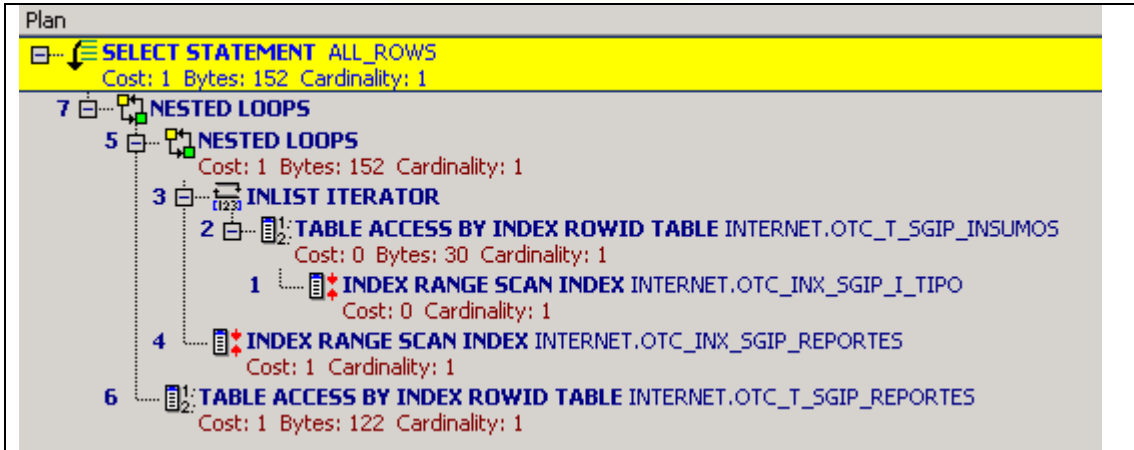


Script: OTC_T_SP_REPORTES_ACTIVOS

```

/* Formatted on 2012/06/01 17:52 (Formatter Plus v4.8.8) */
SELECT i.id_caja AS caja, r.num_serie AS serie, r.num_abonado AS abonado,
       r.num_celular AS celular, r.cod_cliente AS cliente,
       r.cod_cuenta AS cuenta, r.cod_plantarif AS codigo,
       r.des_plantarif AS PLAN, r.cod_ciclo AS ciclo, r.fec_alta AS alta,
       r.num_imei AS imei
FROM otc_t_sgip_reportes r, otc_t_sgip_insumos i
WHERE r.num_serie = i.num_serie
      AND i.tipo_insumo IN ('S', 'T')
      AND r.num_abonado > 0;
  
```

Plan de Ejecución

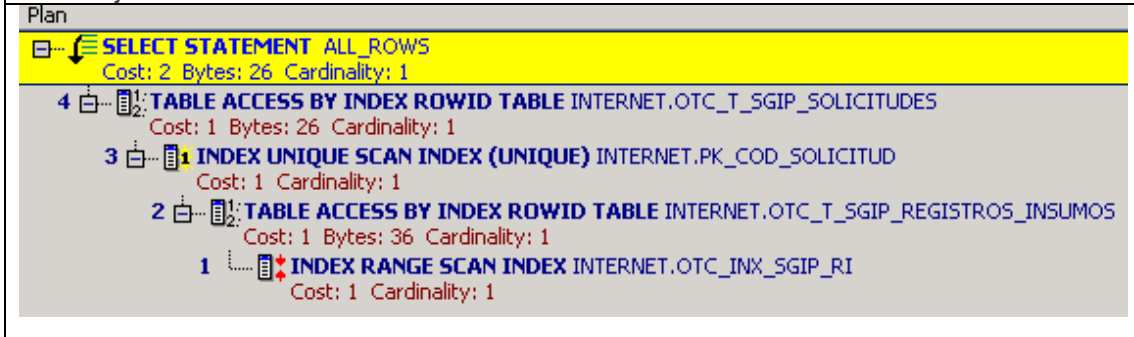


Script: OTC T SP UP ASIGNADO NOTAS

```

/* Formatted on 2012/06/01 20:31 (Formatter Plus v4.8.8) */
SELECT cod_solicitud, req_ppm
  --INTO SOL,REQ
FROM   otc_t_sgip_solicitudes
WHERE  cod_solicitud =
       (SELECT cod_solicitud
        FROM otc_t_sgip_registros_insumos
        WHERE cod_solicitud > 0
              AND num_serie = '8959'
              AND fecha_devolucion_real IS NULL);
  
```

Plan de Ejecución



TKPROF DE EJECUCIÓN

No Aplica

INFORME DE PRUEBAS

Detallado en el apartado manual de usuario

FECHA	09 de Enero 2012
BASES DE DATOS	SCLPREECU
PARÁMETROS	Definidos en WEB.config
TIEMPO DE EJECUCIÓN	N/A
RESULTADOS	Las pruebas internas de funcionalidad están identificadas en la sección manual de usuario.
OBSERVACION	

6. PASO A PRODUCCIÓN

DESCRIPCIÓN DE PASO A PRODUCCIÓN

DEPENDENCIA	Requerimiento/OTT
1	N/A

PAQUETIZACION	Requerimientos/OTT's
1	N/A

Base de datos

SECUENCIA	TAREA	SERVIDOR	AFECTA EL SERVICIO?	TIEMPO ESTIMADO
1	De los fuentes que se encuentran en la siguiente ruta: svn://otecelblcarreta/automatizaciones_qtp/ desarrollo/solicitud_cambios/ Sistema_Inventarios/scripts	otecelblcarreta	NO	5 min
2	Ejecutar los scripts: - Creación de tablas - OTC_PKG_SGIP	MISC3	NO	5 min
3	Si no hay errores pasar al procedimiento " Servidor WEB "	MISC3	NO	5 min

Servidor WEB

SECUENCIA	TAREA	SERVIDOR	AFECTA EL SERVICIO?	TIEMPO ESTIMADO
1	De los fuentes que se encuentran en la siguiente ruta: svn://otecelblcarreta/automatizaciones_qtp/ desarrollo/solicitud_cambios/ Sistema_Inventarios	otecelblcarreta	NO	5 min
2	Copiarlos y pegarlos en el servidor, en la dirección D:\ServiciosWeb Notas: Esta dirección se encuentra definido como directorio para servicios WEB del servidor.	10.5.1.121	NO	5 min
3	Configurar el archivo WEB.config con la información de la base de datos en producción.	10.5.1.121	NO	5 min

	<addname="ecu_reportes" connectionString="Provider=MSDAORA.1; userid=ecu_reportes;data source=sclprecu;password=xxxx;" />			
4	Crear el Webservice en la consola administrativa del IIS.	10.5.1.121	NO	10 min
5	Realizar el test de browser	10.5.1.121	NO	5 min
6	Verificar el acceso al sistema ingresando a la URL del servidor.	10.5.1.121	NO	5 min

REVERSO**Base de datos**

SECUENCIA	TAREA	SERVIDOR	AFECTA EL SERVICIO?	TIEMPO ESTIMADO
1	De los fuentes que se encuentran en la siguiente ruta: svn://otecelblcarreta/automatizaciones_qtp/desarrollo/solicitud_cambios/Sistema_Inventarios/scripts		No	5 min
2.	Ejecutar el script: Rollback		No	5 min

Servidor WEB

SECUENCIA	TAREA	SERVIDOR	AFECTA EL SERVICIO?	TIEMPO ESTIMADO
1	Detener el servidor WEB (sistema_inventarios)		No	5 min
2.	Eliminar directorio		No	5 min

7. MANUAL DE USUARIO

- Ingreso al sistema: URL http://localhost/sistema_inventarios/Login.aspx

The screenshot shows the login interface for the 'Sistema de Gestión Inventario de Pruebas'. It features the Telefónica logo in the top left. The title 'Sistema de Gestión Inventario de Pruebas' is centered. Below the title, there are two input fields: 'Usuario:' with the value 'ROZUMBA' and 'Contraseña:'. A blue 'Ingresar' button is positioned below the password field. At the bottom right, there is a link that says 'Cambiar Contraseña >>'. The background is a dark blue gradient on the left and a green grid pattern on the right.

- Acceso correcto:

Bienvenido al Sistema: **ROZUMBA**
Fecha actual: 23/01/2012 16:23:37
Versión: 11.12.27

- ▼ Menú de Opciones
 - ▼ Administración
 - ▼ Parametros Generales
 - ▷ Crear usuarios
 - ▷ Crear perfiles
 - ▷ Mantenedor de insumos
 - ▷ Modificar usuarios
 - ▼ Reportes
 - ▼ Información
 - ▷ Auditoria por insumo
 - ▷ Existencia actual de insumos
 - ▷ Devoluciones por usuario
 - ▷ Insumos asignados
 - ▷ Situación actual del inventario
 - ▷ Insumos activos
 - ▼ Servicios
 - ▼ Acciones
 - ▷ Asignar insumos
 - ▷ Consultar solicitudes
 - ▷ Crear solicitudes
 - ▷ Crear insumos
 - ▷ Procesar solicitudes

- Acceso incorrecto:

Sistema de Gestión
Inventario de Pruebas

Usuario:

Contraseña:

[Cambiar Contraseña >>](#)

Usuario no permitido

- Cambio de clave incorrecta:

Sistema de Gestión
Inventario de Pruebas

Usuario:

Contraseña:

[Cambiar Contraseña >>](#)

Ingrese contraseña asignada por el administrador

- Cambio de clave correcta:

Sistema de Gestión Inventario de Pruebas

Usuario:
Contraseña:

[Cambiar Contraseña >>](#)

Cambio de contraseña exitosa

- Perfil – Solicitante

Bienvenido al Sistema: **ROZUMBA**
Fecha actual: 24/01/2012 15:55:50
Versión: 11.12.27

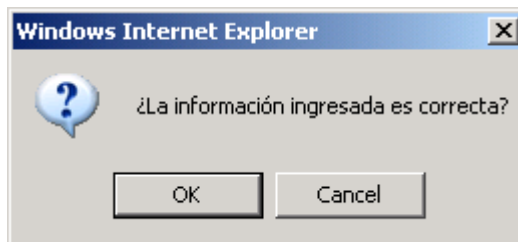
- ▼ *Menú de Opciones*
 - ▼ *Administración*
 - ▼ *Parametros Generales*
 - ▷ *Crear usuarios*
 - ▷ *Crear perfiles*
 - ▷ *Mantenedor de insumos*
 - ▷ *Modificar usuarios*
 - ▼ *Reportes*
 - ▼ *Información*
 - ▷ *Auditoria por insumo*
 - ▷ *Existencia actual de insumos*
 - ▷ *Devoluciones por usuario*
 - ▷ *Insumos asignados*
 - ▷ *Situación actual del inventario*
 - ▷ *Insumos activos*
 - ▼ *Servicios*
 - ▼ *Acciones*
 - ▷ *Asignar insumos*
 - ▷ *Consultar solicitudes*
 - ▷ *Crear solicitudes*
 - ▷ *Crear insumos*
 - ▷ *Procesar solicitudes*

- *Perfil – Coordinador / Administrador*

Bienvenido al Sistema: ROZUMBA
Fecha actual: 24/01/2012 15:56:25
Versión: 11.12.27

- ▼ Menú de Opciones
 - ▼ Servicios
 - ▼ Acciones
 - ▷ Consultar solicitudes
 - ▷ Crear solicitudes
 - ▼ Reportes
 - ▼ Información
 - ▷ Recursos asignados por usuario

- Administración
 - Crear usuarios: Ingresar los datos del nuevo usuario, luego presionar crear usuario. El sistema pedirá confirmación.



Crear Usuarios

<i>Perfil:</i>	<input type="text" value="Administrador"/>
<i>Nombre:</i>	<input type="text" value="MILTON"/>
<i>Apellido:</i>	<input type="text" value="PANCHI"/>
<i>Correo:</i>	<input type="text" value="MILTON.PANCHI@TELEFONICA.COM"/>
<i>Usuario:</i>	<input type="text" value="MIPANCHI"/>
<i>Proveedor:</i>	<input type="text" value="INTERNO"/>
<i>Contacto móvil:</i>	<input type="text" value="098539572"/>
<i>Contacto fijo:</i>	<input type="text" value="2227700"/>
<i>Extensión:</i>	<input type="text" value="3411"/>

Crear Usuario

Notificación del sistema:

Usuario creado correctamente

- Crear perfiles:

Crear Perfiles

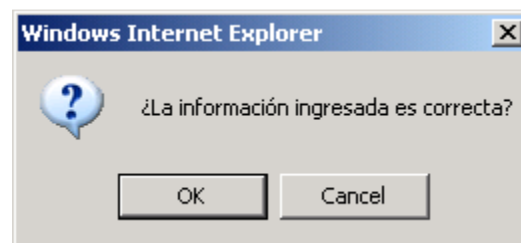
Descripción:

Crear Perfil

Despliegue de datos

CODIGO	DESCRIPCION
1	Administrador
2	Solicitante
3	Coordinador
4	AdminExpress
5	AdminOtt
6	AdminFuncion
7	Proveedor Sonda
8	Proveedor Bayteq
9	AdminReportes
10	AdminPlataforma
11	Facturacion
12	Jefe

- Mantenedor de insumos:
 - Eliminar Insumo – No existe: Ingresar una serie y presionar eliminar, el sistema pedirá confirmación:



Eliminar Insumos

Serie:

Eliminar Insumo

Actualizar Insumos

Serie(s):

1.- Caja:

2.- Ambiente:

3.- Estado:

Alerta: Insumo no existe en el sistema

- Eliminar Insumo – Si existe:

Eliminar Insumos

Serie:

Eliminar Insumo

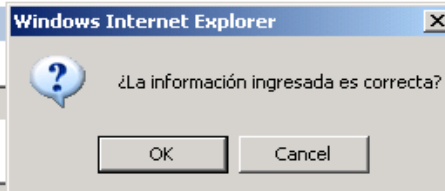
Actualizar Insumos

Serie(s):

1.- Caja:

2.- Ambiente:

3.- Estado:

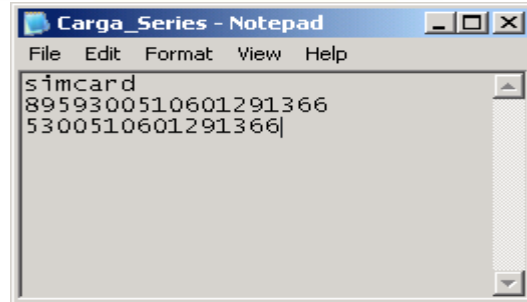


Cargar Insumos

Notificación del sistema:

Insumo eliminado correctamente

- Actualizar Insumo: El usuario debe cargar un archivo con las series a modificar.



```
Carga_Series - Notepad
File Edit Format View Help
simcard
8959300510601291366
5300510601291366
```

Actualizar Insumos

Serie(s):

1.- Caja:

2.- Ambiente:

3.- Estado:

Las series son cargadas a una tabla temporal, muestra al usuario la cantidad de ítems que fueron cargados correctamente.

Posteriormente se habilitan las opciones disponibles.

Actualizar Insumos

Serie(s):

1.- Caja:

2.- Ambiente:

3.- Estado:

[2] Insumos cargados correctamente, actualizar la información según opciones disponibles

- Actualizar caja – ambiente – estado:

Actualizar Insumos

Serie(s):

1.- Caja:

2.- Ambiente:

3.- Estado:

Windows Internet Explorer

¿La información ingresada es correcta?

OK Cancel

Cargar Insumos

Actualizar: 1

Actualizar: 2

Actualizar: 3

[2] Insumos cargados correctamente, actualizar la información según opciones disponibles

Notificación del sistema:

Comando ejecutado correctamente

- *Modificar usuarios: El criterio de búsqueda inicial es por apellido. También puede ingresar una letra y se presentará una tabla con las coincidencias.*

Consultar Usuarios

Apellido:

Despliegue de datos

CODIGO	NOMBRE	APELLIDO	CORREO
2	ELIZABETH	ROCHA	0
26	MARIA	YUGCHA	0
33	ANABELL	HEREDIA	0
34	LOURDES	HUILCATOMA	0
38	BYRON	HERRERA	0
39	MILTON	PANCHI	0
45	MILTON	PANCHI	MILTON.PANCHI@TELEFONICA.COM

- *El usuario ingresa el código del usuario para que se despliegue la información que será sujeta a modificación.*

Modificar Datos del Usuario

Código:

Nombre:

Apellido:

Correo:

Proveedor:

Estado:

Perfil:

Contacto móvil:

Contacto fijo:

Extensión:

- El usuario realiza un TAB desde el teclado o un clic con el Mouse y el sistema busca la información en la base de datos.

Despliegue de datos

CODIGO	NOMBRE	APELLIDO	CORREO	RED	PROVEEDOR	ESTADO	PERFIL	MOVIL	FIJO	EXT
38	BYRON	HERRERA	0	BYHERRERA	INTERNO	ACTIVO	ADMINREPORTES			

Modificar Datos del Usuario

Código:

Nombre:

Apellido:

Correo:

Proveedor:

Estado:

Perfil:

Contacto móvil:

Contacto fijo:

Extensión:

- El sistema presenta las opciones disponibles. Actualizar estado

Modificar Datos del Usuario

Código:

Nombre:

Apellido:

Correo:

Proveedor:

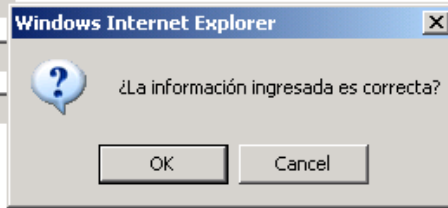
Estado:

Perfil:

Contacto móvil:

Contacto fijo:

Extensión:



Despliegue de datos

CODIGO	NOMBRE	APELLIDO	CORREO	RED	PROVEEDOR	ESTADO	PERFIL	MOVIL	FIJO	EXT
38	BYRON	HERRERA	0	BYHERRERA	INTERNO	SUSPENDIDO	ADMINREPORTES			

- El sistema presenta las opciones disponibles. Actualizar perfil

Despliegue de datos

CODIGO	NOMBRE	APELLIDO	CORREO	RED	PROVEEDOR	ESTADO	PERFIL	MOVIL	FIJO	EXT
2	ELIZABETH	ROCHA	0	NAE48728	SONDA	ACTIVO	SOLICITANTE			

Modificar Datos del Usuario

Código:

Nombre:

Apellido:

Correo:

Proveedor:

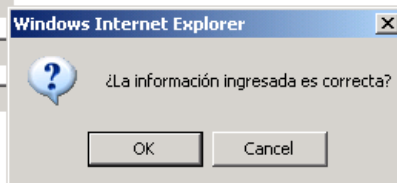
Estado:

Perfil:

Contacto móvil:

Contacto fijo:

Extensión:



Despliegue de datos

CODIGO	NOMBRE	APELLIDO	CORREO	RED	PROVEEDOR	ESTADO	PERFIL	MOVIL	FIJO	EXT
2	ELIZABETH	ROCHA	0	NAE48728	SONDA	ACTIVO	ADMINISTRADOR			

- El sistema presenta las opciones disponibles. Actualizar usuario

Despliegue de datos

CODIGO	NOMBRE	APELLIDO	CORREO	RED	PROVEEDOR	ESTADO	PERFIL	MOVIL	FIJO	EXT
2	ELIZABETH	ROCHA	0	NAE48728	SONDA	ACTIVO	ADMINISTRADOR			

Modificar Datos del Usuario

Código:

Nombre:

Apellido:

Correo:

Proveedor:

Estado:

Perfil:

Contacto móvil:

Contacto fijo:

Extensión:

Despliegue de datos

CODIGO	NOMBRE	APELLIDO	CORREO	RED	PROVEEDOR	ESTADO	PERFIL	MOVIL	FIJO	EXT
2	ELIZABETH	ROCHA	ELIZABETH.ROCHA@SONDA.COM.EC	NAE48728	SONDA	ACTIVO	ADMINISTRADOR	89234833	20505050	107

- Reportes
 - Auditoría por insumo: Se ingresa una serie y luego botón exportar. Si hay datos presenta un archivo Excel, de lo contrario emite el mensaje respectivo.

Auditoría por insumo

Número de serie:

Auditoría por insumo

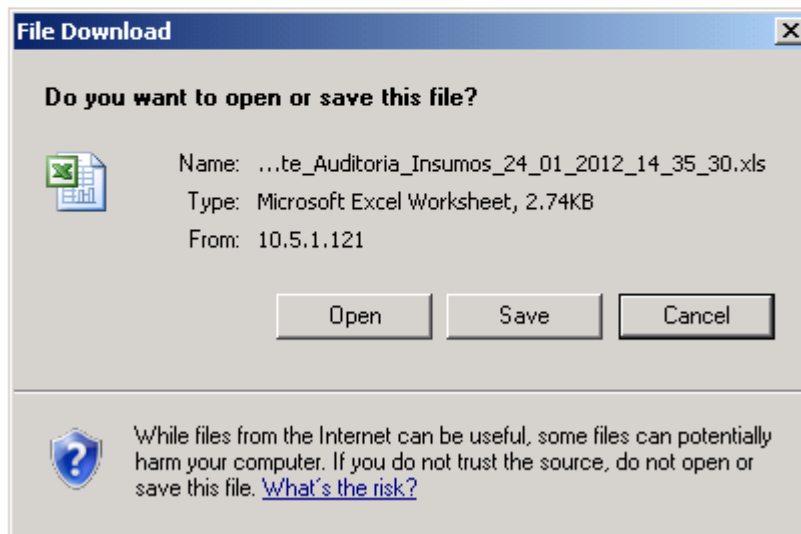
Número de serie: 1234

Exportar datos

Alerta: No existen registros: Consultar_Transacciones**Auditoría por insumo**

Número de serie: 740000103066099

Exportar datos



INSUMO	SERIE	CAJA	CENTRAL	AMBIENTE	FECHA_CREACION	ESTADO	MODELO
TERMINAL	740000103066099'	0	&NBSP;'	PROYECTOS	12/26/2011 16:48:35	DAÑADO	SAMSUNG
TERMINAL	740000103066099'	0	&NBSP;'	&NBSP;'	12/26/2011 16:48:35	DAÑADO	SAMSUNG
TERMINAL	740000103066099'	0	&NBSP;'	&NBSP;'	12/26/2011 16:48:35	ASIGNADO	SAMSUNG
TERMINAL	740000103066099'	0	&NBSP;'	&NBSP;'	12/26/2011 16:48:35	ASIGNADO	SAMSUNG
TERMINAL	740000103066099'	10	&NBSP;'	&NBSP;'	12/26/2011 16:48:35	NUEVO	SAMSUNG
TERMINAL	740000103066099'	10	&NBSP;'	&NBSP;'	12/26/2011 16:48:35	NUEVO	SAMSUNG

- o Existencia actual de insumos: Presenta una tabla con la información existente.

Stock actual de insumos

Despliegue de datos

TIPO	AMBIENTE	ESTADO	CANTIDAD
SIMCARDS	PROYECTOS	NUEVO	9
TERMINALES	PROYECTOS	DAÑADO	1

- o Devoluciones por usuario: Seleccionar un solicitante y fecha

Reporte devoluciones

Usuario:

Ir

Fecha:

<div style="text-align: center;"> nov diciembre de ene 2011 </div>						
dom	lun	mar	mié	jue	vie	sáb
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

Reporte devoluciones

Usuario: Fecha:

DIGITAL	SERIE	SOLICITUD	USUARIO	FECHA_DEVOLUCION	REQ_PPM
NO	8959300510601291366	2	EDUARDO LOAIZA	27/12/11	108999

Reporte devoluciones

Usuario: Fecha: **Alerta: No existen registros: Consultar_Devoluciones**

- Insumos asignados - Usuario:

Insumos asignados

Usuario: Solicitud: Todo: Actual:

Despliegue de datos

TIPO_INSUMO	NUM_SERIE	FISICA	CENTRAL	SOLICITUD	AMBIENTE	USUARIO	MODELO	TECNOLOGIA	ASIGNACION	DEVOLUCION	REQ_PPM
SIMCARD	8959300510601291366	NO	1	4	PROYECTOS	ROBERTO ZUMBA	NA	NA	29/12/11	28/12/11	101999
SIMCARD	8959300510607738042	SI	1	9	PROYECTOS	ROBERTO ZUMBA	NA	NA	17/01/12	17/01/12	113389
SIMCARD	8959300510607739035	SI	1	9	PROYECTOS	ROBERTO ZUMBA	NA	NA	17/01/12	17/01/12	113389
TERMINAL	740006105060099	SI	0	9	PROYECTOS	ROBERTO ZUMBA	Samsung	GSM	17/01/12	17/01/12	113389
SIMCARD	8959300510600397917	SI	1	9	PROYECTOS	ROBERTO ZUMBA	NA	NA	17/01/12	17/01/12	113389

- Insumos asignados - Solicitud:

Insumos asignados

Usuario:
 Solicitud:
 Todo:
 Actual:

Despliegue de datos

TIPO_INSUMO	NUM_SERIE	FISICA	CENTRAL	SOLICITUD	AMBIENTE	USUARIO	MODELO	TECNOLOGIA	ASIGNACION	DEVOLUCION	REQ_PPM
SIMCARD	8959300510601291366'	NO	1	4	PROYECTOS	ROBERTO ZUMBA	NA	NA	29/12/11	28/12/11	101999

- Insumos asignados – perfil solicitante. No está permitido las opciones por usuario - todo:

Insumos asignados

Usuario:
 Solicitud:
 Todo:
 Actual:

Despliegue de datos

TIPO_INSUMO	NUM_SERIE	FISICA	CENTRAL	SOLICITUD	AMBIENTE	USUARIO	MODELO	TECNOLOGIA	ASIGNACION	DEVOLUCION	REQ_PPM
SIMCARD	8959300510601291366'	NO	1	4	PROYECTOS	ROBERTO ZUMBA	NA	NA	29/12/11	28/12/11	101999
SIMCARD	8919400310497739038'	NO	1	7	PROYECTOS	ROBERTO ZUMBA	NA	NA	24/01/12	11/01/12	101999
SIMCARD	8959300510600397917'	SI	1	9	PROYECTOS	ROBERTO ZUMBA	NA	NA	17/01/12	17/01/12	113389
TERMINAL	740000103059099'	SI	0	9	PROYECTOS	ROBERTO ZUMBA	Samsung	GSM	17/01/12	17/01/12	113389
SIMCARD	8959300510607739038'	SI	1	9	PROYECTOS	ROBERTO ZUMBA	NA	NA	17/01/12	17/01/12	113389

- Insumos asignados – Todo:

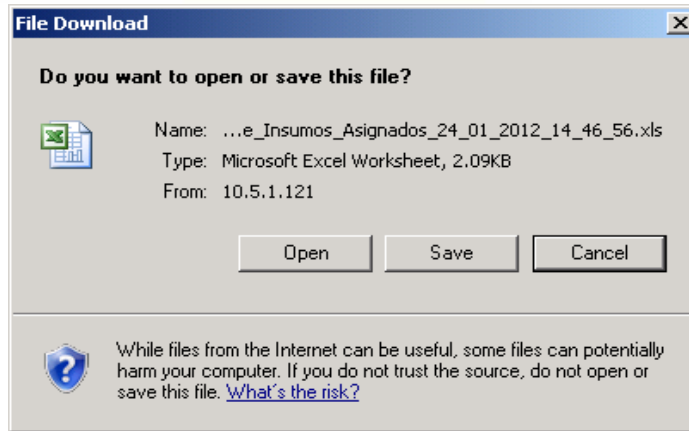
Insumos asignados

Usuario:
 Solicitud:
 Todo:
 Actual:

Despliegue de datos

TIPO_INSUMO	NUM_SERIE	FISICA	CENTRAL	SOLICITUD	AMBIENTE	USUARIO	MODELO	TECNOLOGIA	ASIGNACION	DEVOLUCION	REQ_PPM
SIMCARD	8959300510601291366'	NO	1	4	PROYECTOS	ROBERTO ZUMBA	NA	NA	29/12/11	28/12/11	101999
SIMCARD	8959300510607739042'	SI	1	9	PROYECTOS	ROBERTO ZUMBA	NA	NA	17/01/12	17/01/12	113389
SIMCARD	8959300510607739038'	SI	1	9	PROYECTOS	ROBERTO ZUMBA	NA	NA	17/01/12	17/01/12	113389
TERMINAL	740000103060099'	SI	0	9	PROYECTOS	ROBERTO ZUMBA	Samsung	GSM	17/01/12	17/01/12	113389
SIMCARD	8959300510600397917'	SI	1	9	PROYECTOS	ROBERTO ZUMBA	NA	NA	17/01/12	17/01/12	113389

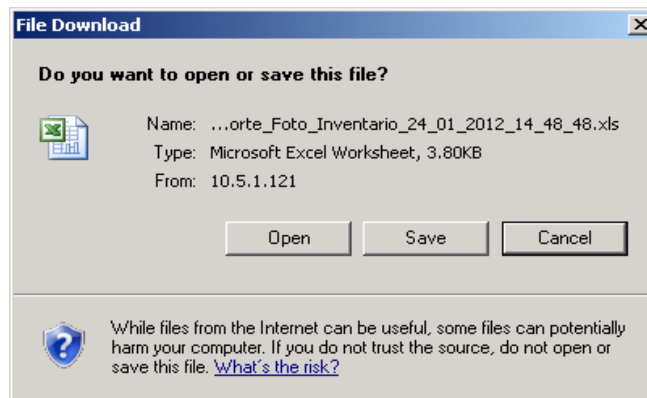
- Insumos asignados – Exportar



- Situación actual del inventario:

Situación actual inventario

Exportar datos



TIPO_INSUMO	NUM_SERIE	ID_CAJA	NUM_IMSI	AMBIENTE	FECHA_CREACION	SITUACION	MODELO
SIMCARD	8959300510600399418'	1	740000102956904'	PROYECTOS	01/04/2012 17:41	NUEVO	NA
SIMCARD	8959300510600399400'	1	740000102956903'	PROYECTOS	01/04/2012 17:41	NUEVO	NA
SIMCARD	8959300510600668952'	1	740000103003858'	PROYECTOS	01/04/2012 17:41	NUEVO	NA
TERMINAL	740000103066099'	20	&NBSP;'	PROYECTOS	12/26/2011 16:48:35	DANADO	SAMSUNG
SIMCARD	8959300510601291374'	1	740000103066100'	PROYECTOS	12/26/2011 16:49:02	NUEVO	NA
SIMCARD	8959300510601291366'	2	740000103066099'	PROYECTOS	12/26/2011 16:49:02	NUEVO	NA
SIMCARD	8959300510601291358'	1	740000103066098'	PROYECTOS	12/26/2011 16:49:02	NUEVO	NA
SIMCARD	8959300510600397917'	1	740000103066201'	PROYECTOS	12/28/2011 14:47:11	NUEVO	NA
SIMCARD	8959300510607739038'	1	740000103066098'	PROYECTOS	12/28/2011 14:47:11	NUEVO	NA
SIMCARD	8959300510607738642'	1	740000103066099'	PROYECTOS	12/28/2011 14:47:11	NUEVO	NA

- Insumos activos: Presenta una tabla con las series activas, si no existe información presenta alerta respectiva.

Líneas activas en: Preproducción

Despliegue de datos

CAJA	SERIE	ABONADO	CELULAR	CLIENTE	CUENTA	CODIGO	PLAN	CICLO	ALTA	IMEI
1	8959300310501758089	18714661	84532357	19370728	11207030	CO6	Multicolor 20	2	09/01/12	115612978301848

Líneas activas en: Preproducción

Despliegue de datos

Alerta: No existen registros: Consultar_Insumos_Activos

- Servicios
 - Asignar insumos: El usuario selecciona un archivo de texto, el cual contiene la lista de series que serán asignadas a un código de solicitud.

Asignar Recursos

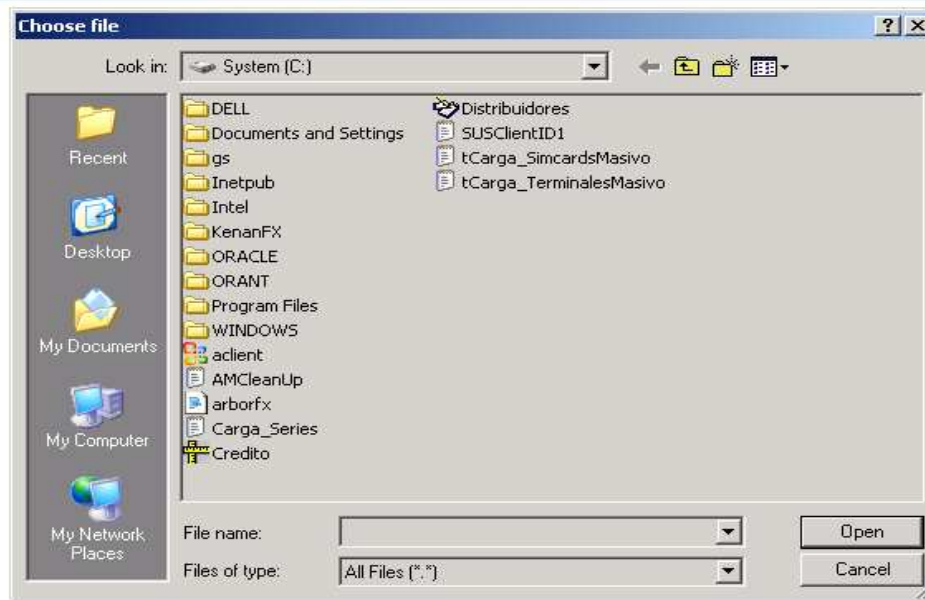
Serie(s): Browse...

Solicitud: 1

Entrega: ALFONSO URQUIZO

Asignar Recursos

Serie(s):
Solicitud:
Entrega:

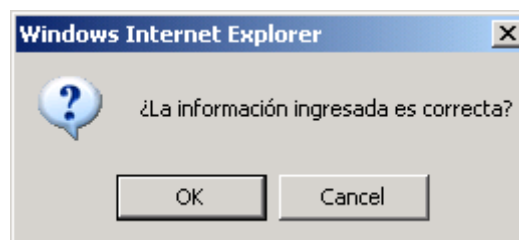
**Asignar Recursos**

Serie(s): C:\Carga_Series.txt Browse...

Solicitud: 1

Entrega: ALFONSO URQUIZO

Seleccionar cargar insumos y el sistema solicita confirmación.



Asignar Recursos

Serie(s): Browse...

Solicitud: Administrador

Entrega:

[1] Insumos cargados correctamente, actualizar la información según opciones disponibles

El usuario debe seleccionar el código de solicitud, si la solicitud seleccionada es 1 entonces es recepción, si es un código diferente de 1 es una asignación de insumos a un solicitante.

Asignar Recursos

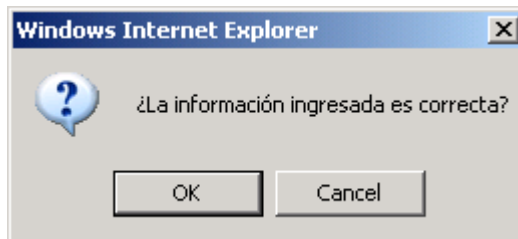
Serie(s): Browse...

Solicitud: Solicitante

Recibe:

[1] Insumos cargados correctamente, actualizar la información según opciones disponibles

Seleccionar: Procesar, solicita confirmación.



Asignar Recursos

Serie(s):

Solicitud: Solicitante

Recibe:

Proceso realizado correctamente

La solicitud, cambia automáticamente de estado a procesado y ya no aparecerá en el como.

Asignar Recursos

Serie(s):

Solicitud: Administrador

Entrega:

- Consultar solicitudes: presenta una lista con las solicitudes pendientes de procesar, si no hay registros presenta el mensaje respectivo.

Consultar Solicitudes

Fecha Inicio: **Fecha Fin:**

Solicitud:

Despliegue de datos

SOLICITUD	PEDIDO	USUARIO	NOMBRE	RECURSO	CANTIDAD	JUSTIFICACION	REQ_PPM	DEVOLUCION	AMBIENTE	FISICA
10	24/01/12	NAE03685	BETHY-CRUZ	Simcards activadas	2	test	101999	26/01/12	PRODUCCION	NO

Consultar Solicitudes

Fecha Inicio: Ir Fecha Fin: Ir

Buscar Solicitudes

Solicitud: Buscar Solicitud

Despliegue de datos

Alerta: No existen registros: Consultar_Solicitudes

Las consultas por solicitud están permitidas para el perfil de coordinador.

Consultar Solicitudes

Fecha Inicio: Ir Fecha Fin: Ir

Buscar Solicitudes

Solicitud: Buscar Solicitud

Despliegue de datos

SOLICITUD	PEDIDO	USUARIO	ENTREGA	RECURSO	CANTIDAD	JUSTIFICACION	REQ_PPM	DEVOLUCION	AMBIENTE	FISICA	ESTADO	DIAS_PRORROGA	NOTAS
4	27/12/11	ROZUMBIA	29/12/11	Solicitudes activadas	2	test	101999	28/12/11	PREPRODUCCION	NO	PROCESADO	0	

Las opciones de consulta por fecha están permitidas para el perfil de solicitante.

No se despliega la información de las solicitudes creadas por otros solicitantes y la opción de consulta por número de solicitud no está habilitada para el perfil solicitante; únicamente para el perfil coordinador.

Consultar Solicitudes

Fecha Inicio: Ir Fecha Fin: Ir

Buscar Solicitudes

Solicitud: Buscar Solicitud

Despliegue de datos

SOLICITUD	PEDIDO	USUARIO	ENTREGA	RECURSO	CANTIDAD	JUSTIFICACION	REQ_PPM	DEVOLUCION	AMBIENTE	FISICA	ESTADO	DIAS_PRORROGA	NOTAS
2	27/12/11	SIMUNDA	27/12/11	Solicitudes movidas	2	test	101999	09/01/12	PRODUCCION	NO	INTRODUCIDO	0	
3	27/12/11	SIMUNDA	27/12/11	Solicitudes	2000	test	101999	28/12/11		SI	INTRODUCIDO	0	no hay acciones terminales
4	27/12/11	SIMUNDA	29/12/11	Solicitudes activadas	2	test	101999	28/12/11	PREPRODUCCION	NO	INTRODUCIDO	0	
5	27/12/11	SIMUNDA	04/01/12	Solicitudes activadas	2000	test	101999	28/12/11	PREPRODUCCION	NO	INTRODUCIDO	0	prueba de negocio
7	10/01/12	SOZUMBA	24/01/12	Solicitudes activadas	3		101999	11/01/12	PREPRODUCCION	NO	PROCESADO	0	
8	17/01/12	SOZUMBA	17/01/12	Solicitudes activadas	2	Prueba del nuevo sistema de producción	113350	17/01/12	PRODUCCION	SI	PROCESADO	0	

- Crear solicitudes Perfil Solicitante:

Crear una solicitud de recursos

Recurso:

Físico:

Cantidad:

Justificación:

Req PPM:

Devolución:

Ambiente:

- Crear solicitudes Perfil Coordinador:

Crear una solicitud de recursos

Usuario:

Recurso:

Físico:

Cantidad:

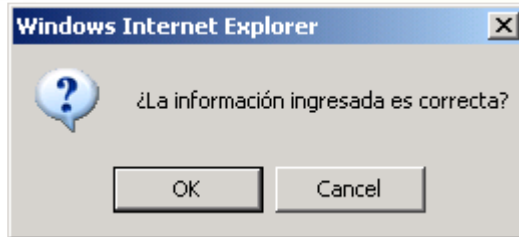
Justificación:

Req PPM:

Devolución:

Ambiente:

Solicita confirmación.



Crear una solicitud de recursos

Usuario:

Recurso:

Físico:

Cantidad:

Justificación:

Req PPM:

Devolución:

Ambiente:

Solicitud [10] creada correctamente

- *Crear insumos:* El usuario debe cargar un archivo plano con la información a grabar. Existen dos formatos: Simcards y Terminales

Crear Insumos

Series:

Caja:

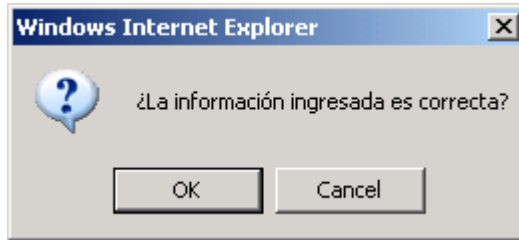
Tecnología:

Ambiente:

Notas:

Las series tipo simcards a cargar deben tener un formato específico "8959xxxx" si no está en este formato la series no serán cargadas para luego grabarse en la base de datos.

Solicita confirmación.



- Procesar solicitudes: El usuario debe ingresar el parámetro de entrada solicitud

Procesar Solicitudes

Solicitud:

Acciones:

Prorroga: días:

Apertura:

Notas:

Solamente se puede actualizar una solicitud a negado o procesado cuando el estado inicial es nuevo. De lo contrario presenta la siguiente alerta.

Procesar Solicitudes

Solicitud:

Acciones:

Prorroga: días:

Apertura:

Notas:

Alerta: Solicitud no existe o está en un estado no permitido

Para aplicar una prórroga a una solicitud, el estado permitido debe ser procesado, de lo contrario se presenta la siguiente alerta.

Procesar Solicitudes

Solicitud:

Acciones:

Prorroga: días:

Apertura:

Notas:

Alerta: Solicitud no permitida para aplicar prórroga

Una solicitud puede ser abierta, pero el estado permitido para esta acción debe ser procesado / negado, de lo contrario se presenta la siguiente alerta.

Procesar Solicitudes

Solicitud:

Acciones:

Prorroga: días:

Apertura:

Notas:

Alerta: Solicitud no permitida para reapertura

Negar una solicitud.

Procesar Solicitudes

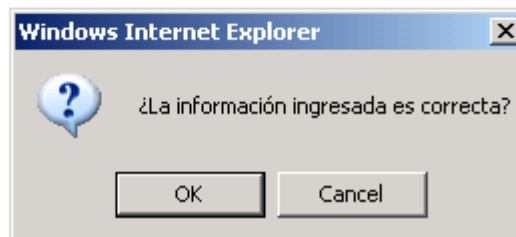
Solicitud:

Acciones:

Prorroga: días:

Apertura:

Notas:



Procesar Solicitudes

Solicitud:

Acciones:

Prorroga: días:

Apertura:

Notas:

Solicitud procesada correctamente

8. [ANEXOS](#)

Anexo 6.

Visión Alcance Sistema de Gestión Inventario de Pruebas



Detalle Funcional de Usuario (DFU)

**SISTEMA DE GESTION Y CONTROL DE RECURSOS FÍSICOS Y
LÓGICOS PARA LA GERENCIA DE CONSTRUCCION AREA PRUEBAS
DE OTECEL S.A**

Fecha: 13-Julio-2011

OTECCEL S.A

La información contenida en este documento está catalogada como reservada y de uso interno para Otecel S.A.
Su divulgación y/o reproducción total o parcial requiere autorización expresa y escrita por parte de Otecel S.A.

CONTROL DE CAMBIOS DEL RF

VERSIÓN	FECHA	CAMBIOS RESPECTO DE LA VERSIÓN ANTERIOR	PREPARADO POR	REVISADO POR	APROBADO POR
1.0	13-08-2011	Versión inicial, previa revisión del usuario	Roberto Zumba		
2.0	18-09-2011	Versión revisada con el usuario interno	Roberto Zumba	Gabriel Carvajal	
3.0	15-10-2011	Versión corregida con el usuario interno	Roberto Zumba	Gabriel Carvajal	
4.0	22-11-2011	Especificaciones funcionales aceptadas por el usuario interno.	Roberto Zumba		Efren Zambrano

Tabla de Contenidos

1	Situación Actual.....	1
2	Objetivos.....	1
3	Detalle Funcional Situación Esperada	1
4	Impacto en Procesos de Negocio.....	6
5	Aplicaciones Afectadas.....	6
6	Plantillas	6
7	Anexos	6

1 Situación Actual

En la Gerencia de Construcción Área Pruebas TI hay un proceso de asignación o préstamo de recursos físicos y lógicos como son; SIMCARDS y TELEFONOS CDMA / GSM a los colaboradores del área, y el registro de esta información se realiza de forma manual y haciendo uso de una hoja de cálculo Excel.

La cantidad de artículos que se maneja en esta área es de aproximadamente 6000 y la cantidad de colaboradores que utilizan estos recursos llegan a 30 personas.

La información que se maneja en las hojas de cálculo Excel, ocasionan inconvenientes relacionados con la organización de los recursos, pérdida de información, no se tiene información histórica de los movimientos generados ya que estos insumos se reutilizan periódicamente entre los colaboradores del área.

Otro problema además de la información es el control en este proceso manual el mismo que ha ocasionado perdidas o mal uso de los recursos.

No existen reportes que permitan realizar una gestión adecuada para los usuarios de la Gerencia de Construcción.

2 Objetivos

Diseñar, desarrollar y poner en producción un aplicativo que apoye a la gestión y control de los recursos físicos y lógicos asignados a los colaboradores de la Gerencia de Construcción Área Pruebas en OTECEL. S.A

3 Detalle Funcional Situación Esperada

Diseñar, desarrollar e implementar el sistema SGCIP (Sistema de Gestión y Control del Inventario de Pruebas)

El SGCIP será una aplicación que funcionará en un entorno WEB que permitirá administrar, controlar y clasificar la información de los recursos físicos y/o lógicos asignados al Área de Pruebas para el uso de los usuarios internos. Esta aplicación ofrecerá soporte a las actividades definidas en el procedimiento interno: "PRTISS0601 Gestión y Control del Inventario de Pruebas.pdf", del cual se cubrirán las siguientes:

- Verificar disponibilidad de inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde obtiene información del estado o situación de los insumos a su cargo.
- Controlar la asignación de inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde realiza actividades de control como son; fechas vencidas, devolución de

recursos tecnológicos obsoletos, compra de nuevos recursos, validación del estado funcional de los recursos tecnológicos.

- Verificar la rotación y reutilización de inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde se realiza una actualización periódica de los recursos tecnológicos registrados en el sistema.
- Registrar situación de inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde se actualiza masivamente o uno por uno los recursos tecnológicos a su cargo en base a parámetros como estado, usuario anterior, fecha de entrega.
- Devolver el Inventario.- proceso que es realizado por un recurso humano con el perfil de usuario solicitante del inventario. En donde realiza la entrega física o virtual de los recursos tecnológicos a él asignados en un plazo establecido.
- Agrupar Inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde toma el inventario físico de los equipos celulares separándolos por tecnología de comunicación y toma el inventario físico de simcards separándolos por los ambientes del sistema comercial.
- Confirmar disponibilidad en sistemas.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde verifica el estado de los recursos tecnológicos en las bases de datos de los sistemas vigentes en la compañía.
- Identificar estado y funcionalidad del inventario.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde verifica uno por uno el estado funcional de los insumos tecnológicos cuando realiza el proceso de recepción de recursos.
- Registrar estado del inventario de Terminales.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde se reciben los recursos tecnológicos y se realizan las actas respectivas en donde los usuarios dejan por escrito la entrega - recepción de los recursos tecnológicos.
- Registrar estado del inventario de Simcards.- proceso que es realizado por un recurso humano con el perfil de administrador del inventario. En donde se reciben los recursos tecnológicos y se realizan las actas respectivas en donde los usuarios dejan por escrito la entrega - recepción de los recursos tecnológicos.

CONSIDERACIONES:

Para el diseño del aplicativo se debe referir al procedimiento llamado: “PRTISS0601 Gestión y Control del Inventario de Pruebas”

El detalle de las especificaciones funcionales se encuentra en el CAPITULO 3 del proyecto de tesis.

3.1.1 Descripción de Formatos de Reportes o Cubos

Descripción del Formato de Requerimiento de Información																							
<p>TITULO <PARAMETROS DE ENTRADA> CRITERIOS</p>																							
<p><DESPLIEGUE DE DATOS> TABLA DATOS</p> <table border="1"> <thead> <tr> <th>Columna1</th> <th>Columna2</th> <th>Columna3</th> <th>Columna3</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>				Columna1	Columna2	Columna3	Columna3																
Columna1	Columna2	Columna3	Columna3																				
<p><NOTIFICACIONES></p>																							

4 Impacto en Procesos de Negocio

Código	Nombre Documento	Impacto	Observaciones
1	GF Gestión de Facturación		
2	GC Gestión de Clientes		
3	CB Gestión de Cobranza		
4	TI Tecnología de Información y Comunicaciones.	x	
5	DH Desarrollo Humano Organizacional		

5 Aplicaciones Afectadas

El Consultor conjuntamente con el Usuario y los Administradores identifican las Aplicaciones y Plataformas impactadas.

Código	Nombre Aplicación / Plataforma	Impacto	Observaciones
1	SCL - Data Guard (Stand By)	X	
2	Sistema de Gestión de Crédito		
3	Switchs, Routers y Firewalls		
4	Telefonía Pública		
5	Otras	X	Misceláneos (MISC3)

6 Plantillas

Plantilla	Nombre Archivo	Observaciones
CONFIGURACIÓN		
Servicios Suplementarios		
Planes		
REQUERIMIENTOS DE INFORMACIÓN		
Plantilla de Reportes		
Plantilla de Cubos		

7 Anexos

Contenido	Nombre Archivo	Observaciones
Propuesta de solución técnica para el desarrollo del sistema de inventarios	Capitulo 4	Sección del documento general de la investigación.